SPRING

Simulation of Processes in Groundwater

Manual

Professional Water Systems Modelling Software.

- Groundwater flow modelling
- Contaminant transport modelling
- Heat transport modelling

SPRING – Brought to you by the Engineering Consulting Firm delta h Ltd.

1 Contents

2 Introduction to SPRING 5

Welcome to SPRING 5, the latest version of the software system for groundwater modelling proven for more than 30 years.

From underground mining to water supply, groundwater remediation and geothermics: SPRING's innovative software system provides optimal support for all your planning and consulting projects.

The software system based on a development by the Ruhr University of Bochum allows you to calculate three-dimensional groundwater flow, surface water, heat and contaminant transport models.

SPRING 5 in combination with the expert knowledge and experience of our staff is the basis for making your project a success.

SPRING = <u>S</u>imulation of <u>P</u>rocesses <u>i</u>n <u>G</u>roundwater

2.1 Highlights in SPRING 5

In SPRING 5, a completely new kernel of software architecture (parallel and scalable) has been developed:

■ Application up to 100 million nodes
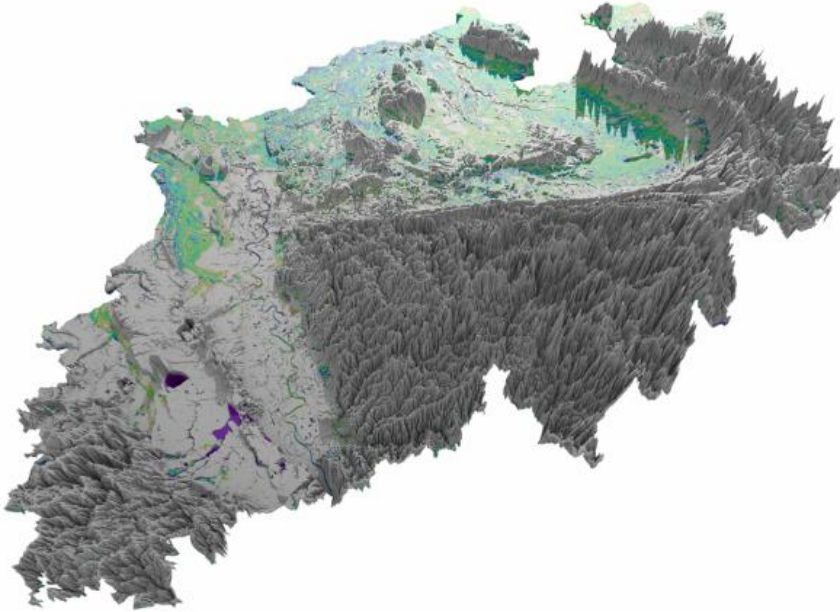
Fig. 1: Application up to 100 million nodes

■ Quickly open, close and save large models
■ Interpolation algorithms are now optimized for scalability with multithreading
■ New handling of transient data with convenient processing options and consideration of mesh refinement (optimization of large amounts of data due to database usage in SPRING 5)

Improved structural and contour management has been introduced so that model meshes can be set up faster than ever before:

■ Program-based integration of compulsory geometries into existing meshes (excavation pits, sheet piles, wells ...)
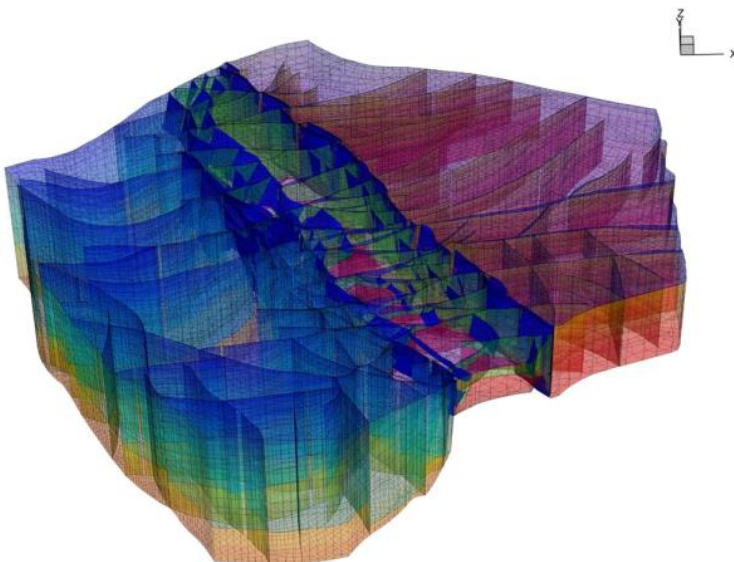
Fig. 2: Easy adoption of geometries in existing meshes

- Extended structure handling (direct interpolation on structures, ...)
- Revised water systems (algorithms for the derivation of the water level of water courses, ...)
- Optimized contour handling (removal of overlaps, ...)

Complex geology can easily be integrated into SPRING 5:
- geometry optimized vertical refinement algorithm
- Easy integration of existing GOCAD models
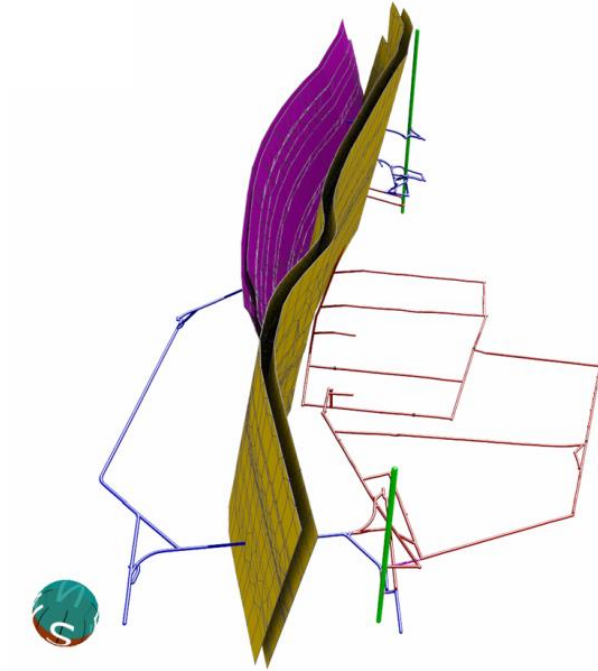
Fig. 3: Installation of geology in SPRING5

Furthermore, the following improvements have been implemented in SPRING 5:
- groundwater recharge calculation tool for the calculation of daily recharge rates
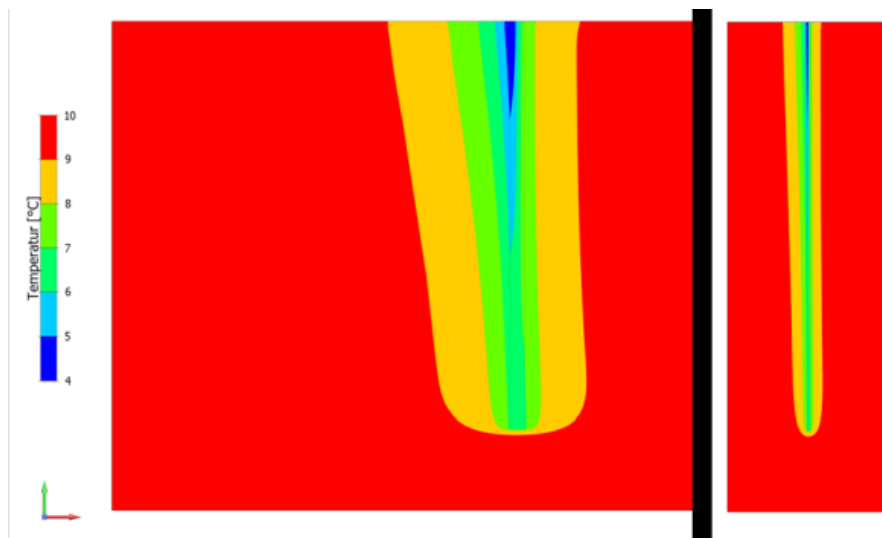
Fig. 4: energy balance dependent depiction of geothermal conditions

- Conditioning for the stable solution of large systems of equations
- Full three-dimensional visualization

2.2 SPRING - General Performance Features

- Finite Element method
- Any combination of 1D, 2D and 3D elements possible
- Interactive mesh generation with automatic mesh generation and mesh refining
- Outflow layers in 3D models
- Statistical calculation of fractures
- Exact representation of geological structures by means of collapsing layers
- Including tunnel structures in the calculation
- Steady-state flow / transient flow calculations
- Flow lines, calculation of path lines and travel times
- Saturated / unsaturated flow, including calculation of the unconfined surface
- Heat transport
- Density-dependent flow
- Contaminant transport, taking into account dispersion / diffusion, adsorption, production, disintegration and decay
- Transport of nuclide chains
- Reactive contaminant transport
- Simulation of flooding using box models
- Automatic calibration methods (gradient method and inverse modelling)
- Drainage flow (leakage, with limited amounts)
- Special boundary conditions for geothermal calculations
- Recharge calculation on the basis of a climatic soil water balance
- Interface with the ArcInfo software system
- Interactive control of the program sequence via SPRING'S graphic interface or via batch files
- Extensive re-editing options of the displayed results with the display of bitmap (TIFF) files and output in DXF format
- Automatic interpolation option of transient boundary conditions between the pre-defined time steps
- Output option for model data and calculation results in a freely selectable format as ASCII file.
- Automatic calibration methods
- Drainage balance, relative to the water body mesh
- Statistical calculation of the basin area

2.3 System Requirements

- Windows 10, Windows 7,
- 2 GB RAM (16 GB recommended)
- At least 300 MB available hard disk storage
- Screen resolution 1280 x 1024 (1920 x 1080 recommended)

2.4 Installation

- Close all open applications.
- Insert the product CD or DVD into the appropriate drive of the computer.
- As soon as the Autoplay screen appears, follow the instructions.

Note: If the Autoplay screen is not loaded automatically, select "Computer" on the desktop, select the drive icon and finally the file "setup.exe".

The number of computers in which the software may be installed can be found in the license agreement which forms part of the software.

If you have received your SPRING program version via download, start the installation with the file "set-up.exe".

2.5 SPRING Update

Delta h makes software updates available for maintenance customers at intervals. You can download these updates conveniently via the delta h Download Centre. This requires an active Internet connection.

- In SPRING 5, click at top right on the *Help → Find Updates* button.
- Select Update and download it to your computer.
- Follow the enclosed installation instructions.

2.6 Program Start

To start the software, click on *Start – Programs – SPRING* or, if you have created a desktop link, double-click on the corresponding icon:

2.7 Licensing

Please note that after installation in accordance with the license conditions, STRING must be activated for the respective workstation. Please follow the on-screen instructions.

2.8 From Nature to the Model
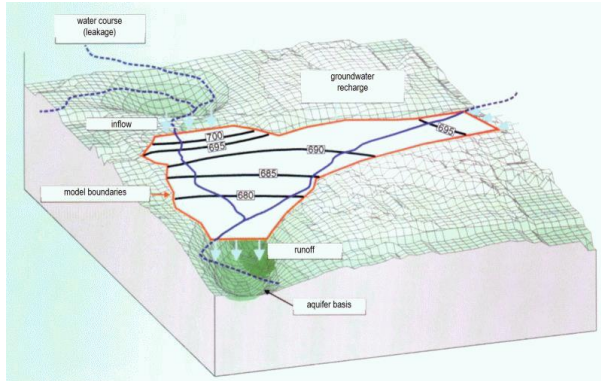
Fig. 5: From nature to the model

A groundwater model must be able to simulate real hydrogeological situations. To this end, first raw data such as groundwater measuring points and water level measurements, geological and topographic maps and other water management documentation must be collected, and the problem under investigation must be formulated.

From this follow the selection of the model boundaries and the aquifers to be included.

This is where you start working with SPRING.
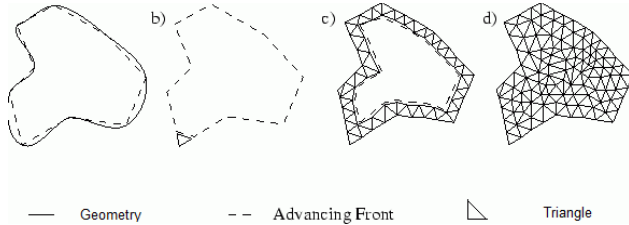
Fig. 6: From the area boundary to the model area

All raw data are read into SPRING by data export or digitisation and saved as structures. The structures are mesh-independent. The data can already be visualised at this point and subjected to a first inspection.

Systematic building of the model using SPRING can be represented by the following schematic sequence:

Data become **Structures,**

mesh-relevant **Structures** become **Contours**,

Contours become **Nodes**,

Nodes become **Elements**.

The nodes and elements of the finished FE mesh are assigned **Attributes**.

2.8.1 Structures

Structural data are points, lines and polygons, which, in addition to their geometric point information may also carry data information (attributes). A distinction is made between:

- Point structures (zero-dimensional geometry)
- Line structures (one-dimensional geometry)
- Area structures (two-dimensional geometry)
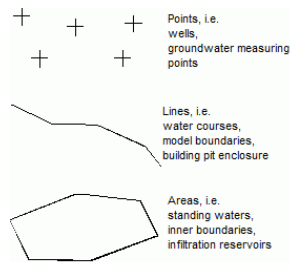
Fig. 7: Structural types

These structural types and the allocation of the corresponding information allow all data required for a groundwater model to be described, such as:

- Point structures:

   Coordinates of pumping wells and disposal wells

   Coordinates of groundwater measuring points plus groundwater levels

- Line structures

   Polygon lines of the course of water course systems, including drainage levels or leakage coefficients or ex-/infiltration volumes

   Building pit enclosure (sheet piling walls or seal walls)

   Ditches, tunnels, navigation channels

   Sewage and drainage systems

- Area structures

   Areas of different groundwater recharge

   Areas of different soil characteristics or hydraulic parameters

   Standing water (lakes or ponds) with and without overflow

   Infiltration reservoir with amount of infiltration

The structures ( ) can be created directly on-screen (e.g. on the basis of a topographic map) or imported from existing ASCII files (e.g. *.txt, *.csv, ARC/INFO generate format or *.dxf). The structure menu also offers extensive modification options.

2.8.2 Contours

Contours ( ) define all geometric constraint points and lines required when generating an FE mesh. They include, for example:

- the outline of the area to be modeled,
- the geometric information required in the area, such as

   the course of water courses,

   the location of lakes and

   the exact location of wells.
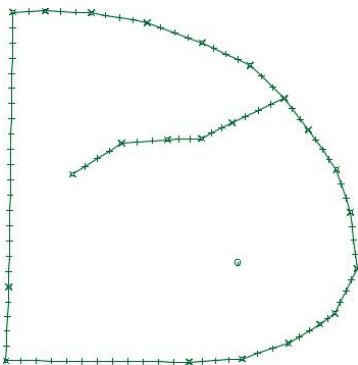
Fig. 8: Contours of a model

These constraint points and lines must be present as mesh nodes or element edges in the later FE mesh and combined in the contour.

The basis of the contour are usually the structures and the required geometric constraint points during generation of the mesh The required components can be added by simply selecting them from the structural data (*Contour → New → From Structure*).

A basic distinction is made between

- I contours,
- k contours
- p contours

I contours are contour lines on which later on element edges will be placed.

k contours are the same lines as I contours, but are used for generating 1D fracture elements. They are generated using the program KLUFTI and contain, apart from the generation parameters provided for I contours, also the thickness and opening width of the 1D fractures. Moreover, k contours are also treated separately in boundary node generation.

p contours are contour lines on which later on mesh nodes will be placed.

The contours must be provided with parameters for further mesh generation (e.g. well parameters for exponential node generation around well nodes, maximum distances or division points for long line contours). To do this, the Contour menu offers numerous options for editing and modification.

2.8.3 Nodes

Once all contours have been adjusted, the nodes will be generated. In general, the following sequence of node generation is recommended.

- Node generation on the contours and perpendicular to the contours using 1-2 sections of the boundary node generator.
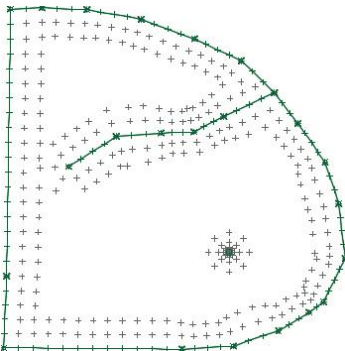
Fig. 9: Boundary node generator

- Generation of regular meshes in areas without contours
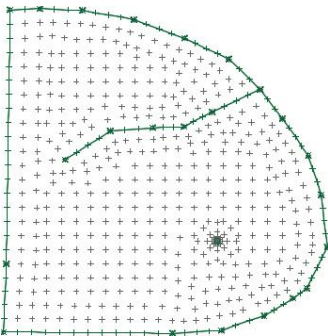
Fig. 10: Boundary node generator

- Generation of additional nodes by hand in the transition areas.

Manual movement and deletion of nodes optimize node distribution.

2.8.4 Elements

Once node generation has been completed in areas or globally, these nodes can be grouped (in areas or globally) by means of elements. The *triangulation algorithm* (advancing front) used in SPRING groups the nodes, first all triangular nodes. Then (if triangles and quadrangles are to be generated) suitable triangles are combined to quadrangles, in order to reduce the number of elements and thus the amount of calculation and memory required.

After creating the elements, there are occasionally areas in which an unfavourable mesh structure becomes only visible after grouping the nodes. These areas can be re-edited by local mesh editing (deletion of areas and regeneration of nodes and elements, combining nodes or parts of quadrangles. SPRING's Mesh menu offers various methods of editing the mesh geometry.
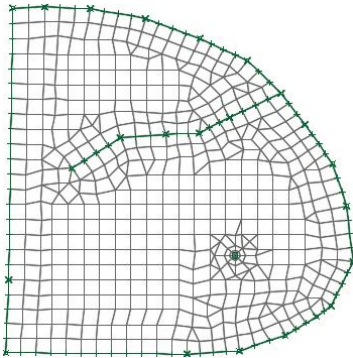
Fig. 11: Finished FE mesh

2.8.5 Attributes

Attributes contain all the data information that goes beyond the point information, such as material data, boundary conditions and auxiliary data.

Once the generation of the mesh geometry is complete and has been inspected, the nodes and elements can be assigned attributes. On the basis of the structural data, data can be assigned in one of two ways:

- Direct assignment: Node or element centres sufficiently close to the structure point containing the data information are assigned its value.
- Assignment by interpolation: On the basis of "isolated" structure data points, values will be interpolated for the nodes or element centres of the corresponding structures lying in-between.

Example:

The structure of a water course only contains water level information at the level measuring points. Any number of additional nodes containing not a water level information are located between these data points. These nodes are assigned linearly interpolated values via *Assignment by interpolation*.

Apart from assignment via structures, attributes can also be assigned to nodes or elements directly or statistically either individually or by area.

All possible attributes are listed in chapter "Description of the data types" (→ Data structure of the groundwater model).

The structure of a horizontal groundwater model is described in detail in chapter "Structure of a 2D model".

2.8.6 Calculation modules

SPRING uses for the various calculations the following program modules:

- Module for the model checking: DADIA
- Module for the calibration: EICHEN
- Module for the interpolation: INTERPOL
- Module of the steady-state flow computation: GEONEU
- Module of the transient flow computation: INSTAT
- Module for the steady-state/transient flow and/or heat- or mass transport calculation and module for the inverse calibration: SITRA

- Module for the plot generation: PLOGEO
- Module for the data export: NACHLAUF

The calculation modules of the flow calculation can be started in the SPRING user interface via the menu item *Calculation* and are described in detail in the corresponding chapter "Calculations" (p. **241**).

2.9 SPRING User Interface

The SPRING 4 user interface consists mainly of 5 areas - menu bar, toolbars, main window, status bar and project manager. The range of functions of all areas depends on the currently open project type. If commands in a menu, in a view or at a certain point in the project are not available, they are shown in grey and are inactive or are not shown.
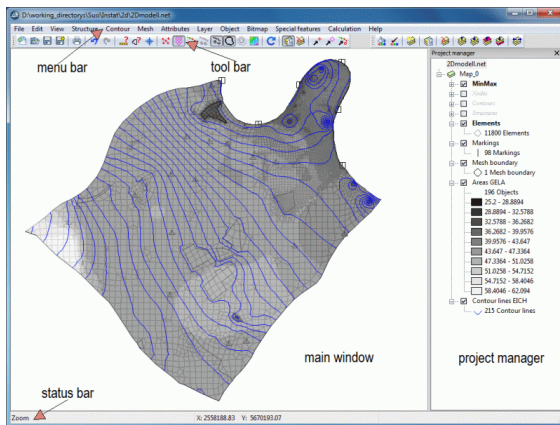
Fig. 12: SPRING User Interface

2.9.1 Menu bar

The menu bar gives access to most SPRING features:
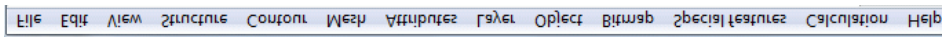
Upon selection of some menu commands, a submenu is displayed containing a list of additional options. Such menu commands have an arrow on the right of the command name pointing to the right.

With some menu commands, a dialog box is displayed in which you have to enter more information for the command to be executed. These commands are marked with three suspension marks (…) following the name of the command.

Some menu commands can be displayed directly by pressing a special key abbreviation or key combination (Ctrl+KEY). For commands that have been assigned a key abbreviation, the key abbreviation or key combination is shown on the right of the command. Thus, for example, the last action can be undone by pressing the key combination Ctrl+Z, and the last menu command can be repeated by pressing F12.

Menu bar:

- **File** The File menu is used to Open/Save/Finish projects, for data import/export, for adding/superimposing plots, for creating plots or for printing.
- **Edit** The Edit menu is used to Restore/Undo editing steps, for project editing and for loading program options.
- **View** The View menu is used to Display/Hide layers/toolbars/3D displays and for creating vertical cuts.
- **Structure** The Structure menu is used for creating and editing the structural data, for example reading in digital data on a line structure.
- **Contour** The Contour menu is used for editing the contour elements.
- **Mesh** The Mesh menu is used for creating and editing meshes, for example mesh generator.
- **Attributes** The Attributes menu is used for editing data attributes, for example copying or calculating data types.

- **Layer** The Layer menu is used for managing the graphic layers.
- **Object** The Object menu is used for editing graphic objects, for example duplicating or moving individual objects.
- **Bitmap** The Bitmap menu is used for editing or saving graphic files, for example tiff files.
- **Special features** The Special features menu gives access to special program functions for water meshes, transient program parameters, fractures or tunnels.
- **Calculation** The Calculation menu is used to start SPRING's calculation tools.
- **Help** The Help menu gives access to the program help and is used to start updates.

2.9.2 Toolbars

The toolbars (see figure below) contain buttons for quick access to the most important menu commands. The name of the command is displayed when the mouse pointer is placed over the button.

The buttons of the toolbars are arranged in groups. In the *View → Toolbars* dialog box, you can set which toolbar groups you want to have displayed. These settings apply to the current view and will be saved when the program is finished. The toolbar groups can also be dragged to a different position of the graphic user interface by clicking on the drag point on the left-hand side of the toolbar and moving it to the desired position while holding down the left mouse button. When the toolbar is dropped in the central area of the application window, it is undocked and shown freely floating; when the toolbar is dropped at the edge of the application window, the toolbar will be docked to this position. The following figure shows the toolbar "standard":

The other toolbars are shown in chapter "SPRING-menus - View - Toolbars" (p. **95**) .

2.9.3 Status bar

The three-part status bar is located at the bottom edge of the application window (see figure). It shows the following information: (left) current input mode, (centre) world coordinates of the mouse pointer, (right) information on menu commands and command buttons in the toolbars when placing the cursor over them.

2.9.4 Main window

The main window is used for creating, editing and visualizing element meshes and plots and is SPRING's centrepiece.

2.9.5 Project manager

The project manager is started by clicking on *View → Additional windows → Project manager* or by selecting the symbol . It visualizes the arrangement of open Maps/Layers of the active project and their legends and allows easy manipulation of the Maps/Layers via drag&drop.

A project is a collection of maps, layers, objects and other resources related to one another via a tree structure. At their specific level, they can be added in any number. An exception is the mesh editing mode, in which only one map is allowed.

For example, the figure below contains a project named Witten.net with Map_0, which in turn is subdivided into various layers, for example mesh boundary, marks or elements. The GELA (ground level surface) layer is folded up and shows the legend of the isoareas shown in the main window.

Only a single project can be open in the project manager. Upon creating a new project or opening an existing one, the project currently open in the project window will be replaced with the new one.

After modifying a project, the project must be saved (by selecting the command *File → Save* or the symbol).

The commands for Maps/Layers are available in the menu bar under the same names. The project manager makes some functions available in context menus (to open them, click on the appropriate entry) (see figure).
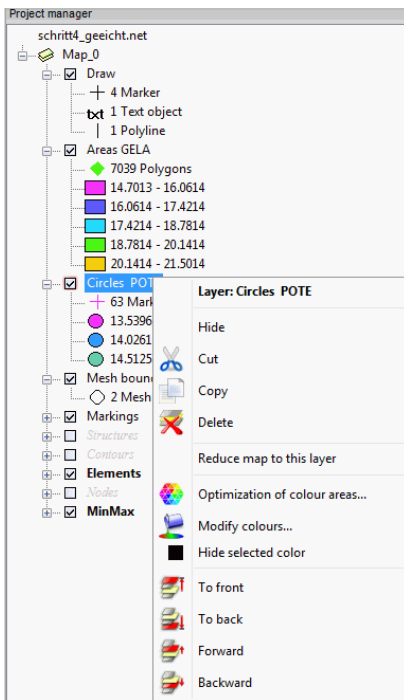
Fig. 13: Project manager in the mode model file

In the project manager, Maps/Layers can be dragged to a different position (drag&drop). A layer can be dragged to a different map, but cannot be moved to a different layer. To delete a selected layer, press the "Del" key.

Maps and layers are "semantic folders" which do not constitute a logical file grouping and do not have to coincide with hierarchical organisation of files on your hard disk.

Layers can correspond to physical files (e.g. tiff maps) located in your file system and are directly related to them. Layers of this type are displayed in the project window with their path specification.

In the mode plot file, it is possible to edit the object of the individual layers (right mouse button). For example, it is possible to remove all polylines of a layer simultaneously, or the values of the object properties (defined before in the menu options) are assigned to all of the polylines.
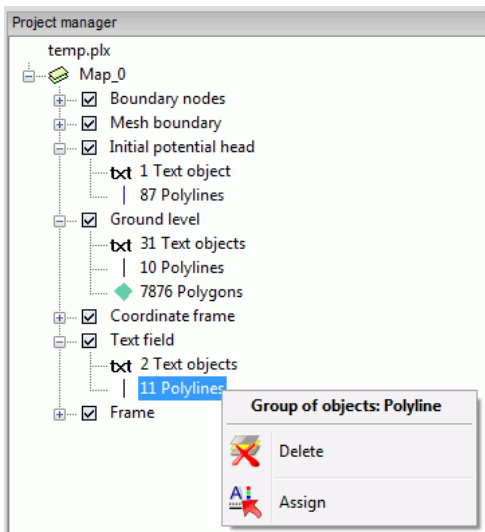
Fig. 14: Project manager in the mode plot file

2.9.6 Project information

The project information window provides an overview of the most important project properties.
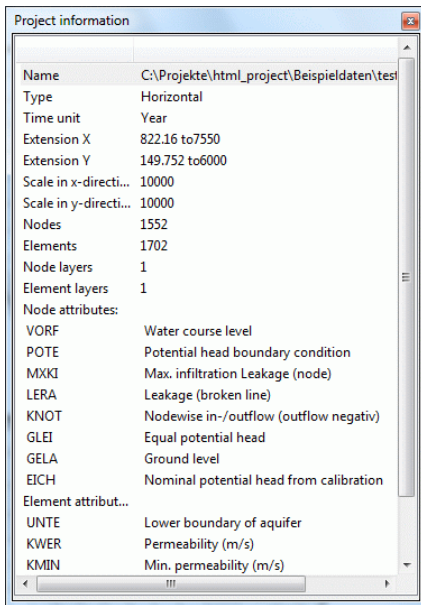
Fig. 15: Project information

2.9.7 3D View

The 3D view displays the element mesh opened in the mesh editing mode three-dimensionally.

Navigation:

- **Zoom**   Mouse wheel
- **Rotation**   Left mouse button + Movement of the mouse in the x or y direction
- **Shift**   Middle mouse button + Movement of the mouse in the x or y direction
- **z scaling**   Ctrl + mouse wheel
- **Reset**   Middle mouse button
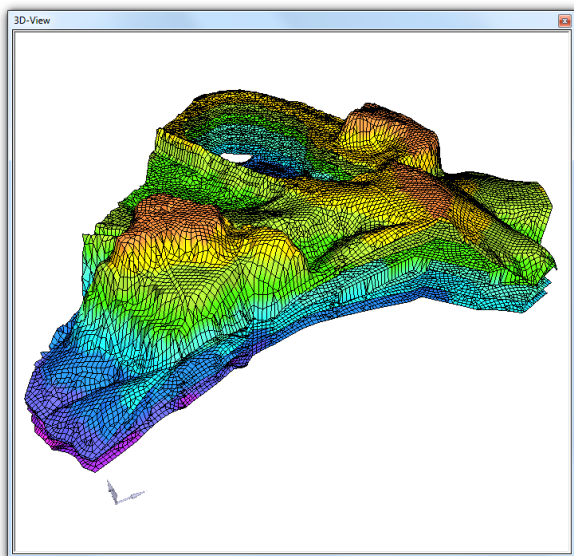
The presentation can be changed with the help of the buttons of the toolbar in the main window (below).

The element mesh, the element areas, the 1D-fractures, the mesh of 2D-fractures or the areas of the 2D-fractures can be displayed. The current 3D view can be exported as a bitmap.

2.9.8 Vertical section

In the mesh editing mode, the vertical section window shows a vertical section through the model with a straight line between two selected points.
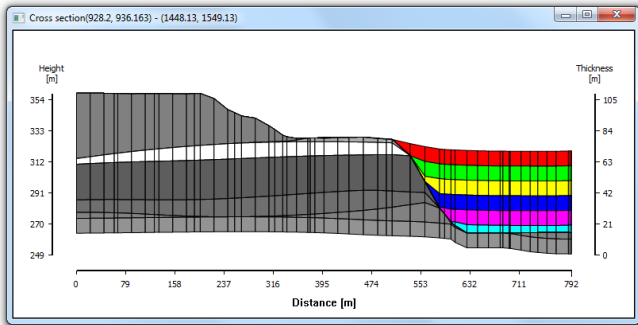
Fig. 16: Vertical section

2.9.9 Settings

SPRING 4 can be configured by means of various settings and options. The settings are described in detail in chapter *SPRING menus → Edit → Options.*

2.10 Help

The complete documentation of the current SPRING version is available under Help. It is a browser-based system which can be accessed via *Help → Manual*. The help topics are updated with the update of the software, so that the latest information is always available.

Important: The Online Help contains all information from the printed user manuals and additional information not available in printed form. A complete PDF version of the printed manual is contained on the product CD or DVD. The latest version is always available to download on our website spring.delta-h.de under Download&Support: Manual SPRING 4.1 (EN).

2.11 Support

If you have concluded a maintenance agreement, you can contact your personal telephone support if you have questions on the installation or detect any product errors. Further information and contact addresses can be found in the Support area of the delta h website.

> www.delta-h.de   Telephone: 0049-2302-91406-0
>
> email: dh@delta-h.de

The Support area of the delta h website also contains information on troubleshooting and allows you to access the Support database, list of frequently asked questions (FAQ) and many more sources of information. Information on training courses/conferences for delta h products is also found on the website.

3 Data Structure of a Groundwater Model

This chapter describes the file structure and data types that constitute the basis of a groundwater model.

3.1 Structure of the model files

The input data for a groundwater model are found in the input file *name.net*. In addition, a three-dimensional model requires the 3d-file *name.3d.*

The **.net*-file contains all data for a two-dimensional model, but only the data of the uppermost layer for a three-dimensional model. The information on the layers beneath are found in the **.3d*-file.

In most cases, these two input files are created and changed in the interface of SPRING. They have ASCII-format and can be edited with any text editor.
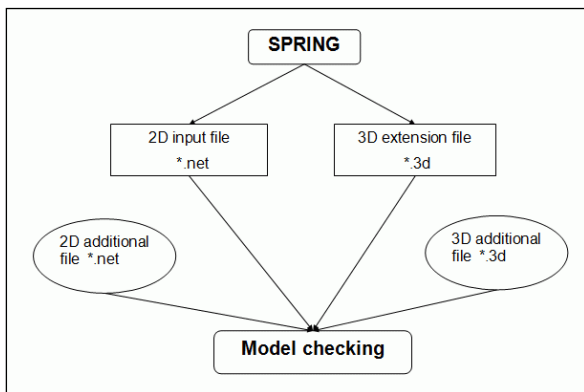
Fig. 17: File processing in SPRING

The contents of the formatted input files are described in the following chapters.

3.1.1 Structure of the 3D extension file **.3d*

This file has no comment lines. If the user wants to change the numbering of nodes in deeper layers, this should be done first. A blank line follows. After that, the boundary of the three-dimensional area has to be defined, followed by a blank line. Then either the parameters for subdivision into layers or the exact z-coordinates follow. The layer definition is completed by the subdivision data type without any parameter. After another blank line, the other input data follow.

3.1.2 Format description of the model and 3D file

Both the geometric data types (node and element incidences) and all attributes are saved in the **.net* – and the **.3d* - files during the modelling in SPRING.

The user should not edit the node and element incidences manually, as this can lead to significant errors in the FE mesh. Should it even be necessary to edit the attributes in the model file by hand, the following stand-ard format applies to almost all of the attributes:

Column 1 – 14: Assignment by the following possible formats:
- Max. 14-digit real number as floating point, e.g. „14.99" or „-1.234"
- E-format: e.g. „1.e-7" or „-2.5e+10"

The values must always have a decimal **point**.

Column 15 – 16: Special characters
- For some attributes, it is possible to put a special character (e.g. F, /, *) in column 15 or 16, which affects the processing of the values. These signs are explained in the individual data types.

Column 17 – 79:

- Node or element numbers in the format I6, after the number follows a comma or a hyphen for a number sequence from … to.
- The element and node numbers are entered in the Fortran format I6, as long as no number is > 1 million. In order to manage larger numbers the keyword **NUM** must be specified with an attached decimal number in the fourth line of the *. net* file. **INUM** values of less than six are ignored. The following formats are then changed from I6 to **I***INUM*.

Column 80:

- If the assignment of a type of data extends to more than one line, a "+" is set in column 80, which connects the lines together.

Each attribute block is completed with a blank line.

Between the individual data packets comment lines can also be added. These begin with a 'C' and three blanks ('C___'). Comment lines are ignored during the model checking.

Some data types (such as flooding or van Genuchten parameters) are stored in a specific format. As this happens automatically in SPRING, there is no need for the user to deal with this.

3.1.3 Additional files

Each of the two model files can be added during the model checking by specifying an additional file or by reading in directly in SPRING with *File → Import → Submesh*…

In these additional files can be any number of data which are treated as if they are in the model files. If an attribute exists twice (e.g. a node with two withdrawal amounts) the last value overrides the previous one.

With the additional files you can save disk space in several model types. All non-changing information (such as mesh data, permeability, etc.) is stored in the model file. For different simulations the withdrawal amounts or the boundary conditions can be stored in auxiliary files.

The additional files have exactly the same structure as the corresponding "master file" (*.net* or *. 3d*).

Attention! When opening a model SPRING reads only the main input files, but never the additional ones. They have to be read explicitly, as described above, or are specified in the dialog of the model checking.

3.1.4 Dimensioning file *.dim

Once a model was created and saved for the first time, SPRING automatically creates a dimensioning file *name.dim*.

The sizing is carried out "pseudo-variable". That is, a maximum number of nodes and elements is pre-defined. The larger this number, the longer take the necessary initialization routines in the model checking, even if the current project specifically requires only a fraction of the pre-defined maximum number of nodes and elements.

A clever sizing reduces the computational effort during the model checking. Is the selected sizing too small, the corresponding dimensioning variable in the dimensioning file can be changed with a simple text editor.

Example:

```
IDIM1     270000  = IDIM2
IDIM2     270000  maximum node-/element number
IDIM3     100000  max. number boundary nodes, fractures and single data
IDIM4     100000  max. number MATE,MARK,KTXT,ETXT
IDIM5         14  max. number 3d-layer
IDIM6    5110000  temporary field, 3d: ca. idim1 * 51 , 2d: ca. idim1 * 20
```

Meaning of the dimensioning variables:

Field	Meaning
IDIM1	maximum number of nodes/elements
IDIM2	maximum allowed node or element number
IDIM3	maximum number of boundary nodes, fractures and point-data values (i.e. potentials (POTE), in- and outflow at nodes (KNOT), concentrations (1KON), ...)
IDIM4	maximum number of values for material data (MATE), markings (MARK), node and element texts (KTXT, ETXT)
IDIM5	maximum number of 3D layers
IDIM6	dimension of a temporary field used during the optimization algorithm (3D models about IDIM1*51, 2D models about IDIM1*20)

3.1.5 Structure and format of the structure data file

In the structure data file, data (geometric or attributes) are stored with their respective X- and Y-coordinates. These data has to be assigned to the nodes or elements during the modelling in SPRING.

The structure data file has the name "name.str". It begins with three comment lines with any text, followed by a blank line and any number of structural data sets. During the modelling in SPRING the contour data blocks are stored later in the project's *. str* file. As this is done automatically, the contour data should NOT be edited manually by the user.

Example:
```
Project XY
Structure file 01.01.2010
SPRING Version 4.0

DATA | l | @ 3 1 0 @ Model boundary
     1   607.2647 1460.3211
     2   257.1429 1220.7641
     3   259.4463  718.6157
     . . .

FLAE | f | @ 7 2 0 @ groundwater recharge
     1   402.2591 1172.3920   0.0000
     2   393.0454  813.0565
     . . .

VORF | l | @ 2 1 0 @ water course
     1   922.8350 1285.2602
     2   904.4075 1197.7298  22.0100
     3   913.6213 1100.9851
     . . .
```

The structural data are separated by blank lines. A structural data set consists of a header line followed by any number of rows of data.

The header line has the format:
```
KENN | x | @ plot parameters @ text
```
That means:

- **KENN** (4 characters) represents the identifier of the structural dataset. It is required to assign the data to nodes or elements of the mesh. You can use the usual data identifiers and the identifier "DATA" for geometric structural data.
- **| x |** (1 character) describes the structure data type:

p → Point structure

l → Line structure

f → Flat structure

- Between the two '@' symbols, you will find the plot parameters of the structural dataset. These will help you distinguish the various characteristics in the graphic window of SPRING by using different colours, markings and line types. These are:

In case of point structures (p):

Colour number, marking type and marking height (pixel),,

In case of line structures (l):

Colour number, line type and flag for markings at intermediate points,

In case of flat structures (f):

Colour number, line type and flag for filling the polygon.

The definition of plot parameters including the '@' is optional. If they are not specified, SPRING will use default plot parameters for the structural data.

The text gives a short description on the structural dataset. Using this, you can identify the different datasets in the selection list.

The data lines have the format:

```
nr,   X,   Y,   Z
I6,F10,F10,F10
```

That means:

- 6 columns for numbering the data. By default, SPRING transfers the numbering into serial numbering. It can also be 6 spaces.
- 10 columns for the x-coordinate of the point,
- 10 columns for the y-coordinate of the point and
- 10 columns for the z-value which is evaluated for the structural-characteristic dataset is assigned to nodes or elements. The z-value is optional.

In addition to the project's structure data file structural data from other *. str files can be read in with *Structure → Import…* in SPRING.

The meaning of the identifiers is defined in the file *xsusi.kenn*. This is described in detail in the appendix (p. 454).

3.1.5.1   Format of the structure data file (ASCII)

During the modelling or in the calculation modules it is sometimes asked for a file in the format of the structure data file (ASCII). This file is different from the "classical" structure data file because it contains neither three comment lines nor the attribute IDs! But the format of the data lines corresponds to the structure of the data file!

The data lines have the format (see chapter: Structure and format of the structure data file):

```
nr,   X,   Y,   Z
I6,F10,F10,F10
```


3.1.6 Initialization file *plogeo.ini*

Some default parameters of the plot generation can be altered by using an initialization file, which is always named *plogeo.ini.* The file is located in the directory:

"C:\Users\Public\Documents\SPRING\Konfig" (Windows 7 or Windows Vista). This path varies depending on the operating system or the hard drive partitioning (e.g. under Windows XP: "C:\Documents and Settings\All Users\Documents\SPRING\Konfig").

If no file of that name exists, the default settings are used.

The file is read line-by-line until the end of the file is reached. Each line defines one parameter. The first four characters in the line (capitals) specify the parameter which is changed. Then the following numerical de-

scribes either height [in cm or mm] or there is a flag (0 or 1) that results in a particular representation or execution.

The plogeo.ini commands are defined by choosing *Options* from the *Edit menu* of the SPRING-user interface. By saving this menu, the selected settings are stored in the file *plogeo.ini*.

Example of a *plogeo.ini* file:

```
HLEG 0.15
HISO 0.18
HWER 0.13
FLAE 1
TEXT 1
TIME 1
```

All kinds of initialization parameters are described in detail in chapter *SPRING-menus – Edit menu – Options – Plot options* .

3.1.7 Initialization file *spring.opt*

A series of default settings of SPRING are saved in the option file with the name *spring.opt*. The file is located in the directory:

"C:\Users\Public\Documents\SPRING\Konfig" (Windows 7 or Windows Vista). This path varies depending on the operating system or the hard drive partitioning (e.g. under Windows XP: "C:\Documents and Settings\All Users\Documents\SPRING\Konfig").

It is edited via the menu *Options* in the *Edit menu* of the SPRING-user interface.

For every option a line is defined containing the option name, followed by a colon. Then there must be at least one separator (space or Tab) and the desired option value. The order of options in the file is arbitrary. If an option exists several times the last entry will be used. Each line will be interpreted, comment lines get a "#" as the first character. If the option is changed with *SPRING → Edit → Options…* the comment lines will not be saved!

When starting SPRING the file *spring.opt* will be read and the settings are adopted.

All possible initialization parameters are described in detail in chapter *SPRING-menus – Edit menu – Options*

3.1.8 Batch files

The batch files are ASCII files with which the individual calculation modules can be started manually. They contain the information of the control parameters and specifications necessary for the calculation. Their structure is described in each chapter of the calculation modules. Examples of the different batch files are available on our homepage.

3.1.8.1 Creating a batch job

To make an automatic sequence of calculation steps of the SPRING modules, all the main processing modules can be started with a parameter that suppresses pressing the <return>-button. Particularly in case of a prolonged absence of the user overnight or on weekends the calculation of various sequences can be started in this way.

The start of a SPRING main processing module has to be started with the addition of *-return*. The start of the calculation module SITRA has e.g. the form:

sitra -return sitra.bsi

The implementation of two job sequences model checking (DADIA) – flow computation (SITRA) - data export (NACHLAUF) – plot generation (PLOGEO) in the MS Windows directories D: \ Projects \ paradise \ Sim01 and Sim02 looks like this:

```
D:\
cd \Projects\paradise\Sim01
dadia -return dadia.bda
```

```
sitra -return sitra.bsi
nachlauf -return gang.bna
plogeo -return flur.bpl
cd \Projects\paradise\Sim02
dadia -return dadia.bda
sitra -return sitra.bsi
nachlauf -return gang.bna
plogeo -return flur.bpl
```

These command lines are stored in a file in ASCII format with the extension *.bat*. After calling the command prompt (DOS shell), the filename is entered and confirmed with <return>. This approach also works in a similar fashion in a terminal on Unix-based operating systems.

3.1.9 Structure of the file *sitr*

In the file *sitr* all material-specific parameters, stop criterion, scaling factors of node-and element values as well as information about the equation solver are stored, which are set through the input window of the calculation modules. The file is automatically created after starting the calculation module SITRA (or INSTAT) in the current working directory. It will initially contain the default settings matching the model type.

After editing the parameters in the dialog box, saving the batch file by pressing the button 🖫 (file save as…) or starting the calculation with *OK* the changed parameters are stored in the *sitr* file. Due to the complex structure of the *sitr* file, it is not recommended to edit the file using a text editor.

3.1.10   Crosslinking of water systems with file sitra_vsys.txt

In SPRING crosslinking of water systems is possible. By default, there is an automatic split into equal proportions in the flow direction of the water course (distribution 50:50). If the user wants to define the proportions by himself, he can do that by using the file sitra_vsys.txt. The amounts can be distributed as a percentage, or a maximum transfer amount is defined.

The file has the following structure

```
################################################
#    A-Bach -> Umlaufgraben A-Bach    #
################################################
#
# CROSSLINK SOURCE_NODE TARGET_NODE PERCENT VALUE
# Transferring 25% of the flow from node 208 to node 2856
# Rest to other branches of node 208
CROSSLINK 208 2856 PERCENT 25.0
#
######################################
#    B-Bach -> B-Entlastungsgraben    #
######################################
#
# CROSSLINK SOURCE_NODE TARGET_NODE QMAX VALUE
# Transferring of max. 17500.0 m3/ZE from node 7713 to 7712
# Rest to other branches of node 7713 divide evenly
#
#comment
CROSSLINK 7713 7712 QMAX 17500.0
```

The lines starting with # are comment lines. The specification:

```
CROSSLINK 208 2856 PERCENT 25.0
```

means, that 25% amount of water of node 208 are transferred to node 2856. The remaining 75% will be given to other existing branches at node 208.

The specification:

```
CROSSLINK 7713 7712 QMAX 17500.0
```

means that the node 7713 gives a maximum amount of water of 17,500 m³/a to node 7712. The remaining amount present is distributed equally between the other existing branches at node 7713.

Once the file sitra_vsys.txt is available in the working directory of the corresponding project, it will automatically read in the flow calculation with the module SITRA (steady-state or transient calculation).

3.1.11 Transient input file

For transient calculations, an additional transient input file is required. It contains the boundary conditions changing during the transient calculation. Furthermore, the time steps for transient modelling are defined. The module model checking does not read this file, but the transient calculation module does. Therefore, an automatic data check is not possible. Hence, the user must create the file very carefully.

3.1.11.1 Structure of the transient input file

A transient input file has the following contents:

The file starts with general data (comments, units of time). Then, the initial time is defined. The following data refer to this time point, until a new reference time is defined. As in the input file, each dataset ends with a blank line. The input algorithm is 5(I6, F10.2). The different time steps have to be arranged from the earliest to the latest one. The file is closed by three blank lines.

Line 1-3: Any text possible. The first two lines are used as headline for time-dependent plots.

Line 4: blank line

Line 5: *time unit for mass flow rates*: The text for the time unit starts in the 18th column and is justified to the left after the identification ZEITEINHEIT (time unit) MENG. The available time units are: SEKUNDE (second), TAG (day), MONAT (month) and JAHR (year). All mass flow definitions (KNOT (sinks and sources) / FLAE(areal seepage)) must use this unit of time.

Line 6: *time unit for the time steps:* This is the time unit used for the definition of time steps. In general, there are two possibilities:

Definition of a time unit: This version requires the identification ZEITEINHEIT (time unit) ZEIT (time) and in the 18th column the definition of the time unit with SEKUNDE (second), MINUTE, STUNDE (hour), TAG (day), MONAT (month) or JAHR (year). Each time point is defined by the identification ZEIT and a real number.

Definition of a date: This version requires the identification BEZUGSDATUM (reference point in time) and in the 13th column the definition of a date using the format TT.MM.JJJJ (DD.MM.YYYY day, month, year, including the full stops). This date represents the initial state. Each point of time is defined by the identification DATUM and an exact date. The programme uses the time unit day internally and transfers all other units corresponding to the real calendar (including leap years).

Line 7: blank line

Lines 8+9: *Point of time:* Depending on the definition in line 6, there are two possibilities to define the points of time for the calculation:

The definition of a time unit (ZEITEINHEIT ...) requires a point of time (ZEIT) where the factor for the selected time unit is specified (real, F10.2). For example,

```
ZEIT (Time)
2.5
```

For the case in which the time unit is seconds, the definition shown above specifies the point of time at 2.5 seconds after the initial time.

The definition of an initial date (BEZUGSDATUM ...) requires a point of time (DATUM) with a date. The following line contains the date for the point of time, for example:

```
DATUM
02.04.1995
```

Line 10: blank line

Line 11: *dataset with the corresponding values*: The first line contains the data identification. The next and the following lines consist of data in the format 5(I6, F10.2), i.e. each line contains five pairs with node / element number and value (free spaces are allowed but no blank lines). In contrast to the usual input algorithm it is not possible to assign several nodes to one value by only specifying the first and the last number (from ... to ...).

Lines 12ff: After the data identifier (line 11) the values are written in the format 5(I6, F10.2), that means, in a row can be entered up to five pairs of values 'nodes or element number, value' ('gaps' are allowed but no blank lines.) An entry in the form of from... to as in the model files is not possible.

Each type of data is completed with a blank line

Separated by a blank line, further datasets can be defined in any order. Otherwise, the next point of time is defined (go back to no. 8). Generally, it is possible to define points of time without any datasets.

Attention: If node numbers have more than six digits, the format writer has to be changed by the entry **NUM 7** in the 7th line of the file. Then the format of the node numbers is allowed up to 7 digits and the format in a line is 5(I7, F10.2).

3.1.11.2 Transient input file created from transient structures

During the modelling water courses and wells are usually created as structures (l-and p-structures). If these structures must be assigned with transient data, the proceeding is as follows:

- ■ Copying the existing structure: *Structure → Copy → select*
- ■ Converting the desired structure in a transient structure: *Structure → Convert → To transient structure …*

If you call the list of structures via *Structure → Edit*, the converted structures are marked with the prefix "g" in the column "Type".

Now the transient data must be assigned to the converted structures. This is done via the menu item *Structure → Edit → List* and selecting the appropriate structure. You will get this input window:
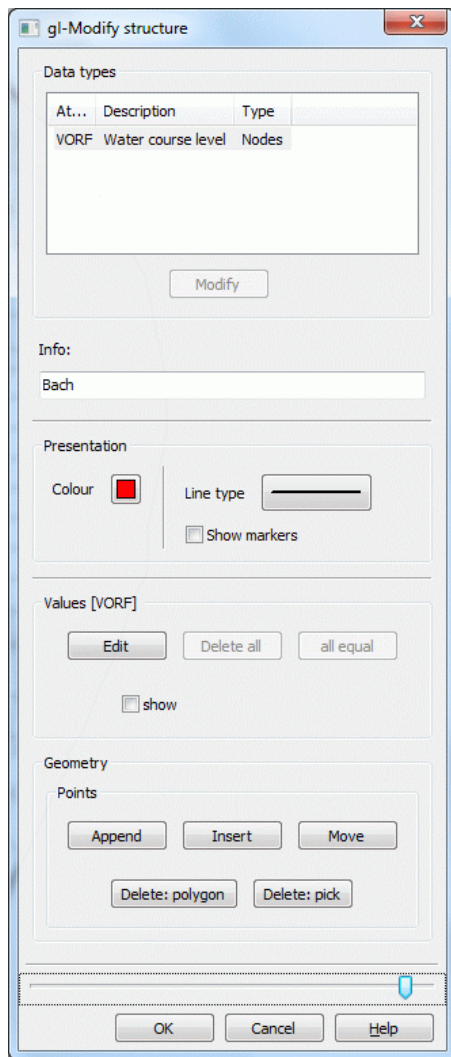
The transient data have to be in the csv-format of the following form:

- Comma as column separator
- Point as decimal separator
- The dates of type date must be specified in the format dd.mm.yyyy.
- The dates of the type time must be specified in the format F10.2

The csv-file for a transient withdrawal with a given date may look like the following:
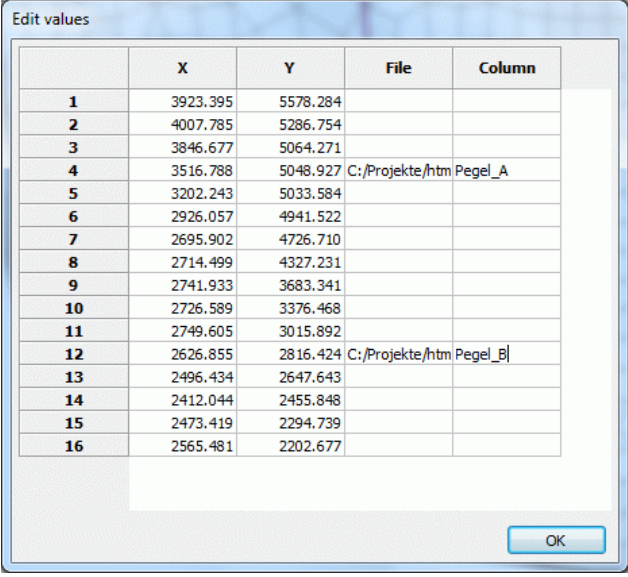
```
Date,Well_1,Well_2,Well_3,
02.01.2004,-998640,-254040,-201480,
10.01.2004,-1585560,-455520,-438000,
17.01.2004,-1226400,-306600,-245280,
22.01.2004,-1024920,-271560,-210240,
```

The first line contains the headings in the following order: date, name of the measuring points (e.g. wells 1, 2, 3, etc.).
The second and subsequent rows contain the time and the associated values of the individual measuring points.

The assignment of the data happens in the menu *Structure → Edit → List* and selecting the appropriate structure.

After clicking the button "values" a table is displayed in which the file and its column with the transient data are assigned to corresponding structure points.

	X	Y	File	Column
1	3923.395	5578.284		
2	4007.785	5286.754		
3	3846.677	5064.271		
4	3516.788	5048.927	C:/Projekte/htm	Pegel_A
5	3202.243	5033.584		
6	2926.057	4941.522		
7	2695.902	4726.710		
8	2714.499	4327.231		
9	2741.933	3683.341		
10	2726.589	3376.468		
11	2749.605	3015.892		
12	2626.855	2816.424	C:/Projekte/htm	Pegel_B
13	2496.434	2647.643		
14	2412.044	2455.848		
15	2473.419	2294.739		
16	2565.481	2202.677		

The file names and the name of the column can be assigned individually, by multiple markings or globally by selecting the column:

After clicking on one or more cells with the left mouse button and simultaneously pressing the "Shift" or "Ctrl" key and then clicking the right mouse button a file selection box or a dialog window opens in which the file or the name of the corresponding column of the selected cells is defined. By confirming with "OK", the data are assigned, but only if a file AND a column are determined. If one statement lacks NO data is assigned to this structure point. This check is also performed when you save the project.

At last, the transient structures have to be assigned. This is done as usually via the menu item *Structure → Assign → List …*

It appears the dialog box "Assign", in which the distance for the assignment and eventually the layers in a 3D model are determined.

After a successful assignment the created transient data can be saved and exported in a transient input file via *Special features → Transient → Export transient input file.*

If the first column of the csv-file is labeled "Zeit" (= time) the exported transient input file requires in the 6th line after a space after "ZEIT" a time unit specified. The choices are: SEKUNDE, MINUTE, STUNDE, TAG_, MONAT, JAHR.

The possible transient data are described in chapter "Transient Data".

3.1.11.3 Transient input files *out66* and *null*

At the end of each simulation, SITRA produces an ASCII file called *out66*. The first line of the file contains several parameters of the transient calculation. For steady state, they are equal to zero. Additionally, the file contains the last computed potential heads or rather pressure head and the last computed concentrations/temperatures (as long as a transport calculation was performed). This lean (or short) version of the *out66*-file is automatically generated after each calculation with SITRA. If the user decides to continue the transient calculation and selects in the dialog box "Save for continuing (*out66*)", he will get the complete file *out66*. This file contains all the data required to continue the calculation.

To continue the transient calculation, the user must rename the file *out66* into *null*.

If a file named *null* exists in the corresponding directory, you have several possibilities for using it:

■ The potential/pressure heads and concentrations of the *null*-file can represent the initial values of an iteration (saturated/unsaturated or density dependent flow). Second, they can be used as initial potential heads or rather initial concentrations for a transient calculation. Both procedures require only the short version of the file *out66* which has been moved to *null*.

■ The continuation of the transient calculation using the data from *null* (warm start) requires the complete version of the file *out66* (renamed into *null*). In that case the time steps are calculated as follows:

The previous calculation stopped at time step nmax. The parameter nmax is the second last parameter of the first line of the file *null*. The calculation is continued with time steps nmax+1, nmax+2, ... The transient time steps of the new calculation start with the results of the most recent of the previous time steps.

■ To continue the transient calculation with another transient input file the parameter nmax (in the first line of the *null* file) is changed to "-1". Then a new transient input file should be used for the continuation of the transient calculation. The starting time (0.0) of the new calculation is then the last calculated time step of the former calculation, with which the warm start data were generated. The transient time steps of the new calculation start at 0.0. This procedure has the advantage that the data already calculated in former time steps is not carried along in the transient input file

3.1.11.4 File with hydrograph data

If the data should be shown over the time after a transient calculation it is possible to compare the calculated curves to the measured values (nominal values).

The measured data have to be present in the format of file with hydrograph data. The file has the following structure:

```
DATUM     01.01.2014


8    #10235
01.05.2014        20.53
05.05.2014        20.80
12.05.2014        21.95
16.05.2014        20.92


3     #256
01.05.2014        18.17
05.05.2014        19.28
12.05.2014        20.12
16.05.2014        19.32
```

The file usually starts with a reference date. In the next line follows a node number. The name of the measuring point can be assigned to the node number behind a hashtag, which is then shown in the plot.

In the following lines are then respectively the measurement date and the measured value (format f10.2) at this node (gauge or groundwater station). If all measured values of a node are listed, the file proceeds with the next node number and its measured values after a separating blank line.

Instead of a date in the first line it is possible to define a time unit corresponding to the time unit of the transient input file (SECOND, MINUTE, HOUR, DAY, MONTH, YEAR). Then a time unit is defined for the measured values of the node instead of the date. Example:

```
STUNDE


1038     #335
1      19.53
2      20.80
```

3.1.11.5 Heat re-injection

In SPRING, it is possible to calculate the re-injection in geothermal groundwater use.

An input file is created, which is taken into account by a corresponding entry in the batch file of the transient heat transport. The file has the following structure:

Time step re-injection node withdrawal node (more than one are allowed) temperature difference [K]

```
1 2575  2577 4 #
2 2575  2577 4 #
3 2575  2577 4 #
4 2575  2577 4 #
......
```

In the sample file, the water temperature (KONZ) of the withdrawal node 2577 is cooled down about 4 degrees and then re-injected at node 2575.

3.1.12   Structure of the zoning files

The zoning files are ASCII files that have basically the following structure:

In the first column are the numbers of the zones that have been assigned in the model file to the elements (attributes KD-Z, Z-AP). These zones will now be assigned according to their task (adsorption or degradation and production) of the required parameters:

Die Zonierungsdateien sind ASCII-Dateien, die grundsätzlich folgenden Aufbau besitzen:

■   e.g. for the adsorption by Freundlich:
```
Zone number   1. Freundlich Isotherm   2. Freundlich Isotherm
        1      0.001       0.001       #zone number CHI1 CHI2
```

■   e.g. degradation and production (Step or tent function):
```
Zone number Calc. method  Max. slope  lower limit   upper limit
      1   1   0   2   10   20 #Zone number, calc. method, PRODF0, PRODF1, U1PRODF, U2PRODF
```

with:

Calculation method = 1 ==> Application of the Step function

Calculation method = 2 ==> Application of the tent function

The individual parameters are separated by spaces or tabs, after a # comments can be written into the file.

If there is no file with zoned parameters it can be created the following way:

Enter a file name in the corresponding dialogue field. Click the button "Create file" ( ) and save the batch file or start a computation. Then the entered values are stored in the "file for zoned parameters".

The relevant dialogue box appears only if the attributes Z-AP (for degradation and production) or Z-KD (for adsorption or heat) are present in the model file.

3.1.13   Files of the inverse modelling

The format and structure of the parameter and observation file are described in chapter "Inverse modelling - Realization in SPRING" by using an example.

3.1.14   File with measured or interpolation data

Files that contain measured or interpolation values have to be in ASCII format of the structure data file. The file *DiffEich.txt*, which contains differences between measured and calculated data and is created during the plot generation is also available in this format.

3.1.15   Result files

The result data of the calculations are written to the output files (*out.**). What data is stored in detail is described in the calculation modules. Most of this data can be imported via the menu item *Attributes → Import model data/computation results* and visualized in SPRING.

There are other files that provide the results of the calculations tabular. This data are differences in measurement, (steady-state or transient), mass flow and leakage rates, concentrations, heat or mine water volumes.

An output of this data depends on the calculation module and happens only if certain attributes are present in the model file. The structure and content of the files *DiffEich.txt*, *C <node no.>.txt*, *reaktion.csv*, *leakageBILK / BILE.txt*, *gruben.csv* and *fracht.csv* are explained in the appendix, chapter "Structure and content of the result files" (p. 446).

3.1.16   File with elements to hide

In the plot generation exists the possibility to hide data in certain elements. The area to be hidden is defined in a separate ASCII file by specifying the corresponding item numbers.

The format of this file corresponds to the format of the model file (16x, 9 (I6, A1), A1) beginning from the first row without identifier or comment lines. The data value is arbitrary or may be absent. If the node or element number has more than 6 digits (NUM6), the corresponding format (e.g., NUM7) must be specified in the first line of the ASCII file.

3.1.17   Units

Predefined are the following units:
- Length in [m]
- Areas in [m²]
- Volumes in [m³]
- Angles in degrees [°]

The other data are defined as follows:
- Permeability coefficients in [m/s]
- Concentrations:

  In a mass transport calculation using a tracer or a density-independent flow it is possible to enter the concentration with an "arbitrary" unit (i.e. [g/l], [[g/g], [mg/l], [kg/kg], etc.).

  In a density-dependent mass transport calculation the concentration has to have the unit [kg (solute)/kg (solution)].

  In a calculation with adsorption the concentration has to have the unit [kg/m³].
- Temperatures in [°C]
- Density in [kg/m³]
- Material data (MATE) in [cm/s]
- Transmissivities (TRAN) in [cm²/s]

The relevant time unit for in- and outflow is determined in the model file. Therefore, all in- and outflows of the model have to be converted to this time unit. The following time units are available: SECOND, DAY, MONTH, YEAR.

Mass inflow and outflow (QKON) are defined per time unit second.

3.2 Data description

The mesh of the finite elements (triangles and four-corner) is the basis for all data. Through this mesh, which consists of nodes and elements, the geometry and geography is defined and reproduced. Each element and each node must be numbered. The individual data are assigned to these numbers. The geometrical data are referred to as an identification. The remaining data types are referred to as attributes. The data can be divided into four groups:
- Geometric data
- Material data
- Boundary condition and source terms
- Auxiliary data for further analysis and presentation

Meaning in the following descriptions:

1K: for models with 1D fractures only
2K: for models with 2D fractures only

When a model type is not specified, the appropriate type of data is useful for several types of models.

3.2.1 Geometric data

This is all the data necessary to generate the FE mesh (nodes/elements/fractures).

3.2.1.1 Definition of elements

Identification: **ELEM** [-]

With the help of ELEM, the triangular and quadrangular elements are defined by specifying the nodes which form the element. The elements can be numbered in any order. The nodes for each element have to be ordered anticlockwise. In the case of triangular elements, the fourth node is 0. Each line of the model file can contain two complete element definitions.

Example:
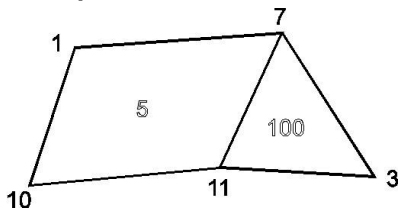
Fig. 18: Definition of the element nodes

Within a line, either the 1st or the 2nd element definition can be deleted without completing the columns again. The element descriptions are the result of the mesh generator and stored automatically in the model file.

3.2.1.2 Mesh generator for uniform rectangular meshes

Identification: **GENE** [-]

This data type makes it possible to generate uniform rectangular meshes. The coordinates of the nodes and the corresponding element definitions are created automatically. Therefore, such areas in the mesh require no input for KOOR and ELEM. It is possible to generate uniform meshes in several rectangular areas of the model. The areas can be connected to each other. Furthermore, the generated mesh can be combined with the data types KOOR and ELEM. As a result of GENE, the generated nodes and elements are printed into the output file of the model checking. Their format is equivalent to the *.net*-file and you can insert them directly. With their help, the existing data can be modified or completed. Copying generated data into the *.net*-file is required for editing the file with SPRING. Each uniform rectangular mesh is defined by the following parameter.

```
Columns 1 - 10: X1 x-coordinate bottom lefthand corner in m
Columns 11 - 20: Y1 y-coordinate bottom lefthand corner in m
Columns 21 - 30: X2 x-coordinate top righthand corner in m
Columns 31 - 40: Y2 y-coordinate top righthand corner in m
Columns 41 - 46: NX number of elements in x- direction
Columns 47 - 52: NY number of elements in y- direction
Columns 53 - 58: NAK first node number, node at coordinates (X 1, Y 1)
Columns 59 - 64: NAE first element number (bottom lefthand element)
Columns 65 - 70: IVH = 1 horizontal node numbering
                 IVH = 0 vertical node numbering
Columns 71 - 80: PHI rotation angle in degree
```

Example:

This combination results in the following picture:

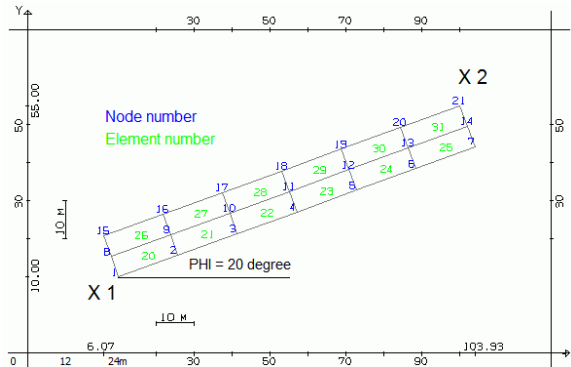(X1 = (10.0, 25.0), X2 = (100.0, 70.0), NX = 6, NY = 2, NAK = 1, NAE = 20, IVH = 1, PHI = 20.0 degree)

Fig. 19: Automatically generated rectangle mesh

3.2.1.3   Definition of fractures

Identifications: **1DEL** (1D fractures), **2DEL** (2D fractures)

Model type: 1K, 2K

With the help of these data, the 1D and 2D fracture elements are defined by specifying the nodes which form the fractures. The fractures can be numbered in any order.
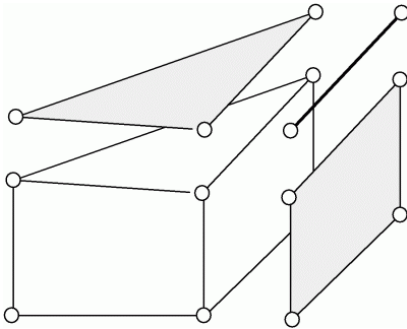
Fig. 20: Possible fracture elements

Each fracture definition contains the fracture number and the two to four node numbers which form the fracture. In the case of 2D fractures with only three corners, the fourth node is 0. Each line in the model - or 3D file (*.3d) can contain two complete fracture definitions.

```
Columns  1 -  6: no. 1st fracture
Columns  7 - 12: 1st node no. 1st fracture
Columns 13 - 18: 2nd node no. 1st fracture
Columns 19 - 24: 3rd node no. 1st fracture (only for 2D fractures)
Columns 25 - 30: 4th node no. 1st fracture (only for 2D fractures)
Columns 37 - 42: no. 2nd fracture
Columns 43 - 48: 1st node no. 2nd fracture
Columns 49 - 54: 2nd node no. 2nd fracture
Columns 55 - 60: 3rd node no. 2nd fracture (only for 2D fractures)
Columns 61 - 66: 4th node no. 2nd fracture (only for 2D fractures)
```

Within a line, either the 1st or the 2nd fracture definition can be deleted without completing the columns again. One-dimensional fractures (1DEL) can currently only be created or shown with the program KLUFTI.

3.2.1.4   Node coordinates

Identification: **KOOR** [-]

The nodes are the corners of the elements. Each node is defined in a rectangular coordinate system by an X- and Y-coordinate. During the modelling with SPRING a number is assigned automatically to each node.

In vertical models the y-coordinates must be equal to the heights above sea-level. This is important for computing the free surface.

One line in the model file contains one to three nodes. Within a line, any node can be deleted without completing the columns again. Only blank lines must not occur.

3.2.1.5 Scale data

Identification: **MASS** [-], 9th line of the model file

The two numbers represent the scales for the directions x and y. They are interpreted for the creation of plots. When building a new model in SPRING with the user interface the scale is set in the dialog *File → New*.

3.2.1.6 Model type

7th line of the model file

Identification: **HORI** for a 2D horizontal model

Identification: **VERT** for a 2D vertical model

Identification: **3D-M** for a coupled 2D/3D model i.e.. complete 3D model

Identification: **V-3D** for a 3D vertical model

The identification specifies whether the model is a horizontal, a vertical or a 3D model (or rather a coupled 2D/3D model).

The new model type is set in the dialog *File → New* when building a new model in SPRING with the user interface. Since you start initially with a 2D model, there is only the choice between a horizontal or a vertical model available.

3.2.1.7 Offset for the numbering of 3D nodes

Identification: **3DNR** [-] (only in 3D model files, *.3d)

The nodes and elements in deeper layers of 3D models are automatically numbered by adding a multiple offset value to the number in the topmost layer. The default offset is 10,000 and can be changed with the help of 3DNR. However, the highest node number of the top node layer should be less than the selected offset. The offset numbering will change automatically when the mesh generator of SPRING creates a node number in the top layer, which is greater than the offset numbering.

Example: 3DNR = 15000

The node 15 in the topmost layer corresponds to node 15015 in the second and 30015 in the third node layer.

3.2.1.8 Boundary nodes for a 3D area

Identification: **3DRA** [-] (only in 3D model file, *.3d)

The definition of a closed boundary (i.e. where the first node is equal to the last one) enlarges the part of the actual model inside this boundary into the 3rd dimension. This can be the whole 2D horizontal model or one or several parts of it. This is done via the menu *Mesh → 3D → 3D-boundary → generate/add*. The numbers of the boundary nodes are stored automatically in their order and anticlockwise separately for each boundary.

To calculate multiple separate 3D areas without surrounding 2D horizontal model, these areas must have the same number of ZKOR layers, may not have any geometric overlapping and must be connected to at least one node by the attribute GLEI.

For the case of one or more partial 3D models, the boundary conditions for the 3DRA nodes can be influenced by plus and minus signs at the node numbers (see Creating 2D/3D models):

- Positive node numbers (plus sign) mean that for the corresponding generated 3D nodes an unknown equal potential head is assumed (= only horizontal flow).
- Negative node numbers (minus sign) mean that for the corresponding generated 3D nodes this boundary condition is not assumed.

The assumption of unknown equal potential heads over the whole depth as mentioned above is only possible for a partial 3D model. For a complete 3D model, the boundary conditions (potential head, reaction rates) have to be specified explicitly. Here the potential heads are not set equal over the depth.

3.2.1.9 Layer definition of a 3D model

Identification: **3DSH** [-] (only 3D model files, *.3d)

The 3D layers are automatically generated with the help of the thickness of the 2D model by specifying the subdivisions (not their absolute thickness).

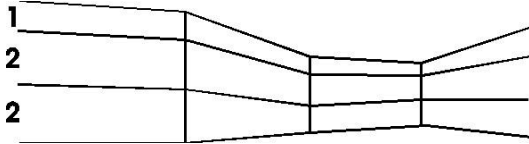Example:

The statement: 3DSH: 1 2 2 does the following:

Fig. 21: Automatic layer definition

In total, four new node layers are generated. The 1st element layer is half as thick as the 2nd and the 3rd element layer. The absolute z-coordinates are determined automatically by the difference between upper and lower boundary of the model.

3.2.1.10 Z-coordinates for a 3D model

Identification: **ZKOR** [-] (only in 3D models, *.3d)

The data type defines the z-coordinates for deeper node layers in 3D models. The node numbers are used without offset and are equivalent to the corresponding node in the topmost layer. The 1st ZKOR dataset defines the coordinates for the 2nd node layer, the 2nd ZKOR dataset defines the coordinates for the 3rd node layer and so on. Hence, the number of the node layer is always greater by 1 than the corresponding ZKOR dataset. The z-coordinates of the topmost nodes are obtained by the 2D input file. The position of the topmost node layer results from the data of the model file: from the ground level or nominal potential heads.

Example:
```
ZKOR    (2nd node layer)
        22.53       1-    125,    235,
        23.11     126,    154,
ZKOR    (3rd node layer)
        16.27       1,      5,
        16.08       2-      4,     25
```

For an offset value of 10,000 the node 10001 in the 2nd node layer lies at z = 22.53 m and the node 20001 in the 3rd node layer lies at z = 16.27 m.

3.2.1.11 Time unit

Identification: **ZEIT** [-], 5th line of the model file

Starting in column no. 17, one of the following time units is inserted justified to the left (only the first four characters are relevant):

SEKUNDE (for second), TAG (for day), MONAT (for month) or JAHR (for year)

The time unit TAG consists of three letters and a space.

Hence, the dataset consists of one line (plus a blank line). When building a new model in SPRING with the user interface the time unit is set in the dialog *File → New.*

The time unit refers to all inflows / outflows in the model, i.e. all these values must be transformed into this time unit.

3.2.2 Material data

This includes all data of the material properties (aquifer parameters such as permeability, storage capacity, porosity and fracture parameters).

3.2.2.1 Terrain settings caused by subsurface mining

Attribute: **BERG** [m] (horizontal or 3D models only)

These values represent the influence of terrain settings caused by subsurface mining on lower and upper boundaries, the aquifer thickness, receiving water course and ground levels. A consideration of these data types is done automatically. Therefore, when entering these values, the altitudes based on the reference time of the subsidences have to be entered. In regions where no settings occur, you must assign the value 0.0 to each node (unit is m).

By subsidences especially the overground proportions of the receiving water are affected. Thus, an exfiltrating water course (receiving water) can cut deeper in the aquifer, by which an increased groundwater discharge results.
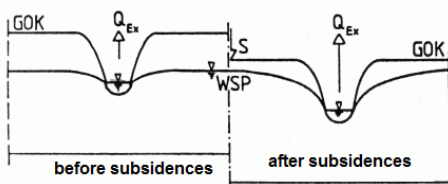
Fig. 22: Influence of terrain settings on an exfiltrating water

On the other hand, the subsidences cause that the previously exfiltrating water course loses his gap and no longer drains its surrounding. Result, it caused water logging at the surface.

For a before infiltrating water, it is possible that the subsidences cause an incision into the aquifer and the infiltration process is reversed in an exfiltrations process.
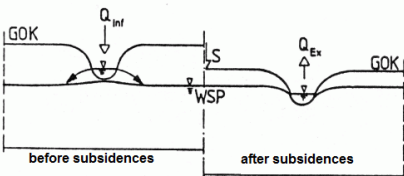
Fig. 23: Influence of terrain settings on an infiltrating water

3.2.2.2 Density parameters

Attribute: **DICH** [kg/m³] (node wise) (transport model only)

Measure of the density of sources / sinks and potentials.

3.2.2.3 Dispersivities

Attributes: **1DDI** (for 1D fractures)**, 2DDI** (for 2D fractures) (for 1K or 2K transport models)

Attribute: **DISP** (element wise) (transport model)

Modelling transport processes requires the longitudinal dispersivity for each element (and each fracture).

In case of matrix elements (DISP), by default the transverse-horizontal value is assumed to be one-tenth (1/10) of the longitudinal value. The transverse-vertical value for 3D models is assumed to be one-hundredth (1/100) of the longitudinal dispersivity. All the factors can be changed in the dialog box of the transport calculation.

In the case of fractures, the transverse dispersivity is equal to 0.
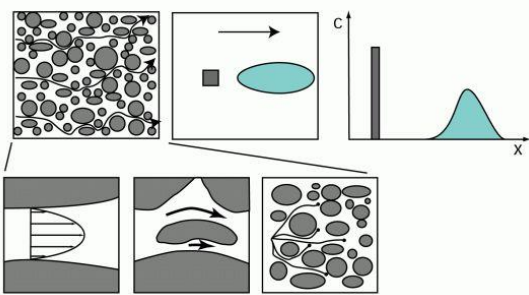
Fig. 24: Schematic representation of a dispersion process

3.2.2.4 Permeability
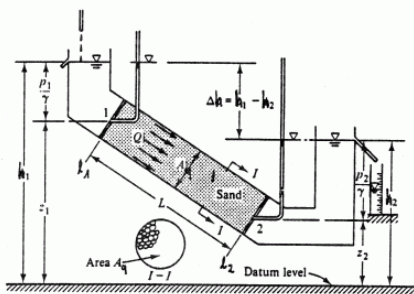
Example for the construction of a permeability test:

Fig. 25: Quantities to describe the permeability (from: Bear, 1979)

The in SPRING defined permeability values KWER, KWEV and KFVH apply for isotropic flow conditions. That is, the permeability is independent of the direction of flow and thus equally strong in all directions.

An anisotropy can be taken into account in the two-dimensional case by assigning the attributes ZONE and MATE.

For the three-dimensional case is a consideration of anisotropic flow conditions via the menu item *Attributes* → *Special features* → *Anisotropic conductivities* possible. This menu item is activated as soon as the attribute Z-KA is assigned.

It is also possible to adjust the conductivities according to the geological structure of the layers, and thus the element geometry. If the conductivities should be corrected for ALL layers on the basis of geometry, this is done via an additional entry at the end of the batch file *sitra.bsi*:

```
ROTATION KWER
```

The individual layers can be treated differentially using the attribute Z-KA and the menu item *Attributes* → *Special features* → *Anisotropic conductivities*.

3.2.2.4.1 K-value

Attribute: **KWER** [m/s]

With this type of data permeability values (only isotropy) may be entered element wise. The unit m/s is independent of the time unit of the model file.

'F' in column 15 means that the permeabilities for the corresponding element numbers are saved as measuring values which are not to be changed during the iterative calibration.

In general, the input K values apply to isotropic conditions by default, e.g., in the horizontal and vertical model it is $k_x = k_y$ and in the 3D model it is $k_x = k_y$ and $k_z = k_x * KFVH$. The factor KFVH for the permeability in the Z-direction is set to 1/10, unless it is explicitly defined. By the factor KFVH <1.0 a degree of anisotropy is considered in the vertical direction.

3.2.2.4.2 K-value, vertical

Attribute: **KWEV** [m/s] (3D model only).

For 3D elements, it is possible to assign vertical permeabilities. When assigning the attribute KWEV it is: $k_x = k_y = $ KWER, $k_z = $ KWEV. The unit m/s is independent of the time unit of the model file. If KWEV is only defined for only some of the elements, applies to the other elements as before: $k_z = $ KFVH*k_x. The factor KFVH for the permeability in the Z-direction is set to 1/10, unless it is explicitly defined.

3.2.2.4.3   K-value, ratio for the vertical permeability

Attribute: **KFVH** [-] (3D models only)

In three-dimensional models, data often have default values to reduce the time necessary for data input. If there are no vertical K-values given, they are calculated from the horizontal K-values (KWER) multiplied by a factor. This factor is set to 1/10 and is stored in the 3D file as attribute KFVH. It can be edited there or changed via the menu item *Attributes → Special features → Adjust factor vertical/horizontal K-value (KFVH)....*

Example: KWEV is assigned to layer 4 (KWEV = 0.0004). Because of the automatically assignment of attributes to deeper layers this definition means that the layers 5 and lower are automatically assigned to the value KWEV = 0.0004. In this case, the factor KFVH only applies to the layer 1-3.

3.2.2.4.4   K-value, limitation

Attributes: **KMIN** [m/s] (minimum K-value), **KMAX** [m/s] (maximum K-value)

The user can limit the range of the calibration results by defining a minimum and maximum permeability at each element. The values have always the unit m/s for any defined unit of time.

3.2.2.5   Ground level

Attribute: **GELA** [m NN] (horizontal or 3D models only)

The ground level is always required where an impermeable layer is defined (UNDU) or the difference between the ground level and the groundwater level should be computed (ground level - groundwater level = difference between ground and ground water level). Furthermore, the ground level is required to compute the initial and maximum thickness and it can define the topmost node layer in a 3D model. The ground level is always assigned to nodes.

Based on the terrain data the following relief image can be created with the program Surfer:
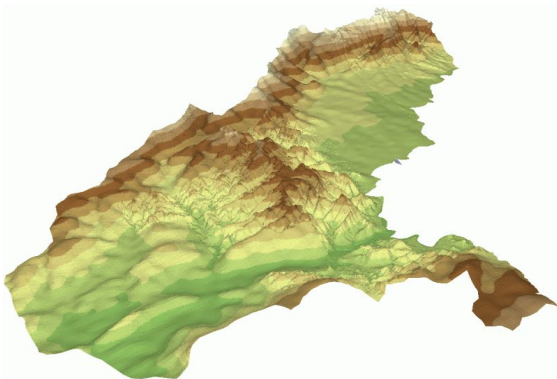
Fig. 26: Ground level as relief image

3.2.2.6 Fracture properties

Example of the form of fractures in reality:
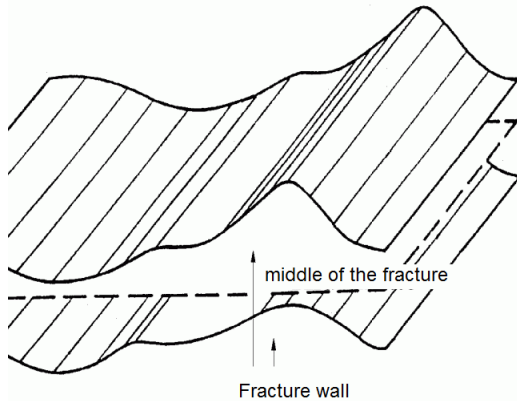
Fig. 27: Real form of fractures

3.2.2.6.1 Fracture thickness

Attribute: **1DMA** [m] (1K models only)

2D models which include 1D fractures require the definition of a fracture thickness (width across the aquifer thickness) for each fracture. With its help, the fracture volume that is available for flow and transport processes is obtained. The following figure illustrates the flow in a fracture:

Fig. 28: Flow in a fracture

3.2.2.6.2 Fracture opening width

Attributes: **1DBR** [m] (for 1D fractures, 1K models), **2DBR** [m] (for 2D fractures, 2K models)

Models which include 1D or 2D fractures require the definition of a fracture opening width for each fracture. The fracture opening width is used in calculating the fracture volume, which is available for flow and transport processes. Because of the so-called cubic law, the fracture opening width b is important in the calculation of flow. The permeability of fractures is proportional to $b^2/12$.
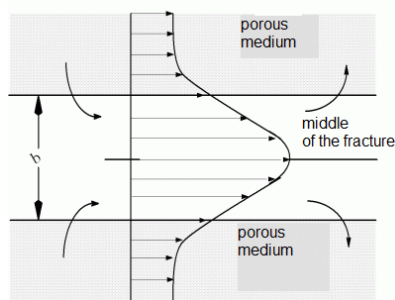
Fig. 29: Idealised form of fractures with a velocity profile, fracture opening width b

3.2.2.7 Compressibility of the entire system

Attribute: **KOMP** (nodewise) [m²/N] (only 3D models)

The compressibility of the entire system, consisting of matrix and fluid, can be set with the attribute KOMP. If the attribute KOMP is not defined in the model file, the matrix (α)- and fluid (β) compressibility that are entered in the input dialog of the transient calculation (see p. 300) are considered. The formula for the compressibility is:

KOMP = (1-n)*matrix compressibility + n*fluid compressibility, with n = porosity

3.2.2.8 Leakage properties

3.2.2.8.1 Leakage coefficients for receiving water courses

Attributes: **LEKN** (node wise), **LERA** (polyline wise), **LEEL** (element wise, horizontal and 3D models only)

These data types define the leakage coefficient, which represents a linear relationship between the difference from the receiving water course to the groundwater level h_{vorf} **-h** and the leakage rate **Q.**
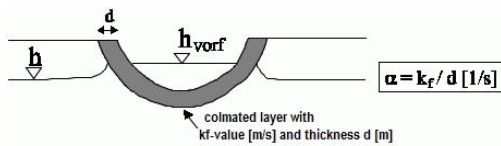
Fig. 30: Schematic representation (1) of the leakage parameters
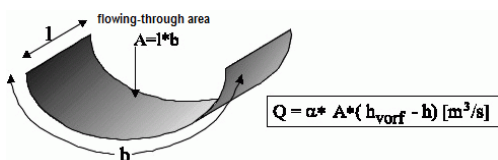
Fig. 31: Schematic representation (2) of the leakage parameters

The calculation modules require a transformed leakage coefficient α in the time unit [TU] of the model file:

L [m²/TU]= α **A** [s/TU]

With the help of this value, the leakage rate (per TU) can be computed directly from the potential difference. **A** represents the area in the bottom of the receiving water course where the water flows through. The term "[s/TU]" transforms the time unit second (s) into the time unit of the model file (TU). The plot generation can represent both the original input values (LEKN, LERA, LEEL) and the corresponding transformed values for **L**.

The leakage coefficient α [1/s], the input values (LEKN, LERA and LEEL) and **L** have the following relationships between each other:

Nodes (LEKN):

LEKN [m²/TU] = α **A** [s/TU]   i.e. **L** = LEKN

Polylines (LERA):

LERA [m/TU] = α**b** [s/TU]   i.e. **L** = **l** * LERA

The corresponding length (l) for each node of the polyline/receiving water course is computed in the model checking. It is defined as the sum of half the distances to the adjacent nodes. Therefore, the leakage coefficients with regard to a polyline must be put in in the correct order.

Elements (LEEL):

The meaning of LEEL depends on the data for UNDU. There are two different meanings:

■   UNDU defined at any nodes of the element or at all nodes = 0:

LEEL [1/TU] = α [s/TU]   i.e. L = A * LEEL

The element area A is computed in the model checking.

■   UNDU with values > 0 at minimum at one node of the element:

LEEL [m/s] = k_v [m/s].

The input value represents the vertical permeability in m/s for the covering layer. The value UNDU is averaged from all nodes in the element, where no input value means UNDU = 0. The average value represents the thickness of the covering layer. The final leakage coefficient L requires the element area A which is obtained in the model checking:

L = LEEL / UNDU A [s/TU]

Leakage coefficients always have to use the time unit from the model file (TU) except for LEEL combined with UNDU.

The leakage calculation always requires the definition of a receiving water course (VORF). Otherwise, the leakage data are ignored.

3.2.2.8.2   Leakage coefficient, ratio

Attributes: **LKFA** [-] (node or polyline wise), **LEFA** [-] (element wise, horizontal and 3D models only)

For the accurate mapping of the leakage boundary conditions at nodes (or polylines) and elements with known leakage the ratio between ex- and infiltration is defined:

At nodes and polylines:

exfiltration coefficient / infiltration coefficient = LKFA, (here the exfiltration coefficient is LERA or LEKN)

In Elements:

exfiltration coefficient / infiltration coefficient = LEFA, (here the exfiltration coefficient is LEEL)

3.2.2.8.3   Leakage coefficient, limitation

Attributes: **MXKE** for max. exfiltration rates [m³/TU/m] (polyline), **MXEE** for max. exfiltration rates [m³/TU/m²] (elements, horizontal and 3D models only), **MXKI** for max. infiltration rates [m³/TU/m] (polyline), **MXEI** for max. infiltration rates [m³/TU/m²] (elements, horizontal and 3D models only)

These values define maximum infiltration and exfiltration rates for leakage relationships. Limiting the rate along a polyline requires a value per unit length [m³/TU/m]. The maximum rate for the single nodes is obtained by multiplying by the proportional distance. The distance is defined as the sum of half the distances to the adjacent nodes. That's why the correct order of the nodes is very important! The maximum rate for an element has the unit [m³/TU/m²]. It is obtained by multiplying by the corresponding element area. Maximum infiltration rates can also be defined with the help of tear-off heights (data type ABRI with water course level (VORF)).

Examples:

- MXKI = 0 means, that no water infiltrates **into the groundwater** from these nodes.
- MXEI = 0 means, that no water infiltrates **into the groundwater** from these elements.
- MXKE = 0 means, that no water **from the groundwater** reach these nodes.
- MXEE = 0 means, that no water **from the groundwater** reach these elements.

(In case of inverse modelling, maximum infiltration and exfiltration rates are not available.)

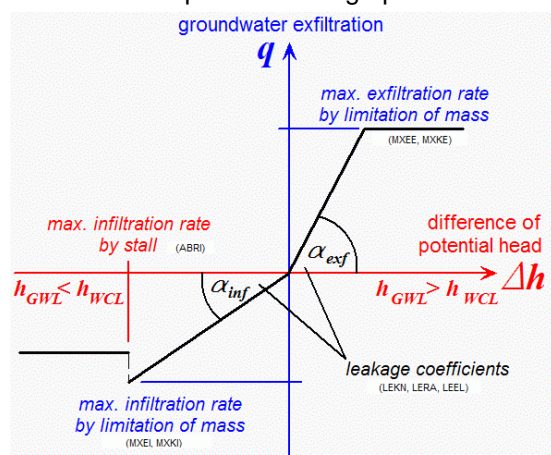The relationship of the leakage parameters illustrates the following graphic:

Fig. 32: Relationship of the leakage parameters

If in a model maximum in- and/or exfiltration rates are assigned an iteration is done automatically during any flow calculation.

Exceeds the calculated leakage rate at a node or element the maximum in- or exfiltration rate, the leakage boundary condition is switched off and the appropriate maximum rate is set as new boundary condition.

The iteration is continued until the conditions are fulfilled for all nodes and elements. The leakage limitations can be defined as transient boundary condition as well using the attributes KMKE and KMKI.

3.2.2.9 Thickness

Attributes: **MAEC** [m] (element wise), **MAEK** [m] (node wise), **MAXM** [m] (node wise)

The thickness represents the aquifer height for the case of a horizontal model and the section width of the aquifer in the case of a vertical model. The thickness is assigned to elements or nodes. The unit is m.

If column 15 contains an 'F', the thickness for the corresponding elements or nodes is fixed and cannot be changed during an iteration of thickness (horizontal model). Hence, the aquifer is confined. The thickness can be defined as transient boundary condition as well.

3.2.2.9.1   The following chapter describes the calculation of the initial and maximum thickness Calculation of the initial and maximum thickness

SPRING has various possibilities for defining the thickness of an aquifer. The model checking (see p. 242) is able to determine the thickness with the help of one of the following data combinations:

- explicit definition of the thickness using the attribute MAEC or MAEK (then an iteration is not possible)
- the difference between upper and lower boundary (attributes OBER - UNTE) (2D only)
- difference between the ground level (GELA) and the lower boundary (UNTE or UNTK) considering the thickness of the impermeable layer (UNDU), thickness = (GELA - UNDU) -UNTE or UNTK)
- difference between the nominal potential heads and the lower boundary (attributes EICH – UNTE or UNTK)
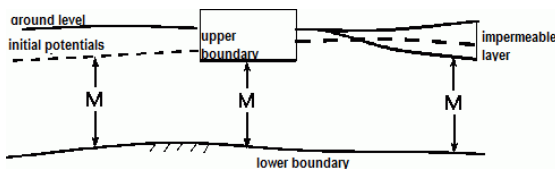
Fig. 33: Calculating the thickness (M) from existing data

The value for the thickness is assigned to elements or nodes. The input data GELA, UNDU and EICH are values assigned to nodes. Therefore, they must be interpolated first from node wise values to element wise values, when the thickness is assigned to elements (provided that each node has one value). Hence, the interpolation can finally lead to less exact results for the thickness.

It is useful to define the upper boundary in case of a natural or artificial (building) upper restriction of the aquifer. It is possible to define an upper boundary in partial areas of the model. If the aquifer is covered by a relatively impermeable layer, the thickness of this layer and the ground level are the best input data. Then leakage processes can be taken into account automatically. For 3D models, an upper limit can be defined by generating an additional layer with very low permeability.

The definition of nominal potential heads (measured groundwater level) is only necessary if a calibration is done. Nevertheless, further usage of EICH after the calibration is recommended, because the resulting initial thickness is the best input for a possible iteration of thickness. If the user has defined values for upper boundaries as well as for nominal potential heads, the program calculates two values for the thickness and finally chooses the smaller one.

After running the model checking, the thereby-computed thickness is used for the iteration of the free surface in horizontal 2D models.

In a fashion similar to the calculation of the initial thickness, during the model checking the maximum possible thickness is calculates for each element, in order to decide whether confined or unconfined conditions prevail. The maximum thickness is calculated as the differences between:

- the upper and lower boundaries (attributes OBER – UNTE)
- the ground level and the lower boundary considering the thickness of the impermeable layer, thickness = (GELA - UNDU) -UNTE or UNTK
- the ground level and the lower boundary (attributes GELA – UNTE or UNTK)

Based on these initial and maximum possible thicknesses the calculation of the free surface is done during the flow calculations.

3.2.2.10 Material data for anisotropic zoning (2D)

Attribute: **MATE** [-] (horizontal models only)

As mentioned in the paragraph on zone identification numbers the data type MATE assigns values for permeability and angle to each zone identification number in a horizontal model. The attribute MATE requires necessarily the attribute ZONE (zone numbers).

The unit for the permeability is in this case always m/s.

Example:

```
ZONE identification number
          1        7-    11,    15
         10       16-    20
         10        1-     6,    12-    14
MATErial data
     1        0.005       0.005       0.0
    10        0.001       0.0009     30.0
```

The elements 7-11 and 15 have an equal permeability of 0.005 m/s in both directions without any rotation. The elements 1-6, 12-14 and 16-20 are anisotropic.

The same material all over the model area means one material dataset with any identification number.

An angle different from 0, or different permeabilities for the directions x and y, mean that the corresponding element is not changed during the calibration!

The following figure illustrates the geometric allocation:
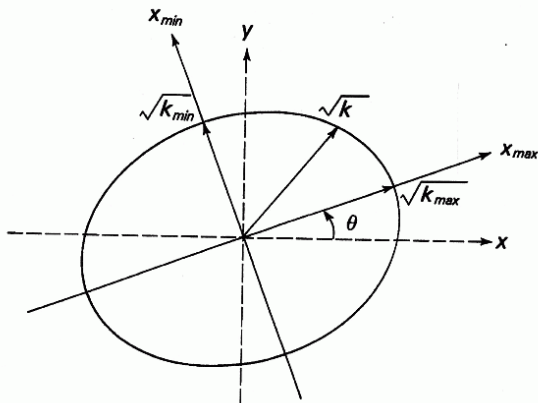
Fig. 34: Anisotropy in 2D

The axes X_{max} and X_{min} denote the axes X and Y which are rotated by the angle Θ. K_{max} and K_{min} are the maximum and minimum conductivities in the direction of the respective axes. K is the resulting effective conductivity in the direction of flow, which is calculated within the flow calculation.

The anisotropic material values can be entered manually in the *.net file or automatically assigned via the menu item *Attributes → Special features → Anisotropic conductivities...* (p. 163).

3.2.2.11 Upper boundaries

Attribute: **OBER** [m NN] (horizontal models only)

Upper boundaries can be used in elements where the lower boundaries are specified as well. The corresponding elements do not require the definition of thickness. However, the initial and maximum thicknesses are obtained by using both values, the lower and the upper boundary.

The data types thickness, upper and lower boundary can be used simultaneously in one model. However, for each element only one thickness definition can exist. The upper boundary is always assigned to elements. The unit is m NN. For 3D models, an upper limit can be defined by generating an additional layer with very low permeability.

3.2.2.12 Porosities

Attribute: **PORO** [-]

The porosity defines the part of the pore space which is used as flow path. The parameter is required for computing the field velocity or the storage coefficient. Porosity is defined as the ratio between the volume occupied by the groundwater and the total volume of the aquifer. The porosity is always assigned to nodes.

3.2.2.13 Saturation parameter, hysteresis

Attributes: **ASAT** [-] (initial saturation), **USAT** [-] (zone number for Van-Genuchten-parameter, vertical- and 3D models only)

The flow computation with the module SITRA contains three different parameter combinations for the residual saturation (S_{res}), the reciprocal of the water inlet pressure ($a = 1/p_e$), the pore size distribution index (n) and specific parameters in the function of the relative k-value (l). The determination of the soil (sand, silt or clay) depends on the assigned permeability. For the calculation, the corresponding soil parameters are used:

K-value [m/s]	S_{res}	a	n	l	Soil type
$> 9.81*e^{-4}$	0.2	$1.4*e^{-3}$	1.5	0.5	Sand
$9.81*e^{-4}$ bis $9.81*e^{-7}$	0.4	$2.5*e^{-4}$	1.35	0.5	Silt
$< 9.81*e^{-7}$	0.9	$1.25*e^{-4}$	1.3	0.5	Clay

For the reason of numerical stability from these functions calculated K-values are limited down to $1.e^{-2}$ m/s.

Should other classes as the set parameters been used the definition has to be based on soil areas (element areas):

Then the attribute USAT assigns to each element in an area, which should be obtain a certain parameter combination, a zone identification number.

Each of these zone identification numbers uniquely identifies a set of Van Genuchten parameters. USAT needs to be fully defined. If the attribute USAT is assigned to all elements the parameters can be modified in SPRING with *Attributes → Special features → van-Genuchten parameters*. The following dialog box appears:
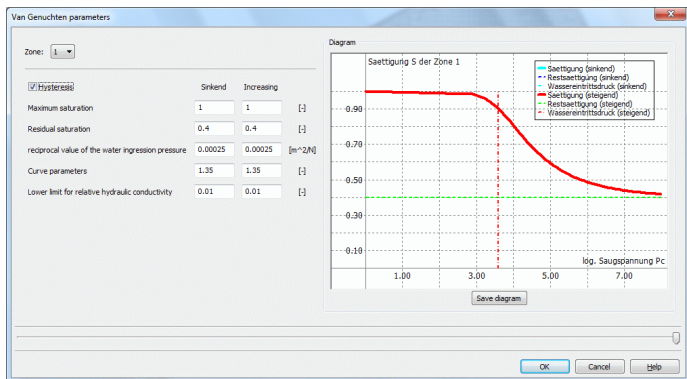
Fig. 35: Input of new defined van-Genuchten parameter

The parameters are stored in the model file in the following format:

```
Van Genuchten Parameter:
1 0.400000 0.000250 1.350000 0.010000 0.500000 1.000000
Hysteresis-Parameter:
1 0.400000 0.000250 1.350000 0.010000 0.000000 1.000000
```

The subsequent assignment of the attribute USAT in 3D models is performed automatically in the model checking.

For the flow and transport calculation in consideration of unsaturated circumstances for each element the saturation at the beginning (ASAT) can be defined. Then it is possible to choose that the initial value for the initial pressure is calculated by the saturation at the beginning. ASAT must be completely defined. In 3D-modells the later seizure if necessary executed in DADIA automatically.

For the definition of the Van-Genuchten parameters must the area defined with names for which the particular parameter set should obtain. Also USAT must completely defined.

The subsequent assignment for 3D models is performed automatically in the model checking.

3.2.2.14 Storage coefficients

Attributes: **SPEI** [-] (element wise) or **KSPE** [-] (node wise)

Meaning in horizontal models:

In case of a transient calculation, you can define a storage coefficient S [-] at each element (SPEI) or rather at each node (KSPE) of an unconfined aquifer. In a confined aquifer, the default storage coefficient $S_0 = S /$ thickness $= 3.3 \ 10^{-6}$ is used. The value can be changed in the file sitr.

Meaning in 3D models and vertical models:

Usually, the calculation of the specific pressure-storativity in the transient pressure equation (theory see p.):

$$S_s = S_r \ _s \left(\alpha(1-n) + \beta n \right) + \ _s \frac{\partial S_r}{\partial p}$$

Fig. 36: Transient pressure equation

uses for the variable n [-] the specified porosity (PORO). For the case either SPEI or KSPE is defined, this value is used instead of the porosity, otherwise, the porosity is initialized with 0.2.

For the transient flow computation the user has to decide which calculation module to use. The module SITRA requires node wise storage coefficients or the porosity. For SITRA, it is better to define the attribute KSPE. Otherwise, the element wise attribute SPEI must be averaged to node wise values. In contrast to SITRA the module INSTAT uses element wise storage coefficients or the porosity. Therefore, it is preferable to use SPEI.

3.2.2.15 Transmissivities

Attribute: **TRAN** [cm²/s] (horizontal or vertical models only)

The transmissivity is always assigned to elements. The unit is cm²/s. Anisotropic properties cannot be taken into account.

If column 15 contains an 'F', the transmissivity for the corresponding elements is fixed and cannot be changed during an iteration.

3.2.2.16 Impermeable layer

Attribute: **UNDU** [m] (horizontal models only)

By defining an impermeable layer, the thickness of the aquifer is limited, especially in the case of a transient calculation. The thickness of an impermeable layer is taken into account for the calculation of the initial and maximum thicknesses. Additionally, it is used together with the attribute LEEL (element wise leakage value) for calculation of the leakage coefficients. The impermeable layer is always assigned to nodes.

3.2.2.17 Unsaturated zone

See: Saturation parameters

3.2.2.18 Lower boundaries

Attributes: **UNTE** [m NN] (element wise), **UNTK** [m NN] (node wise), (horizontal models only)

The lower boundary is required to compute the horizontal free surface or the initial and maximum thicknesses together with upper boundaries, initial potential heads or ground levels. For an iteration of the thickness indicating the lower boundary is essential. The lower boundary is assigned to elements or nodes. The unit is m NN.

3.2.2.19 Zones

3.2.2.19.1 Zones of anisotropy (2D)

Attribute: **ZONE** [-] (zone number, horizontal model)

Within the mesh, zones can be defined that have anisotropic material properties. This attribute must be selected if anisotropic conductivities should be considered. Each zone is characterized by a positive integer which has to be assigned to each element of each zone. For each zone number K-values and angles must be defined within the material data (MATE). The zone numbers (ZONE) require necessarily the attribute MATE (material data).

If in the entire mesh only isotropic K-values with uniform anisotropy values are available, a zone number is not required. The value must be an integer greater than 0.

Example: see chapter "Material data for anisotropic zoning (2D)", p. 43.

3.2.2.19.2 Zones of anisotropy (3D)

Attribute: **Z-KA** [-] (zone number, 3D model)

With this type of data elements are assigned to different zones. For each zone the conductivities k1, k2, k3 and their corresponding angles a1, a2, a3 can be assigned via the menu item *Attributes → Special features → Anisotropic conductivities.* The value must be an integer greater than 0.

3.2.2.19.3 Zones of decay and production parameters

Attribute: **Z-AP** [-] (zone number, transport model)

With this data type the elements are attached to different zones.

For each zone number a decay rate or a concentration limit can be defined in the transport calculation. The value must be an integer greater than 0.

3.2.2.19.4 Zones of adsorption coefficients or thermal parameters

Attribute: **Z-KD** [-] (zone number, transport model)

With this data type the elements are attached to different zones. For each zone number an adsorption coefficient can be defined in the transport calculation. For the heat transport calculation a zoning of the matrix properties can be made. The value must be an integer greater than 0. The zone numbering must start with 1 and then numbered consecutively.

3.2.2.19.5 Zones of the inverse modelling

Attributes: **Z-KW** [-] (zones for K-values), **Z-SP** [-] (zones for storage coefficients),

Z-LR [-] (zones for polyline wise leakage coefficients)

With these attributes the data which should be calibrated during the inverse modelling are divided into different zones.

The attribute Z-KW divides all elements in zones for all permeability attributes such as KWER, KFWV, KWEV.

The attribute Z-SP divides all nodes in zones for all storage coefficients (SPEI or KSPE). The zoning is node-based, because the inverse modelling performs a calibration of the node-based storage coefficients.

The attribute Z-LR divides the polyline wise leakage coefficients (LERA) in zones. It is as relevant for the data type Z-LR as for the attribute LERA to keep the node sequence! The value must be an integer greater than 0.

For a more precise meaning of the zoning see the detailed description in the chapter "Inverse modelling".

3.2.3 Boundary conditions and source terms

This group includes all inflows and outflows, seepage, potential heads, concentrations, well withdrawals, and the like (see figure).
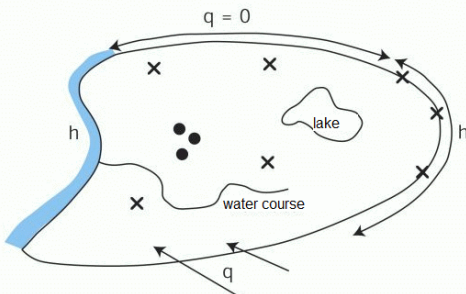
Fig. 37: Possible boundary conditions

Minimum required boundary conditions in a flow model:

For at least one node has to be determined either a potential head or a mass flow.

To this node a potential head (attribute POTE) or a lekage boundary (VORF and LERA or LEKN or LEEL in a neighboured element) can be assigned.

For the boundary nodes the following definitions are available:

■   To a boundary node the attribute POTE (potential) is assigned:

Then the node has a boundary condition of 1st order. (The data types KNOT (sink/sources) or RAND (on-/outflow rates) are not available for this node anymore; inflows FLAE (areal seepage) in horizontal or 3D models in adjacent elements are available.) Inflow or outflow rates across this boundary section are automatically computed from the mass balance.

■ The attribute POTE is not assigned to a boundary node:

Inflow or outflow rates across the boundary are assigned with the help of KNOT or RAND (boundary condition of 2nd order). In three-dimensional models areal recharge rates (FLAE) are boundary conditions of 2nd order as well. They define an inflow rate across the upper boundary of the model.

No inflow or outflow rates across the boundary are assigned by using KNOT or RAND (or FLAE in an adjacent element for the case of a 3D model). Any inflows and outflows are set to zero (impermeable or no flow boundary).

Regardless whether a potential head POTE was defined for a boundary node or not, the inflow from the top can be defined in a horizontal model with the help of FLAE

The nodes where none of the data types POTE or RAND or KNOT (or FLAE in case of a 3D model) is specified in the input file have a boundary condition of 2nd order i.e. the boundary is set impermeable (e. g. using sheet pile walls or the boundary is perpendicular to the potential lines).

Minimum required boundary conditions in a transport model

For the boundary nodes the following definitions are available:

■ A fixed concentration 1KON is assigned to the node (boundary condition of 1st order).

■ At nodes where either an inflow rate from RAND or KNOT or FLAE (in adjacent elements) is defined or a mass flow rate from a fixed potential head is computed or an inflow rate from leakage boundary conditions (LEKN or LERA or LEEL and VORF) exists an inflow concentration KONZ has to be specified. Otherwise it is automatically assumed as zero.

■ At boundary nodes where an outflow exists, natural boundary conditions are automatically defined, i.e. the dispersive flow across the boundary is set to zero.

3.2.3.1 Flooding parameters, mines

Attribute: **GRUB** [-] (zone number), retention water level elevation and additional quantity (3D models only)

For the simulation of the flooding of pit buildings are the pits marked with a region number (> 0) at the nodes.

For the simulation of the flooding of mines, are the mines to be marked with a zone number (> 0).

This number is then used to describe the retention-height-volume relationship.

If the attribute GRUB is assigned, the water level (HMAX) and the additional quantity (MENG) can be defined for each mine in this dialog box: *SPRING → Attributes →Special features → Flooding parameters*. At this point, the relation to the retention-height-volume-file grubeX.hv (in *.csv format) is made, too.

The csv-file should have the following format:

```
10.0,0.0
15.0,234900.0
30.0,511600.0
35.0,602800.0
50.0,783000.0
```

The attributes HMAX and MENG can be defined as transient boundary conditions as well.

3.2.3.2 Siphon lines

Attribute: **HREI** [-], (3D models only)

Waterworks can operate wells in so-called siphon lines. In this operating system is no individual control of the individual wells possible. In the operation of those wells under extreme conditions, it is possible that some wells of a siphon line fall below their allowed drawdown target by adjacent, deeper pumping wells thus cause the total failure of an entire series.

Polder wells are understood as a well, which meets a defined drawdown target, but under the restriction not to exceed a defined maximum possible amount of withdrawal - usually the limits of the pump.

If the free water level in one or more wells of a siphon line, e.g. due to adjacent cones of depression below the allowed drawdown target, the wells of the entire series are shut down or the allocated boundary conditions at the affected element nodes are undone (principle of "total failure"). The following figure shows the principle of a siphon line:
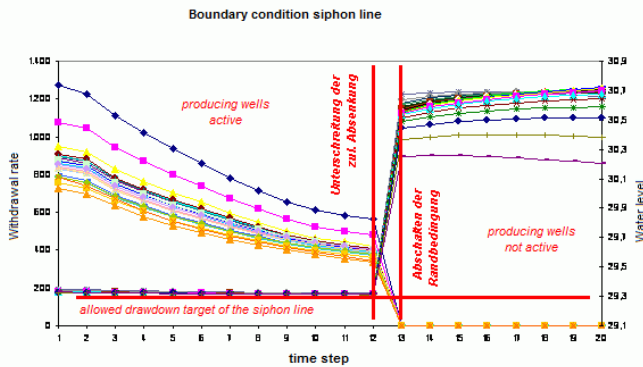
Fig. 38: Principle of a siphon line

The installation of such a siphon line in the groundwater model is as follows: Those mesh nodes which form a siphon line are united by assigning a unique group number (HREI) to each node. The allowed drawdown target is defined for each of these nodes as receiving water course level (VORF) with a quasi-infinite leakage coefficient (LERA). In addition, there is a restriction on the amount of withdrawal (MXKE), for example by the limit of the pump as well as a restriction of the infiltration rate (MXKI) to zero.

The latter restriction is therefore absolutely necessary because a node of the siphon line is model technically still a water course node (Cauchy boundary condition), so the leakage rate can be positive or negative and can therefore be an additional amount of water in the groundwater when the water level is below the water course level.

3.2.3.3   Concentrations or temperatures

The data types described in the following chapters are used for mass transport and heat transport calculation. The names in menus and dialogs are always the ones of the mass transport calculation. The user has to transfer the names for a heat transport calculation independently.

The unit of the concentrations attributes depends on the type of transport calculation:

- In a mass transport calculation using a tracer or a density-independent flow it is possible to enter the concentration with an "arbitrary" unit (i.e. [g/l], [[g/g], [mg/l], [kg/kg], etc.).

- In a density-dependent mass transport calculation the concentration has to have the unit [kg (solute)/kg (solution)].

- In a calculation with adsorption the concentration has to have the unit [kg/m³].

3.2.3.3.1   Initial concentrations/temperatures

Attribute: **AKON** = [Unit depends on the type of transport calculation.] (transport model only)

AKON defines the initial concentration/temperature for transient transport calculations. Initial concentrations/temperatures are always assigned to nodes.

3.2.3.3.2   Fixed concentrations/temperatures

Attribute: **1KON** = [Unit depends on the type of transport calculation.] (transport model only)

1KON defines a constant concentration/temperature at single nodes (boundary condition of 1st order). The boundary condition 1KON can also be defined as transient boundary condition (transient data 1KON).

3.2.3.3.3   Inflow concentrations

Attribute: **KONZ** = [Unit depends on the type of transport calculation.] (transport model only)

KONZ assigns concentration/temperature values to defined inflows (defined by KNOT, FLAE, RAND, RANQ, RANX) and computed inflows (leakage rates, mass flow rates at fixed potential heads). Inflow rates without any concentration specified have automatically the default concentration 0. The concentration is always assigned to nodes. The boundary condition KONZ can be defined as transient boundary condition as well (transient data KONZ).

3.2.3.3.4   Mass flow

Attribute: **QKON** = [mass/node/second] (transport model only)

The mass flow in-/output has to be in time unit "second"!

With the data type QKON, a fixed mass flow (mass per second) can be defined at individual nodes without the input of a corresponding volume flow (fluid). The boundary condition QKON can be defined as transient boundary condition as well (transient QKON)

3.2.3.4   Multi component parameters

Attributes: **1M***nr* [mass unit*/m³], **AM***nr* [mass unit*/m³], **KM***nr:* Inflow concentration (c_{nr}^{in}) [mass unit*/m³], **QM***nr:* mass in-/outflow ($q_{c,nr}$) [mass unit*/s] (transport models with module XTRA only)

For the calculation of transport processes with multiple components in SPRING with the module XTRA additional model data is available for the definition of inflow terms, boundary and initial conditions.

The substance number is to define in double digits in the model files (01, 02, 03, ... 99), such that the respective identifier is a total of 4 characters. The attributes are to handle analogical to the attributes for the transport calculations with only one substance.

(1Mnr →1KON, AMnr →AKON, QMnr →QKON and KMnr →KONZ).

The zone numbers Z-KDhave to be integer and positive. Elements with no Z-KD value are automatically assigned to the zone number 0.

*: The mass unit in which the concentrations of the substance are specified are defined in the initialization file xtra.ini!

3.2.3.5   Groundwater recharge

The groundwater recharge is determined of different factors according to the methods. The following figure shows the scheme and input data for calculating the groundwater recharge:
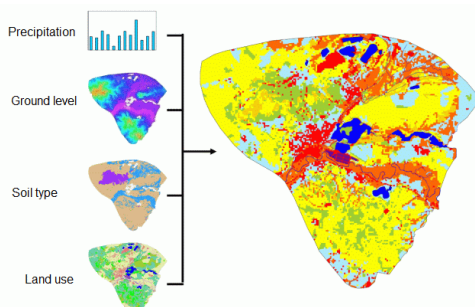
Fig. 39: Scheme of the groundwater recharge calculation

The following methods of calculating the groundwater recharge are used in SPRING:

- Method by Meßer
- Method by Schroeder&Wyrwich
- Method of soil water balance
- Method by RUBINFLUX

The methods are described in detail in chapter "How To - Calculating groundwater recharge rates" (p. **412**).

3.2.3.5.1   ATKIS data

Attribute: **NATK** [-], (horizontal and 3D models only)

A detailed classification of land use can be made on the basis of ATKIS data. These are provided through the attribute NATK for all elements. The used land use data key can be converted in the in SPRING required attribute NMFN (recharge calculation by Meßer, *Attributes → Computation → Groundwater recharge*).

3.2.3.5.2   Soil types, - classes

Attributes: **NMBT** [-] (method of by Meßer), **NSBT** [-] (method of Schroeder&Wyrwich), (horizontal and 3D models only), **BTYP** [-] (according to NSBT, required for the transient recharge calculation considering the soil water balance), **NKBT** [-] (method of RUBINFLUX)

Depending on the procedure to calculate the groundwater recharge, the elements are classified according to specific soil types. In the method of Meßer sandy or clay soils are differed depending on their effective field capacity, as well as loess and soils with stagnic properties. In the method of Schroeder & Wyrwich the soils are classified by terrestrial sand or clay soils and semi-terrestrial soils.

In detail the methods are described in the chapter calculation of the groundwater recharge.

The following figure shows the distribution of soil types in a model:

Fig. 40: Distribution of soil types in a model

3.2.3.5.3   CORINE land use data

Attribute: **NCLC** [-], (horizontal and 3D models only)

A detailed classification of land use can be made on the basis of CORINE land use data. These are provided through the attribute NCLC for all elements. The used land use data key can be converted in the in SPRING required attribute NMFN (recharge calculation by Meßer, *Attributes → Computation → Groundwater recharge*).

3.2.3.5.4   Run off by Meßer

Attribute: **NMAD** [m/TU], (horizontal and 3D models only)

The calculation of the direct runoff by Meßer is based on the slope classes, soil types, the land use and the classes of the distance between the ground – and the groundwater level.

The calculation is based on dbase files which are stored during the installation of SPRING in the directory "C: \ Users \ Public \ Documents \ SPRING \ Konfig" (on Windows 7). This path varies depending on the operating system or the hard drive partitioning (e.g. under Windows XP: "C:\Documents and Settings\All Users\Documents\SPRING\Konfig")

Details of the calculation method are described in the chapter „How To – Calculation of the groundwater recharge".

3.2.3.5.5   Land use

Attributes: **NMFN** [-] (method of Meßer), **NSFN** [-] (method of Schroeder&Wyrwich), (horizontal and 3D models only), **NUTZ** [-] (according to NSFN, required for the transient recharge calculation considering the soil water balance), **NKFN** [-] (method of RUBINFLUX)

Depending on the procedure to calculate the groundwater recharge, the elements are clas-sified according to the specific land use. The method of Meßer divides into fields, forests, water areas and dumps and especially divides the building development after its percentage degree of sealing.

The method of Schroeder & Wyrwich classifies into field, forest types, building development and water areas.

Details of the calculation methods and their parameters are described in the chapter „How To – Calculation of the groundwater recharge".

For large parts of the Ruhr area exist detailed digital land use data from the Re-gionalverband Ruhr (RVR) as the so-called *type of use catalog of the land use map.*

This land use data key can be converted in the in SPRING required attributes NSFN (recharge calculation by Schroeder&Wyrwich) or NMFN (recharge calculation by Meßer).

For this the RVR-Code has to be provided by the element attribute KVRN.

When using the standardized land use codes such as CORINE land use data and ATKIS data it is also possible to convert them in the divisions given by Meßer for the attribute NMFN. The following figure shows the distribution of land use in a model:
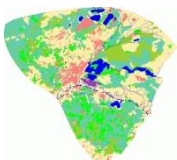
Fig. 41: Distribution of land use in a model

3.2.3.5.6 Areal seepage (recharge rate)

Attribute: **FLAE** [see below] (horizontal and 3D models only),

The data type defines inflow and outflow rates which refer to elements or rather areas. For example, recharge or exfiltration rates can be assigned to elements where inflow is positive or negative, respectively. The boundary condition FLAE can also be defined as a transient boundary condition (transient FLAE data and EFLA data).

Meaning of column 16 in the model file:

Blank = the unit of the value is m³/element/TU (TU = time unit), the value is assigned to each element in the following list as explicit inflow / outflow rate.

'*' = the unit of the value is m³/m²/TU, the value is assigned to each element in the following list as inflow / outflow rate per m² (i.e. infiltration rates, recharge).

'/' = the unit of the value is m³/all elements of the list/TU, i.e. the value is a total value which is distributed to all elements in the following list in proportion to their element areas

Example:

```
FLAE recharge rate
        0.50 *    1-    10
      -20000.    11-    14
    50000.00 /   15-    19
```

Elements 1 to 10 have a recharge rate of 0.5 m³/m²/TU from precipitation. Elements 11 to 14 are withdrawal wells, where each well extracts 20,000 m³/TU from the aquifer. Elements 15 to 29 are infiltration basins with a total rate of 50,000 m³/TU. The total amount is distributed to the single element areas.

3.2.3.5.7 Distance between ground – and water level

Attribute: **FLUR** [m]

The node attribute FLUR is required to determine the code for depth to water table in the recharge calculation by Meßer. The attribute can either be read from the result data of the flow calculation or determined directly by calculating the ground level with the calculated / measured water table.

3.2.3.5.8 Code for depth to water level

Attribute: **NMFK** [-] (method of Meßer only), (horizontal and 3D models only),

The elements are divided into four different classes according to their depth to water level. The division of the classes is described in detail in chapter "calculation of the groundwater recharge" (p. **414**) .

3.2.3.5.9 Surface gradient

Attribute: **GGRD** [%] (horizontal and 3D models only)

The surface gradient is calculated for all elements via the menu *Attributes → Computation → Surface gradient*. The unit is [%].

3.2.3.5.10 Incline of ground

Attribute: **NMGK** [-] (method of Meßer only), (horizontal and 3D models only)

The elements are divided into four different classes according to their relief energy. The division of the classes is described in detail in chapter calculation of the groundwater recharge. The classes of incline of ground are based on the digital ground level model. The following figure shows an example of the ground level of a model:
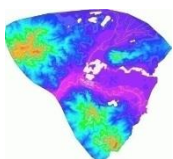
Fig. 42: Ground level of a model

3.2.3.5.11 Latitude

Attribute: **NKBR** [°] (method of RUBINFLUX) (horizontal and 3D models only)

The latitude is required for the calculation of the reference grass evaporation. The unit is [°].

3.2.3.5.12 Climate classes, zones

Attribute: **NMKL** [-] (method of Meßer only), **NKID** [-] (method of RUBINFLUX), (horizontal and 3D models only)

According to Meßer the elements are divided into six climatopes depending on their potential evapotranspiration (attribute NETP). According to RUBINFLUX the IDs for the associated climate data are assigned to the elements. The division of the classes is described in detail in chapter "Calculation of the groundwater recharge"(p. 412).

3.2.3.5.13 Layer height

Attributes: **NKLH** [mNN, mNHN, or mamsl] (method of RUBINFLUX) (horizontal and 3D models only)

The layer height can be derived in SPRING from the node attribute GELA via the menu item *Attributes →
Copy → Node -> element...*

3.2.3.5.14 Precipitation

Attribute: **NIED** [m/TU or m³/m²/TU], (horizontal and 3D models only)

The data type NIED requires the time unit „Jahr" (= year) in the model file!

The data type precipitation is assigned element wise.

For the conversion of the total precipitation on the individual elements the input "*" in column 16 of the model file is required. In general the precipitation data is provided in Excel files. The following figure shows an example of a precipitation distribution:

Fig. 43: Distribution of precepitation

3.2.3.5.15 RVR classes

Attribute: **KVRN** [-], (horizontal and 3D models only)

For a detailed classification of the land use the so-called *type of use catalogue of the land use map fr*om the Regionalverband Ruhr (RVR) can be provided.

This land use data key can be converted in the in SPRING required attributes NSFN (recharge calculation by Schroeder&Wyrwich) or NMFN (recharge calculation by Meßer, *Attributes → Computation →Groundwater recharge*).

For this the RVR-Code has to be provided by the element attribute KVRN.

3.2.3.5.16 Rapid runoff

Attributes: **NKAV** [vol.-%], **NKAG** [m/TU] (method of RUBINFLUX) (horizontal and 3D models only)

The attributes NKAV (sealing dependent outflow component A_{VERS}) and NKAG (rapid runoff component of soil-dependent parameters A_{SCS}) resulting from transient recharge calculation and can be read as result data during the calculation.

3.2.3.5.17 Evapotranspiration, potential

Attribute: **NETP** [m/TU], (horizontal and 3D models only)

To divide the elements in climate zones (attribute NMKL) the potential evapotranspiration has to be specified for each element.

Details of the calculation method are described in chapter „How To – Calculation of the groundwater recharge".

3.2.3.5.18 Evapotranspiration, actual

Attribute: **NMET** [m/ZE], (horizontal and 3D models only)

The actual evapotranspiration by Meßer is calculated by the climate zones, soil types, depth to the water level and land use. The calculation is based on dbase files which are stored during the installation of SPRING in the directory "C: \ Users \ Public \ Documents \ SPRING \ Config" (on Windows 7 or Vista). This path varies depending on the operating system or the hard drive partitioning (e.g. under Windows XP: "C:\Documents and Settings\All Users\Documents\SPRING\Konfig"). Details of the calculation method are described in chapter „How To – Calculation of the groundwater recharge".

3.2.3.5.19 Surface sealing

Attribute: **VERS** [%], (horizontal and 3D models only)

The calculation of the soil water balance requires the sealing data from the element attribute VERS in %.

For the calculation of the groundwater recharge by Meßer or RUBINFLUX a sealing component is asked for the mixed vegetation classes of the land use, but within the conversion of the input data (RVR code, CORINE land use or ATKIS) in land use by Meßer the sealing component is program internally calculated.

3.2.3.5.20 Water content of the soil

Attributes: **NKFK** [vol.-%], **NKWP** [vol.-%], (method of RUBINFLUX) (horizontal and 3D models only)

A distinction is made between the water content at field capacity (NKFK) and the water content at permanent wilting point (NKWP). A detailed description of these data is in chapter „How To – Calculation of the groundwater recharge: Transient recharge" (p. 412).

3.2.3.6 Potential heads

3.2.3.6.1 Water course level, tear-off height at water courses

Attributes: **VORF** [m NN], **ABRI** [m]

The constant potential head of the receiving water course is required to compute the potential difference h_{vorf} - h and the corresponding leakage rate(linear relationship). Water course levels are always assigned to nodes.

'F' in column 15 means that the receiving water course level is not considered in the calculation with terrain settings. (In case of a transient calculation with transient terrain settings the water course level is charged with the transient settings. The character 'F' is in this case meaningless.) The boundary condition VORF can also be defined as transient boundary condition (transient VORF data).

The attribute VORF is effective only in conjunction with a leakage coefficient

Attribute **ABRI:**

To take reduced infiltration conditions into account a value can be defined at a node at which the maximum possible amount of infiltration is achieved, i.e. the flow stops. This value is obtained when the groundwater level (h_{GW}) falls below the difference bottom height (h_{bottom}) - ABRI. In this case the leakage rates are calculated on the basis of the potential difference of h_{Vorf} - $h_{bottom,}$ in this h_{bottom} corresponds to the ground level.

ABRI conditions can also be defined as transient conditions (transient ABRI data).

The following representation shows the relationship between the height of the water level and the tear-off height:
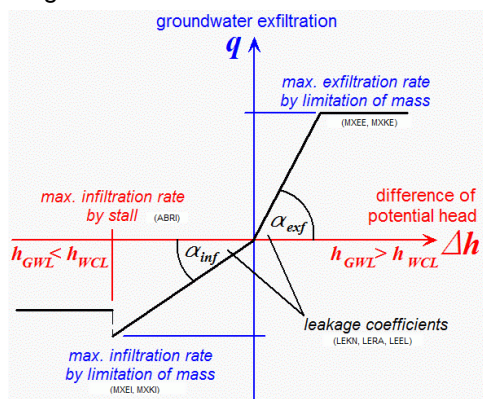
Fig. 44: Relationship between the height of the water level and the tear-off height

3.2.3.6.2   Initial potential heads

Attribute: **EICH** [m NN]

The data specifies potential heads at nodes as a nominal value. The initial potential heads are used for the iterative fitting of permeabilities (calibration) and in some cases for the computation of the initial thickness.

The initial potential heads can be used as potential heads of the initial state at the time t = 0 in transient calculations.

3.2.3.6.3   Fixed potential heads

Attribute: **POTE** [m NN]

Potentials can be prescribed at each node. Usually this makes only sense, where the inflow and outflow quantities are not precisely known (e.g. at the boundary nodes or at receiving water courses with a strongly developed groundwater contact).

The boundary condition POTE can be defined as transient boundary condition as well (transient data for POTE).

For each flow calculation a fixed potential head must be defined on at least one node, otherwise the system of equations is singular (the physical problem is not clearly defined).

3.2.3.6.4   Equal unknown potential heads

Attribute: **GLEI** [-]

To determine the position of a potential line (groundwater isoline), whose value is unknown, the attribute GLEI is assigned to nodes, which receive in the modelling process all the same potential value.

- In the case of vertical models, the definition of unknown potential head is useful with pumping wells where the pumping rate is defined or should be varied. All nodes along the well beneath the water table should have the same potential head. The attribute GLEI with value = 0. is assigned to these nodes.
- In case of horizontal models, GLEI can be assigned to the shore nodes of a lake where the water table depends on the ground water table.
- In 3D models, GLEI is used for fully penetrating wells, because the potential head must be equal over the full length of the well.
- In partial 3D models, along the border between the 2D and the 3D part, unknown equal potential heads are used to define equal potential heads over the complete depth (see chapter Creating a 2D model combined with a partial 3D model).

All nodes of one list have the same potential head. A list is complete, if no continuation line follows (blank in column 80).

The value in columns 1 to 14 can be used to define a constant potential difference between two nodes. The potential heads remain unknown. However, they are determined so that the difference in the potentials is equal to the prescribed value.

Example:

```
GLEI ks3 : nodes with equal unknown potential heads
      0.00000       64-    65,    67,    66,    69,    68,    71-    73,
      0.00000       75,    74,
```

Nodes 64 to 69 and 71-73 have the same potential head. Nodes 75 and 74 also have the same potential head, but the value is not necessarily equal to the previous list. Thus several different sets of data can be entered.

3.2.3.7   Seepage nodes

Attribute: **SICK** [-] (vertical and 3D models only)

This data set contains the node numbers along a possible free seepage face. Entering a value (column 1 to 14 of the model file) is not required.

The point where the free surface of a well or a dam emerges to the atmosphere naturally lies higher than the water table. The distance between these two points is referred to as free seepage.

The free seepage is determined automatically when the attribute SICK is assigned to the nodes of the edge area, in which the exit point of the free surface is. During the calculation, the seepage nodes are at first nodes with a fixed potential head (potential head = geographic height). The calculation module switches them iteratively to nodes without inflow (q = 0 boundary) in the moment the potential head boundary condition leads to an inflow rate.

In case of a transient calculation, the seepage nodes can be switched on and off for different time steps (transient data for SICK).

Attention: Currently, the seepage boundary condition is only available in the module SITRA!

3.2.3.8 Temperatures

For the heat transport the attributes of the concentrations are used.

Attribute: **AKON** [K] = initial temperatures

In the case of transient heat transfer calculations it is necessary to specify the data type AKON as an initial state. The attribute AKON is always assigned to nodes.

Attribute: **1KON** [K] = fixed temperatures

1KON defines a constant temperature at single nodes (boundary condition of 1st order). The boundary condition 1KON can also be defined as transient boundary condition (transient data 1KON).

Attribute: **KONZ** [K] = Inflow temperatures

KONZ assigns temperature values to defined inflows (defined by KNOT, FLAE, RAND, RANQ, RANX) and computed inflows (leakage rates, mass flow rates at fixed potential heads). Inflow rates without any temperatures specified have automatically the default temperature 0 K. The temperature is always assigned to nodes. The boundary condition KONZ can be defined as transient boundary condition as well (transient data KONZ).

3.2.3.8.1   Heat production rate

Attribute: **QKON** [W] (heat transport models only)

The heat production rate results from the amount of heat that is generated in 1 m³ of material in a second (eg, by friction or radioactive decay). In SPRING QKON is entered in the unit [W], which refers to 1 m³ surrounding material (water and matrix).

With the data type QKON a fixed rate of heat production can be defined at individual nodes. The boundary condition QKON can be defined as transient boundary condition as well (transient QKON data).

3.2.3.9 Deformation parameters

Attributes: **RLAX** (edge load in X-direction), **RLAY** (edge load in Y-direction), **VERX** (solid deformation in X-direction), **VERY** (solid deformation in Y-direction)

The coupled flow-crack propagation calculation is performed using the two software systems FRANC2D and SPRING.

3.2.3.10 Infiltration basin

Attribute: **MULD** [m²] (horizontal and 3D models only)

It is an attribute that is used in the calculation of the transient recharge rates. The given value indicates the sealed area in square meters, which is connected to the infiltration basin. Here, a basin can consist of several elements to which the specified area is then distributed.
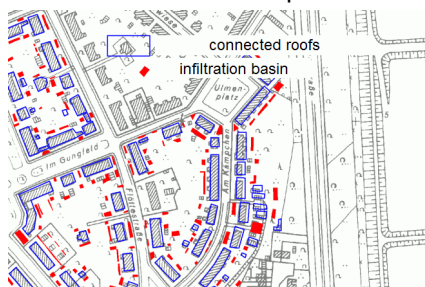
Fig. 45: Use of the attribute MULD

The presentation highlights the use of the attribute MULD: The roofs and other sealed areas of a plot of land (blue) drain their precipitation water off to the corresponding infiltration basins (red).

3.2.3.11 Water course parameters for the simulation of the interaction between groundwater and surface water

Attributes: **VKST** (Manning/Strickler coefficient [m$^{1/3}$/s]), **VBRT** (wetted perimeter [m]), **VGRD** (bottom slope [pro mille]), **VKNO** (inflow at a water course node [m³/TU]), **VINZ** (incidences of stream edges [-]), **VSYS** (incidences of the water systems [-]), (horizontal and 3D models only)

During the flow simulation SPRING can perform an automatic estimation of the water course levels (steady-state and transient).

For this, individual water segments or the entire water system are to be parameterized with the wetted perimeter (VBRT), the bottom slope (VGRD) and the Manning-Strickler coefficient (VKST). These three parameters are needed to calculate the water levels in the water courses. They each relate to the edge of an element. The parameter VKNO is an additional inflow at nodes of a water system. He is taken into account in determining the flow rate, the amount is based on the time unit of the model file. VKNO can be defined as transient boundary condition as well.

The saving of the water systems is done analogously to the 1D fracture systems.

The attribute VINZ sets up an incidence table for the affected edges of the elements, which in turn are grouped by the attribute VSYS to water systems. The attribute assignment is edge-wise, that is for the 1D elements stored in VINZ.

There is an automatic allocation of the quantities in equal proportions in flow direction or the proportions can be defined in the file sitra_vsys.txt (p. 24).

3.2.3.12 Inflow, nodewise

Attribute: **KNOT** [m³/TU]

Inflow / outflow rates refer to single nodes e. g. a well. Inflow rates are greater than 0, and outflow rates are less than 0. The boundary condition KNOT can be defined as transient boundary condition as well (transient KNOT data). In a 3D model is to make sure that the attribute KNOT is always set in the saturated region. This can be ensured by using same potentials (attribute GLEI).

Meaning of column 16 in the model file:

- Blank = unit m³/node/TU, i.e. the value is assigned to all nodes in the following list as explicit inflow / outflow rate.
- '/' = unit m³/all nodes/TU, i.e. the value is a total value which is distributed uniformly to all nodes of the list.

Example:

```
KNOT in-/outflow
        150.       80,     97-     99
       -200. /   106,    105,
```

The following values are assigned to the nodes: 150 m³/TU inflow at the nodes 80, 97, 98, 99; -200 / 2 = -100 m³/ZE outflow at the nodes 105 and 106.

3.2.3.13 Inflow/Outflow rates along lines

Attributes: **RAND** (rate (m³/TU) per m or m²), **RANQ** (rate (m³/TU) for the complete polyline), **RANX** (rate (m³/TU) per m(x0) or m(x)*m)

With the attributes RAND, RANQ and RANX along a line in-and outflows can be set. Inflows are positive, outflows are negative.

Meaning of column 16 in the model file:

'*' = the transformation of the rate to node wise values considers the thickness (at present only for 2D models).

RAND without '*':

- The value represents a rate per meter along the polyline. The distance is obtained from the distance between the nodes. The corresponding inflow / outflow rate is computed for each node depending on the part of the polyline which corresponds to the node.

RAND with '*':

- The value represents a rate per m². The area is obtained from the distance between the nodes multiplied by the width of the model (thickness). The thickness is averaged from the edges to the adjacent elements.

The corresponding inflow / outflow rate is computed for each node depending on the part of the area which corresponds to the node.

RANQ without '*':

■ The value represents a total rate for the polyline (m). The sum is distributed to the nodes of the polyline corresponding to the partial distances.

RANQ with '*':

■ The value represents a total rate for an area (m²). The area is obtained from the complete distance multiplied by the width of the model (thickness). The sum is distributed to the nodes of the polyline corresponding to the partial areas (distance * thickness).

RANX without '*':

■ The value represents a rate per meter x-coordinate. In contrast with RAND, the distance is computed using only the x-coordinates of the nodes. The corresponding inflow / outflow rate is computed for each node depending on the part of the polyline which corresponds to the node.

RANX with '*':

■ The value represents a rate per meter x-coordinate * m. In contrast with RAND, the distance is computed using only the x-coordinates of the nodes multiplied by the width of the model (thickness). The thickness is averaged from the edges to the adjacent elements. The corresponding inflow / outflow rate is computed for each node depending on the part of the area which corresponds to the node.

The following figure shows a part of a model with different boundary inflows. The section of the *.net file refers to this figure.

Example:
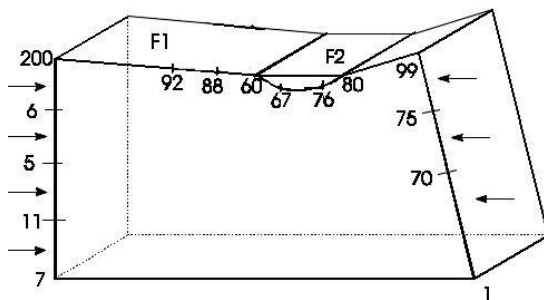
Fig. 46: Boundary in-/outflow

```
RAND
        20.00       200,       6,       5,      11,       7
         0.50 *     200,      92,      88,      60,
RANQ
       300.00         1,      70,      75,      99,
RANX
         1.00 *      60,      67,      76,      80,
```

Boundary inflow over the left edge (nodes 200, 6, 5, 11 and 7) of 20 m³/m/TU.

Areal seepage for the area F1 with 0.5 m³/m²/TU. The area is computed from the distance between the nodes from 200 to 92 multiplied by the width (thickness).

Boundary inflow over the right edge (nodes 1, 70, 75, and 99) with a total rate of 300 m³/TU.

Areal seepage for the area F2 (nodes 60, 67, 76 and 80) of 1.0 m³/m²/TU. F2 can represent the water surface of a river where the infiltration rate is often defined in m³/km/TU. This unit is transformed into m³/m²/TU.

Nodes 200 and 60 automatically have two inflow parts (node 200 from two RAND definitions, node 60 from both a RAND and a RANX definition).

3.2.4 Auxiliary data for further analysis and presentation

To this belong markings in the plots and lines of text as well as node and element-related information for further evaluation of the calculated results

3.2.4.1 General node- and element wise data

Attributes: **KKKK** (node data), **EEEE** (element data)

These data types can be used for any data; for example, for graphic representation or for linking different data. The corresponding data are written into the binary files, i.e. they are recorded in the model checking.

3.2.4.2 Balance areas

Attributes: **BILK** [-] (node wise), **BILE** [-] (element wise)

Only the integer part of the value (columns 1 to 14) is interpreted. The list of numbers starts in column 17. Each closed list of numbers represents a mass balance group.

BILK requires nodes which have a fixed potential head and / or a leakage rate and / or a seepage boundary condition. As a result the mass flow rates / leakage rates / seepage rates at the specified nodes and the total mass flow rate are listed in the output file. Using different lists it is possible to define several balance areas (groups). The modules ignore the nodes in the list which do not have the necessary boundary condition. The different balance areas are then identified by the group number that is printed to the output file.

BILE requires elements which have a leakage boundary condition. As a result, the leakage rates for the specified elements and the total mass flow rate are listed in the output file. Using different lists it is possible to define several balance areas (groups). The modules ignore the elements in the list which do not have a leakage boundary condition. The different balance areas are then identified by the group number which is printed to the output file.

Balance areas are required where several calculations are compared by mass balances for certain boundaries (potential head), receiving water courses or element leakage areas.

3.2.4.3 Control lines

Attribute: **KONT** [-]

In columns 1 to 14, no value is required.

The list of numbers starts in column 17. Each closed list represents a control line.

With the help of control lines, certain mass flow rates across element boundaries and diagonals can be computed. The user defines the lines by a list of node numbers where the control line passes through. A control line can be closed by returning to the starting point. As a result, the user obtains mass flow rates across the single boundaries and the total mass flow rate across the complete control line. For instance, the amount of water that reaches an exfiltration well from a certain catchment is determined. The number of control lines is not limited.

3.2.4.4 Markings

Attribute: **MARK** [-]

With the attribute MARK, individual nodes or element centers are provided with symbols or connected by a line. Markings on element centers can be assigned only by editing the *.net file or the file *.3d. The entered value has to be negative.

The definition of markings requires an integer number. Hence follows that only the integer part of the value is interpreted. The identification numbers have the following meaning:

Value < 0: the list of numbers contains only element numbers, the markings are placed into the element centre.

Value > 0: the list of numbers contains only node numbers.

Single nodes or element centres can be marked by a symbol or connected by a line.

The points are connected according to the order in the input file. A closed list of numbers represents one line.

Value +/-1 → Quadrangle at a node / in an element centre (⊓)

Value +/-2 → Octagon at a node / in an element centre (⊘)

Value +/-3 → Triangle at a node / in an element centre (△)

Value +/-4. → Cross at a node / in an element centre (+)

Value /-10 → connection line between nodes / elements

Value /-(100+X) → PLX marker type X at a node / in an element centre

Of the 36 possible marker symbols only the first 4 are shown directly with a value MARK = 1, 2, 3 or 4 in the user interface or after the plot generation (see figure below):
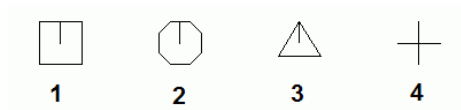
Fig. 47: Marking symbols with the value MARK = 1, 2, 3 or 4

The others have to have the value MARK = 100+X (see figure below):
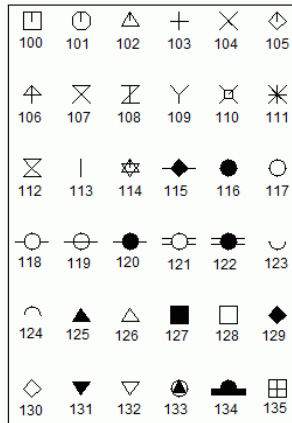
Fig. 48: Possible markings in SPRING

The node or element markers can be changed directly in SPRING. For this the appropriate toolbar first has to be displayed by the menu item *View → Toolbars → Object attributes*. By activating the marker selection all possible types of markers appear from which the desired type is selected. After activating the button *Assign object attributes* and selecting the appropriate node the marker is assigned to the node or an existing marker at the node is changed.

Example:

```
MARKINGS
        -3.0     111-   114,     14
       133.0      99
         1.0      23-    25
       10.01      23-    25,     33
       10.02      99,    88,     75
```

Elements 111, 112, 113, 114, 14 are marked by a triangle. Node 99 is marked by a polymarker of type 33 (filled triangle in a circle). Nodes 23, 24 and 25 are marked by a quadrangle and connected by a line. Nodes 99, 88 and 75 are also connected by a line.

3.2.4.5 List of boundary potential heads

Attribute: **PRAN** [-]

With the help of PRAN, a list of node numbers is specified. These numbers are printed together with the corresponding computed potential head to the output file. The format is equivalent to the model file (*.net). The data type is required for modelling a small part of a large model, where the results from one model must be transferred to the other.

3.2.4.6 Texts

Attributes: **KTXT** (node wise), **ETXT** (element wise)

With the help of these data, comments consisting of a maximum of 20 characters at nodes or elements are possible. The input format is different from the other data. Each line contains a node / element number (I10)

and the corresponding text (A20). SPRING interprets the comments as attributes. Thus they can be helpful during editing the input data. The plot module is able to plot the node and element text.

3.2.5 Transient data

In the transient input files only the data types can be defined which are listed in the following chapters.

3.2.5.1 Tear-off height

Attribute: ABRI

Details are described in the chapter transient VORF data.

3.2.5.2 Terrain settings

Attribute: BERG

The data type defines transient terrain settings caused by mining. Nodes without a value defined for single time steps automatically receive the value 0.0. Transient terrain settings are automatically interpolated at intermediate time steps. The values are subtracted from lower boundaries, z-coordinates, ground levels and receiving water course levels.

The definition of transient terrain settings always refers to the initial time point of the transient calculation. Terrain settings from 10.0 m to 9.0 m and then to 7.0 m are equivalent to an absolute settlement of 1.0 m (10.0 - 1.0 = 9.0) and then 3.0 m (10.0 - 3.0 = 7.0).

3.2.5.3 Areal seepage

Attribute: EFLA, unit: [m³/element/TU]

Already defined values for EFLA can be changed. New values can be specified. This is equivalent to FLAE in the model file without a '*' in column 16.

Attribute: FLAE, unit: [m³/m²/TU]

Already defined values for FLAE can be changed. New values can be specified. This is equivalent to FLAE in the input file with a '*' in column 16.

The unit of time refers to the unit in the transient input file.

3.2.5.4 Flooding parameters

Attributes: HMAX [m], MENG [m³/TU], HT_0 [m]

The values for flooding parameters (only applicable with SITRA) are changed with the identifier HMAX for the water level definition and with MENG for the inflow volume.

This should only be performed if the retention water level elevation and the additional quantity are defined for the corresponding mine already in the model file.

To identify the mine, to which the change should apply, the number of GRUBinstead of a node or element number has to be defined. If a water level definition should be removed, the area number must be indicated by a negative value. The value plays no role, but it has to be given because of the format constancy.

A detailed description is found in "How To – Simulation of flooding processes" (p. 436).

The attribute HT_0 can be specified in the first time step of the transient calculation. It defines an initial water level of the mine.

3.2.5.5 Inflow/outflow at nodes

Attribute: KNOT, unit: [m³/node/TU]

The input values refer to the time unit defined for rates in the transient input file. It can be different from the one in the model file. The value 0 has to be assigned to a node explicitly. If there is no input value, the program by default uses the value of the former time step.

3.2.5.6 Concentrations/Temperatures

In the transient input file, the attributes 1KON, KONZ and QKON can be given. The unit is: [?/node]. The "?" refers to the concentration unit in the model file: if the concentration is given in [kg / kg], the unit of the concentration in the transient input file has to be the same. When calculating the heat transfer the unit is always [K].

Fixed concentrations:

Attribute: 1KON [see above]

Values for fixed concentration can be created or changed. It is not possible to switch off fixed concentration conditions (like potential heads)!

Inflow concentrations

Attribute: KONZ [see above]

Values for inflow concentrations/temperatures can be created or changed. Inflow concentration or temperature conditions are switched off by assigning the value 0.

Mass flow

Attribute: QKON, unit: [mass/node/second]

Values for mass flow concentrations can be created or changed. Mass flow concentrations are switched off by assigning the value 0.

The mass flow has to be specified in the time unit "second" (independent of the time unit "MENG", which is defined in the header of the transient file)!

Heat production rate

Attribute: QKON, unit: [W/node]

Here the values for the heat production rate can be set for the first time or can be changed. The value has always the unit [W], which refers to 1 m³ surrounding material (water and matrix), independent of the time unit of the transient input file. The heat production rate at a node can be switched off, when assigning the value 0 to that node.

3.2.5.7 K-values

Attributes: KWER, KWEV, unit: [m/s]

With the attribute KWER a new horizontal K value is assigned to an element. Than this K value is isotropic: kxx=kxy=const, angle = 0 (also when with MATE and ZONE anisotropic K values were entered before).

With the attribute KWEV a new vertical K value is assigned to an element in a 3D model.

3.2.5.8 Leakage limitations

Attributes: KMKI [m³/TU/m], KMKE [m³/TU/m²], (polyline, nodewise)

If maximum infiltration or exfiltration rates for leakage relationships are defined in the model file (attributes MXKI or MXKE), they can be changed via the transient attributes KMKI or KMKE in the transient input file.

3.2.5.9 Leakage coefficients

Attribute: LEKN, unit: [m²/node/TU]

The data of the node wise leakage coefficients used for the leakage calculation can be changed. Entries are made in the defined unit of time for the quantities. This can be different from the time unit in the model file.

3.2.5.10 Thicknesses

Attributes: KMAE, MAEC, unit: [m]

For 2D models which are calculated with the module INSTAT transient element thicknesses can be determined with the attribute MAEC. Is a 2D model calculated with the module SITRA the attribute KMAE has to be used for the transient thickness of nodes.

3.2.5.11 Upper boundary

Attribute:: KOBE, unit [m NN]

The transient attribute KOBE is the node-specific upper boundary and can only be used in the transient calculation with the module SITRA.

3.2.5.12 Potentials, fixed

Attribute: POTE, unit: [m NN]

First, already existing nodal values of fixed potential heads (of the model file) can be reassigned. Secondly, fixed potential heads can be assigned to nodes which had no boundary condition before. Thirdly, it is possible to switch off a defined potential head boundary condition for single nodes. The latter is done by a negative node number and a suitable value for the potential head. Later the boundary condition can be switched on again by a positive node number and a corresponding value.

A calculation where transient boundary conditions are interpolated (Time Step Management) has to follow certain rules:

- Switching off an existing fixed potential head requires a suitable potential head assigned to the negative node number. The potential head is necessary to interpolate the intermediate points of time between switching on and off the boundary condition. For example the transient input file contains the following data:

      Point of time 10   POTE 10.0   1 (on)
      Point of time 20   POTE   5.0   -1 (off)

  A calculation with interpolation for POTE uses for the intermediate time steps 12, 14, 16 and 18 the interpolated values 9.0, 8.0, 7.0 and 6.0 m. The boundary condition is not switched off until time step 20.

- Switching the fixed potential head on after it was switched off will switch on the corresponding boundary condition on at the specified time step. There is no interpolation between the old value when switching off and the new one.

3.2.5.13 Seepage boundary condition

Attribute: SICK, unit [-]

Already existing seepage boundary conditions (defined in the model file) can be switched off by assigning a negative node number. It is possible to define new seepage boundary conditions by assigning a positive node number. The corresponding value is not necessary and will be ignored. The attribute SICK does not make sense in 2D horizontal models.

3.2.5.14 Water course level, tear-off height

Attributes: VORF, ABRI, unit: [m]

The water course levels used for the leakage calculation can be changed. Also it is possible to switch off a water course level at a node. This is done with a negative node number. Than the leakage node coefficient for this node is set to 0.0 in the calculation program. In the calculation with interpolation of the transient boundary condition VORF the leakage coefficient is incremental set to 0.0. It is not possible to switch on such a water course node again (in opposite to POTE) because the information about the "old" leakage value is lost!

The maybe in the *.net* file given subsidences caused by mining are not considered for the water course levels in the transient input file! So the input is equal to an input in the model file with "F" in the 15th column. (This does not deal with transient subsidences caused by mining!)

Should tear-off heights be changed only the node number and the new value has to be defined.

3.2.5.15 In- or outflow at a water course node

Attribute: VKNO [m³/TU]

The inflow or outflow at a water course node can be changed in a transient calculation. The value 0 has to be assigned to a node explicitly. If there is no input value, the program by default uses the value of the former time step.

3.2.6 Required data and combinations of data

Each calculation requires data of the first three groups: The mesh has to be defined, a transmissivity has to be assigned to each element and at minimum to one node a potential head or a leakage boundary condition has to be assigned, so that the system of equations is not singular.

There are several input possibilities which can be different in each element. For the case of a vertical or a horizontal model these are:

1. K-values and thicknesses

2. K-values and lower and upper boundaries of the aquifer (horizontal model)

These two possibilities are suitable for isotropic material. In the second case the thickness is automatically obtained from the lower and the upper boundary.

3. Transmissivities with/without thickness or K-values.

If no thickness is specified, the computation of velocities uses a constant thickness of 1.0 m.

4. Material data, zone numbers, thicknesses

5. Material data, zone numbers, lower and upper boundaries (horizontal model).

The combinations 4. and 5. are suitable for anisotropic material. A zone number is assigned to each element. The material data (k_{fx}, k_{fy}, angle).are then assigned to the corresponding zones).

6. K-values, lower boundaries and initial potential heads (horizontal model)

7. K-values, lower boundaries and ground level (horizontal model)

The entire combinations must be used as a whole, because one dataset is useless without the other one. If this fact is not taken into account, errors can arise. Computation and input is done element wise. Therefore, the possibilities 1. to 6. can be used simultaneously within the same model. For instance, you can use 4. only in an anisotropic part of the model; the remaining elements are defined with combination 1. Several input types for one element are not possible. The value zero cannot be assigned to the attributes permeability, transmissivity and thickness (use a very small value or remove the element).

Inflow / outflow rates are not absolutely necessary.

Warning: No input for inflow / outflow means inflow / outflow rate = 0.0.

3.3 Discretization in space

Definition:

The study area is discretized by dividing it into a finite number of simple subregions, the so called elements.

3.3.1 Theory of the discretization in space

Discretization in space means that a study area is divided into individual, geometrically easily detectable areas for which the mathematical shape functions are defined. These sections with the corresponding shape functions are the finite elements which lead to the linking of the study area. The shape functions are defined by so-called reference points, these are the element nodes. The shape functions, also called form functions are functions that approximate the course of the real function of the element as best as possible. Condition is to fulfil the continuity condition. Since the nodes are shared by at least two elements, the continuity condition is fulfilled when using the values in these points.

The sought function curve is determined approximately by interpolating the values in these nodes. To express the course of the function **u(x, y)** through the nodes, the shape functions are used. These have the property to be in the current node always 1 and the remaining nodes to be 0, so that **u(x, y)** results as the sum of the number of nodes $u_i * N_i(x, y)$, where **i** is the number of the node in the element and **u** represents the value at the node (abstract from Wikipedia).

The linear shape function for the unit triangle with the coordinates of three vertices (0,0), (1.0) (0.1) is as follows (**linear approach**):
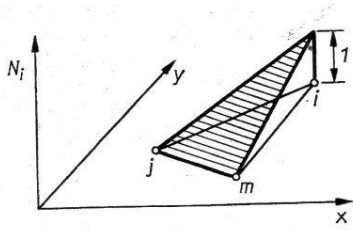
Fig. 49: Linear approach of a triangle

For quadrilateral elements a bilinear approach has to be used, since the edge of a quadrilateral element with a Z-coordinate not equal 0 is a curved line:
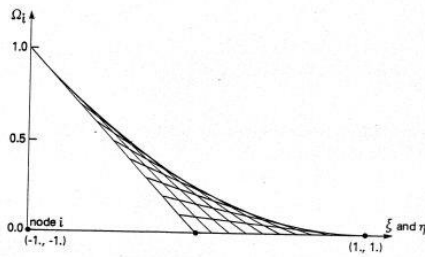
Fig. 50: Bilinear approach for a quadrangle

Even more clearly is the shape function in the figure below:
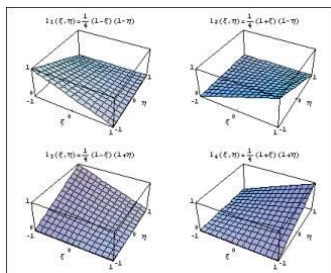
Fig. 51: Bilinear approach for a quadrilateral element

3.3.2 Implementation in SPRING

When creating a numerical groundwater model the previously defined model area is discretized. In SPRING, the finite element method is used due to its high flexibility with respect to the exact location mapping of irregular structures and the possibility of local grid refinements.

In SPRING irregular shaped triangles and rectangles are used as element forms in the two dimensional or penta- and hexahedrons are used in the three dimensional. Because of their variable forms the elements can approximate the geometry of the considered structures piecewise linear. The corners of the elements are the nodes of the model.

Especially the point, line and areal structures have to be represented by the mesh geometry.

For the discretization of three-dimensional models it is customary to project the horizontal grid to the underlying layers to create the elements of the individual layers. The position of the node layers usually corresponds to the situation of geological interfaces. For numerical reasons it may sometimes be necessary to refine further layers without a geological reason.

The discretization is to choose always in view of the problem that should be solved and the existing database. If in a model no measures are planned and no hydraulic structures act, the space between the nodes can be 25 to 50 meters. Depending on the model size or the questions to be answered a higher resolution may be useful.

The following table gives a recommendation for the size of discretization of a model in specific areas:

	Recommended distance of the nodes
Wells	Under 5 m
Planning area	5 to 15 m
Water with planned measures	5 to 10 m
Water without measures	10 to 15 m
Sewerage or drainages	10 to 25 m
Shipping channels	20 to 50 m
Model boundary area	30 to 60 m

3.3.3 Special case: discretization of wells

Groundwater withdrawal wells create a cone of depression, which is characterized by an increasing gradient with increasing proximity to the well. To represent this situation in a model adequately, a radially symmetric refinement is required around the well. As the gradient which has to be described increases with the approach to the well, an exponential refinement is advantageous.

The parameters of the refinement must be chosen depending on the hydraulic conditions. Thus areas with a low permeability and thus a strong lowering of the groundwater level require a small inner radius (e.g. 2 m) but only a small outer radius (e.g. 50 m) as well.

In flat cones of depressions these parameters can be set higher. Based on the distance of the inner radius of the well, the nodes are arranged radially symmetrical with exponentially increasing distance to the well, until the desired outer radius is reached or exceeded.

The well parameters are determined within working on the contours during the mesh generation. Via *Contour → Edit p-contours → well parameters* and after selecting the appropriate node, an input window appears where the parameters are entered. The following figure shows the relationship between the parameters and the element mesh refinement:
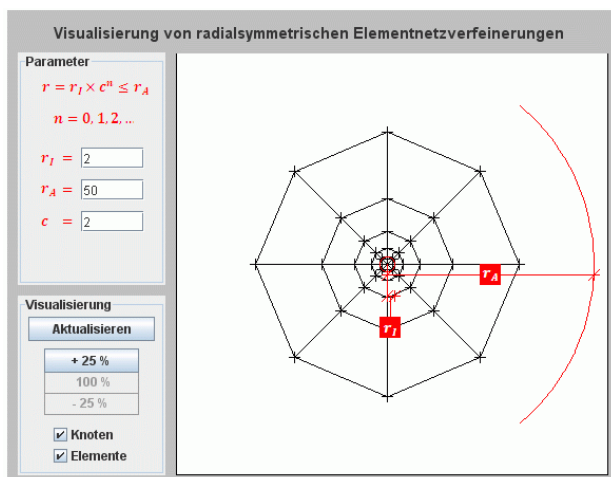
Fig. 52: Radial symmetrical arrangement of nodes during the well discretization

Representing horizontal filter wells, the individual lines have to be considered in the discretization. Each line should be represented by multiple nodes with a distance of 5 m to 10 m. An exponential mesh refinement as in vertical wells is not necessary because groundwater drawdowns in horizontal filter wells are more areally developed. The distribution of the total amount of withdrawal should not be equally on all nodes, but in a way that a highly uniform potential distribution results at each line.

3.3.3.1 Example of use

The principle model of a well looks as follows:
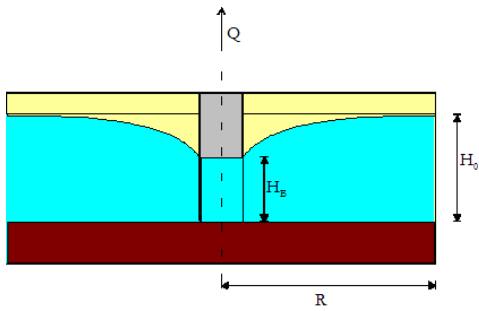
Fig. 53: Principle model of a well (R = influence of a well)

The following figure shows the detail of a model, if no well parameters are set:
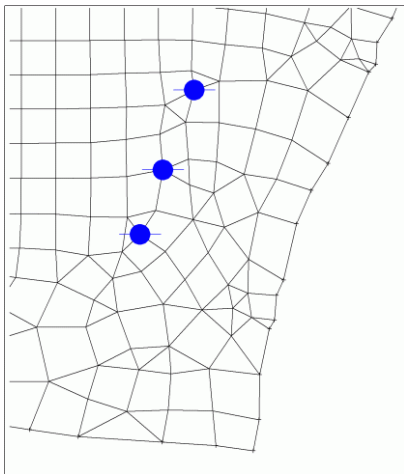
Fig. 54: FE-mesh without setting well parameters

For the present example, the parameters R = 30.0 m, Ra = 150.0 m and C = 2.0 were set. After confirming the dialog the resulting contour points around the well are shown in the user interface. So the user can check the parameters and change if necessary. The parameters are stored in the structure file and are considered during the node generation. A node and element generation including an areally refinement around the well gives the following figure:
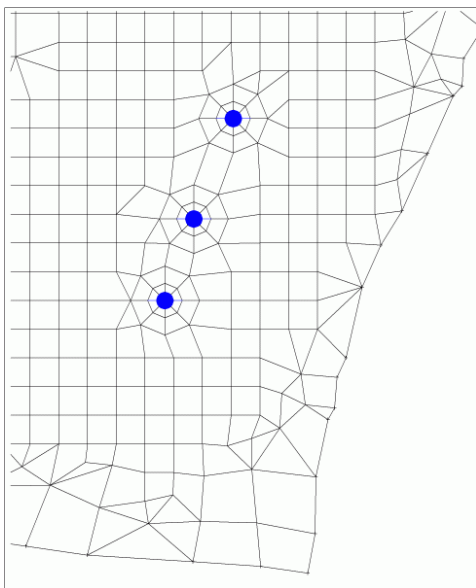
Fig. 55: FE-mesh with setting of well parameters

Comparing the two figures you can see clearly the radially symmetric distribution of the nodes around the well, similar to a "spider web".

The following vertical sections show the course of the free surface (green line). Caused by the finer discretization using well parameters it behaves similar to the cone of depression of a well (lower figure).
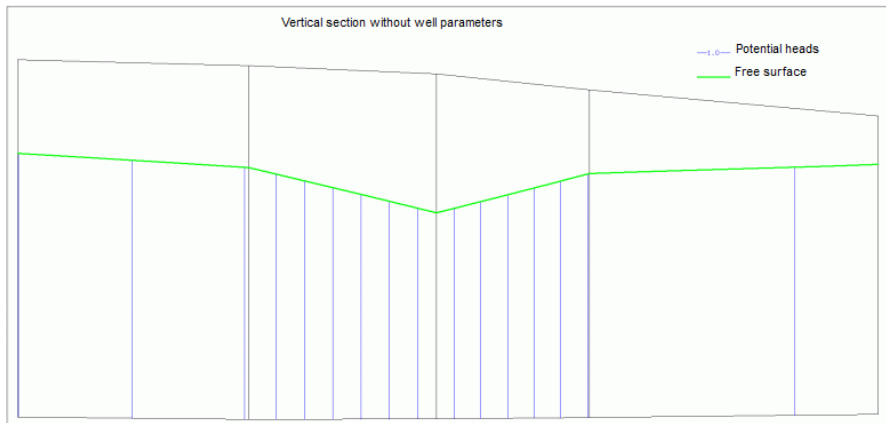
Fig. 56: Vertical section without setting well parameters
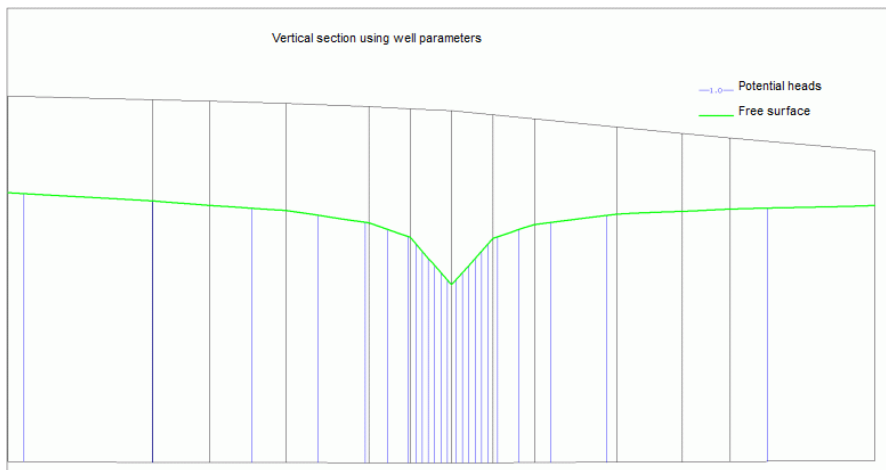
Fig. 57: Vertical section with setting well parameters

3.3.4 Stability criterion of the discretization in space

The geometry of the mesh has an undesirable effect on the computed material spreading, if the elements are too large and their length is perpendicular to the flow direction. The following figure shows how in areas with unfavourable discretization the material spreads much more than in areas with finer discretization. This effect is known as numerical dispersion.
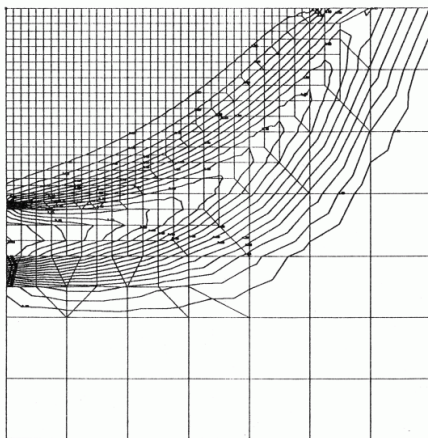
Fig. 58: Influence of the discretization on the results of the mass transport calculation

(from: König, C.: Numerical computation of the three-dimensional mass flow in the groundwater. – Technisch-wissenschaftliche Mitteilungen des Instituts für konstruktiven Ingenieurbau, Ruhr-Universität Bochum, 91-93, (Bochum))

Taking into account the following criteria this effect can be so small that it does not exceed the natural dispersion:

$$\Delta t < \frac{L}{V} \quad \text{and} \quad L < 2\alpha_L$$

L is the longest dimension of an element in the flow direction, α_L is the dispersion coefficient in longitudinal flow direction.

Oscillations of the numerical solution are related to the fact that the transport equation is a differential equation with partially hyperbolic character. This is difficult because the algorithms used are designed primarily for the solution of differential equations of parabolic type. Now if the part of the advective transport is large compared to the dispersive part, the hyperbolical character of the equation outweighs.

A measure of this ratio is the Peclet number P_e, which should be less than 2 so that the proportion of the non-hyperbolic part predominates.
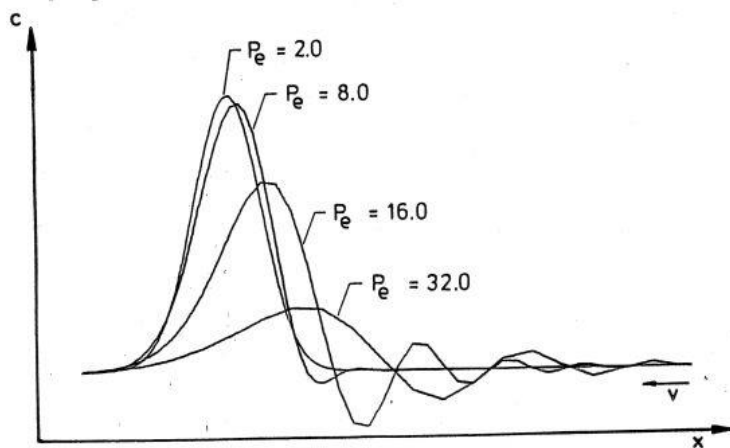
Fig. 59: Dependence between Peclet number and oscillations (König, 1991)

It describes the ratio of the advective part to the dispersion part (D) with respect to a characteristic length (side length of the elements, Δl).

The lower the Peclet number, the less iterations are necessary to achieve a predefined maximum value of the residuals. Once this dimensionless number exceeds the value 10, is no longer guaranteed that the solution converges. An optimal discretization in space results for a Peclet number < 2.

3.4 Time discretization

For transient groundwater modelling a temporal discretization (the increment between time points to be calculated) has to be chosen. The size of a suitable time step size depends to a great extent on the time course of the event to be described. Thus, the modelling of a flood wave requires time steps in the order of hours to a day, while for the calculation of annual series of the groundwater level fluctuation time steps of a few days to a month are allowed.

The mapping of short-term changes of withdrawal in a well may require time steps of minutes, while in a subsequent phase with unchanged production rates, the progressive enlargement of time steps is allowed.

To avoid any impact on the model results due to large time steps a sensitivity analysis can be carried out at the beginning of a project, where relatively large time steps are reduced to small time steps until no difference is seen between the results of the last two model computations. A further reduction of the time step would increase the computational effort without producing an improvement in the model results.

For the time discretization of the unsteady flow equation an implicit Euler method is used in SPRING:

$$\frac{\partial p}{\partial t} = L(p) \quad \rightarrow \quad \frac{p(t + \Delta t) - p(t)}{\Delta t} = \theta L\big(p(t + \Delta t)\big) + (1 - \theta)L\big(p(t)\big)$$

with $\theta = 1$.

The transient boundary conditions are by default treated fully implicit ($\theta = 1.0$), i.e., to calculate the state at the time t_{n+1}, the boundary conditions of the time t_{n+1} are used.

In flow rate boundary conditions is only an explicit treatment of the unsteady boundary conditions possible, i.e., to calculate the state at the time t_{n+1} are the boundary conditions of the time t_n are used.

3.4.1 Special case: transport calculation

For the transport equation (for transport of a solute in the fluid the mass transport equation and the energy transport equation for temperature-dependent problems), the discretization of the asymmetrical operator leads to an unbalanced system of equations.

This cannot be solved with the iterative PCG solver. Therefore, in steady-state transport the system of equations for the transport part is solved always using an LU-reduction (direct solver). In large models with many nodes this can result in an extreme computing time.

For transient transport processes, the time discretization is performed using the by König ['Numerical calculation of three-dimensional solute transport in groundwater', TWM 91-13, RUB 1991] developed operator split method.

Using this method in each time step two systems of equations with symmetric positive definite matrices have to be solved for the transport part. Here, the iterative solver can be used. Should steady-state transport processes be calculated for large models, it is recommended to calculate the steady state by a transient calculation (to reach the steady state) to reduce the computation time.

3.4.2 Stability criterion of the time discretization

The Courant number is used in the numerical flow simulation for the discretization of time dependent partial differential equations:

$$C_O = \left| \frac{\Delta t v}{L} \right| \leq 1$$

Here Co is the Courant number, v the velocity, Δt the discrete time step and L is the longest dimension of an element in the flow direction.

Since the upper limit for L is already given by the Peclet number and usually is relatively small, so the time step size Δt has to be chosen sufficiently small depending on the velocity in order to fulfil the Courant criterion.

The following figure shows the influence of the Courant number on the calculated concentration distribution in otherwise identical model parameters:
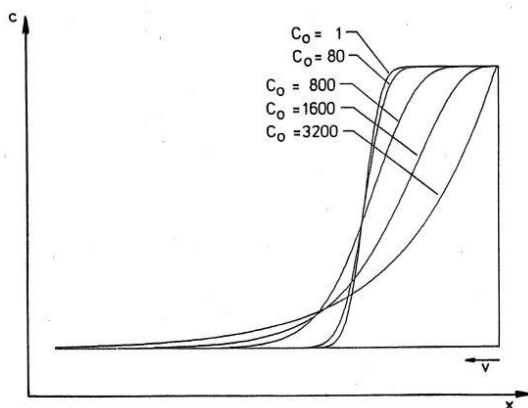
Fig. 60: Dependency of Courant criterion and numerical dispersion (König, 1991)

4 SPRING-Menus

All processes in SPRING can be started from the menu bar. Depending on the opened file (model file or plot file), the menu bars differ in some menu items. The different standard menu bars depending on the type of the opened file are shown below. In the following chapters the individual menus, icons and toolbars are described.

Menu bar for a model file (*.net):
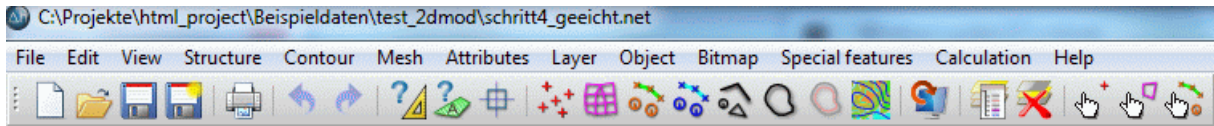
Menu bar for a plot file (*.plx = SPRING internal graphics format):
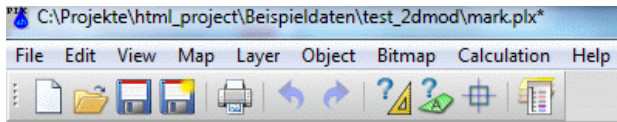
4.1 File

The following menu items are selectable:
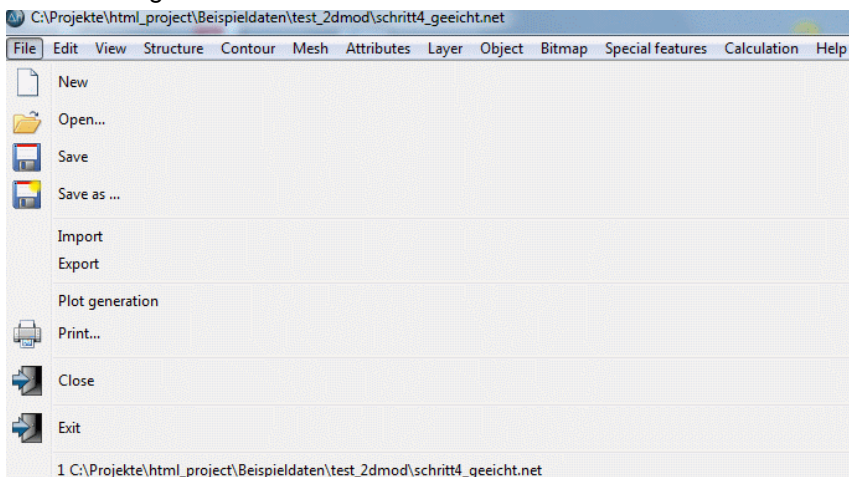
New

After activating this menu item a window opens in which the properties of the new project can be established. Details are described in Compilation of a 2D model.

Open…

After activating this menu item, a file selection window appears where the file can be selected. The following file formats are possible:
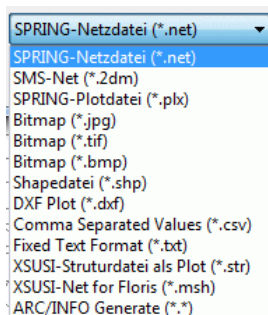
Fig. 61: Possible data formats in SPRING

If a project is already open, it is possible to open or to overlay a file in one of these formats directly using drag and drop.

Save and save as… ,

Using the menu item *Save* the project or the plot is stored under the name previously assigned. Using *Save As ...*, the project (or plot) is stored under a different name, whereas the previous project (or plot) remains under the old name.

Import

Model file:

Here can be selected if a submesh should be imported, a file should be overlaid or the transient boundary conditions should be imported:
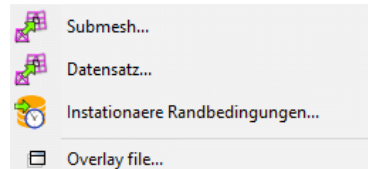
The formats are the same as when opening a file. The import of a submesh is described in detail in chapter "Creating a 3D model - Import of a submesh in an existing model" (p. 225).

The import or overlaying of *data sets* are possible. There are the same formats to select as when opening a file (*.str. *.plx, *.shp, etc).

The import of *Transient boundary conditions*... is described in detail in chapter *Special Features - Transient...*(S. 174).

Plot file:

The possibilities are to add a file on the right, to attach on the left, to add at the top, to attach below or to overlay at the existing file.
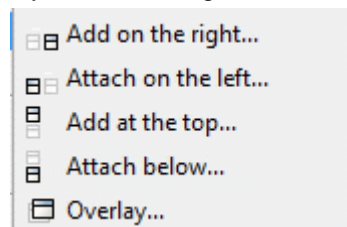
The formats are the same as when opening a file.

Export

The available export options are described in the chapter data export.

Plot generation

The available possibilities to generate a plot are described in chapter plot generation.

Print…

If you select *Print* the following window appears in which the various printing options can be set.
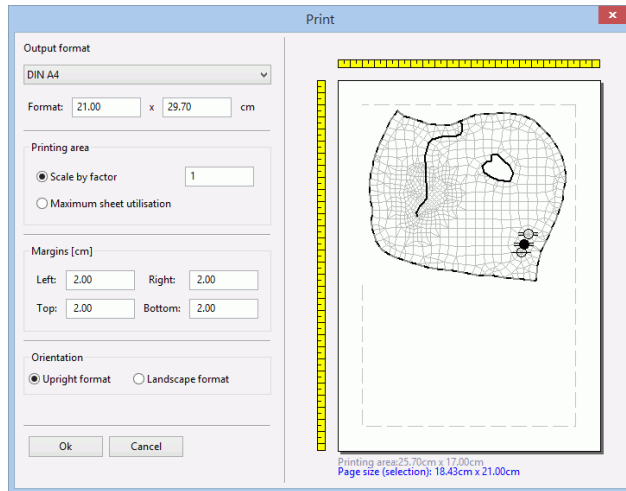
The dimensions of the "*Printing area*" are shown below the print preview. They result from the subtraction: Page dimensions - margins. The "*Page size*" is derived from the sum of: selection area + margins. This information is required for printing out from the pdf document. If the page size is specified there, the pdf document will get exactly the size you want, including the margins.

After confirming with OK, a window for selecting a printer appears.

Close, Exit

When selecting *Close* only the project will be closed, SPRING remains open. When selecting *Exit* the project and SPRING will be closed.

4.2 Edit

The following menu items are selectable:
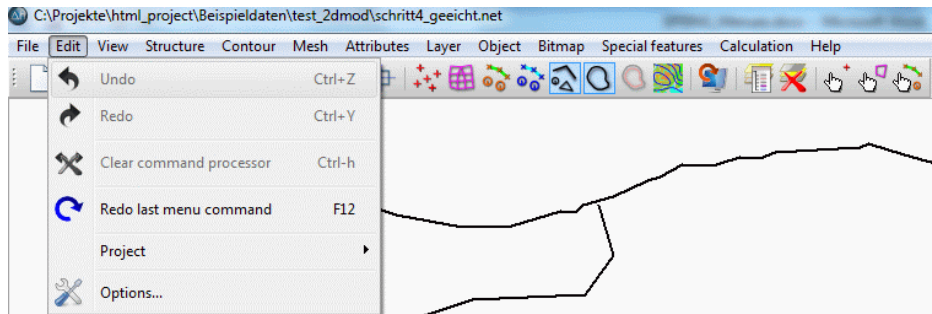
Undo/Redo

By choosing *Undo*, or the appropriate shortcut icon, the last action (e.g. *move nodes*) is reversed. If an object was deleted it will be restored with *Undo* (shortcut or symbol). The function *Redo* repeats the last action (e.g. *insert nodes*).

Clear command processor

This function deletes the last stored functions.

Redo last menu command

(only with opened model file)

By selecting this menu item or pressing the F12 key, the last menu function is repeated.

Project

Since this menu item is very complex, it is treated in a separate chapter.

Options

Since this menu item is very complex, it is treated in a separate chapter.

4.2.1 Project

(This menu item is only available with an opened model file.)

The following menu items appear:
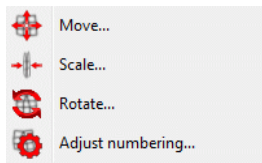
Move

With this menu item it is possible to move the origin of a model, for example to change from relative (x, y) coordinates on Gauss-Krueger coordinates. The offset in x- and/or y-direction has to be entered.
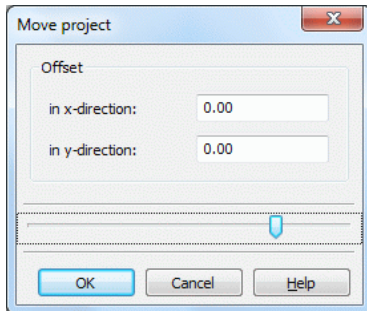
Scale

With this menu item a model can be re-scaled, i.e. the factor enlarge or reduce the distances in X and/or Y-direction. The scale factor in x- and/or y-direction has to be entered.
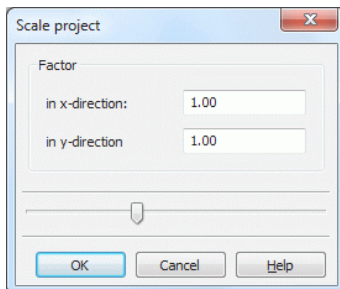
Rotate

With this menu item the coordinate system of a model is rotated. It is expected to enter a rotation angle (in degrees counter-clockwise).
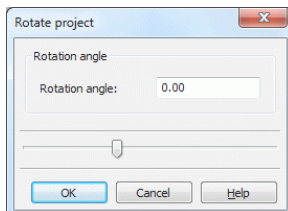
Adjust numbering

The following input is possible:

Adjust numbering	

Current node numbers:
  min: 1
  max: 6888
  Number: 6888

Current element numbers:
  min: 1
  max: 7359
  Number: 7359

Model modifications

Scale in x-direction: 5000

Scale in y-direction: 5000

3D-no.-offset: 20000

min. node number.:   1

min. element no.:   1

☐ Consecutive node numbering
☐ Consecutive element numbering

Transparency of the dialogue

[OK]   [Cancel]   [Help]
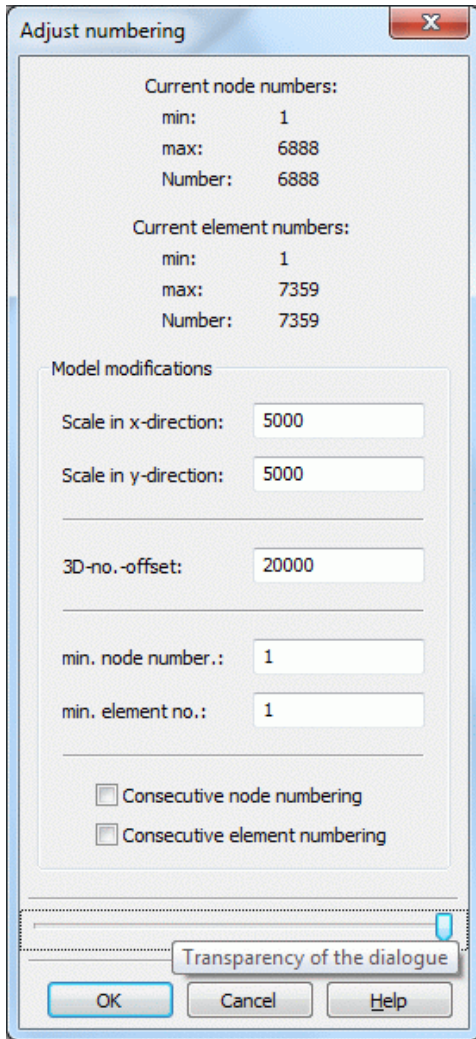
Scale in x-/y-direction

With this menu item the scale of the model file in x- and/or y-direction is changed.

3D-no.-offset (3D model only)

With this menu item the 3D-no.offset (3DNR) of a 3D model is enlarged or reduced. If an enlargement is necessary in the modeling / refinement, the 3D offset is automatically adjusted.

Min. node-/element number

All node numbers and/or element numbers can be modified by changing the minimum node number or the minimum element number by a certain offset higher numbered or numbered down. The node and element numbers have to (by numbering down) remain positive. The no longer appropriate offset for the 3D-numbering changes automatically.

This feature is used primarily for the coupling of two models.

Consecutive node/element numbering

From the mesh generation or editing the mesh develop gaps in the node and element numbering. This can lead to a too small 3D offset, although there are not so many nodes or elements. In these cases it may be useful to number the nodes and/or element consecutively.

4.2.2 Options

In this menu are the SPRING options from the graphic interface (formerly XPLT and XSUSI options, *spring.opt*) and the plot options from the plot generation (formerly PLOGEO options, *plogeo.ini*) summarized. The settings are saved in the files *spring.opt* and *plogeo.ini* in the directory:

"C:\Users\Public\Documents\SPRING\Konfig" (Windows 7 or Windows Vista). This path varies depending on the operating system or the hard drive partitioning (e.g. under Windows XP: "C:\Documents and Settings\All Users\Documents\SPRING\Konfig").

The format of the files is described in chapter: Data structure – Initialization files.

The options are split in the sections *In general, Mesh, Plot, Objects* and *colours*. In the following chapters, the possible configuration options are described.

4.2.2.1 In general

In the general options the following input window appears:
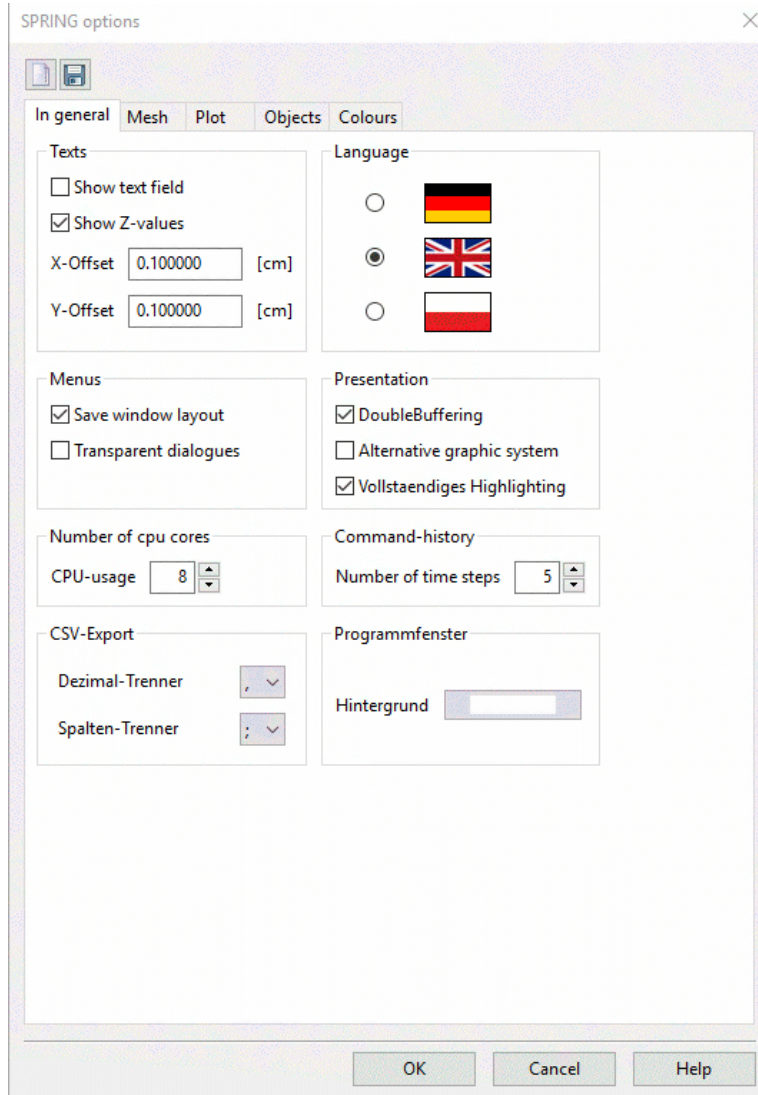
The buttons in the header of the input window allow you to save the input parameters (🖫) in the initialization files *spring.opt* or *plogeo.ini* or to reset the input parameters to default (🗋). Reset means all settings are reset, i.e. the toolbars are reset to the default toolbar as well.

Texts

SPRING options for overlaying *.txt*- and *.str*-files

■ Show text field (Variable SPRINGTextBox)

If this check box is activated (default), a rectangle is displayed for each text on the screen that marks the location and extent of the text in the plot. Deactivating the check box eliminates the appearance of the text box. The text box is not shown in the plot. Due to the fact that the exact metric of the plotter fonts is not known, the expansion of the text box is only estimated.

■   Show Z-values (Variable SPRINGShowZValues)

If this check box is activated, texts or values (3rd column of the files) from *.txt and *.str files are shown in the map. The text position relative to the coordinates is determined by the X-and Y-offset. Deactivating the check box suppresses the representation of texts.

■   X- and Y-Offset (Variables SPRINGTextXoffset, SPRINGTextYoffset)

In addition to overlaying graphics files SPRING is able to import structure files (*.str) or ASCII-formatted files *(*.txt)*, e.g. to represent values or texts at the defined coordinates. The variables SPRINGTextXoffset and SPRINGTextYOffset define the position of the labels relative to the coordinates specified in the file. The default values of 0.1 cm start right and above the coordinates. For positions on the left or below the coordinates negative values must be specified.

The values for the X and Y offset have to be defined **before** importing the data set.

Language

(Variable SPRINGLanguage)

With this radio button you can switch the language of the user interface to German, English or Polish. The language changes immediately.

Menus

■   Save window layout

Here you can choose whether the current layout (e.g. arrangement or opening of other windows) should be saved for the next SPRING session.

■   Transparent dialogue

If this check box is activated, a slider is attached to some dialogues (e.g. Assign attributes), with which this can be switched to semi-transparent. So these dialogues do not cover selected objects any more.

After activating this checkbox appears the following bar in the dialogue- and input windows:

By moving the pointer to the left the transparency of the window enhances while moving the pointer to the right the transparency of the window decreases.

Presentation

■   Double Buffering

By buffering in the framebuffer of the graphics card memory, the display is speeded up and it avoids the flickering of the screen.

■   Alternative graphic system

You can choose between a 2nd Graphics system with support for antialiasing and transparency effects. This has to be activated in advance if the transparency of a layer (p. 168) should be changed.

■   Full Highlighting

When activated, all selected objects are highlighted (for example, all structure points), when deactivated, only the bounding box of the data is displayed.

Number of cpu cores

(Variable SPRINGMaxCPUusage)

SPRING is able to parallelize the performance of the on a computer existing CPU processors. The user can determine at this point how many logical CPU cores he wants to use for SPRING. He chooses e.g. 3 of 4 processors, so that he can do something else. Does he select 4 of 4 available processors, the computer operates at full power for SPRING.

Command history

(Variable SPRINGMaxCommandHistory)

The user can choose how many commands are to be repeated at maximum when using the button *Undo/redo*.

CSV export

The user can choose which decimal (dot or comma) or column separator (comma or semicolon) is used in the output files (e.g. hydrograph-files, DiffEich.txt, fracht.csv and so on). Default is the dot as decimal separator and the comma as a column separator.

Window background

Here you can choose the background color of the SPRING interface.

4.2.2.2 Mesh

It appears the following input window for the mesh options. Here, on different notebook pages (mesh, data, structures and contours) representation attributes are set:
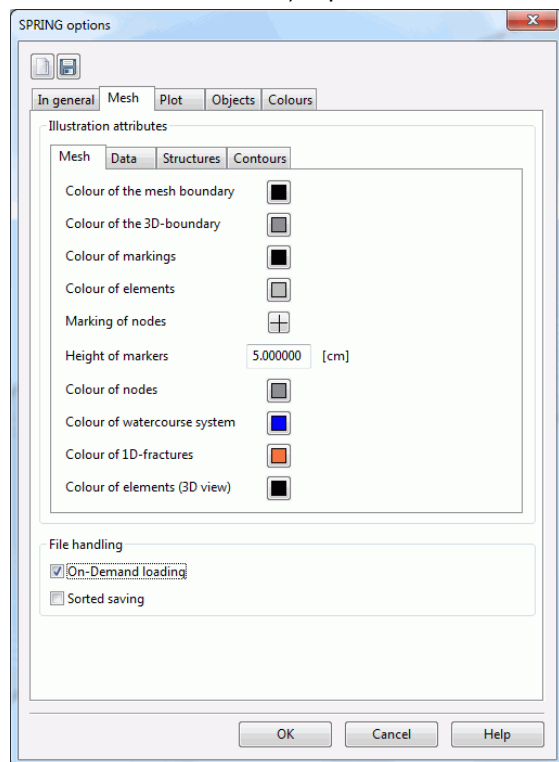
In the note book side „mesh" colours are set which are used in the presentation.

- Colour of the mesh boundary (variable XSUSIRandPen)
- Colour of the 3D boundary (variable XSUSI3DRandPen)
- Colour of the markings (variable XSUSIMarkPen)
- Colour of elements (variable XSUSIElemPen)
- Marking of nodes (variable XSUSIKoorMark)
- Height of marker (variable XSUSIKoorHeight)
- Colour of nodes (variable XSUSIKoorPen)
- Colour of water course systems (variable XSUSIStreamNetworkPen)
- Colour of 1D fractures (variable XSUSI1DKluftPen)
- Colour of elements (3D-view) (no variable is set)

A colour changed in the options menu is used only in a renewed presentation of the boundaries or the markings.

A colour of the element or node representation changed in the options menu is used only when opening a new project. If the existing representation colour of the elements or nodes should be changed, this is only possible via the menu *Layer → Modify colours.*

Changing the existing markers at nodes or the height of the markers is only possible via the toolbar of the object attributes (*View → Toolbars → Object attributes*).

On-Demand loading

(Variable XSUSILoadOnDemand)

The opening of the project can be accelerated by activating the check box "On-Demand loading" (default, variable gets value = 1). Thus, the node and element attributes are not initially read, but only when an attribute is to be edited (*Attributes → Edit nodes/elements...*).

If the attributes should be loaded immediately, the variable must be set to 0 (uncheck the box) and the options have to be saved. The next time the project is opened, the attributes are then loaded directly.

Sorted Saving

(Variable XSUSISaveSpeed)

To accelerate the saving of the "*.net" and "*.3d" file, the sorting of the attributes is switched off. For this, the variable has the value 1 (default). If the attributes should be saved in their sequence of numbers the variable must be set to 0 (activating the check box).

4.2.2.2.1 Mesh – Data

For the representation of data the options shown in the following input window are available:
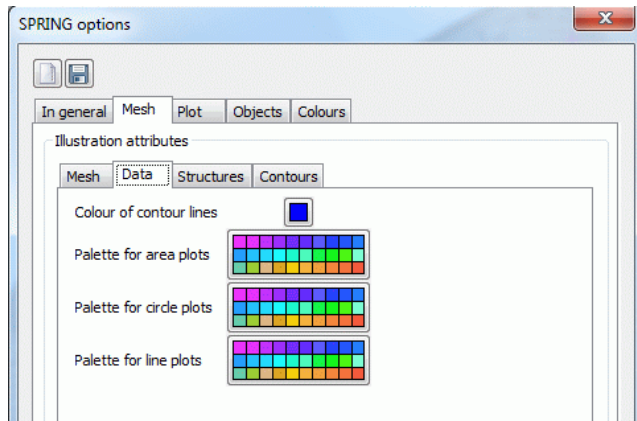
- Colour of contour lines (variable XSUSIIsolPen)
- Colour for area plots (variable XSUSIFlaePen)
- Colour for circle plots (variable XSUSIKreiPen)
- Colour for line plots (variable XSUSILinePen)

The chosen colour palettes are used for representing contour lines or area -, circle or line plots in the menu *View → Show attributes.*

4.2.2.2.2   Mesh – Structures

For the representation of structures the options shown in the following input window are available:
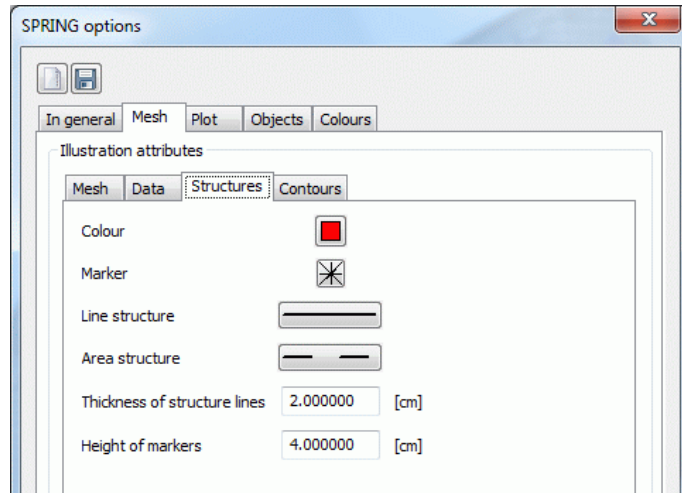
- Default colour for presenting structures (variable XSUSIStruPen)
- Setting the default marker with the variable XSUSIStruMtyp and setting the height of the default marker with the variable XSUSIStruHeight
- Default line type of line structures (variable XSUSILtyp)
- Default line type of area structures (variable XSUSIStruFtyp)
- Default line width of structures variable (XSUSIStruPenWidth)

If one of these options is changed in the options menu, this has no effect on the appearance of existing structures. Their presentation attributes can be changed via the menu *Structure → Edit*.

4.2.2.2.3   Mesh – Contours

For the representation of contours the options shown in the following input window are available:
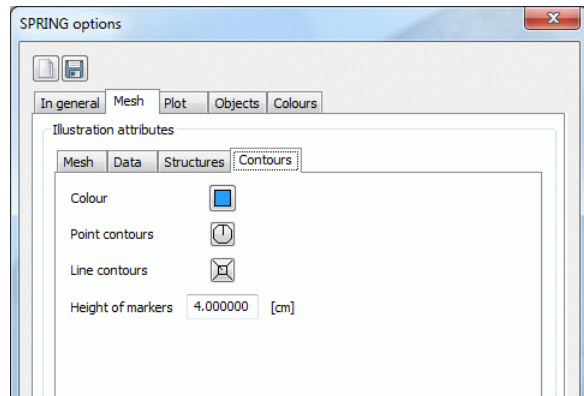
- Default colour for presenting contours (variable XSUSIKontPen)
- Setting the default marker for point contours: variable XSUSIKontPmark and setting the height of the marker with: XSUSIKontHeight
- Setting the default marker for line contours variable XSUSIKontLmark and setting the height of the marker with: variable XSUSIKontHeight

If one of these options is changed in the options menu, this has no effect on the appearance of existing contours. Their presentation attributes can only be changed via the menu *Layer → Modify colours*.

A modified parameter of the contour presentation is used only when opening a new project. Changing the types of marker and the height of the markers of an existing contour layer is not possible.

4.2.2.3   Plot

The important initialization parameters for the plot generation are defined in this menu. They are stored in the initialization files plogeo.ini and spring.opt. Some parameters are defined during the plot generation in the extended settings.

The following input window appears:
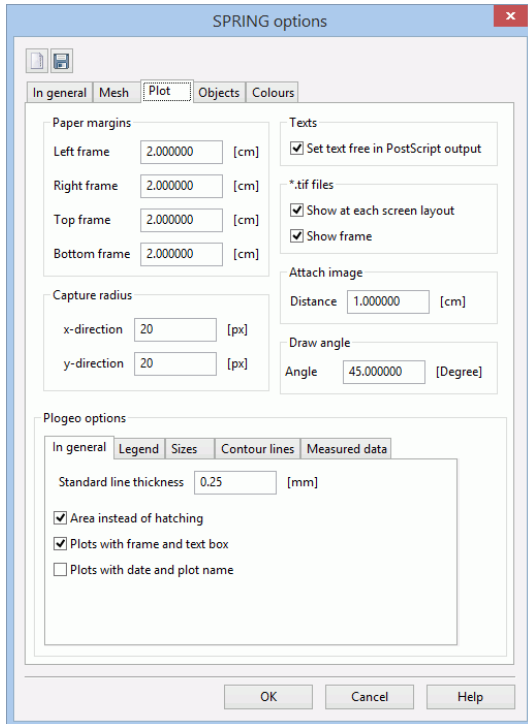
Defining the paper margins for printing

(Variables XPLTLinkerPapierRand, XPLTRechterPapierRand, XPLTObererPapierRand, XPLTUntererPapier-Rand)

Here, the sizes of the paper margins can be modified, the default settings are 2.0 cm.

Capture radius for selecting objects

(Variables SPRINGFangradiusX, SPRINGFangradiusY)

The capture radius defines up to which maximum distance the cursor selects an object.

Crop texts

(Variable XPLTTextSetFree)

If this check box is activated (value "1" in the spring.opt), the text in a PostScript-output is released from the background. The effect of this function is not visible on the screen.

TIFF files

(Variable XPLTShowTiff)

If this check box is activated (value "1" in the spring.opt), the TIFF files are displayed by each presentation. By deactivating ("0" in the spring.opt) the check box, the representation of the TIFF files is suppressed.

Attach image

(Variable XPLTMapOffset)

With this option it is possible to determine the distance from a new map to an already existing map. These two maps are laid down next to each other. The set values are considered in the functions *File → Import… → Attach…* in the plot mode. The default setting is 1.0 cm.

Definition of the angle of a line

(Variable XPLTLineAngle)

With this option the angle (between +90 and -90 degree) is set by which a line is drawn when calling the menu *Object→ Draw → Line*. The angle is, however, only active when the CTRL-key is pushed. In this case the line clicks into place if it reaches the defined angle or if it is horizontal or vertical aligned.

4.2.2.3.1 Plot – In general

After selecting the tab *In General* appears the following input block:

Standard line thickness

The plogeo.ini command PENW set the default line-width (in mm) for all objects where no line-width is explicitly defined.

Area instead of hatching

By activating this check box, the plogeo.ini command FLAE (not to confuse with the type of data FLAE or the PLOGEO batch file command FLAE!) is set to the value "1", that means, contour line area plots are coloured areawise and hatch plots are not hatched but coloured as areas.

Deactivating the check box means that the value is set to "0" so that contour line area plots are carried out with close line hatches and hatch plots are really carried out as hatching.

The output of an area or a hatch plot, which was created with the plogeo.ini command FLAE = 1, requires a printer that can also handle colour areas (not a pen plotter!).

Angles, distance and line width data from the hatch parameters are irrelevant for FLAE = 1!

In case of the contour line area plot only the entered partial areas are coloured. That means, the entered extreme interval boundaries are not completed by the global extrema (this is only the case when the plogeo.ini command FLAE is set to 0).

Plots with frame and text box

By activating this check box the plogeo.ini command NORA is set to the value "0", that means, each plot is provided with a plot frame. Is the check box deactivated the plot will be displayed without plot frame.

Plots with date and plt-name

By activating this check box the plogeo.ini command TIME is set to the value "1", that means, every plot is labelled at the left bottom with the date of generation, time and plot name (absolute path name). When the check box is deactivated the plots are not labelled (TIME = 0 in the plogeo.ini file).

4.2.2.3.2 Plot – Legend

After selecting the tab *Legend* appears the following input block:

Height of labelling

Definition of the height of the labelling in the legend in cm (plogeo.ini command HLEG)

Distance between legend and coordinate frame

Here the distance of the legend to the coordinate frame can be set in cm (plogeo.ini command DLEG).

Number of columns for domain legends

Here, the number of columns in the legend of an areally representation is defined (plogeo.ini command NLEG). You can choose between the possibility to show the legend in one or two columns.

Labelling as editable texts

By activating this check box the plogeo.ini command TEXT is set to the value "1", that means, the text is displayed in text format. Otherwise, the legends and frame labels are displayed as in polylines converted characters (TEXT = 0). A subsequent change of the text is then no longer possible.

4.2.2.3.3   Plot – Sizes

After selecting the tab *Sizes* appears the following input block:

Values/Numbers, Height of labelling

The height of the label [in cm] for values or numbers of nodes or elements is set (plogeo.ini command HWER).

PLX-marker, Height

The height of the PLX-marker [in cm] used as markings (data type MARK) is set (plogeo.ini command HMAR).

Circle diameter

The maximum radius [in cm] is determined which is used in a circle plot (plogeo.ini command RADI).

4.2.2.3.4   Plot – Contour lines

After selecting the tab *Contour lines* appears the following input block:

Height of labelling

The height [in cm] of the contour line labels is set (plogeo.ini command HISO).

Number of decimal places

The number of decimal points in the labelling of contour lines (for floating-point format) is established (plogeo.ini command NAKO). A presentation without decimal places is also possible.

Labelling as editable texts

By activating this check box the plogeo.ini command TXT2 is set to the value "1", that means, the text is displayed in text format (Font Helvetica, normal), when the plogeo.ini command TEXT is set to "1". Otherwise, the legends and frame labels are displayed as in polylines converted characters (TXT2 = 0).

Labelling in direction of

By activating this check box the plogeo.ini command ISBE is set to the value "1", that means,

the contour line labels are "uphill" (i.e. the label of the contour line is so oriented that the smaller values of the data type are below and the larger ones are above).If the check box is disabled, the contour line labels are "readable", that means, the labels are turned at maximum up to 90 degrees to the x-axis of the plot (ISBE = 0).

4.2.2.3.5 Plot – Measured data

After selecting the tab *Measured data* appears the following input block:

Marker height for measured data

The marker height [in cm] for the presentation of measurement data is set.

Type of marker for measured data

The marker type (plogeo.ini-command ETYP) for the presentation of measurement data is set. After clicking on the marker the following table appears with all possible marker types:
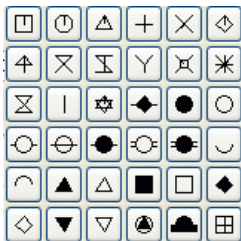
The corresponding marker number is then assigned to the plogeo.ini command ETYP (see p.). Default is the marker type 3 (cross). For the first marker of the table is ETYP = 0, then the markers are numbered line by line.

ASCII-output file with differences of measured data

By activating this check box the plogeo.ini command EDIF is set to the value "1", that means, an ASCII file is created. Depending on the plot type there are two possibilities:

■ Computing the differences to measured data

The computed differences between calculation results and measured data (EICH) are written to the ASCII file *DiffEich.txt/csv* using the format for characteristic data (6X, F10.2, F10.2, F10.2). The data are X-coordinate, Y-coordinate, difference, measured value, calculated value.

■ Plotting a hydrograph with nominal values

In this case, an ASCII file called C<node/element number>.csv is created for each node or element. Available are the following data:

Calculated value, measured value, time, node/element number (format F10.2, 1X, F10.2, 1X, F10.2, 1X, I10)

Both ASCII files can be used for creating so-called scatter plots (comparison of calculated and measured values in an x/y-diagram).

If other than the described plots are created, or the check box is disabled (EDIF = 0), no ASCII file is created.

Hydrographs

The line width for the presentation of hydrograph curves is set (plogeo.ini command GSTR). By activating the checkbox symbols at supporting points are set (plogeo.ini command GMARK).

4.2.2.4 Objects

The following window appears:
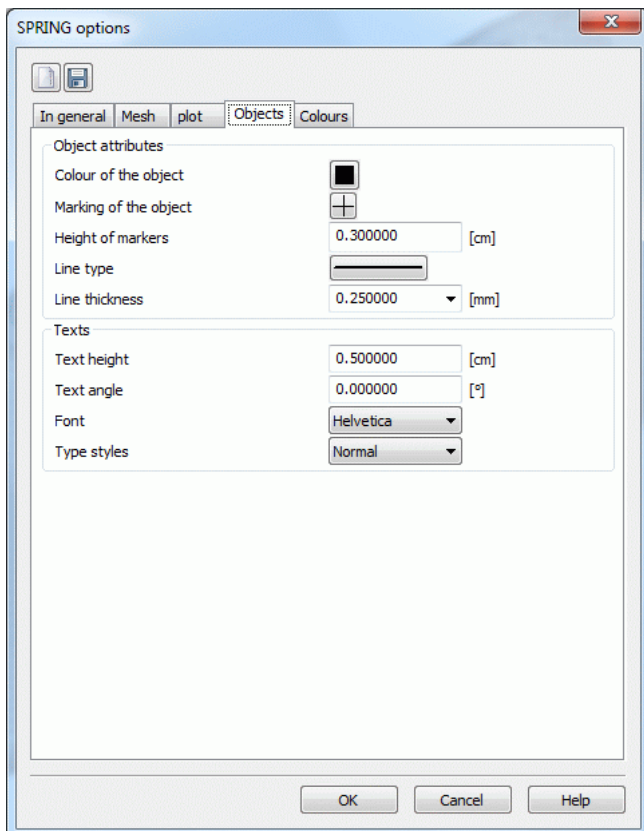
The object attributes which are selected here are related to the objects that can be created using the *View → Toolbars → Object attributes* (markers, lines, geometric figures, text).

Object attributes

■ Default colour for an object (Variable SPRINGcolDefault)

■ Default marking of an object (Variable SPRINGmtypDefault)

■ Default marker height of an object (Variable SPRINGmhgtDefault)

■ Default line type of an object (Variable SPRINGltypDefault)

■ Default line width of an object (Variable SPRINGpenwDefault)

Text attributes

- Default text height (Variable SPRINGtxthDefault)
- Default text angle (Variable: SPRINGanglDefault)
- Default text font and type style: (Variable SPRINGfontDefault), selectable are the fonts: Helvetica, Roman, Courier, Symbol and the type styles: normal, bold, italic, bold italic

The items *text font* and *type style* are defined together in the variable SPRINGfontDefault. It has two flags combined in an integer greater or equal than 10.

Meaning of the first digit (text font):

1 = Helvetica

2 = Roman

3 = Courier

4 = Symbol

Meaning of the second digit (type style):

0 = normal

1 = bold

2 = italic

3 = bold and italic

That means, if the variable SPRINGfontDefault has the value "32", is the text displayed in Courier and italic.

The font Symbol is only possible "normal" (= 40).

4.2.2.5 Colours

The following window appears for selecting the colours:
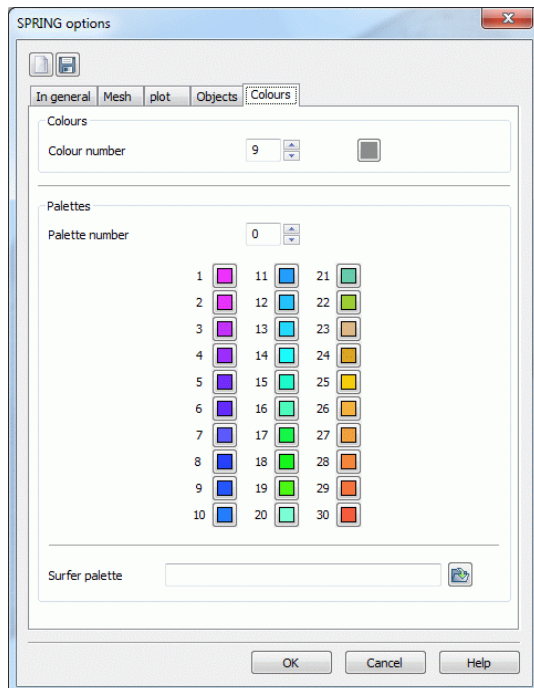
Colours, colour number

By selecting a colour number and activating the colour button a different colour is assigned to this number. By activating the colour button the following selection window appears:
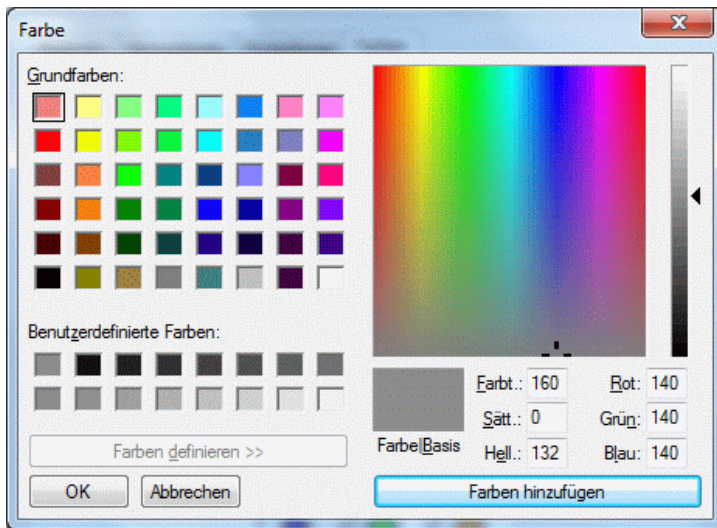
The user can create its own colours.

Palettes, palette number

(Variable SPRINGColor)

There is also the possibility, to set the number of the colour palette and the number of the corresponding colours manually by defining the variable XSUSIPalette in the initialization file spring.opt. Example:

```
XSUSIPalette: 1 10
 11 12 13 14 15 16 17 18 19 20
```

When SPRING is installed the file spring.opt is provided, which includes the colour palettes 0 to 9. By editing the file a total of 15 colour palettes can be defined.

After selecting a palette the individual colours in the palette corresponding to the possible number can be set. Or the palette compilation is done via the file selection window. Here, palettes in the Golden Software Surfer format (*.clr) are imported. These consist of any number of colour intervals and are interpolated on the SPRING-intervals of the respective palette that means colour gradients of 10 or 30 colours are created.

4.3 View

The menu item "View" includes different sub items, depending on the opened file type (model - or plot file). The following menu items are selectable in the model mode:
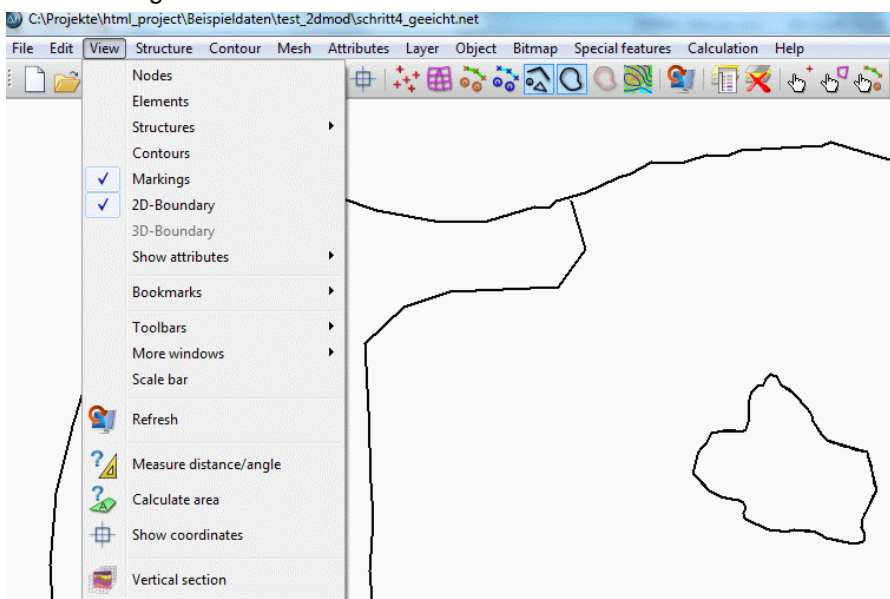
The following menu items are selectable in the plot mode:
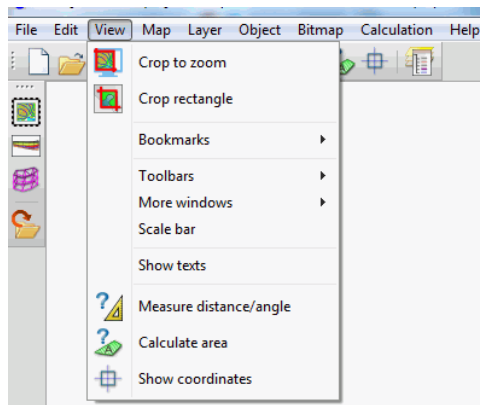
First, a description of the sub items of the mode model file follows:

Nodes/Elements

(model file only)

The nodes or elements are shown or hidden.

Structures

(model file only)

Since this menu item is very complex, it is treated in a separate chapter.

Contours, markings, 2D or 3D boundary

(model file only)

If contours, markings or boundaries (3D model or 2D with 3D model section) are present, they are shown or hidden after activating the corresponding menu item.

Show attributes

(model file only)

Since this menu item is very complex, it is treated in a separate chapter.

Bookmarks

(model - and plot file)

If you have zoomed in a project area that you would just call back later, you can create a bookmark.

The following menu appears when a bookmark is set (*View → Bookmark*), but there are not necessarily all entries immediately visible:
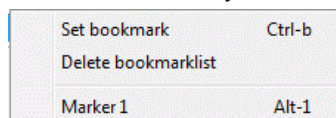
By selecting *Set Bookmark* or the shortcut Ctrl-b the selected zoom range is stored as a bookmark.

Subsequently, the display area can be changed.

If a zoom range has been defined, additional entries will appear in the menu. The bookmarks list is reset via the menu item *Delete bookmark list*.

When a zoom range was bookmarked, a menu item for each preset zoom range appears with an entry labeled "Marker" and a sequential numbering. To return back to a stored zoom range, either the corresponding menu entry "Marker no." or the shortcut Alt-no. can be used.

Toolbars

(model - and plot file)

Since this menu item is very complex, it is treated in a separate chapter.

More windows

(model - and plot file)

On request, other windows are opened which contain information about the project.

Project information...
Project manager
3D view...
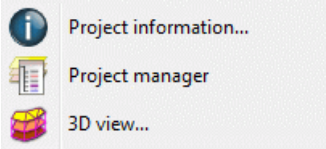
They are described in detail in chapter "Introduction to SPRING – SPRING user interface".

In the plot mode is only the menu item *Project manager* available.

Scale bar

(model - and plot file)

After selecting this menu item, a scale bar appears at the left bottom of the main window. The scale bar adapts on the changing proportions in a zoom.

0   345   690m
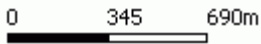
Refresh

(model file only)

After selecting this menu item, the view is updated in the main window (repaint).

Measure distance/angle

(model - and plot file)

The function *Measure distance/angle* calculates the distance between two points in the graphic. The distance is picked analogously to the function *Object → Edit → Line*. The computed distance and the angle are printed to status line in [m] and [degree]. The virtual line is created only temporarily and does not appear in the database.

The function *Measure distance/angle* is enabled until the right mouse button is pressed or another function is activated.

Calculate area

(model - and plot file)

The function *Calculate area* calculates the area of a specified object. It can be used for (poly-)lines and polygons. Open polylines are automatically closed by connecting start- and endpoint with each other. Polygons must have a minimum of three corners.

The computed area is printed to the status line in m² or in km². The program assumes one unit in the map corresponds to one meter, e. g. Gauss-Krueger-Coordinates. Consequently, you have to transform the area in the case that your map uses another unit. In any case, the area can only be computed correctly if the specified scale corresponds to the graphic scale. Pay attention especially to files which contain no scale information at all, because their scale is automatically set to 1:10,000. The scale can be corrected afterwards using the function *Map → Edit → Scale → Scale* .

The function *Calculate area* is enabled until the right mouse button is pressed or another function is activated.

Show coordinates

(model - and plot file)

The function *Show coordinates* shows the X- and Y-coordinates of the current cursor position in the status bar. Using the right mouse button cancels the function.

Vertical section

(model file only

By selecting this function a straight line is drawn between two points, for which a Vertical section is displayed in a new window.

The following items are available only in the plot file mode.

Crop to zoom/crop rectangle ,

(plot file only)

The function *Crop rectangle* clips the actual graphic to the domain specified in the current zoom. As a result, all objects outside the domain are removed completely and objects which are partly inside the domain are clipped along the boundary. If several plots are to be clipped to the same size, the first of these is opened and defined by the zoom window. Then, the graphic is clipped and the file saved. The next plot to be loaded will be automatically zoomed correctly, provided that its coordinate system is identical with the first map. You can clip it immediately after loading.

Remarks: TIFF-images which lie completely or partly inside the zoom domain remain complete. Clipping TIFF-images is only possible with the menu *Bitmap → Cut*. Text objects are not decreased or clipped as long as their bottom left hand corner lies inside the clipping domain.

Warning! The program clips not the domain visible on the screen but the domain specified in the last zoom. Generally, the visible domain is larger because it corresponds to the proportion of the screen. Thus more parts of the graphic are visible.

The function *Crop to zoom* clips the visible domain.

Text

(plot file only)

By activating the function *TextBox* in the plot options all text boxes present in the plot file are displayed. This text box shows the approximate size, in which the printed texts appear.

By activating the function *Texts* the existing text fields are visible or hidden.

4.3.1 Structures

This menu contains all the representations that are possible for structures. If a model does not contain one of these items, this item is disabled, i.e. it is gray.
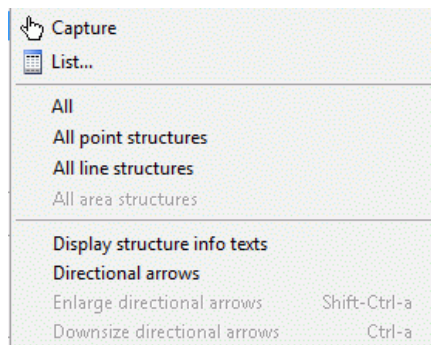
Capture **/ List,**

With the menu item *Capture* currently visible structures can be selected in the main window. A specific selection of structures is possible via the menu item *List…*. A list with the following columns of all existing structures is shown:

- Name (= name of the structure)
- Structure type (point p, line l, area f, transient gp, gl, gf)
- Attribute identifier

The selected structures (marked) are then displayed in the main window.

Show structures		
Name	**Type**	**Attributes**
Bach	l	VORF
Q=0	l	DATA
Randbed. Festpot NE-Rand	l	POTE
See	l	GLEI
Brunnenentnahmemengen	p	KNOT
Bach	l	MARK
Bach leakage	l	LERA
GW-Sohle	p	UNTE
14.	l	EICH
gela.txt	p	GELA
Seerand	l	MARK

OK   Cancel
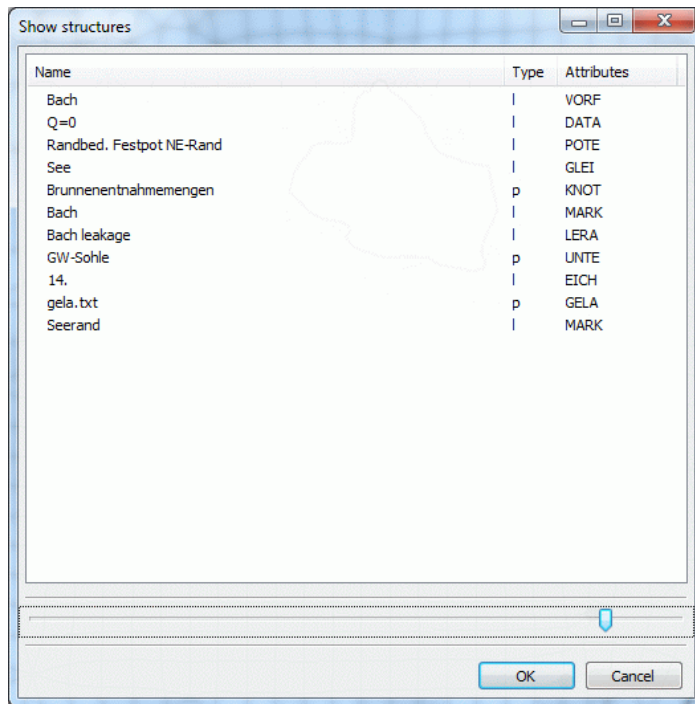
A block of structures is selected by dragging the cursor while pushing the left mouse button, or by selecting the first structure by clicking and selecting the final structure of the block by "Ctrl" and clicking.

The list can be sorted by clicking the column headings. The column width can be changed by dragging the column divider.

All, All Point-, Line-, Area structures

Depending on the selected menu item, the respective structures are displayed or hidden.

Selecting *All*, all structures are displayed or hidden. Alternatively, only point, only lines or only surface structures can be displayed or hidden.

Display structure info texts

The *structure info text* (name) helps to identify the structure in the dropdown list (see above) and can be switched on and off in the main window by selecting this menu item.

Directional arrows, enlarge, downsize

At first, the *Directional arrows* have to be displayed in the main window, then the user can *enlarge* or *downsize* them. The arrows indicate in which order the nodes of a line or area structure were created.

By selecting the menu items *Enlarge* or *Downsize directional arrows* they are scaled.

4.3.2 Show attributes

This menu contains all the representations that are possible for attributes. If a model does not contain one of these items, this item is disabled, i.e. it is gray.

	Isolines/area plots/values...
	Attribute-sign...
	Velocities...
	Transient...
	Hydrograph...
	Watercourse systems
	On watercourse systems...
	Text at nodes
	Text at elements
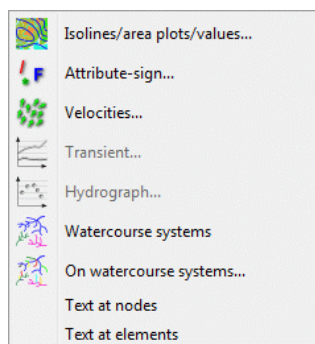
Isolines/area plots/values...

With this menu item contour lines, coloured areas, circular or linear representations of node and element data are created. It establishes a list of all attributes which are assigned to nodes or elements and the user can define which data and how the data is represented. After calling the function, the following input window appears:

Identifier, details of the attribute

First, the attribute is selected which should be represented. In a 3D model the choice of the layer number is additionally required. If a 2D data type (e.g. ground level) is selected, the layer number is automatically set to 1 and cannot be changed.

The min./max. range indicates the minimum and maximum value of this attribute. The next field shows to how many nodes or elements the attribute is assigned in relation to the total number of nodes or elements.

Type of presentation

Possible types are:

- Contour lines in a special colour
- Colour areas (In case of nodal data, the regions between contour lines are coloured; in case of elemental data, the individual elements are coloured depending on the element value.)
- Circle plot, where in each element / at each node a coloured circle is plotted. The radius of the circle depends on the value defined for the element/node. The minimum radius for the circles is 6.0 pixels. In the box *max. radius*, the maximum radius is defined. To plot all circles with the same radius, set the maximum radius to 6.0 pixels.
- Line plot, recommended for attributes that are assigned primarily along a line (e.g. leakage values or equal unknown potential heads along an edge of lake)

Intervals

There are various options:

■   The *equidistant* intervals are obtained by a uniform division of the distance between the minimum and the maximum value found. The number of divisions is defined in the box *Number*.

■   The type *From/to/number* defines an interval which is then divided evenly in the chosen number of intervals (text field: *number*). The defined intervals can be divided logarithmically by activating the check box *logarithmic* (recommended for K values).

■   The type from/to/distance is an interval determined by the user.

■   If *separate values* are entered, a box appears for the desired interval number and a table in which the interval limits can be defined individually.

If the presentation attributes are chosen, the selected data is stored in a special layer. This layer is listed in the project manager and can be displayed or not. Additionally, the legend of the layer can be displayed by selecting the check box.

Attribute-note…

By using the function *Attribute-note*, you can display the nodes/elements where for a certain attribute a note is defined, e.g. all VORF nodes with the note ' F' or all FLAE elements with the note ' *'. The following dialogue appears:
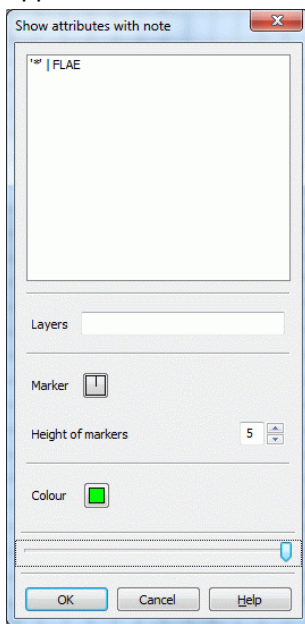
The corresponding menu contains a list of all assigned nodal and elemental attributes from which you can select.

For 3D models, the layer has to be specified. At last, the type, the height and the colour of the marker are determined.

Velocities…

With this menu item it is possible to represent a vector plot of the horizontal velocities defined by the element data types VV_X and VV_Y.

The data has to be imported first from the result data (*Attributes → Import data/computation results…*). The attribute identifier is pre-set to "VV_X" and "VV_Y". The following dialogue appears:
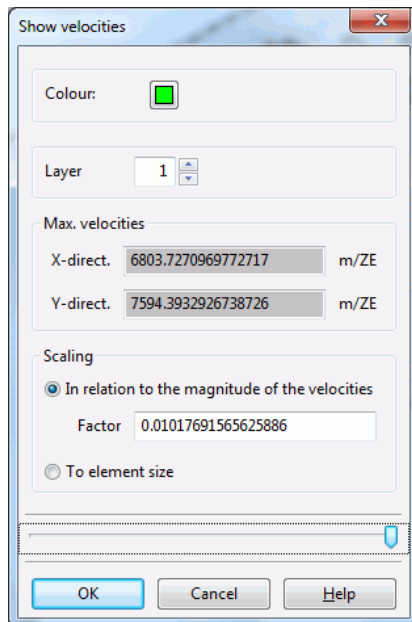
The display colour and, in a 3D model, the layer number have to be selected. The maximum velocities in the x-and y-direction are logged. You can choose between two forms of representation:

- The vector length is scaled depending on the velocity. This representation type is equivalent to the representation in the *plot generation*. The vector length is proportional to the velocity. The arrow is plotted into each element, with the arrowhead at the element centre. For very small velocities, the vectors are plotted with a minimum length.

- Alternatively, the vector length is scaled automatically, so that the arrow fits into the element. In this case, the arrow only represents the direction of the velocity, but not its value. The vector length is 80% of the average edge length in the element. The resulting arrow is plotted with its centre at the centre of the element.

Transient…

Transient data are only represented after they were imported via the menu items *Special features* → *Transient* → *Import transient input file.*

After selecting this menu item the following input window appears:
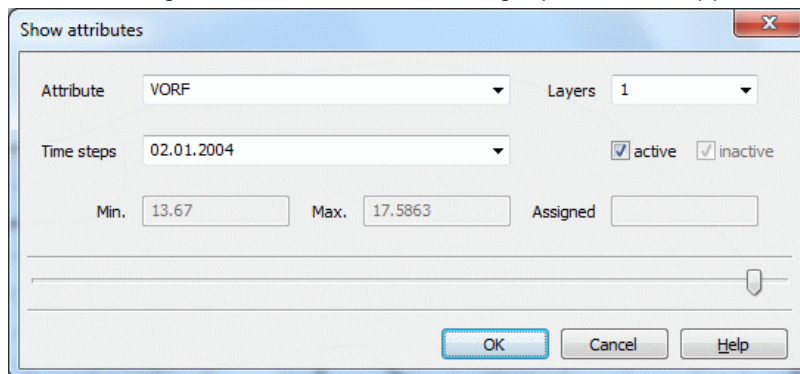
The attributes of the transient input file are shown. Depending on the attribute and the time step the existing minimum and maximum values and the number of assigned nodes or elements are displayed. In a 3D model, the desired layer can be selected.

By selecting the check boxes *active/inactive* you can distinguish between nodes or elements which are active or inactive. *Active* means e.g. for the attribute POTE, that at this time step the potential at this node is kept on the entered value. *Inactive* is a potential node, if it is provided with a minus sign in the transient input file. Then the potential is switched off.

The nodes or elements are marked with a circle plot, which are assigned to the chosen transient attribute, if the dialogue is closed with *OK*.

Hydrograph

After importing a transient input file via *Special features → Transient → Import transient input file*, the data can be displayed as a hydrograph by selecting a node or element with the appropriate attribute

Then the hydrograph of the selected object appears:
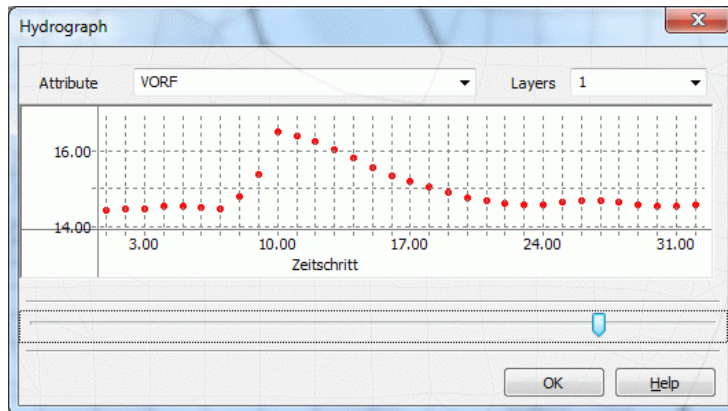
Transient data of deeper layers can also be selected.

Water course systems

After selecting this function, the existing water course systems are marked with arrows.
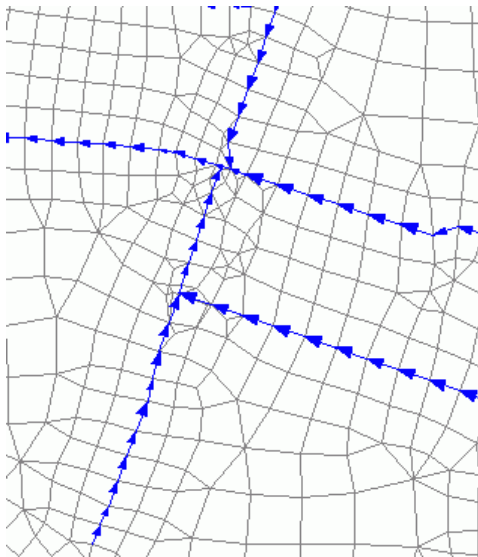
On water course systems…

By selecting this menu item it is possible to represent the attributes of the water course systems. These are the attributes:

VKST (Manning/Strickler coefficient, unit: $m^{1/3}$/s)

VBRT (wetted perimeter, unit: m)

VGRD (bottom slope, unit: pro mille)
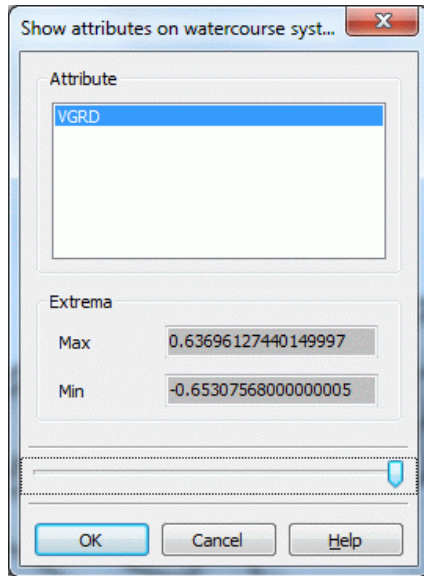
In particular, the assigned attributes of the water course systems can be checked with this menu item.

If the slope gradient (attribute VGRD) was calculated in the menu *Special features → Watercourse systems → Attributes* the slope direction on the element edges can be represented by arrows. Black arrows indicate a negative slope direction.

Texts at nodes/elements

Selecting this menu item, the texts at the nodes or elements specified in the mesh and/or 3d file are displayed.

4.3.3 Toolbars

Here, for each of the following menu items
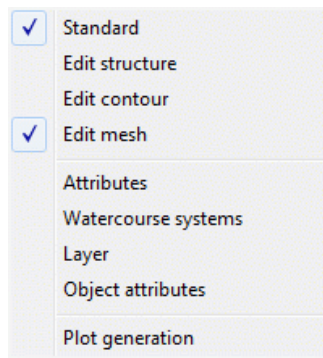
toolbars are displayed so that the corresponding buttons can be accessed directly depending on the processing step. Some toolbars are only available in the mode the of model file, the toolbars Standard, Layer, Object attributes and Plot generation are available both in the mode of the model file and of the plot file.

4.3.3.1 Toolbar Standard

The toolbar shown below corresponds to the "Standard" toolbar in the mode model file. In the plot file mode some toolbar buttons associated with the mesh generation are missing.

(p. 70)

(p. 70)

🖫	(p. 70)
🖫	(p. 70)
🖶	(p. 70)
↩	(p. 72)
↪	Redo (p. 72)
?📐	(p. 86)
?	(p. 86)
⊞	(p. 86)
⁺⁺⁺	(p. 86)
⊞	(p. 86)
	(p. 86)
	(p. 86)
	(p. 86)
⬭	(p. 86)
⬭	(p. 86)
	(p. 86)
	(p. 86)
	(p. 86)
✖	(p. 168)
👆⁺	(p. 139)
👆▱	(p. 139)
👆	(p. 106)

(model file only)

Symbol:	Meaning:
	Show structures (p. 89)
	Display only point structures (p. 89)
	Display only line structures (p. 89)
	Display only area structures (p. 89)
Text	Display structure names (p. 89)
✛	Generate point structure (p. 101)
✛	Generate line structure (p. 101)
✛	Generate area structure (p. 101)
	Modify structure - capture (p. 106)

Symbol:	Meaning:
	Modify structure from list (p. 106)
	Copy structure - capture (p. 101)
	Copy structure from list (p. 101)
	Deleta structure - capture (p. 101)
	Delete structure - in polygon (p. 101)
	Delete structure from list (p. 101)
	Delete all structures (p. 101)

4.3.3.3 Toolbar Edit contours
(model file only)

Symbol:	Meaning:
	Show all contours (p. 86)
	Generate point contour (p. 120)
	Generate line contour (p. 120)
	Create contour from structure - capture (p. 120)
	Create contour from structure - list (p. 120)
	Move contour points (p. 120)
	Assign maximum distance - capture (p. 120)
	Assign maximum distance - in area (p. 120)
	Divide contour n-times - capture (p. 120)
	Divide contour into n-parts - in area (p. 120)
	Capture contour segmentation points - pick (p. 120)
	Capture contour segmentation points - in area (p. 120)
	Delete contour - capture (p. 120)
	Delete contour - in area (p. 120)
	Delete all point contours (p. 120)
	Delete all line contours (p. 120)
	Delete all contours (p. 120)

4.3.3.4 Toolbar Edit mesh
(model file only)
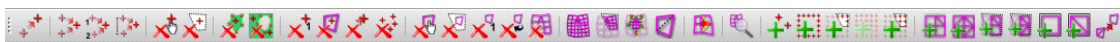
Symbol:	Meaning:
	Move node (p. 126)
	Merge 2 nodes - capture (p. 126)
	Merge 2 nodes - by number (p. 126)

Symbol:	Meaning:
	Merge 2 nodes by distance (p. 126)
	Delete nodes - capture (p. 126)
	Delete nodes - in polygon (p. 126)
	Delete nodes - inside selected structure (p. 126)
	Delete nodes - outside selected structure (p. 126)
	delete nodes by numbers (p. 126)
	delete nodes without elements or fractures (p. 126)
	Delete nodes with identical coordinates (p. 126)
	Delete all nodes (p. 126)
	Delete element - capture (p. 126)
	Delete element - in polygon (p. 126)
	Delete elements by numbers (p. 126)
	Delete elements with wrong orientation (p. 126)
	Delete all elements (p. 126)
	Refine mesh - global (p. 126)
	Refine mesh - in polygon (p. 126)
	Refine mesh - in area inside structures (p. 126)
	Divide quadrangle element (p. 126)
	Unbend mesh (p. 126)
	Mesh checks (p. 126)
	Insert nodes (p. 132)
	Generate boundary nodes - global (p. 132)
	Generate boundary nodes - in area (p. 132)
	Generate grid - globall (p. 132)
	Generate grid - in area (p. 132)
	Generate triangle and quadrangle elements - global (p. 132)
	Generate triangle elements - global (p. 132)
	Generate triangle and quadrangle elements - in area (p. 132)
	Generate triangle elements - in area (p. 132)
	Generate triangle and quadrangle elements -from inner boundary (p. 132)
	Generate triangle elements from inner boundary (p. 132)
	Merge triangles to quadrangles if possible (p. 132)

4.3.3.5 Toolbar Attributes

(model file only)

Symbol:	Meaning:
	Edit node -capture (p. 139)
	Edit node by number (p. 139)
	Edit element - capture (p. 139)
	Edit element by number (p. 139)

4.3.3.6 Toolbar Watercourse systems

(model file only)

Symbol:	Meaning:
	Group automatically (p. 101)
	Identify flow direction (p. 101)
	Change direction (l-structures only) (p. 101)
	Manage/assign structure(-groups) (p. 101)
	Show attributes of watercourse systems (p. 90)
	Show flow direction (p. 90)
	Calculate gradients from ground surface (p. 174)
	Controls (p. 174)

4.3.3.7 Toolbar Layer

(model and plot file)

Symbol:	Meaning:
	Modify colours (p. 168)
	Optimize coloured areas (p. 168)
	Activate layer (p. 168)
	Display project manager (p. 86)
	Delete layer (p. 168)
	Move layer forward (p. 168)
	Move layer backward (p. 168)
	Move layer to front (p. 168)
	Move layer to back (p. 168)
	Move layer (p. 168)

4.3.3.8 Toolbar Object attributes and drawing

It appears a toolbar to define the object properties and another for drawing new objects. Both toolbars are used in the mode model and plot file.

Object attributes:

By displaying this toolbar object attributes can be selected and assigned to the corresponding object (marker, line, text) by clicking on the button "Assign object attributes" (⬆).

Before, the existing object properties can be displayed by selecting the button "Display object attributes" (⬇) and then selecting the desired object with the left mouse button.

Symbol:	Object attributes:
	Select object colour
	Select type of marker
	Select line type
Helvetica / Standard	Select the font and type
H	Select font size or marker height
W	Select angle
T	Insert text
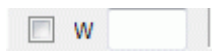
Drawing

| Symbol:	Meaning:
	Display object attributes
	Assign object attributes
	Draw marker (p. 170)
	Draw line (p. 170)
	Draw triangle (p. 170)
	Draw rectangle (p. 170)
	Draw polygon (p. 170)
	Draw filled triangle (p. 170)
	Draw filled rectangle (p. 170)
	Draw filled polygon (p. 170)
	Generate text object (p. 170)
	Enlarge all text objects (p. 170)
	Scale down all text objects (p. 170)
	Move object forward (p. 170)
	Move object backward (p. 170)
	Move object to front (p. 170)
	Move object to back (p. 170)

4.3.3.9 Toolbar Plot generation

It appears a toolbar from which the three types of plot generation (Top view/map presentation , View/cross

section presentation , 3D wireframe) can be selected. Also it is possible to reload a document.
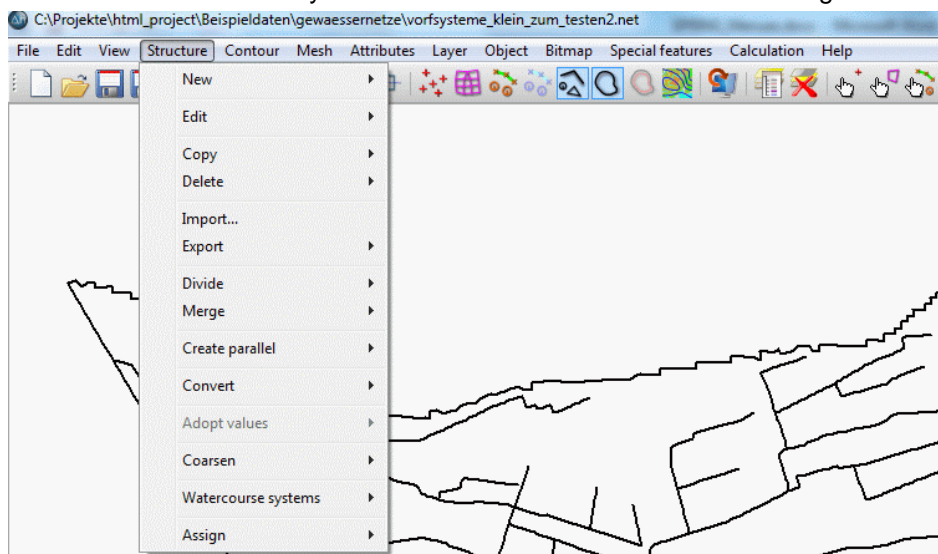
Plot generation:
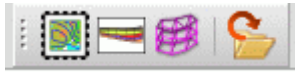
The types of the plot generation (p. 355) are described in the corresponding chapter.

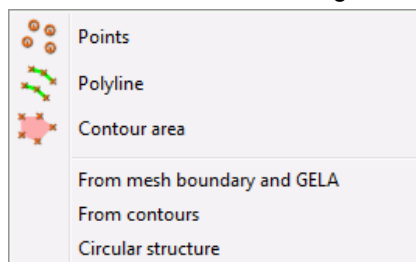The button makes it possible to reload the currently open document.

4.4 Structure

The structure menu is only available in the model mode. The following menu items are available:

New

In the submenu the following items can be chosen:

It is chosen what structure type

- Points
- Polyline
- Flat
- From mesh boundary and GELA
- From contours
- Circular structure

should be created.

After selecting the type of structure, the desired structure can be created in the main window of SPRING. After confirming the structure with the right mouse button, the input window: "(p, l, f) modify structure" appears.

This input window is described in detail in the menu item *Structure → Edit.*

The menu item *From mesh boundary and GELA* creates a new l-structure on the mesh boundary (with points at all boundary nodes) and assigns to them the values of the ground level GELA. This is very useful if the user wants to create boundary potential heads with a value "GELA - x".

The menu item *From contours* generates from the existing contours new structures. It is active only when contours are already present.

Circular structure

Selecting the corresponding menu item, a circular structure can be created. The centre of the circle is determined with the cursor. An input window appears, in which the parameters of the circle are defined:
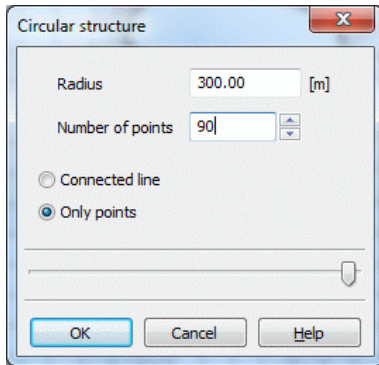
The *Number of points* describes the number of structural points that will be created for the discretization. At 90 points, the distance of the points to each other is 4 ° of the circumference.

The selection *Connected line* and *Only points* describes how the circular structure is displayed. The following figure shows the representation *Only points*:
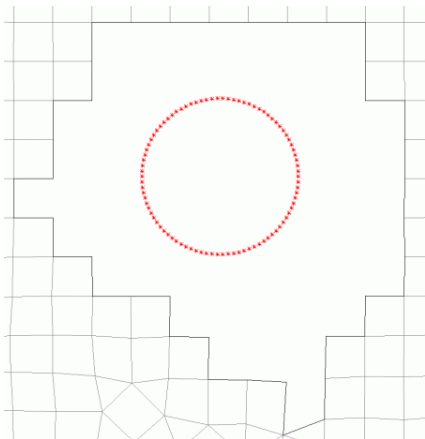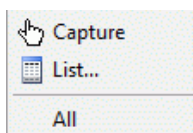

Fig. 62: Circular structure, displayed with points

Edit

Since this menu item is very complex, it is treated in a separate chapter.

Copy

The submenu allows the choice which structure is to be copied:

With *Capture* or *List* single structures can be specifically selected. It is also possible to copy All structures

The selected structure is completely copied (attributes, structure type, coordinates, values and representation parameters). A copy can be recognized by the appended @ at the structure name.

With the help of copying structures it is possible:

■   To generate several geometrically identical structures with different identifiers and values. A line structure for a receiving water course is, for example, three times required: for the water course level (attribute

VORF), the leakage coefficients (attribute LERA) and for the automatic marking of the polygon (attribute MARK).

■ To generate geometrically similar structures by copying and then editing.

■ To save structures which are converted in transient structures. This process is not reversible

Attention! Original and copy are aligned, so that only the copy is captured as the first object. Here it is helpful to choose the structure out of the *List*.

Delete

The submenu offers various options to delete structures:
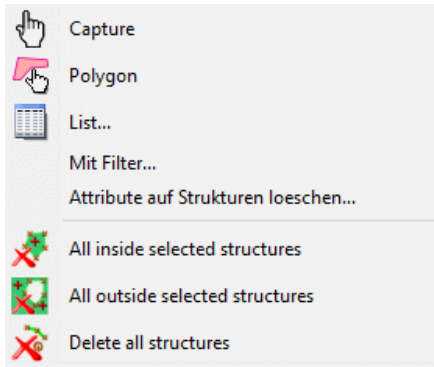
By choosing *Capture* (separately), one or more structures are successively selected and then deleted.

By choosing *Polygon (Polygon),* all structures in the selected polygon are deleted.

By choosing *List…* the structures which should be deleted are selected in the list.

By choosing *With Filter...* a further selection window appears for the definition of the filters:
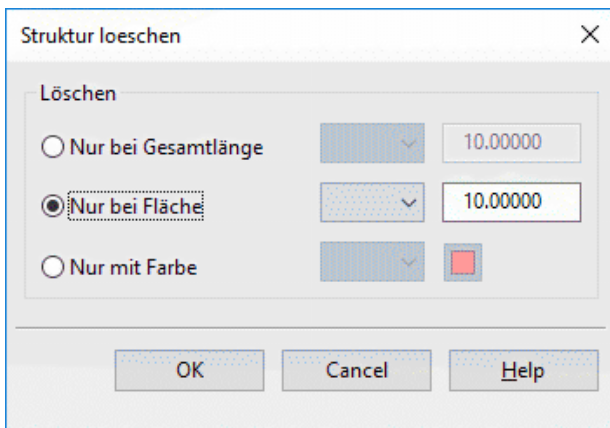
By selecting *Delete attributes on structures…* a further selection window appears to define the attributes:
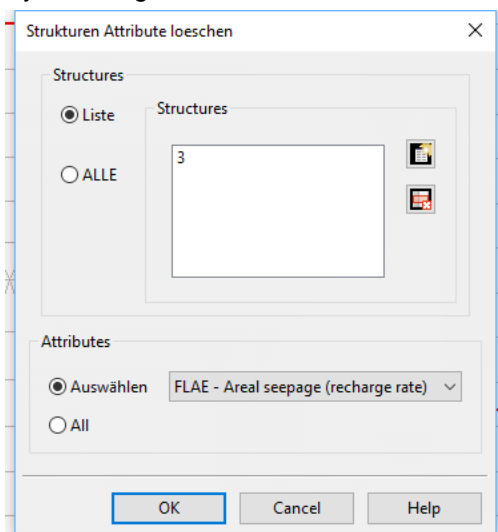
The attribute selection can be used to delete special attributes on structures that have been assigned multiple attributes or to delete all attributes of a type.

By choosing *All inside/outside selected structures* the corresponding structures within or outside the selected area are deleted.

By choosing *Delete all structures* all structures are deleted.

Import…

Since this menu item is very complex, it is treated in a separate chapter: Import of other data format

Export…

By activating the menu item *Export ...* it is possible to export a selection of structures by *List...* or *All...* structures. If a selection of structures is to be exported the list of structures appears. After selecting the structures and confirming with *Save* the file selection box appears. If all structures are exported, the file selection box appears at once and the following input window appears:
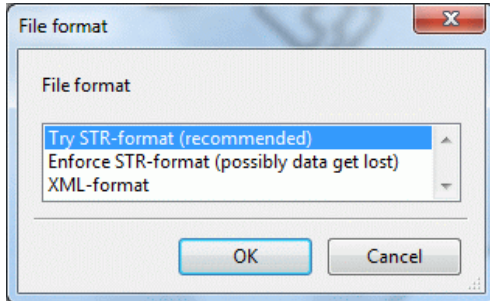
It is possible to save the files in the structure file format (ASCII) or in the structure format (XML).

Divide

With the help of this menu item, existing line structures can be divided at several points. For instance, you can divide a LERA-structure and assign different leakage coefficients to the segments.

After the line structure has been selected (*Capture/List…*), it is automatically divided at the chosen points. Each segment inherits the identification, structure type, coordinates, values and plot properties from the original line structure. Only the structure info text is completed by the text "(n)" where n represents the number of the segment.

Merge

With the help of this menu item two existing characteristic point or line structures can be merged into one.

It is possible to *Capture* the point or line structures in the main window or to select them in the *List...*

Merging two selected point structures means the points of the second (coordinates x, y + values) are added to the first. The info text of the second dataset is lost.

For merging two selected line structures, the distances between start and end points are computed. The lines are merged at the ends with the shortest distance between them. If the two lines have a shared start or end point, one of the values at this point is deleted. The new line dataset inherits the description of the structure line which was selected first. The second description is lost!

Create parallel

The procedure and the input windows are explained in detail in the chapter: "Incorporating a new structure in an existing model" of the tutorial.

Convert

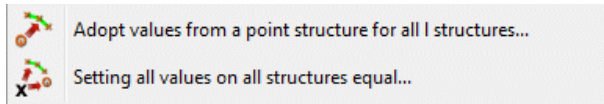First, a submenu appears in which the following choices are available:
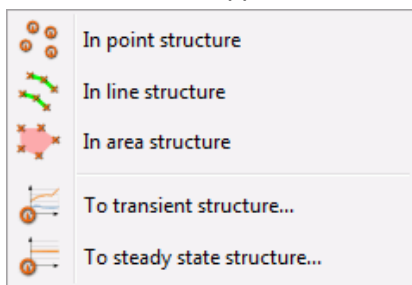
After converting the structure by selecting it the change is in the structure list recognizable (p, l, f). It is a transformation in any direction possible.

Converting a structure in a transient one the change is recognizable by the additional "g" in the structure type.

Adopt values

The following menu item appears:

|	
	Adopt values from a point structure for all l structures...
	Setting all values on all structures equal...

This item is only active when at least one point and a line structure are present.

■   Adopt values from a point structure for all l structures...

The menu item allows you to place the values of a point structure (eg. a terrain model) to all line structures.

■   Equate values of all structures... x

With the help of this menu item, a single value can be assigned to all structures. It appears the following input window:
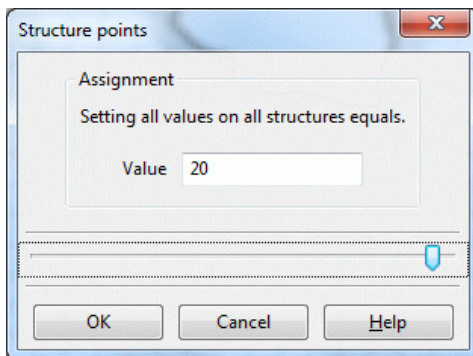
After entering and confirming the value with *OK* all line, point or area structures get the specified value at their existing structure points.

Coarsen

After choosing the mode of selection in the submenu:

	Capture
	Polygon
	All...
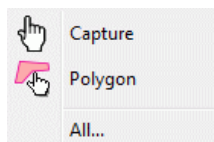
and selection of the structure another submenu appears in which the maximum distance for the coarsened line structure is defined. Points are deleted if the distance of the point to the simplified line is smaller than the defined distance.
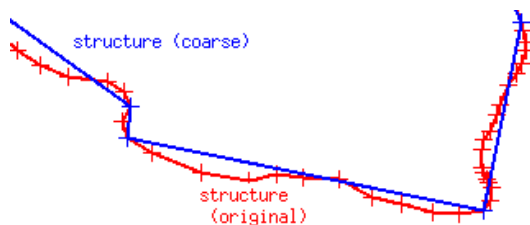
Watercourse systems

Assign

4.4.1 Edit structures

After selecting the structure via

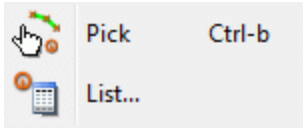appears one of the following input windows. Which one depends on the structure type (To pick a structure it has to be displayed in the main window: or via *View → Structures*).

The blocks *Data type* and *Info* are equal for all structure types.

The blocks *Presentation*, *Values* and *Geometry* depend on the type of the selected structure.

Data types and Info:
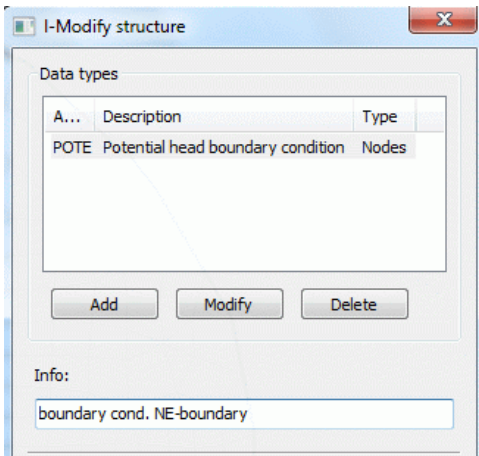
Data type: The attribute is defined which is assigned to this structure. By choosing *Add* more than one attributes are assigned to this structure (e.g. VORF and LEKN at a water course). A window appears with the selectable attributes.

By choosing *Modify* the attribute of the structure is changed.

Delete deletes the marked attribute.

The field *Info* shows the structure name. When the mouse pointer moves over the data type window, the statistics of the active structure are displayed.
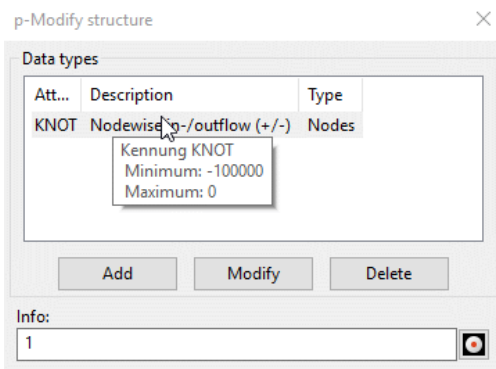
Presentation, Values, Geometry:

■   in p-structures is the following input possible:
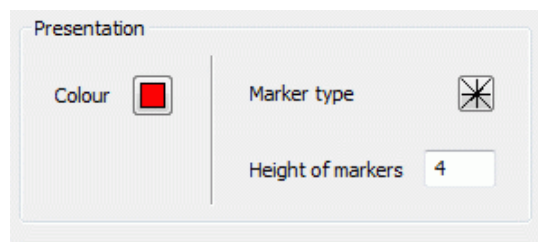
Presentation:

The default colour, the marker type and the marker height (in pixel) can be changed. This helps to distinguish the various structures in the graphical representation.
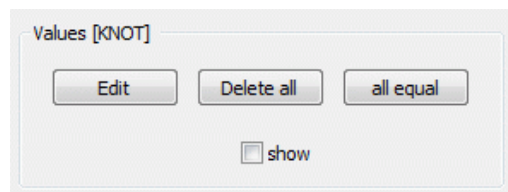
Values:

Edit: A table with the x, y coordinates and the assigned values of the individual structural points appears. To each structure point a value can be assigned:

The values can be assigned individually, by multiple markings or globally by selecting the column:

After clicking on one or more cells with the left mouse button and simultaneously pressing the "Shift" or "Ctrl" key and then clicking the right mouse button the following context menu opens:
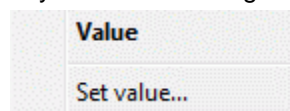
After choosing "Set value..." a dialog window opens in which the value of the selected cells is defined:
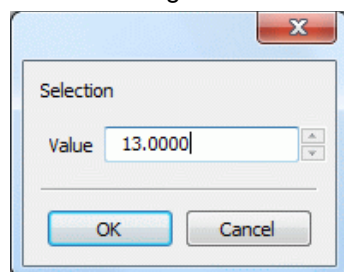
By confirming with "OK", the values are assigned.

Delete all: All assigned values at the structure points are deleted.

All equal: All structure points get the same value:
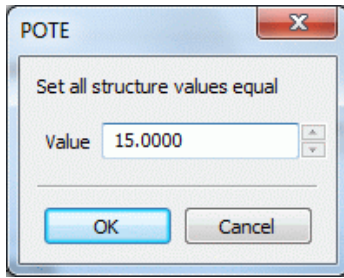
Show: The values of the structure are shown in the main window.

Geometry:
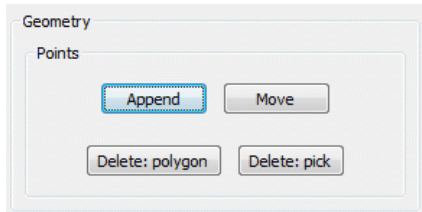
Append, Move: By *Append* further structural points are added to a p-structure. By *Move* individual points are changed in their position.

Delete: Polygon/Delete: pick: Structure points can be deleted in a polygon or individually.

Only by confirming with the OK button the changes take effects to the structure!

■   in l-structures is the following input possible:
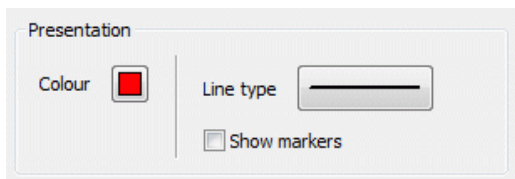
Presentation:

The default colour and the line type can be changed. This helps to distinguish the various structures in the graphical representation. In addition, by checking the box *Show markers* all structure points are displayed in the main window, so the individual sections of the line structure are visible.

Values:
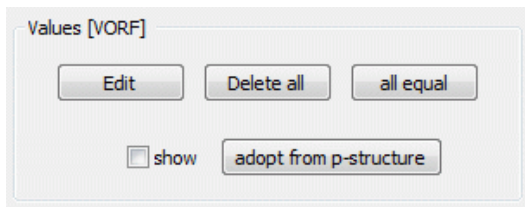
In addition to the input described at the p-structures the following assignment is possible to the l-structures:

Adopt from p-structure: The values of a point structure are applied to a line structure. In the dialogue a maximum coordinate distance (dx) and the ratio of the x-and y-distances of the node / element center point (eps (x) / eps (y)) to the structure point are defined.
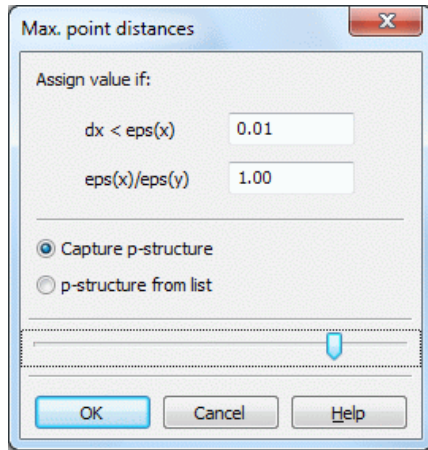
If a point of the selected point structure is close enough to a point of a line structure, the line structure point inherits the value from the point structure point.

The desired p-structure can be either *captured* (if the structures are displayed) or selected from a *list*.

Geometry:
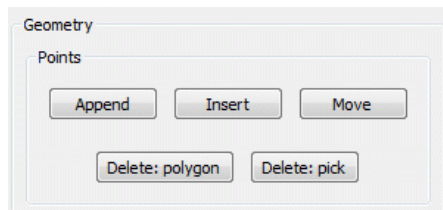
Append, Insert, Move:

By *Append* further structural points are added to the end of an l-structure. By *Insert* more points can be added between two existing structure points. By *Move* individual points are changed in their position.

Delete: Polygon/Delete: Pick: Structure points can be deleted in a polygon or individually.

Only by confirming with the OK button the changes take effects to the structure!

■   in f-structures is the following input possible:

Presentation:
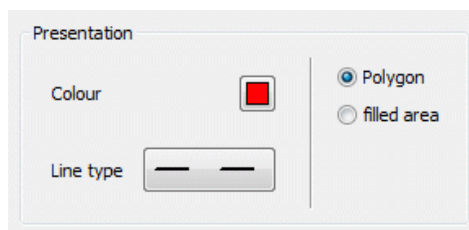
The default colour and the line type can be changed. The line type is only relevant if the structure is displayed as *polygon*. The colour is used when the structure is displayed as *filled area*. This helps to distinguish the various structures in the graphical representation.

Values:
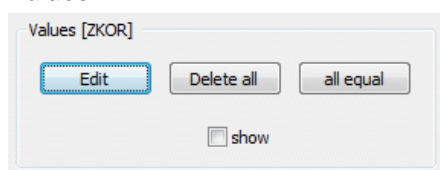
The possibilities for the input are those that are possible by p-structures.

Geometry:
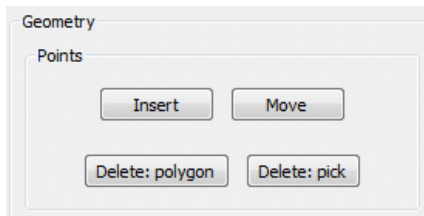
Insert, Move: By *Insert* more points can be added between two existing structure points. By *Move* individual points are changed in their position.

Delete: Polygon/Pick: Structure points can be deleted in a polygon or individually.

Only by confirming with the OK button the changes take effects to the structure!

4.4.2 Import of other data format

For the *import of structures* the user can choose between various data formats:
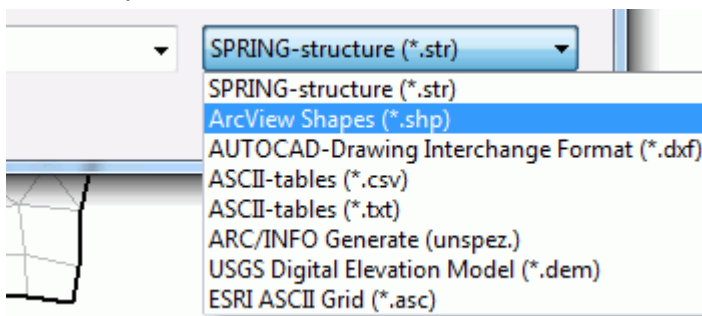
While there are any other queries when choosing the structure file format, further input windows appear when choosing "SPRING-foreign" formats.

- ArcView Shape format (*.shp)

Point, line or area information available in the ArcView shape format, can be imported as structures.

After selecting a *.shp file appears the following input window:
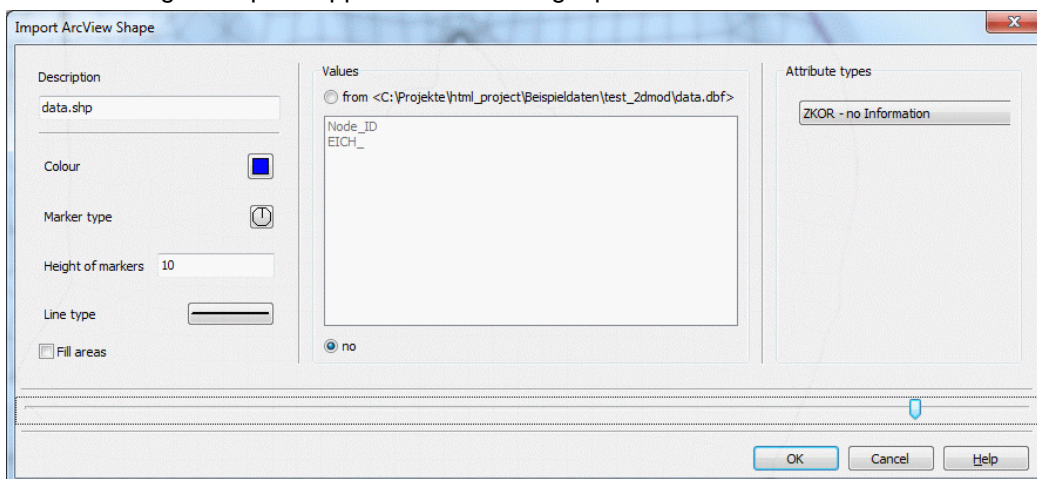
Like *editing* a structure the display attributes are chosen and the values and attributes are assigned.

- AUTOCAD *.dxf-format

Point, line or area information available in the AUTOCAD *.dxf-format can be imported as structures.

After selecting a *.dxf file appears the following input window:
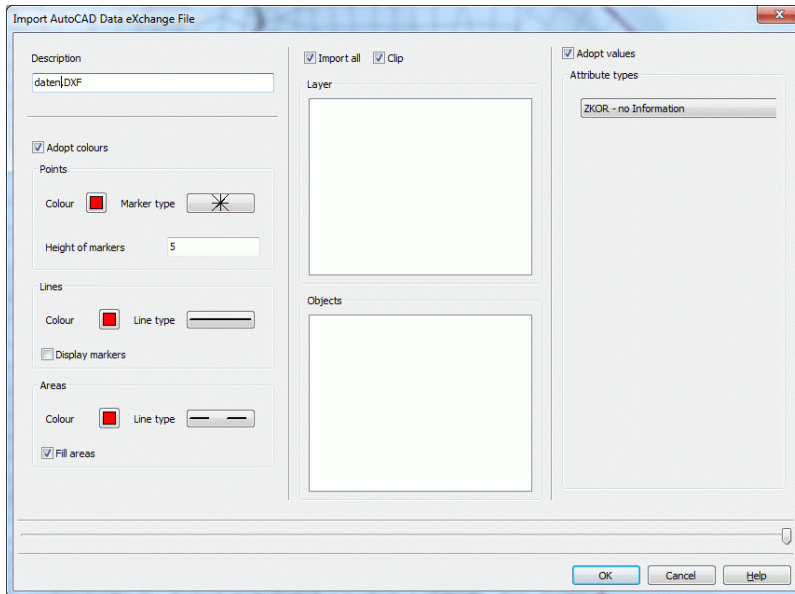
In addition to the presentation attributes for the various structure types and the attribute assignment it can be chosen if all layers of the *.dxf file, or only some are imported from a list selection. The values can also be taken over.

■   ASCII-Tables *.csv-format:

In many cases required structures already exist in a data base. It is possible to save the data, i.e. coordinates and possible values, as a tabular ASCII file. This can be a tabular text with comma separated values (*.csv format). Such data can be imported as point-structures in SPRING.

After selecting a *.csv file appears the following input window:
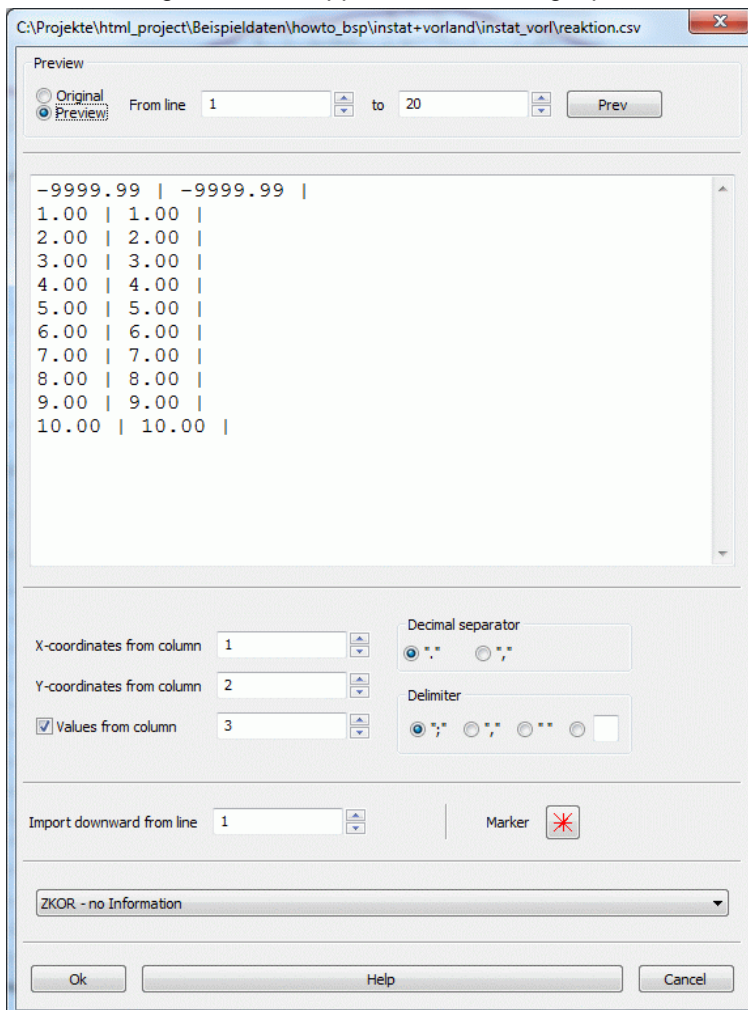
Here the user must enter explicitly to which column the x and y coordinates and z values are assigned. The used decimal and column separator is indicated. A coloured marker is assigned to the data type by which the data points are shown.
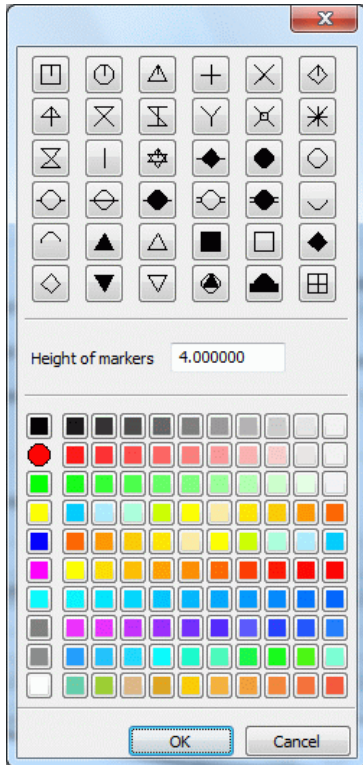
The import of the data can be done from a specific line. If column headings are present in the file, then the import ist to start from row two.

Finally, the data type is selected, to which the z values are assigned.

■   ASCII-tables *.txt-format:
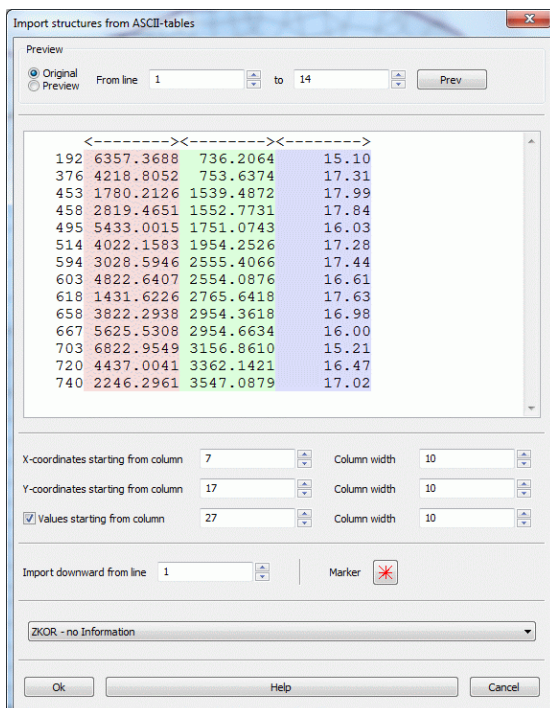
The respective starting column and the column width are associated with the x and y coordinates and the z-values. A coloured marker is assigned to the data type by which the data points are shown (see *.csv format). If column headings are present in the file, then the import is to start from row two. Finally, the data type is selected, to which the z values are assigned

■   ARCINFO generate-format (*.bru, *.pkt, *.umr)

Point, line or area information available in the ARCINFO generate format, can be imported as structures.

After selecting a *.bru, *.pkt or *.umr file appears the following input window.
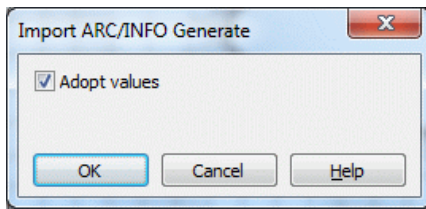
There is only the question if the existing values should be transferred.

■   USGS (*.dem) and ESRI ASCII Grid (*.asc)

Data files are imported directly.

4.4.3 Water course systems

This menu item is about the treatment of watercourse systems. The following items are available:

■   River system preparation...

The following input window appears:
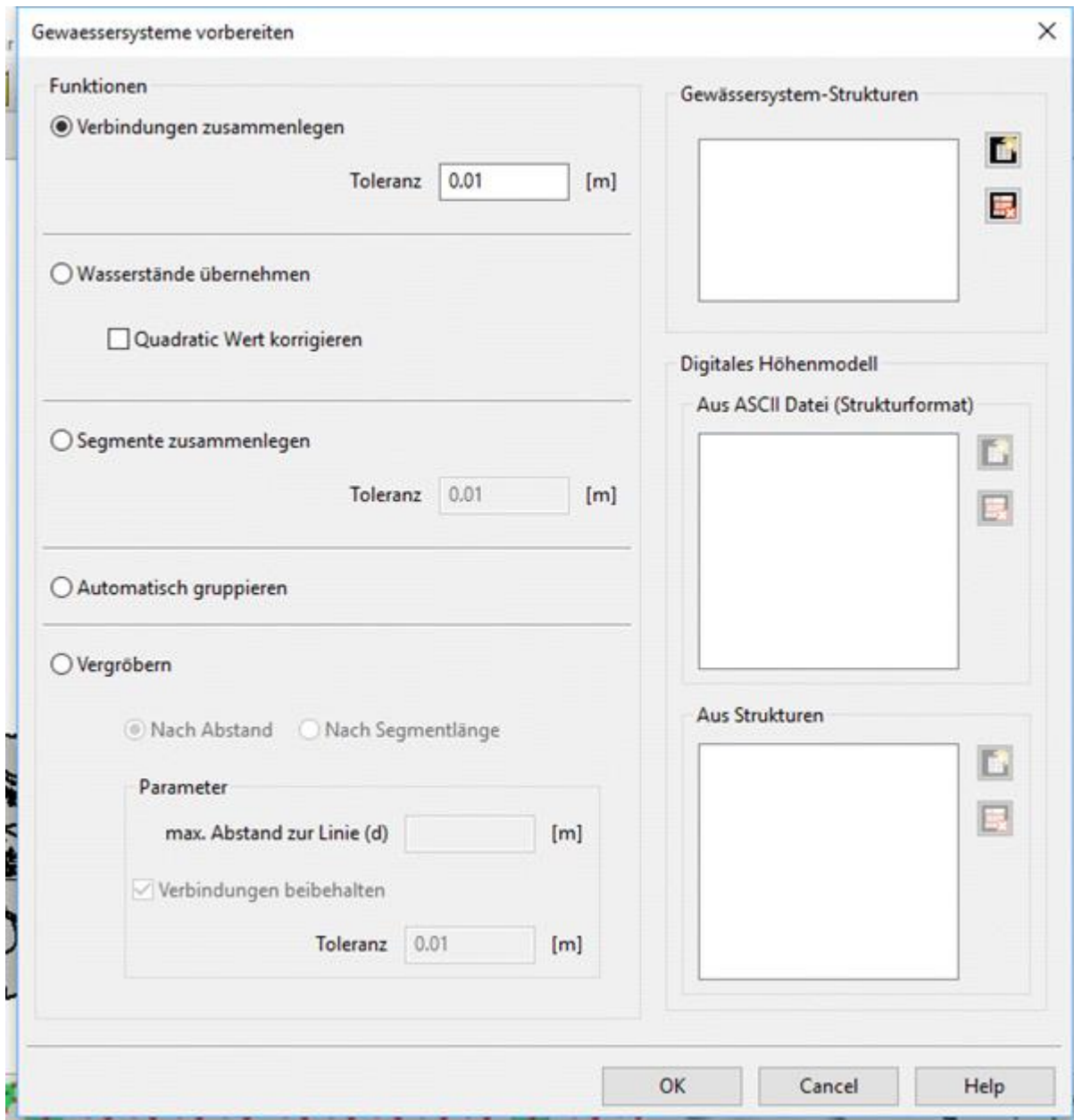
■   Joint connection tool

The joint connection tool is used to automatically detect and connect structure joints that lie within a user-specified radius of each other throughout the model. This is especially important with regard to further manipulations for which the river system must first be grouped, since the grouping algorithm generally construes as disparate points and therefore rivers whose coordinates do not precisely coincide. The algorithm simply searches for points within the specified radius and subsequently attempts to join them in case they belong to separate structures—points from the same structure will not be joined, which would otherwise lead to an overlap. All structures will retain their initial values as no calculation is involved the process. The number of modified points is shown when the procedure ends.
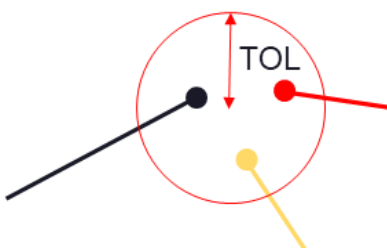
■   Water level calculation

This feature can automatically calculate and correct water levels and determine flow directions of river systems in the model. The algorithm employs a robust combination of geometrical and geological analyses for this purpose. Needless to say, the outcome of this tool heavily relies on the correctness and accuracy of the provided input data.

In general, the following steps are required to utilize this tool. Firstly, the structures corresponding to the river system must be selected. These structures are necessarily line structures, and any other type would internally be rejected. Nonetheless, care should be taken regarding the correctness of the selected structures since any line structure would indiscriminately be acceptable as the algorithm has virtually no way of determining whether a given line structure is in fact part of the river system or not. River structures may be selected from a list on the top left side of the dialog shown in Figure 1. Next, the digital terrain model data should be selected, which could be in the form of a structure in the model or an ASCII file. In case the digital terrain model is selected as a structure, it must necessarily contain the GELA attribute since this attribute is essential for interpolation and subsequently the statistical analysis in the algorithm. Multiple structures and/or ASCII files may be selected simultaneously, in which case they will be treated as a single aggregate point cloud.

After the successful selection of the river system structures and the digital terrain model data, all the necessary data for this tool is available. The algorithm determines water levels through all river systems in the model using a combination of Gaussian interpolation and a specialized correction algorithm. Flow directions are detected by a robust algorithm that employs a combination of geological and statistical analysis of the river system. Subsequently, the obtained values are corrected based on a series of criteria that ensure both continuity and compatibility through each river system. Moreover, flow directions are visualized with arrowheads, and the results of the analysis are stored in the VORF attribute of each individual structure—note that the existing data in this attribute will be overwritten.

To add a few caveats worthy of mentioning, it is reiterated that one should particularly be scrupulous to ensure the soundness of the selected input data, namely, continuity of the structures at joints where a single river is represented by multiple structures must be checked since the algorithm does not automatically check for discontinuities—the joint connection tool may be used prior to this tool for this purpose. Additionally, the accuracy and appropriateness of the digital terrain model should be checked for the river systems since water levels are inescapably dependent on the interpolation results, and while flow directions are computed using a combination of methods that try to deduce the most accurate results from the provided information, an excessively coarse digital terrain data might lead to insufficient values for individual rivers. Furthermore, the correction algorithm works at its acme when river structures are as long as possible, i.e., there are abundant data points for the algorithm to avail itself of. Therefore, it is recommended that the structures be prepared in such way prior to water level calculation, ideally with each river corresponding to a single or only a few structures. The merging tool might be of help in cases where manual manipulation of the structures is strenuous. This tool runs in parallel to boost efficiency on several layers whenever possible.

■   Merging tool

The merging tool will simply look throughout the river system and detect river segments that may safely be combined without the potential risk of changing the geometrical interpretation of the system. This tool is helpful in a number of scenarios, e.g., when the river system is imported from shape files, leading to individual rivers being represented by numerous small segments that could easily accumulate to thousands of structures in relatively large models. Handling fewer structures, of course, is often more desirable and at the very least facilitates the modeling process, not to mention that it is sometimes necessary to work with a continuous structure.

It is important to note that the algorithm does not attempt to make speculations regarding the general path of rivers, and the merging process stops when there are ambiguities. For example, a Y-shaped group of rivers would be merged up to the three main branches, but one path is not further preferred over the other. On the other hand, structures are not split under any circumstance. These rules are to ensure generality of the algorithm for all potential geometries. Another important note is that as there are no attribute calculations involved in this tool, all structures will retain their initial values. Furthermore, the same rules from the main structure menu item for manually merging two structures apply here. This tool runs in parallel whenever possible..

■   Automatic grouping tool

The automatic grouping tool is used to put together connected assortments of structures in individual groups. This procedure is necessary prior to some calculations. This tool does not discriminate the selected structures based on their attributes, i.e., all selected structures are valid as long as they are line structures; hence, when used for river systems, care should be taken to include only structures in the model that are part of the river system.

■   Coarsening tool

Coarsening is typically useful where the handling of the model and calculations are weighted down by existing structures carrying excessive or redundant points that may be represented by a coarser structure with fewer points without losing substantial information. The coarsening tool allows users to coarsen structures based on two different criteria.

The **first method** is based on the perpendicular distance between each point within the structure and the line formed by connecting the two points immediately surrounding it. The maximum distance may be specified using the corresponding field in the dialog. Points with a shorter distance than the specified value will be detected and removed by the algorithm.

The **second method** is based on the length of each segment in the structure. Specifically, the algorithm removes the minimum number of points required for the resultant structure to consist of segments whose lengths are greater than or equal to the user-specified value.

In addition, it is possible using the joint detection option to exclude points with connections from the coarsening process. In other words, when this option is enabled, points within a structure that are connected to other structure points will remain in the coarsened structure regardless of the location in the initial structure and the selected coarsening parameters. This option ensures, for example, that the coarsened structures within a given river system will retain their initial connections, which prevents loss of geometrical information while allowing the system to be represented using fewer nodes. One important point here is that a connection is defined when a structure point lies within a user-specified value of another structure point. The tolerance defines the radius within which a point is considered as connected to another point. It translates into precise connections if the tolerance is small enough, while a larger tolerance will lead to the exclusion of points that are only close enough to other structure points. All existing line structures in the model—and only line structures—are taken into account for this option.

Finally, it is important to note here that structure points that satisfy the mentioned criteria along with all their attributes are removed during this procedure. Therefore, a proper choice of the coarsening method and parameters is crucial to obtaining appropriate results..

■   Group automatically

The automatic grouping tool is used to put together connected assortments of structures in individual groups (line structures with attribute VORF). The related water course systems are coloured identically.

■   Join connection automatically...

Es erscheint ein weiteres Eingabefenster:
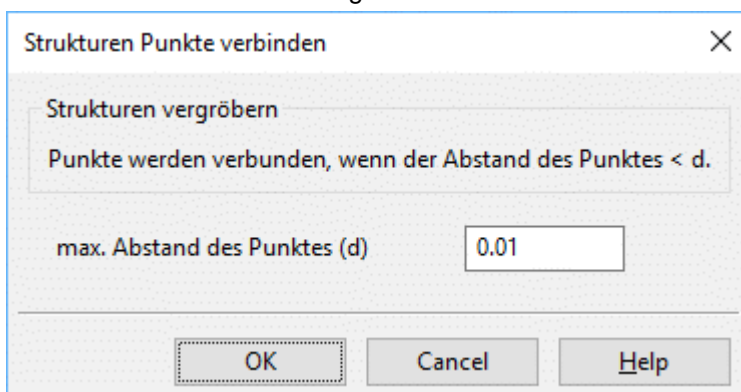
Strukturen Punkte verbinden   ✕

Strukturen vergröbern

Punkte werden verbunden, wenn der Abstand des Punktes < d.

max. Abstand des Punktes (d)   0.01

OK   Cancel   Help

The joint connection tool is used to automatically detect and connect structure joints that lie within a user-specified radius of each other throughout the model.

■   Identify flow direction

This menu item determines the direction of flow of watercourse systems by their water level. The flow direction is marked by arrows.

■   Change direction (l-structures only)

By this menu item the direction of selected watercourse systems is changed.

■   Directional correction of values

By selecting this menu item, structure values, which heights (e.g. water course levels) do not descend in the flow direction, are sorted out.

■   Manage/Assign…

After *automatically grouping* the structure groups can be managed. An input window appears to *Show/Disband/Assign* structure groups:
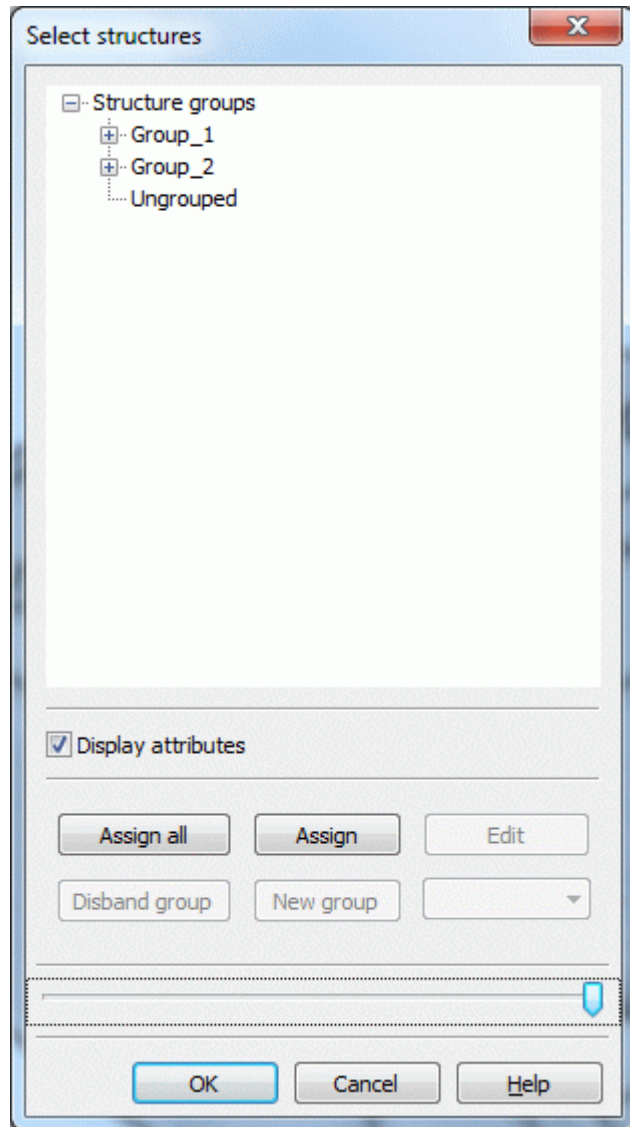
The dialogue shows:

■   the to individual water systems combined structure groups and
■   the structures not combined to water systems.

Not to the water systems belong point or area structures and linear structures to which the attribute VORF is not assigned. If there are structural groups, these can be assigned either "all" or "individually".

With *Disband group* individual groups can be deleted. Via *New group* new structure groups can be created.

After assigning them all related water systems can be displayed via *View → Show attributes → Watercourse systems.*

The graphics show a comparison of water systems that are not prepared and grouped (left) and that are geometrically prepared and grouped (right).

Without preparing and grouping	With preparing and grouping
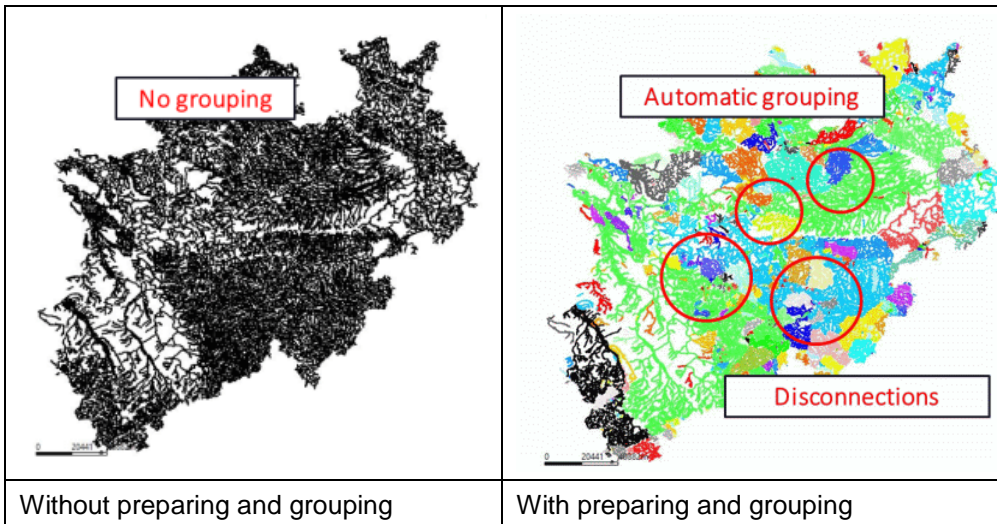
The same colors show water systems, which form a unit through their connection and their bottom or flow gradient.

4.4.4 Assign structures

Structures contain data i.e. values at the points of the structures beside the geometry information. These data, combined with the structure identifier, can be assigned to nodes and elements of the mesh.

First, the structure must be specified via *Capture* or *List…*

Then an input window appears, in which the assignment parameters are defined:

For point structures:
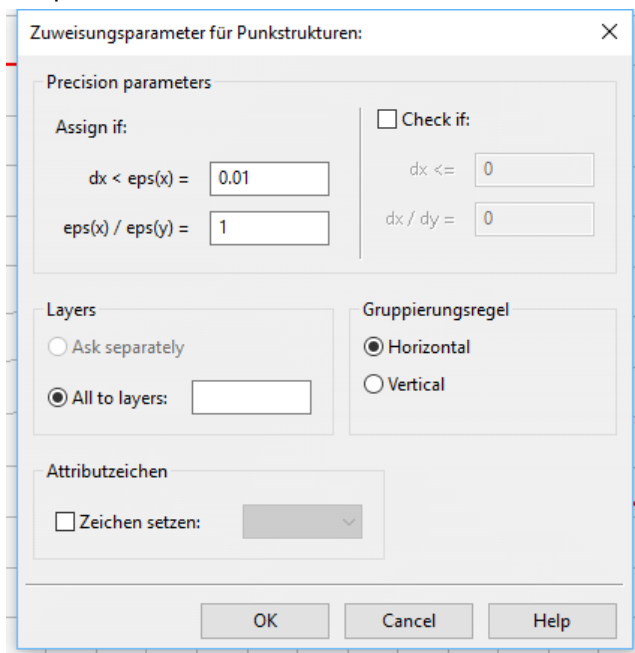
Meaning of the dialogue entries:

- Assigning point structures:

In case of nodal data, the node must lie within the specified distance dx < eps(x) from the structure point. Then the value (if existing) is assigned automatically. In case of elemental data, the element centre must lie within the specified distance dx < eps(x) from the structure point.
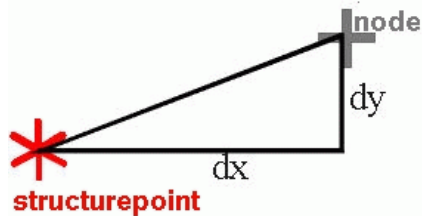
Fig. 63: Distance between the node and the structure point

If the ratio eps(x)/eps(y) is equal 1, the structure data are assigned to all nodes or element centers within a circle of radius dx around the structure point. However, the distance dx <eps (x) = 60 m and the ratio eps(x)/eps(y) is set to 10, the structure data are assigned to each node or element centers within an ellipse with a semi-axis x = 60 m and y = 6 m around the structure point. If the distance dx <eps (x) = 60 m and the ratio eps(x)/eps(y) is set to 0.1, the structure data are assigned to each node or element centers within an ellipse with a semi-axis x = 60 m and y = 600 m around the structure point.

If the *check if* is activated a tolerance limit can be determined. That means, the nodes or element centers that have a slightly larger distance than dx to a structure point, get the structure data anyway after a query.

For 3D models, the node or element layer is to be specified. It is possible to specify several layers at once (layer numbers separated by comma or semicolon or joined by a hyphen). The data can be grouped immediately horizontally or vertically.

Depending on the data type, the appropriate attribute character can be assigned immediately.

- Assigning line structures:

The dialog corresponds to the point structure assignment, except for the point-specific properties.

Assigning line structures makes only sense for node wise data. There must be at minimum one value defined at a point of the structure line.

All nodes which are according to the distance definition above are close enough to the line structure get automatically a value of the relevant structure data.
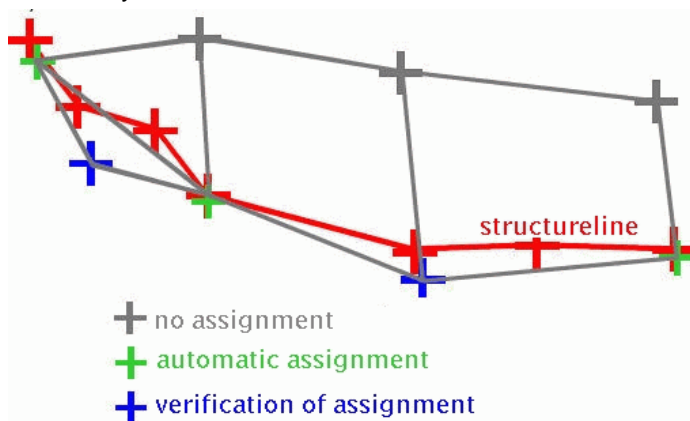
Fig. 64: Assignment of structure lines

Depending on the type of data a different assignment is used:
- In case of individual values, the value is linear inter- or rather extrapolated.
- In case of data along polylines and group data, all nodes receive the first value of the line structure.

Attention: The assigning of data along polylines or group data (e. g. LERA, MXKI or BILA) always uses only one value. Therefore, different values along the structure line are possible but not useful. For the representation of a receiving water course with several segments with different LERA values, different structures must be used.

- Assigning area structures:
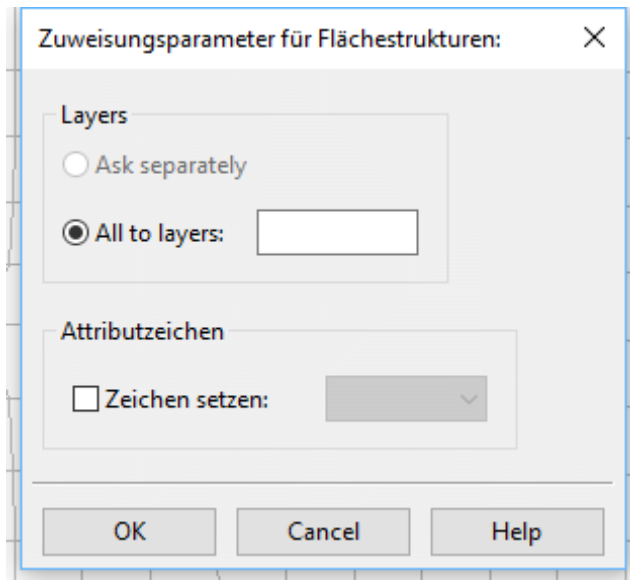
Area structures can only be assigned to single-valued data and group data. Each node or element centre lying inside or at the boundary of the area structure automatically receives a value. There are no distances defined or verifications of assignment, as there are for line or point structures.

The assigned value at the appropriate node results as follows:

- In case of group data, all nodes or elements receive the first value defined at a corner of the area structure.
- In case of single-valued data, the first three values defined at corners of the area structure are used for the definition of a skew plane. The final value depends on the position of the node / element centre in the plane. If there is only one corner with a defined value or the first three values found are equal, a constant value is assigned. For more than three values, the remaining ones are not significant for the assignment. If there are only two values, the second value is not interpreted and the first value is assigned as a constant.

Attention! The assigning of group data (e.g. BILA) always uses only one value.

Attention! The assigning of area structures to single value nodes or elements uses a maximum of three values. Thus more than three different values at the corners of the area structure are possible yet not meaningful.

The proper assignment of polygon- and group data can be examined via *Attributes* → *Special Features* → *Check group attribute respectively polygon line attribute*. If the generated attributes overlap with existing attributes of the same type of data, the affected attributes are displayed as green lines and the conflicting nodes with green circles.

4.5 Contour

The nodes and element edges are created in the mesh generation from the contours. Therefore, the distances of contour nodes in particular should be controlled.

The contour menu is only available in the model mode. The following menu items are available:
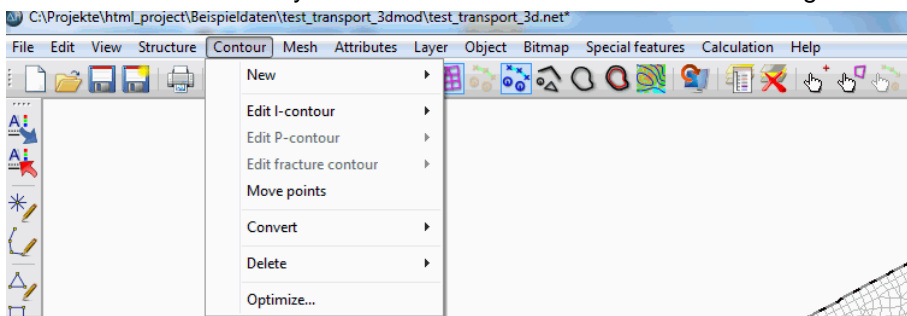
New

At first
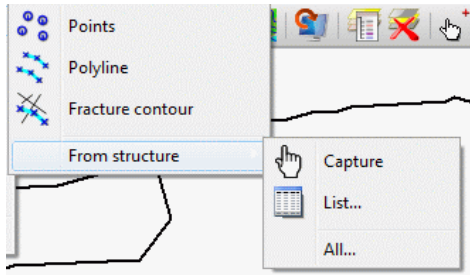
The user selects which type of contour (point, line or fracture contour) should be created. After that the contour can be created in the graphical interface. It is possible to set the capturing mode. Either the point can be captured *Freely*, a *mesh node* or a *point* can be captured.

Es kann eine vorhandene Struktur in eine Kontur umgewandelt werden, indem diese mit *Aus Struktur → Fangen,* mit *Aus Struktur → Liste* oder mit *Aus Struktur → Alle* ausgewählt wird

Should an existing structure be converted in a contour it can be selected via *From structure →Capture, From structure → List…* or *From structure → All...*The following input window appears:
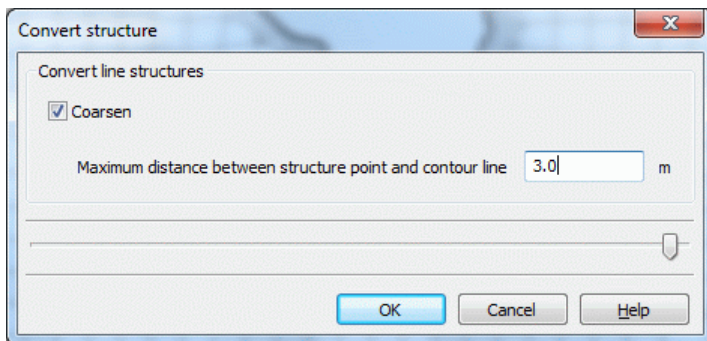
All structure point issues that have a smaller distance than the input distance to the generated contour line will not be considered (Fig. 61).
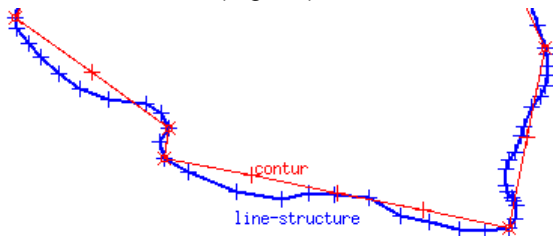
Fig. 65: Contour lines based on coarsened structure lines

The contours inherit from the structures only graphic information, i.e. coordinates of points and start and end points of lines. Afterwards, no information is available regarding whether the contour object is based on a structure object (and if so, which one), or whether it was picked directly on the screen.

Edit l-contour

The length of a line contour can be modified in different ways. The following submenu appears:
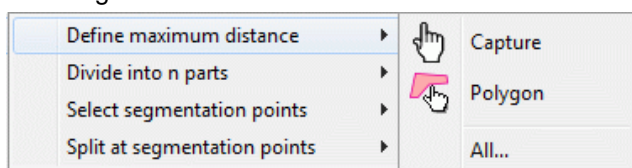
■ Define maximum distance:

After selecting this menu item, a single contour line can be captured, an area or all contour lines can be selected, in which the length of the l-contours should be modified. After pressing the right mouse button, an input window appears in which the desired length is entered. If only one contour line is selected, the actual length of it is displayed, and a new length can be defined. If you select several l-contours within a polygon, the following window appears:
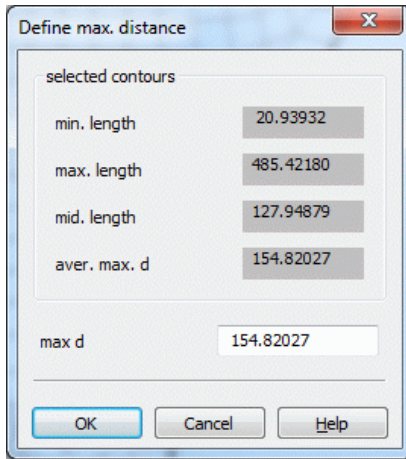
In this case the minimum, maximum and average lengths of the selected l-contours are displayed and a new length can be defined.

Attention! All existing segmentation points inside these contours are deleted!

- Divide into n parts:

After selecting this menu item, a single contour line, an area or all contour lines can be selected, in which segmentation points for the l-contours should be defined. After pressing the right mouse button, an input window appears in which the desired number is entered. If only one contour line is selected, the actual length of it is displayed, and the number of the segmentation points can be defined. If you select several l-contours within a polygon, the following window with the statistical parameters of the selected l-contours appears:
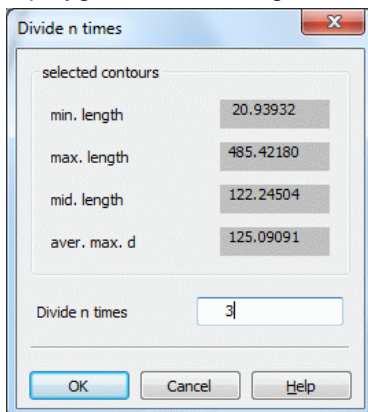
By specifying the number of the segmentation points, the l-contours are divided in equidistant parts. All existing segmentation points inside these contours and an eventually defined maximum distance are deleted.

- Select segmentation points:

After selecting an l-contour (Capture) and confirming with the right mouse button the desired segmentation points can be selected with the left mouse button and confirmed with the right mouse button.

After selecting a number of l-contours within a polygon, one l-contour is selected. The desired segmentation points are then selected with the left mouse button and confirmed with the right mouse button. The next l-contour is subsequently selected with the left mouse button.

- Disconnect at segmentation points:

If some contours were divided into segments these segments could be converted in individual contours. The desired l-contour (individually, in a polygon or all) is selected. By confirming with the right mouse button, the corresponding l-contours are disconnected at their segmentation points.

Edit p-contour

At this point, the well parameters for one or more p-contours are set:

This menu item is explained in detail in the tutorial, chapter: "Generating contours" and in chapter: "Discretization in space: Special case: Discretization of wells".

The node generation depending on the well parameters is explained in chapter: "Algorithm of the boundary node generator".

Edit fracture contours

Because this menu item is very complex it is treated in chapter: "Generate fracture contours".

Move points

After selecting this menu item a window appears in which can be chosen whether the contour points are moved to a new or to an already existing point.

Convert

The following menu item appears:
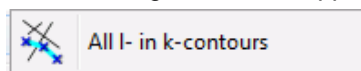
After selecting this menu item all (separated) l-contours are converted in k-contours. L-contours with segmentation points are neglected.

Delete

The type of contour is selected which should be deleted.
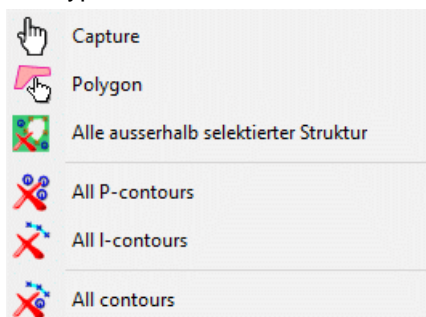
If one or more contours should be deleted, the contour can be captured or several contours are selected using a polygon. Contours outside of a selected structure can be deleted. There is also the possibility to delete *All p-contours*, *All l-contours* or *All contours*.

Optimize…

The contours have to be optimized before they can be considered for the mesh generation, i.e.:

- Contour lines must not intersect with each other.
- Contour lines must not overlay (partly or completely).
- Contour points on contour lines are only allowed when there is a segmentation point at this place.
- Contour points, start and end points of lines and segmentation points must not lie too close to each other.
- Furthermore, global node and element generation algorithms can be used only if the outer contour boundary is closed.

By using the function *Optimize…*, you can check the existing contours and optimize them. The following menu appears:
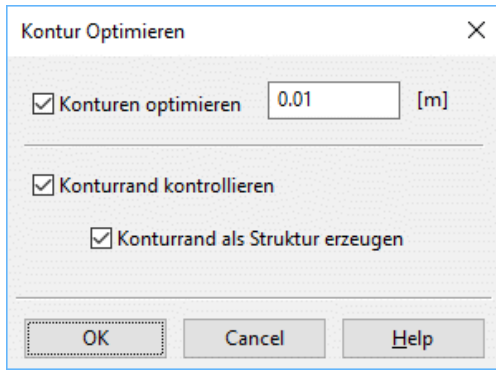
The contour points are merged if the existing distance is less than or equal to the entered value.

During the optimization the following is done:

- Intersected contour lines are divided.
- Segmentation points are generated where contour points lie on a contour line.
- Overlaying contour lines are separated and shortened.

Example of an optimization of overlapping and duplicate contours:
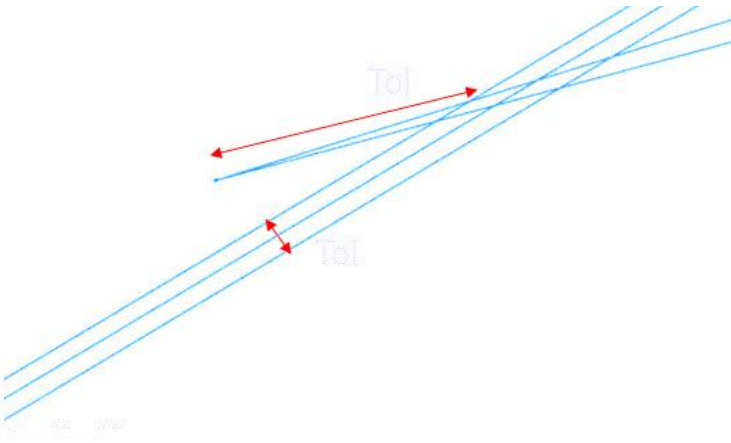
Fig. 66: Before contour optimization
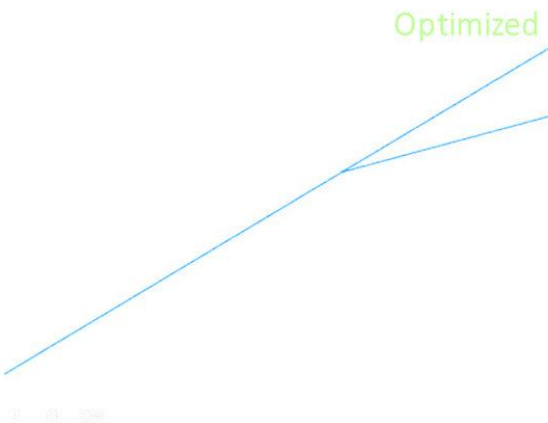
Fig. 67: After contour optimization

Afterwards, the program computes the outer contour boundary. If this is successful, in the status bar appears the note: "Contour boundary closed!" If this is not the case an error note occurs. Then the contour distance has to be edited, because mesh nodes are generated only with a closed contour boundary.

Alternatively, it is possible to control only the contour boundary or to save the contour boundary as structure.

4.5.1 Generate fracture contours

At present, SPRING can create only two-dimensional fractures interactively on-screen. 1D fractures have to be created with the help of the fracture generator "Klufti.exe". Details are available on our website www.spring.delta-h.de under "Download&Support".

Via the menu item C*ontour* → *New* → *Fracture contour* 2D fractures can be generated. The contour lines are the intersection lines of the fractures in the plane. After defining the fracture line by two points the menu item C*ontour* → *Edit k-contours* is selectable.
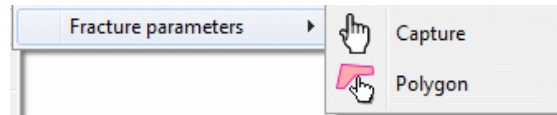
After selecting the fracture (the fractures have to be displayed in the main window) the following input window appears in which the fracture parameters are defined:
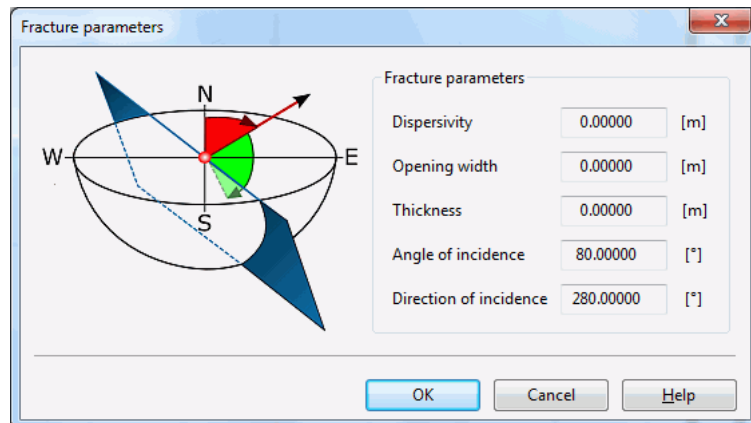
Fracture parameters

■   Dispersivity

The dispersivity of the fracture is defined.

■   Opening width

The opening width of the fracture is defined.

■   Thickness

The thickness of the fracture is defined, that affects only 1D fractures.

■   Angle of incidence

The angle of incidence describes the gradient of the fracture in space (green segment in the figure above). This is only important in 3D models.

■   Direction of incidence

The direction of incidence describes the direction of the fracture in the plane (red segment in the figure above). She is the perpendicular to the sectional area of the fracture (= course of the k-contour). This is only important in 3D models.

Generating k-contours and input of fracture parameters is possible both in a 2D and a 3D model. Transferring the k-contours as fracture elements is, however, exclusively possible in a 3D model via *Special features* → *Fractures* → *Generate 2D fractures from fracture contour*.

4.6 Mesh

The mesh menu is only available in the model mode. The following menu items are available:
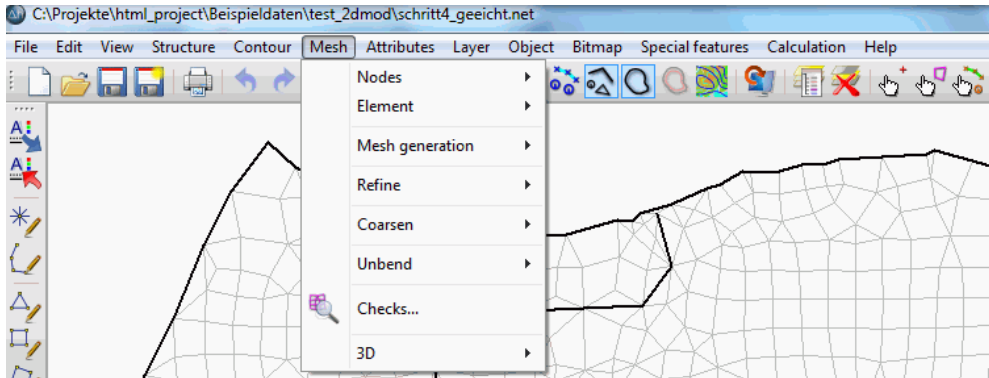
Nodes

The position of the nodes
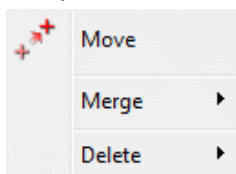
can be edited.

■   Move → nodes:

By using this function, you can change the position of an existing node directly with the help of the cursor. The node can be moved to a free coordinate (Freely) or to an already existing point (Capturing mode) by pressing the left mouse button. When moving nodes, the user should always take care of the affected elements. He has to avoid moving nodes into inadmissible elements.

By using *Edit → Undo* or selecting the button ↶ the moving is undone.

Moving the nodes, does not change the node attributes at all. Depending on the attribute, it can be necessary to assign a new value to the node afterwards. Moving a node by several metres can require a new interpolation of the ground level or the receiving water course level.

■   Merge nodes:

In many cases two nodes which are too close to each other and could be merged are not visible until elements are generated between the nodes. The second case where the function *Merge* can be quite useful is the merging of two separate models which have a shared boundary. In these cases, the function provides the following steps:

■   Both nodes are moved to their geometric centre by changing their coordinates.
■   The node attributes existing at both nodes are averaged (except inflow / outflow rates KNOT).
■   One of the nodes is deleted.
■   The incidences of the adjacent elements are corrected.

The two nodes to be merged are selected by one of the following methods:
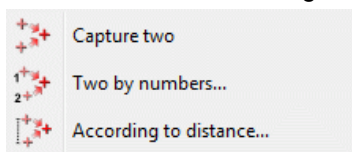
Capture two: Two nodes are chosen with the left mouse button.
Two by numbers...: An input window appears in which the two numbers are entered:
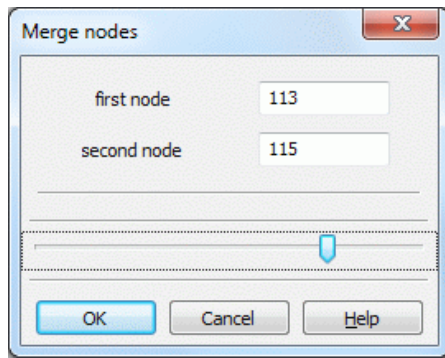
According to distance...: An input window appears in which the maximum distance in x- and y-direction is entered:
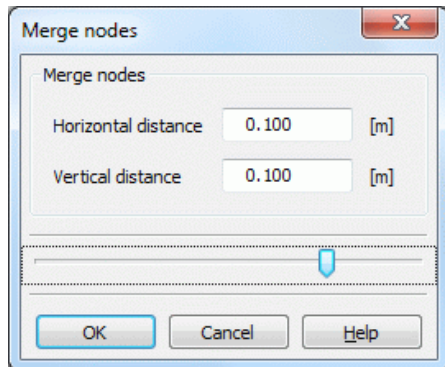
In case of a super elevated vertical model, the distance in the x-direction should be larger than the one in the y-direction. All nodes whose distances are smaller than the entered values are merged automatically by SPRING, they are coloured and displayed in the main window.

■ Delete nodes:

The following submenu appears:
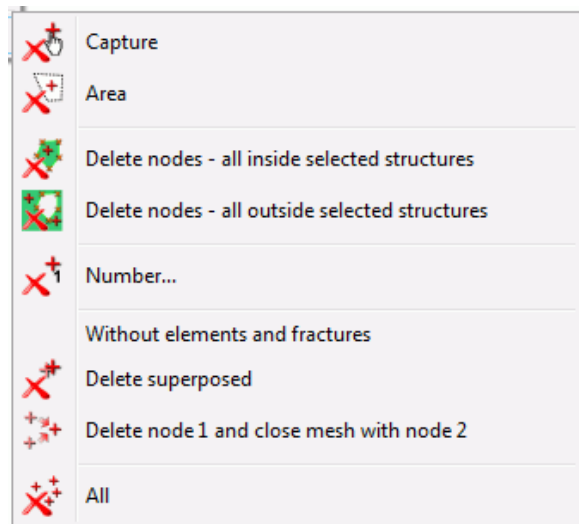
The nodes can be deleted manually (Capture), inside a polygon (Area), inside or outside selected structures or by number.

By selecting the menu item *Number…* appears an input window in which the number of the node to be deleted is entered:
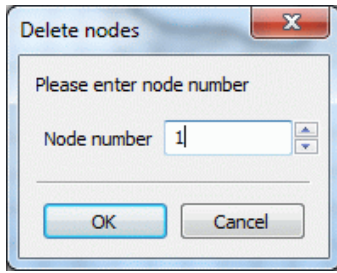
It is also possible to delete nodes which do not belong to an element or a fracture or whose coordinates are equal to another node (Delete superposed).

Attention! If a node is deleted, all adjacent elements of the node are deleted automatically.

The menu item "*Delete node 1 and close mesh with node 2*" allows the user to delete a node (node 1, by selecting it with the left mouse button) by moving it to and merging it with another node (node 2). In this process, no elements are deleted. Node 2 can be selected with the left mouse button, or, by pressing the right mouse button, the closest node is automatically selected by the program. The element and node attributes do not change.

Element

In the submenu *Mesh → Element → Delete* are different possibilities available to delete one or more elements:

By selecting the menu item *Number…* appears an input window in which the number of the element to be deleted is entered.

By selecting the menu item *With filter…* an input window appears in which an item can be deleted based on its attribute value:
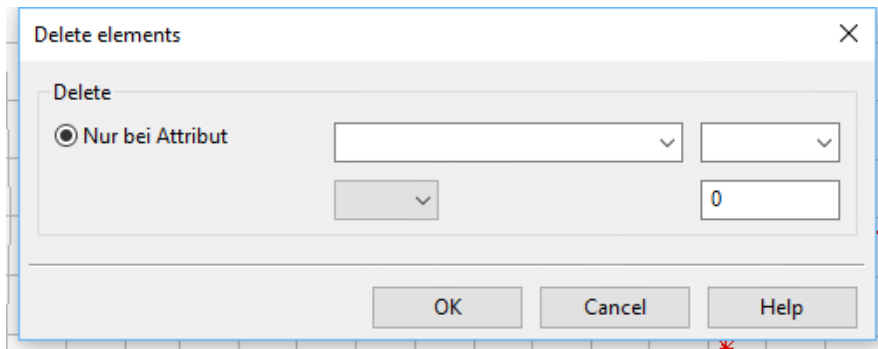
When deleting elements based on selected structures only those elements are deleted whose nodes and element center are completely inside or outside the structure or on its edge.

Note to the menu item *Element → Delete → All with wrong orientation:*

The element generation in SPRING is generally counter-clockwise. If the user has inadvertently created elements with the wrong orientation by editing the model file, he can automatically delete them here.

Via the menu item *All* all elements are deleted!

Mesh generation

Since this menu item is very complex, it is treated in a separate chapter.

Refine

The following submenu appears:
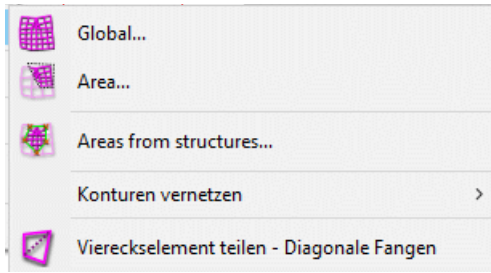
- By activating the function G*lobal…*, the following window appears:
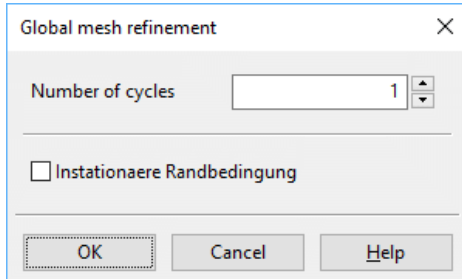
The *number of cycles* defines how many refinement steps should be carried out. Each step divides each element into four new elements by bisecting the element edges. Hence, one refinement step bisects the grid distance in the refinement region, two steps quarter the distance etc.

If a transient input file was imported prior to meshwork processing, the transient data is also refined.

- If *Area…* or *Area from structures is selected*, only the elements in these areas are refined. The procedure corresponds largely to the global refinement. In refining an area a minimum element size can also be specified. The region between the refined and the non-refined elements must also be edited. For a good transition from the refined to the non-refined region, some elements outside the specified polygon must be refined. There are two different strategies to do that:
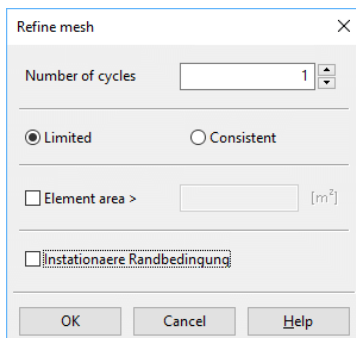
Limited Refinement:

An additional edge is only bisected automatically in case a defined length ratio exists in the adjacent element. The criteria are not as strict as for the consistent strategy. The resulting transition zone is thinner, but elements with a bad length ratio can appear.

Consistent refinement:

If the bisection of an edge leads to a bad ratio of lengths in the adjacent element, the adjacent edges are bisected automatically, too. This can lead to very short edges in some elements and to a relatively wide transition zone between the refined and the non-refined elements.

When using Area from Structures a structure has to be selected by clicking the left mouse button after confirming with OK. For this, the structures have to be displayed.

If a transient input file was imported prior to meshwork processing, the transient data is also refined areawise.

- Interlinking contours:

If new geometries (such as building outlines or new watercourses) are to be added to an existing mesh, they can be installed automatically with this menu item. Only the corresponding contour must be created in advance. After selecting from
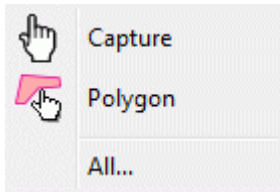
the contours to be inserted can be selected.

- Divide quadrangle element

By using the function "capture diagonal", you can divide a quadrangular element along its diagonal into two triangles. The corresponding corner nodes are captured with the left mouse button. The new elements get the attributes of the former element.

Coarsen

The coarsening can take place globally, by area or individually:
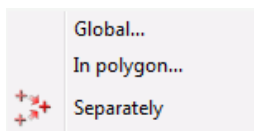
With a global coarsening, another window appears, in which the corresponding criteria are selected:
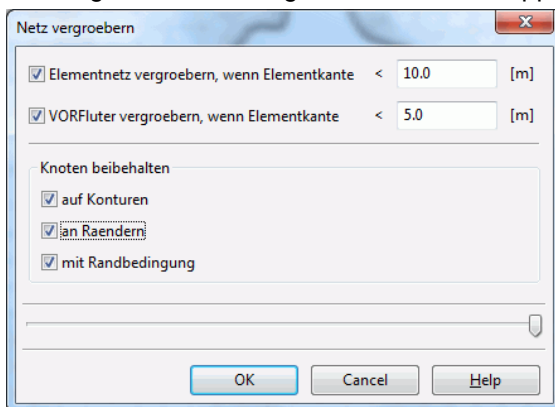
Unbend

Unbending the mesh can be done globally or regionally:
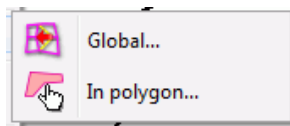
When using various grid-generation algorithms, regions with different grid distances and manually-generated regions, the transition between the corresponding meshes can be abrupt. By using this function, you can smooth these transient regions globally or in a polygon (selecting the desired area with the left mouse button). The algorithm automatically moves nodes to achieve the best mesh geometry. You must define how many times the algorithm should smooth the mesh (see also: Unbending the mesh):
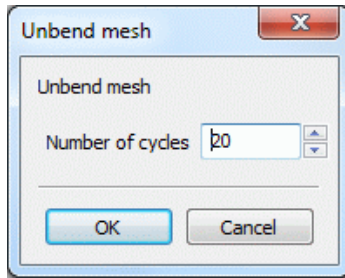
The larger the *number of cycles* chosen the larger is the influence of the smoothing algorithm in the transient regions into the regions with generated grids. Furthermore, the algorithm moves the nodes around wells (contour points) which were generated logarithmically from well parameters.

Nodes and element edges which lie on or along contours are not moved.

Checks…

The mesh generation should be completed with several geometry checks before any data are assigned to the mesh. Errors in the mesh geometry can lead to the finite element calculation not being able to start. By activating this function, the following window appears where the different algorithms can be selected:
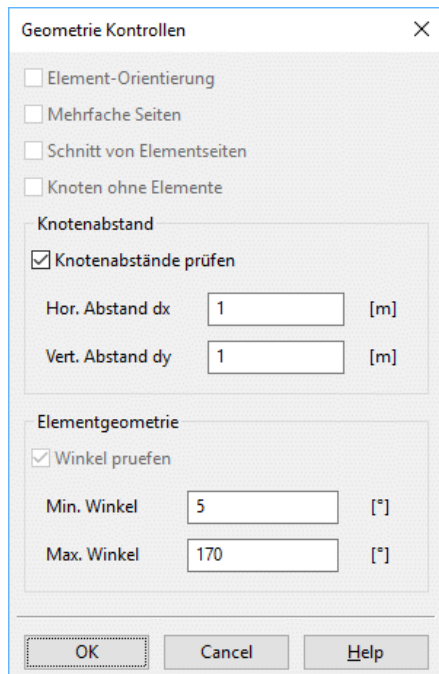
The algorithm *Element orientation* checks whether the nodes in each element are oriented anti-clockwise. Elements with a clockwise node list (resulting from manually picked elements) have to be deleted and selected again counter-clockwise.

The algorithm *Manifold edges* checks whether any element edge exists too often: element edges exist at most twice. At the mesh boundary, element edges exist once, while in the interior, element edges exist twice yet with different directions. Multiple edges are caused by multiple elements or wrongly-oriented elements.

The algorithm *Intersection of element edges* checks whether any element edge intersects with another one. (For a large mesh, this can take some time.)

The algorithm *Nodes without elements* searches for nodes which do not belong to any element.

The four control mechanisms mentioned are executed automatically and cannot be selected.

The algorithm *Node distance* helps to find nodes existing twice or nodes very close to each other. Nodes where the distance of the x-coordinates is smaller than DX or rather the distance of y-coordinates is smaller than DY produce an error.

The control of the element geometry helps to detect elements with poor quality (with very small or large angles). First, a corresponding error message appears:
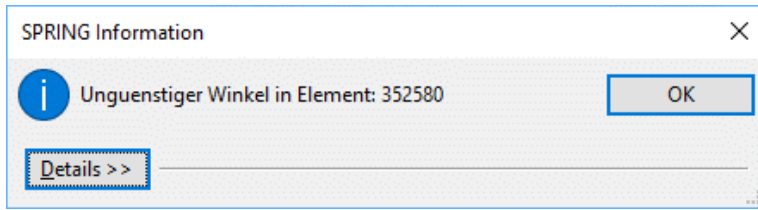
The minimum (attribute AGMN) and maximum (attribute AGMX) angles are stored as attributes AGMN and AGMX and can be displayed after the mesh control.
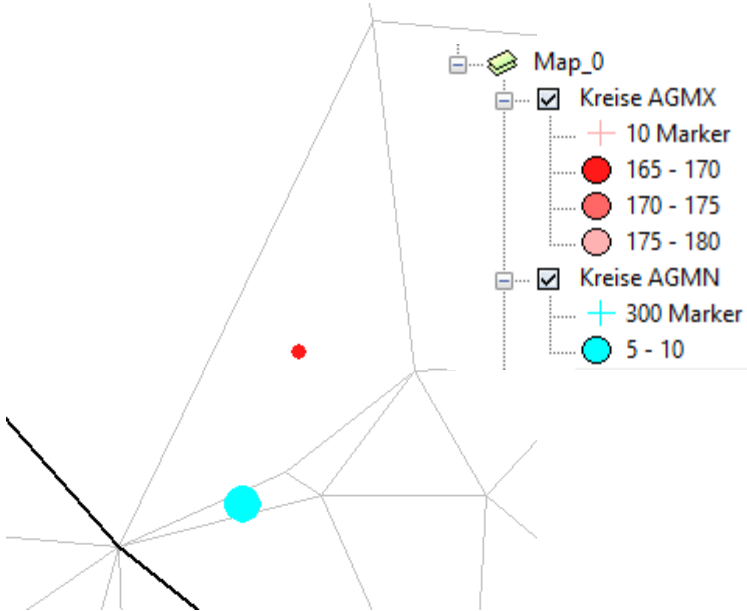
Fig. 68: Control of the element geometry

3D

Since this menu item is very complex, it is treated in a separate chapter.

4.6.1 Mesh generation

In this chapter all menu items and processing steps are briefly described. Then the algorithms of node generation are described in detail. The following menu items are available
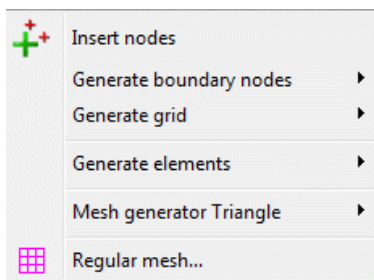
Insert nodes

At any point nodes can be created with the left mouse button.

Generate boundary nodes

The algorithm is described in chapter: Algorithm of the boundary node generator.

In the first step of each mesh generation you should generate nodes at all contour nodes and perpendicular to the contour lines. This can only be done if a closed contour boundary exists. Hence, the function *Generate boundary nodes* (*Global... or Area*) starts with a computation of the contour boundary. Only for the case that the contour boundary has been computed successfully does the following menu appear. In this menu, you must define the number of generation steps (n):
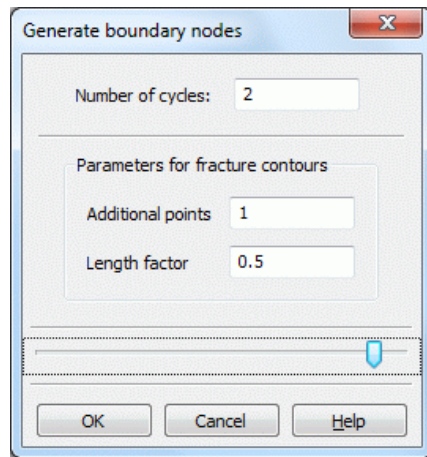
For projects without any fracture contours only the *Number of cycles* is required. Inserting n=0 means the algorithm passes only step 0 of the boundary node generation and generates nodes directly at contour nodes. Inserting n >0 means the algorithm generates nodes directly at the contour nodes plus n nodes perpendicular to the contour line.

For projects containing fracture contours two further parameters are required. The *Length factor* is used to decrease the distance between generated nodes along fractures. The number of *Additional points* controls the discretization perpendicular to the fracture. The boundary node generation algorithm generates the nodes perpendicular to the fracture with a distance which is exponentially increased in each step.

Area…:

Is the computation of the contour boundary successfully, the area where the boundary nodes are to be generated can be defined. It should be ensured that all existing nodes are captured in the definition of polygon for the node generation.

Notes:

It is recommended to start the boundary node generator at least with the 0th. initialization step because problems in the mesh generation occur if there are no nodes on contour nodes. Moreover, depending on the model 1-2 cycles of boundary node generation are recommended.

If in an area a large number of generation cycles is necessary the following approach is recommended:

First, the boundary node generator performs a small, global reasonable number of generation steps. In a second step, all nodes in the area where more generation steps should be performed are deleted.

Thereafter, the boundary node generator performs a greater number of generation steps. In this case, no new mesh nodes are generated from contour points that lie outside the deleted region. That is, even the 0th step of the boundary node generator fails because of the termination criterion "mesh node in the vicinity already exists" and thus no other generation steps are carried out from these points.

Generate grid

The algorithm is described in chapter: Algorithm of the grid node generator.

A global grid generation presumes that a closed contour boundary exists. Hence, the function *Generate grid* (Global.../area) starts with a computation of the boundary contour.

If the grid should be generated in an area this area has initially to be defined. In defining the polygon for the generation edge, make sure that all the existing nodes are included.

Only for the case that the boundary contour is computed successfully does the following menu appears:
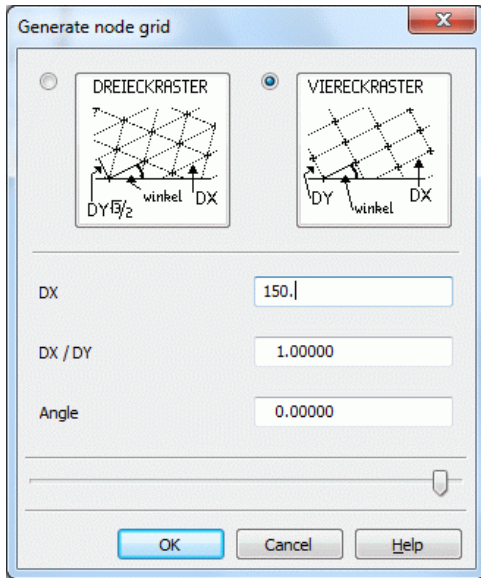
The nodes are generated so that the resulting triangles or quadrangles are as close to equilateral as possible.

The default type is a rectangular grid, a triangular grid requires due to the higher number of elements only a higher storage and computational effort.

The default value DX is obtained from the average length of the contour lines along the outer contour boundary (without considering the intermediate nodes) scaled with 0.3.

The default ratio DX/DY is obtained by the ratio of scales in the directions x and y defined for the project: Usually, the scales in the direction x and y are identical for 2D horizontal and 3D models. Then the default ratio DX/DY=1 leads to the generally favoured regular grids. In case of vertical models, the scale in the y-direction is often super elevated compared to that in the x-direction. Then the resulting ratio DX/DY>1 leads to the generally favourable rectangular grid with long horizontal and short vertical edges.

By entering an angle the element edges are rotated against the horizontal.

By confirming the menu with OK the specified grid is generated. The point of origin of the grid is assigned to the minimum x- and y-coordinates of the boundary contour.

Using this menu is described in detail in the tutorial: Step 2, generating nodes.

Generate elements

With SPRING the elements of an FE mesh can be created in different ways. Models can be created only with triangular elements or they can be created with triangular and quadrangular elements.

Therefore, the menu item *Generate elements* has the following submenu:

- ■ Generating *Triangle, Triangles and quadrangles* (global/region/from inner boundary):

After inspection of the closed contour boundary and any definition of the region/inner boundary, the Triangulation starts.

- ■ All existing elements which were completely inside the specified boundary are deleted, including their attributes.
- ■ All nodes inside the specified boundary are triangulated considering all contour data completely inside the polygon.
- ■ For the generation of triangles and quadrangles, triangles which are suitable for this purpose are combined to quadrilaterals after the triangulation.
- ■ The menu items *Triangle and quadrangles* i.e. *Triangles (region)* are to be used for the manual generation of elements as well.

The mesh generator uses the polygon segments as element edges! The triangulation takes more time if there are more nodes and contours which need to be considered

■   Quadrangle generator:

By using this function you can combine suitable triangles to quadrangles afterwards. Please notice that by doing this, the attributes of one of the triangles will be lost.

Mesh generator Triangle…

Using the mesh generator Triangle is described in detail in chapter: "Using the external mesh generator Triangle" (p. 195).

Regular mesh…

Generating a regular mesh is described in detail in chapter "Compilation of a 2D model: Generating an FE-mesh with automatic mesh generation" (p. 186).

4.6.1.1   Algorithm of the boundary node generator

Inside the polygon defined by the closed boundary contour the nodes are generated by the following principle:

0th step:

Step 0 generates one node at each of the contour points, the start and end points of contour lines and the division points between them.

The following steps generate nodes along contour lines:

■   At segmentation points of contour lines the algorithm produces a node normal to the line.

■   At start and end points of contour lines without adjacent lines the nodes are generated at 90, 150, 210 and 270 degrees to the contour line.

■   At start and end points of contour lines which are continued by another contour line the algorithm generates nodes depending on the angle between the two lines. For an angle between 90 and 180 degrees, the nodes are generated along the centre of the angle. If the angle is larger than 270 degrees, the algorithm produces nodes normal to each of the contour lines.

The distance between the contour line and the new node is calculated from the average distance "dist" between the corresponding node and its two adjacent nodes on the contour line:

For the case of standard contour lines (l-contour) step n of the boundary node generator generates nodes with at a distance of n*dist from the contour line (see figure).
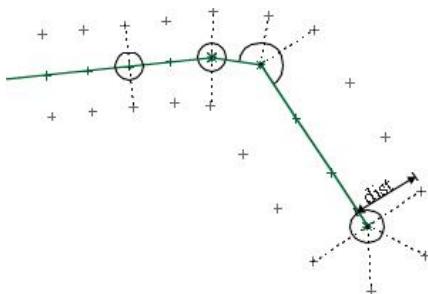
Fig. 69: Boundary node generator for l-contours (step 1)

For the case of contour lines which represent fractures (k-contour lines) step n generates one node at a distance of n*fakt*dist from the contour line. The scaling factor fakt (length factor) is an individual parameter and can be defined by the user. Finally, the algorithm automatically generates nodes between the node on the contour line and the new one. The distances between them increase exponentially (see figure).
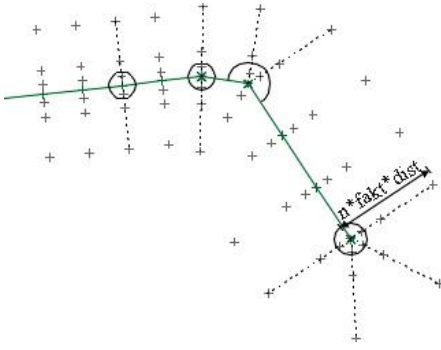
Fig. 70: Boundary node generator for k-contours (step 2, fakt = 0,8)

Further steps generate nodes at 45, 90, 135, 180, 225, 270 and 315 degree around contour points which represent wells with the corresponding parameters (R_i, R_a and C>0). The distance is calculated from $R\_n = R_i * C^n$.

Starting with an initial, final or division node, the node-generation algorithm continues to generate nodes until one of the following criteria is fulfilled:

The distance reaches the defined limit

■   n*dist in the case of contour lines (l-contour lines),

■   n*fakt*dist in the case of contour fracture lines (k-contour lines) and

■   maximum outer radius R_a in the case of contour nodes with parameters for wells, or

■   there is already a mesh node near the node to be generated, or

■   the distance between another contour line and the node to be generated is too short, or

■   the node to be generated would be out of the model area. Into this direction the algorithm will not generate any further nodes.

4.6.1.2 Algorithm of the grid node generator

The grid node generator requires a closed polygon to generate the grid inside. Hence, the global grid node generator requires the contour line which marks the model boundary to be closed. In the case of the local grid node generator, the user must choose the polygon.

The algorithm starts with the contour data inside the polygon. It generates mesh nodes at contour nodes and nodes corresponding to defined well parameters as well as at the start, end and division points of contour lines.

Depending on the grid type (triangular or quadrangular grid), the distances DX and DY and the angle the nodes are finally generated. If there is an already-existing node nearby, the grid node is not generated (see Fig. 64).
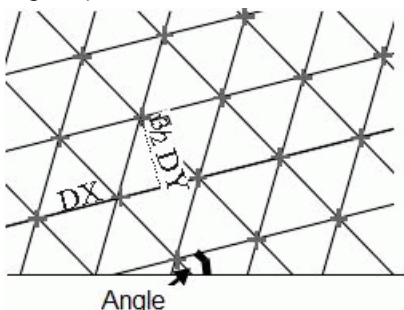
Fig. 71: Triangular mesh based on a triangular grid

A triangular grid where DX = DY gives equilateral triangles with a lateral length of DX. The defined angle can be found between the triangles and the axis x. If DX is not equal DY, the generated nodes create isosceles triangles with a base length DX and a height DY*sqrt(3)/2. Again you will find the angle between the base line and the axis x (see Fig. 65).
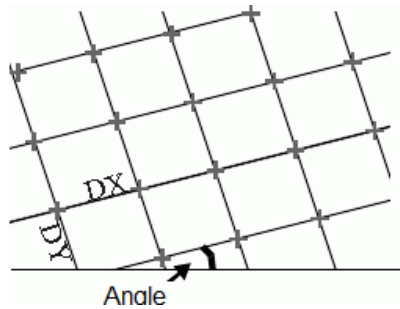
Fig. 72: Quadrangular mesh based on a quadrangular grid

The quadrangular grid finally gives rectangular elements with the width DX and the height DY. The defined angle lies between the base line and the axis x.

4.6.1.3   Triangulation in SPRING

Having completed the node generation in the entire model area or parts thereof, you can start generating triangular elements. The element generation algorithm uses the advancing front technique.

The global mesh generator starts at the outer contour boundary. It generates triangles until each of the nodes is linked to the mesh. Each element is optimized following the criteria for the angles of an element.

The local mesh generation does not require any contour boundary. The generation area is defined by selecting the outer boundary manually where finally the mesh generation starts. Furthermore, the local mesh generation uses the same algorithm as the global generation.

The triangulation algorithm automatically takes the contour data into account: An element is not allowed to intersect with an existing contour line!

The triangulation algorithm can only be used on condition that

- each contour point, each start, end and division point of a contour line is a node of the mesh,
- there are no mesh nodes on contour lines which are no division points of the contour line,
- the contours are optimized,
- no double nodes (two or more nodes with the same coordinates) exist.

SPRING includes a mesh-optimization code, with several algorithms for the boundary and the grid nodes. You should run the optimization after the contours have been completed. It will indicate the errors, a short description and the corresponding node number. Errors incurred during the triangulation can be caused by:

- manually selected nodes (double nodes, nodes on contours without contour points ...),
- additionally inserted nodes (contour points without nodes, contours without optimization, ...)
- nodes which have been moved (double nodes, contour points without the nodes) or similar operations.

You can find the errors and remove them with the help of one of the following algorithms:

- optimizing the contours
- starting the boundary node generator with 0 steps (generates mesh nodes at contour nodes)
- controlling the distances between the nodes (*Mesh → Checks…*)

The mesh generation with triangles and quadrangles always starts with a triangulation algorithm which produces only triangles. Subsequently, neighbouring triangles are combined into quadrangles. This is necessary to reduce the number of elements and thus the required computing time and memory space.

4.6.1.4   Unbending the mesh

The resulting mesh may still contain areas with "bad" elements because of the local generation strategy. Acute angles or poor aspect ratios in individual elements can be observed. The mesh can be improved with the help of the smoothing algorithm (*Mesh → Unbend…*).

In each smoothing step, the program determines the coordinates of a node as the average of n adjoining nodes, according to:

$$x = \frac{1}{n}\sum_{k=1}^{n} x_k \quad \text{and} \quad y = \frac{1}{n}\sum_{k=1}^{n} y_k$$

The node moves to the new coordinates on condition that all elements fulfill the geometric requirements for the finite element mesh. Exceptions are:

- nodes on contours,
- nodes on the mesh or 3D boundary
- nodes with markings (identification MARK).

None of these node types are moved.

4.6.2 3D

With the menu 3D settings are made for a 3D model. The following menu items are available:

3D-Boundary	▶
Check nodes and elements	
New layer arrangement...	
Divide layer...	
Delete layer...	
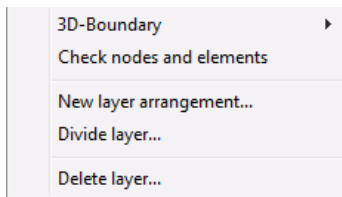
3D-Boundary

For 3D models, a 3D edge has to be set. For this are different ways available. In the submenu appears, therefore, the following selection:

| Mesh boundary = 3D-boundary |
| Generate |
| Add |
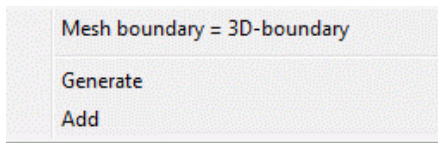
- Mesh boundary = 3D-Boundary

By selecting this menu item the mesh boundary will be automatically the 3D-boundary so that a complete 3D model can be created.

- Generate

By selecting this menu item, a 3D-boundary within an existing model can be generated. An already existing 3D-boundary is deleted! It is important to ensure that the 3D-boundary is defined counter-clockwise along the element edges! With this, a coupled 2D/3D model can be created.

- Add

By selecting this menu item an additional 3D-boundary is generated within a model in addition to an already existing 3D-boundary. It is important to ensure that the 3D-boundary is defined counter-clockwise along the element edges! With this, a coupled 2D/3D with several parts can be created.

Check nodes and elements

SPRING does not check the 3D boundary (attribute 3DRA) while loading the input data. Hence, there is no control whether the model is a complete 3D model or only a 2D model with a partial 3D model. This is why in SPRING it is always possible (but not always meaningful) to assign nodal or elemental attributes also for deeper layers in the 2D area of a 2D / 3D model. Activating the button *Check nodes and elements* forces SPRING to check for all nodes, whether they lie inside or directly at the 3D boundary or outside the 3D boundary -- depending on the actual mesh geometry and the actual 3D boundary. All elements are checked, irrespective of whether they lie inside the 3D boundary, across the 3D boundary, or outside the 3D boundary.

If all elements and nodes lie inside or directly at the 3D boundary, SPRING reports a complete 3D model in the status bar.

If some elements or nodes lie outside the 3D boundary, SPRING reports a 2D/3D model.

If some elements cross the 3D boundary, SPRING reports an error. It is recommended to represent the 3D boundary graphically (*View → 3D-boundary*). Then you can find the elements crossing the boundary on the screen and change the boundary in the corresponding regions

New layer arrangement

In chapter: „Creating a 3D model - Creating an entire 3D model" this menu item and the corresponding input windows are described in detail.

Divide layer...

In chapter: „Creating a 3D model – Creating partial layers in a 3D model" this menu item and the corresponding input windows are described in detail.

Delete layer...

In the following dialog the number of the layer to be deleted is entered:
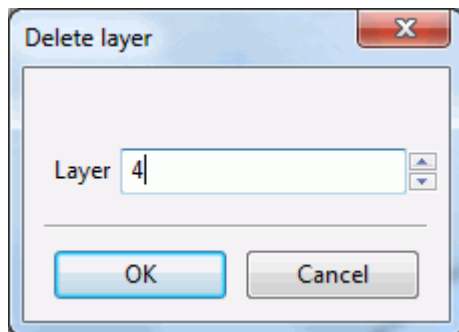
All data of the layer 4 and the associated Z-coordinates ZKOR 4 are deleted.

4.7 Attributes

The attribute menu is only available in the model mode. The following menu items are available:
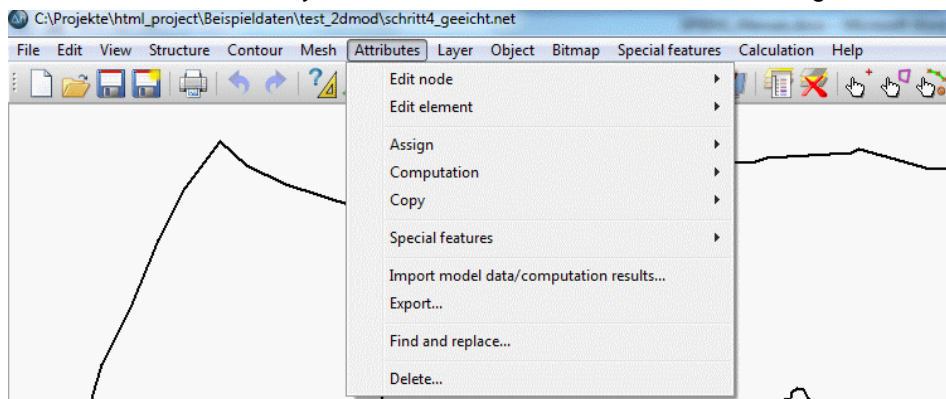
Edit node

The node to be edited can be selected either by its *number* or by *capturing* it or by using the button of the toolbar. The selected node is then marked and the input window where the attributes of the node can be edited appears on the screen:
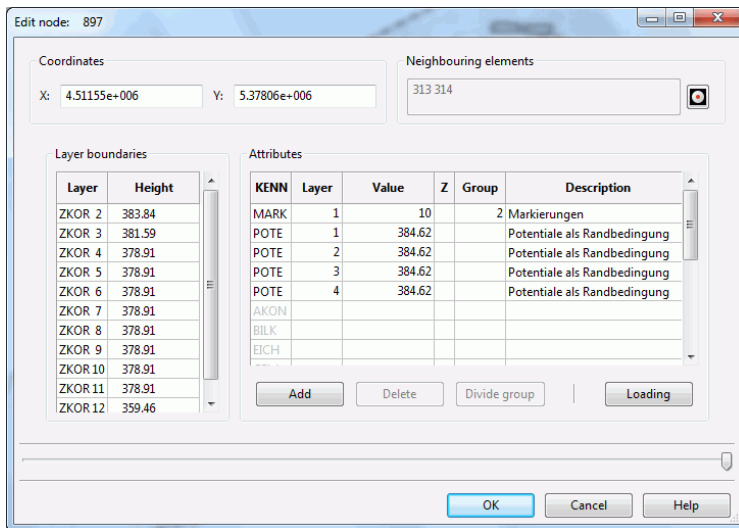
■ Coordinates:

The text boxes containing the *x- and y-coordinates* of the present node make it possible to edit the coordinates directly. But be careful, and do not move the node too far from its original position. Otherwise, the elements to which the node belongs to could be distorted. The attributes assigned to the node will stay the same regardless of whether the coordinates are changed.

■ Neighbouring elements:

The box *Neighbouring elements* contains the numbers of the contiguous elements of the node. By activating

the button ![icon] zooms the representation to the selected node.

■ Layer boundaries:

Here, the geometric data are displayed, e.g. the ground level as the upper layer boundary or in a 3D model, the heights of the Z-coordinates.

■ Attributes:

The attributes of the node can be added (left button), deleted (right button) or changed by manually editing the cells of the table (value, group or attribute sign (Z)). After opening the model, the node attributes are not loaded yet. At first, the attribute to be edited has to be marked. Then the button "Loading" is activated. After pressing the button the corresponding node values are read in from the model file and can be edited.

First, all in the model file existing node attributes are shown. Is a selected attribute at that node non-existent, the entry disappears after clicking the button "Loading".

■ Divide group:

If the node has got a group attribute like LERA or MARK and the attribute is selected in the dialog the button "Divide group" is activated. After pressing the button the attribute group is split at the subsequent node. The following node is derived from the original allocation direction. A new group is then created automatically for the following nodes.

If the window is closed with *OK,* all modifications of the node data are stored. The modifications can be undone with *Undo* ↶ .

If the window is closed with *Cancel,* all modifications are ineffective: the node data remain as they were before opening the input window.

Edit element

The element to be edited can be selected either by its *number* or by *capturing* it or by using the button ![icon] of the toolbar. The selected element is then marked and the input window where the attributes of the element can be edited appears on the screen:

- **Corner nodes:**

The box *Corner nodes* contains buttons for the nodes associated with the element. By activating the button

 zooms the representation to the selected element.

- **Attributes:**

The attributes of the element can be added (left button), deleted (right button) or changed by manually editing the cells of the table (value, group or attribute sign (Z)). After opening the model, the element attributes are not loaded yet. At first, the attribute to be edited has to be marked. Then the button "Loading" is activated. After pressing the button the corresponding element values are read in from the model file and can be edited.

First, all in the model file existing element attributes are shown. Is a selected attribute at that element non-existent, the entry disappears after clicking the button "Loading".

If the window is closed with *OK,* all modifications of the element data are stored. The modifications can be undone with *Undo* .

If the window is closed with *Cancel,* all modifications are ineffective: the element data remain as they were before opening the input window.

Assign

Since this menu item is very complex, it is treated in a separate chapter (p. 147).

Computation

Since this menu item is very complex, it is treated in a separate chapter (p. 149).

Copy

Since this menu item is very complex, it is treated in a separate chapter (p. 156).

Special features

The following submenu appears:
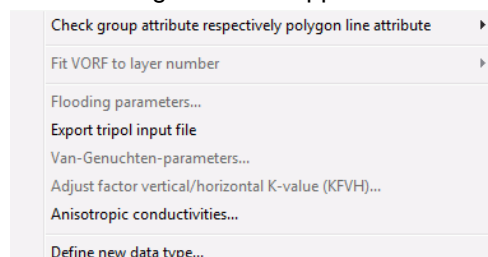
- Check group attribute respectively polygon line attribute

By using this menu item can be checked if existing attributes (single or all) with group or polygon data overlap with similar attributes.

Input window *Single attribute*:
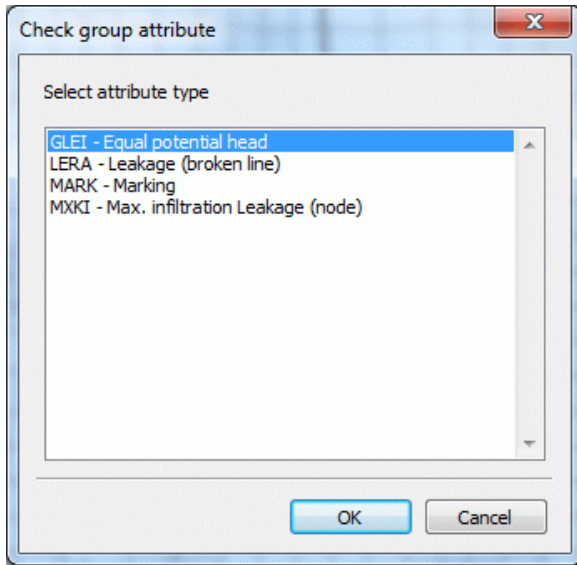
As a result, the overlapping group or polygon data are shown as green lines. The nodes on which attributes are defined twice, are marked by a green circle.

Via the menu item *Attributes → Edit node → Capture* these nodes are selected and in the appearing input window the corresponding groups are deletd.

■   Fit VORF to layer number

This menu item is only available in a 3D model in which the attribute VORF is assigned. You can choose between two possibilities: by equal potentials or by moving.

By equal potentials:

This procedure is perfect for discharge system data sets which describe a discharge system near the surface. The following query appears:
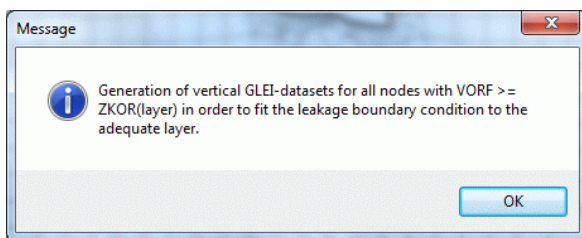
If a 3D-modell is discretized relative fine in vertical direction the discretization is not sufficient in most cases for a discretization of the correct terrain decrease at a discharge system. Here the mesh nodes on the discharge system in the top layer normally hang in the air that is above the water level (see the lower figure). In this case it makes sense that nodes beyond the level are set equal so that the leakage boundary conditions reach at the correct position.

For this it is simplest to assign the attribute VORF and the corresponded leakage coefficients (LERA or LEKN) on the first layer. By activating the function *Fit VORF to layer number → by equal potentials* a vertical GLEI-data set will be generated from the among each other laying nodes where the input for the discharge system potential (VORF) is greater than or equal to the Z-coordinate of the node.
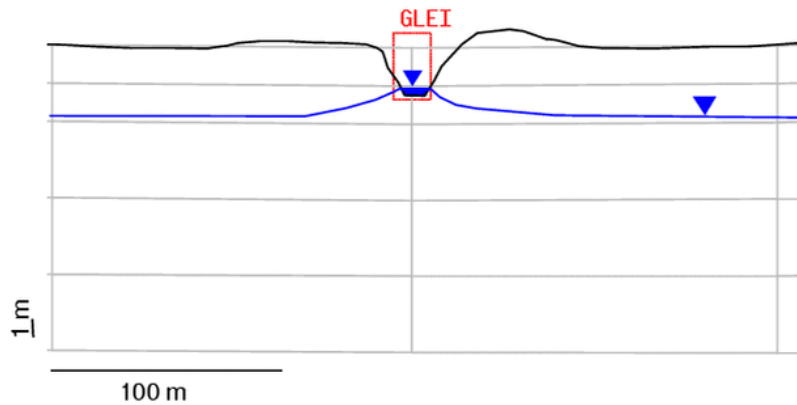
Fig. 73: GLEI-nodes at a discharge system (vertical section)

By Moving:

This procedure is perfect for discharge system data sets which describe a leakage boundary condition "inside" (e.g. ducts):
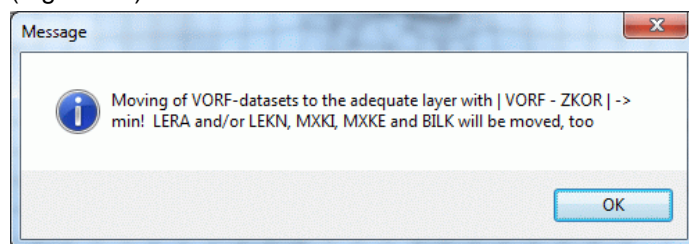
With this procedure the Z-coordinate is determined where the distance to the VORF value is at minimum. By activating this function all data types of a VORF-node which correlate with the discharge system node boundary condition (VORF, LERA, LEKN, MXKI, MXKE and BILK) are removed to the corresponding adequate layer!
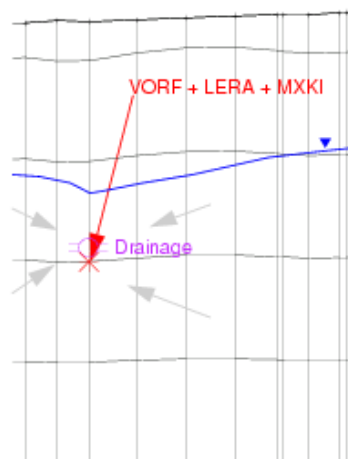
Fig. 74: Determination of the "correct" position of the leakage data

■   Flooding parameters…

If the attribute GRUB (balance nodes for mine flooding) is assigned, the water level and the additional amount of water for the individual mines are defined in the following input window. A detailed description can be found in the "How To – Simulation of flooding processes" (p. 436).
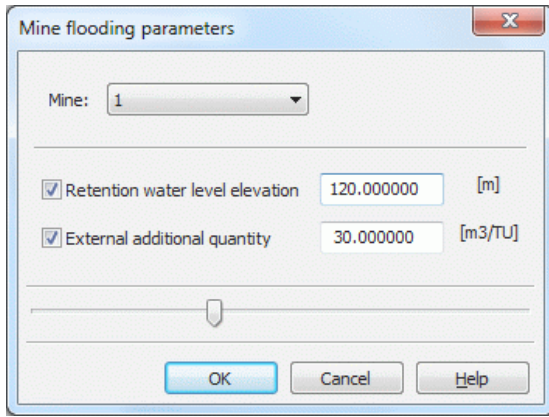
■   Van-Genuchten-parameters…

If the attribute USAT is assigned in the model file, the parameters of the individual saturation regions are defined using this menu item:
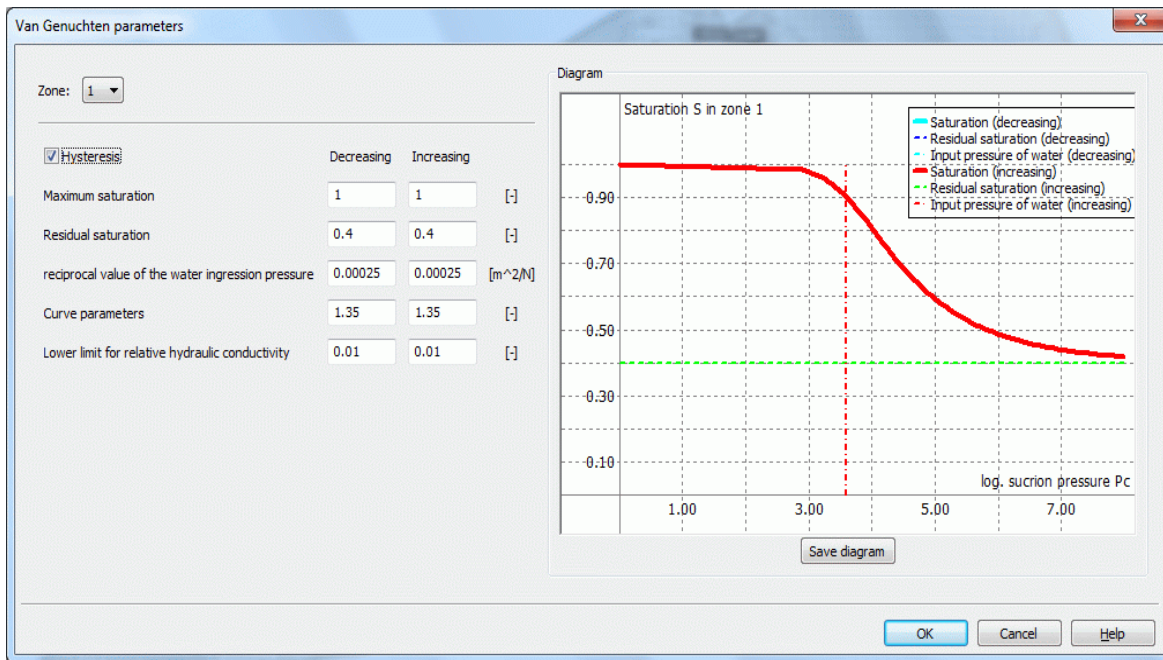
This menu item and the corresponding input window are described in chapter: „Data structure of the groundwater model – Saturation parameters, Hysteresis" in detail.

■   Adjust factor vertical/horizontal K-value …

The following input window appears:
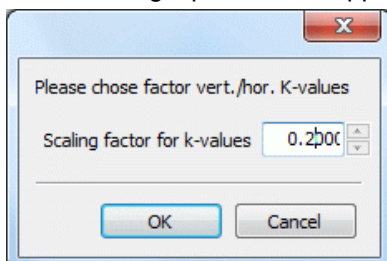
By using this function you can change the ratio between the horizontal and the vertical permeability used in 3D models. The default value is 0.1.

■   Anisotropic conductivities …

Since this menu item is very complex, it is treated in a separate chapter.

■ Define new data type

The following input window appears:
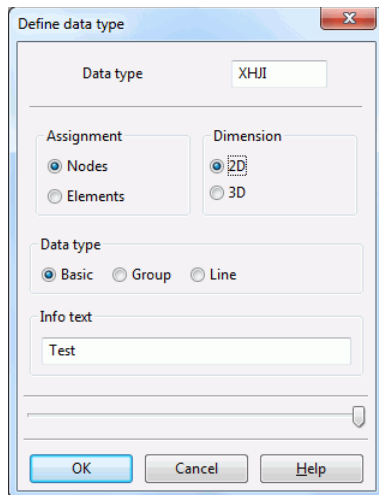
Here new data types can be defined, which are not registered in the file *xsusi.kenn*. Details can be found in the notes in the chapter "Information of the file xsusi.kenn" (p. 454).

Import data/computation results…

By using this function you can import model data from other directories or calculation results from the different SPRING modules. First, a file selection box appears:
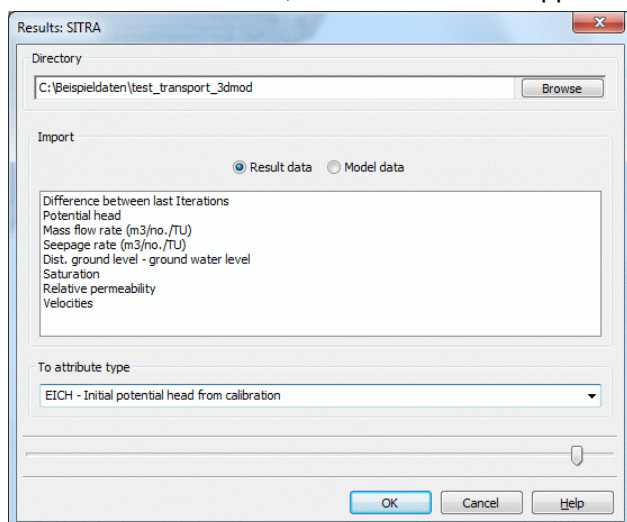
You must specify the directory where the calculation was carried out or in which the model data is found. (SPRING does not check whether the data existing in the binary files fit to the current project.)

The box *Import* contains a list of result data types where you can select one. The box *To attribute type* defines the identifier where the imported data in the current model are assigned to. For all data types except the velocities, you can define any ID-code, even identifiers unknown to SPRING. An acceptable ID is a combination of 4 letters or numbers (underscore is also allowed).

Depending on the selected result data type, SPRING automatically knows whether to save the data as nodal or elemental data. For the case of velocities, it is not possible to define an individual identifier, because the values are automatically assigned to the elemental data VV_X, VV_Y and eventually VV_Z.

By confirming the menu with OK, the selected results are imported and assigned to the corresponding nodes/elements in the appropriate layer of the current project.

Export…

By using this function, you can create an ASCII file containing formatted data directly in SPRING. The available format is the STR-format (I6, F10, F10, F10).

The following window appears:
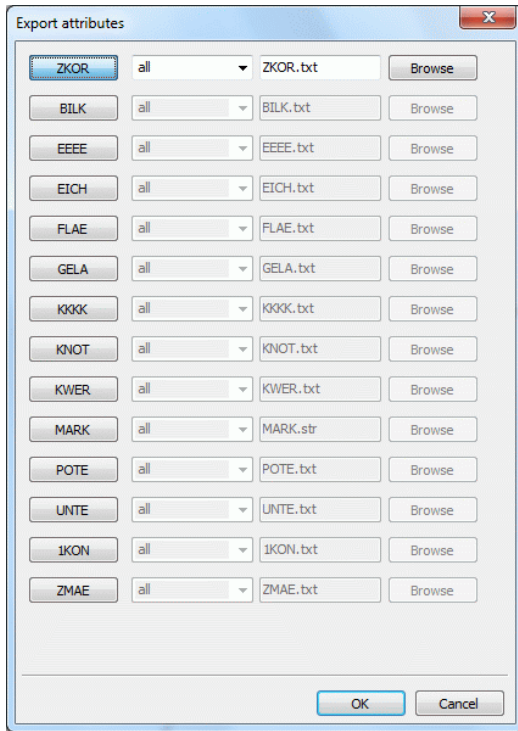
By clicking on one or more buttons the attributes to be exported are selected. Subsequently, the desired layers in a 3D model and the name of the output file are specified. With the button *Browse* a different directory in which the data should be stored can be selected. By clicking on the OK button the export starts.

Find and replace...

Using this menu item, attribute values or attribute characters in one or all layers (3D model) are automatically replaced in the model file. A possibly existing attribute character will not be automatically replaced with, but must be specified explicitly.

The following input window appears:
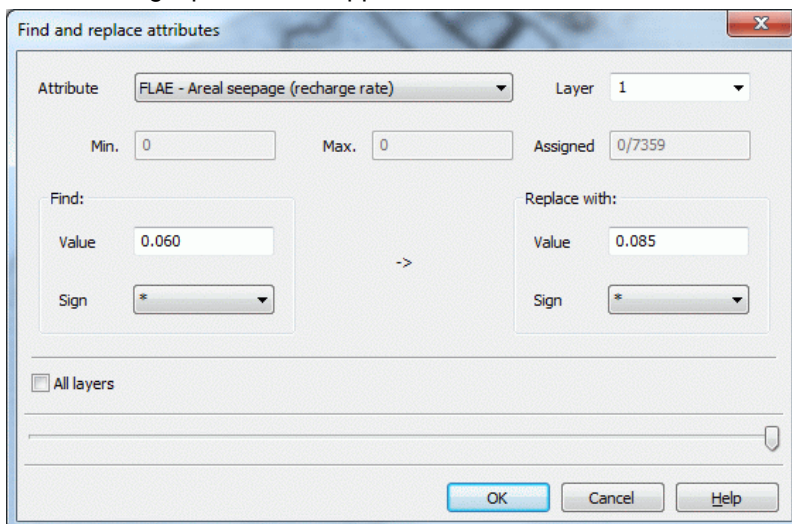
For orientation, the minimum, maximum and the number of assigned values are given. With the checkbox "All layers" values can be simultaneously found and replaced in all layers.

Delete…

By using this function existing attributes can be deleted. It is possible to delete either a single attribute or several attributes simultaneously:
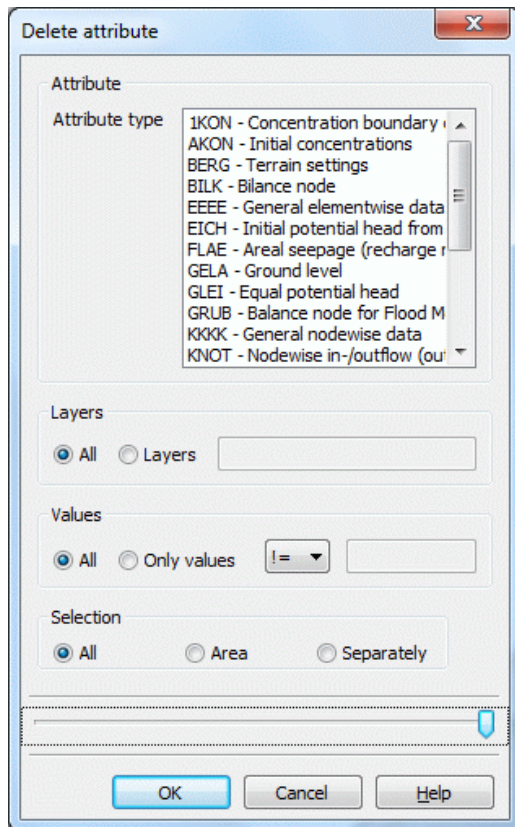
In case of a 3D model you can delete the data either in all layers or only in one single layer.

You can decide whether all values or only a certain value, in the whole model (all) or only in a certain area defined by a polygon or separately should be deleted. The different options can be combined any way you want (e.g. "only POTE in layer 3 in a certain region" or "only LERA in a certain region with the value 200.0" ...).

Only the values of nodes or elements are deleted that are entirely within the selected polygon.

4.7.1 Assign attributes

The following submenu appears:
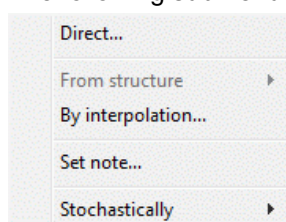
Direct…

With this input window:
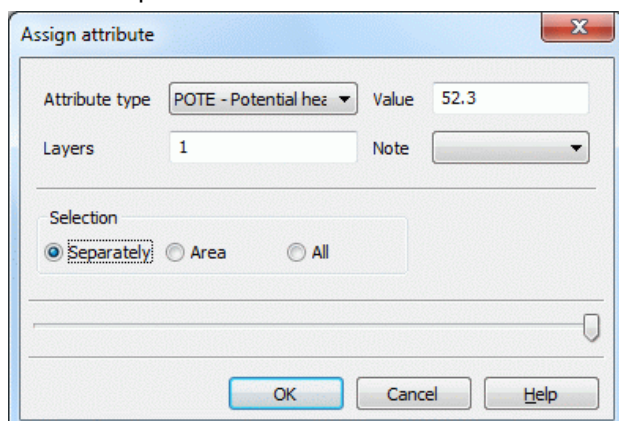
to all or single nodes or elements of one or more layers attributes with a defined value and/or note are assigned.

From structure

After selecting a structure by *Capture*, *List…* or *All…* the input window in which the assignment parameters are defined appears. This is described in detail in chapter: "Assign structures".

By interpolation…

Some data are available as grid data (eg, a digital elevation model with a 5 m grid). The ground level must be assigned to each node. The location of the model node does not correspond to the grid of the elevation model, so that the data of the elevation has to be interpolated to the model nodes. Various methods are available.

The input parameters are described in chapter: „Tutorial: Step 3: By interpolation of external data formats" in detail.

The theory and the different interpolation algorithms are described in chapter: „Calculations – Interpolation" in detail.

The following input window appears:
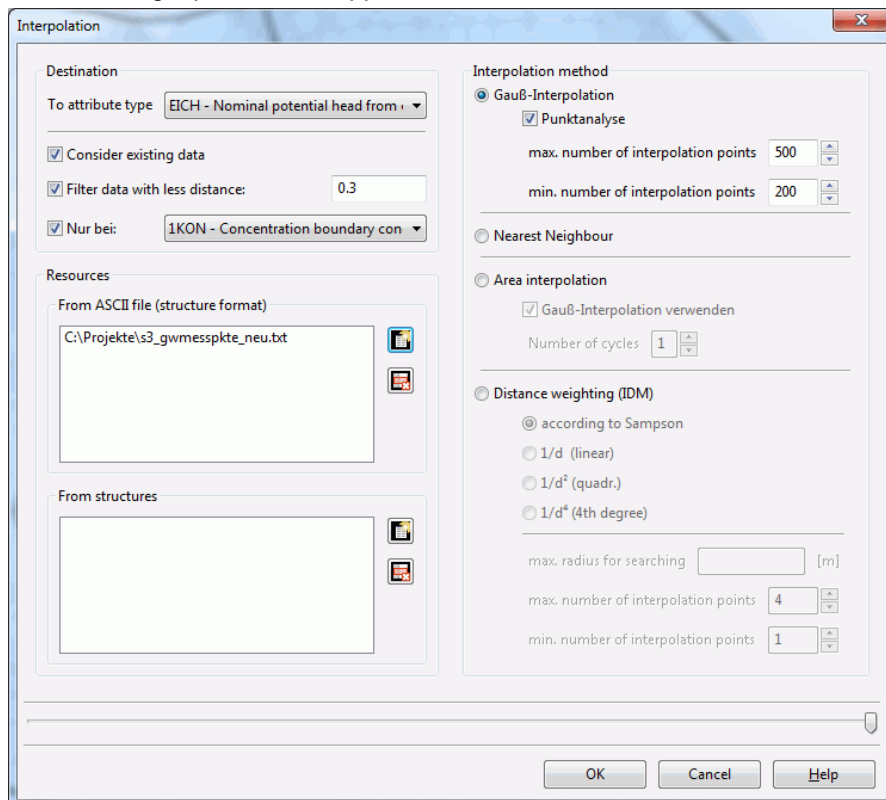
A special feature is the nearest neighbour method. This interpolation method is only available in the attributes menu.

Here, to each node the value of the nearest interpolation point is assigned.

Set note…

The following input window appears:
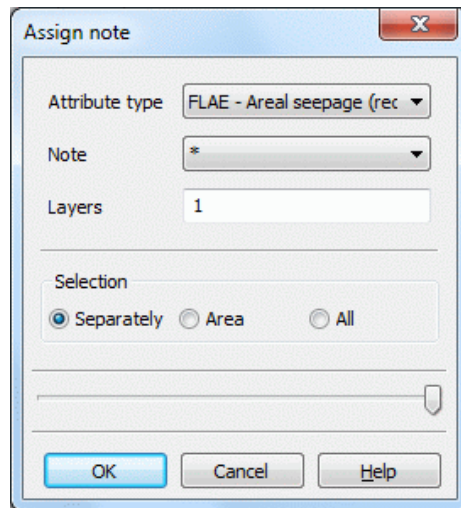
By using this function, you can redefine the signs in the 15th and 16th column in the input file for existing data (see chapter…).

Only the data types are displayed, for which the setting of a note is appropriate or possible.

The note is selected by the arrow button (/, *, F). The layer is determined, and whether the note applies for individual nodes or elements, for nodes or elements in an area or for all nodes or elements.

Stochastical:

With this menu item to all or individual nodes or elements in a layer an attribute with a specific stochastic distribution can be assigned. Since this menu item is very complex, it is treated in a separate chapter (see chapter…).

4.7.2 Computation of attributes

The following submenu appears:
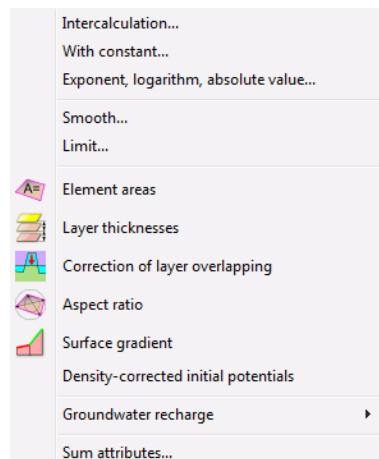
Intercalculation…

The following window appears:
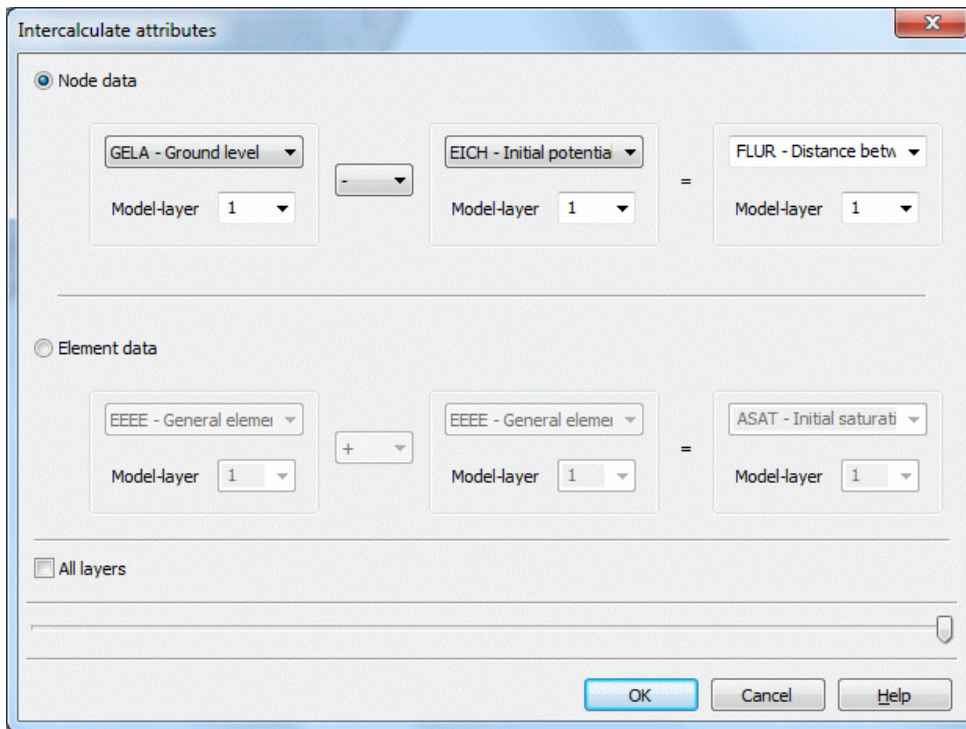
By using this function you can intercalculate two node wise or element wise data types and assign the result to a third node wise or element wise data type.

In case of a 3D model the data can be intercalculated either only in one layer or in all layers.

The resulting data type can be any identification even unknown to SPRING. Depending on the initial data types SPRING will save the new identification as node wise or element wise data type.

With constant…

The following window appears:
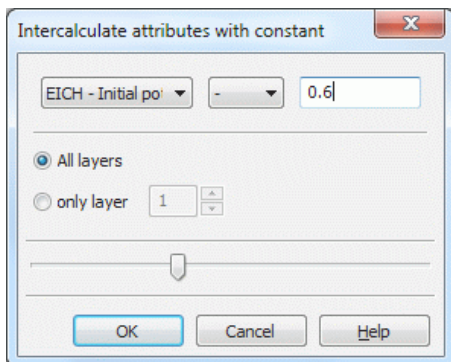
A selected node or element data type (for a 3D model in all or only one layer) can be calculated with a constant. As calculation method +, -, *, / and ^ are available. The result is assigned to the node/element instead ofthe former value. There is no further dialogue on substituting the original values by the calculation results.

Exponent, logarithm, absolute value…

In a 3D model the following window appears:
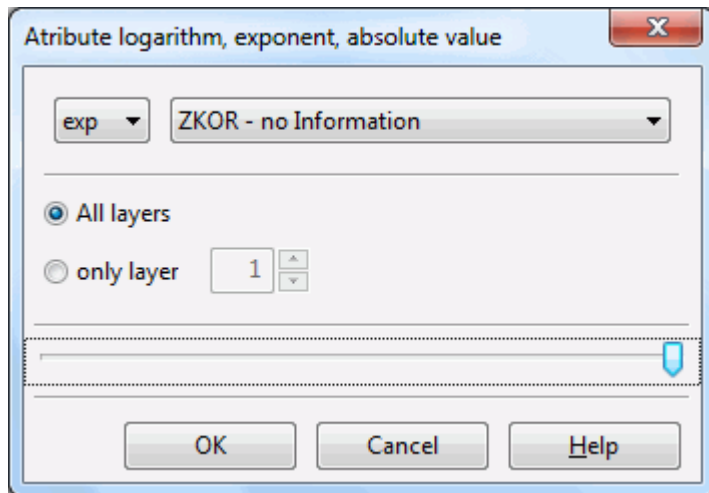
A 2D model lacks the input of a node or element layer.

By using this you can generate new data from your existing data with the help of the following functions:

f(x) = exp(x),

f(x) = 10^x,

f(x) = ln(x) or

f(x) = log(x)

f(x) = abs(x)

In the case of a 3D model you can decide whether the values for all layers or the values for only one layer are intercalculated (transformed) by applying a logarithmic function. At each node or element (in all layers or only one layer) where a value x with the selected identifier exists it is transformed. The result is assigned to the node/element instead the former value. There is no further dialog on substituting the original values by the transformation results.

Smooth…

The following window appears for a 3D model:
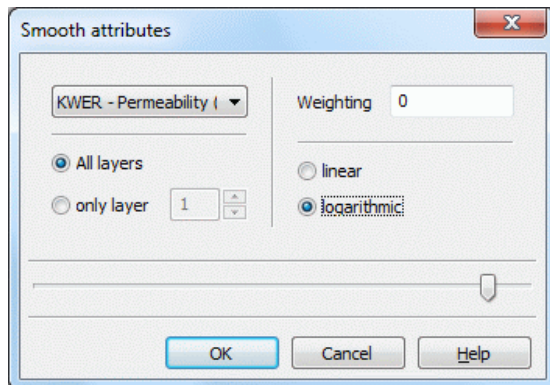
By using one this function you can smooth existing node wise or element wise data by mean value computation.

The new value for a node / element is obtained by adding the mean value of the adjacent nodes / elements to the original value:

Linear smoothing:

$$w^{neu} = \frac{1}{\omega + 1}\left[\omega * w^{alt} + \frac{1}{n}\sum_{i=1}^{n} w_i\right]$$

Logarithmic smoothing:

$$\ln(w^{neu}) = \frac{1}{\omega+1}\left[\omega * \ln(w^{alt}) + \frac{1}{n}\sum_{i=1}^{n}\ln(w_i)\right]$$

with:

    w^{neu}:   new value of a node or element

    ω: weighting

    w^{alt}:   old value of the node or element

    w_i: values of the adjacent nodes or elements

The larger you choose the weighting factor ω the smaller is the smoothing effect.

In case of a 3D model, you can choose whether the data should be smoothed in only one layer or in all layers, e.g. smooth KWER only in layer 7 or in all layers. Nevertheless, the smoothing algorithm uses only the horizontally adjacent element or node values, e.g. the values in layer k are smoothed depending only on values in the same layer k.

Limit…

The following window appears in a 3D model:
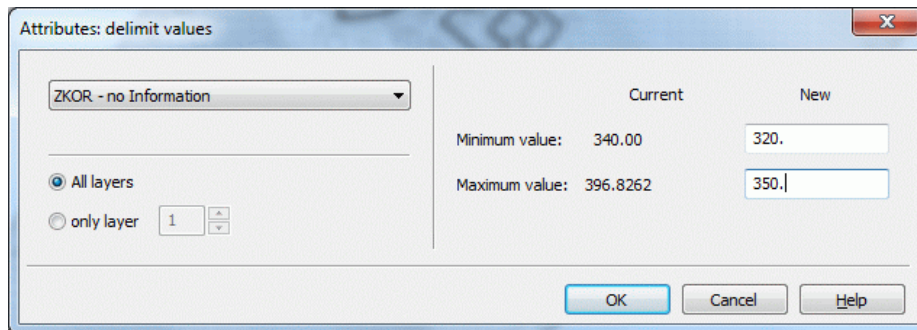
By using this function you can define upper and lower limits for existing data. In the case of 3D models, the values can be limited optionally in all layers or only in one layer.

After selecting the identifier and eventually the layer, the right part of the window reports the current minimum and maximum value. In the text boxes *New* the current extreme values can be substituted by new values. If you confirm the menu with OK the algorithm will compare all existing values with the new extreme values. The current value is substituted by either the new minimum or the new maximum for the case it is less or larger than allowed.

Element areas

By using this function the area of all elements is calculated and stored in the modul file using the attribut AREA [m²].

Layer thicknesses (3D model only)

In case of a 3D model, the function *Layer thickness* computes the layer thickness for all 3D layers by using the z-coordinates (ZKOR) in the following way:

    ZMAE (layer n) = ZKOR (layer n) - ZKOR (layer n+1).

The layer thickness is stored as attribute ZMAE [m] in the model file.

Correction of layer overlapping (3D model only)

When an object (eg a deep pit) cuts through several layers, the nodes inside the object must be placed on the bottom (or top) layer affected. This is done using this function.
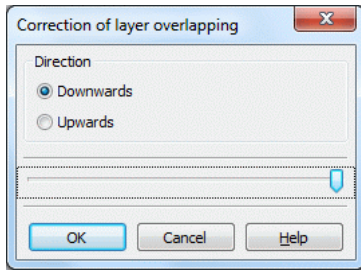
The incorrect layer boundaries are colored red in the node attributes dialog:

ZKOR 3	1000.42
ZKOR 4	-83.17
ZKOR 5	128.66
ZKOR 6	100.20
ZKOR 7	-83.17
ZKOR 8	56.92
ZKOR 9	-83.17
ZKOR 10	-1380.00

Aspect ratio

The quality of the mesh, i.e., of the individual elements, inescapably has a significant effect on the outcome of the numerical simulation, and a poor-quality mesh may lead to an array of problems ranging from solution inconsistencies to more extreme cases where computation is altogether terminated. On the other hand, various factors during the mesh generation process contribute to the emergence of poor quality elements in the final grid; therefore, it is often desirable to detect and resolve such elements prior to the FE calculations.

There are numerous checks and criteria that describe the geometrical quality of elements for finite element analysis, including side length aspect ratios, angles, skewness, Jacobian, warpage, etc. Several different criteria are actively incorporated in SPRING to evaluate element quality as follows.

The Jacobian ratio (JR) is currently used in SPRING as an element quality criterion.

The following attributes are calculated and saved in the model file:

ASPT: aspect ratio

PADV: parallel deviation

MXAG: maximum angle

RATE: element ratio

Attribute "RATE":

The rating system attempts to provide a one-stop overview of the element quality by integrating the aspect ratio, maximum angle, and parallel deviation—when available—criteria into one attribute and taking into account the specific ranges in each criterion that correspond to different quality levels. Therefore, it may be used as a quick approach to analyse the quality of elements and, especially, to identify the extremes of the quality spectrum, i.e., excellent and poor elements. Nonetheless, a closer look at individual criteria might be necessary, either due to the relative importance of one criterion over another, which should be determined on a scenario-to-scenario basis, or in case there is a significant discrepancy between the integrated criteria for an element. The table shows the different quality levels of the RATE attribute along with their implications.

Range	Quality level
1.0-2.0	Excellent
2.0-3.0	Good
3.0-4.0	Average
4.0-5.0	Poor
>5.0	Extremely poor

For 3D models, the tolerance for merging the Z coordinates is also queried:
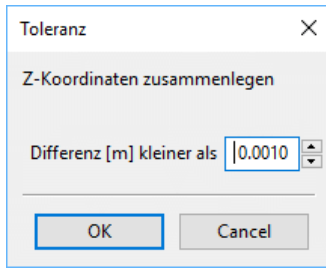
Surface gradient

With this function the surface gradient is calculated for all elements. It is stored as attribute GGRD [%] in the model file.

Density-corrected initial potentials

This menu item is only active if the attribute AKON is assigned in the model file (*.net or *.3d).

The potential heights h [m] EICH, POTE and VORF, entered in the model files or transient input file, must be converted in pressures p for the density-dependent flow calculation:

$$p = (h - z) * \rho * g$$

This is done with the density $\rho = \rho_0$ for the concentration $c_0 = 0$ or temperature $T = T_0$.

If the attributes 1KON or AKON are assigned to nodes with potential heights and hence the concentration $c \neq 0$ (or $T \neq T_0$ in the heat calculation), the potential heights must be converted with the density $\rho = \rho(c)$ in a "corrected" water level. This is done with the following dialog:
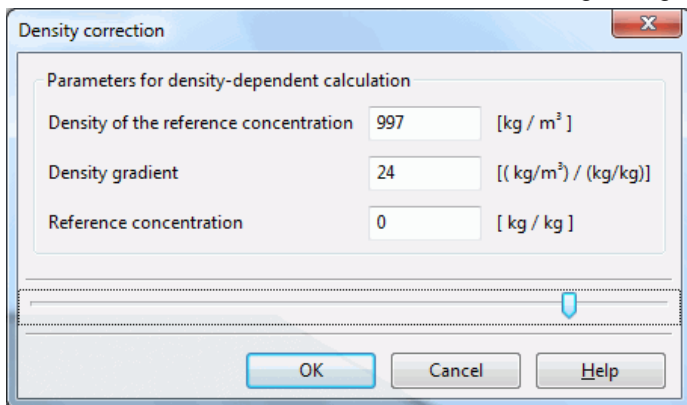
By entering the appropriate density parameters and confirm with OK, the potential heights are corrected and automatically stored in the model file. The parameters entered here correspond to the values that are entered in the dialog of the density-dependent mass transport (p. 324).

Example:

The formula for the density-corrected potential height h_{korr} is:

$$h_{korr} = (h_{Start} - z) * \rho/\rho_0 + z$$

with:

h_{korr} [m] = with the density parameters corrected potential heights EICH, POTE, or VORF

h_{Start} [m] = POTE, EICH, or VORF at a node with a given 1KON or AKON

z [m] = height position of the node, in the 3D model is this the Z-coordinate, in the vertical model is this the Y-coordinate

ρ_0 [kg/m³] = density of the reference concentration $c_0 = 0$

ρ [kg/m³] = known/desired density $\rho(c = c_{max})$, c_{max} is usually the as 1KON or AKON entered maximum concentration

The in the dialog required density slope α is obtained by changing the linear density function that is implemented in the program module SITRA:

$\rho(c) = \rho_0 + \alpha(c - c_0)$

to:

$\alpha = (\rho(c_{max}) - \rho_0) / (c_{max} - c_0)$, if $c = c_{max}$ is used.

Further information about the required parameters and conversions can be found in chapter "Density-dependent mass transport" (p. 323).

Groundwater recharge

For the destination of the groundwater recharge rate are three methods available in SPRING. These are the methods:

- by Schroeder & Wyrwich with the following menu items:

> Determine land use from RVR-Code (KVRN->NSFN)
>
> Transform land use (NMFN->NSFN)
>
> Compute groundwater recharge (NSFN+NSBT+GELA+NIED->FLAE)

- by Meßer 2008 with the menu items:

> Determine land use from RVR code (KVRN->NMFN)
>
> Determine land use from ATKIS code (NATK->NMFN)
>
> Determine land use from CORINE code (NCLC->NMFN)
>
> Identify climatope from ETpot (NETP->NMKL)
>
> Determine classes of depth to groundwater table (FLUR->NMFK)
>
> Determine classes of incline of ground (GELA->NMGK)
>
> Verdunstung berechnen (NMKL,NMBT,NMFK,NMFN->NMET)
>
> Direktabfluss berechnen (NIED,NMET,NMGK,NMBT,NMFK,NMFN->NMAD)
>
> Neubildung berechnen (NIED,NMET,NMAD->FLAE)

- by soil water balance with the menu items:

> Transform land use (NMFN->NSFN)
>
> Compute groundwater recharge (NSFN+NSBT+VERS+GELA+EICH->FLAE)

If the basic attributes exist (KVNR, NATK or NCLC, NETP, FLUR, GELA), more attributes (NSFN, NMFN, NMKL, NMFK and NMGK are computed here. After that the parameters for the water balance equation (direct runoff NMAD and real evapotranspiration NMET) are to be calculated, so that at last the groundwater recharge rate can be destined.

A detailed description of the methods and the input data is given in chapter: „How To – Calculation of averaged groundwater recharge rates"as well as on the website:

"Ein vereinfachtes Verfahren zur Berechnung der flächendifferenzierten Grundwasserneubildung in Mitteleuropa"

- RUBINFLUX...

This menu item leads to the transient recharge computation with RUBINFLUX. A detailed description of the method and the input data is given in chapter: How To – "Calculating a transient groundwater recharge rate" (p. 428).

Sum attributes…

The following window appears in a 3D model:
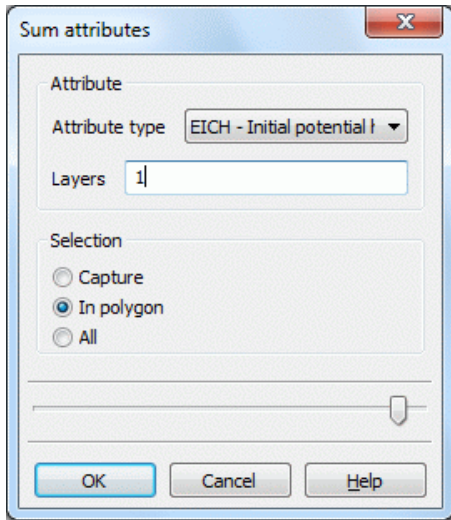
With this function attribute values are added up single, in an area or all and eventually for one or more layers. The result is shown in a dialogue box.

4.7.3 Copy attributes

The following submenu appears:
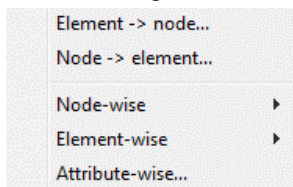
Element → node, Node → element

By using these functions you can convert node into element data and element into node data.

The following input windows appear:
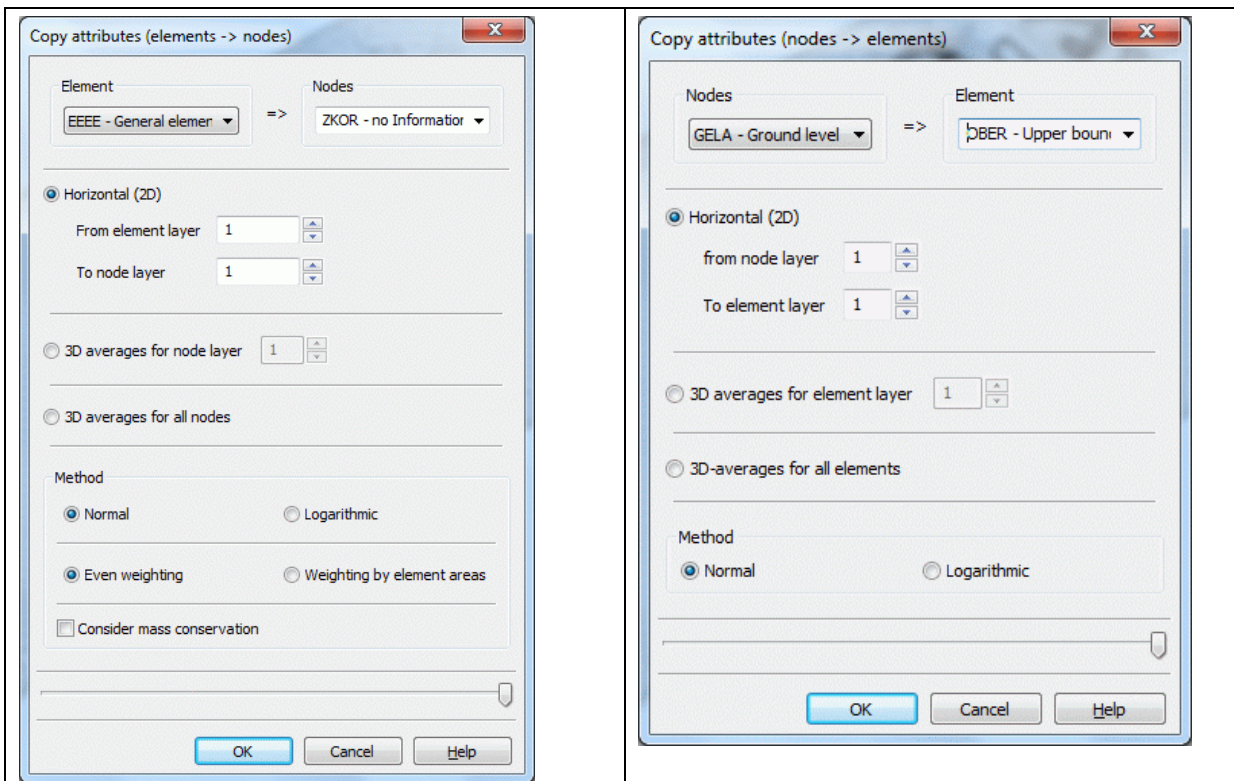
First, the attributes to be copied are defined.

In case of a 3D model, you can select one of three algorithms to compute the mean value:

■ The *horizontal* mean from an element layer to a node layer is computed from the element values horizontally adjacent to the node in the selected element layer and assigned to the corresponding node in the selected node layer (e.g. UNTE layer 1 is averaged to ZKOR layer 5).	■ The *horizontal* mean obtained from a node layer applied to an element layer is computed from the (3 or 4) corner node values in the selected node layer, and is assigned to the corresponding element in the selected element layer (e.g. ZKOR layer 7 is averaged to UNTE layer 1).
■ The *3D mean value for only one node layer* is computed in the same way as the element values of the two element layers adjacent to the selected node layer. In case the node layer k was selected, the algorithm uses the values from the element layers k-1 and k (e.g. KSPE layer 7 as mean value from SPEI).	■ The *3D mean value for only one el*ement is computed from the corner node values of the 3D element (3 or 4 nodes at the top and the bottom). In case the element layer k was selected, the algorithm uses the values from the 3 or 4 corner nodes in the node layers k and k+1 (e.g. SPEI layer 1 as mean value from KSPE).
■ The *3D mean value for all node layers* is computed in the same way as the 3D mean value for one node layer. But the resulting value is assigned to all node layers instead (e.g. KSPE for all node layers as mean value from SPEI)	■ The *3D mean value for all element layers* is computed in the same fashion as the 3D mean value for one element layer. However, the resulting value is assigned to all element layers instead (e.g. SPEI for all element layers as mean value from KSPE).

You can choose between a normal and a logarithmic mean value determination:

Normal (arithmetic) mean value μ:

$$\mu = \frac{1}{n}\sum_{i=1}^{n} w_i$$, w_i are the values in the adjacent elements or nodes which should be converted and n

is the number of values.

Logarithmic mean value μ:

$$\ln(\mu) = \frac{1}{n}\sum_{i=1}^{n} \ln(w_i)$$, w_i are the values in the adjacent elements or nodes which should be converted

and n is the number of values. This method is useful for positive element values with different magnitudes i.e. permeabilities.

Alternative to the arithmetic mean the algorithm can perform *weighting by element areas* where values from larger elements are weighted more heavily than those from smaller elements. A weighting of the element area is via the computation rule: Normal: $$\mu = \frac{\sum_{i=1}^{n} F_i * w_i}{\sum_{i=1}^{n} F_i}$$ e.g. Logarithmic: $$\ln(\mu) = \frac{\sum_{i=1}^{n} F_i * \ln(w_i)}{\sum_{i=1}^{n} F_i}$$ Where F_i is the area oft he adjacent elements.	For the conversion from node to element values just a normal (uniform) weighting is possible: $$\mu = \frac{1}{n}\sum_{i=1}^{n} w_i$$, w_i are the values in the adjacent nodes which should be converted and n is the number of values.

Consider mass conservation:

If the mass conservation is switched on (e.g. conversion of FLAE into KNOT), the element values (e.g. inflow rates) are proportionally distributed to

the corner nodes associated with the element. Then weighting and the type of mean (logarithmic / normal) are unimportant. Each node receives from an adjacent quadrilateral a quarter, or rather from an adjacent triangle a third, of the element rate, etc.

Node wise or element wise

By using one of these functions you can transfer one or more attributes from a certain node (source node) or a certain element (source element) to other nodes or elements. The source node / element is selected either by capturing it in the graphic window or by defining its number. An input window with the existing attributes of the node or element appears:
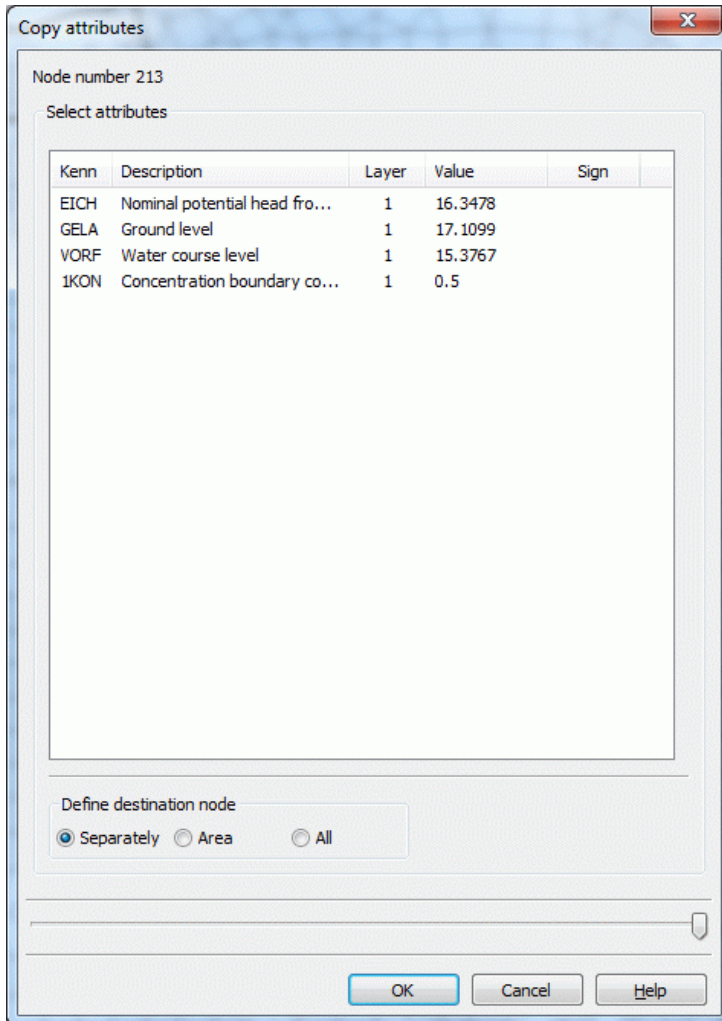
You can select one or more attributes to be copied. The selected attributes are assigned to the target nodes or elements (single, range, or all).

Attribute-wise…

By using this function NODAL you can copy complete data sets of already existing data types. The following window appears (3D model):
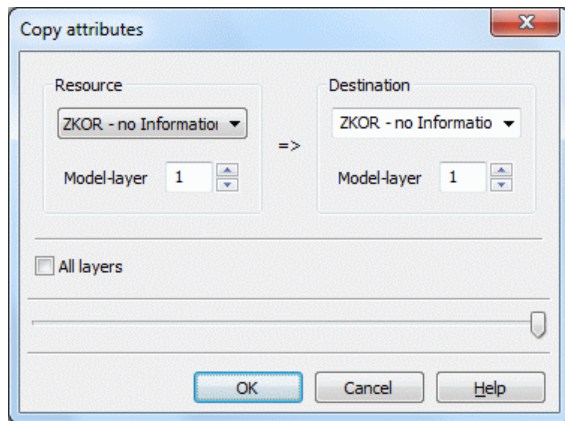
In case of a 2D model, you can copy one existing node or element data type and paste it to another identifier (e.g. POTE → EICH): That means all nodes or elements where the initial data type exists get a new attribute with the identifier of the resulting data type but the same signs and values.

In a 3D model it is possible to copy the data simultaneously in one layer or all layers.

You can enter as target data type any new data identifier (SPRING conform description), but this new type of data gets the same characteristics (e.g., node data, 2D) as the "source data type".

4.7.4 Stochastical assignment of data

The algorithm differentiates between correlated and uncorrelated data.

Uncorrelated generation

The *uncorrelated generation* does not take into account any similarity of data depending on their position relative to one another.
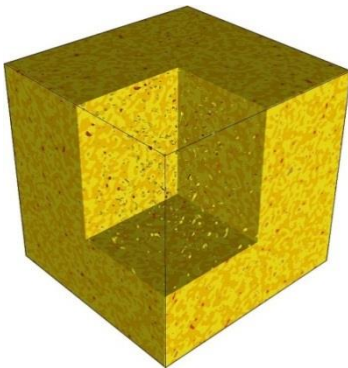
Fig. 1: Stochastically generated data without spatial correlation

Input window: Uncorrelated generation

At first, the attribute, the attribute-note and the layer (in a 3D model) is defined.

For the generation of uncorrelated data the normal-, log-normal - and the exponential-distribution is available. The distribution functions are described in the next chapter.

The textbox *Average* requires the input of the mean value (μ) of the values to be generated as a floating point number. The standard deviation (σ) of the used distribution is required. The input of a standard deviation for the exponential distribution is not necessary!

The generated values are assigned partially or to all nodes or elements of the layer.

Correlated generation

The correlated generation of data in SPRING is based on the turning bands method (see also "A. Tompson, R. Ababou, L. Gelhar"; Implementation of the Three-Dimensional Turning Bands Random Field Generator; Water Resources Research; Vol. 25, No. 10, 1989). The implemented algorithm works with an exponential function for the variogram where the correlation lengths can be different for the three spatial dimensions.
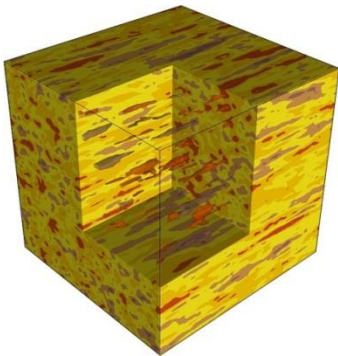
Fig. 2: Stochastically generated data with spatial correlation

Input window: Correlated generation

At first, the attribute, the attribute-note and the layer (in a 3D model) is defined. For the generation of correlated data only the normal and the log-normal but not the exponential distribution are available.

For generating correlated data using the turning bands method the input of *correlation lengths* (in m) is necessary. It only makes sense for the following two conditions are fulfilled: the mesh density, i. e. the element size has to be significant less than the defined correlation length. The region the values are generated inside has to be at minimum twice as large as the correlation length in the corresponding direction.

The implemented algorithm requires general information on the mesh density. The *Point distance* to be inserted in the corresponding textbox should lie between the smallest and the mean element sizes in the corresponding directions. The quality of the generated data will be better when these parameters are defined as finely as possible. Admittedly, finer values will increase the processing-time requirement.

The typical density function for the generated values will only result for a sufficient number of generated values. Additionally, for the case of a correlated generation, the relationships between mesh density, node ditch and correlation length mentioned above should be fulfilled.

4.7.5 Distribution functions

The stochastically assignment uses the following distribution functions:

■   Normal distribution

The normal distribution has the density function:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

.

Input values for the generation of normally distributed values (x) are:

σ = Standard deviation

σ^2 = Variance

μ = Mean value
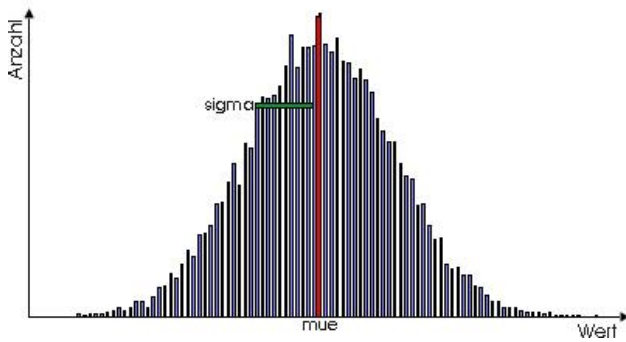
Fig. 75: Density function for nearly 18,000 generated values using a normal distribution

■   Log-normal distribution

In the log-normal distribution the logarithm of the generated values is normally distributed:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}}\, e^{-\frac{(\ln x - \mu)^2}{2\sigma^2}}$$

Input values for the generation of log-normally distributed values (x > 0) are:

σ = Standard deviation

σ^2 = Variance

μ = Mean value
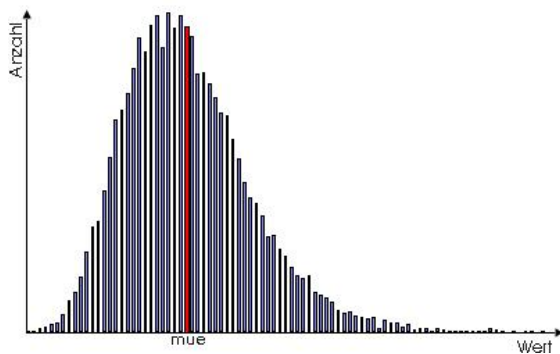
Fig. 76: Density function for nearly 18,000 generated values using a log-normal distribution

■   Exponential distribution (uncorrelated generation only)

The exponential distribution has for x > 0 the density function:

$$f(x) = \frac{1}{\mu}\, e^{-\frac{x}{\mu}}$$

Input value for the generation of exponential distributed values x > 0 is the mean value μ.
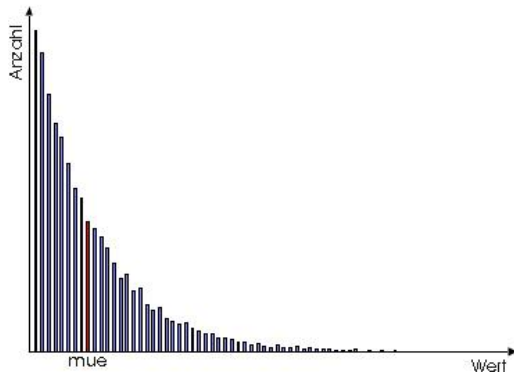
Fig. 77: Density function for nearly 18,000 generated values using an exponential distribution

4.7.6 Anisotropic conductivities

If anisotropic conductivities should be taken into account the elements with the same anisotropy are assigned to a zone (attribute ZONE in a horizontal model, attribute Z-KA in a 3D model) first (*Attributes → Assign → Direct...*).

Then the corresponding K-values and angles are assigned to the individual zones via the following input window (3D):
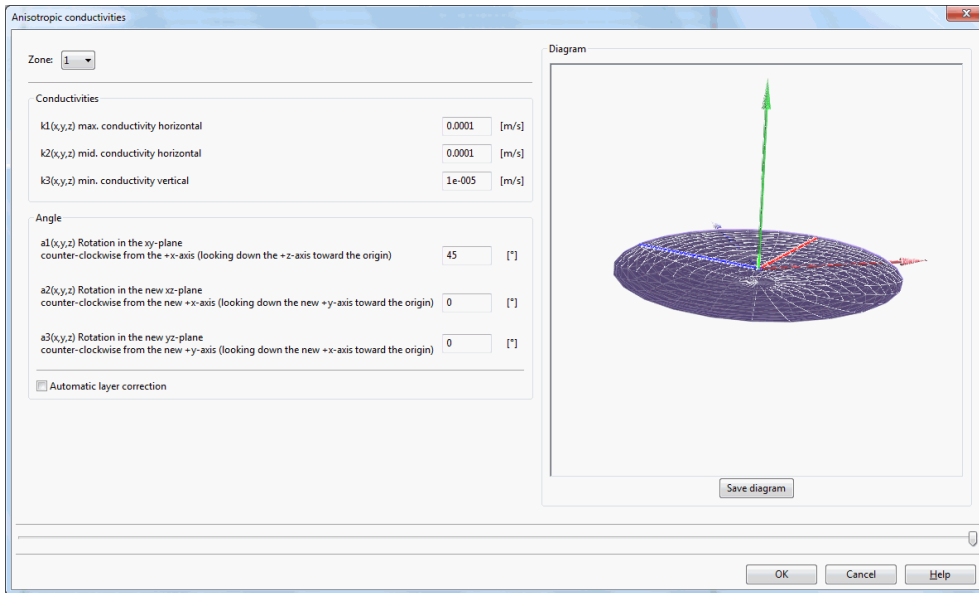
In a horizontal model the following input is required:
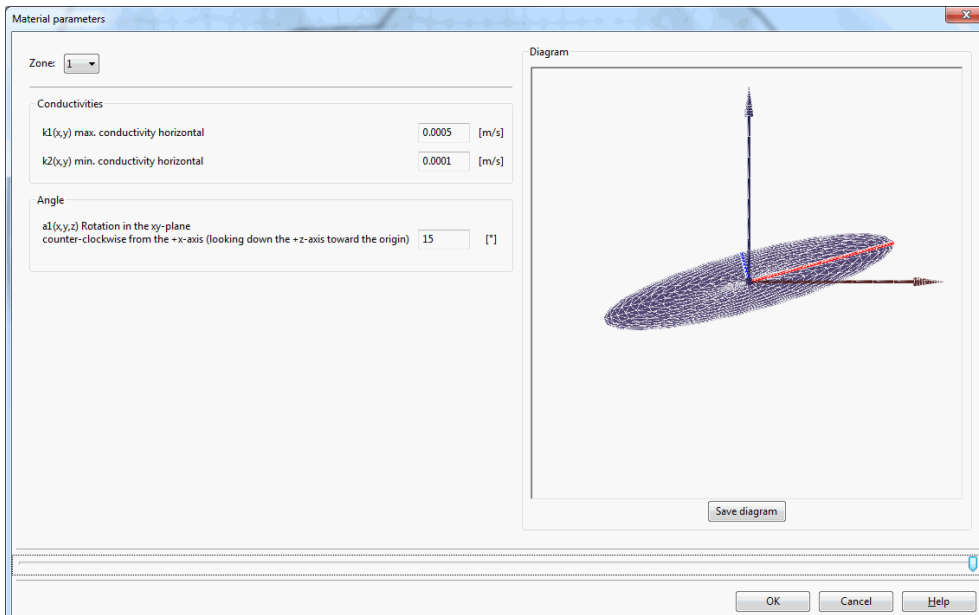
Conductivities

At this point anisotropic conductivities can be assigned differentiated according to their spatial direction.

In a 3D model the input conductivities k1, k2, k3 are stored together with the angles a1, a2 and a3 at the end of the *.3d model file after confirming with *OK* and saving the model. They are stored for each zone automatically. The attribute Z-KA will automatically be assigned to the deeper layers, if it is not assigned there.

In a horizontal model the input conductivities k1 and k2 are stored together with the angle a1 automatically as attribute MATE in the *.net file after confirming with *OK* and saving the model.

Angles

In addition to the direction-dependent conductivities the angles a1 (horizontal and 3D model), a2 and a3 (3D model only) can be defined, if the conductivity-ellipsoid differs from the Cartesian coordinate system.

By taking into account these angles, the conductivities are corrected in accordance with the following general rotation algorithm:

$k_{korr} = H * k_{orig} * H^T$

with:

k_{korr} = Tensor of the (layer-)corrected conductivities

k_{orig} = Tensor of the conductivities of the model data

H = Rotation matrix based on the angles a1, a2, a3

H^T = transposed rotation matrix

Angle a1: Rotation in the XY-plane, counter clockwise from the X-axis (looking down the Z-axis toward the origin)

Angle a2: Rotation in the new XZ-plane, counter clockwise from the new X-axis (looking down the Y-axis toward the origin)

Angle a3: Rotation in the new YZ-plane, counter clockwise from the new Y-axis (looking down the new X-axis toward the origin)

Preview diagram

The dashed axes in the preview (3D example) indicate the Cartesian coordinate system x, y, and z. The angle a1 = 45 ° means that the maximum conductivity is applied in the direction of 45 ° relative to the x-axis. In the example it can be seen that this information results in a rotation of the xy-plane by 45° (the solid red axis is the new x-axis). The diagram can be saved as an image at any location.

It is possible to change the filled shader model in a line graph with the key "f" (after activating the preview window with the left mouse button). Similarly, the model can be illuminated with the key "l".

Shader model as line (key "f")	Light model on (key "l")
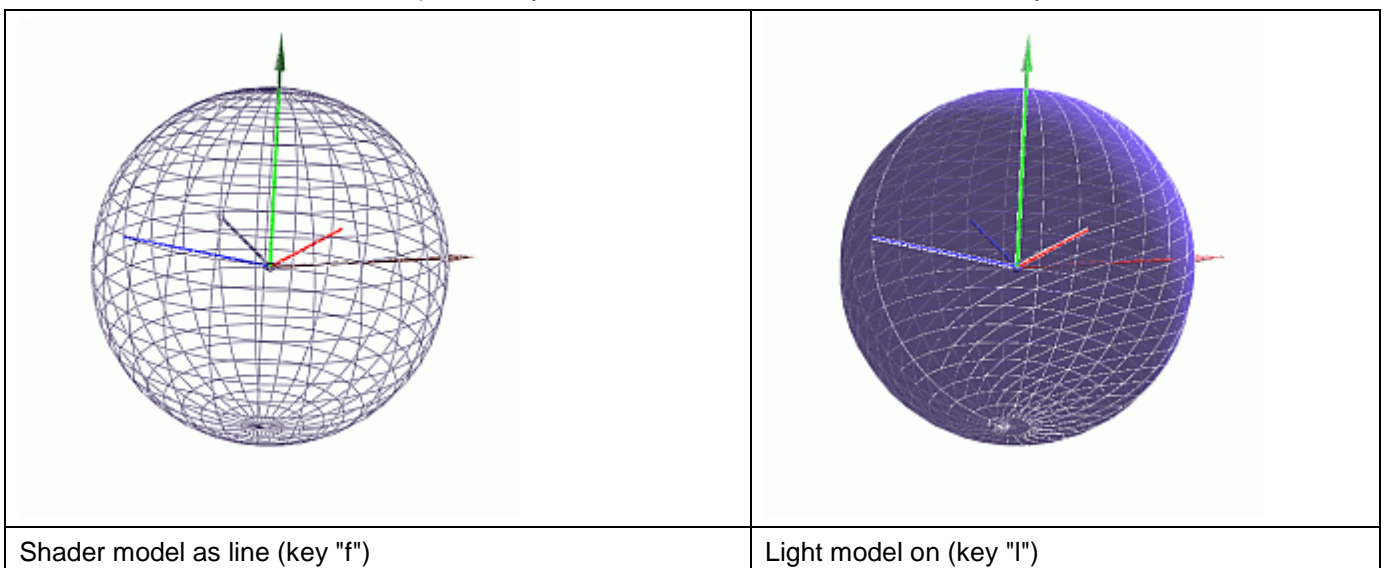
For this illustration, k1 = k2 = k3 and a1 = 45 ° were set.

Automatic layer correction (3D only)

The permeability values (attribute KWER) defined in SPRING are generally for isotropic flow conditions. That means the resulting flow direction is given by the pure potential gradient of the aquifer without considering the geometric discretization.

By activating this button an optional (and automatic) layer correction takes place in which the conductivities are automatically adjusted to the element or layer geometry. Since the element geometry adapts typically the geological stratification of the model area, a geometric correction of the permeabilities takes place without giving angles.

For the internal calculation the average level between the upper and lower "element surface" is generated, and its normal is calculated. The deviations of the normal to the respective coordinate plane result in the angles a1, a2, a3, with which the conductivities are corrected in accordance to the general rotation algorithm described above.

If the conductivities should be corrected for ALL layers, an assignment of the attribute Z-KA is not required. For this purpose, it is enough, to add the following line at the end of the batch file *sitra.bsi*:

```
ROTATION KWER
```

The following figure shows the result of the flow calculation with (red potential lines) and without (blue potential lines) consideration of the layer-corrected conductivities for all layers:
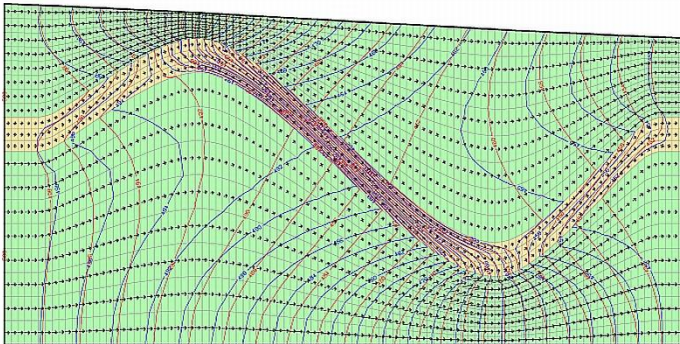
Fig. 78: Comparison of the potential lines with (red) and without (blue) layer corrected conductivities for all layers

The yellow layer is impermeable against the green layers by a factor of 100.

If the conductivities should be corrected only for selected layers, this is done by zoning the layers with the attribute Z-KA.

For this purpose, the appropriate zone is selected in the dialog and the button "Automatic layer correction" is enabled.

The following figure shows a model in which only the layers 9 to 11 were defined as Zone 2. The conductivities are the same for all layers (horizontal 0.0001, 0.0001 vertically), as shown by the associated element attributes:

KENN	Layer	Value	Z	Group	Description
KWER	1	0.0001			Permeability (m/s)
UNTE	1	0			Lower boundary of aquifer
Z-KA	1	1			Zone for ansiotropic permeability
Z-KA	9	2			Zone for ansiotropic permeability
Z-KA	12	1			Zone for ansiotropic permeability

Due to the automatic assignment of deeper layers the attribute Z-KA must be reassigned from the layer 12 on, because zone 2 shall apply only to the layers 9 to 11.

Then an automatic layer correction was carried out for Zone 2. The following figure shows the result of the flow calculation:
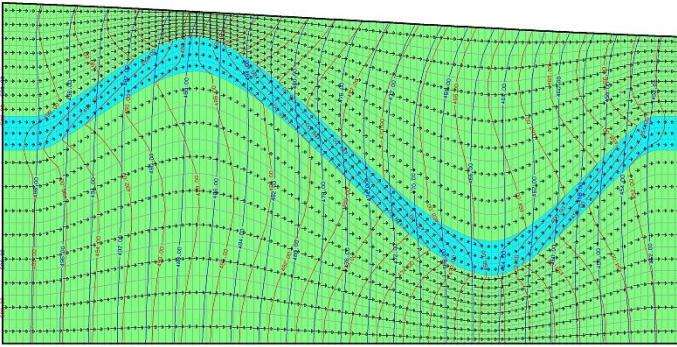
Fig. 79: Comparison of the potential lines with (red) and without (blue) layer corrected conductivities in zone 2

Zone 2 includes the layers 9 to 11, which are colored in turquoise. Zone 1are all other layers (green).

4.8 Map

The map menu is only available in the plot mode. The following menu items are available:
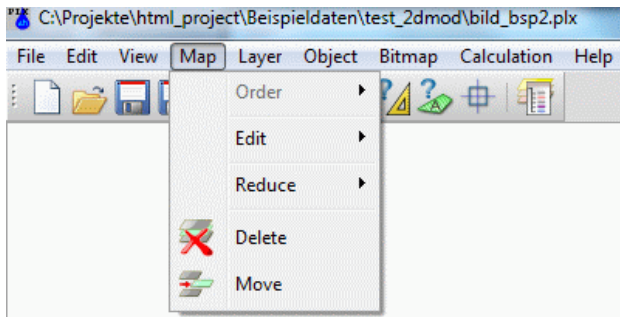
Order

The following submenu appears:

Sometimes it is necessary to change the order of the represented maps. The functions in the submenu provide the corresponding algorithms. Activating one of the functions transforms the cursor into a select cursor to identify the corresponding map by picking an object.

The order of graphic objects represented on the screen and on the plot depends on the order of the objects in the graphic files. For the case of loading/overlaying more than one file, the map loaded last is represented last (i.e. on the top). The order of representation influences which objects cover other objects. This is especially important for filled domains or exactly equal lines.

Edit

The following submenu appears:
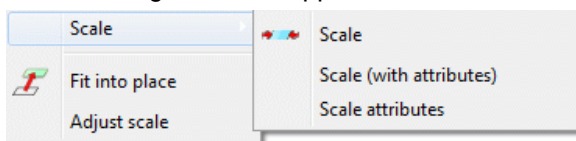
■   Scale ▬▬, Scale (with attributes)

The function *Scale* can change the scale of maps represented on the screen. For the case of several maps on the screen, the required map must be identified using the select cursor. Then a dialogue box appears, in which the scales are inserted:
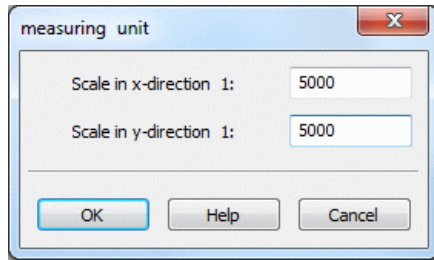
It is possible to define different scales for both coordinate directions. They do not depend on each other. For a simultaneous scaling of the object attributes (i.e. font size, mark size and line-width) use the function *Scale with attributes*.

■ Scale attributes

Often the font sizes, mark sizes and line-widths in a map are generally too small or too large. The function provides the possibility to scale all attributes of a map using one scaling factor. A scaling factor > 1.0 increases the attributes while a factor < 1.0 decreases them. The map retains its original scale. For the case of several maps on the screen, the required map must be identified using the select cursor. Then a dialog box appears where the scaling factor is inserted:
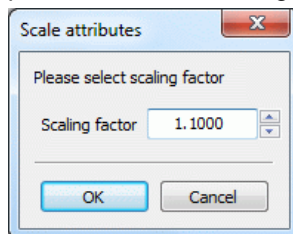
■ Fit into place

This function helps to fit the coordinate system of a map to prepare the overlaying of another map:
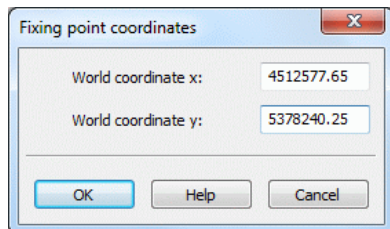
Details how to use the function are described in the menu item *Bitmap → Fit into place* .

■ Adjust scale

Files without any scale information (structure files, dxf-files) are loaded and are assigned to a default scale of 1:10,000. You can correct the scale using the function *Adjust scale*. The corresponding map is identified using the cursor. For the case that more than one map has been loaded a dialogue box appears where the correct scales are inserted. This function does not change the representation size of the map.
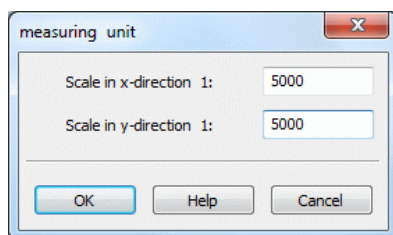
Reduce

To a single layer:

Objects can change their position in the order only inside the corresponding layer. But sometimes it is necessary to change the position of a single object in relation to another layer. This function can solve the problem. For the case of several maps on the screen, the required map must be identified using the select cursor. Then all layers of the map are merged into one layer. The different layers of the map are lost.

To a single map:

This function merges all currently represented maps into one new one, on the condition that the maps were added successively. Often logos are inserted as a separate map and then positioned and scaled into the already-existing frame. Afterwards it is useful to merge the logo and the frame to one map, because they can be handled together much easier. Merged maps are always reduced to one coordinate system of a reference map. The reference map is identified using the select cursor. The coordinate systems of all other maps are lost.

Delete ✗, Move ⇄

The function *Delete* removes the whole map and every graphic object belonging to it.

The function *Move* can be used when several maps are loaded and represented on the screen. To identify one of the represented maps, you must use the cursor to capture a graphic object belonging to the map. Then, press the left mouse button and move the map which is shown as rectangular frame to the new position. By releasing the mouse button, the map is represented at the new position.

4.9 Layer

The layer menu is available both in the plot and model mode. It is up to the menu item "New" (plot mode only) identical in both modes. The following menu items are available:
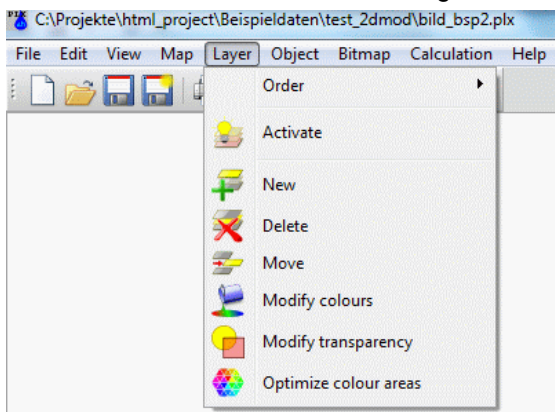
Order

The following submenu appears:
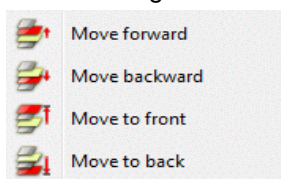
The functions *Move forward* or rather *backward* change the position of the layer that it is represented more in the foreground (later) or more in the background (earlier) of the map. The function *Move to front* puts the layer into the foreground, i.e. it is represented last and covers all other layers in the map. The function *Move to back* puts the selected layer into the background of the map i. e. it is represented first and is covered by all other layers.

The order of representation for layers can be changed only inside the corresponding map. Therefore, the function *Move to front* places the layer in the topmost position of the corresponding map and the function *Move to back* puts the layer in the bottom position of the map. To change the order of two layers belonging to different maps, you must use the corresponding functions for maps. Alternatively, you can merge all loaded maps to one map.

Activate 🔅

This function defines the layer in which the objects created via *Object → Edit* are inserted or whose properties are to be changed. The layer is identified by choosing an object belonging to it.

New

Using this menu item a new, empty layer will be created in a map. On this layer then new objects can be drawn.

Delete

This function removes the whole layer and every graphic object belonging to it. To identify one of the represented layers, you must capture a graphic object belonging to the layer with the cursor. If there is only one layer, it is automatically removed by activating the function.

Move

This function provides the possibility to move all objects belonging to one layer at once. The layer is identified with the select cursor and moved by pressing the left mouse button until reaching the new position. For orientation purposes, the bounding box of the layer is shown on the screen.

When multiple cards are opened (plot mode), the layer in the layer list can be moved into another map through the project manager (drag & drop) and the scaling is automatically adjusted to the "goal card" (the layer is overlaid). The map, from which a layer should be moved to a different map, it must contain at least one more layer. If necessary, a new blank layer can be easily created in this map. Subsequently, the layers are shifted.

Modify colours

The function *Modify colours* can re-define single colours. The changes influence only the selected layer. For the case of several layers on the screen, the required layer must be identified using the select cursor. The window where the colour changes are specified appears.
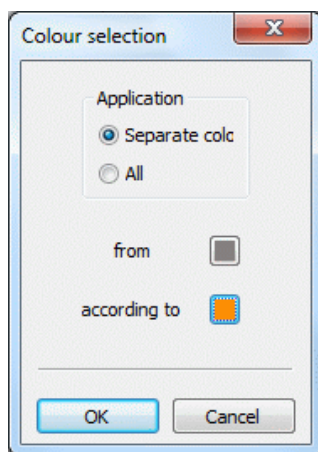
First is to decide whether the choice of colour applies to an object of the layer or the entire layer.

If the existing colour should be changed a colour palette appears by pressing the button *according to* in which the desired colour can be selected.

Modify transparency

The transparency of a layer may be set between "invisible" (= 0, total transparency) and "opaque" (= 255, no transparency) using this input window:
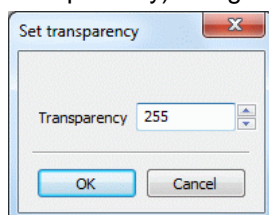
The alternative graphic system (p. 75), however, must first be enabled in the the menu *Edit → Options → In General.*

Optimize colour areas

In an area plot arise according to the discretization of the elements at first colour-filled polygons. This function provides an algorithm to decrease the number of coloured domains in the current graphic. It optimizes layers with many adjacent polygons of the same colour by merging them into one polygon. The function is typically useful for equal value domains and hatched domains i.e. plots where the attributes for element wise values are represented. The optimization has advantages for data storage as well as for working with data in SPRING. The latter is so because of a much faster graphics-repaint rate. For the case of several layers on the screen, the required layer must be identified using the select cursor. The following window appears:
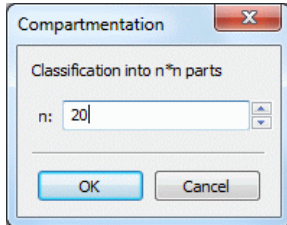
Before optimizing the layer is divided into n sections in which the areas are first sorted. After that each section is optimized separately, which brings a significant speed advantage. The following applies: The more section areas, the faster the optimization.

4.10 Object

The object menu is available both in the plot and model mode. It is identically in both modes. The following menu items are available:
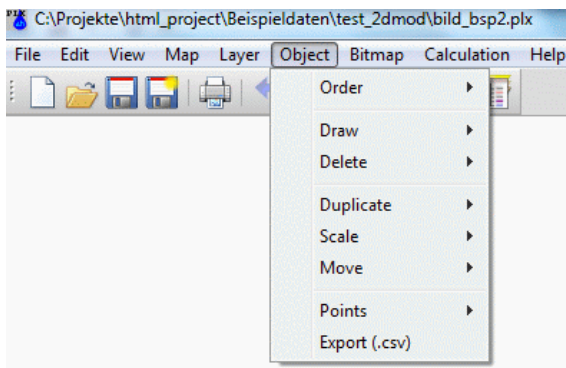
Order

The following submenu appears:

The four items allow a change in the position of individual objects in the order within their layer.

The functions *Move forward* or rather *backward* change the position of the object that it is represented more in the foreground (later) or more in the background (earlier) of the layer. The function *Move to front* puts the object into the foreground, i.e. it is represented last and covers all other objects in the layer. The function *Move to back* puts the selected object into the background of the map i.e. it is represented first and is covered by all other objects.

The order of representation for objects can be changed only inside the corresponding layer. Therefore, the function *Move to front* places the object in the topmost position of the corresponding layer and the function *Move to back* puts the object in the bottom position of the layer. To change the order of two objects belonging to different layers, you must use the corresponding functions for layers. Alternatively, you can merge all loaded layers to one layer.

Draw

The following submenu appears:
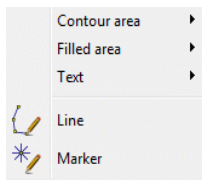
■ Flat, Filled area

There is a choice between these forms:

The menu item *Flat* draws only the outline of an area while the menu item *Filled area* draws an area filled with colour.

■ Text

The following selection is available:
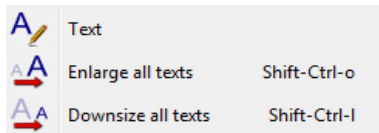
When selecting the menu item *Text* new text is added to the layer. For this, first a position has to be set by clicking with the left mouse button. This will open a dialog window in which the desired text is entered.

Selecting the menu item *Enlarge all texts* all existing texts are automatically scaled larger.

Selecting the menu item *Downsize all texts* all existing texts are automatically scaled smaller.

If only a single text (or object) should be increased or decreased, this is done with the toolbar Object properties. This has to be displayed before via the menu *View → Toolbars → Object*. With the button "Show Properties" () and selecting an object (left mouse button) the properties (colour, height, etc.) of the selected object are displayed in the toolbar. Then there can be entered, for example, a different height H for the text. The checkbox at the left side of H must be enabled, so that the height in the subsequent assignment of the object properties is changed. The assignment of the object properties is done by clicking on the button in the toolbar and then selecting the text to be changed with the left mouse button. This approach is possible for all objects.

■ Line

The menu item *Line* allows you to enter a line with any number of points which are connected by straight lines. After selecting the menu item, the cursor is positioned on the starting point of the line by clicking the left mouse button. The cursor goes to the second point and marks it again by clicking the left mouse button. This procedure is repeated until all points of the polyline are entered. Pressing the right mouse button ends the function without defining a new point.

The application remains in the line drawing mode, until the function is ended with the right mouse button or another function is activated.

■ Marker

With the menu item *Marker* symbols are placed at the selected location of the graphic by clicking the left mouse button. There are 35 marker types available. Markers are produced so long until the right mouse button or another function is activated. The symbols and their height are selected via the toolbar *View → Toolbars → Object properties*. The toolbar is to be displayed before.

Delete

Using this menu item objects can be deleted individually () or in a defined area () by clicking the left mouse button.

The deletion can be undone by the *Undo* function or repeated by the *Redo* function .

Duplicate

This function creates a copy from an already existing graphic object. The corresponding object is identified with the cursor. In the appearing selection window the layer can be chosen in which the selected object is copied:
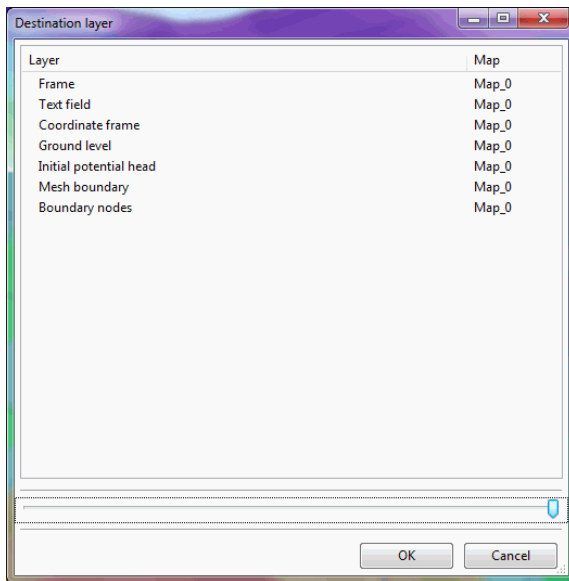
After activating the chosen layer in the project manager the duplicate can immediately be moved to the right position via the menu item *Object → Move*.

Scale

Using this menu item, an object can be scaled individually (🖑) or in an area (🖑). After selecting one or more objects with the left mouse button and confirming them with the right mouse button the following input window appears:
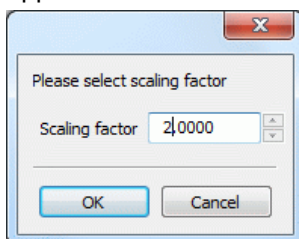
A scaling factor > 1.0 enlarges the objects, a factor between 0 and 1.0 makes the objects smaller.

The scaling can be undone by the *Undo* function ↶ or repeated by the *Redo* function ↷ .

.

Move

Using this menu item objects can be moved individually (🖑) or in an area (🖑). The desired object or the area are selected with the left mouse button and dragged to the desired location. When selecting an area it is important to note which layer is active.

Moving can be undone by the *Undo* function ↶ or repeated by the *Redo* function ↷ .

Points

This menu item is used for editing line or polygon objects.

The following submenu appears:
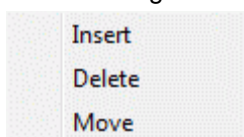
Using the menu item *Insert* intermediate points can be inserted into existing polylines or polygon objects. The object is selected with the left mouse button and the points are marked at the desired location with the left mouse button.

The menu item *Delete* deletes the selected points from an existing object. For this, the object is selected first with the left mouse button. With further clicks of the left mouse button on the points to be deleted the object changes.

With the menu item *Move* the position of individual points of a line or polygon object can be changed. For this, the object is selected first with the left mouse button. By holding the left mouse button the point is moved. By releasing the mouse button, the point remains at the desired position.

The right mouse button ends each function for the selected object. By selecting with the left mouse button, a further object can be edited. To terminate the individual functions either a different function is called, or the right mouse button has to be pressed again.

Export *.csv

After selecting an object with the left mouse button, the file selection window appears, where the name of the *.csv file is set. In the file, the coordinates of the object are stored in the following manner:

- Point object (e.g. marker ate wells): The coordinates of the centre of the marker are stored.
- Line object: The coordinates of each line section are stored.
- Area object: The coordinates of the corners are stored.

About another selection with the left mouse button a further object can be exported. To terminate the function either a different function is called, or the right mouse button has to be pressed again.

4.11 Bitmap

The bitmap menu is available both in the plot and model mode. It is identically in both modes. The following menu items are available:
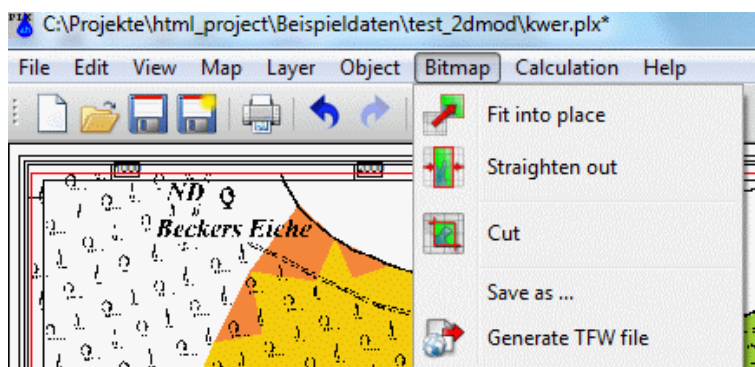
Fit into place , Straighten out

The functions *Fit into place* and *Straighten out* provide algorithms for connecting the bitmap with the corresponding coordinate information. This is necessary for overlaying other graphics at the correct position. The procedure is as follows:

The original TIFF file does not contain any scale information. Thus, SPRING loads this file and assigns the default scale of 1:10,000. First, you must specify the correct scale using *Bitmap → Fit into scale*. If the scale is unknown, it can be determined by selecting an object whose real length is known, and calculating its apparent length from the relationship between the real and the apparent length, this is done using the function *View → Measure distance/angle*. From the difference between the real distance and the computed length, the correct scale of the bitmap can be obtained.

Afterwards, you should activate the function *Straighten out* and define a point whose map coordinates are known. A dialog box appears, in which you can insert the coordinates of the point:
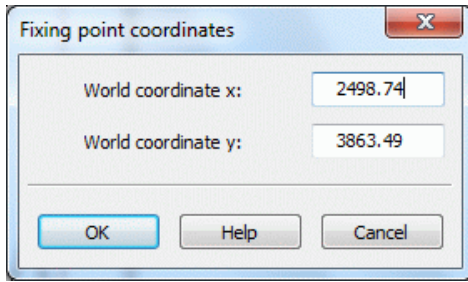
If the distortion of the bitmap has to be corrected, the point should, if possible, be located near the bottom left hand corner.

To correct distortions of the bitmap one can use the function *Straighten out* where you have to define a second point whose coordinates are known. The distortion is corrected linearly using the two points. It is not able to correct complex distortions of the bitmap.

If a TIFF file is read, for which a *.tfw file (in the same directory) is present, the fitting of the bitmap is not necessary. The file names of the "*.tif" and the "*.tfw" file have to be (except the file extension) identical.

Cut

The function *Cut* can clip a rectangular window from all loaded bitmaps. The function is useful where only a part of the graphic is required for the further work. Equivalent to the zoom function, the window is defined by left mouse clicks in two opposing corners. As a result, all areas outside this frame are removed from the corresponding bitmaps. Afterwards, you should save the bitmap by using *Save*. Note that bitmap modifications are not saved by means of the normal saving procedure as plx-file.

Save as…

The function *Save as…* saves the actual state of the bitmap including all changes. Note that bitmap modifications are not saved by means of the normal saving procedure as plx-file. That's why you should save the bitmap separately after changes have been made. For more than one bitmap loaded in the actual map, you must identify the corresponding bitmap with the left mouse button and save the bitmaps consecutively. A file selection window appears where the name of the TIFF file can be entered.

Black-white TIFF files are stored in the compression CCITT FAX 4, coloured TIFF files in the compression PACKBITS.

Generate TFW file

In addition to the normal saving as TIFF files the geographic information for the scaled and positioned bitmaps can be saved to a TFW file. This file format is often used in ARC/VIEW to load TIFF files. For more than one bitmap loaded in the actual map you have to identify the corresponding bitmap with the left mouse button and save the bitmaps one after each other. The name of the *.tfw file is automatically formed from the name of the bitmap file.

4.12 Special features

The special features menu is only available in the model mode. The following menu items are available:
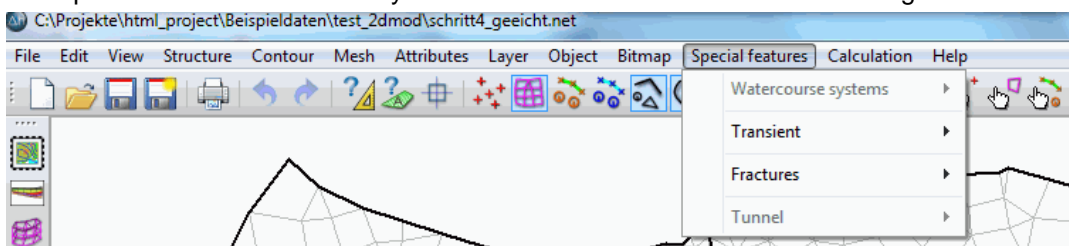
Water course systems

In SPRING is an integrated network of water-related systems available in order to consider the exchange processes between surface water and ground water. This tool is available for steady state and transient problems in a 2D or 3D model.

The network of water-related systems determines the groundwater dependent runoff also about the deeper layers along the flow direction. This requires the definition of water course systems in the deeper layers, but not the parameters "wetted perimeter", "Manning-Strickler coefficient", "slope" or "gradient".

To use this enhancement of the interaction between surface water and groundwater it is necessary to ensure that a clear direction of flow and the link of the water branches are given.
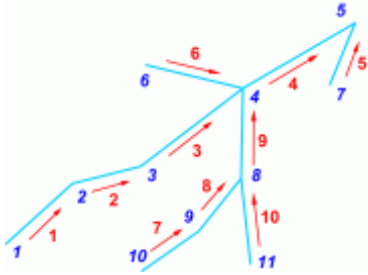
Even if a derivation of water levels (VORF data as a result) in deeper layers is technically possible, this is not provided, because the layer height of the node layer is considered. It makes sense, therefore, the definition of the water course systems is assigned to the top layer.

Attention:

■   A multiple assignment of water systems and water system-related parameters should be avoided, since there is no attribute check.

■   A water system can only lie in a single node layer, it cannot partially change the layer.

The following functions are selectable in the submenu:
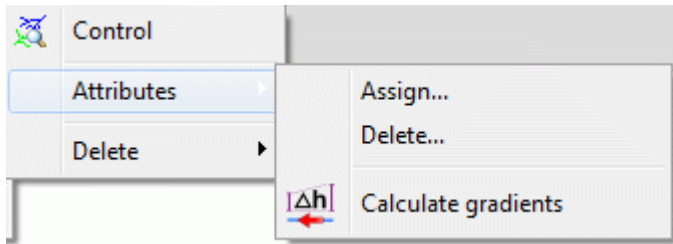
■   Control

A check on the accuracy of the flow direction of the water systems takes place. Due to the conservation of mass flows at each node exactly one water stream flows away from one node and any number may flow to the node.

The separation of a water in two opposite flow directions results in an error message.

■   Attributes assign/delete

An input window appears in which the water system attributes can be defined and assigned to nodes.
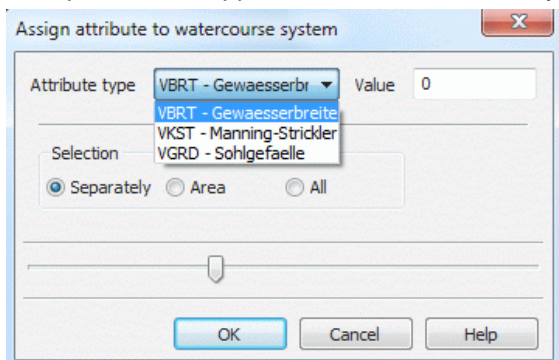
The following window appears when activating the menu item *Delete*:
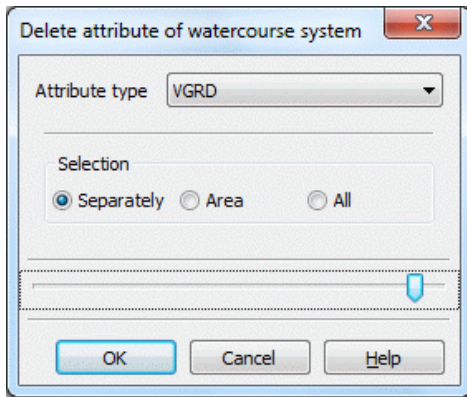
■   Calculate gradients

After selecting this function, a dialog window appears where the user can choose whether the slope gradient is calculated from the receiving water course level or the ground level.
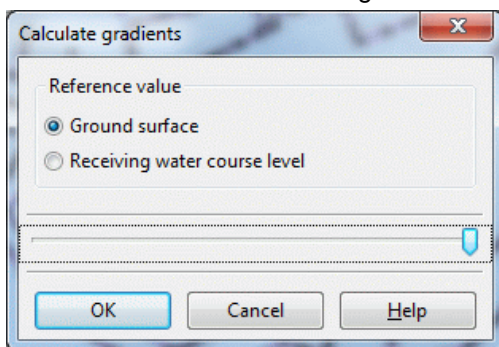
The attribute VGRD is calculated for all water course systems (unit: promille). After saving the project the attribute is stored in the model file.

The gradient (corresponding to the bottom slope) is determined by the ratio of the difference of the ground surface or receiving water course level to the distance between adjacent nodes along the element edge. In the menu *View → Show attributes → On watercourse systems...* the gradient direction on the element edges can be displayed as arrows by selecting the attribute VGRD.

■   Delete

If water course systems are available, they can be deleted individually, or in an area or completely, because they are stored as a separate layer.

Transient

Since this menu item is very complex, it is treated in a separate subchapter (4.12.1).

Fractures

If fracture contours are present (chapter: "Generate fracture contours"), the following functions are selectable in the submenu:

Show 1D-fractures
Show 2D-fractures (top view)

Generate 1D-fractures from element edges
Generate 1D-fractures from fracture contour ▶
1D-fractures geometry check...

Assign 1D fracture attribute...

Delete 1D-fractures ▶

Generate 2D-fractures from fracture contour
2D-fractures from LERA
2D-fratures from Attribute...
Pick 2D-fractures...
2D-fractures geometry check...

Assign 2D fracture Attribute...

Delete 2D-fractures...

Export 2D-fractures...

■ Display 1D, 2D fractures

When available, the fractures are presented.

Example of a 2D fracture (direction of incidence: 280°):
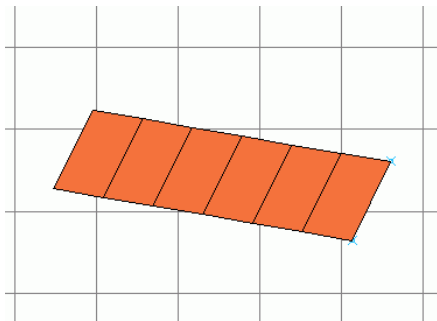
Fig. 80: Top view of a 2D fracture (direction of incidence 280°)

■ Generate 1D-fracures from element edges

The following input window appears:

Fracture parameters

Fracture parameters

Opening width   0.01

Thickness   0.1

◉ All layers
◯ Horizontal (node layer)   1
◯ Vertical (element layer)   1

OK   Cancel   Help
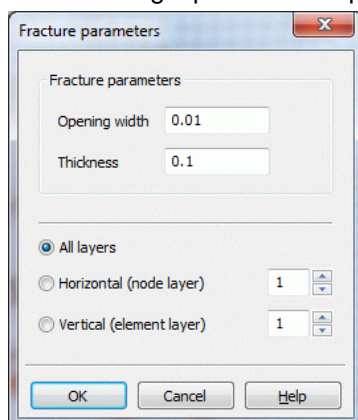
After selecting a layer or all layers fractures can be created along the horizontal or vertical edge of elements. The fracture parameters have to be set.

■ Generate 1D or 2D fractures from contours

Only when choosing this menu item, the previously generated K-contours are imported as fractures in the model. The appropriate entry in the project information window (1D-fractures: ... or 2D-fractures: ...) shows

this. The 1D-fractures can be captured, selected in an area or selected all (previously the contours have to be represented). The 2D fractures can be selected manually (previously contours represent).

Via *View → More windows → 3D-view* the fractures are displayed in space.
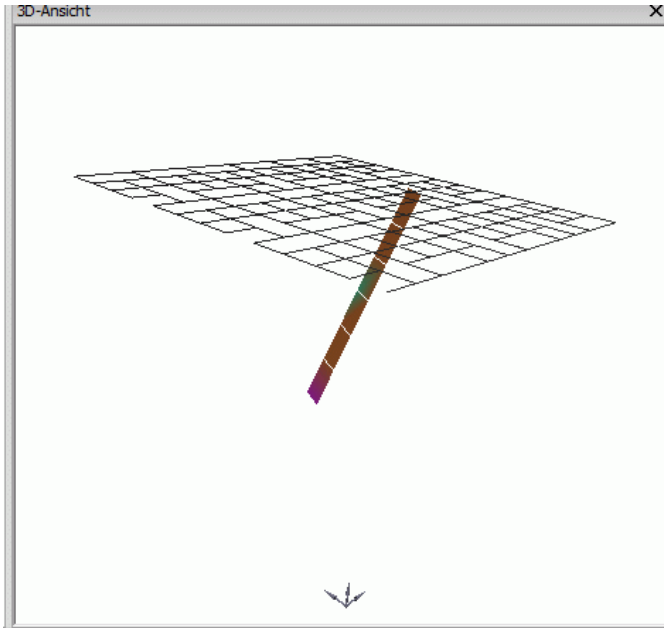
Fig. 81: 2D fracture with angle of incidence 85°

■   1D- , 2D fractures geometry check

When activating these functions, the system checks if the fractures intersect. This is currently technically not possible.

■   Assign 1D or 2D fracture attribute...

An input window appears:
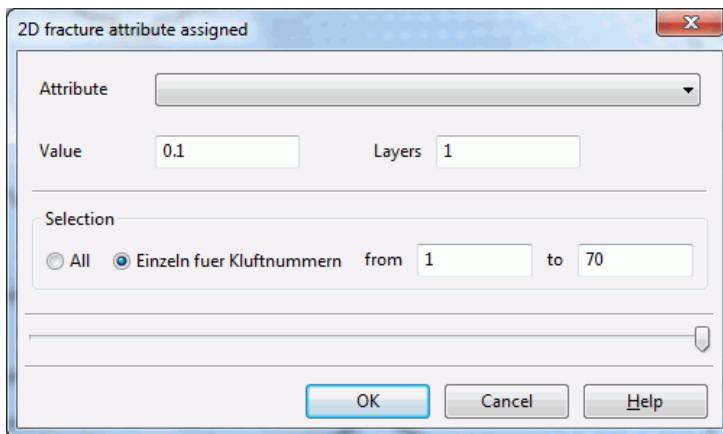
Attributes are assigned to fractures with the help of this input window.

■   Delete 1D-fractures

The 1D-fractures can be deleted manually by capturing, they can be deleted in an area or all at one time.

■   2D-fractures from LERA

This menu item contains the function "Lera2Kluft" which previously was only available as a Python script. The menu item is only active when the data type LERA is available in the model file. The operation is described in chapter: "Creating a 3D model", (p. 230).

■   2D fracture from attribute...

The following input window appears:
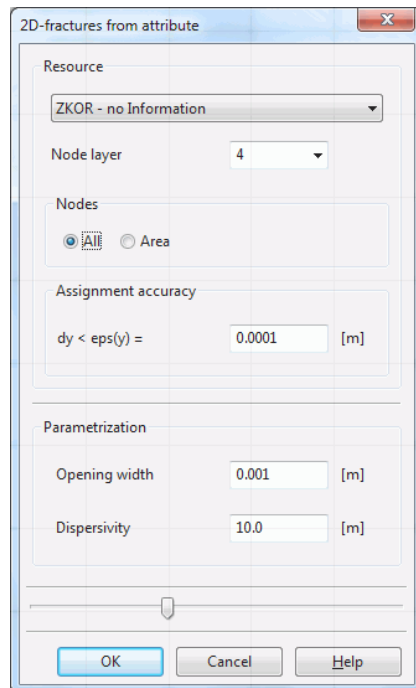
2D-fractures are generated from existing attributes.

■ Pick 2D-fractures

This menu item allows the user to create vertical or sloping fractures via the user interface. The following input window appears:
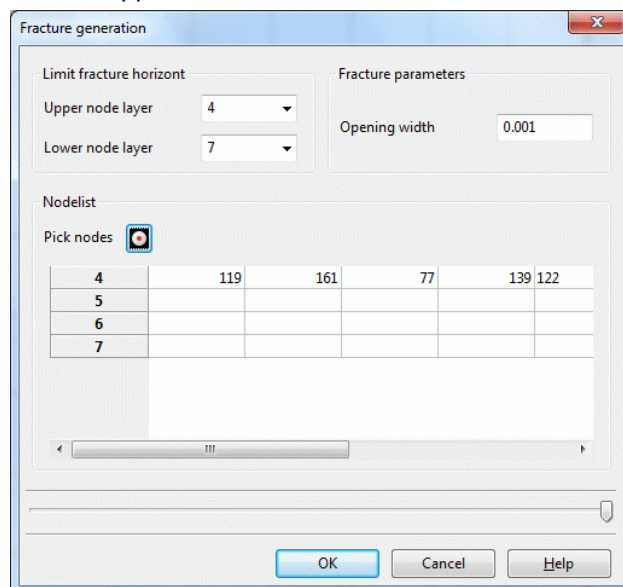
The upper and lower node layers of the fracture as well as the fracture opening width are defined in this input window. In the node list, the fracture nodes are defined manually or interactively through the user interface. The following figure shows a group of fractures thus created:
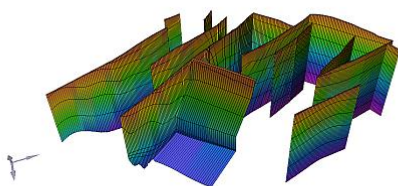
Fig. 82: Example of a group of fractures, made up of vertical and sloping fractures

- Delete 2D-fractures

This menu item allows the user to delete all fractures. There is also the option to delete 2D fractures selectively either on the edge length or the area. The following input window appears:
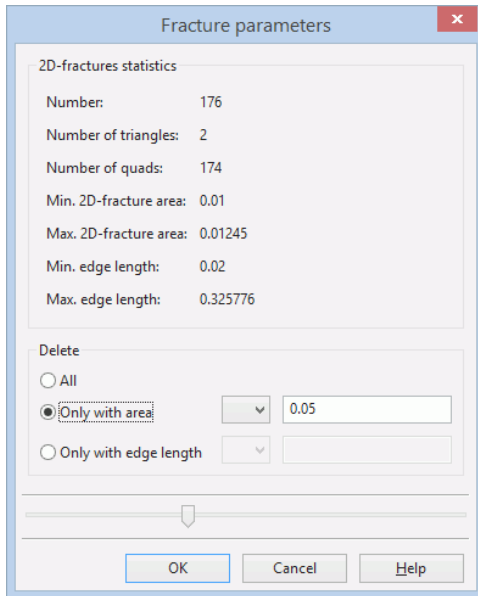
After confirming with OK is in the status bar at the bottom of the screen displayed, how many fractures were deleted with the appropriate feature.

- Export 2D fractures

A file selection window appears in which the export file for the program Tecplot is defined.

4.12.1   Transient

Creating transient structures is described in chapter: "Data structure, Transient input file from transient structures"

The following functions are selectable in the submenu:
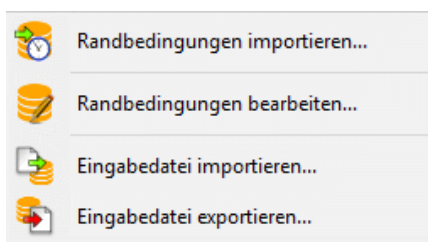
- Import boundary conditions ……

With this menu item an already existing transient database in spt format can be read in at the beginning of the model processing. This is useful, for example, if the mesh is to be refined. Then the transient data of the spt database is automatically refined.

- Manage boundary conditions……

This function is only active when a transient input file has been imported. The following dialogue box appears:
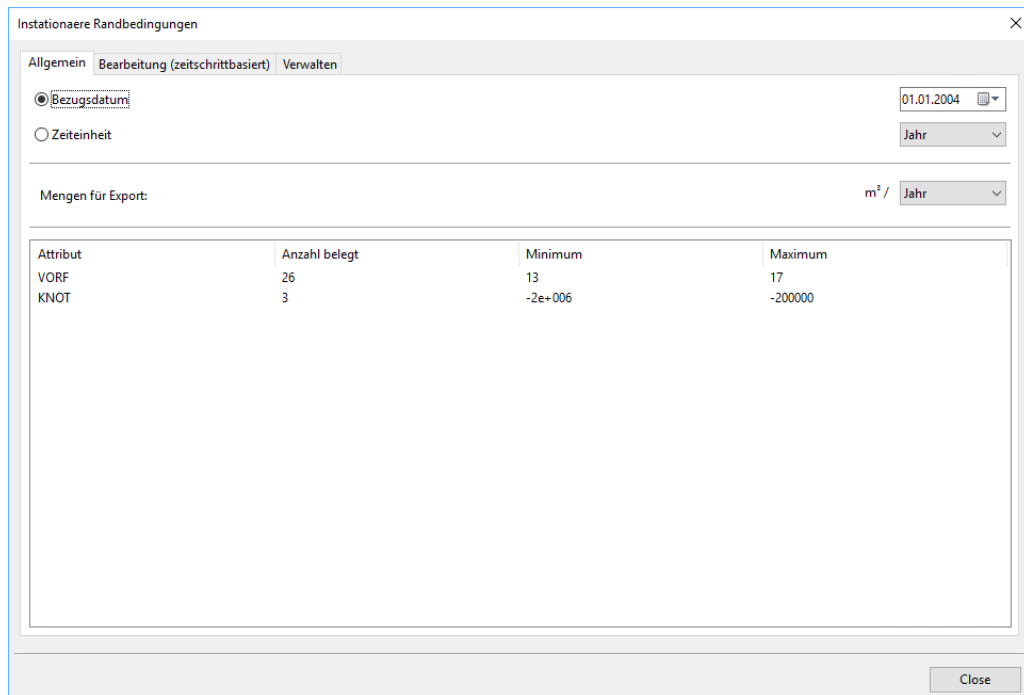
The first dialog page *General* gives an overview of the attributes in the transient file as well as their minimum and maximum values.

Here, the reference date, time and the unit of the imported transient input file are changed.

Choice of quantity unit:

The desired time unit is set which applies to quantity-based transient data. The unit of time applies only to the transient input file.

In the second dialog page *Edit (time step based)*, the transient attributes can be edited time-step-based:
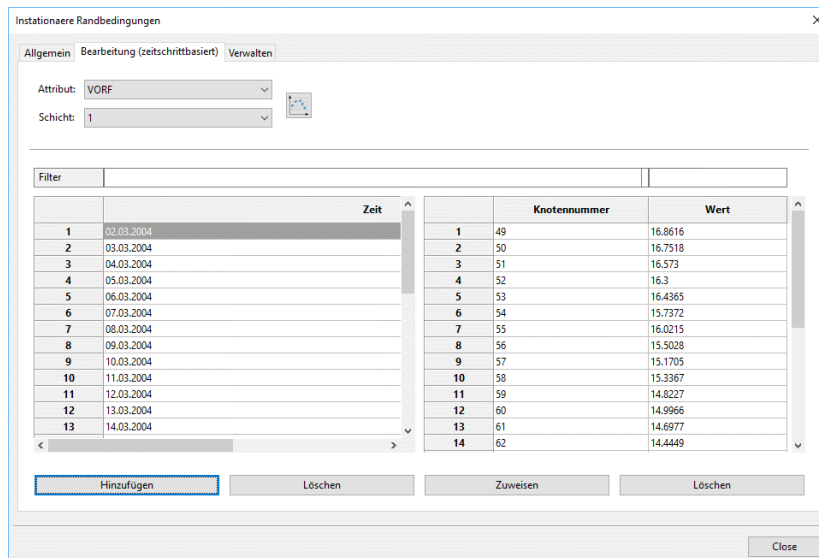
For the selected attribute, the associated hydrograph can be displayed for any node (or element, depending on the data type).

In the left-hand list, the existing dates or times are logged. Times can be inserted or deleted via the buttons below.

In the right-hand node or element list, the assigned nodes or elements are displayed with their respective value at the selected time. Via the button Assign a new node (or element) with a value can be inserted. The following dialog appears

The entries are saved in the database immediately. Analogously, values can be changed or deleted at nodes or elements.

In the third dialog page Manage, the transient attributes can be managed:

Depending on the selected action, the respective attribute can be deleted, deactivated or reactivated, e.g. layer by layer, by area or only at some time steps. The changes are immediately saved in the database.

■   Import transient input file

If so far only a transient txt input file exists, it can be read in via this menu item. It is saved as a new spt file.

■   Export transient input file

After selecting this menu item, a file selection window appears in which the user can export the edited transient database as a new transient input file in txt format for later calculations.

4.12.2   Generating tunnels

Basis for creating a tunnel is an I-structure, which level corresponds to the central axis of the planned tunnel. The height can be assigned as the attribute KKKK.

The following functions are available:

The menu item *Generate geometry-contours* ( ) is active when an l-structure is present. If this item is activated, the desired l-structure can be selected with the left mouse button, and after pressing the right mouse button, the following input window appears:
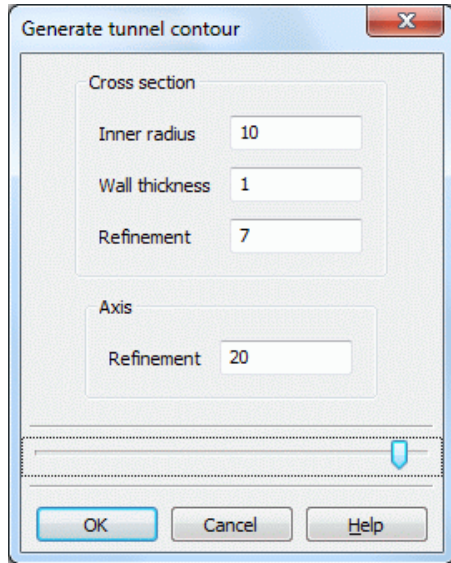
The inner radius and the wall thickness are given in m. The refinement of the cross section defines how the tunnel is shown. In this case, the tunnel cross section is shown as a heptagon. The refinement of the axis corresponds to the menu item *Contour → Edit l-structure → divide in n parts.*

Instruction for the generation of tunnels

- Create an l-structure of the central axis of the tunnel, values (attribute KKKK) at the start and end points of the structure correspond to the NN level of the central axis and are interpreted from the program as height, regardless of the data type.
- Call the menu item *Special features → Tunnel → Generate geometry-contours …*
- Select the l-structure with the left mouse button and calling the parameter menu with the right mouse button, input the parameters: The contours parallel to the central axis appear.
- Delete existing mesh in the area of the tunnel, create new nodes and elements.
- Important: Reassign attributes, especially the ground level and Z-coordinates in the new mesh area.
- Call the menu item *Special features → Tunnel → Generate geometry-contours* again, select the tunnel axis and re-enter parameters.
- Via the menu item Special features → *Tunnel → Generate tunnel* the new tunnel is created.

The result can be visualized in the menu *View → vertical section.*
The previously created tunnel-contour looks in the top view as follows (clipping):
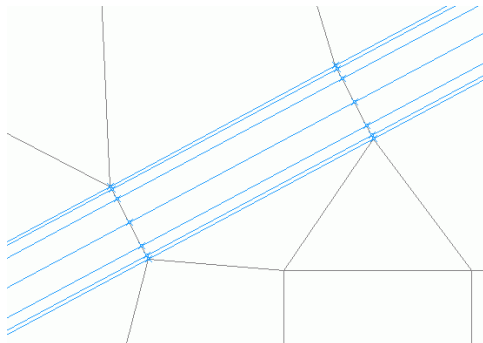
Fig. 83: Top view of an already newly connected tunnel contour

The seven lines represent the cross section refinement, the centre line is the original l-structure.
Vertical section of the completed tunnel:
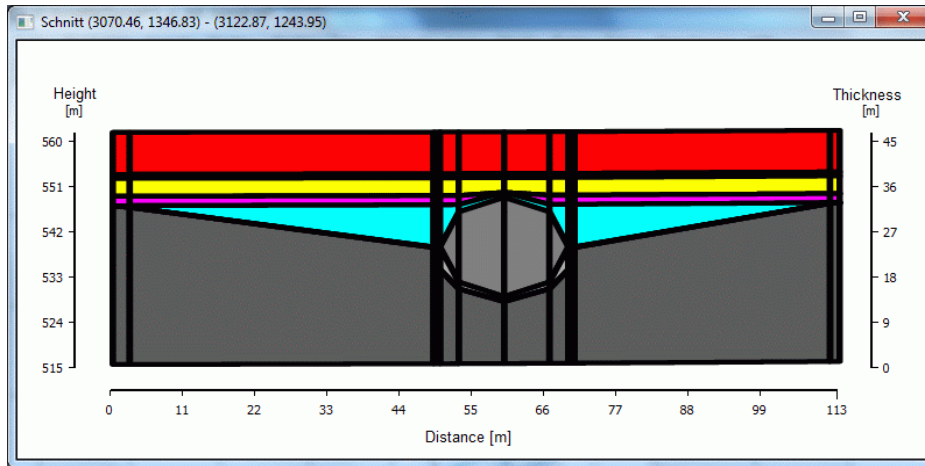
Fig. 84: Vertical section of the completed tunnel

4.13 Calculation

The calculation menu is available both in the plot and model mode. It is identically in both modes. The following menu items are available:
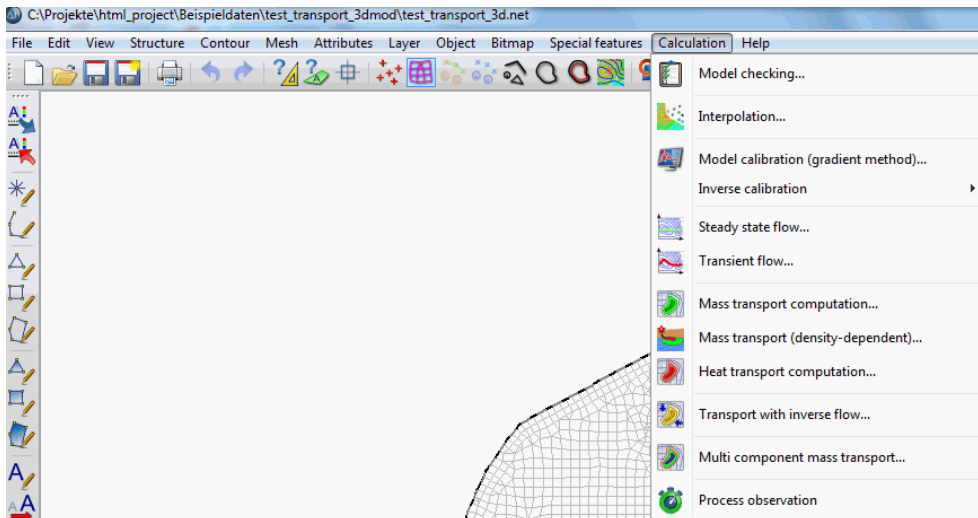
Because of the complexity of each calculation module they are described chapter by chapter.

Multi component mass transport

This calculation module is currently being developed.

Process observer

During a transient calculation, it is possible to track the time progress of the calculated results live by starting the SPRING Process Observer. A detailed description is given in chapter: "Transient flow - Monitoring the time progress" (p. 302).

4.14 Help

The help menu is available both in the plot and model mode. It is identically in both modes. The following menu items are available:
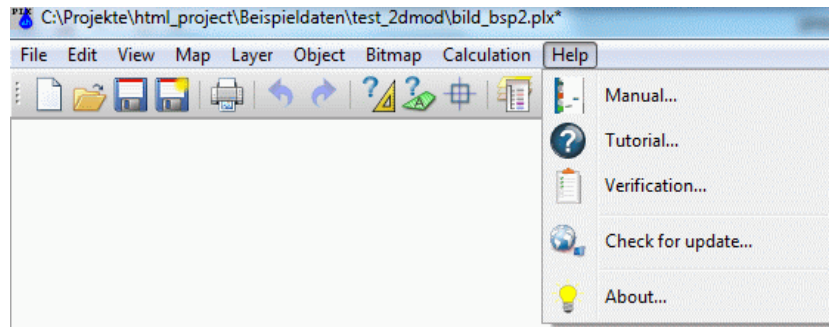
When you call the menu item *Manual…* the built-in SPRING manual is opened, the menu items *Tutorial…* and *Verification…* link to the website of delta h. Choosing the menu item *Check for update…* the program checks if a newer version of SPRING is available.
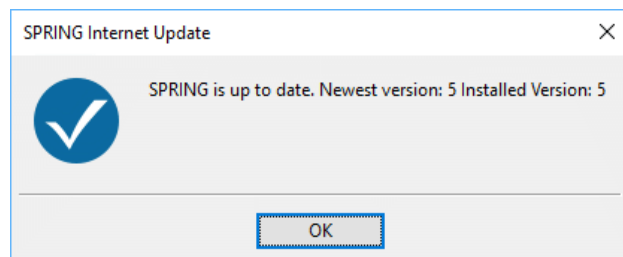
Calling the menu item *About…* the current program version and license agreement of the delta h engineering company is shown:
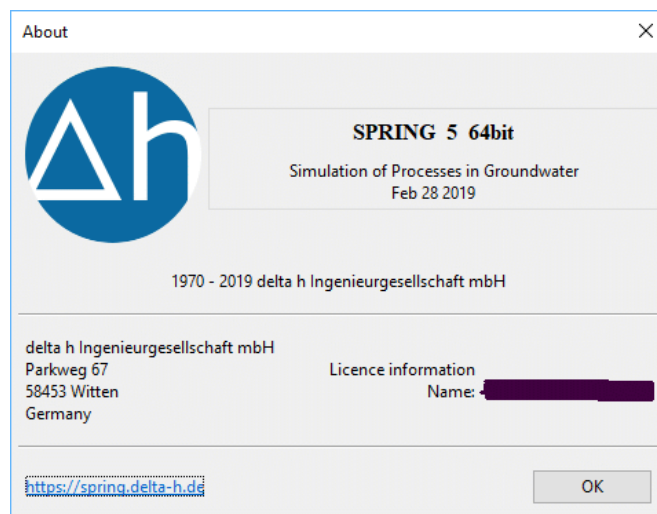
5 Compilation of Different Model Types

The following chapters show the user how to compile different groundwater models containing all the required data step-by-step. Practical examples illustrate the procedures.
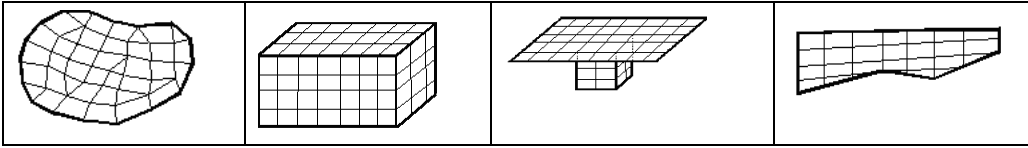
The chapter Compilation of a 2D-Model allows the user to compile a horizontal groundwater model containing all the required data step-by-step.

Before you get started with it, you should import the required sample files from our website http://spring.delta-h.de   in the menu Download&Support. You find them in the zip-file under: Tutorial (EN): „Tutorial_bsp_files/Tutorial_2d_bsp_files" ( ).

5.1 Compilation of a 2D-Model
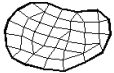
The central files in SPRING are the ASCII model files *name.net* and *name.3d*. The *name.net* file contains all the information for a 2D model or the data of the topmost layer for a 3D model. The *name.3d* file contains the information for the deeper layers. These files are usually created or modified by SPRING interactively. However, they can also be edited using any editor.

During the *model checking*, the model files are read in and written to binary background files. The subsequent calculation modules will read the data required for you from these background files and also save the results there. For the graphic visualisation of the input or results data (*plot generation*), the data are read from the background files and used for generating the relevant plot. This can be followed by having SPRING redisplay, re-edit and print out the plots on a plotter or printer.

The SPRING interface is used for compiling the model, calculating the flow and generating the plot and is

opened by double-click with the left mouse button on the icon   on the desktop.

5.1.1 Step 1: Generating an FE mesh by automatic mesh generation

Compilation of a groundwater model first requires creating a Finite Element mesh. To do so, the module SPRING is started by double-click. By clicking on *File → New*, the following window for creating a new project appears. Basic data (type of model, time unit, scale, extension in x-y direction) can be predefined here. In the example, the predefined settings are applied except for the max. y-coordinate.
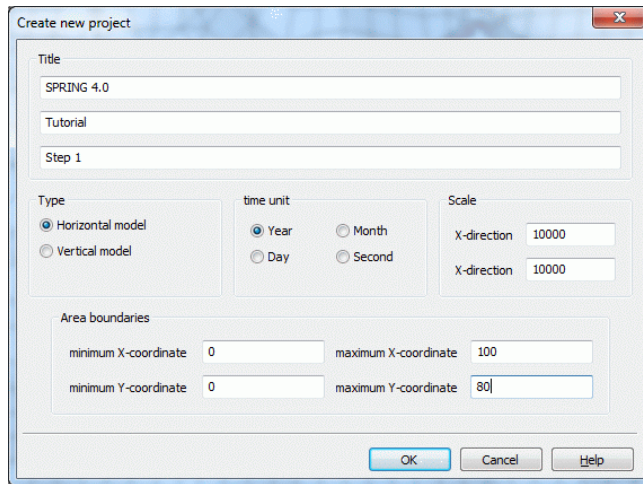
Fig. 85: Creating a new project

In the example, the maximum y-coordinate is set to 80.0 m.

After closing the window by pressing OK and double-click with the right mouse button, the program returns to the surface of the SPRING module. After clicking on *Mesh → Mesh generation → Regular mesh*, a further menu opens, in which the dimensions for the Finite Element mesh to be generated can be entered.
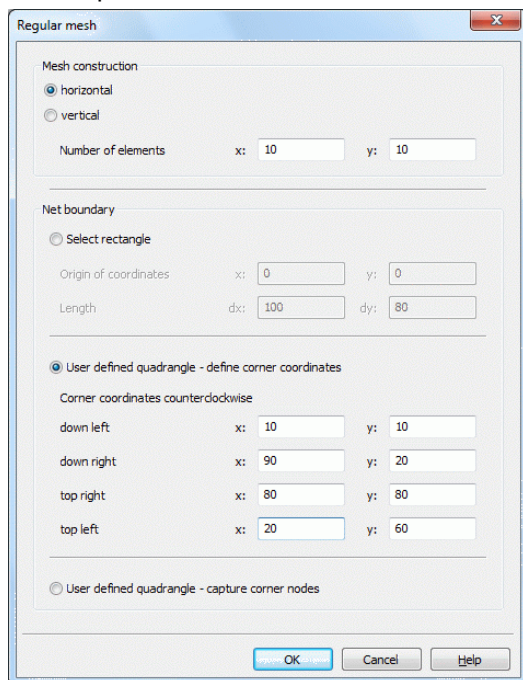
Fig. 86: Generating a regular FE-mesh

In this dialogue the number of elements in x-and y-direction and the direction (horizontal or vertical) is set. Furthermore the mesh boundary can be set in three different ways:

■   Select rectangle:

Here the origin of coordinates and the lengths in the x-and y-direction are to be specified. It is important to ensure that the selected origin and the dimensions are within the selected area when creating the model boundaries.

■   User defined quadrangle – define corner coordinates:

In this method the four vertex coordinates are set. It is important to ensure that the selected origin and the dimensions are within the selected area when creating the model boundaries.

■   User defined quadrangle – capture corner nodes:

Here, the corner coordinates of the FE mesh can be selected interactively in the GUI by selecting with the left mouse button. The corner coordinates are set counterclockwise. This feature is currently not implemented.

Upon applying these settings by pressing the OK button, the following FE mesh is generated, which can be displayed either by clicking on *View → Elements* or by activating the corresponding button ( ):
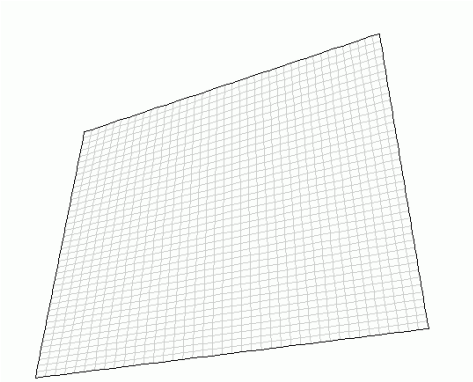
Fig. 87: Automatic generated FE-mesh

During compilation of the model, it is recommended saving the intermediate version from time to time. The first time, the file will receive a name of its own when clicking on *File → Save as* …. During the further course of the project, it is sufficient to save the project.

Now the data assignment (p. 199) would be the next step, but because the automatic mesh generation is insufficient for most hydrogeological problems, step 2 is carried out next:Generation of an FE mesh having irregular constraint geometries

5.1.2 Step 2: Generation of an FE mesh having irregular constraint geometries

5.1.2.1 Description of the investigation area

The area under investigation of the model to be generated looks as follows:
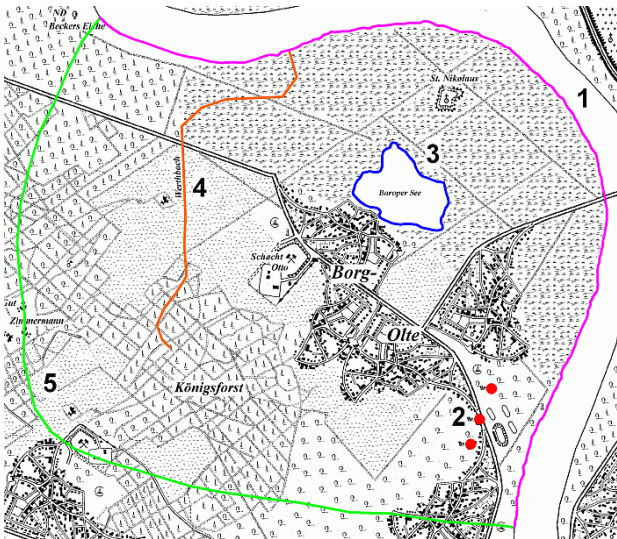
Fig. 88: Area under investigation

The geo referenced tiff file "topo1.zip" and all other necessary files can be downloaded from our homepage: under Download&Support: Tutorial (EN) "Tutorial_bsp_files.zip" ( ).

The investigation area can be described as follows:

A large-sized main water course system, into which a smaller-sized water course flows, extends from southeast to northwest. The area under investigation also contains a lake with groundwater contact. There are also three pumping wells that deliver groundwater in the southeast. Since these constraint points affect the local groundwater situation, they must be included in the area of the model.

5.1.2.2 Issues on a groundwater model

A groundwater model is created when a variety of factors influence the groundwater situation and complex boundary conditions are in place and an analytical solution of the problem is no longer possible.

Possible questions for a groundwater model include:

- How is the groundwater flow affected by a change in the withdrawal amounts?
- Which depths of groundwater level does it have?
- What is the impact of subsidences caused by mining on the present depths of groundwater level?
- What are the consequences of a flood wave and how change the flood plains?
- What impact do interventions in the water course have: stream relocation, changes to the stream bed?
- What happens when constructing new production wells and what is their effect on the existing situation?
- etc. etc. etc.

These and many other questions can be answered by a groundwater model.

5.1.2.3 Data retrieval

The following parameters are required as data basis for the following calculation:

Main water course system: Water levels in m (in most cases m above sea level) at least at two points: fixed potential boundary → POTE

Secondary water course system: Water level in upper reach (m), if required, discharge measurements for determination of leakage → VORF and LERA

Withdrawal wells: Withdrawal amounts → KNOT (negative sign, convert in m³/time unit of the mesh file)

Groundwater recharge: Precipitation, soil type, soil use, gradient for calculating recharge → FLAE(mm/time unit of the model file)

Geology: Permeability values in m/s → KWER

Ground level surface: → GELA, and, if required, data on an impermeable layer in the underground → UNTE, otherwise calculation using the thickness

Thickness: →MAEC

Measured values of the groundwater for determining the calibration status: Groundwater surface → EICH

Remark: The time unit of the model file is set when you create a new project (*File → New*).

These data can be delivered by the following sources:

- Ground level elevations and TIFF-files: Land Survey Office, Land Registry
- Groundwater measuring points, levels on water courses: Federal environmental office, water boards, water works, etc.
- Withdrawals: Well operators
- Geology: geological maps, exploration drillings, pumping tests
- Leakage: for example calculation via K values of the surroundings and discharge measurements
- Groundwater recharge: for example on the basis of RVR data, according to Schroeder and Wyrwich or Meßer

5.1.2.4 Determining the model boundaries

Before a new project can be created, the area of investigation boundary of the FE model is determined. For this purpose, first the model boundaries are set.

In the western and southern area under investigation, there is a divide, i.e., a Q=0 boundary can be defined here as boundary of the model. The main receiving water course in the north and east is assumed to another model boundary.

In the chapters that follow, the area under investigation will be reproduced step-by-step in a groundwater model.

5.1.2.5 Creating a new project

As described in Step 1 (regular mesh), for the creation of a new project the minimum and maximum coordinates of the study area are required.

Using *File → Open → Data type selection Bitmap (*.tif)* you open the already saved file "topo1.tif" (directory: „…/Tutorial_bsp_files/Tutorial_2D_bsp_files/topo1.tif").

The coordinates of the mouse pointer are displayed in the status bar of the screen.

The display area of the map can be moved by pressing the middle mouse button. By rotating the mouse wheel can be zoomed into or out of the presentation. Pressing the right mouse button takes you back to the long shot.

The coordinates of the topographic map and the selected model boundary can thus be determined for
 the lower left corner of the map to x = 800, y = 120 and for the upper right corner of the map to x = 7800 y = 6100.

To return to the mode model file SPRING is opened again. In the dialog *File → New* the three comment lines and the coordinates determined for the boundaries are entered:
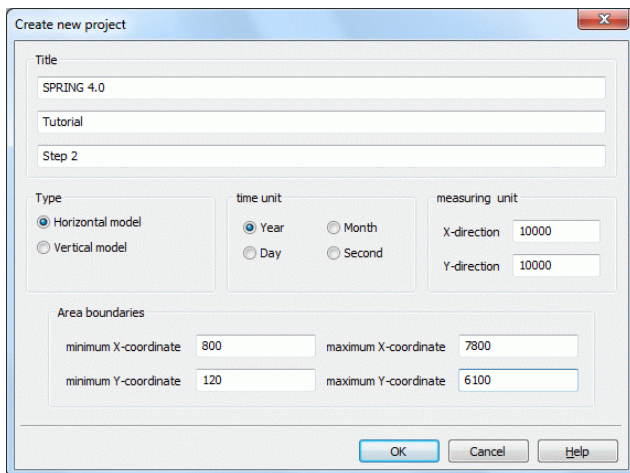
Fig. 89: Coordinates of the new project

After confirming with the OK button the project is created and should be named with *File → Save as….*

5.1.2.6 Creating and importing structures

Normally the user must read in these data him-/herself via a geoinformation system or digitise them on-screen (e.g. via *Structure → New → Points* on the basis of a Tiff-file).

In chapter "SPRING Menues - Structure" on p. **101** possibilities of creating and importing structures are described in detail.

The necessary model structure data are already digitized and are available in the already downloaded directory under *…./Tutorial_2D_bsp_files/s1_daten1.str*. Via *Structure → Import… →* (file selection) **.str* the file "s1_daten1.str" can be read.
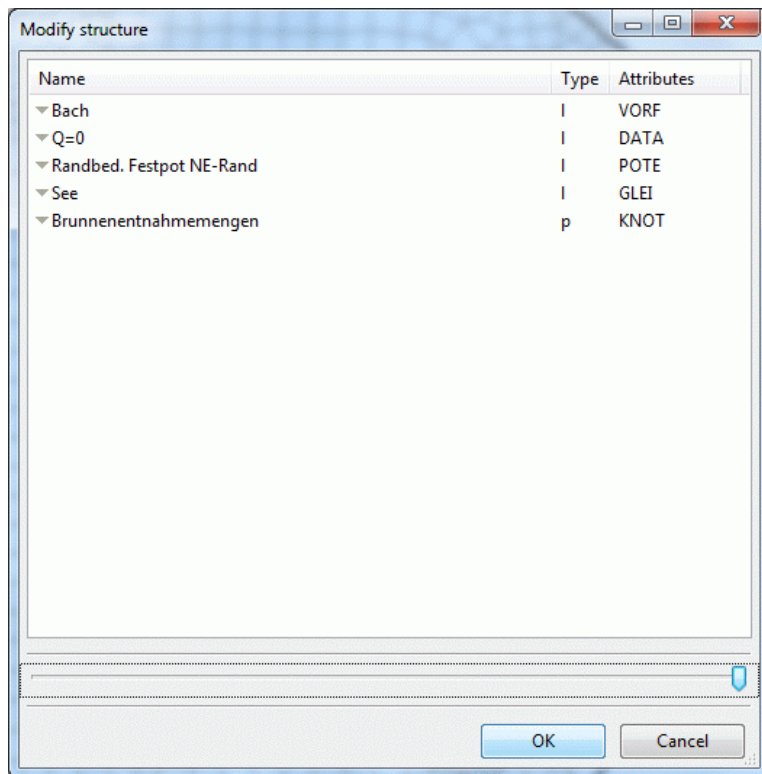
Fig. 90: Imported structures of the file s1_daten1.str

If it does not yet contain any nodes or elements, the project must be saved and opened again, before the structures can be displayed.

After activating the display under *View → Structures → All* or pressing the corresponding button

( ), the following figure appears:
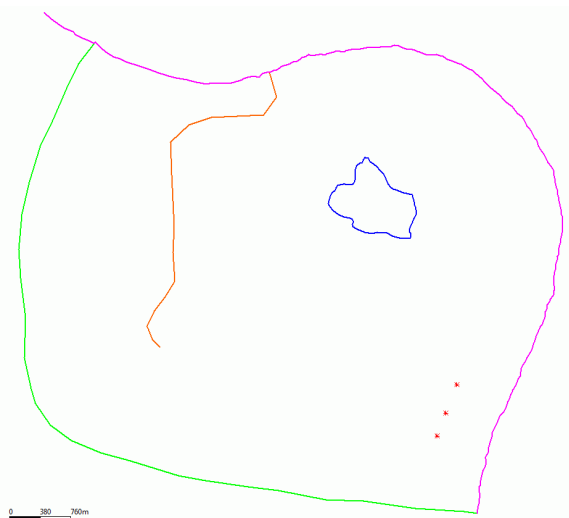
Fig. 91: Existing structure data

5.1.2.7 Modifying structures

Some structures have been data obtained which get lost by a processing (e.g. delete points of the structure).

To prevent this all structures are copied (*Structure → Copy*) which should be used later as contours and which possess structure points outside the model boundary.

It is recommended to rename the copied structures (*Structure → Edit*) and to add the words "for contours" in the information text.

The copied structures can now be edited (*Delete points, Move or Add*), without losing valuable data.

Via *Structure → Edit → Pick* or activating the corresponding button in the symbol bar (⟨🖑⟩) the desired structure can be selected. First, the northeastern model boundary (pink) is selected. The following dialog appears:
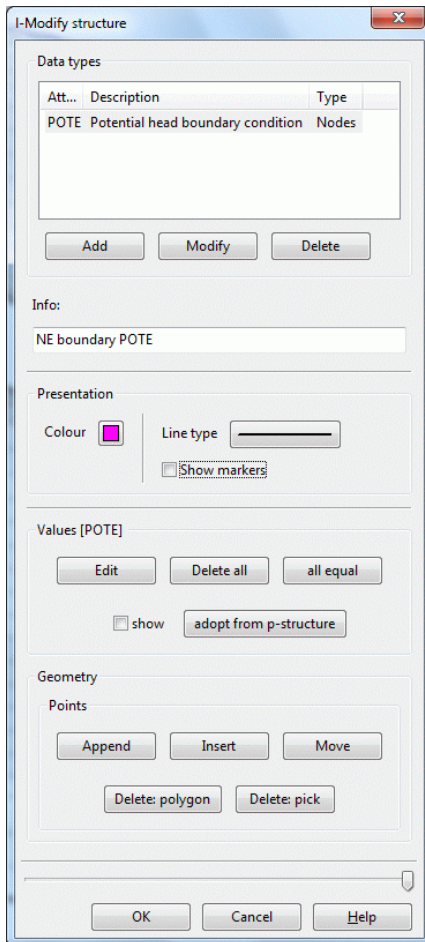
Fig. 3: Modifying l-structures

After *Show marker*, the superfluous structure points at the western end of the structure can be eliminated via the following items: *Delete: Pick or Delete: polygon.* The actual deletion of the points will not take place until the structure menu is finished by pressing OK.

You should also check whether the endpoints of the north-eastern and south-western model boundary coincide. If this is not the case, the endpoints can be moved toward each other by using the menu item *Points: Move* of the *Structure* menu. In the selection window, you will be asked whether the point is to be moved "free" or onto a "Point". Select "Point".

In this menu, it is also possible to append or insert points into the existing structures. This is required for the watercourse located within the model (orange). This structure should be attached to another point that lies on the main water course. In the structure of the main water course should then be a further point inserted at this point (Capturing mode: Point).

5.1.2.8   Generating contours

All input data are kept available in the structure data. Not all structure data are automatically constraint geometries for the FE mesh (e.g. measuring points or geological explorations outside the area under investigation). Only all constraint geometries of the structures are transferred. Contour editing is the basis for a homogeneous FE mesh. The individual structure points can be visualised while they are still in the Structure menu by clicking on *Marker → Show*.

Initially not all structure points of the north-eastern boundary should be transferred as contour points.

This is why the distance of the contour points is increased by clicking on *Contour → New → From structure* and then entering the distance in the input window that appears:
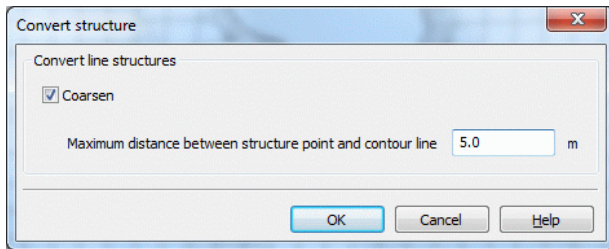
Fig. 92: Converting structures into contours

The maximum possible distance to the original boundary is set to 5 m. To generate a regular distribution of the boundary nodes, the distance of the contour points set to 300 m by clicking on *Contour →Edit I contour → Define maximum distance → Polygon.*
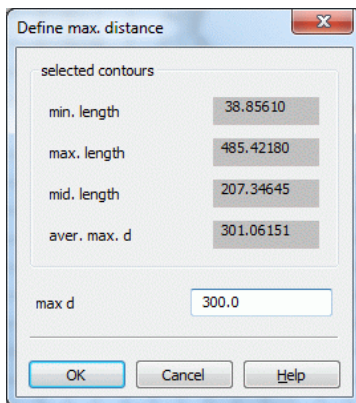
Fig. 93: Setting the maximum distance

This corresponds approximately to the distance of the structure points of the southwestern boundary. The same procedure is recommended for the lake, although in this case a smaller distance of the contour points should be selected (approx. 35-50 m), in order to obtain later on a smaller element edge length.

The remaining structures are included in the list by clicking on *Contour → New → From structure → Capture / List*. When selecting the list, you can also convert several structures simultaneously by pressing the CTRL key and selecting the desired structures simultaneously.

At the south-eastern structure boundary and at the water course, the distance of the points can be reduced for individual contour elements. This is done via *Contour → Edit I contour → Define maximum distance → Capture.* This will select the contour elements that seem "too large" to the user. An input window appears in which the desired maximum distance can be entered.

At the point where the water course flows in the main water course system, the contiguous contour of the main water course must be divided once (*Contour → Edit I contour → Split into n parts: 1*), to ensure that the contour point of the water course flows in at the division point (*Contour → Move points*).

This gives the contours shown in the following figure:
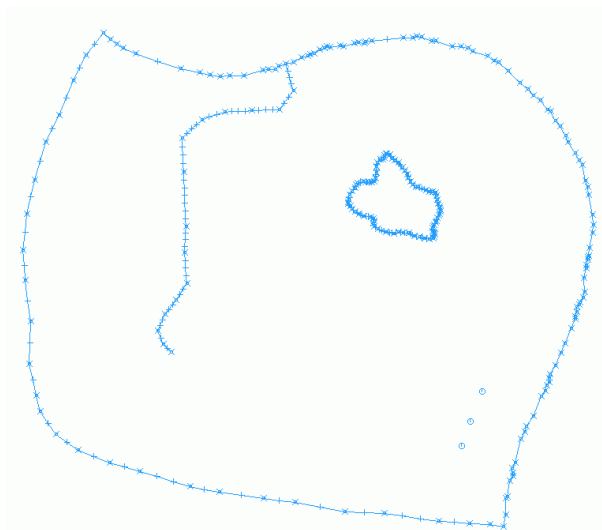
Fig. 94: Existing contours

For better visualisation of the logarithmic cone of depression around a production well, the contour points of the wells must be assigned suitable well parameters. This prevents oscillations caused by local changes in potential in the proximity of the well. After *Contour → Edit p-contour → Well parameters*, the following input window appears:
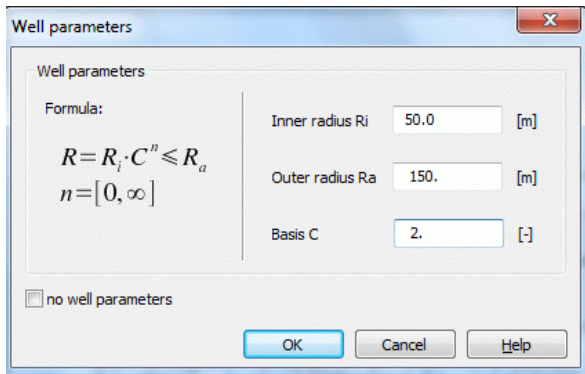
Fig. 95: Setting well parameters

For a detailed description of the parameters, please refer to chapter: Data structure of the groundwater model: Local discretisation: Special case: Well discretisation.

Before node generation, you have to check whether the contour boundary is closed, and the distances of individual contour points are not too small. This is done via *Contour → Optimise*. The setting of the accuracy parameters is confirmed by pressing "OK". If the check is free of errors, the comment "Boundary closed" appears at the bottom edge of the graphic interface. If errors occur, they are displayed and must be eliminated by changing the accuracy parameters.

Note: For these contours as well, the following applies: When the project is reopened, the contours must be displayed by clicking on *View → Contours* or the corresponding button ( ).

5.1.2.9   Generating nodes

In node generation, first nodes on and in the area of the contours are generated by clicking on *Mesh → Mesh generation → Generate boundary nodes → Global*. The number of passes establishes how many node rows are to be generated in parallel to the contours. 2-3 passes are recommended. Selected in the example: 3 passes.

The contour-free areas are then filled with regular meshes by clicking on *Mesh → Mesh generation → Generate Raster → Global*. The approximate node distance is defined in the following input window:
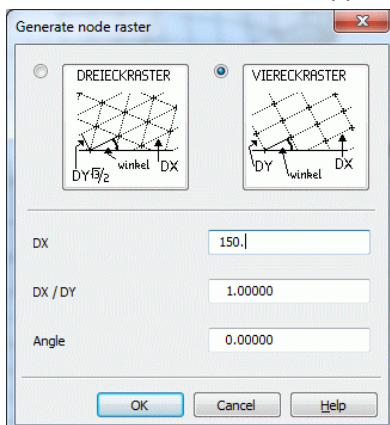
Fig. 96: Generating raster nodes

In the *DX* box, the average node distance of already generated nodes appears as default. It can be modified by the user. In the example, DX is set to 150.0 m. Next the generated regular mesh is displayed.

You now have to check whether the position of the individual nodes produces a homogeneous pattern. Nodes can be added to voids (*Mesh → Mesh generation → Insert nodes*), too many nodes can be deleted (*Mesh → Nodes → Delete*) or nodes can be combined from a defined distance by clicking on *Mesh → Node → Merge → According to distance*.

Note: To zoom, the current function (e.g. "Insert nodes") is finished with the right mouse button. The desired zoom level is drawn with the left mouse button. Press the F12 key, to reopen the current function. (This new feature applies to all editing functions!)

5.1.2.10 Generating elements

Automatic generation of elements is done by clicking on *Mesh → Mesh generation → Generate elements → Triangles and quadrangles*. To release the mesh, for example, twenty times, in order to obtain optimum angles between the element sides, press *Mesh → Unbend ….*
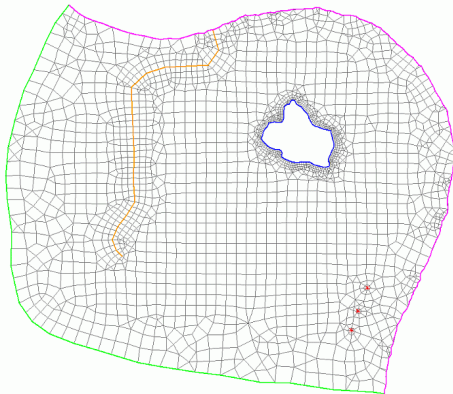
Fig. 4: Finished FE-mesh

In the above picture, for the sake of clarity, nodes and contours have been hidden (*View → Node* or *Contours* or press the corresponding buttons).

The markers on the structures were deleted by pressing *Structure → Edit → (→ Click on structure*) Show *Marker* or by deactivating the Marker button.

Finally the nodes and elements inside the lake are deleted via *Mesh → Nodes → Delete → Area*, because the lake is treated as an internal boundary in this model.

The last step in mesh generation is to carry out mesh controls (*Mesh → Checks…*) and eliminating the errors that may have occurred, before step 3: Data assignment is started!

The finished mesh and the structure file are available to download on our   or was already downloaded (…/Tutorial_bsp_files/Tutorial_2D_bsp_files/schritt2_tutorial.zip). It can be used for further processing.

5.1.2.11 Using the external TRIANGLE mesh generator

It is also possible to generate a mesh by means of the external Triangle mesh generator. The advantage of the mesh generator is that the user can predefine, for example, the minimum area or the minimum angle of a triangle element, in order to comply directly with form factors for the element quality (ratio of internal circle to external circle).

Before that, the program Triangle must be installed on the computer (http://www.cs.cmu.edu/~quake/triangle.html).

Since Triangle is not integrated directly into SPRING, SPRING must first create input files for Triangle and then start it as an external program. The output generated by Triangle must in turn be read in by SPRING. This will overwrite the existing mesh information.

To open the relevant dialog, go to the menu item *Mesh → Mesh generation → Triangle mesh generator*. Another menu opens:
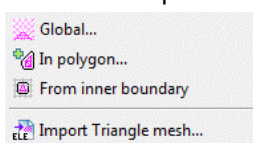
With the help of the menu item "Import Triangle mesh" a mesh (*.ele) generated with the external mesh generator Triangle can be imported.

After selecting one of the first three menu items (Mesh generation globally, in polygon or from an inner boundary) opens the associated dialog:
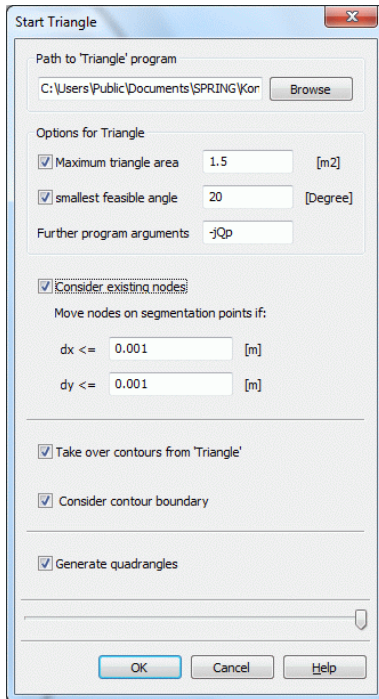
Fig. 97: Dialog for Triangle mesh generator

"Path for the program Triangle":

Here the path is selected under which Triangle has been installed. During the SPRING installation it is installed in the folder "C:\Users\Public\Documents\SPRING\Konfig" by default.

Note: In the operating systems Windows Vista or Windows 7, the file "triangle.exe" must NOT be present in the folder "C:\Programs\SPRING\bin", as there is no writing authorisation in this folder.

"Maximum triangle element": (Triangle Option "-a")

This option makes Triangle generate additional nodes, preventing the generation of triangles whose area is larger than the specified value.

"Smallest feasable angle": ()

This option makes Triangle generate additional nodes, preventing the generation of angles smaller than the specified value. The default is 20°. This option generates high-quality meshes, but can result in Triangle not being finished correctly.

"Further program arguments": ()

Triangle offers many more options of influencing mesh generation. The corresponding parameters for loading Triangle can be entered here.

"Consider existing nodes":

This writes the existing nodes as additional constraint geometries into Triangle's input file. Thus, for example, first a regular mesh can be generated using SPRING's regular mesh generator, which is then meshed by Triangle.

"Take over contours from Triangle" and "Consider contour boundary":

If Triangle generates additional nodes, in order to comply, for example, with the angle criterion, these new points will also be generated on the constraint geometries below (contours). The option "Accept contours from Triangle" makes SUSI, after a successful Triangle run, read in not only the mesh geometry, but also overwrite the existing contours with the one modified by Triangle.

"Generate quadrangles":

This will start SPRING's quadrangle generator, following a Triangle run.

5.1.2.12 Inserting a new structure in an existing model

In this chapter, a concept for later incorporation of a new structure in an already existing mesh is presented to the user.

Before starting with the edition of the mesh geometry, it is advisable to make copies of the original mesh and structure file. All attribute data affected by deletion of an element area should be saved by clicking on *Attributes → Export*. A list of all available attribute data is displayed. By selecting the individual attributes, they can be saved as independent "KENN.str" files and then assigned again later on to the modified mesh (interpolated).

Attention! When working with the 3D model, do not forget the data of the 3D file!

After the preliminary work, the stepwise conversion of the model file is started.

Generating the new structure:

Structure → New → Lines (at the water course or sheet piling wall)

To visualize a sheet piling wall, it is possible to automatically generate an identical second structure in parallel to the first structure. The distance between the two structures is freely selectable. To do so, select the structure under the menu item *Structure → Generate parallel → Catch/List* and determine the side on which the parallel is to be generated with the left mouse button (for area structures, click it on the inside or outside). An input window appears in which the desired distance is entered:
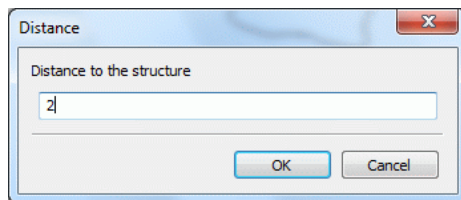
Fig. 98: Distance between parallel and structure (m)

After confirmation with right mouse click, the structure appears in the mesh.

The following picture shows a model detail of the new structure along with its parallel (elements and nodes have already been deleted):
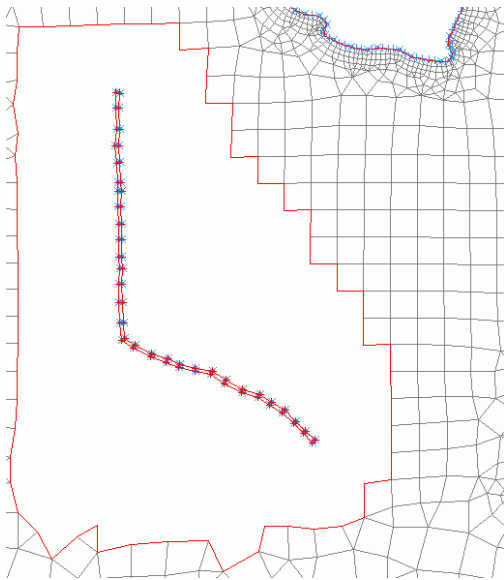
Fig. 99: Model with new structure and already assigned contour (blue)

(To make the parallel visible, a distance of 20 m was entered here.)

Deleting elements in the wide-space area of the new structure:

Mesh → Nodes or Elements → Delete → Area

To apply the new structure as contour:

Contour → New → From structure → Capture

Generating new mesh nodes:

Mesh → Mesh generation → Bitmap generation→ Area: *Catch mesh nodes:* Here it is important to catch the nodes of the existing boundary! The node distance dx should be approximately equal to the existing side length of the elements (in the example, dx = 120.0 m). For neat element grouping, some nodes are inserted manually.

Note: If a closed contour boundary is present, nodes can only be generated within this contour boundary.

Example: An area in an existing model is deleted, for example, to insert a new building. For this building, a closed contour will be created. Now, it is only possible to generate nodes within this contour. Only if the contour is deleted, nodes can also be generated outside the building.

Generating new elements:

Mesh → Mesh generation → Element generation→ Triangles and quadrangles from inner edge: The area in which new elements are to be generated can be selected immediately by a single mouse click.
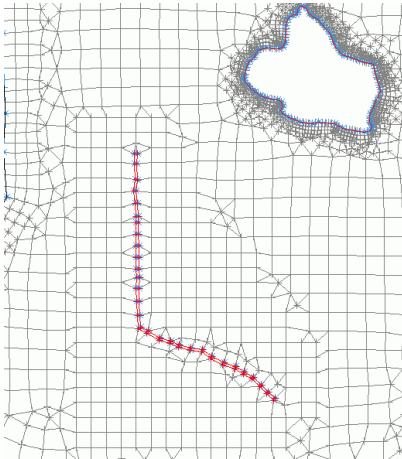
Fig. 100: New nodes and elements in the area of the new structure

After completion of the mesh, geometry checks have to be carried out, and the data must be saved!

In the area of the newly generated elements, gaps have been formed in the database.
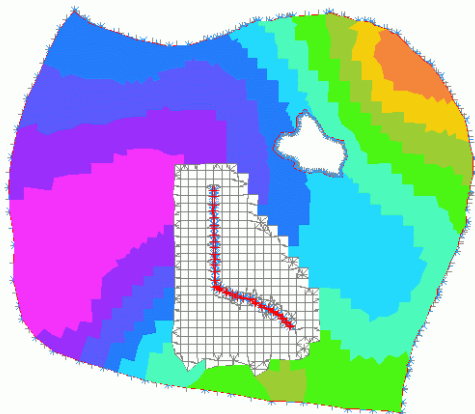
Fig. 101: Missing data in the area of the new elements, using the attribute "UNTE" as an example

The missing data must be completed by new assignment of the attributes.

Attributes → Assign → By interpolation: In the case of large data volumes the saved attribute files must be reduced to the area in question before the new assignment. This is done via *Structure → Import → *.txt*, allocation of the columns and attribute code, and then *Structure → Edit → Delete points:* in the area. Pressing the "Invert selection" button in the Catch mode is sufficient for selecting the area to be assigned, while the remainder will be deleted.

When prompted for the interpolation algorithm, you should select "Gauss", if possible, "Include existing data" and "Filter data with small distance".

Note:

If a local mesh modification has changed the value of only a few nodes, the following procedure is recommended:

If the existing node data of the unmodified nodes are sufficient for interpolation, interpolation without an ASCII file and without additional data from structures is in general only sufficient with "Include existing data". Even when the number of nodes is large, using the Gauss algorithm is suitable for the node data, since prior to interpolation all existing node data that are not in the vicinity of the nodes to be interpolated will be deleted from the interpolation points.

If, due to local mesh modifications, element data (e.g. recharge rates or permeabilities) that are constant within areas must be assigned new values, the following procedure is recommended:

The values can be interpolated without using further ASCII files and without additional data from structures only if the option "Include existing data" is used. By using distance weighting, setting a very large search radius and setting the number of minimum and maximum interpolation points to 1 (exactly one interpolation point), you can ensure that the corresponding elements receive the value of the element assigned with data closest to them.

5.1.3 Step 3: Data assignment

In chapter „Data structure of a groundwater model – Description of the data types" on p. **31** all attributes of a SPRING groundwater model are described. To start a flow calculation, at least the following data are required:

- Permeability values for all elements: attribute KWER [e.g. 0.0001 m/s]
- Thicknesses for all elements: attribute MAEC [e.g. 10 m]
- Fixed potential head at minimum 2 nodes (potential slope): attribute POTE[e.g. 5-1 m NN]

5.1.3.1 Direct assignment of existing structure data

After completion of the FE mesh, the required model data are assigned. They include, for example, the fixed potentials as boundary condition of the 1st type (POTE), water course potential levels and leakage coefficients on the courses of streams (VORF and LERA), groundwater recharge rates (FLAE), permeability values (KWER), withdrawals from wells (KNOT) or areas with identical potentials (GLEI).

Please note that data required in the entire model area, such as permeability values, lower boundaries or ground level elevations, must be assigned by interpolation, to ensure that all elements or nodes are completely assigned with values.

Data required only locally or in the form of lines, such as leakage coefficients, withdrawals from wells or fixed potentials at the model boundary, can be assigned directly.

The existing structure data file *schritt2_netz.str* contains the data types POTE, VORF, KNOT and GLEI, which are assigned by the following procedure:

Upon clicking on *Structure → Assign → List* or *Attributes → Assign → From structure…→ List*, the following menu appears:
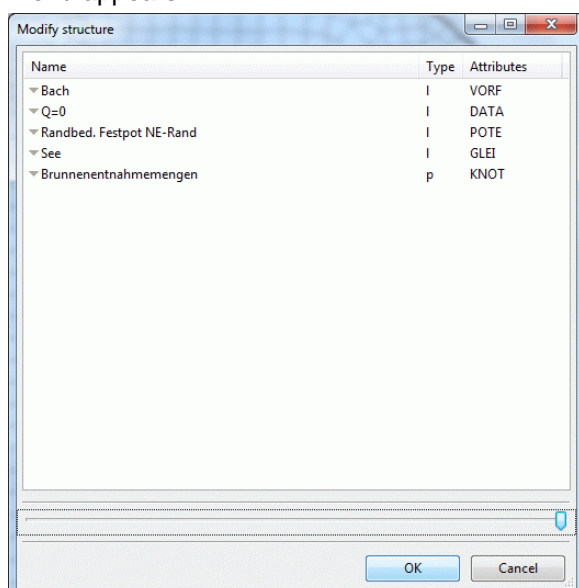
Fig. 102: Assigning structure data

Once all structures have been selected by left mouse click and pressing the Up key and confirmed by pressing "OK", another menu appears in which the accuracy parameters of the assignment can be defined. (Note: The data type DATA is always ignored during the assignment.)
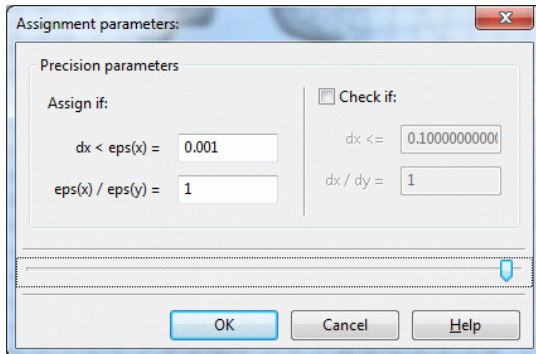
Fig. 103: Assignment parameters

If no further entries are made, the data will be assigned, after pressing the OK button. To click on individual nodes for which the attribute POTE appears at the northeastern edge, the attribute VORF appears at the nodes of the course of the stream, the attribute GLEI appears at the nodes of the lake and the attribute KNOT appears at the three wells, use the menu item *Attributes → Edit nodes → Capture.*

Upon visualisation of the potentials under *View →Show Attributes → Isolines/area plots/values*, using the "Circles" display type,
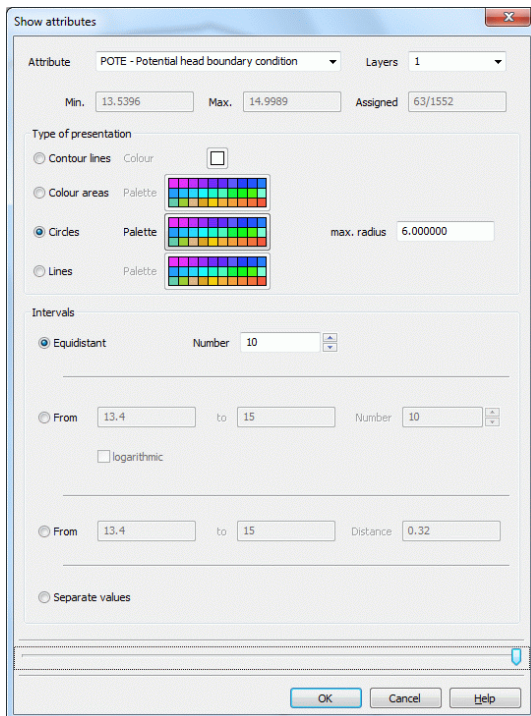
Fig. 104: Display of attributes

it becomes apparent that not all nodes of the north-eastern edge have a potential value. This is due to the accuracy parameters of the data assignment!
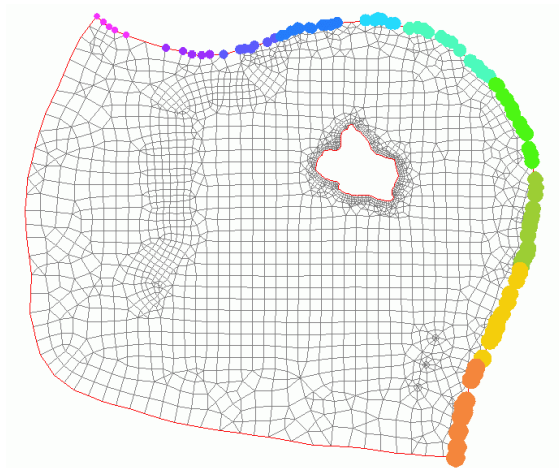
Fig. 105: Display of the assigned (not yet complete) potentials

Reassignment of POTE from the structures (*Structure → Assign → List)* with the following parameters:
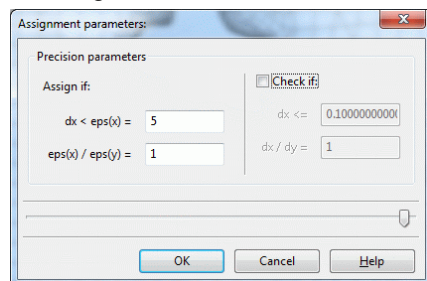
Fig. 106: Entering the accuracy parameters

and confirmation of the prompt "Overwrite all" assigns all nodes of the north-eastern boundary a potential value! (For control, the display can be repeated as described above.)

For the GLEI data type lake boundary nodes, the completeness of the assignment should also be checked. By default, GLEI cannot be displayed with circles. Using the detour via a different data type (e.g. 1KON, KKKK), it can be displayed in spite of that. Use *Attributes → Copy → Attribute-wise* to copy GLEI, for example, to KKKK and then display it. After control, the auxiliary data should be deleted again.
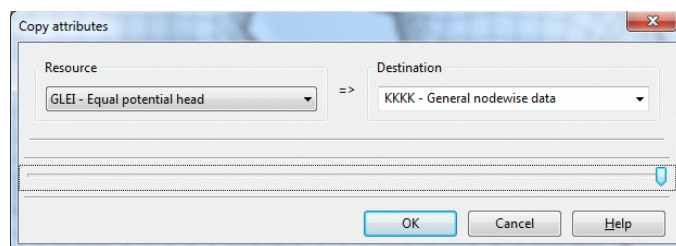
Fig. 107: Copying attributes

5.1.3.2 By interpolation of external data formats

It is also possible to read structure data from "non-project" files (external format") into an existing project. These files do not have to be present in the *.str format. Different formats can be read in by clicking on *Structure → Import ... → .str, .txt, .csv, .arc, .shp.* In the example, the structure data file s3_daten2.str was selected. (It was already downloaded from our).

By saving the project (if desired, under a new name), the "old" and "new" structure data are written to the existing structure data file.

The new displayed structures are: Depth data of the lower edges of the model (UNTE), a MARK of the course of the stream and of the line-related leakage values of the course of the stream (LERA). Whereas the attributes MARK and LERA can be assigned directly as described in Direct assignment, the attribute UNTE must be interpolated in the present example element by element.

The entries for the individual interpolation methods are described in detail in chapter Calculation - Interpolation.

Via *Attributes* → *Assign* → *By interpolation*, the following menu appears:
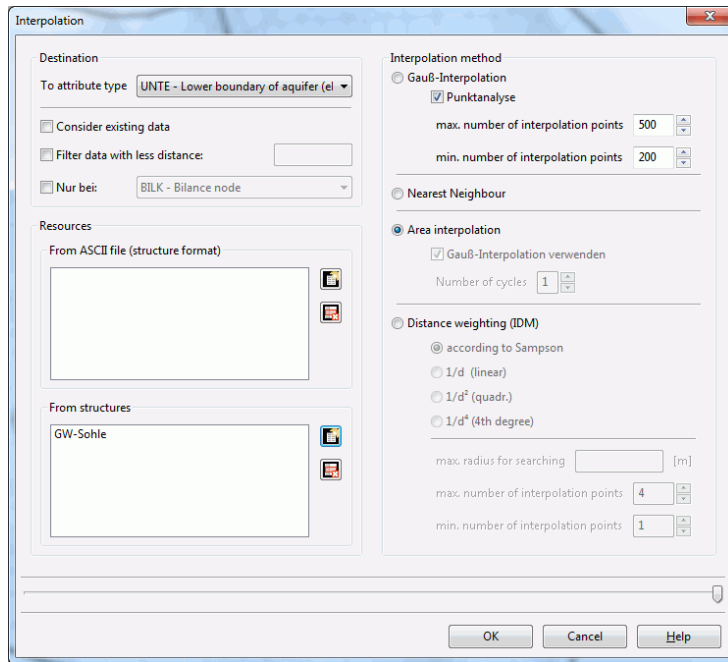
Fig. 108: Interpolation of element data

When selecting *Sources*: "*From structures*" *Add entry* (button), another selection menu appears in which all existing structure data are listed. "UNTE" is selected. The results data type selected is also "UNTE". After pressing the OK button, the data will be interpolated and, after successful calculation, the following control window appears:
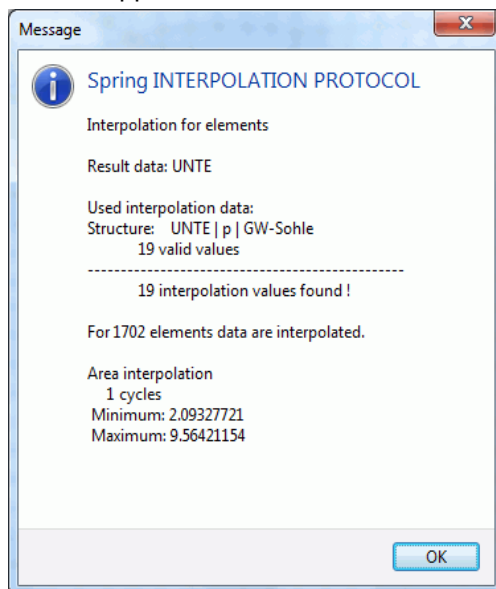
Fig. 109: Interpolation protocol

The interpolation of the calibration potentials must be performed with particular care, since the calibration potentials are one of the most data types for calibrating the groundwater model. The groundwater measuring points in the area under investigation usually contain measured values at the time of calibration which have been saved to a file. Information regarding water levels of the main water course (POTE) and of the secondary water course system (VORF) is also known. These three data types are interpolated to give a calibration state which, after its first display, must be revised manually.

An example of a required revision are interpolated groundwater isolines which show infiltration into groundwater where there is none. This situation is produced when the bed height of small secondary water course systems is set as the water course height, even though they are not carrying any water at all. In this case, the water course heights at the nodes for interpolation must be deleted.

To include the attributes VORF and POTE (they were already assigned in the section Direct assignment), they are transferred from "VORF" or "POTE" to "EICH" via *Attributes → Copy → Attribute-wise*, in order to incorporate them in the interpolation as "existing data".

The menu for the interpolation of node data is available via *Attributes → Assign → By interpolation*. There the file s3_gwmesspkte_neu.txt (download on our !) is selected under *Sources*: *"From ASCII file" Add entry*. In addition, the menu item "*Include existing data*" is activated. This will include the already existing data under data type selected as results data (here EICH) in the interpolation points.

Here the results data are the calibration potentials (EICH). The data type and, for a 3D project, the layer number are defined under which the interpolated are data to be saved. The data type may differ from the codes of the structure data that may have been selected for interpolation!
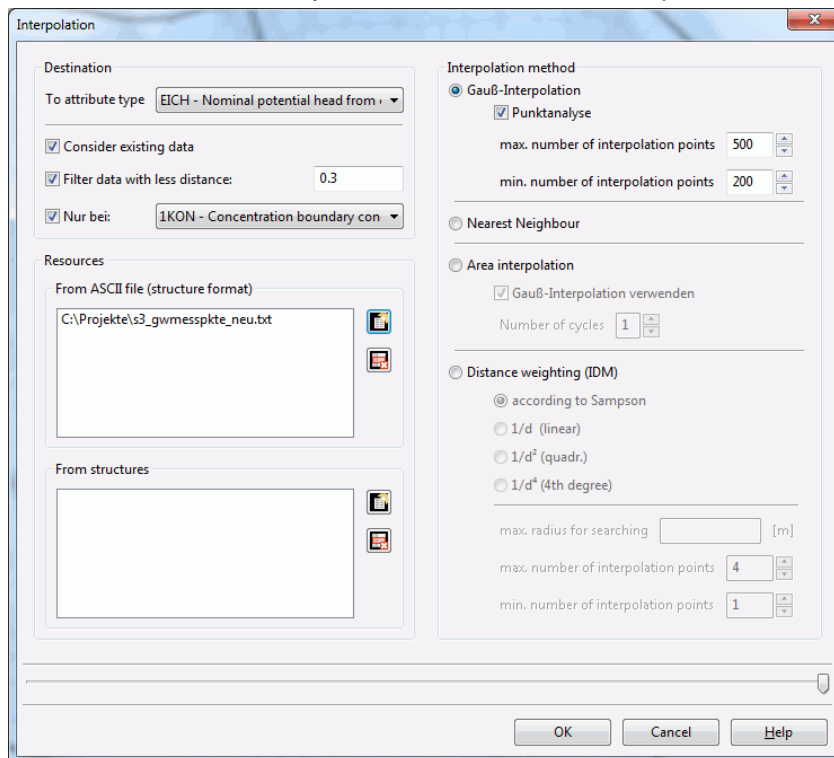
Fig. 5: Interpolation of node data

Boundary conditions fo the interpolation:

By checking the check box "*Filter data with less distance*" interpolation points with nearby or identical coordinates can be excluded from the interpolation. The distance between all interpolation points for the minimum distance entered in the text box is checked. If two interpolation points have a smaller distance, the second interpolation point will not be used. First the existing data, then the structure data according to their sequence in the list and then the data from the ASCII files will be checked.

By selecting the check box *only by* node data can be interpolated in the area in which the selected data type is assigned.

Attention: In distance weighting, you have to ensure in the interpolation protocol that all FE nodes receive a value. If not, the "search radius" must be increased.

Moreover, the measured values and the position of the measuring points can be displayed at any time in the following input window by clicking on *File → Import → Overlay file → *.txt*.

First, the "Prev" button is clicked. At the bottom of the dialog then the columns are selected. In addition, a marker can be selected (type, height and colour).
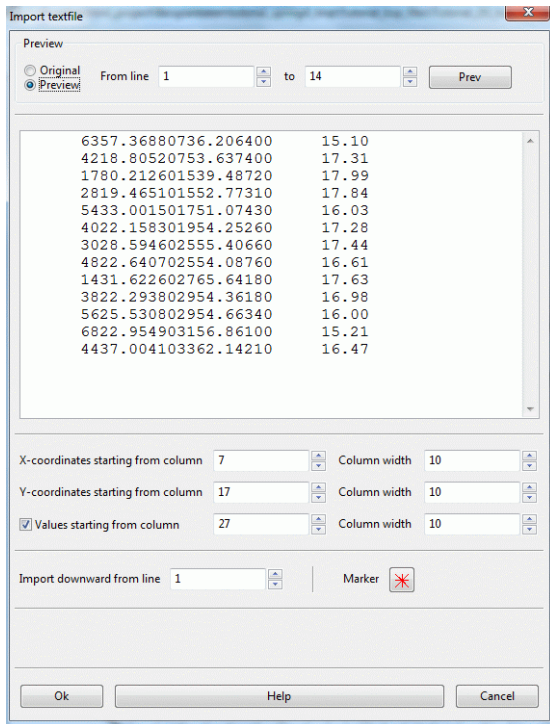
Fig. 110: Importing a text file

After successful interpolation, the calibration potentials (EICH) can be displayed via *View → Show attributes → Isolines/Area plots/Values* in the following form (Isolines in interval 12.50 to 20.0 and distance 0.5 m):
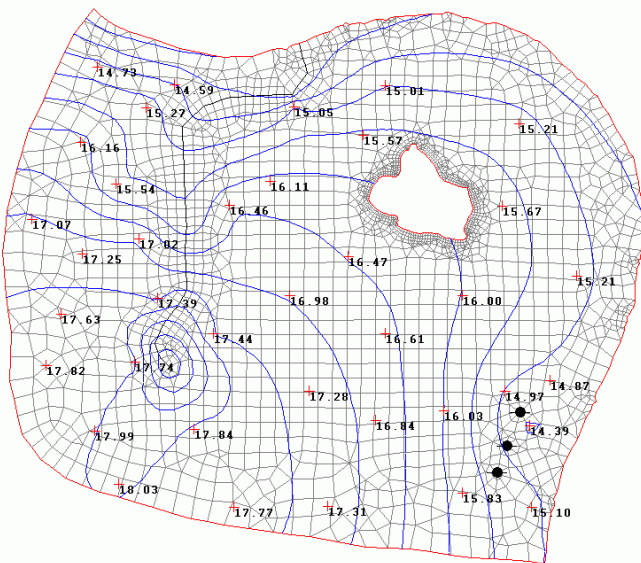
Fig. 111: Interpolated initial potential heads, contour lines with a distance of 0.5 m

In the upper reach of the stream, the previously described, but not really present, groundwater infiltration of the stream can be seen, which is made visible by means of a "groundwater elevation". For calibration, the data were processed as follows: At 17 nodes of the upper reach of the water course system, the water course system potentials (VORF) are deleted, and the interpolation of the calibration potentials (EICH) described above is carried out again.
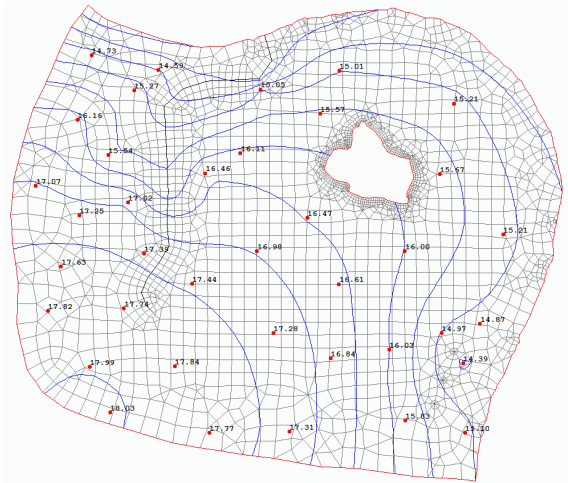
Fig. 6: Interpolated initial potential heads considering the real groundwater situation

This is followed by reassigning the complete water course potentials (VORF) stored under an auxiliary attribute (e.g. KKKK).
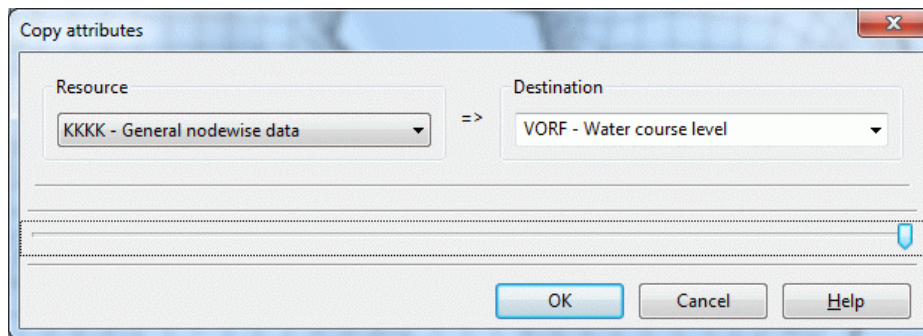
Due to the low data density in the area of the pumping wells, their groundwater depletion is not reflected in the curve of the isolines.

Moreover, the ground level elevation must also be assigned. In most cases, ground level elevations are available as ASCII data records, which first have to be read into the structure data file: here "s3_gela.txt (zip)" (download on our and unzip!).

The data are again assigned via *Attributes → Assign → By interpolation, Sources: From ASCII file: .txt*.

5.1.3.3 Manual attribute assignment

For an easy overview, the marker "120" (typical well marker) is assigned to the pumping well as follows:

The withdrawal nodes can be displayed via *View → Show attributes → .../Values → KNOT* and assigned the corresponding mark via *Attributes → Assign → Direct: MARK = 120.*

For the subsequent flow calculation, the groundwater recharge (FLAE) and the permeabilities still have to be assigned. These data types can be assigned globally to all elements by clicking on *Attributes → Assign → Direct*. In the example, the groundwater recharge for the entire area is set to FLAE = 0.200 with the added character " * ", which corresponds to a recharge rate of 0.200 m^3/m^2/time unit or 200 mm/year.
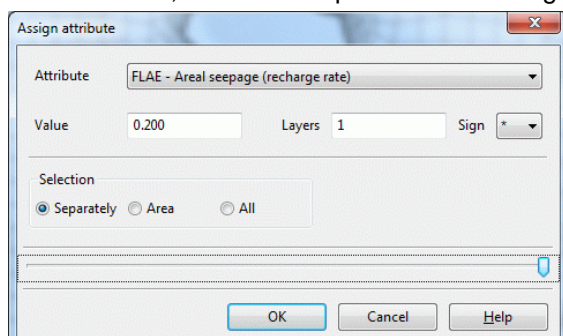
Fig. 112: Direct attribute assignment

The same is now done for the definition of the permeability values (KWER) of the model area, but without additional character in the 15th/16th column. The starting K values before starting calibration are set to 0.00005 m/s. In addition, an upper (KMAX = 0.002 m/s) and a lower (KMIN = 0.000001 m/s) limit for calibration are introduced for the K values, which must also be assigned to all elements.

The assigned marks can be displayed via *View → Markings*, or the corresponding button ( ) can be pressed.

The mesh generated up to this point, including the structure data file, was already downloaded from our in the directory: ".../Tutorial_bsp_files/Tutorial_2D_bsp_files/s3_schritt3_3.zip".

5.1.4 Step 4: Calibration of a groundwater model

5.1.4.1 Visualisation of the initial state

If the FE mesh has been generated and the data assigned, a model inspection has to be carried out prior to calibration (and each time data of the mesh file are changed). This is done via *Calculation → Model checking*. For quicker processing, the model data are stored in so-called background files (aaa, bbb, ccc).

5.1.4.2 Generating a model data plot

Following the model inspection, plots can already be generated from the model data. To illustrate the previous model version, the calibration potentials (EICH) can be displayed in an isoline diagram. Upon clicking on *File → Plot generation → Top view/Map generation*, the following input window appears:
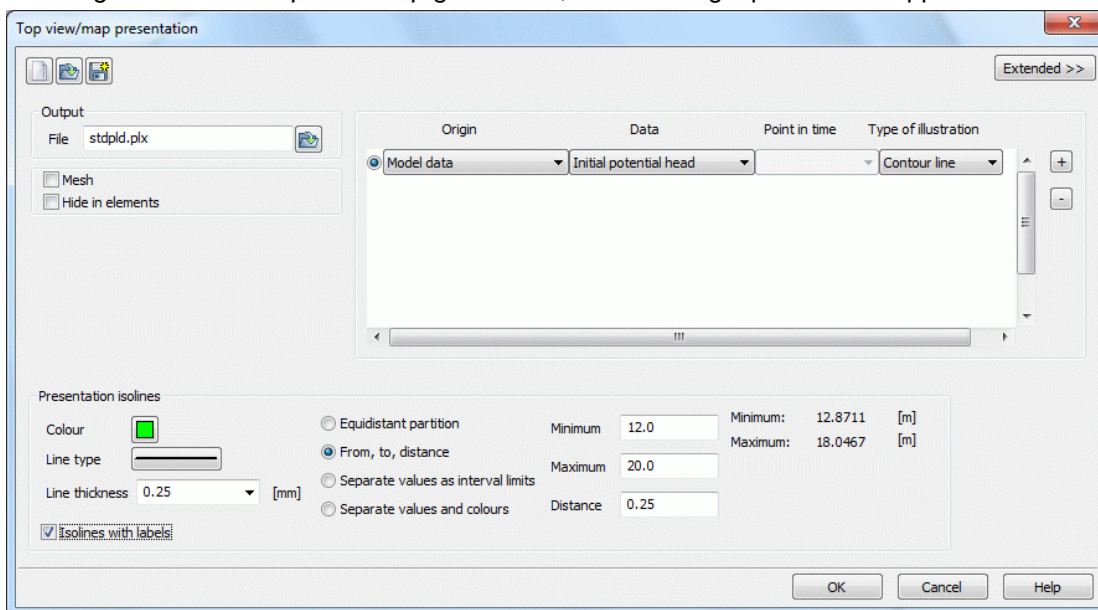
Fig. 113: Plot generation

After selecting the data type *Model data → Initial potential heads (EICH) → Contour lines,* the display *From to at a distance:* 12.0 - 20.0, 0.25 m, *Colour:* green is selected. Then a name is entered for the plot ("eich_intpol.plx"), and the calculation run is started. To view the plot, press *File → Open: *.plx*. The following picture appears:
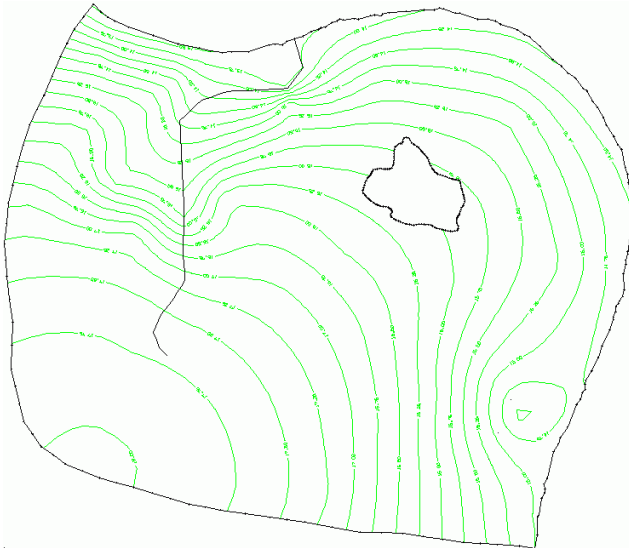
Fig. 114: Contour lines of the interpolated initial potential heads

5.1.4.3 Calibration by the gradient method

First, the existing (starting) K values are used to start an iteration run: *Calculation → Model calibration*. The following dialogue appears:
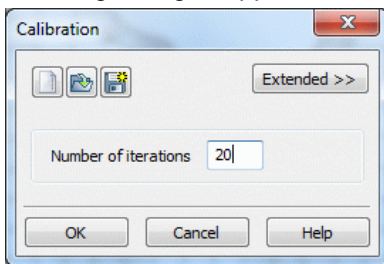
Fig. 115: Calibration

After opening „Extended" further settings can be made:
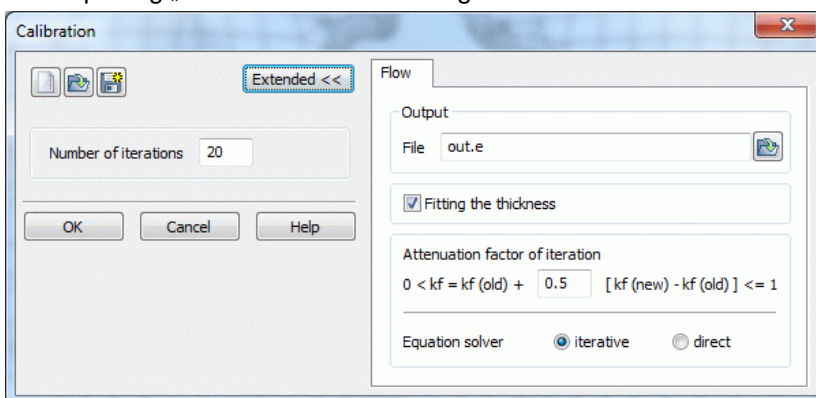
Fig. 116: Extended parameters

After the calculation run, the isolines interpolated from the measured values are contrasted with the isolines obtained from the K values.

To do so, the model data type EICH and the resulting potentials are displayed after calibration (red) in the input window under *File → Plot generation* "*Top view/Map generation*". Upon closing the module, the settings for this plot are saved to the batch file "plo.bpl" (predefined).

The following plot shows the result of the first calibration:
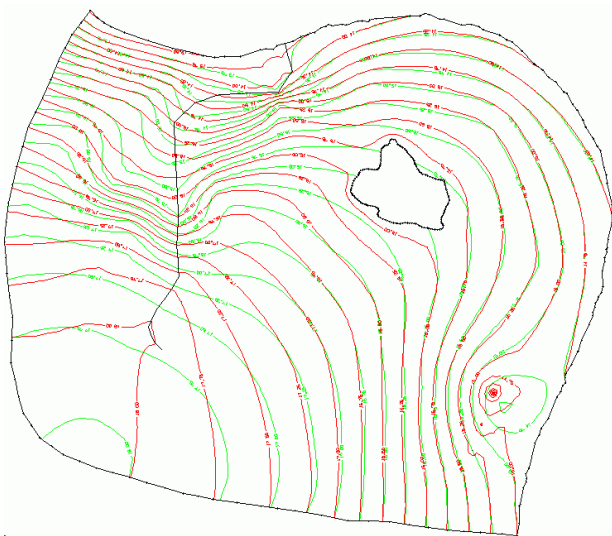
Fig. 117: Calculated (red) and interpolated (green) potential heads after the 20th calibration

5.1.4.4 Generating a measured data difference plot and saving the data

When a measured data file is available, the differences between measured and calculated groundwater levels can be displayed at the measuring points (usually groundwater measuring points) by "*Differences as circles*".

This is done during the plot display by selecting "Measured data" under data origin. An input window appears in which the file containing the measured values and the data to be used for calculation are defined:
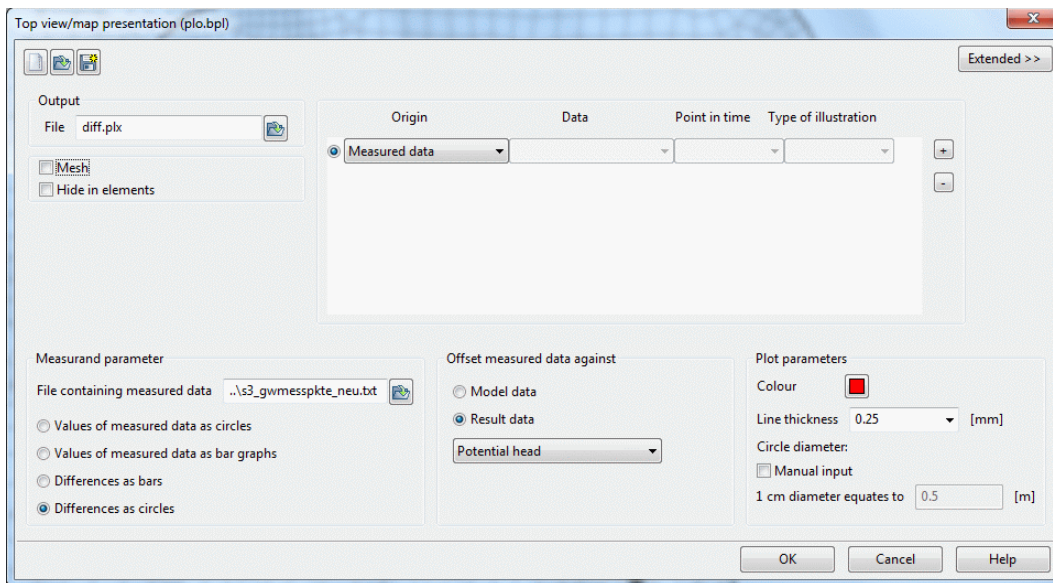
Fig. 118: Generating a difference plot

The file containing the measured data was already made available for download in Step 3: Data assignment.

In the input window, differences are selected as circles, giving the following picture:
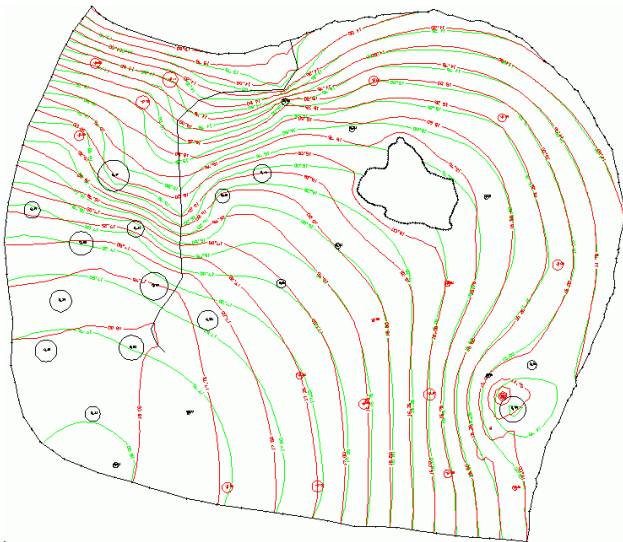
Fig. 119: Contour lines with measured differences

The calculated isolines already reflect relatively closely the curve of the isolines determined by measurement, in particular in the eastern part. The deviation at the measuring points is at most 0.47 m, being on average 0.10 m. (These deviations are displayed in the protocol window of the plot display. The deviations in the calibration protocol window refer to all nodes and are at most 1.24 m, on average 0.11 m). The calculated differences are shown for each measuring point:

Positive values (black) mean: The model calculation produces larger values than the measurement. In this example, that means: In the model, there is too much water at this position. Negative values (red) mean that the calculation produces smaller values than the measurement. In this case, that means: In the model, there is too little water at this position.

Note: If the calculated differences are to be saved as a file for later evaluation, this can be done under *File → Export → Difference to measured data:*
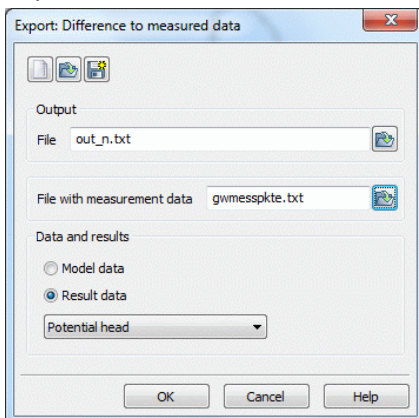
Fig. 120: Export of measured data differences

5.1.4.5   Calibration of the example in three steps

The first calibration produces a good starting base for the K values. They are imported to the mesh file via *Attributes → Import model data/calculation results*.
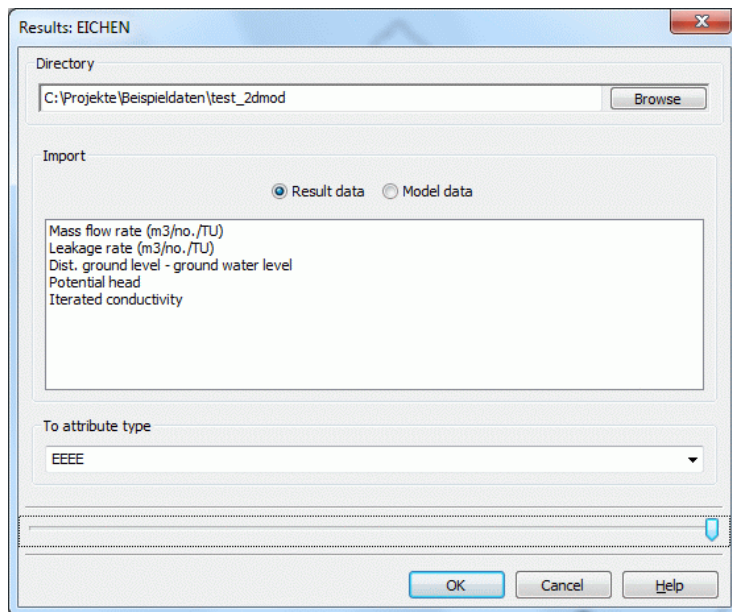
Fig. 121: Import of result data

When the attribute "KWER" is selected, the existing K values are overwritten. To avoid large jumps of the K values in adjacent elements, the K values are smoothed as follows:

Attributes → Computation → Smooth: "KWER". The weighting factor is set to 0.5, and the weighting type is preset to "linear". The mesh file is saved. Then the model is inspected (*Calculation → Model checking*), a flow calculation (*Calculation → Stat. flow*) is carried out with 5 iterations and the module GEONEU (*→ Advanced settings*), and a new plot is generated using the previously saved "plo.bpl" batch file. The result is shown on the following picture:
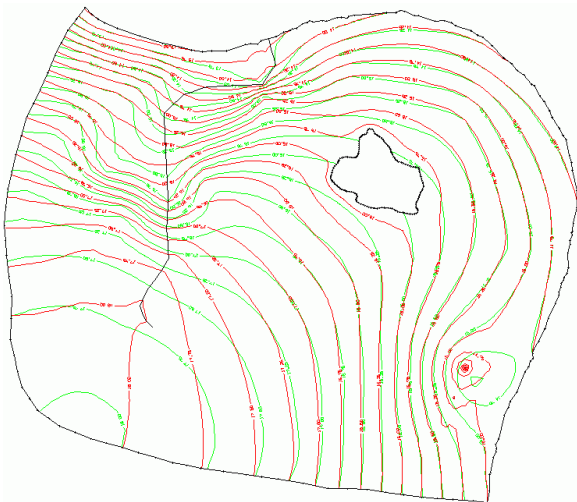
Fig. 122: Contour lines with difference plot after smoothing of the K-values

The maximum deviation at the measuring points is 0.50 m, the average being 0.09 m.

The curve of the isolines in the upper reach of the water course indicates an infiltration of the stream water into the groundwater. Upon "crossing" the water course, the isoline "17.75 (red)" forms a tip in the direction of flow of the water course.

Since this is not in line with the actual situation (stream bed is probably dry), the parameter MXKI is assigned the value 0.0 at the last 14 nodes, i.e., an infiltration is prevented (*Attributes → Assign → Direct*).

After performing the model inspection, carrying out a steady-state flow calculation and generating the plot, the following picture is obtained:
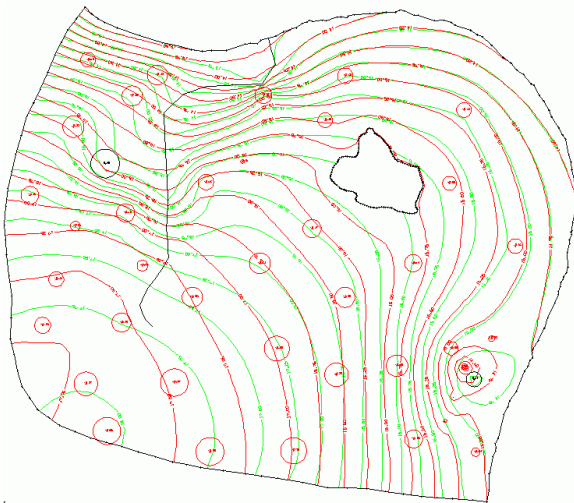
Fig. 123: Contour lines with difference plot after introduction of "MXKI" at 14 nodes

The infiltration in the upper reach of the water course has been eliminated. The curve of the isolines is more plausible, but the average deviation of the measuring points has deteriorated. The maximum deviation is now 0.39 m and the average 0.17 m. A large portion of the model seems to contain too little water!

It seems logical to start another calibration using the parameters modified up to now. The result is shown on the following picture:
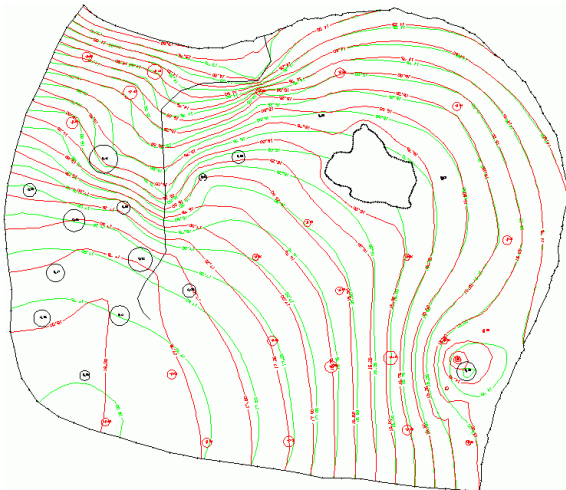
Fig. 124: Contour lines with difference plot following a new calibration

Finally the iterated K values are copied to the mesh file, and a new flow calculation is carried out.

The updated mesh file "s4_kalibriert.zip" (Download on our , *.zip) gives the following picture:
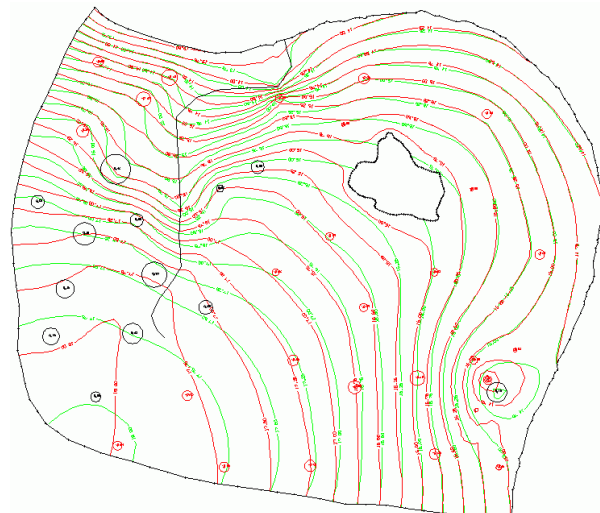
Fig. 125: Contour lines with difference plot of the calibrated model

The deviations at the measuring points are at most 0.41 m and on average 0.08 m. This is considered suffi-ciently accurate for the example model.

The distribution of the permeabilities now looks like this:
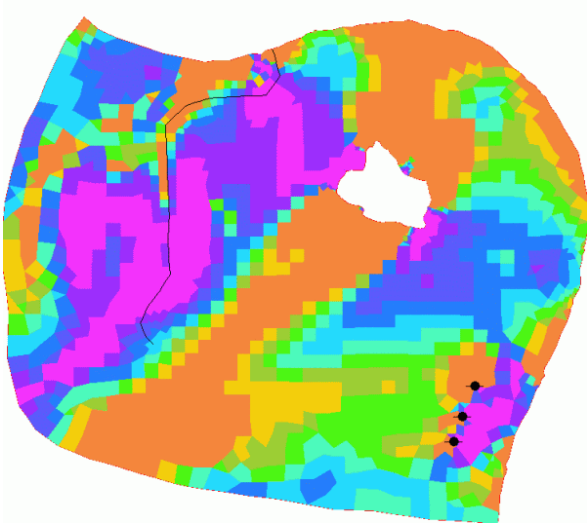
Fig. 126: Distribution of the calibrated permeabilities

The permeabilities were displayed in the interval 1.0e-06 to 0.002 within 10 ranges of values (via *View →
Show attributes → Isolines/Area plots/Values*). Upon clicking on *View → More windows → Project manager*
the layer list appears in a new window in which the arrangement of the individual layers can be changed via
drag&drop.

(Note: "Top" in the layer list corresponds to all the way at the back, "Bottom" corresponds to all the way at the
front). To display the legend of the corresponding layer, click on the check box.
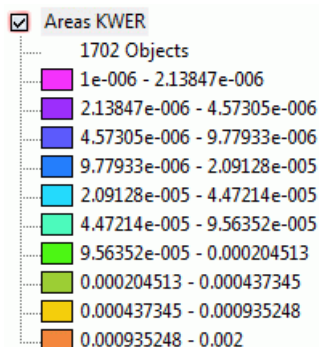
Fig. 127: Legend of the permeability values [m/s]

A stepwise model calibration can look like the one described, but it always requires knowledge by the user on
the local situation and cannot be carried out by the book. Moreover, a calibrating should always end with a
plausibility check of the (iterated) parameters.

5.1.4.6   General information on calibration

The K values iterated during model calibration are saved to the output file out.e in the file format of the *.net
file and can be copied from there to the mesh file. The procedure (*Attributes → Import background data ...*)
was already explained at the beginning of chapter Calibration of the example in three steps .

Due to the complexity of a groundwater model, the automatic change of the permeabilities does not necessari-ly lead to a plausibly calibrated model state. This is why the model must be recalibrated "by hand". The basic
procedure is as follows:

5.1.4.7   Schematic procedure during calibration

1. Copy the automatically iterated K values to the mesh file

2. Model checking

3. Flow calculation

4. Comparison of the groundwater isolines from the measured data with the calculated groundwater isolines, if available, it will include the display of differences (circle plot) at the measuring points,

5. Change in the permeabilities by area according to the isoline curves

   - if the calculated potentials are too high, then the ground is too impermeable, i.e., it may be necessary to increase the K values

   - if the calculated potentials are too low, then the ground does not hold enough water, i.e., it may be necessary to set the K values to less permeability

6. In addition, there are, of course, other parameters which likewise can be changed by area:

   - increasing or reducing the leakage factor of the water courses

   - Check inflows from the boundary

   - Smoothing of the K values via *Attributes → Computation → Smooth:* Select KWER and set weighting factor, for example, to 0.5. This will balance jumps in adjacent mesh elements.

After changing the parameters (tip: do not change too many parameters at a time!), the model is inspected, and then a flow calculation is carried out. Then step 4 of the schematic procedure is started again. This process is repeated until sufficiently accurate deviations are obtained, and the isoline curves seem plausible.

5.2 Creating a 3D-Model

The zip file „s4_2D_schulung3D.zip" includes the files used in this chapter. You can download it from our homepage (http://spring.delta-h.de) at Download & Support: Tutorial (EN): Tutorial_bsp_files.zip ( ).
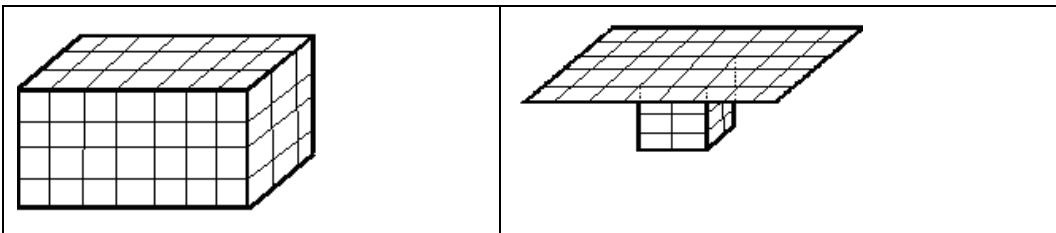
During an investigation the local conditions often request the modification of a two dimensional model into a 3D model, because of geological faults, a second aquifer or a groundwater withdrawal from a deeper groundwater layer. Besides it is always useful to build up an entire 2D model first and then to change it into a 3D model after resolving all geometrical errors. Based on the existing model "s4-2D_schulung3D.net" the way how to create a 3D model is described below.

5.2.1 Creating an entire 3D model
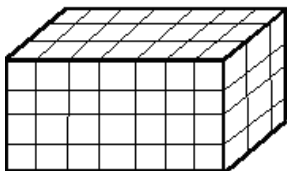
Example of a homogeneous 3D-model

5.2.1.1 Theoretical background

SPRING has certain rules to be followed in three-dimensional modelling. The main concepts which simplify the 3D mesh generation are:

- A 3D mesh is always based on the 2D mesh of a corresponding horizontal model.
- All nodes in one line over the depth of the model have the same x and y coordinates.
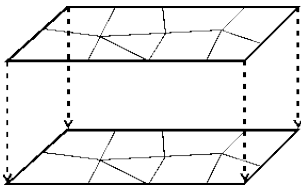
Fig. 128: Vertical projection of a horizontal mesh

A completely variable 3D finite element mesh with nodes and elements arranged freely in 3D space would require complex data management involving considerable memory space and computing time. With this type of mesh there is a risk that the user can lose track of the data. The decoupling of the planar and spatial mesh generation is actually the better and more efficient way to create a 3D mesh. First, equivalent to a planar horizontal model, the 2D finite element mesh is generated. Don't forget the compulsory points in the vertical projection, e.g. exfiltration wells, gauge wells and boundaries of geological layers. In a second step, the 2D mesh is vertically projected into the third dimension.

Therefore, all nodes and elements lie above or below corresponding nodes and elements. All 3D elements have the same projection on the horizontal plane. The user needs only one mesh structure for locating any node or element in any given layer.

Beyond that, the basic 2D horizontal model has two main advantages. The user can complete the planar mesh and check it for errors before projecting it into the vertical direction. Already existing two-dimensional models can easily be enlarged into the 3rd dimension for the case that the calibration or the flow calculation shows that the 2D model cannot exactly model the real processes.
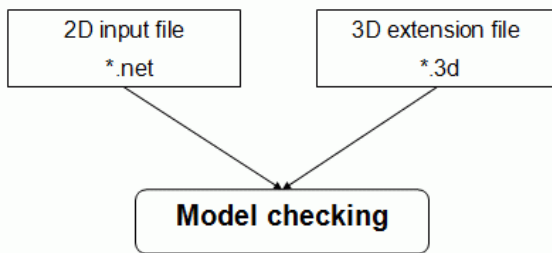
Fig. 129: Separated 2D and 3D data

The three-dimensional mesh generation is done during the model checking. The additional 3D-file contains the necessary parameters. Because of the simplification, there are only a few additional input data. In general, two different algorithms are available:

■   3D-enlargement by definition of the number of layers and proportion factors (3DSH)
■   3D-enlargement by the definition of z-coordinates (ZKOR)

5.2.1.2 3D-Enlargement by the definition of the number of layers and proportion factors (3DSH)

The bottom of the aquifer, known from the main input file, represents the lowest node layer. In general, the ground level (GELA) defines the topmost node layer. In case no GELA data exist, the defined ground water level (EICH) defines the topmost node layer. In case no GELA and no EICH data exist, the model boundary at the top is interpolated to nodes (OBER). By specifying the proportionality factors which define the ratio between the layer thicknesses, the layers between the lowest and the topmost layer can be generated.
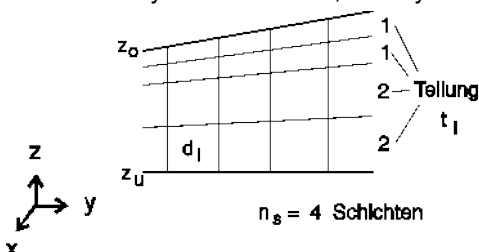
Fig. 130: Generation of a 3D mesh by the definition of proportionality factors

The thickness between the lowermost and the uppermost layers is usually different for every node of the mesh. It is divided into the defined parts and the corresponding z-coordinates are determined:

$$d_i = \frac{z_o - z_u}{\sum\limits_{j=1}^{n_s} t_j} * t_i$$

The above algorithm is useful for relatively homogeneous aquifers without complex geological layers where the spatial flow has external reasons, e.g. a partially penetrating well, flow around a building or beneath a river.

5.2.1.3   Realization in SPRING

After opening the existing project in SPRING you define the boundary of the 3D area with *Mesh → 3D → 3D-Boundary → Mesh-Boundary = 3D-Boundary*. With *View → More Windows → Project information* you see that the model type has changed to "3D-model".

Project information	
Name	C:\Projekte\html_proj...
Type	3D model
Time unit	Year
Extension X	4.51061e+006 to4.5124...
Extension Y	5.37688e+006 to5.3784...
Scale in x-directi...	5000
Scale in y-directi...	5000
Nodes	6888
Elements	7359
Node layers	20
Element layers	19
3D-boundaries	1
Node attributes:	
ZMAE	no Information
POTE	Potential head bound...
KNOT	Nodewise in-/outflow...
KKKK	General nodewise data
GLEI	Equal potential head
GELA	Ground level
EICH	Nominal potential he...
BERG	Terrain settings
AKON	Initial concentrations
1KON	Concentration bound...
ZKOR	no Information
Element attribut...	

Fig. 131: Project information

Next you have to arrange the new layers with *Mesh → 3D → New layer arrangement…* in the following dialog box:

New layer arrangement

New layer arrangement

ALL data of deeper layers will be lost

Surface from
- GELA
- EICH (- ZKOR layer 1)
- OBER (- ZKOR layer 1)
- ZKOR layer 1

Undersurface from
- UNTE
- MAEC (upside)

Proportions of layer thicknesses:
(seperated by comma or blank)

1,2,1

OK   Cancel   Help
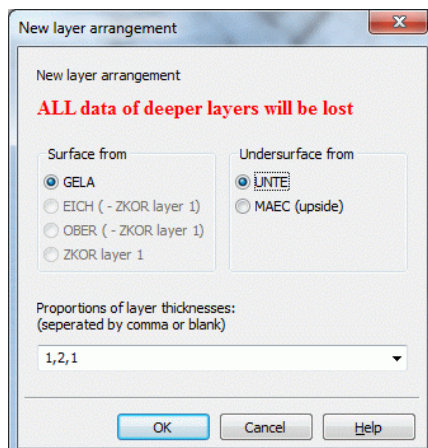
Fig. 132: Arrangement of new layers

If you have considered which and how much element layers are necessary (e.g. dependend on exploratory drillings) these numbers are entered in the input field. They have the following meaning:

■   The number of the figures (here: 3) defines the number of new element layers.

■   The sum of digits of the numbers (here: 1+2+1=4) defines the partial thickness of each element layer. That means, the entire thickness (= GELA - UNTE) is divided by 4 and multiplicated with the given numbers. So the first element layer gets a thickness of 1/4th of the entire thickness, the second gets a thickness of 2/4th of the entire thickness and the third gets a thickness of 1/4th of the entire thickness.

The following vertical section shows the proportions of each generated layer:
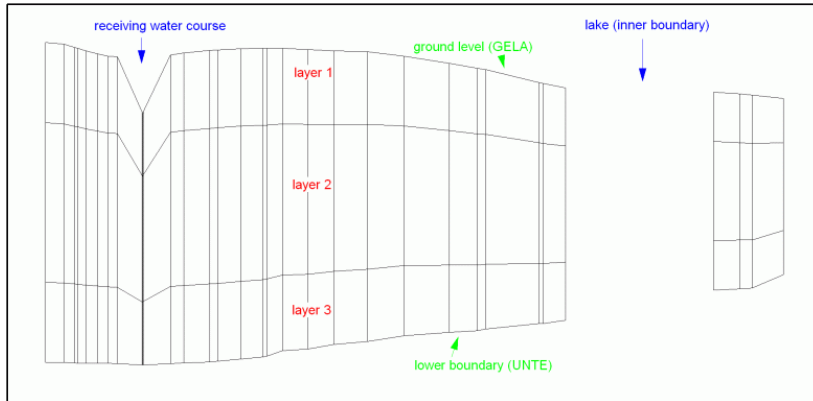
Fig. 133: Vertical section of the new 3D model (hundredfold super-elevation in Y-direction)

It is possible to create a user-defined section at every time with *View → Vertical section*: You select two points by pressing the left mouse button and the vertical section appears in an additional window.
This feature is independent of the model type.
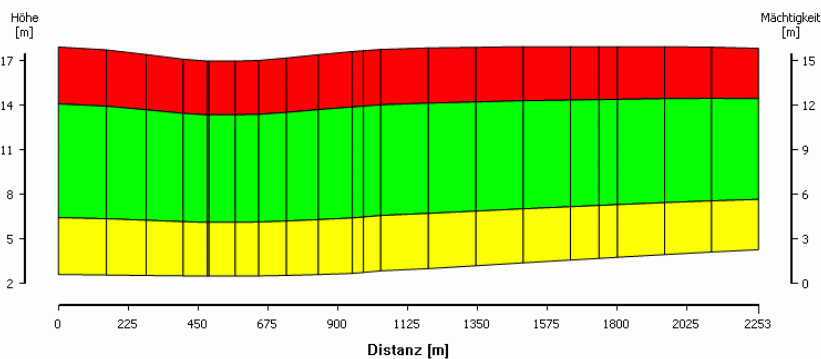
Fig. 134: User-defined vertical section through a 3D-model

The layers of the 3D-model can be visualised with *View → More windows → 3D-View*. An additional window opens, in which you can move the model in all directions by pressing the left mouse button:
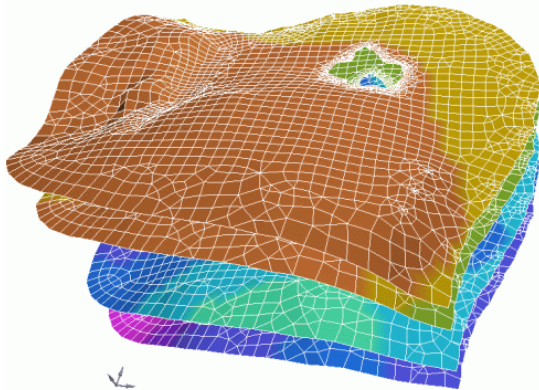
Fig. 135: Three dimensional view of the model

After saving the project you get the files "schulung_3d.net" and "schulung_3d.3d". You can download them from our for further proceedings in the next chapter: "schulung_3d.zip"

For a common flow computation it is sufficient to assign the permeability KWER to the elements of the new layers with *Attributes → Assign → direct*. For difficult hydrogeological conditions a calibration for each layer is requested and other necessary attributes have to be assigned to the generated nodes and elements (e.g. saturation parameters or impermeable layers, etc.).

5.2.1.4 3D-Enlargement by the definition of Z-coordinates (ZKOR)

This algorithm requires a z-coordinate at each node, except in the topmost layer (= ZKOR from the 2D input file or = GELA or EICH). The interpolation module (*Calculation → Interpolation*) can be used to facilitate this input. It can interpolate planar information (e. g. the ground level) from data at points or along lines. The interpolation can be used to compute the z-coordinate at each node of the horizontal layers. The x- and y-coordinates are known from the 2D model. Because of the variable z-coordinates, the horizontal element boundaries can follow geological layers.
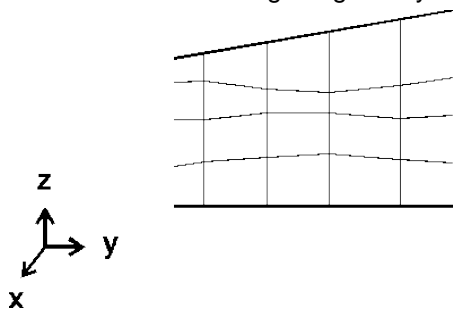
Fig. 136: Generation of a 3D mesh by the definition of z-coordinates

5.2.2 Creating partial layers in a 3D model

SPRING gives you the possibility to create partial layers and to do local mesh refinements in a model.
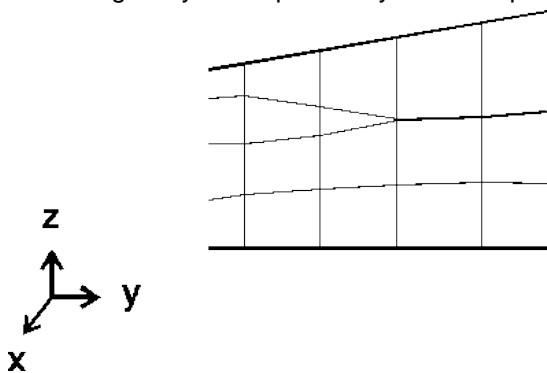
Fig. 137: Partial layers

5.2.2.1 Theoretical background

To consider geological faults or local limited aquitards in a 3D model a vertical mesh refinement is requested. By using the function *Mesh → 3D → Divide layer* in SPRING you can refine the mesh in the vertical direction. The attributes for the new elements and nodes are automatically generated (equivalent to the horizontal mesh refinement).

Only with the help of the exact z-coordinates can partial layers, which do not run through the entire three-dimensional model, be possible. To create the end of a layer, the z-coordinate of the lower node layer must be equal to that of the upper layer, for all nodes along the layer boundary. Afterwards, the model checking eliminates the superfluous nodes and elements. Then, the calculation modules will manage only the absolutely necessary data.

A parameter in the input window of the model checking decides if the z-coordinates are merged or not. A value of 0.01 m is set, that means, if the vertical spacing of two nodes is lower or equal 1 cm the nodes are merged to one node.
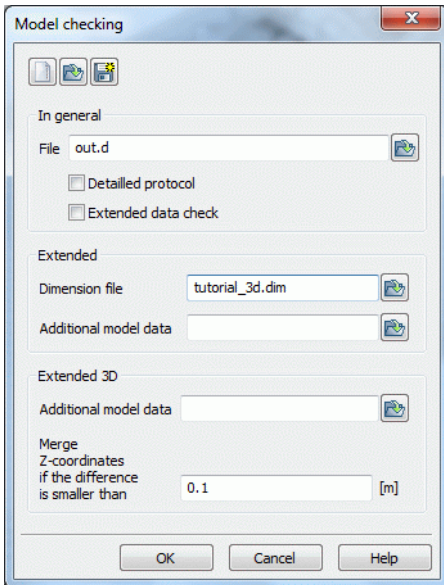
Fig. 138: Model checking with 3D extension

5.2.2.2   Realization in SPRING

Creating local vertical mesh refinements in SPRING are done with *Mesh → 3D → Divide layer*. Data of the new nodes and elements are generated automatically.

At first, the user has to decide with which method the division takes place

5.2.2.2.1 Conservative new layer arrangement

After pressing the button *Conservative* the following dialog box appears:
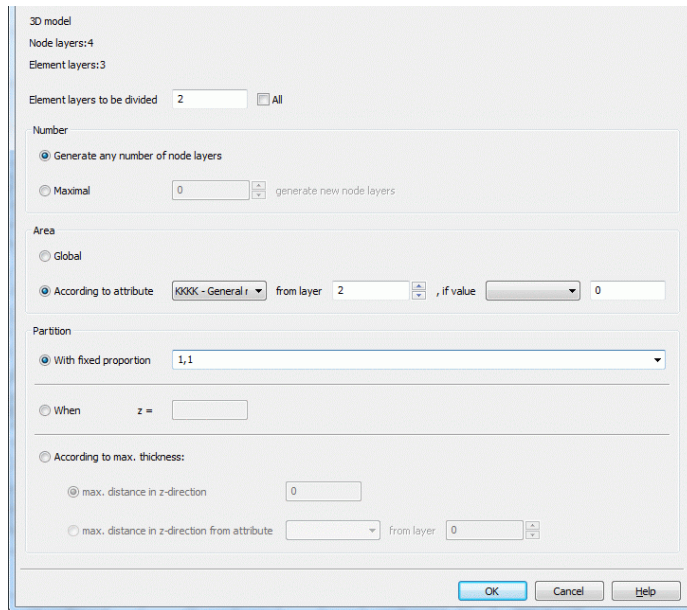
Fig. 139: Divide layers for a local mesh refinement

To show how to divide a special layer by attributes the attribute KKKK was assigned to some nodes of the second element layer.
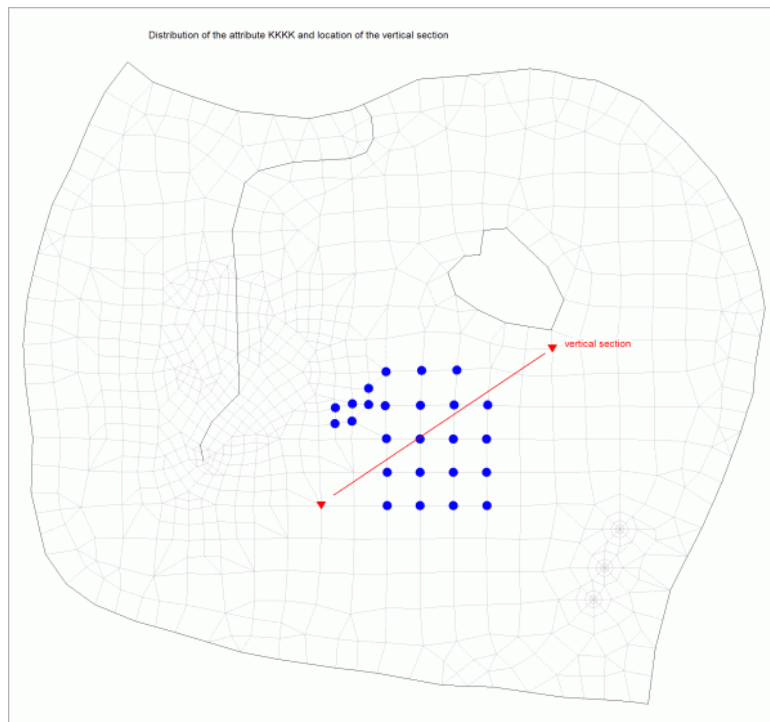
Fig. 140: Horizontal view of the nodes with the attribute "KKKK" and the location of the vertical section

First the element layer for dividing must be defined. Here layer 2 is selected. Now a new element layer is generated at all nodes to which the attribute KKKK is assigned. You must define how the selected layer is to be divided. There are two possibilities: either you define the number of new layers to be generated by using partition factors (equivalent to 3DSH), or you define a maximum layer thickness. In the latter case, the number of layers to be generated depends on the layer thickness specified. If you define partition factors (*Divide with fixed proportion*) for the division of the computed total layer thickness (equivalent to the attribute 3DSH), the

number of newly generated layers is constant at all nodes. If you define the maximum layer thickness (*According to max. layer thickness*), the number of new layers depends on the specified layer thickness.

Here *Divide with fixed proportions* was selected with the partition factors "1, 1". This input generates a uniform partition into 2 element layers.

After saving the project the user can visualise the new layer partition with *View → Vertical section*. Result is the following picture:
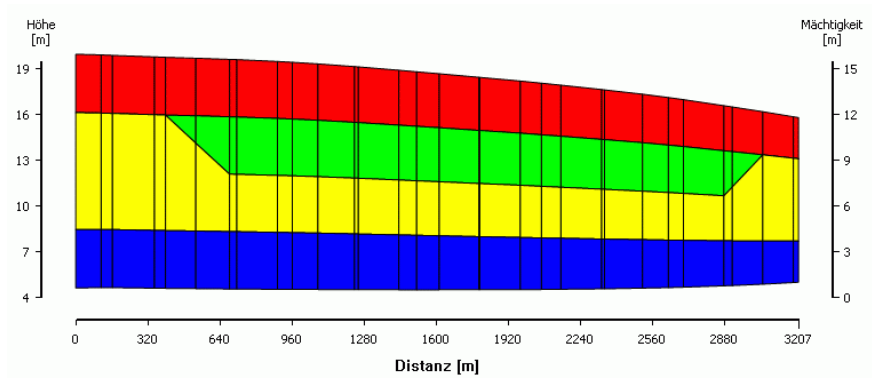
Fig. 141: Local vertical mesh refinement of the 2nd element layer, where the attribute „KKKK" is assigned (green area)

If the selected attribute exists only at a few isolated nodes the following vertical section results (partition factors: 1, 3):
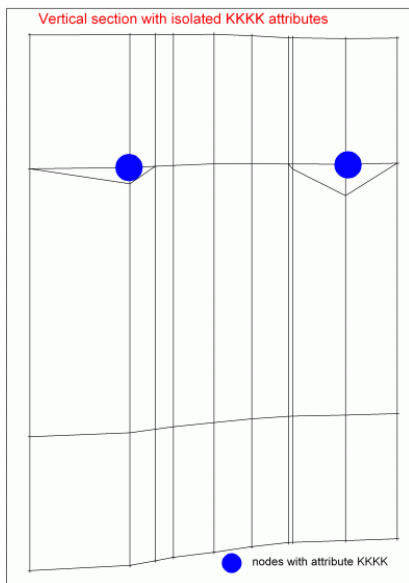
Fig. 142: Local vertical mesh refinement of the 2nd element layer, when the attribute KKKK is assigned only to a few isolated nodes

So you see that it is not always useful to define the partial layers by attributes. Mostly a manual treatment of the mesh based on drilling profiles is necessary.

5.2.2.2.2 Geometry optimized layer arrangement

When selecting the geometry-optimized layer arrangement, the following input window appears:

The following figure displays a 3D model with several partial layers to show the complexity of geological conditions.
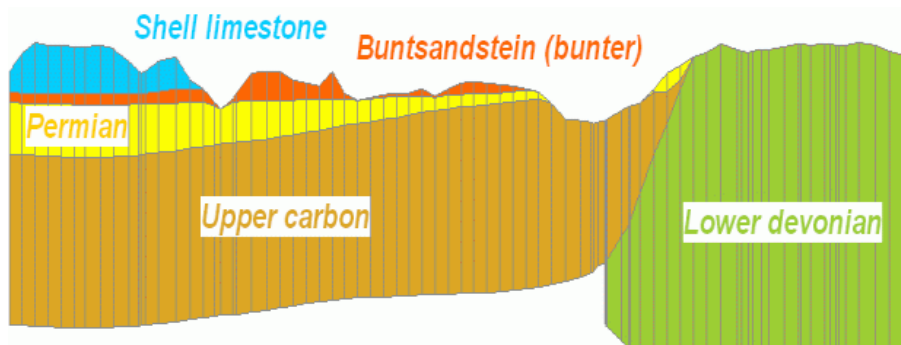
Fig. 143: Partial layers in complex geological conditions

In the geometry-optimized layer division, the top layer edge is obtained during the vertical refinement, but not the geological layer assignment. The layer-specific attributes should be saved beforehand with this division variant. The generation of poor elements is avoided in SPRING5.
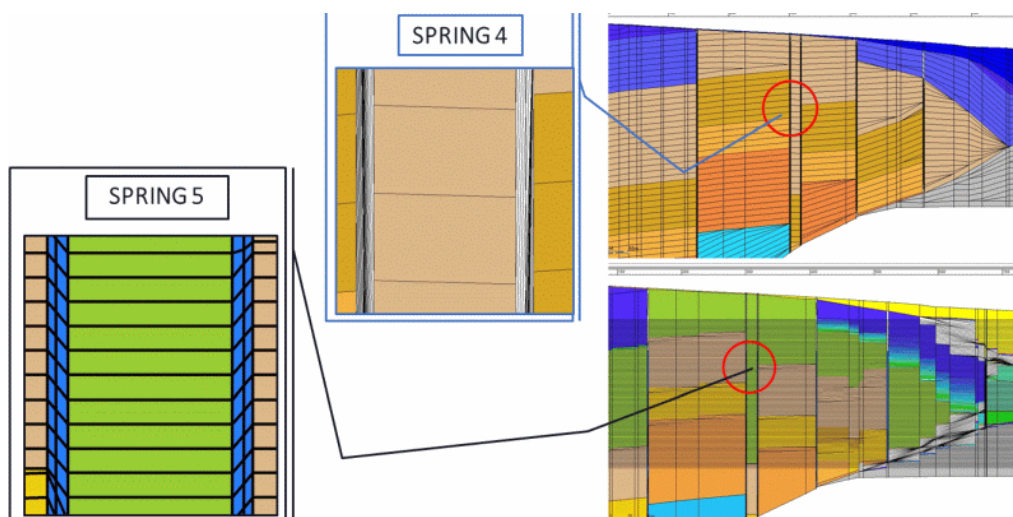
Fig. 7: New layer arrangement in SPRING5

5.2.3 Creating a 2D model combined with a partial 3D model

It is possible in SPRING, to extend a two dimensional horizontal model into a partial 3D model.
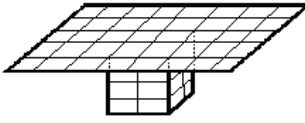
Fig. 144: Example of a horizontal model with a three dimensional part

5.2.3.1 Theoretical background

In most cases, three-dimensional groundwater models are quite small. Predominantly, the local three-dimensional flow is limited to a small area, but requires a fine discretisation into the third dimension too. This causes large numbers of nodes. Hence, the problem is to find correct boundary conditions which do not change during the simulation. Two-dimensional models can be enlarged until some natural boundary such as a watershed divide or a receiving water course is reached. For three-dimensional models, the number of nodes would grow quite fast. Otherwise, oversized horizontal elements in 3D models cause numerical problems. The best way of solving the problem is represented by a large 2D model combined with a partial 3D model.

The partial 3D mesh is generated automatically, and is equivalent to the complete three-dimensional model. In addition, the definition of one or more closed boundaries for the 3D area by node numbers (3DRA) is required. The program identifies the elements inside the boundaries and projects them into the 3rd dimension.

5.2.3.2 Transition zone between the 2D model and the 3D area

The critical zone in a combined 2D-3D model is the transition zone between the two model types. Every two-dimensional horizontal model assumes (based on Dupuit) exclusively horizontal flow. Hence, the partial 3D model must be large enough that the vertical velocity (direction z) in the transition zone is equal 0.

Specifying equal potential heads over the full height at the boundary nodes of the 3D area is a useful method for the transition condition. It means that the potential head at the nodes is unknown and must be computed during the calculation. But there is a second condition that defines a constant potential head for all nodes, one below the other (see identification GLEI). Mathematically, in the equation system, the single nodes are summed and represented by one representative node. The other nodes are eliminated.

The program automatically uses this procedure. Comparing the boundary of the 3D area (3DRA) with the 2D model, it can identify the 3D model as complete or partial. Equal potential head can never be used for the boundary nodes of a complete 3D model. Instead, explicit boundary conditions of 1st order (potential head), or rather of 2nd order (mass flow rate), are required.
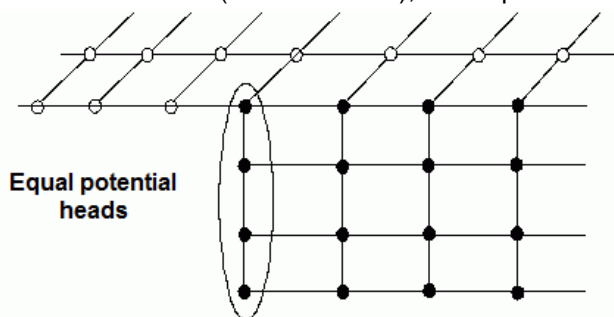
Fig. 145: Transition condition between the 2D and the 3D area

5.2.3.3 Realization in SPRING

For realising a partial 3D model in SPRING you have to define the boundary of the 3D part first with *Mesh → 3D → 3D-boundary → Generate*. All boundary nodes have to be captured.

After confirming the boundary with the right mouse button the new layers have to be arranged (see 3D-Enlargement by the definition of the number of layers and proportion factors (3DSH)).

Here the partition factors "1, 3, 2" were defined. Resulting are 3 layers with a partial thickness of 1/6, 1/2 and 1/3 of the entire thickness.

With *View → 3D-boundary* you can display the new 3D-boundary. Because the predefined way of representation is not very obvious change it with the Object toolbar (*View → Toolbars → Object attributes*). Change the colour and line width and assign them to the object.

You get the following figure:
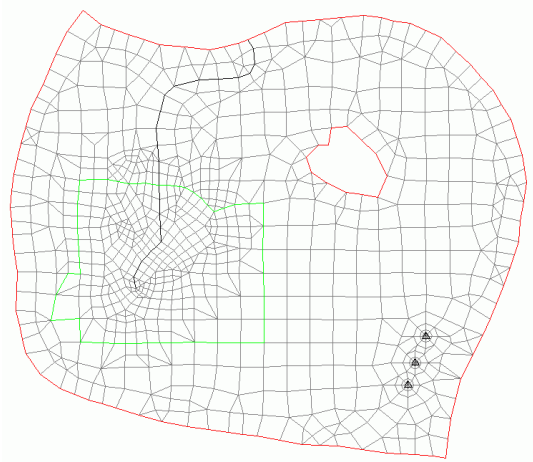
Fig. 146: Displaying the 3D boundary in SPRING

At least, the permeability (KWER) has to be assigned to the new generated elements to run a flow computation.

Besides after running the model checking it is possible to display the modified model with *File → Plot generation → 3D wireframe*. The following input parameters were defined:
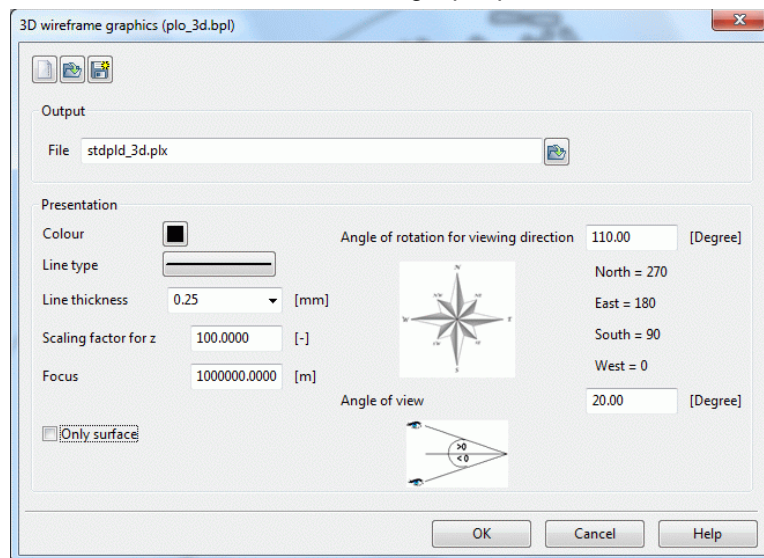
Fig. 147: Input for a 3D wireframe

The following figure of the partial 3D model results:
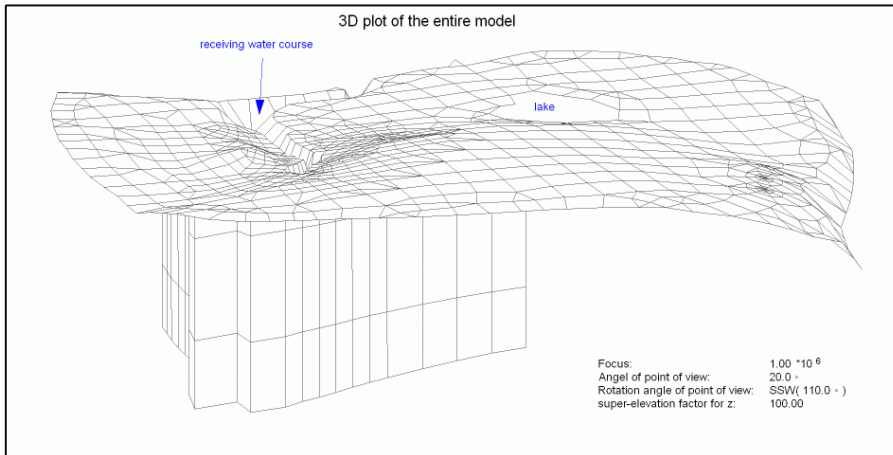
Fig. 148: Mesh plot with partial 3D area

5.2.3.4 Automatic numbering

The numbering of nodes and elements in the deeper layers is carried out automatically. By using arrays, no continuous numbering is necessary. The user can individually define the constant difference between the numbers of nodes and elements in two layers, one below the other. The default offset is 10,000. It allows maximum 9,999 nodes, or rather elements, per layer. The node or element number in the next layer is obtained by adding 10,000.
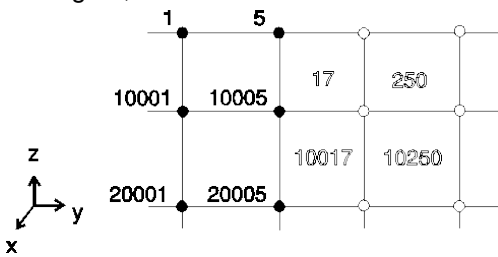
Fig. 149: Numbering of nodes and elements

The numbering with the help of a constant offset has the advantage that the general mesh structure can be obtained from the topmost layer and its node and element numbers. Any node or element in deeper layers can be identified by adding (n-1)*10,000 where n represents the layer number.

Large models can easily have more than 9,999 nodes or elements per layer. In such cases, an offset of 10,000 is not enough. With the help of the attribute 3DNR in the main input file, the value can be edited. For easier management, it is recommended to use always multiples of 1,000.

The input parameters (e.g. hydraulic conductivity or storage coefficient) which are assigned to the 2D model are assigned to the projected layers of the three-dimensional model (3D elements) as well.

5.2.3.5 Automatically assigned data

For a fast and effective data input for 3D models, the model checking automatically assigns the majority of the necessary data. Each layer receives the existing data from the layer above.

The following data types are assigned automatically

- Initial potential heads (EICH)
- Permeabilities (KWER, KWEV, KFVH)
- Limitation of the permeabilities (KMIN, KMAX)
- Dispersivities (DISP)
- Porosities (PORO)
- Storage coefficients (SPEI, KSPE)
- Initial concentrations (AKON)
- Anisotropic conductivities (Z-KA)
- Compressibility of the entire system (KOMP)

For instance, the input file for a 3D model with 6 node layers contains initial potential head for the first and third layers. Therefore, it follows that the second layer gets the values of the first one, while the fourth, fifth and sixth layers get the values from the third one.

If there are no vertical permeabilities or rather no ratio between the horizontal and vertical values specified (KFVH), the ratio is set to the default value 0.1, that means KWEV = 0.1*KWER.

5.2.4 Import of a submesh in an existing model

For many projects, the task is first to set up a large-scale overall model. Later a partial model is cut, refined and edited for detailed questions. In the following the detailed results are to be imported back into the overall model.

The following chapters provide guidance which steps have to be done before and after importing the partial model into the overall model.

5.2.4.1 Deleting the partial model area in the overall model

In the first step, the model nodes that lie within the partial model edge are deleted from the overall model. The following figure (Fig. 140) shows the area of the partial model before importing it into the overall model.
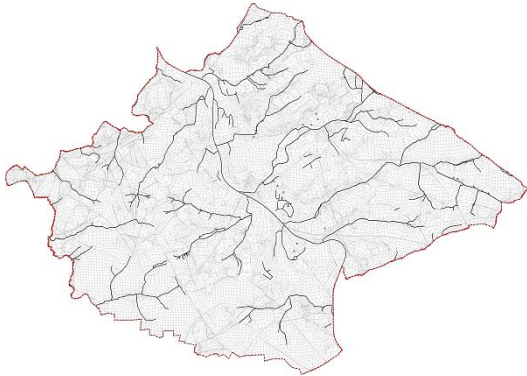
Fig. 150: Partial model area before importing it into the overall model

Along the outer edge of this model, the nodes and elements from the existing overall model are removed (*Mesh → Nodes → Delete → All inside selected structures* or *Mesh → Element → Delete → All inside selected structures*). The following figure (Fig. 141) shows the remaining mesh of the overall model.
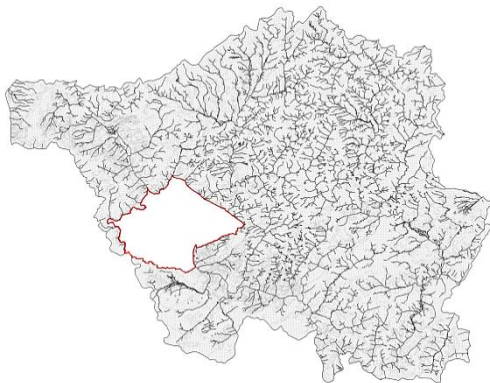
Fig. 151: Partial model area, removed from the overall model (red)

In case of concave edge gradients it happens that elements remain in the area of the partial model because their corner nodes lie on the model edge (Fig. 142, cyan highlighted elements that are outside the red border line). These elements have to be removed manually.
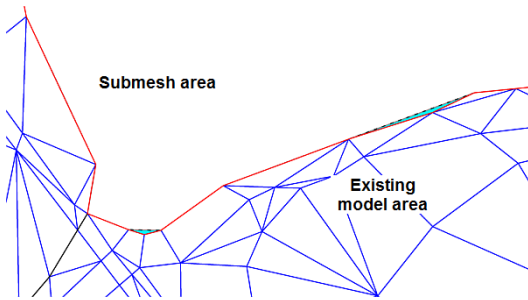
Fig. 152: Elements, which are not automatically removed

A second step is to examine whether linear data structures (e.g., water courses) intersect the model edge multiple times. If this is the case, the lines have to be separated between the point at which the line leaves the remaining model area and the point at which the line runs again into the model area.

The following figure (Fig. 143) shows such a situation. The image (a) represents the initial situation in the overall model. In image (b) the mesh along the broken red line was cut. In this case a blue line runs out of the remaining mesh and at another point back inside the mesh. The locations where such a situation occurs, have to be identified and must be adjusted by separating the lines (c).

This step is not only required for a correct presentation (e.g. markings), but also for a proper consideration of line-related leakage boundary conditions.
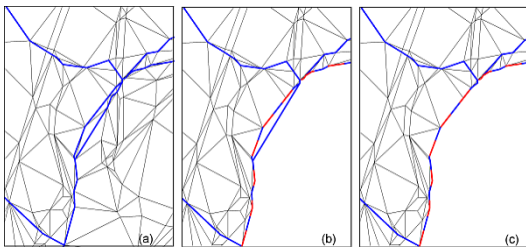
Fig. 153: Correction of line data

5.2.4.2 Horizontal border adjustment

Since the mesh of a partial model is refined for detailed questions, an adjustment of the boundary nodes between partial and overall model is required. For this, the boundary nodes of the partial model are reduced to the boundary nodes of the overall model. The advantage of this approach is to develop triangular elements from quadrilateral elements or elements fall away completely. The result is in this case that there is no need to create a new element, which has to be parameterized. The following figure illustrates this step:
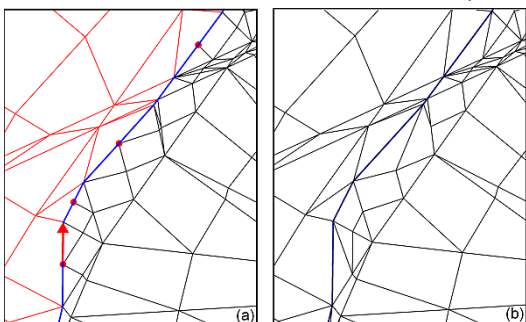
Fig. 154: Adjustment of the boundary nodes

Image (a) shows in black the elements of the refined partial model and in red the elements of the existing overall model. The blue line indicates the seam line of the two models. The red dots mark those boundary nodes of the partial model, which have no counterpart in the overall model. Thus, these nodes have to be eliminated before the integration of the two models. Simply deleting these nodes is not useful, as well as the adjacent elements are deleted, too. SPRING supports manually moving nodes to adjacent nodes (red arrow). By using the menu sequence *Mesh → Nodes → Merge → Capture two* the nodes are merged and the affected elements are automatically removed. The image (b) shows the corrected meshes:

It is important for the numerical calculation that no unfavorable or illegal element geometries remain (Fig. 145).
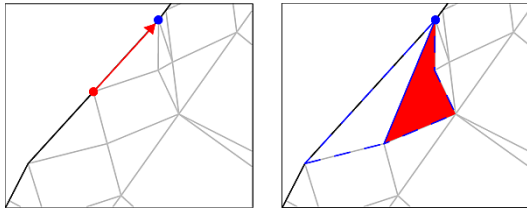
Fig. 155: Unfavorable or illegal element geometries

This has to be considered for each merging of nodes. In very few cases, if both possible directions of merging lead to illegal elements, a manual shift of a single node or dividing a quadrilateral element is essential.

Note:

To avoid this step, it is generally recommended not to perform the mesh refinement (both horizontally and vertically) up to the boundary nodes.

5.2.4.3 Vertical border adjustment

In dealing with a partial model the need can arise, to make changes to the layer structure of the model. The reasons for this can be of different nature:

■   Recent studies and drilling results require a change in the depth position of layer boundaries.

■   The processing of detailed questions in the partial model requires a more differentiated analysis of individual geological units, so that a further subdivision of certain model layers is required.

■   For the correct representation, for example, of imperfect wells or built components (sheet piles, tunnels, etc.) is another vertical division of individual model layers necessary.

Since the added vertical layers of the partial model have no equivalent in the overall model such an assignment has to be created. The following cases are differentiated:

■   In the partial model has been no change in vertical layers. In this case the boundary nodes correspond to the intersection of the two meshes, and the layer mapping is automatically correct. Image (a) of the following figure illustrates this situation.

■   In the partial model, the depth position of layer boundaries, for example, was changed due to new drilling data. However, a vertical mesh refinement was not carried out, so that the number of layers and the assignment between partial and overall model is identical. In this case, the assignment automatically leads to a correct result. If the depth position of layer boundaries was also changed to the edge of the partial model, the assignment nevertheless leads to a correct result. Since there is a unique layer assignment, the depth of the layer boundaries is defined by the imported model part. This means if the overall model is opened first and then the partial model is imported the depth of the boundary nodes of the partial model overrides the depth of the overall model. This situation is shown in image (b).

■   One or more layers in the partial model were vertically refined, except for representing a more differentiated geological structure or a proper hydraulic consideration of measures (eg imperfect wells, vertical wells, sheet piles, tunnels, etc.). Such a situation is schematically illustrated in image (c). In this case, a layer assignment must be performed at the import of the partial model. In the general case the vertical refinement is carried out to the edge of the partial model, so that the depth of the existing model layers has changed there. In this case, an automatic assignment of the model layers is not clearly possible.
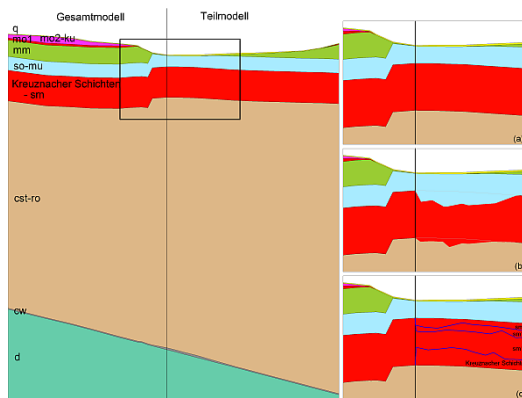
Fig. 156: Assignment of layers between partial and overall model

In the third case, the number of layers in the partial model has changed. The added layers of the partial model must exist formally in the overall model after the import, ie outside the partial model they merge geometrically together with hitherto existing layers. The assignment of the boundary nodes is controlled during the import of the partial model.

Although these eliminated layers are actually not considered in the simulations, they must be maintained for the modelling and for the node and element numbering.

Thus, if another partial model is imported into an overall model in which a vertically refined layer already exists because of a previous import, should as many of the existing coincident layers be used.

Conclusion: In general, the refinement of the layers of the partial model should not be performed on the model boundary. Thus, the depth position of the added layers can be properly assigned to the layer boundaries of the overall model.

5.2.4.4 Installation of the partial model in the overall model

In the previous steps, it was ensured that the boundary nodes of the partial model correspond geometrically with the inner boundary nodes of the overall model. Now, the attribution of the boundary nodes of the partial model is examined in view of the effect in the combined model.

- To the boundary nodes of the partial model is the in- and outflow assigned which was determined in the overall model. This boundary condition has to be removed from the partial model. Otherwise nonexistent in-or outflows are simulated.
- For the partial model has to be checked, whether line data extend beyond the boundary, too.
- Further is to examine whether the same attributes are defined on the corresponding nodes in the partial- and in the overall model. For data types that can be present on a node only once, this step is of minor importance, since in this case the value of the imported mesh overwrites the existing value of the opened mesh.
- For attributes that are allowed several times on a node (eg, line-related leakage coefficient), the corresponding attribute is to be removed in one of the two meshes. Otherwise, for example, has a receiving water course, which runs on the model boundary, after the integration of the partial model, a double-leakage effect. This needs to be avoided in any case.
- The user should have an overview of the existing attributes on the coincident boundaries to recognize present contradictions and to clarify the cause before certain data on the total mesh is definitively lost.

If the nodes of the coincident boundaries are adjusted in both models, both in relation to the geometrical location and on the attribution, the models are combined with SPRING.

The following strategy is advantageous:

First, the mesh of the overall model is opened, from which the area of the partial model was removed. Then the partial model is imported (*File → Import → Submesh ...*). If an attribute is already defined on a boundary node, the value of the partial model overwrites the existing value from the overall model. Since the partial model is usually up-to-date, this allocation is desirable.

If node and element numbers shall remain unchanged in one of the models, this model is opened first and then the other model part is imported, because the node numbering of the imported model is changed.

If the partial model is opened first and the overall model is imported, the 3D boundary must be recalculated (*Mesh → 3D → 3D-Boundary → Mesh boundary = 3D-boundary*).

After saving the integrated model, a new model exists without internal boundaries.

5.2.4.5   Implementation of consistency and plausibility checks

5.2.4.5.1   Geometry an data checks

After importing the partial model, it is recommended to check the geometry of the integrated mesh (*Mesh → Checks ...*). If errors in the mesh occur, this is usually due to the fact that one of the steps described above has not been carried out correctly.

The most common source of error results from the fact that two nodes do not coincide exactly at the inner boundary. Especially for large models, it may happen that a boundary node of the refined mesh was not removed during the manual correction. This node then lies on the model boundary of one mesh, but has no counterpart in the other. These errors are identified by the geometry checks and can then be corrected.

It has proven to correct the errors in the respective models, and then perform the import again. After this two intact models exist. Advantage of this method is that the detailed model can also be used for future considerations further and a new import into the overall model can be done with reduced effort. For a new simulation with the partial model is to be noted that the already deleted boundary condition of in- and outflow has to be calculated and assigned again.

The model checking (*Calculation → Model checking...*) gives possibly indications of inconsistencies in the model data which are not identifiable in the pure geometry checks. If there are no relevant data errors detected, the actual model calculation can be performed.

5.2.4.5.2   Comparison of the calculation results

The next step is to carry out with the geometrically correct overall model a flow calculation. To obtain comparable results, it should be ensured that in the calculation of the partial model and the overall model the same combination of calculation module and parameters are used.

The validation is done by comparing the groundwater contours of the free surface of the integrated overall model with those calculated with the old overall model and the partial model.

As expected, differences will occur around the boundary of the partial model. Since the recalibration of a partial model is not restricted generally to the inner model, changes occur in the flow situation, which also have an influence on the model boundary.

The constant boundary in- and outflows lead to changes in groundwater levels in the peripheral region during the calibration adjustments of the partial model.

The prescribed boundary condition is no longer being forced when calculating the total model and the model can achieve a balance between the situation in the total model and the new situation in the improved partial model.

A comparison of the given boundary in- and outflow of the partial model with the amounts that flow in the new overall model over this surface can also be performed.

5.2.4.6   Notes and recommendations

In a large-scale principle model, detailed questions cannot always be managed efficiently and economically. Caused by the size of the model high computation times and the discretization are often not detailed enough.

One approach is to separate and to consider the region of interest as a partial model. The coupling of the partial model and the overall model is done by given boundary conditions, usually by mass flows. The advantages of this approach can be summarized in the following points

- Large flexibility of choosing a region of interest
- Mesh changes and the varying of parameters can be handled efficiently.

To keep the influence of the model changes on the boundary conditions low, the boundaries of the partial model should be defined far away from the region of interest. When defining the model boundary the following criteria should also be considered:

■ Define boundaries as far as possible along stream lines,

■ sufficient distance to groundwater withdrawals,

■ no open water areas on the boundary,

■ the model boundary of the partial model must be always on element edges of the overall model, there must be no quadrilateral elements cut,

■ the partial model area must be large enough so that changes in the transition region between partial and overall model which are usually not completely preventable during the import, have no influence on the results of the region of interest.

■ In order to allow a smooth import of a partial model in the overall model, the model boundary of the partial model has to remain geometrically unchanged. It may not be refined either horizontally or vertically. If this condition is not observed, the reimport requires not only additional work by the need for a boundary node cleanup, but also an interpretative effort in the case of a vertical refinement. In this case has to be determined how the refined layers are correlated with the layers of the overall model. Vertical mesh refinements, based on local needs (eg installation of sheet pile walls, tunnels, wells and locally common hydrogeologic inhomogeneities) should always be carried out only in the environment of the concerned region.

■ The partial model should always be calculated using the same program module as the overall model. Also the calculation parameters such as the number of iterations or the attenuation factor should be maintained.

5.2.5 Generating fractures in a 3D model

With the menu item *Special features → Fractures → 2D-fractures from LERA* vertical and horizontal fractures can be inserted in a 3D model. This is based on the Python script "lera2kluft.py", which runs automatically when you click this menu item.

If the data type LERA (leakage for a polygon) is available the 2D fractures are written with their opening width in the * .3d file of the model.

The opening widths correspond to the values of the respective LERA-range. They have to be assigned by the user with the appropriate values.

5.2.5.1   Assign attribute LERA

The attribute LERA is assigned to any future fracture line.

The easiest way to find the fractures again is when they are numbered and the number reflects the value of the LERA attribute. It makes sense to remove all LERA-attributes that are not used for fracture generation before or to store them as another data type.
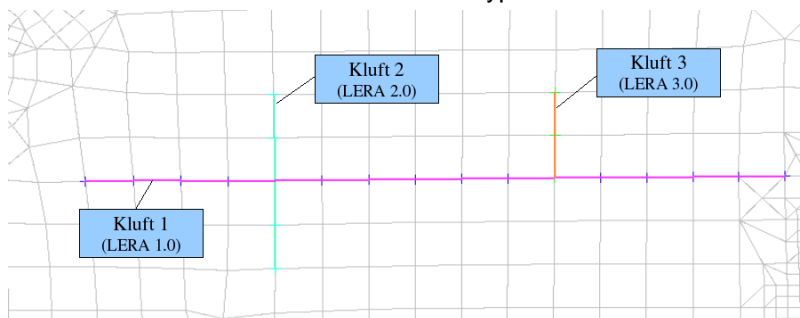
Fig. 157: Displaying the fracture system as LERA attributes

5.2.5.2   Editing the output in the *.3d file

After executing the menu item, the so generated 2D fractures (attribute 2DEL) are written in the * .3d file of the model. The fracture opening widths should be adjusted (data type 2DBR) to the actual fracture opening widths.

Since the LERA attributes were used only as auxiliary attributes, they should be deleted after restart of SPRING. Previously defined LERA attributes are assigned again.

5.3 Creating a transient transport model

The files of this example can be downloaded on our homepage (http://spring.delta-h.de) at Download & Support: Tutorial (EN): Tutorial_bsp_files.zip ( ).

The zip-file contains in the directory "Transport_2d" the model file, the transient input file "instat_RB.txt", the file "sitr" and the corresponding batch file „sitra.bsi" for a transient mass transport calculation.

In this chapter the technical steps will be presented which are required to create a transient transport model. A groundwater model is a transport model, when concentrations are present (attributes 1KON, AKON or KONZ).

5.3.1 Model description

The aquifer is homogeneous, isotropic and confined. The question is, how the pollutant disperses over the time. The length of the model is L = 300 m and it has a width of B = 100 m. The model area will also have the following characteristics:

Permeability	$k_f = 5 \times 10^{-4}$ m/s	KWER
Thickness	m = 18 m	MAEC
Porosity	$n_e = 0.2$	PORO
Dispersivities	$\alpha_L = 7$ m $\alpha_T = 0.7$ m	DISP (sitr-parameters)
Initial potentials	$H_l = 36.0$ m $H_r = 30.0$ m	POTE
Concentrations of the pollutant	$c_0 = 240$ mg/m^3	1KON
Factor of delay	R = 1	-
Decomposition constant	$\lambda = 0$	-

The pollutant entry is made permanent and constant. The location of the source is located at the boundary of the model (node nr 156), so there was a particularly fine discretization chosen.

The following figure shows the model mesh with the pollution source
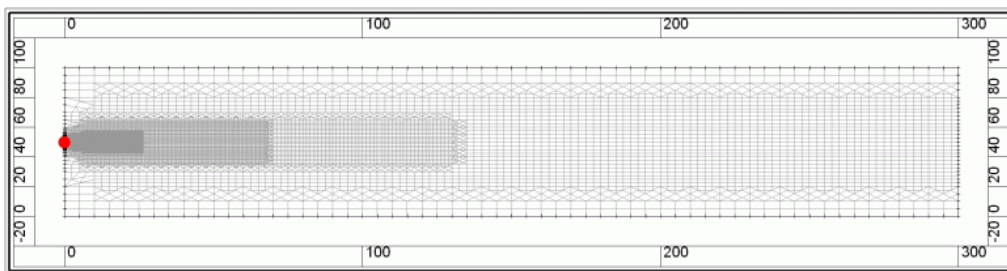
Fig. 158: FE-Model with pollution source

5.3.2 Necessary data for the mass transport

Initial and boundary conditions

- Fixed concentrations: attribute 1KON, Boundary condition of first order, set fixed concentrations. In the example, the choice of this boundary condition is required to represent the permanent constant source of pollution.

- Inflow concentrations: attribute KONZ [kg/kg]: With the data type KONZ concentrations (node wise) are assigned to entered inflows (from KNOT, FLAE, RAND, RANQ, RANX) and calculated flow quantities (leakage rates, mass flow rates from fixed potential heads). Example: In the surface of a landfill is a known quantity of pollutant (KONZ), which is solved but only by a certain amount of rain (FLAE).

- Initial concentrations: attribute AKON [kg/kg]: The initial state (node wise) for a transient calculation is preset. AKON functioned analogously to the initial potential heads (EICH) of the steady state flow.

Material data

■ Porosity: If the porosity PORO , the default value of PORO = 0.2 is used for the calculation.

■ Dispersivity: The longitudinal dispersivity α_L and the transversal dispersivity α_T are defined explicitly together in the menu of the *material transport calculation* under "Extended" → "Transport".

The following input window shows the possible entries in the extended settings of the material transport:

Fig. 159: Parameters of the material transport

The attributes 1KON (or KONZ), DISP and PORO are absolutely necessary for a mass transport calculation.

5.3.3 Transient input for the mass transport

After the *model checking* follows the calculation of the *mass transport*. The following input window appears:
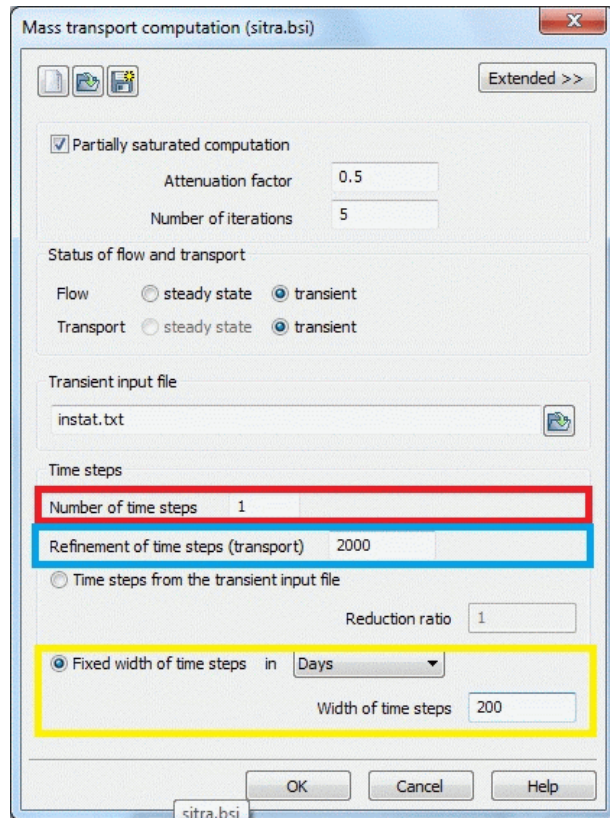
Abb. 160: Transient parameters (3D)

Explanation of the time steps

Red marking: The number of time steps, here 1, determines the duration of the transport calculation. Duration = number of time steps * time step width (here: 200 days).

Blue marking: Time step refinement, will indicate the subdivision of the time step size in time steps of the transport calculation (here: 200 days/ 2000 = 1/10 day).

Yellow marking: The time step width defines the duration of one time step of the flow calculation.

The quantity-based results are stored to the same time steps in which the potentials are saved in the background files ("Extended" → *Output*):
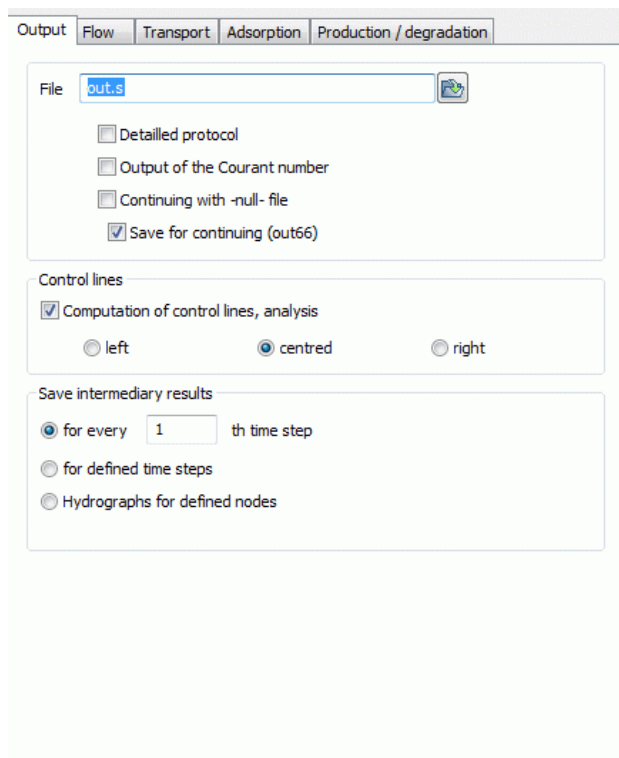
Fig. 161: Output parameters

After completion of the *model checking*, *flow calculation* and *plot generation* you obtain the following concentration distribution after 200 days:
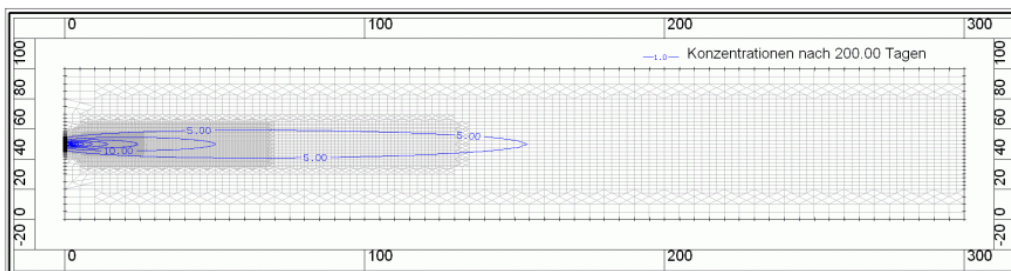
Fig. 162: Concentration distribution after 200 days, isolines in an interval 5 to 50 mg/m^3 with a distance of 5 mg/m^3

5.3.4 Transport with inverted flow

The procedure of the "Transport with inverted flow" is described in chapter How To – Catchment areas.

5.4 Creating a 3D vertical model

In a vertical model, the geometric elements are discretized in the vertical in contrast to a horizontal model. The construction of a vertical model is, however, by the same rules as that of a horizontal model. Typical examples of the use of a vertical model are dams, retaining walls of dams or a flow-related consideration of tunnel or main cross-sections in the vertical.
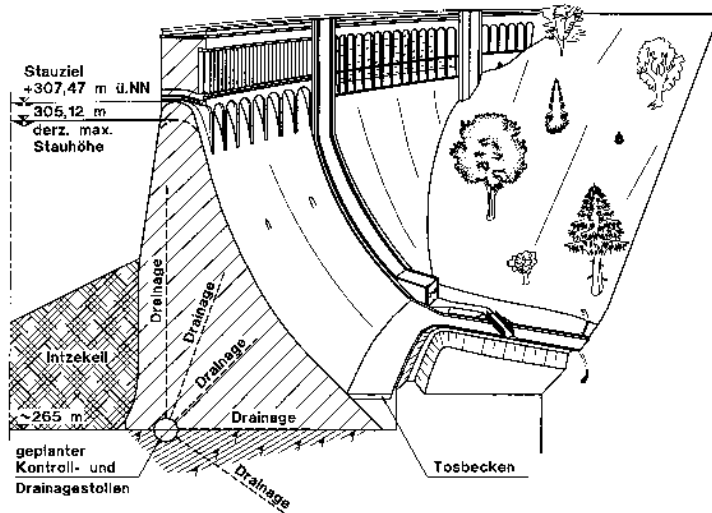
Fig. 163: Cross-section of a retaining wall of a dam

The discretization of this dam as a two-dimensional vertical model looks in SPRING like this:
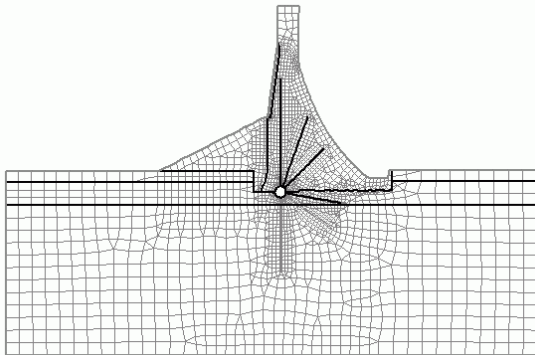
Abb. 164: Discretization of a vertical model in SPRING

To generate the 3D mesh, the 2D mesh is projected in the third dimension ("backwards"):
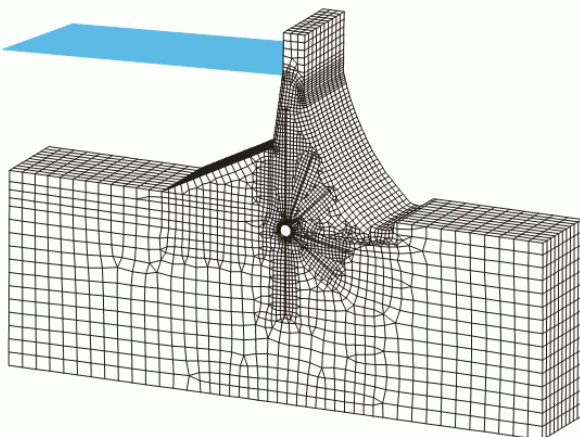
Fig. 165: Discretization of vertical 3D model (model type V-3D)

The corresponding boundary conditions in the present case look like this:
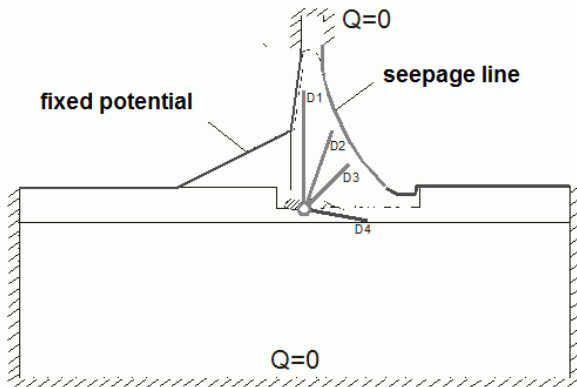
Fig. 166: Boundary conditions of a dam

5.4.1 Special features in the 2D vertical model

In the vertical model describes the X coordinate the position on the section line. The Y-coordinate describes the layer height. The specification of an areal recharge rate (attribute FLAE) in this model type makes no sense, but instead the attributes RAND or RANX are used. The thickness (attribute MAEC) describes the depth perpendicular to the vertical mesh.

The two figures below show how the coordinate systems are defined in reality (left) and in SPRING in the vertical model (right):
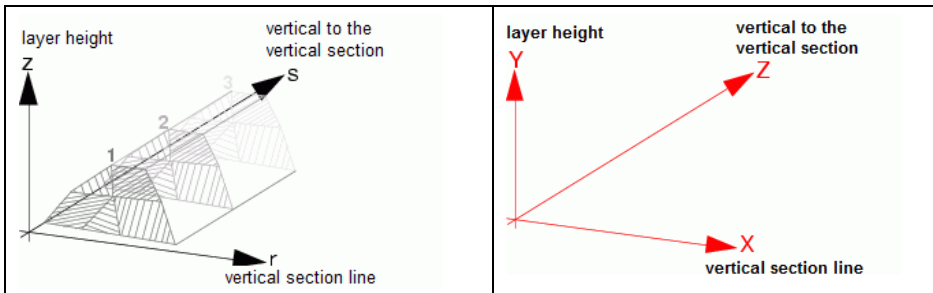
Fig. 167: Coordinate systems in reality (left) and in a vertical model in SPRING (right)

5.4.1.1   Creating a 2D vertical model from an existing 3D model

In SPRING, it is possible to "extract" a 2D vertical model from an existing 3D horizontal model.

For this, the background files of the 3D model as well as a previously created file *name.bpl* of a vertical section matching the model are required. From that, the information of the node coordinates is taken. With the following command used in a DOS shell:

> *plogeo -mesh name.bpl*

the 2D vertical mesh is created. By assigning the attribute MAEC and the new assignment of any necessary flow-relevant attributes the 2D vertical model will be complete.

5.4.2 Extending to three dimensions

Perpendicular to the intersection, the 2D-vertical model is extended into the third dimension. Then the thickness is divided into 3D layers (*Mesh → 3D → Divide layer...*), so that the Z-coordinate (ZKOR) represents the depth. In contrast to horizontal 3D models, the Z-coordinates of the node layers of a vertical 3D model lay side by side and not above each other.

It follows that the Z-coordinates in the 3D vertical model increase from top to bottom, whereas in a 3D horizontal model they are becoming smaller with increasing layer number.

Example 3D vertical model: GELA (= 0.0), ZKOR2 = 1.0, ZKOR3 = 2.0, ZKOR4 = 4.0, etc. Not taking this into account leads to an error in the calculation!

The following two figures show the different layer arrangement and the associated coordinate system:
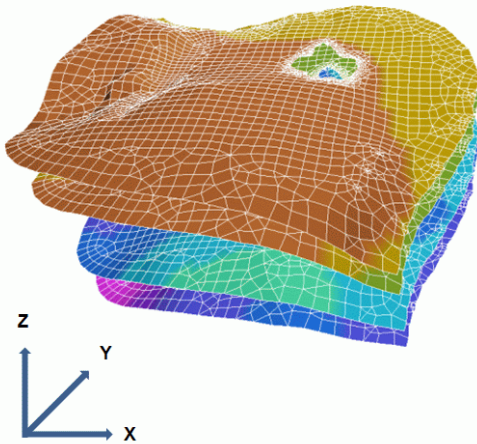
Fig. 168:
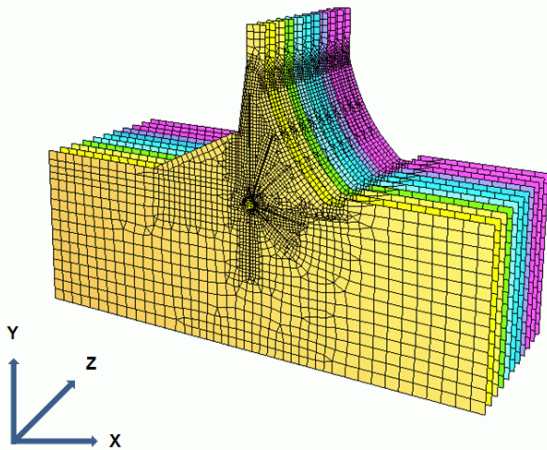
Fig. 169: Layer and coordinate axes in a vertical 3D model

The first node layer of the Z-coordinates (= GELA) lies on the vertical section line and has the value GELA = 0.0. After saving the finished 3D model the model type in the model file is set to "V-3D" with an editor. This is necessary because only the user (and not the software) can decide whether it is a 3D horizontal or a 3D vertical model. Since a 3D vertical model is a very specialized application, no query is done during saving the 3D model which type you have. This manual change of the model type is needed only once.

5.4.3 Presentation of sections in a 3D vertical model

The usual presentation of data and finite element mesh is done in 3D models in a top view/map presentation or in a vertical view / cross section presentation (Plot generation). These are presented for 3D vertical models in SPRING as follows:

5.4.3.1 Top view / map presentation in a 3D vertical model

Should the data of a deeper layer be presented in a 3D vertical model, corresponds this to the top view ("horizontal section") in a horizontal 3D model. The FE mesh is identical in these sections!
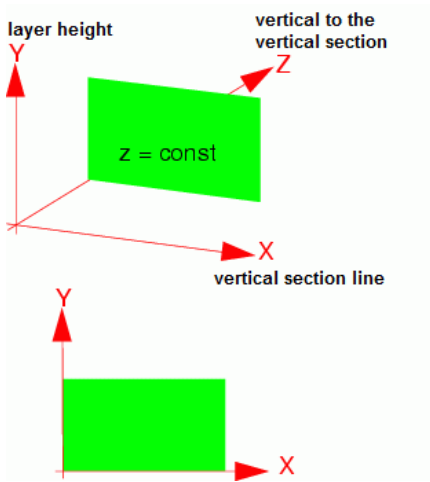
Fig. 8: Section parallel to the X-axis
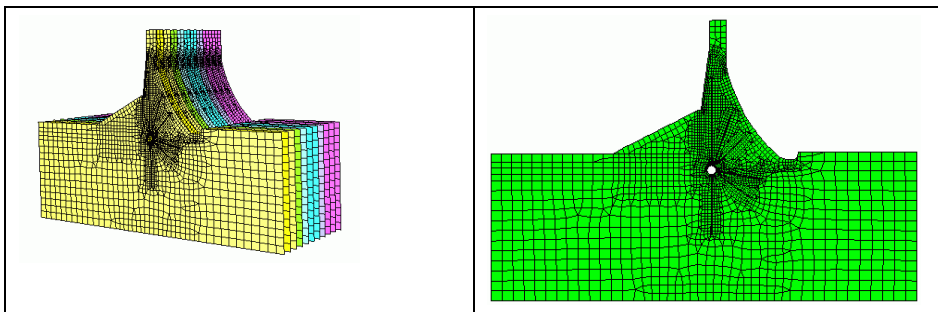
Fig. 170: Section parallel to the X-axis in SPRING

The specification of the layer number "4" in the *Plot generation → Top view / map presentation* (green layer in the left figure) results in the top view of the data of the 4.th layer in the X-Y coordinate plane (right figure).

5.4.3.2 View / cross section presentation in a 3D vertical model

If you want to create a section in the X-Z or Y-Z plane in a vertical 3D model, this corresponds to the view ("vertical section") in a horizontal 3D model. Which coordinate plane is shown reveals itself by the position of the intersection line defined by the user.

A section in the X-Z plane looks theoretically as follows:
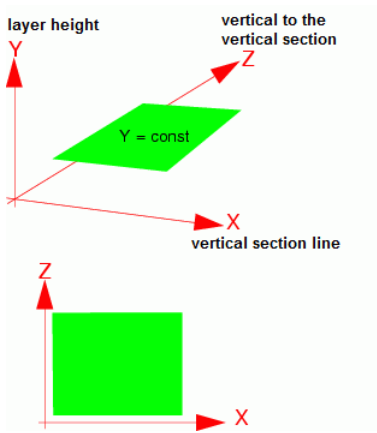
Fig. 9: Presentation in the X-Z plane

In SPRING it is implemented in the *Plot generation → View / cross section presentation* by choosing a section line that runs approximately parallel to the X-axis (green line in the upper figure). The result (lower figure) is a representation of the data (here: Z-coordinates of 0 to 2 m) in the X-Z plane ("view from above at the dam").
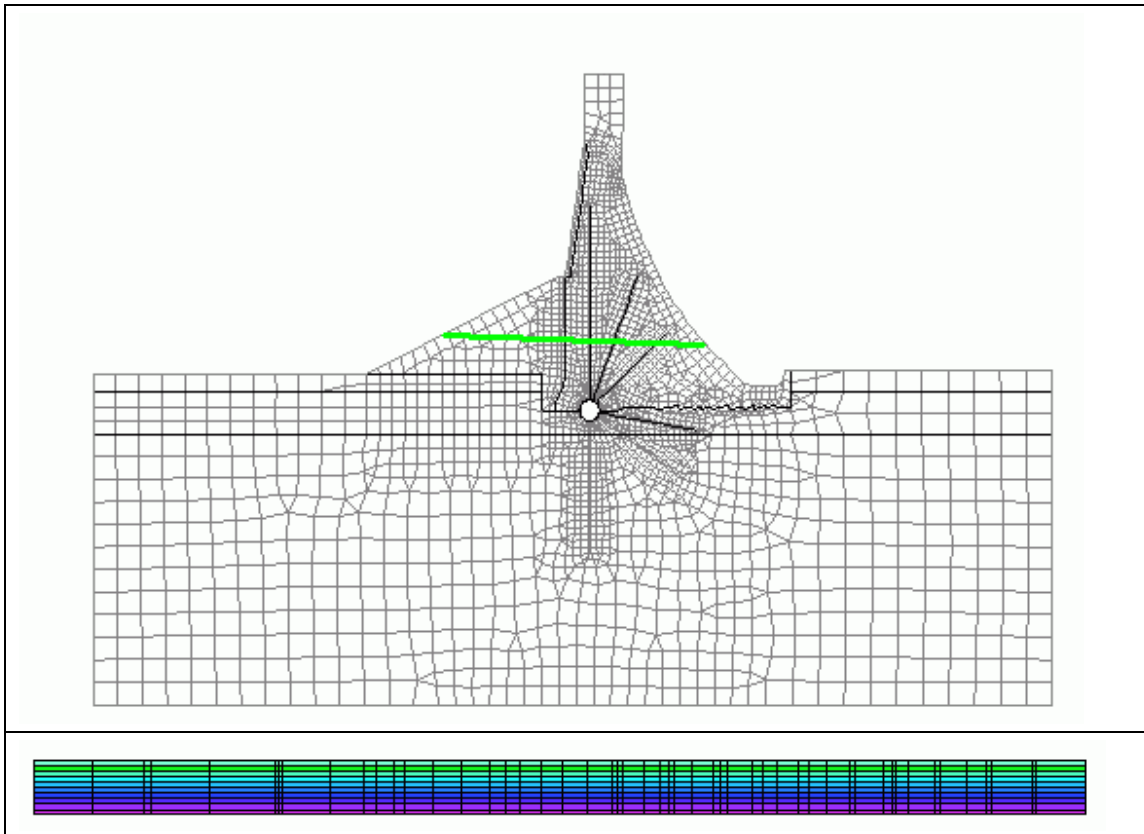
Fig. 171: Presentation in the X-Z plane in SPRING

A section in the Y-Z plane looks theoretically as follows:
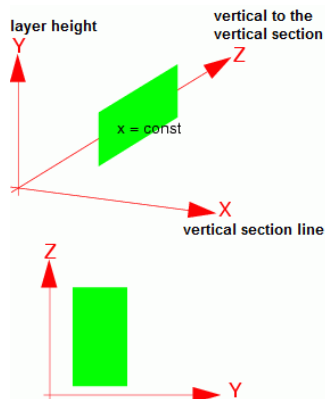
Fig. 10: Presentation in the Y-Z plane

In SPRING it is implemented in the *Plot generation → View / cross section presentation* by choosing a section line that runs approximately vertical to the X-axis (green line in the left figure). The result (right figure) is a representation of the data (here: K-values) in the Y-Z plane ("view from the right at the dam"). For a better understanding the resulting plot has been rotated 90 degrees to the left.
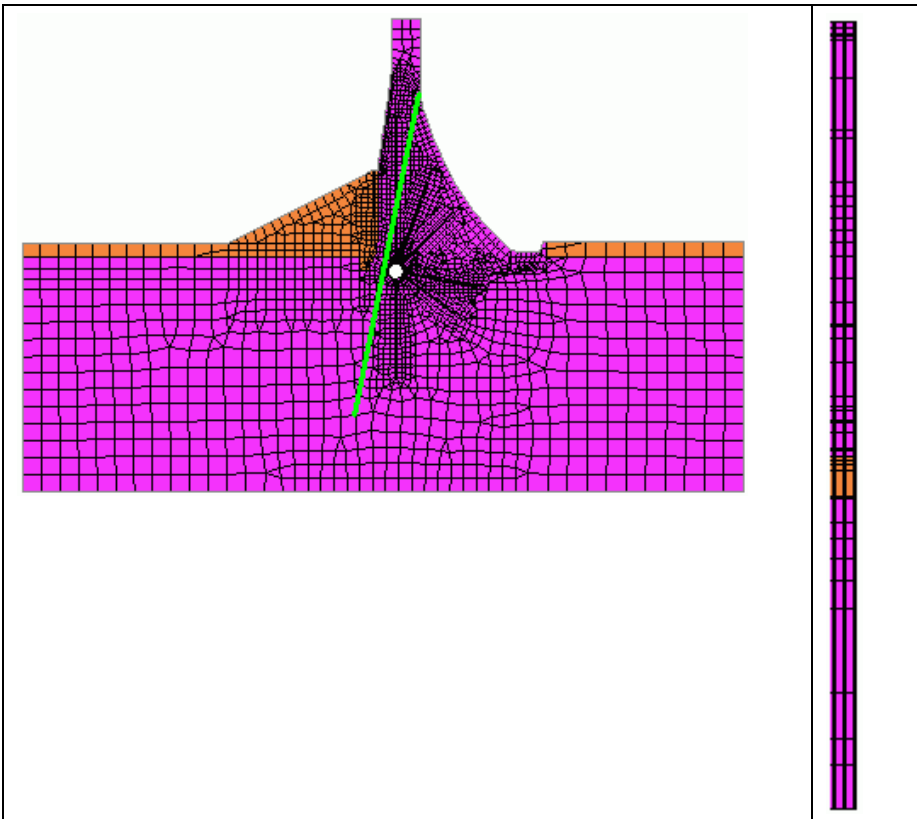
Fig. 172: Presentation in the Y-Z plane in SPRING

5.4.4 Special features in the plot generation

In the presentation of model and result data of a vertical 3D model the different allocation of the coordinate axes has to be considered. Moreover, not all sections are shown as you might intuitively be expected.

Since the usual presentation of the free surface does not make sense in a vertical 3D model, the presentation of the free surface in a vertical 3D model requires a detour on the following procedure:

- The Y-coordinates (= layer height) are saved as the attribute KKKK (*Attributes → Copy → Attribute-wise...*).
- Then the attribute KKKK is subtracted of the calculated potentials (*Attributes → Import model / result data...*) and stored on any node attribute (*Attributes → Computation → Intercalculation…*). The free surface is obtained as the contour lines of this attribute with value = 0.0.
- Hiding of the results above the free surface is not possible.

6 Calculations

In this chapter, the different calculation modules are described in terms of input parameters and characteristics.

All calculation processes start in this menu:
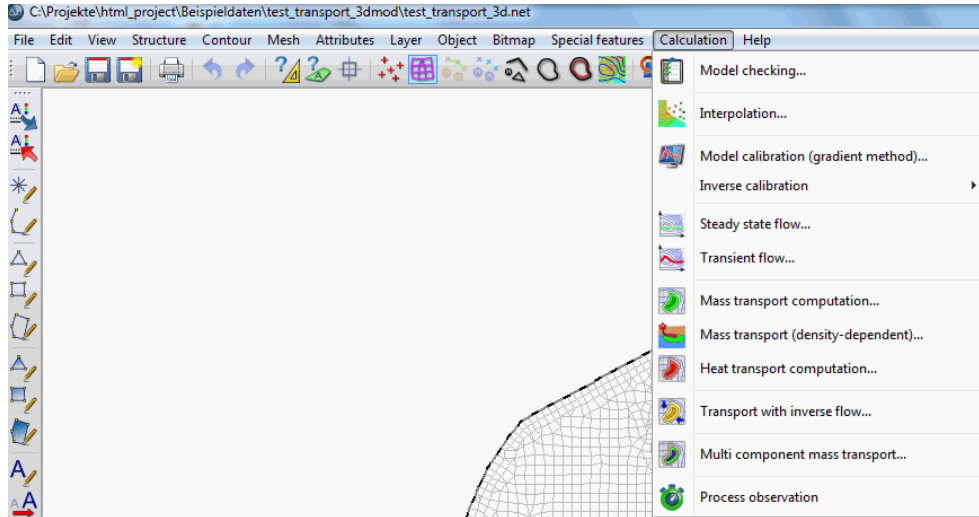
Multi component mass transport

This calculation module is currently being developed.

Process observer

During a transient calculation, it is possible to track the time progress of the calculated results live by starting the SPRING Process Observer. A detailed description is given in chapter: "Transient flow - Monitoring the time progress" (p. 302).

After entering the parameters and start the selected calculation module opens the SPRING Process Launcher. It shows the protocol of the calculation.

Remark: During the calculation, it is possible to work further on the current model!
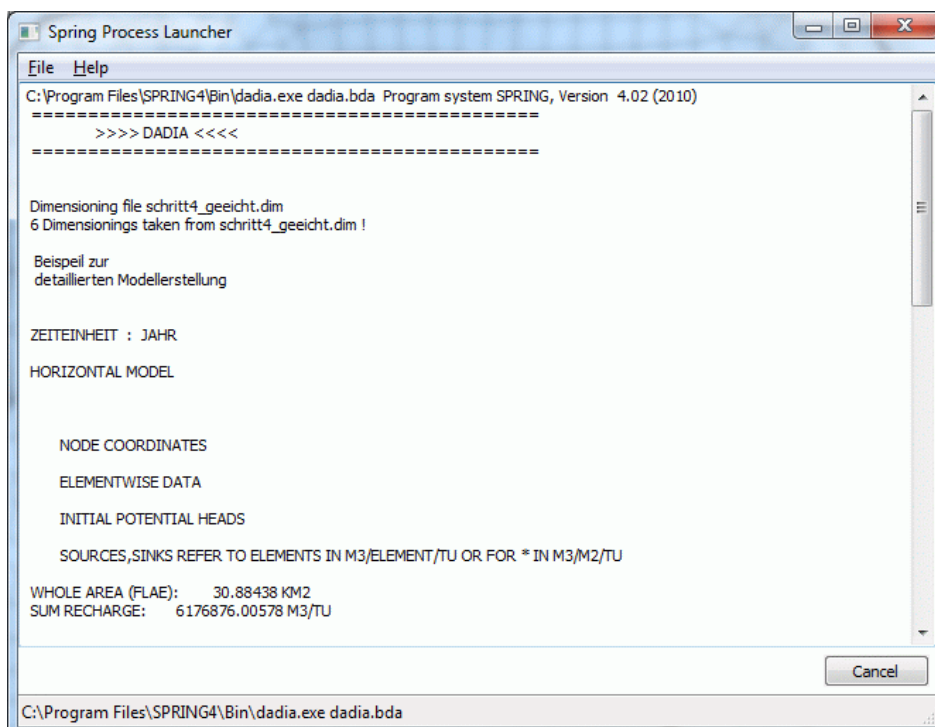
Fig. 173: Protocol window of the calculation

6.1 Model checking

Call via: *SPRING → Calculation → Model checking…*

This calculation module checks the input data and prepares them for the finite element calculation. It reads the *.net-file and, if applicable, the *.3d-file and other additional files (if provided). The program checks if the data are complete and (optionally) if they are plausible. All identified errors are recorded. Furthermore, a bandwidth optimization which is specialised for 3D models is carried out. The superfluous nodes and elements resulting from partial layers are eliminated. The prepared data are output into binary files for the subsequent calculations.

In the model checking internal calculations may be performed and values of input attributes are subsequently assigned.

If the *.net-file contains data for terrain settings as a result of subsurface mining (BERG), they will be considered automatically in the calculation. Hence the values are subtracted from the following data:

■   Ground level (GELA)
■   Lower boundaries (UNTE)
■   Upper boundaries (OBER)
■   Z-coordinates (ZKOR in 3D model)
■   Water course levels (VORF)

Terrain settings enlarge the water filled thickness.

Example:
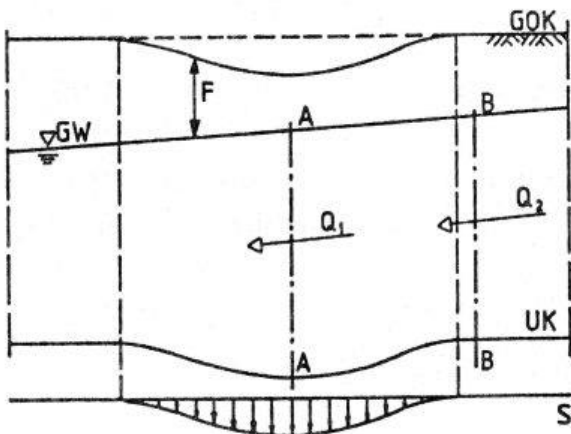
Fig. 174: Section of an aquifer influenced by subsidences

From section A – A to section B – B the continuity condition has to be satisfied ($Q_1 = Q_2$). In areas influenced by subsidences the water filled thickness is reduced so the flow condition changes. The setting of the ground level decreases the distance ground level – ground water level (F).

In order to keep the size of the input file as small as possible, in a 3D model the data of the higher layers are automatically assigned to the deeper layers, in case there are no data for these layers in the *.3d file. The following attributes are automatically assigned:

■   Initial potential heads (EICH)
■   Permeabilities (KWER, KWEV, KFVH)
■   Limitation of the permeabilities (KMIN, KMAX)
■   Dispersivities (DISP)
■   Porosities (PORO)
■   Storage coefficients (SPEI, KSPE)
■   Initial concentrations (AKON)
■   Anisotropic conductivities (Z-KA)
■   Compressibility of the entire system (KOMP)

The model checking makes it possible to use other modules in spite of errors in the input files. The errors are divided into possible mesh errors (no. > 0) and possible data errors (no. > 0). For instance, in case of data errors the mesh can be plotted (*File → Plot generation*) or data can be interpolated (*Calculation → Interpolation*). Absolutely necessary are the coordinates of the nodes and the element description.

Unknown data types are skipped, resulting in a warning being generated and recorded. The defined identifications KKKK for any node wise data, and EEEE for any element wise data, are useful for e.g. possible calculations between different input data or comparisons with results using the plot generation.

6.1.1 Input parameters

Attention: The project must always be stored before starting the model checking! Otherwise the last modifications are not considered!

Via the menu item *Calculation → Model checking* the following input window appears (for a 3D model):
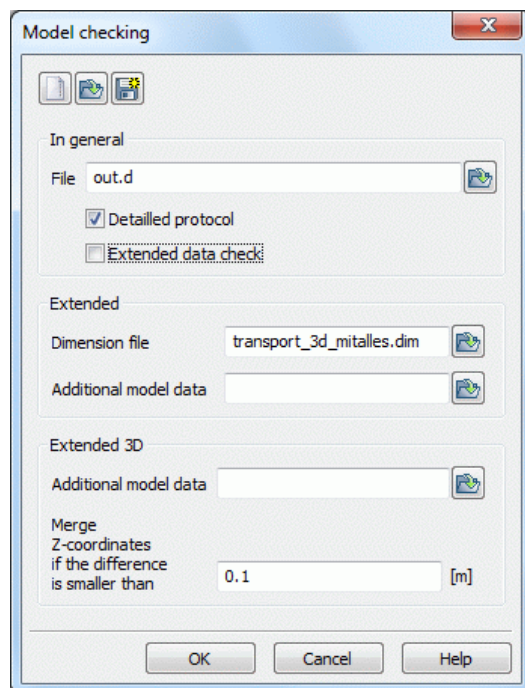
When calling the module, the program reads the batch file with the default name dadia.bda (if it exists), and the default settings in the mask may be changed accordingly.

In general

In the block "In General", the name of the *output file* (default: out.de) is entered and selected, whether a *Detailed Protocol* or an *Extended data check* is desired.

By default, the coordinates of nodes, the incidences and the data types (without values) are recorded in the output file. Activating the protocol leads to complete data sets in the output file with the identification and the corresponding values.

In the case of large models, a *detailed protocol* of the input data (Detailed check of input data) can significantly increase the computing time. That's why the user should activate this function only after generation of a completely new mesh, or after having made significant changes in the input data. By default, each time a minimal check of the mesh and the input data is carried out. The detailed check considers:

■ Geometry of the mesh (nodes in elements, intersection of elements ...)

■ Double node numbers

■ Fixed potential head and inflow / outflow rate as boundary condition for the same node

■ Completeness of data for the whole mesh

■ Check of data and plausibility controls, for most data types

Extended

In the block "Extended" the name of the *dimension file* (*.dim, generated automatically during the saving) and possibly an *additional file* to be considered has to be specified. These can either be entered directly or selected from any directory by pressing the button "File open".

The additional files have the same structure as the model files. Here, single or multiple data types can be specified to complete the inputs of the model files or (with the same node or element number) over write. In this way, e.g. various withdrawals can be separated from the rest and kept in different files.

This procedure is suitable when multiple variants are to be calculated.

Extended 3D

The block "Extended 3D" appears only during the model checking of a 3D model. Here you are asked for the name of the possibly to be considered (three dimensional) additional file.

At last, the minimum difference of z-coordinates is defined. In a 3D model nodes are merged together if the vertical distance of their z-coordinates is equal or less than the defined value.

The buttons at the top of the input window allow resetting the input parameters ( ), opening an existing batch file ( ) or saving the current batch file under another name ( ).

The buttons at the bottom of the input window start the calculation (*OK button*), close the dialog box (*Cancel*) or open the digital manual (*help button*).

6.1.2 Batch file model checking

It is possible to execute the *model checking* directly in the command line without using the dialog box, if a batch-file *. bda* exists in the directory. For this you have to change to the directory in which the calculation is to be executed. The input of *dadia* and confirming with the Enter key starts the model checking. The default file name is *dadia.bda.* This file is searched in the directory.

To use another batch file, type *dadia file name* (the extension ".bda" is automatically attached). Batch files can be created, viewed and modified with the help of any text editor.

A detailed batch file can be viewed in the online help. Similarly, there is a batch file of the model checking for further processing provided.

6.2 Interpolation

Please note that there are two interpolation modules available in SPRING:

■   Interpol, a separate interpolation application accessible from the SPRING menu (*Calculation → Interpolation*)

■   SPRING Interpolation: implemented inside SPRING, which is accessible from the SPRING menu (*Attributes → Assign → Interpolation*)

In principle, the interpolation makes a difference between two types of data interpolation:

■   Area Interpolation from data at nodes or elements: Different interpolation algorithms can be used for interpolation of values (z_i) at points (x_i,y_i) to all nodes or elements of the FE-model (in case of 3D models only to the topmost layer). Additional Information can be integrated to the interpolation. In the special case of the Interpolation of a groundwater contour map multiple functions are implemented to add information - already stored in the FE-model - to the interpolation process.

■   Linear interpolation from scatter points (also transient data / curves) to all nodes along a defined node track.

In the following figure are the different interpolation algorithms compared to each other.

Element mesh with interpolation points	Nearest Neighbour method
Distance weighting (500+2)	Area interpolation (1 pass)
Gauß interpolation	Kriging (Fitting polynom)
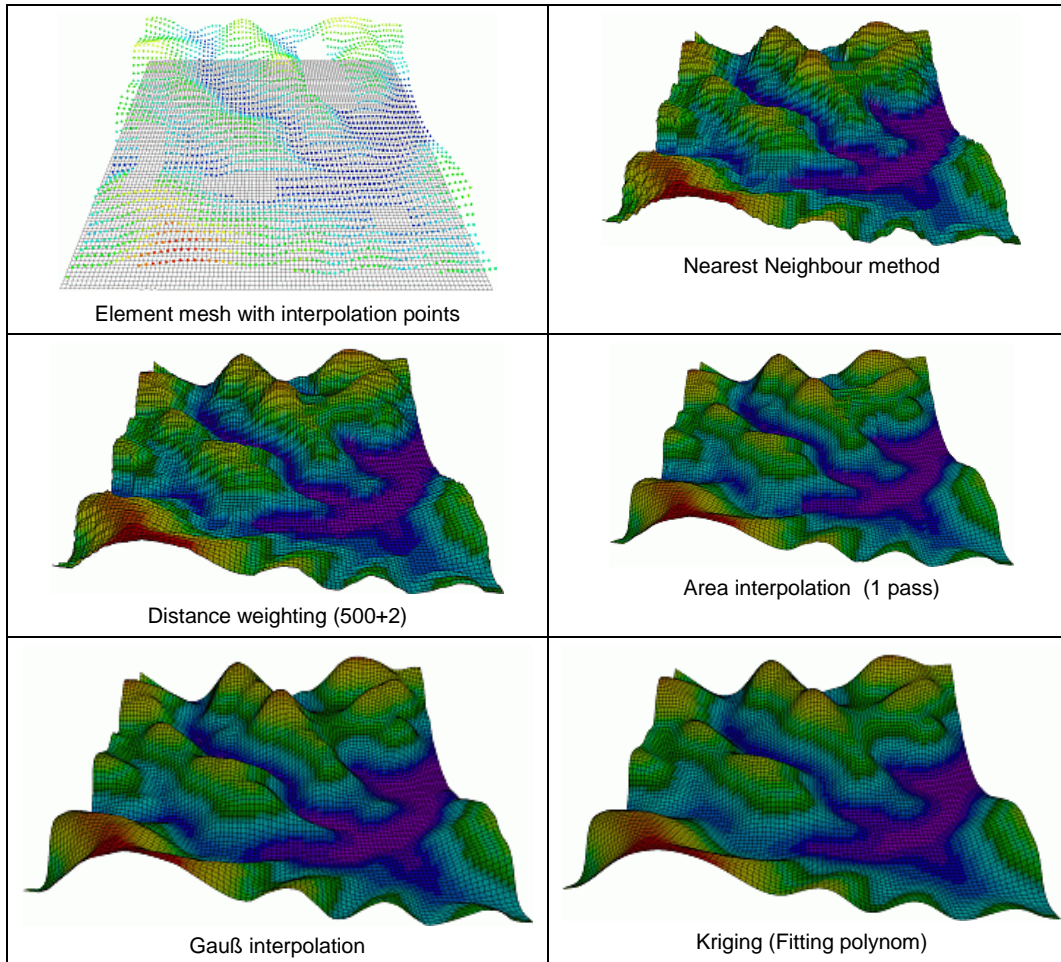
The interpolation algorithm „Nearest Neighbour " is provided only in the SPRING menu *Attributes* (*Attributes → Assign → By interpolation…*), whereas the interpolation methods distance weighting, area and Gaussian interpolation are provided as well in the attribute menu as in the calculation menu (C*alculation → Interpolation…*). The interpolation method Kriging is only available in the calculation modul *Interpolation*.

In general, the SPRING interpolation should be preferred as it includes the latest implementations.

6.2.1 Input Parameters of the Calculation Modul Interpol

Clicking on the menu item *Calculation → Interpolation…,* the following dialog box appears:
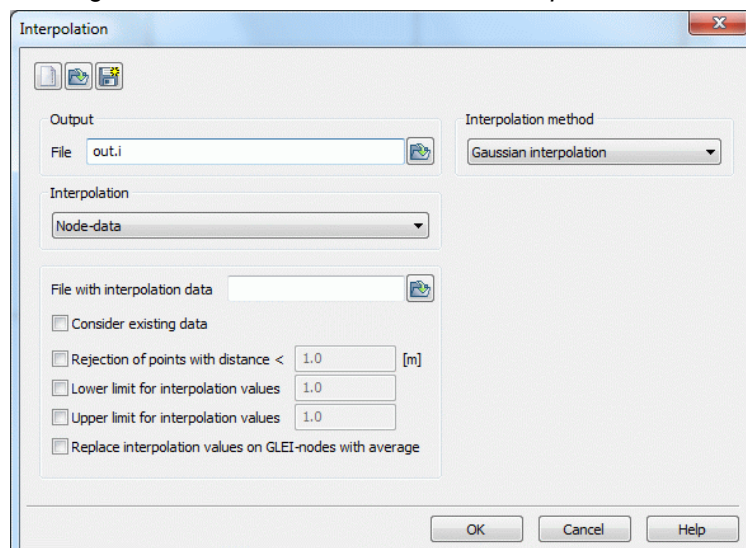
When opening the dialog box, the program reads the batch file with the default name *inter.bip*, if it exists. Then the default settings in the dialog box may be changed accordingly.

Output

The text box requires a name for the *output file* (default: out.i).

Interpolation

Here the user decides whether the interpolation is done

- to node data
- to element data (interpolation to the centres of the elements)
- for the groundwater contour map (only available in a 2D model)
- to a node track

The choice of the interpolation data determines the future appearance of the input window.

When choosing "groundwater contour map" more input possibilities open whereas when choosing "node track" the block of the interpolation algorithm is closed.

Additional data

In this block information is given about the data to be included in the interpolation.

- *File with interpolation data*:

  The name of the file with the additional interpolation data is required. It can be typed manually or chosen from the file list (activate the button on the left of the text box). The corresponding file contains only lines with:

      Number, x-coordinate, y-coordinate, measured value

  When using the format I6, 3F10.2 (equivalent to characteristic data). blank lines between the data are not allowed.

- Consider existing data:

  The interpolation can consider data already existing in the model file, e.g. potential heads. In such cases, activating the button causes a selection list of existing and anticipated data to appear, from which the user can choose. Pay attention to the units of the data: they can be different from the units in the model file. This selection is only available if node or element data should be interpolated.

- Rejection of interpolation points with *(minimum) distance*:

  Data points that are spaced closer than the defined limit with respect to other data points are eliminated. (The interpolation algorithms Gauss and Kriging and the interpolation of a contour map cannot handle double coordinates!)

- *Lower and upper limit* for interpolation values:

  The resulting interpolation values can be limited upwards and / or downwards. For instance, terrain settlements as a result of subsurface mining (attribute BERG) should not be less than 0.

- Replace interpolation values on GLEI-nodes with average:

  In case of interpolation of potential heads from measurements, surface waters connected to the aquifer can be taken into account. With the help of GLEI, the interpolation algorithm allows to use an average value for equal potential heads at the surface waters. This menu item does not apply to the interpolation of element data or if the attribute GLEI is not assigned.

- For the interpolation of a groundwater contour map additional auxiliary nodes can be considered. The procedure is described in detail in chapter 6.33.3.

Interpolation method

Here, the interpolation method is selected that will be used. The following algorithms are available:

- Distance Weighting Area Interpolation Kriging Gaussian InterpolationBy activating one of the algorithms more input options open up that are described in more detail in the chapters below.

The buttons in the header of the input window allow the resetting of the input parameters ( ), the opening of an existing batch file ( ) or the saving of the actual batch file under another name ( ).

The buttons at the bottom of the input window start the calculation (*OK button*), close the dialog box (*Cancel*) or open the digital manual (*help button*).

6.2.1.1 Distance Weighting

The following input options appear:
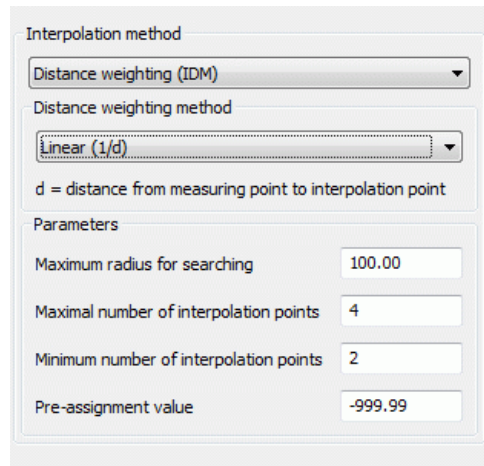
Fig. 175: Input for distance weighting

The measured value is weighted depending on the distance d to the unknown node or element centre. For the weighting process, four different functions are available:

- Based on Sampson (see in the following),
- linear (1/d),
- squared (1/d²),
- 4th degree (1/d^4),

where d represents the distance between the measuring point and the interpolation point.

The user defines the *maximum radius* where measuring points are searched, the *minimum and maximum number* of measuring points used in each interpolation. If the defined radius does not contain enough measuring points, the value at the corresponding node or element is not interpolated. For these cases, the user defines a *pre-assignment value*.

Remark: In contrast, when interpolating data in the menu *Attributes → Assign → By interpolation*... nodes or elements for which were not enough interpolation points found get no value.

With the linear function, the influence of the measuring point decreases linearly with increasing distance. Having linear transitions between adjacent measuring points effects very rough interpolation areas. For a 4th-degree weighting, the influence of the point decreases with distance to the 4th power. Hence it follows that the value of the closest measuring point is the most significant one. The algorithm produces very smooth interpolation areas which are slightly graded around the measuring points. Squared weighting averages the two extreme values.

The algorithm based on SAMPSON (1978) was published as a "constrained distance-squared weighting", and follows the weighting function

$$(1 - d/(1,1 * d_{max}))^2 / (d/(1,1 * d_{max})^2$$

Compared to the other possibilities, this procedure delivers the best results.

6.2.1.2 Kriging

The method according to Kriging is a generic term for stochastic interpolation as opposed to deterministic methods (e.g. distance weighting).

The Kriging algorithm determines the value in an interpolation point (x, y) by weighting all measured data:

$$\mathbf{z = z(x, y) = \lambda_1(x, y)z_1 + \lambda_2(x, y)z_2 + ... + \lambda_n(x, y)z_n}$$

Hence z_i are the measured values of the n measuring points (x_i, y_i), i=1,...n.

The weighting factors $\lambda_i(x, y)$ not only depend on the distance between measuring and interpolation points but, corresponding to the similarity of the measured data, are computed in an equation system.

6.2.1.2.1   Creating a Variogram for the Kriging Algorithm

The most important step in an interpolation with the Kriging algorithm is to create a variogram. This shows the difference between two measured values versus the distance between the corresponding measuring points. The resulting statistical information is approximated by a variance function.

The first step creates a so-called experimental variogram. For that, distance classifications (distances in m) are defined:

$$\mathbf{K_k = [h_{k-1}, h_k], k = 1,...,i} \text{ with } \mathbf{0,0 = h_0 \leq h_1 \leq h_2 ... \leq h_i}$$

Inside each class, a semivariance $g_k(h)$ is computed from the measured values of all pairs of measuring points with the corresponding distance between:

$$\mathbf{g_k(h) = \frac{1}{2*m_k} \sum_1^{m_k} (z_i(x_i, y_i) - z_j(x_j, y_j))^2} \text{ with } \mathbf{\left|(x_i, y_i) - (x_j, y_j)\right| \in K_k}$$

where m_k represents the number of pairs (i, j) whose distance is part of the class K_k.

z_j denotes a point that has a distance h to the point z_i.

The user can define a maximum of 16 different distance classes. For each of them a table containing the significant variogram characteristics is output:

  Column 1: class number k,

  Column 2: maximum distance h_k,

  Column 3: average distance h of all pairs in this class,

  Column 4: semivariance $g_k(h)$

  Column 5: number of measuring point pairs in this class m_k,

  Column 6 and 7: *imax* and *jmax*, numbers of the two data points for which the difference between the values *z(imax)* and *z(jmax)* is maximum

  Column 8: *d(z(imax),z(jmax)) = 0.5 * (z(imax)-z(jmax))²*, comparative value for the variance $g_k(h)$

E. g. the table for an equidistant partition into 10 classes has the following appearance:

Output of variogram parameters in table form

Output in table form:

	upper limit of the class [m]	Average distance	g(h)	Number points	imax	jmax	d(Z(imax),Z(jmax))
1	13.5197	11.0312	78.625	2	8	13	156.645
2	27.0394	22.319	23.3786	32	8	9	169.28
3	40.5591	34.3009	32.9698	40	7	8	151.38
4	54.0788	47.297	32.4282	46	4	13	118.58
5	67.5985	59.5892	42.9031	45	8	24	162
6	81.1182	74.2137	37.7159	56	8	25	172.98
7	94.6379	87.2678	31.2511	32	21	25	137.78
8	108.158	99.8825	36.8714	33	9	21	134.48
9	121.677	113.55	28.3105	11	10	21	84.5
10	135.197	128.863	22.9883	3	1	25	60.5

Fig. 176: Tabular output of the variogram parameters

Additionally, to enable better comparison, the maximum distance between two data points is recorded.

The selected partition should guarantee that each class contains a minimum of one pair of data points. If possible, the number of pairs should have the same order of magnitude for all classes. If one pair of data points cannot be classified, the maximum distance of the last class is increased to the necessary value.

The experimental variogram shows the computed variances $g_k(h)$ (column 4 of the table → y-axis) versus the average distances h (column 3 of the table → x-axis).

Limit of the x-axis is the calculated maximum distance between the measuring points to each other

Hence a function is to be found that best approximates this experimental variogram. This function is then used to compute the weighting factors $\lambda_i(x, y)$ for the interpolation.

SPRING offers the following six types of functions as fitting functions:

■ Variogram h^A

■ Spherical variogram

■ Exponential variogram

■ Gaussian variogram

■ Cubic polynomial

■ Fitting polynomial

The types from (1) to (5) require the parameters PEP, PI and A:

■ PEP >= 0: so called 'nugget effect': Measure of the variability of the measured value (measurement error) g(0)=PEP

■ PI >= 0: Variance of the observed values

■ A >= 0: size of the region whose points are used for interpolation, for the types (2) to (5) the parameter A should be approximately the maximum distance

The following representation shows the meaning of each parameter schematically:
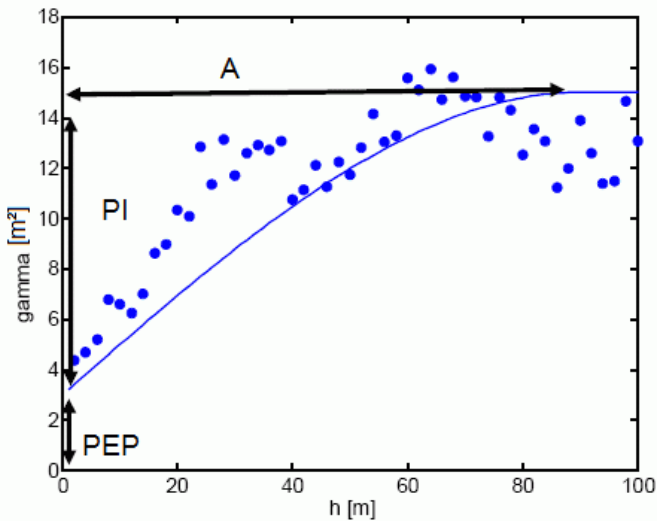
Fig. 177: Schematic representation of the variogram parameters

The following representations show the experimental variogram (red point) overlayed with the chosen function (blue) and the corresponding data input in SPRING for the different fitting functions.

(1) Variogram h^A:

The fitting function for this type of variogram is:

$g(h) = PEP + PI * h^A$

As a function of the parameter A there are three curves of the variogram possible:

Case 1: A = 1 (straight line)
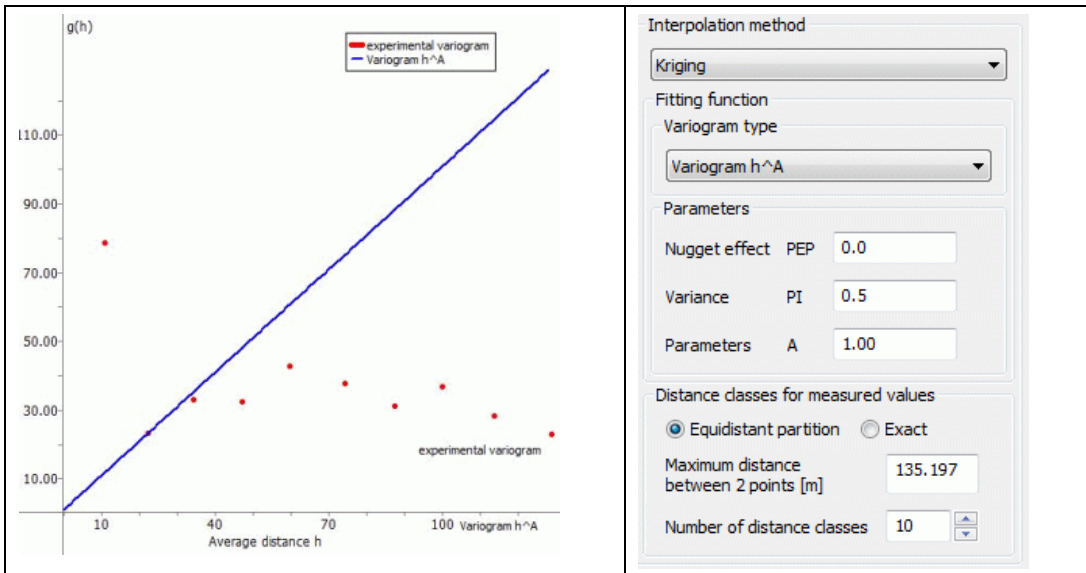
Case 2: A < 1
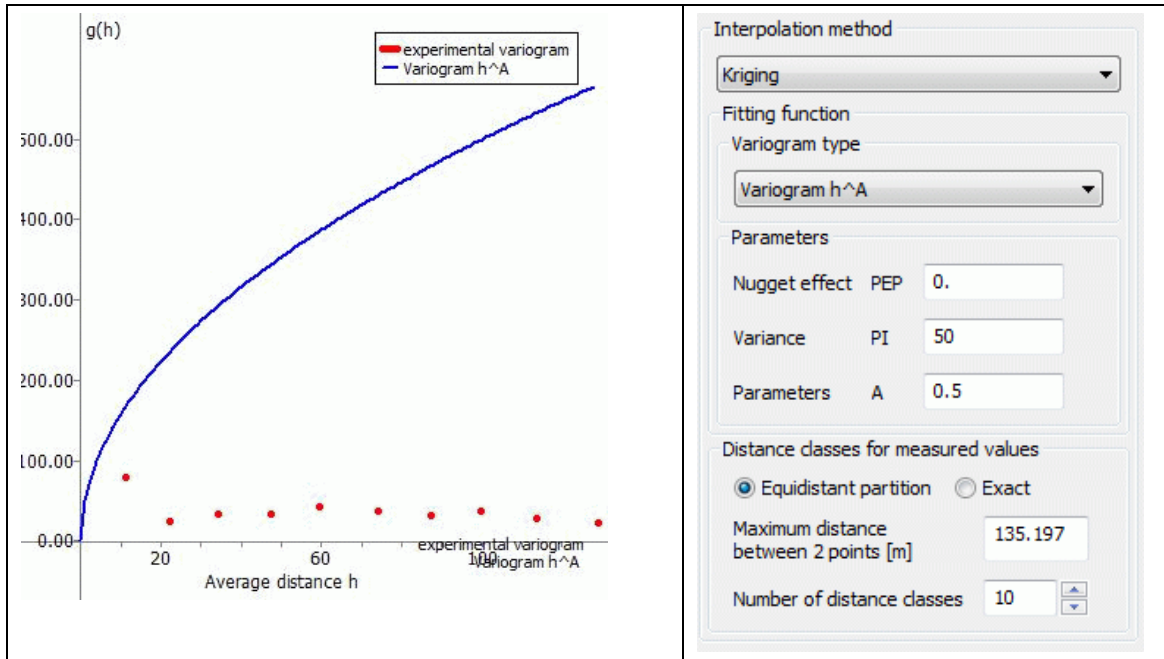
Case 3: A > 1
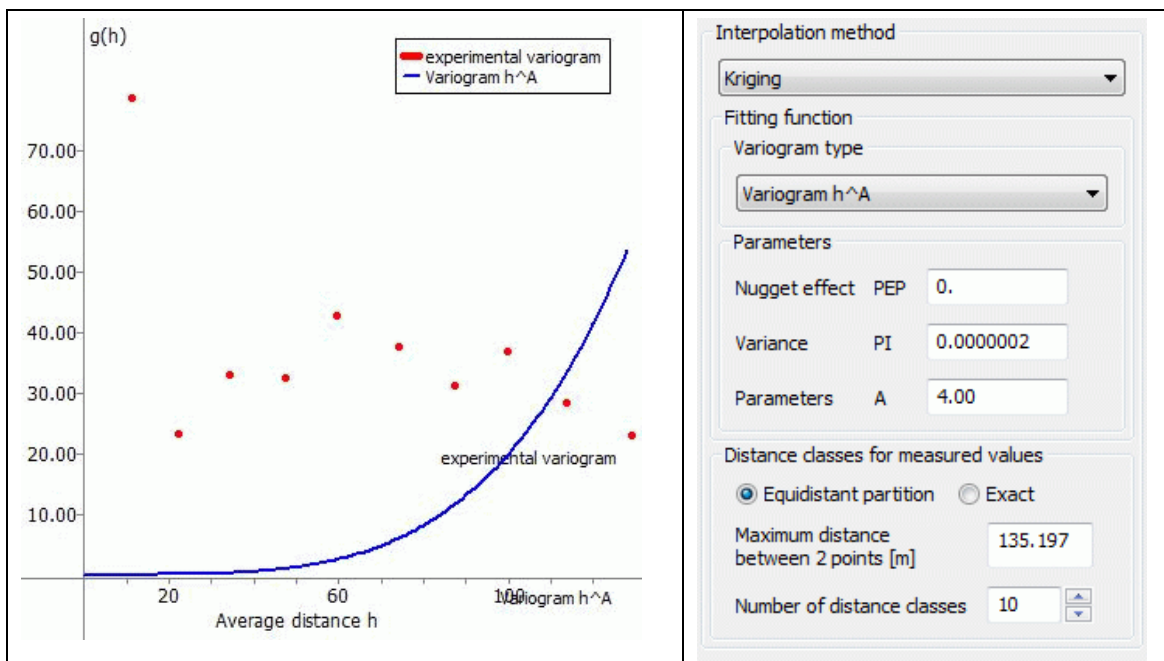
(2) Spherical variogram:

The fitting function for this type of variogram is:

$$g(h) = \begin{cases} PEP + PI \cdot (1{,}5 \cdot H - 0{,}5 \cdot H^3) \ for \ H = \dfrac{h}{A}, H < 1 \\ PEP + PI \qquad for \ H = \dfrac{h}{A}, H > 1 \end{cases}$$
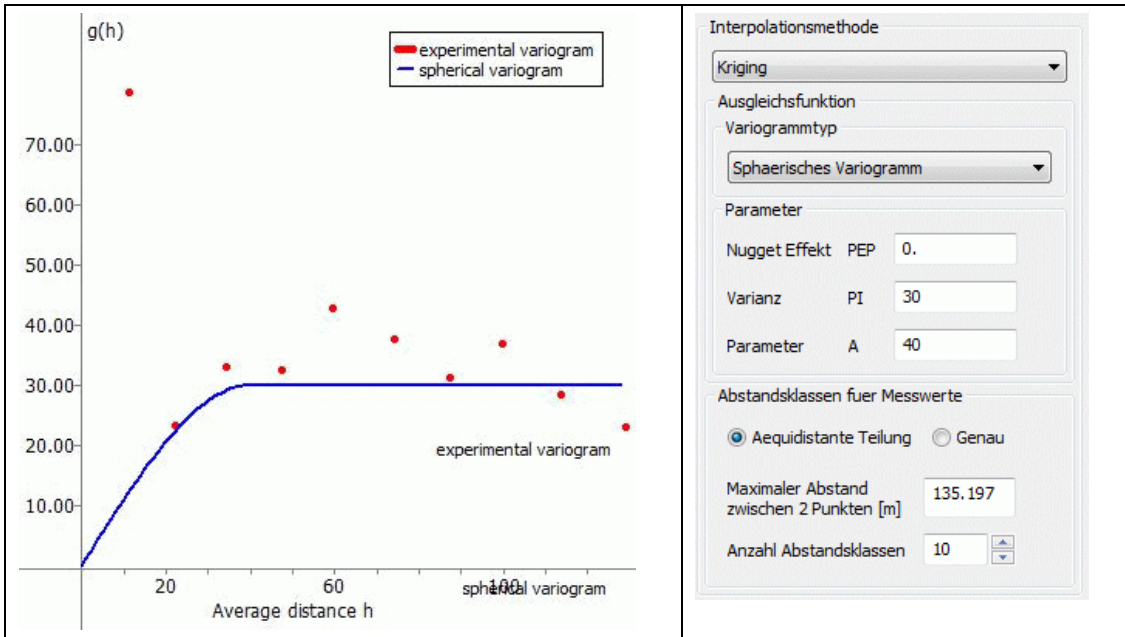
(3) Exponential variogram:

The fitting function for this type of variogram is:

$$g(h) = PEP + PI * (1-exp(-H)) , \text{ with } H = h/A$$
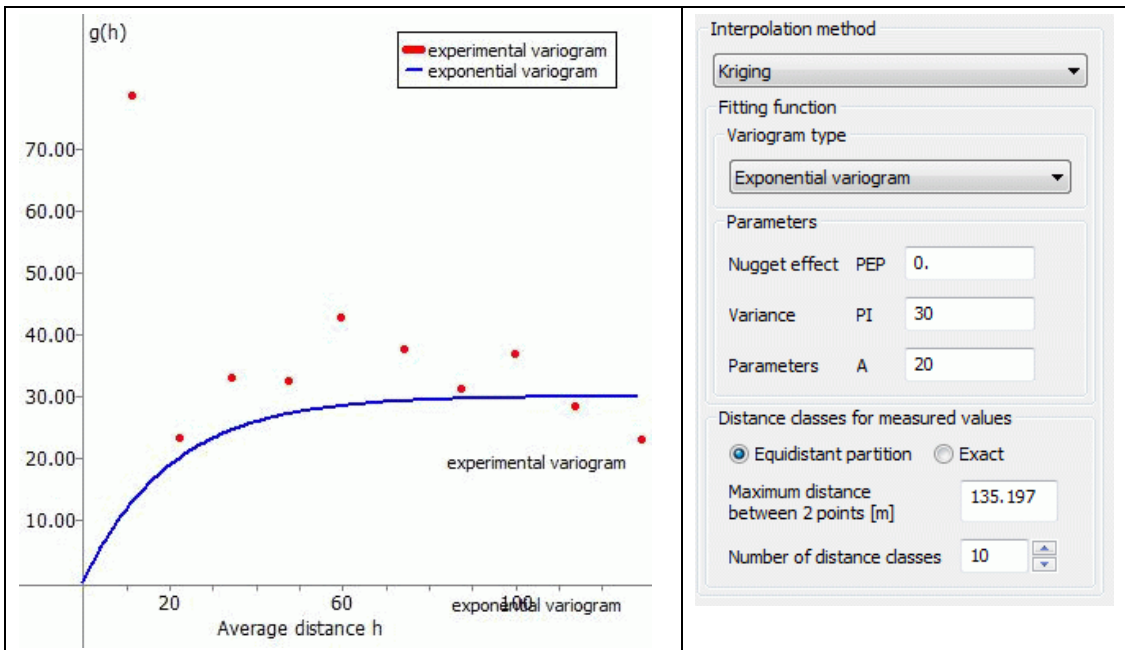
(4) Gaussian variogram:

The fitting function for this type of variogram is:

$$g(h) = PEP + PI *(1-exp(-H^2)), \text{ with } H = h/A$$
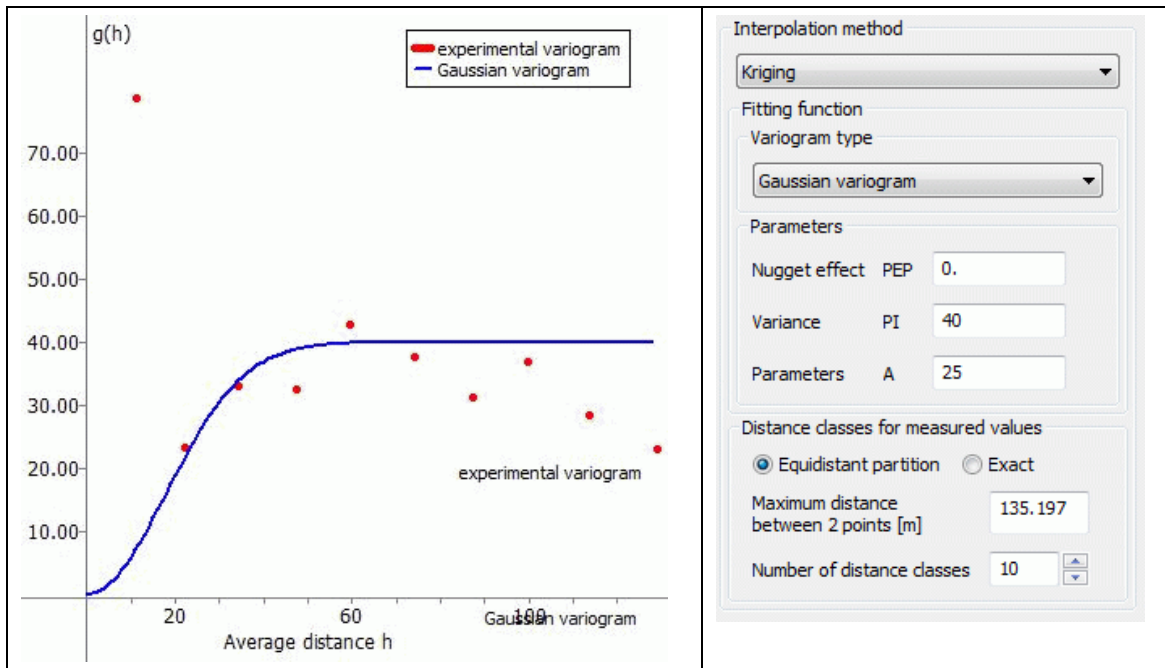
(5) Cubic polynomial:

The fitting function for this type of variogram is:

$$g(h) = \begin{cases} PEP + PI \cdot (7 \cdot H^2 - 8,75 \cdot H^3 + 3,5 \cdot H^5 - 0,75 \cdot H^7) & for \ H = \dfrac{h}{A}, H < 1 \\[2em] PEP + PI & for \ H = \dfrac{h}{A}, H > 1 \end{cases}$$
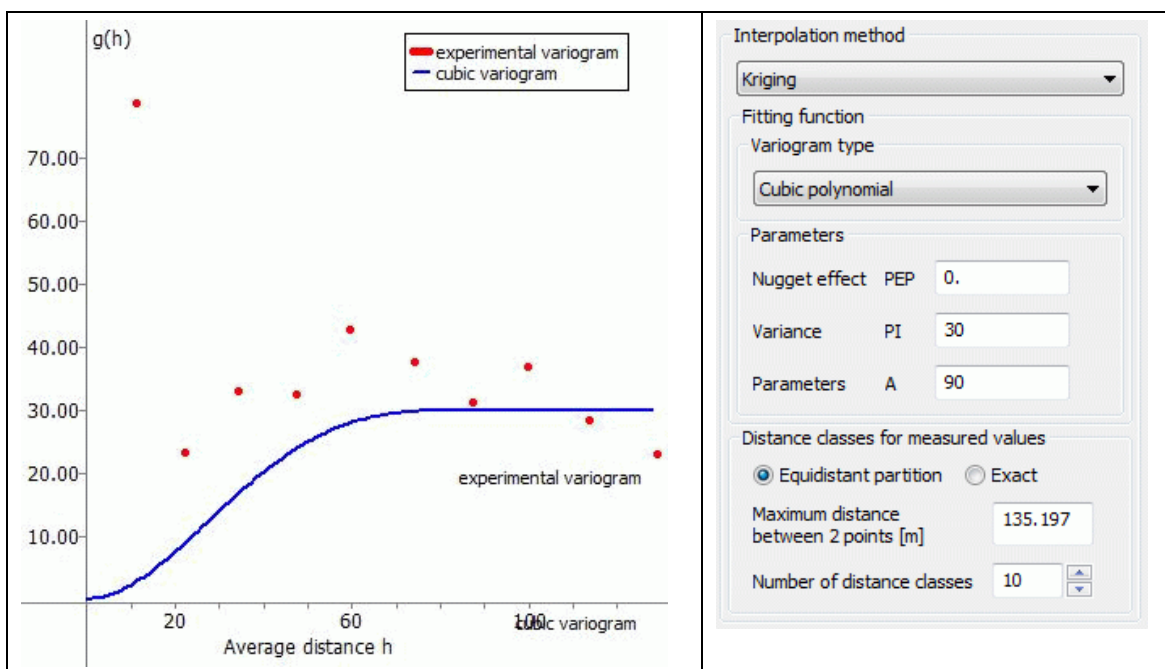
(6) Fitting polynomial form of a degree between 1 and 9

If the user selects type a fitting polynomial form, the degree of the polynomial (between 1 and 9) and the number of approximation points (for which the maximum is the number of classes) have to be defined.

The given example produces the following fitting polynomial form:
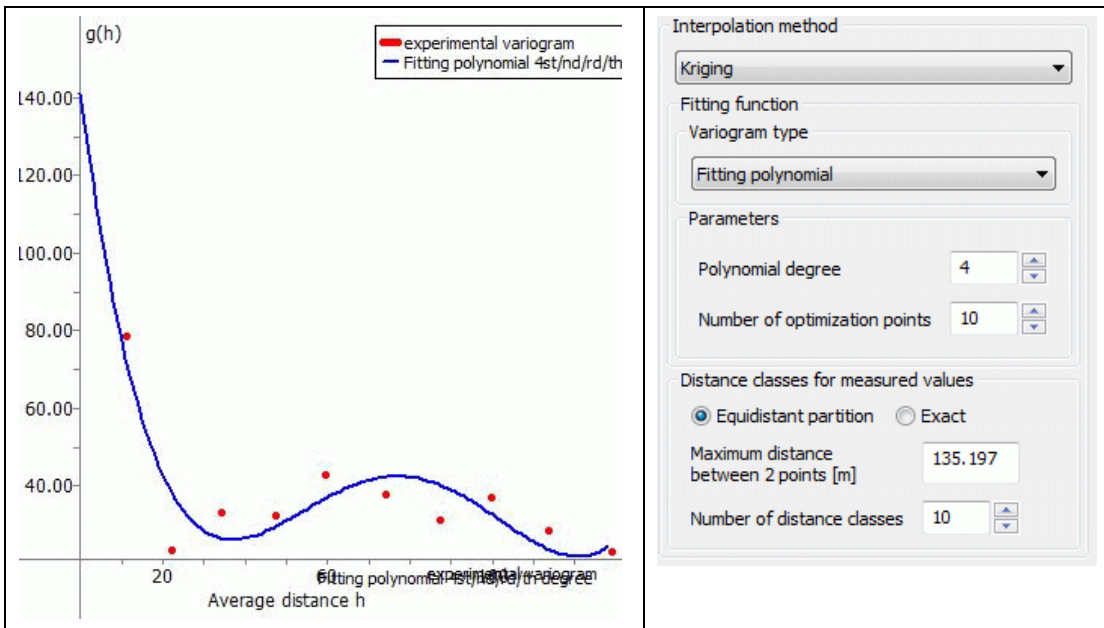
In general, the resulting functions should be increasing monotonously, and should approximate the given values as closely as possible. The quality of the variance function g(h) significantly influences the quality of the interpolated values using the Kriging procedure.

The creation of the variogram is the most important step, because of the high parameter sensitivity of the Kriging interpolation.

6.2.1.2.2   Kriging Interpolation

Using the variance function g(h), the weighting factors $\lambda_i(x, y)$ at the interpolation point (x,y) are determined, such that the following ('moving average equation') holds

$$\sum_{j=1}^{n} \lambda_j(x,y) * g\big(\|(x_j,y_j)-(x_i,y_i)\|\big) = g\big(\|(x,y)-(x_i,y_i)\|\big) - \lambda(x,y) \quad \text{with i = 1,...n}$$

with

$$\sum_{i=1}^{n} \lambda_i(x,y) = 1$$

This leads to a $((n+1) \times (n+1))$ dimensional equation system for the weighting factors $\lambda_i(x, y)$, $i=1,...,n$ and the parameters $g(x, y)$:

$$\begin{pmatrix} g\big(\|(x_j,y_j)-(x_i,y_i)\|\big) & ... & 1 \\ ... & ... & ... \\ 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} \lambda_i(x,y) \\ ... \\ n \end{pmatrix} = \begin{pmatrix} g\big(\|(x,y)-(x_i,y_i)\|\big) \\ ... \\ 1 \end{pmatrix}$$

The matrix is independent of the interpolation point (x,y). Therefore, the interpolation modul performs an LU factorization. The solution at each interpolation point (x,y) is determined by back substitution. The variance for each interpolation point is

$$v(x,y) = \lambda(x,y) + \sum_{i=1}^{n} \lambda_i(x,y) * g\big(\|(x,y) - (x_i, y_i)\|\big)$$

and must be equal to 0. If this is not the case, numerical problems or truncation errors appear in the interpolation. The program gives a warning, and indicates the number of points with v < 0.

6.2.1.2.3   Input Parameters for Kriging

When you call the Kriging interpolation the file selection window appears first. There you have to select the file with the variogram data if not already an interpolation file has been selected before. This file has to be in the Format of the structure data file (ASCII) . Then the following dialog box appears:
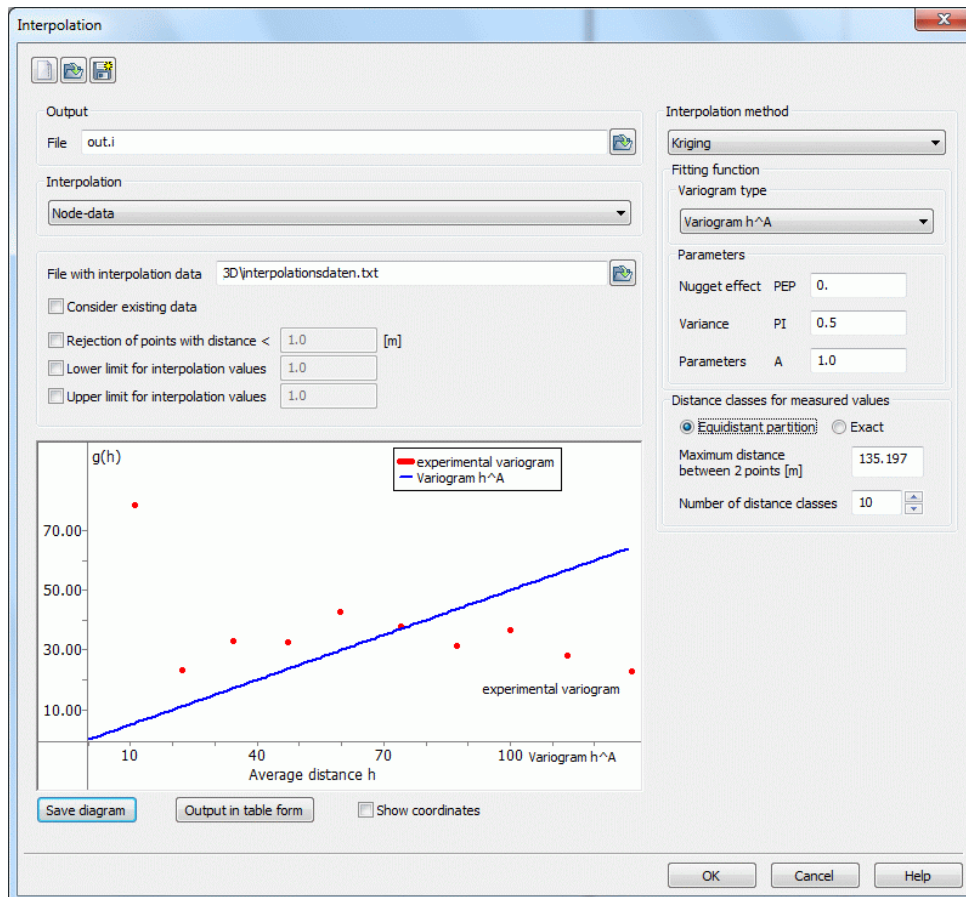
Fig. 178: Input for Kriging

Variogram type

The user can choose from the 6 possible variograms, which are described on p. 248.

Parameters

Depending on the chosen type, different parameters are required:

In case of types from (1) to (5), the user must define the parameters PEP, PI and A.

Type (6) requires the degree of the fitting polynomial form and the number of points in the experimental variogram where the polynomial is approximated. The parameters are described in detail in chapter 6.2.1.3.1.

Distance classes

To create the experimental variogram, it is necessary to define the number of distance classes and the partition of the classes. The button *equidistant partition* causes a uniform partition. In the resulting dialog box, the number of classes and the maximum distance can be chosen. Using the button *exact* the upper and lower boundary of each class can be determined individually (in another dialog box).

Choosing the individual input after the uniform distances makes it possible to edit the values from the uniform partition. If the user changes the settings from uniform to individual partition, the distance from the last class represents the maximum uniform distance. The number of classes corresponds to this value.

Preview window

Here appears the variogram based on the chosen input data. The user is able to define a fitting shape function interactive.

By using the button „Save diagram" the variogram is saved as an image file (*.bmp, *.jpg, *.png). By activating „Output in table form" the data is saved in a table. By activating the check box „Show coordinates" and moving the mouse inside the diagram the coordinates of the interpolation data are shown.

Remark:

The Kriging algorithm is not suitable for large numbers of measured values. Even for the variogram, the computing time and required memory space increase with the square of the number of measured values. The real interpolation computing time for an LU factorization of the weighting matrix is proportional to n³. Each variogram requires many calculations for each pair of measuring points. Therefore, interactive parameter determination and variogram computing require much time without really carrying out the Kriging interpolation.

6.2.2 Interpolation of a Groundwater Contour Map

If a groundwater contour map is to be interpolated, a simple interpolation of the measurement values, as a rule, leads to unsatisfactory results. In spite of this, the model files often contain data which can be taken into account during the interpolation. These include:

- fixed potentials (POTE),
- source/sink terms (KNOT),
- receiving water course (VORF and idealised leakage data LERA resp. LEKN),
- areas of equal unknown potential heads (GLEI),
- minimum values for distance between ground and groundwater-level,
- Q=0 boundary conditions and
- transmissivities (TRAN)

After activating the check box "additional interpolation points" more check boxes appear where the user can define which additional data should be considered during the interpolation.
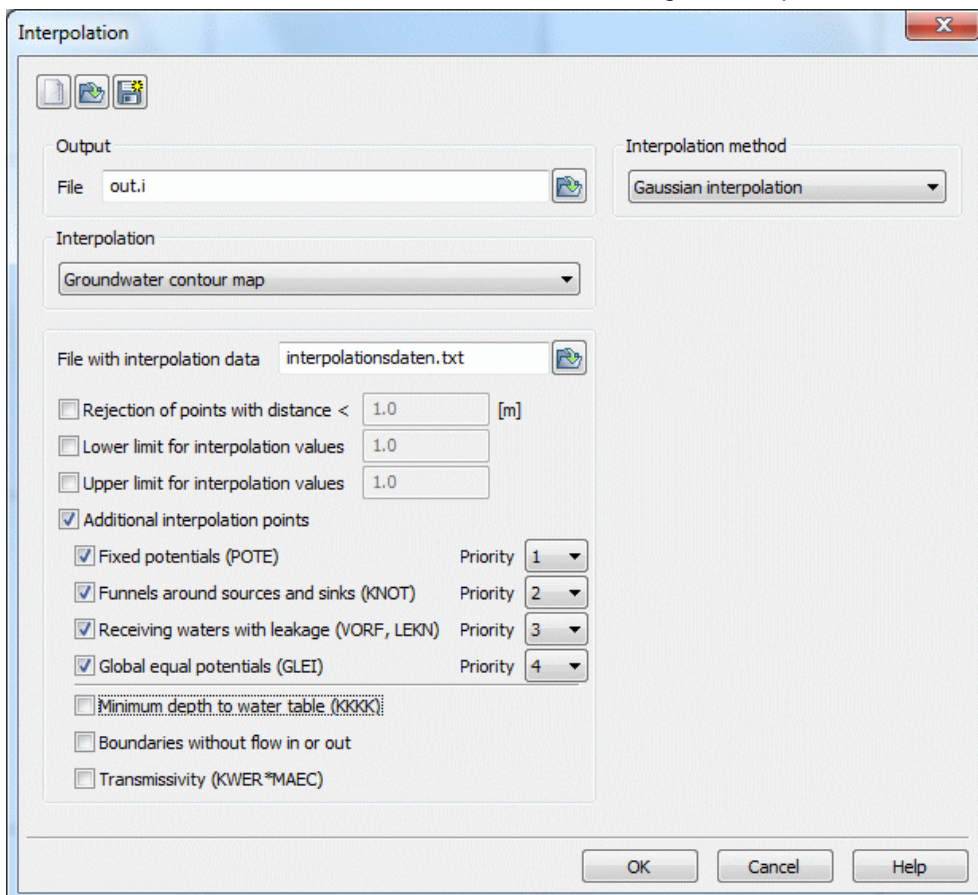
Fig. 179: Additional data for the interpolation of a groundwater contour map

After each interpolation some potential heads are generated for a new interpolation job. They are used as additional auxiliary points in the local sphere of influence of the listed data. The results of the first interpolation with measured data (file including data for interpolation) are used for the calculation of these auxiliary points

As „Additional interpolation points" appear only data which is assigned as attribute in the model file.

Example: If instead of the attribute „LEKN" only the attribute „LERA" is assigned in the model file, the check box „receiving waters with leakage" does not appear. („LEKN" has to be assigned explicitly!)

(1) fixed potentials: All nodes assigned to POTE are used as additional auxiliary points (equal to measured data) for a new interpolation.

(2) funnels around sources and sinks: For nodes assigned to KNOT and inside the estimated range additional auxiliary points will be created with the help of the initial k-values. Then these nodes are assigned to the calculated reduced or increased potential head for another interpolation.

For this k-values (KWER) and element thicknesses of the aquifer have to be available global after a previously performed model checking.

The thicknesses need not necessarily be defined on the data type MAEC, but can also result from the
Calculation of the initial and maximum thickness.

(3) Receiving waters with leakage:

The qualitative influence of an aquifer on the groundwater can be simulated through the definition of a substitute leakage coefficient for all nodes with defined water course level (VORF). For this purpose you must assign the data LEKN to all nodes with defined water course level. Dependent on their connection to the groundwater they get values between 0 (without connection) and 1 (total contact). For these nodes now additional auxiliary points are generated. The potential head is calculated linear between the water course level and the interpolated potential.

Attention: The above described substitute leakage coefficient overrides the for the groundwater simulation needed and defined leakage. It is only useful for the interpolation of water table contours with additional information. For all other calculations you must reinstall the original leakage coefficients!

Therefore, the original leakage values have to be restored or deleted from the model file using an editor or via the menu: *Attributes → Delete…*

(4) Regions with equal potential heads: Taking into account regions of unknown equal potential heads GLEI is equivalent to the interpolation of nodewise data. But in contrast to the interpolation of nodewise data here the spatial effect is taken into account.

(5) Minimum distance between ground level and groundwater level: With the help of the data KKKK (general node data) you can control keeping the nodewise potential head at a minimum distance between ground level and ground water level. For this you have to assign a specific minimum distance greater than 0 to the affected nodes. These values are checked before and after each interpolation step.

(6) Boundaries without in- and out flow: For taking into account this boundary condition you can assign the data KNOT = 0.0 to the affected parts of the boundaries. The element centres of the so created boundary poly-lines are reflected and you get constraint points at both sides of the boundaries. The potential head at these points is calculated with the help of another interpolation with max. 5 iteration steps.

(7) Transmissivities: The influence of the permeability (KWER) and the initial thickness (MAEC) can be taken into account while determining the element transmissivities between two points of measurement. Additional auxiliary points are only created in aeras where no other auxiliary points exist and then only when the transmissivities between two measuring points differ more than 10%.

Priorities:

When taking into account all values of influence (1) - (4) it is possible, that different influences create auxiliary points at the same FE-node. Since this causes local contradictions you have to specify clear and complete priorities for these special cases. Number 1 means greatest priority.

The given priority results in the order of selection of the parameters of influence. She can be changed later. Giving priorities to the other parameters of influence is not required, because the there created auxiliary points

base on the other values. The parameters of influence (5) - (7) are dealt with in the order of their window position.

The interpolation with additional auxiliary points creates the data file <stuetzTMP.txt> (standard text format). The file includes all created auxiliary points with their potential heads, additional to the file with measuring data.

6.2.2.1   Realization in SPRING

Beside the measured values at groundwater observation wells it is essential to take boundary conditions into account when interpolating a groundwater contour map. There is to say fixed potential heads, boundary inflow/outflow at nodes, receiving water courses, leakage or equal unknown potential heads. Especially for the automatic calibration it is important to consider the hydrogeological circumstances in the investigation area

The actual net file should be copied for this interpolation, because some of the used attributes make sense only in this algorithm.

On our homepage http://spring.delta-h.de is a zip archive available for download under Download & Support at Tutorial (EN) (🖳) containing sample files. The directory Tut_Intpol_GwGl contains a mesh file and a file with measured groundwater levels, which should be used in the interpolation of a groundwater contour map.

The following processing steps are carried out in order to create an interpolated groundwater contour map.

The interpolation is performed after running the model checking when choosing the interpolation type *groundwater contour map* and activating the button "additional interpolation points".

A receiving water course is considered by assigning the attribute LEKN (value between 0.0 and 1.0) to the corresponding nodes.

LEKN = 0.0 means, there is no groundwater contact.

LEKN = 1.0 means, there is full groundwater contact.

Existing leakage coefficients (LERA or LEKN) should be deleted, otherwise this receiving water course is not taken into account in the interpolation.

In the example the nodes 204 - 215 of the receiving water course get the value LEKN= 0.9 via *Attributes → Assign →Direct*. The nodes of the headwater which is supposed to have no groundwater contact get the value LEKN = 0.0.
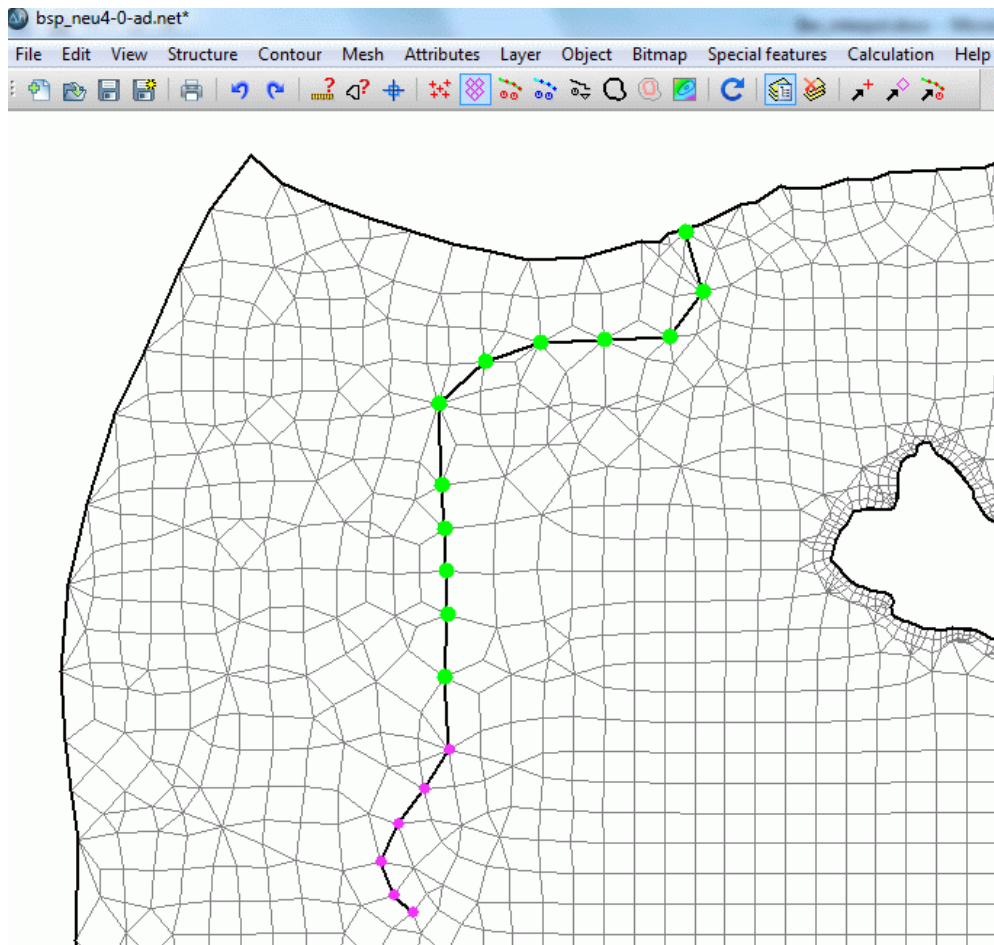
Fig. 180: Assigned LEKN attributes (green = 0.9, pink = 0.0)

If the boundary nodes without inflow should be considered in the interpolation the attribute KNOT = 0.0 has to be assigned to the corresponding nodes.

The attribute KNOT = 0.0 is assigned to the western and southern boundary nodes.

After saving the project and running the model checking you can now activate the check boxes "receiving waters with leakage" and "boundaries without flow in or out" when you have activated the check box "additional interpolation points" before.

The numbers 1 to 4 behind the chosen attributes define the priority of how these attributes are considered in the interpolation:
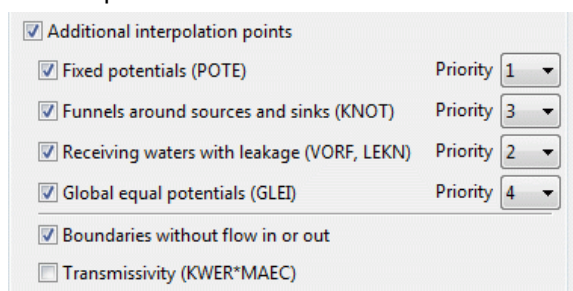
After selecting the "File with interpolation data" ("gwmesspkte_int_gwgl.txt") the calculation can be started.

The following figure shows the groundwater contour map of an interpolation with (red) and without (blue, only the measured data in the file "gwmesspkte.txt" are considered) considering the additional interpolation points.
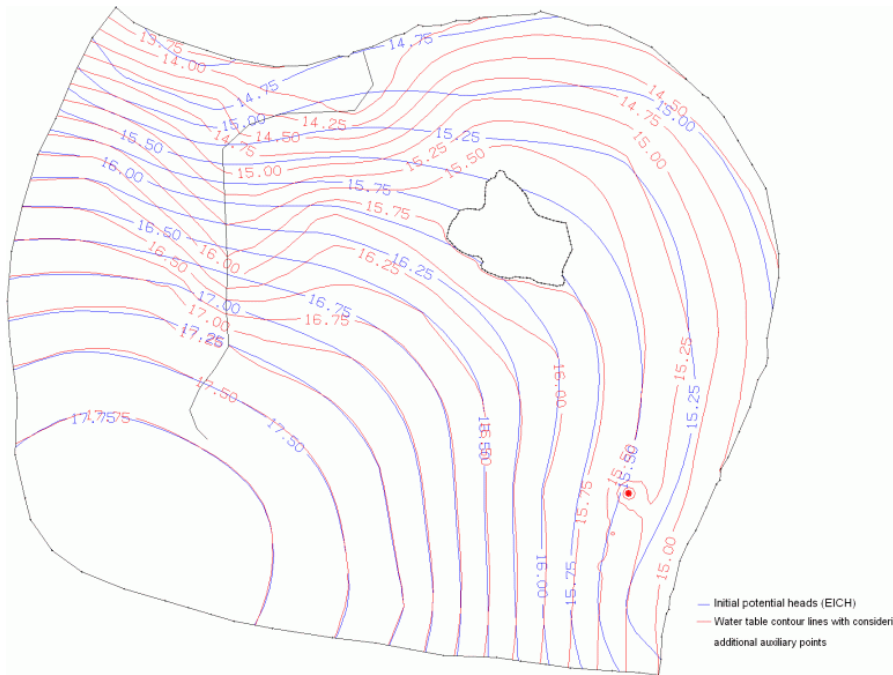
Fig. 181: Comparison of the contour lines after interpolation with (red) and without (blue) considering additional interpolation points

Especially in the area around the lake you can see that considering the equal unknown potential heads (GLEI) prevent the water table contour lines from intersecting the lake surface. The following figure shows in detail how taking the receiving water course level into account has effects on the course of the equal value lines: now the contact between the receiving water course and the groundwater level is visible!
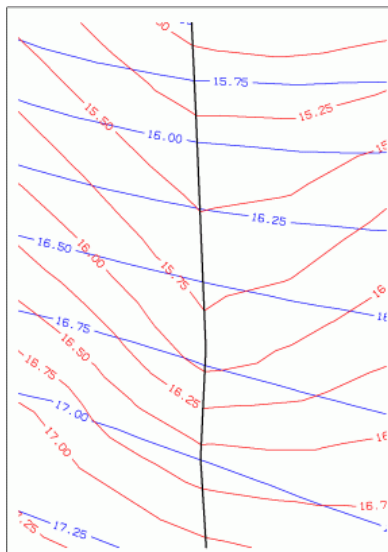
Fig. 182: Visible groundwater contact of the receiving water course by considering the leakage interaction in the interpolation

The calculated potential heads are assigned to the attribute EICH (initial potential heads) with *Attributes → Import data/computation results...*

Attention: For the following flow computation it is recommended to delete the attribute KNOT = 0.0 and to reassign the leakage attributes (LERA). Values between 0 and 1 make only sense for the interpolation and not for the flow computation!

The calculated additional auxiliary points are saved in the file "stuetzTMP.txt". If you join this file with the file including the measured data (here: "gwmesspkte.txt") you get a file with all required interpolation points.

Beside the effects of the parameters shown above sometimes the consideration of other parameters is advisable:

With the help of the attribute KKKK it is possible to keep a minimum depth to the water table. The prescribed distance between groundwater table and the ground level is assigned to the attribute KKKK (in meters). Is the

calculated depth to the groundwater table lower than the required value of KKKK, the calculated depth is adjusted to this value.

To consider the influence of the permeability (KWER) and the thickness (MAEC) of the water saturated zone a transmissivity-weighted interpolation between the groundwater observation wells is required. Necessary for this procedure is a global definition of the attributes KWER and MAEC. The transmissivity of the elements, intersected by an assumed line between two adjoining observation wells, is calculated and their hydraulic gradient is weighted.
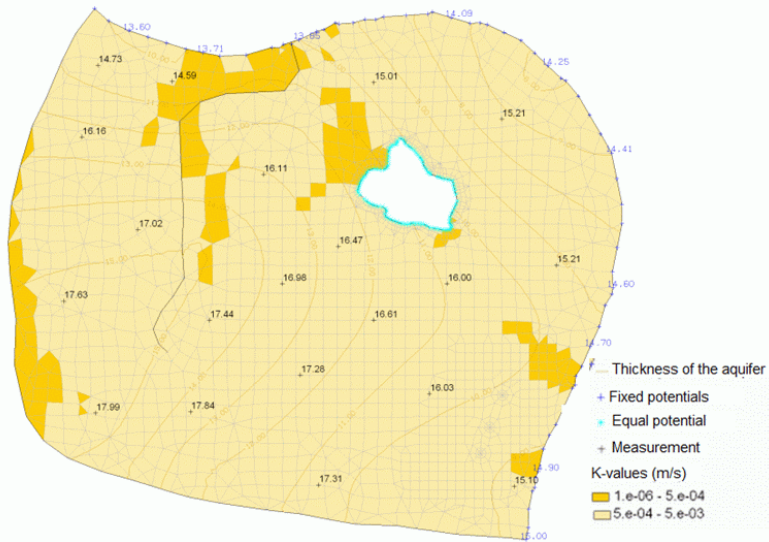
Fig. 183: Figuring the initial model data
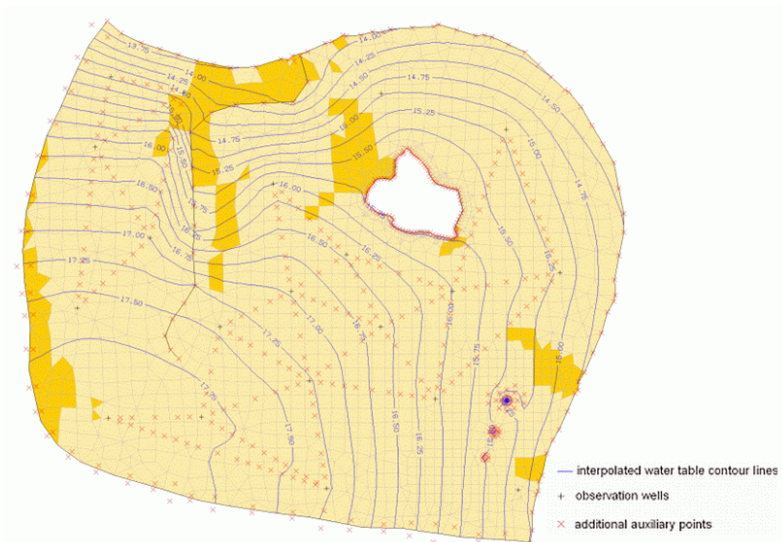
Fig. 184: Groundwater contour map after consideration of different hydrogeological parameters

A transmissivity-weighted interpolation takes place only when there are more than three FE elements between two adjoining observation wells or when the transmissivity varies more than 10 %.

Considering the transmissivity in the interpolation algorithm requires detailed knowledge about the permeability and water saturated thickness of the system before the calibration. This functionality should only be used under the following circumstances:

- only a few groundwater observation wells
- a coarse mesh discretization

6.2.3 Interpolation to a Node Track under considering the Floodplains

This algorithm can, for example, interpolate or rather extrapolate measured water levels in a receiving water course to all nodes of the mesh representing the stream. If the measured values involve pathlines, then the interpolation can work with several time steps. (A one-time interpolation of data on a continuous line can possibly be made easier in the SPRING-surface.) This is useful for time-dependent measured values. For instance, it is possible to create a transient input file which models a high-water wave (flood crest) in a river.
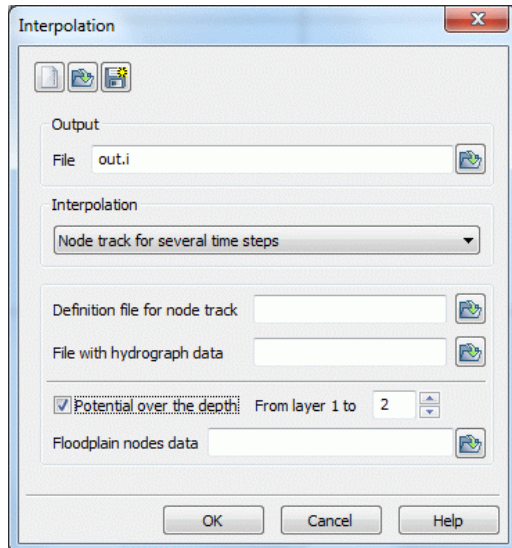
The procedure requires two additional files:

■ The first one contains a list of the nodes along the polygonal line, e. g. the receiving water course (format of the model file).

■ The second file contains the measured values (single values or time-dependent) in the format of nominal time-dependent lines. The first and the last measuring point can be outside of the modelling area. In that case, the user defines the distance.

Attention: The interpolation considers only the time steps of the path line in the data file. All data of other time steps are not considered for the interpolation! Furthermore, note that if the value of a point on a path line is not defined at one of these time points, then this value is never completed with the help of another time-step!

To avoid misunderstandings, please keep the following points in mind:

■ The time steps, on which time-dependent data are defined, should be the same for all nodes lying on the nodal line.

■ If there are missing values in some time-dependent lines, at some time steps you should insert convenient values before the interpolation!

Potential over the depth

Furthermore, the interpolated transient boundary conditions can be assigned to the topmost layer of a 3D model as well as to the layers below.

This is done by activating the check box *Potential over the depth* and selection of the deepest layer number.

Floodplain node data

As part of the interpolation a transient input file with "floodplain nodes" is additionally created. The file contains a node number per line. Their boundary condition is determined by a projection to the nodal line. If the computed potential head is lower than the ground level (GELA), then the program automatically switches off the boundary condition of the corresponding node (output with a negative node number).

The consideration of the foreland is often required for large receiving water courses in order to define the impact of a flood wave on the foreland, eg to define flood areas.

The procedure is described in detail in the following chapter.

6.2.3.1   Realization in SPRING

Creating the file with the node track

The node track is defined in or against the flow direction of the receiving water course (note the order of the nodes!):

```
    1,     2,     3,     5,     6,     4,     8,+
```

Since the format of the file corresponds to the model file, the proceeding to generate this file is like this:

In a temporary copy of the model file the attribute LERA is deleted. Instead, the attribute LERA is assigned only to the linear structure of the water course which shall be interpolated.

LERA should contain all nodes of the water course (control by representing the attribute as circles).

After saving the model file and opening it with an editor the LERA data is copied and pasted in a new ASCII file. If necessary, "from-to"-signs have to be replaced by the missing node numbers. The finished file should only contain the sorted node numbers.

Creating the file containing the hydrograph data

The following figure is an example how to create hydrograph file. At two points of the receiving water course are gauge values. At the measuring point P2 a measured water level for one time step is missing.
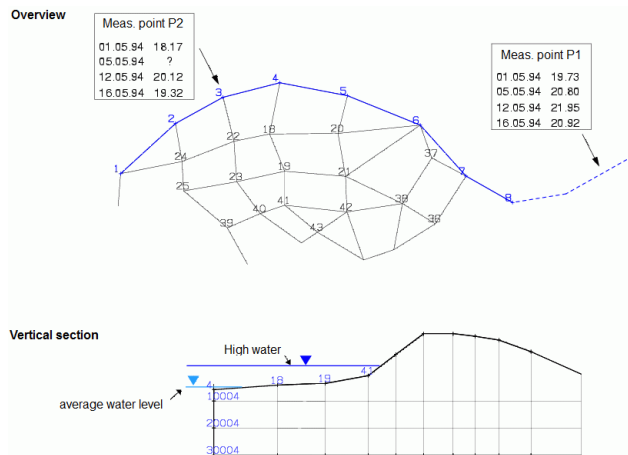
Fig. 185: Interpolation to a receiving water course with floodplain

The hydrograph file begins with the date line. After a blank line follow the data blocks of each node, separated again by a blank line:

```
DATUM    01.01.1994

8  987.00
01.05.1994    19.53
05.05.1994    20.80
12.05.1994    21.95
16.05.1994    20.92


3
01.05.1994    18.17
05.05.1994    19.28
12.05.1994    20.12
16.05.1994    19.32
```

The 2nd line defines the measuring data not at node 8, but at a point 987 m away. The missing value (see figure above) for the time step 05.05.1994 at station P2 (node 3) was completed with an approximated value.

This file has to be created manually in the appropriate format.

Creating the file containing the floodplain nodes

First, a linear structure of the main receiving water course (*Structure → New → Polyline*) is created. The particular peak value of the flood wave is assigned to the "gauge node".

Example:

Gauge P1 = Node 1 → Peak value = 15.94 m NN
Gauge P2 = Node 68 → Peak value = 16.20 m NN
Gauge P3 = Node 112 → Peak value = 17.75 m NN

The line structure of the main receiving water course is assigned to any empty type of node data (e.g. KKKK) with an accuracy of about 1000 m. Then the calculation of the data types GELA - KKKK = KKKK is done and the values of KKKK limited to max. = 0.0 (*Attributes → Computation → Intercalculation…* and after that *Attributes → Computation → Limit…*).

Next all values KKKK = 0.0 are deleted with *Attributes → Delete…only values = 0.* The attribute KKKK is displayed as circles and then the attribute KKKK is deleted manually at the nodes of the water courses, at nodes behind a dam or at nodes which are too far away from the flood plain.

After a model check the attribute KKKK is exported via *File → Export → Formatted Data Output…*in the output format *No., value* (see Data Export).

The new file shall be reduced to the column with the node numbers with the help of MS Excel or something like that. Then, the file with the.floodplain node data is ready for use.

When all necessary files are provided the interpolation is performed after the model checking.

The following dialog box appears:
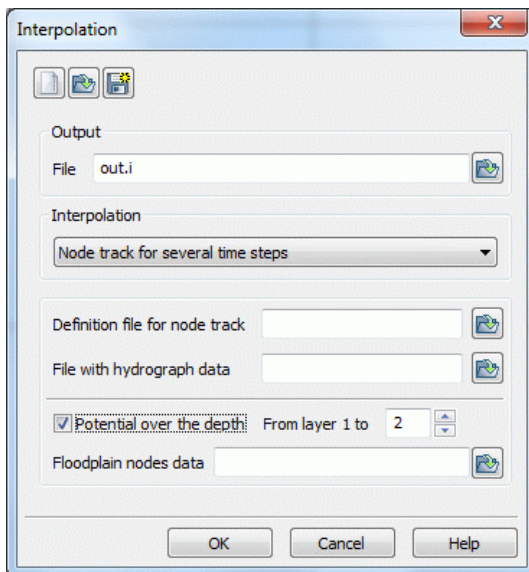
Fig. 186: Interpolation to a node track under considering the floodplain

Here all names of the created files are entered.

The output file "out.i "of the interpolation contains the data of the transient potential heads of the main receiving water course and the floodplain nodes (marked with "-"). Potential heads of lower layers are also in this output file, if they were considered.

The output file is converted to the format of a transient input file with the following headers:

Example of a transient input file, converted from "out.i"

```
ZEITEINHEIT MENG JAHR
BEZUGSDATUM 01.01.2004
NUM 7


DATUM
02.01.2004


POTENTIALE (Knotenzug) Schicht 1-4
      2 18.084476 10002 18.084476 20002 18.084476 30002 18.084476      3 18.170000
 10003 18.170000 20003 18.170000 30003 18.170000      4 18.262771 10004 18.262771
 20004 18.262771 ...


POTENTIALE (Vorland)
   -18 18.227260   -19 18.296584   -20 18.368244   -21 18.416917   ...
   -41 18.306354


DATUM
...
```

The marking **NUM 7** indicates that the node numbers in the file out.i are written in the format I7 (normally the transient input file has the format I6).

6.2.4 Input Parameters of the SPRING Interpolation

The following window will appear by navigating to the menu item *Attribute → Assign → By interpolation:*
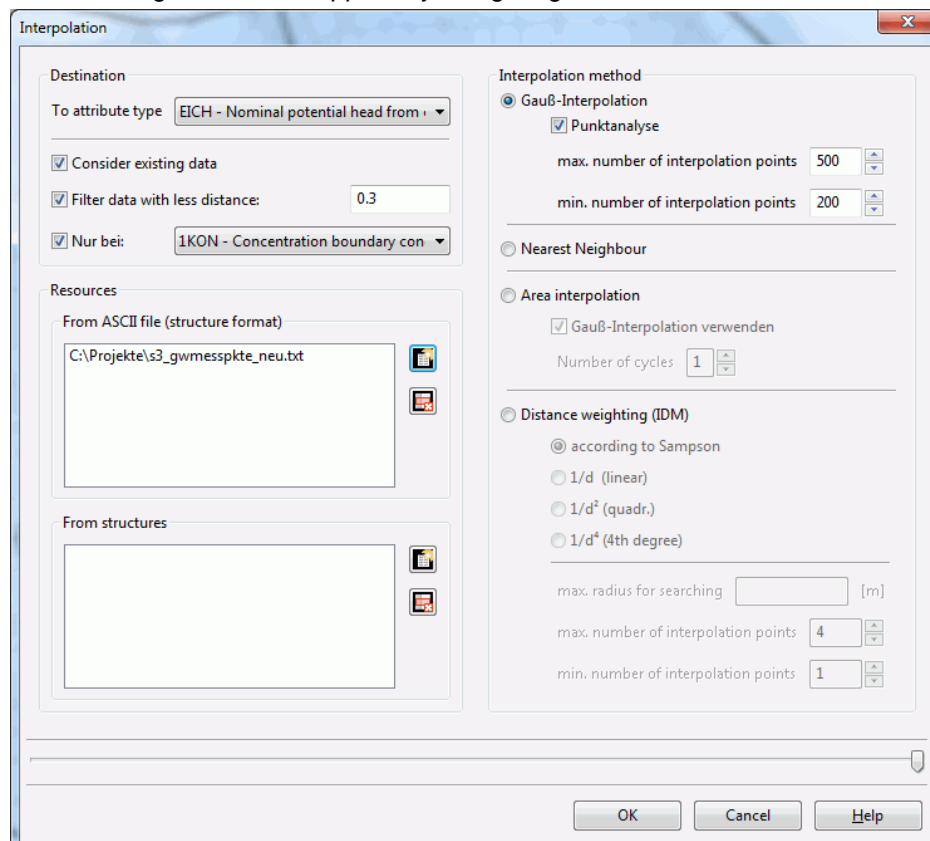
In the interpolation window, various configurations, including the interpolation method and their specific settings, the target attribute, input data from either an external file or a structure, and additional input parameters may be set. A detailed description is found in chapter: "Compilation of a 2D model" (p. 201).

6.2.4.1 Gaussian Interpolation

This interpolation algorithm requires no additional input parameters.

In the first step of a Gauss interpolation, an interpolation function $z(x, y)$ is fitted to the measured values. This interpolation function is a linear combination of Gaussian bell-shaped curves for the measuring points. The values of the function at the interpolation point obtain the interpolated value z at this point.

There are n measuring points (x_i, y_i), $i=1,...n$, with the corresponding measured values z_i.

The interpolation function of a Gauss interpolation procedure is always based on the Gaussian normal distribution functions (fundamental form g_i) for the measured values (x_i, y_i). They have the form

$$g_i(x, y) = \frac{1}{1 + \dfrac{\sqrt{(x - x_i)^2 + (y - y_i)^2}}{m^2}}, i = 1,...,n$$

where m: average distance between all measuring points.

The fundamental functions g_i represent surfaces of rotation whose axis of rotation is defined in the corresponding measuring point (x_i, y_i). They are linked linearly to a surface function:

$z(x, y) = b_1 g_1(x, y) + b_2 g_2(x, y) + ... + b_n g_n(x, y)$

The coefficients b_j for the linear combination are unknown and represent the parameter of a family of the surface function. The measured values must lie on the defined surface. This leads to an equation for each measuring point i. With the help of this condition, the parameter of a family can be determined:

$z_i = z(x_i, y_i) = b_1 g_1(x_i, y_i) + b_2 g_2(x_i, y_i) + ... + b_n g_n(x_i, y_i)$

Hence follows a symmetric $(n \times n)$ equation system for the n unknowns b_i, $i=1,...n$.

The resulting values b_i complete the surface function $z(x, y)$. Substitution of the coordinates x and y returns the corresponding value at every node or rather element centre.

To prevent oscillations of the Gaussian surface function $z(x,y)$, this is then scaled with an additional surface function.

$Z = Z(x, y) = B_1 g_1(x, y) + B_2 g_2(x, y) + ... + B_n g_n(x, y)$

The coefficients B_i, $i = 1, ... ,n$ are determined using

$Z_i = Z(x_i, y_i) = B_1 g_1(x_i, y_i) + B_2 g_2(x_i, y_i) + ... + B_n g_n(x_i, y_i) = 1$ for $i=1,..,n$.

Finally, the interpolation value I at a given point (x, y) is defined as

$I(x, y) = z(x, y) / Z(x, y)$.

Implementation in SPRING

The new implementation takes advantage of data structures and runs in parallel, making it quick and efficient in terms of resource management. Nonetheless, there are new parameters that should be chosen carefully to obtain the most appropriate results and performance. This section provides general guidelines on how to configure these parameters. The following figure shows the Gauss interpolation input window.
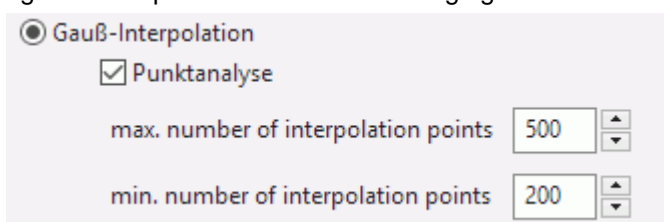
Fig. 11: Gauß parameter

The most important parameters in this method are the minimum and maximum number of points. In general, the larger the minimum number of points, the smoother and more accurate the final results; however, it should be noted that using an unnecessarily large minimum number may increase calculation time without drastically affecting the results. On the other hand, the maximum number acts as a constraint and controls the behavior of the solver. Moreover, it is observed that a close minimum and maximum combination results in a slight decrease in performance. A rule of thumb would be to use a maximum number that is roughly 2-2.5x the minimum number. Furthermore, the point analysis checkbox invokes a two-pass analysis module that controls several criteria to ensure the best possible output is obtained. This option is found to enhance the final output in most cases; in addition, the increase in computational demand for using this option is negligible compared to the main calculations. Hence, it is recommended to keep this option enabled in general cases.

It should be clear that the mathematical model for the Gauss interpolation requires at least two data points.

Different input parameters will of course leas to different results, and especial care should be taken to use suitable input parameters; nonetheless, the algorithm has been implemented in a way to reduce any adverse effects stemmed from inappropriate input parameters as far as possible. Furthermore, this interpolation fully runs in parallel.

6.2.4.2 Area Interpolation

Area interpolation considers the closest measuring points to the interpolation point on a grid-based surface. In case enough measuring points are not found for an interpolation point, no value is assigned to that point. The following figure shows the area interpolation input window:
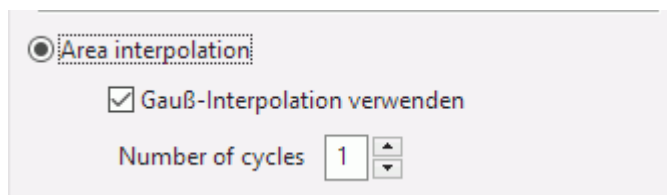
Fig. 12: Area interpolation

Area interpolation may be configured to perform multiple cycles or passes. This is especially useful in case enough measuring points are not available or their distribution is not uniform. In each cycle, interpolated points from the previous cycle(s) are used to help interpolate points without value in an efficient way. Furthermore, the Gauss interpolation check box configures the program to integrate a Gaussian interpolation into the area interpolation, which is recommended in case measuring points are insufficient beyond what can be refined by increasing the number of cycles. This option ensures that all points are interpolated. In case this option is selected, one cycle of area interpolation would normally be enough. This interpolation fully runs in parallel. In contrast to the case of distance weighting, the area interpolation can produce interpolation values larger than the maximum and less than the minimum measured value. This is caused by the extrapolation process, and occurs when an interpolation point lies outside of the defined surface.

6.2.5 Choosing the correct interpolation algorithm

The following table facilitates selection of the most appropriate interpolation algorithm. The recommendations mainly depend on the necessary computing processes and the resulting computing time.

	Number of Data Points			Remark
	few (<5000)	medium	many (>5000)	
Weighting Interpolation	0	(x)	x	tends to form "steps"
Area Interpolation	(x)	x	x	very suitable for digitalized values
Kriging	x	(x)	-	smooth values with the right parameters, determination of the parameters requires considerable time
Gaussian Interpolation	x	x	x	Smooth values and contours. Accurate and efficient. Recommended for most cases,
Nearest Neighbour	0	(x)	x	

Meaning: - : not suitable / 0: less suitable / (x): suitable / x: very suitable

The current implementation of the Gauss interpolation is suitable for all ranges of data points from just a handful to extended clouds over the whole model; hence, it should be generally preferred as it yields the most accurate results and leads to very smooth contours.

In especial cases, when the data cloud is exceptionally large and excessive measuring data is available, the raw initial data is of little use to the Gauss interpolation. A good option in such cases is to use area interpolation and complement it with an integrated Gauss interpolation using the option available in the interpolation dialog window. For more information, please refer to area interpolation.

6.2.6 Batch File Interpolation

It is possible to execute the *interpolation* directly in the command line without using the dialog box, if a batch-file *. bip* exists in the directory. For this you have to change to the directory in which the calculation is to be executed. The input of *interpol* and confirming with the Enter key starts the interpolation. The default file name is *interpol.bip*. This file is searched in the directory.

To use another batch file, type *interpol file name* (the extension ".bip" is automatically attached). Batch files can be created, viewed and modified with the help of any text editor.

A detailed batch file can be viewed in the online help. As well, there is a batch file provided for further processing.

6.3 Model calibration (gradient method)

Call via: *SPRING → Calculation → Model calibration (gradient method)…*

When assembling the input for a groundwater model, the different types of data (i.e. the permeability or the potential heads) are usually determined independently of each other. The parameters and boundary conditions are either known or estimated. In running the model, these different datasets are linked together through the mass balance which must always be satisfied. In most cases, the calculated groundwater levels will then differ from the measured ones, even after any obvious errors (wrong exfiltration rates or boundary conditions) are eliminated. The remaining error is generally due to data uncertainty, and usually the permeability K is the most uncertain parameter.

If the user provides an initial permeability, the permeability of the elements can be iteratively calibrated with the help of the model calibration with the gradient method. The iteration is done until the differences between calculated and measured potential heads reach a minimum. The following model types can be calibrated:

- 2D horizontal models
- 2D/3D models
- 3D models

The calibration is done for an assumed steady state. Hence, it is necessary to use initial conditions that are as close to a steady state as possible. The gradients of this initial state must comply to the overall flow direction. Otherwise, the resulting permeabilities might lack the scientific background. Please be careful with sources, sinks and water sheds.

The desired condition is defined by the dataset of initial potential heads (Attribute EICH) at each node. These can be interpolated from field measurements. Alternatively, the necessary values can be gained from scanning equipotential lines from hand-made maps. The results of an interpolation should be considered as such rather than as "real" field data, as small errors within the input data sets may lead to large errors in the resulting calibrated permeabilities. Please consider the suggestions in the preceding paragraph!

The chapter: "Interpolation of groundwater contour maps" describes how all the data which affects the groundwater is taken into account.
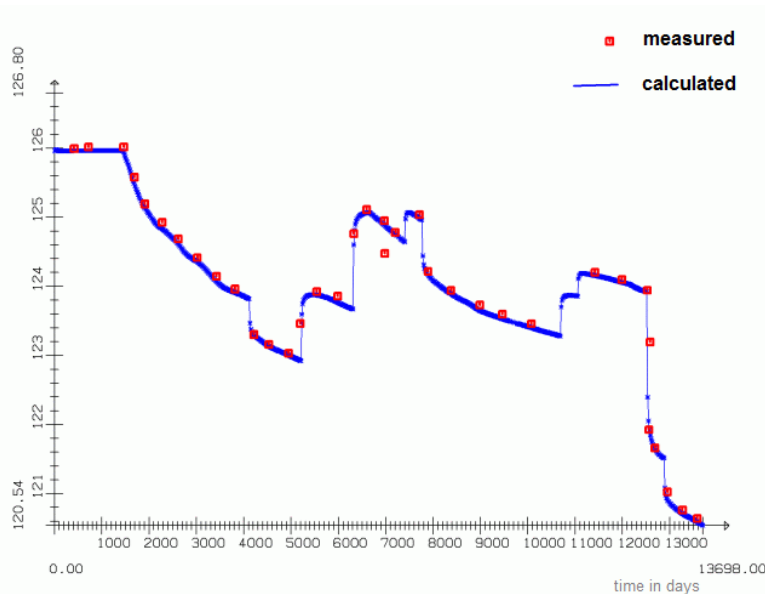
Fig. 187: Example of a transient calibrated hydrograph

In a following chapter the calibration with the help of the inverse modelling is described.

6.3.1 Iterative calibration of the horizontal K-values

For changing the horizontal permeability K, the expected and current gradients are considered. The new value is then computed by:

$$k_{n+1} = (I / I_s) * k_n$$

with:

k = permeability of the elements

I = gradient of the model calculation

I_s = gradient of the nominal state

n = iteration step

- The permeability K is only changed for elements where the angle between the expected and the current gradient is less than or equal to 60 degrees.
- Elements with an anisotropic permeability K (using ZONE and MATE) are not subject of the calibration!

■   The permeability K assigned by the calibration can be limited to a desired range of values (KMIN / KMAX).

6.3.2 Local attenuation of the relative change of the K-values

After changing the relative K-value in an element follows automatically a local attenuation by considering the K-values of the adjacent elements.

This method avoids large oscillations in adjacent elements during and after the calibration. For each element, the relative change (in %) is compared with the relative changes for the adjacent elements. Depending on the differences, the value is fitted. The procedure is illustrated in a two-dimensional example

Element	K-value of the last iteration step	New calculated K-value	Relative change in %
1	0,40	0,01	2,5 %
2	0,30	0,45	150 %
3	0,35	0,70	200 %
4	0,30	0,15	50 %
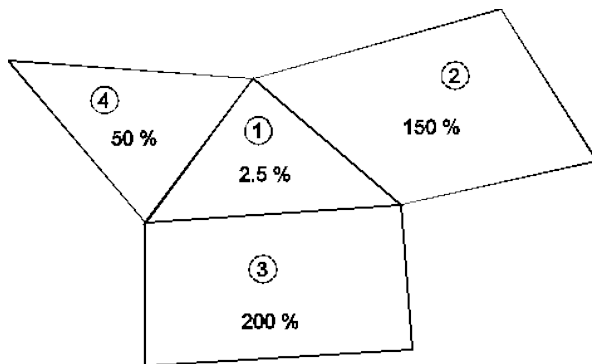
The resulting permeability for element 1 is the arithmetic mean of:

■   the relative change for the hydraulic conductivity in element 1 [%] and
■   the average of the changes [%] for the adjacent elements

Sample calculation:

½ * (2,5 % + (150 % + 200 % + 50 %)/3) = ½ * (2,5 % + 133,33 %) = 67,92 %

→ The permeability K for element 1 is decreased to 67.92 % of the initially computed value. It is now decreased from 0.40 to 0.271.

In the case of a three-dimensional model, the adjacent elements in the vertical direction are also taken into account. Layers with permeabilities of different magnitudes (aquifers and aquitards) are automatically detected and taken into proper consideration. Adjacent elements which belong to hydraulically different layers are weighted with smaller factors.

6.3.3 Input parameters

Via *Calculation → Model calibration (gradient method)…* the following input window appears:
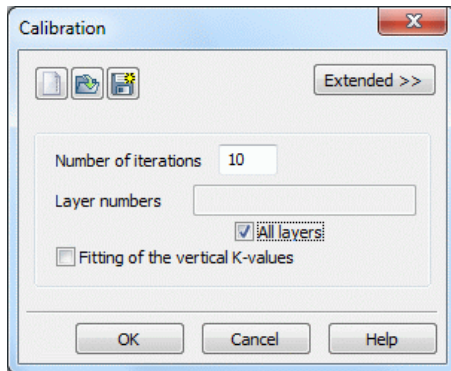
Fig. 188: Input for the calibration (3D)

When opening the dialog box, the program reads the batch file with the default name *eichen.bei*, if it exists. Then the default settings in the dialog box may be changed accordingly.

Number of iterations

The number of iteration determines how often the iterative change of the K values takes place.

Layer numbers (3D model only)

For 3D models, it is possible to calibrate only the elements of the selected element layers (e.g., only the top layer). The dialog field *Layer numbers* allows the choice of layers to be considered for the calibration: either all layers - or just individual layers. This allows i.e. leaving the K-values of an aquitard untouched.

Fitting of the vertical K-values (3D model only)

The change of the vertical permeability can be controlled as well in a 3D model. The values for the vertical permeability can either remain constant or they can be altered in a way that the relation between the horizontal and the vertical permeability remains constant. Only the horizontal permeability is calibrated directly by the ratio of the gradients.

The buttons at the top of the input window allow resetting the input parameters ( ), opening an existing batch file ( ) or saving the current batch file under another name ( ).

The buttons at the bottom of the input window start the calculation (*OK button*), close the dialog box (*Cancel*) or open the digital manual (*help button*).

6.3.4 Extended

By activating the button "Extended" more parameters can be defined:
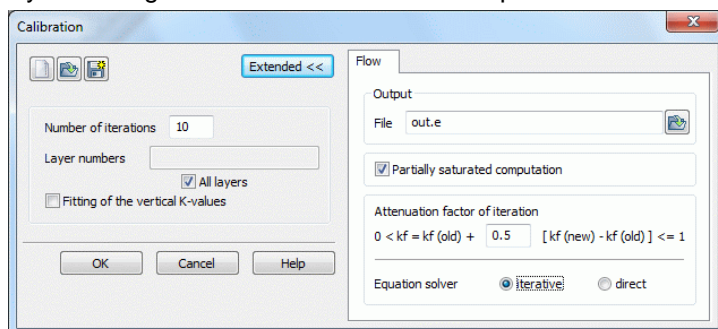
Output

The name of the output file is entered, default is the name *out.e*.

Fitting of the saturated thickness (2D) or partially saturated computation (3D)

For three-dimensional models, consideration of the saturated and unsaturated zones is possible (see also iteration of the free surface). The factors for the unsaturated elements are also computed using the previously calibrated data set. For two-dimensional models, the thickness can also be fitted, using the calibrated condition and the lower boundary of the aquifer, while respecting an eventually given maximum thickness. Fitting the thickness is only useful if the input thicknesses recorded during the model checking is different from the thickness of the saturated zone. Generally, this should not occur.

Attenuating the relative change of the K-value in an element (3D model only)

The changes in the K-values of an iteration to the next can be attenuated in various ways. This is recommended because the iteration oscillates slightly at a 'bad' initial state and K-values of adjacent elements can vary extremely.

A local attenuation of the relative change of the K-values considering the K-values of the adjacent elements is done automatically and was described above. For the simple attenuation of the changes of the K-value from an iteration to the next, the user can determine at this point an attenuation factor.

Instead of the computed new K-value k_{n+1} (see iterative calibriation),

$$k_{neu} = k_n + x * (k_{n+1} - k_n) \text{ mit } 0 < x <= 1$$

is applied with a default value of $x = 0.5$.

Equation solver

As solvers, the iterative PCG-solver or the direct Cholesky solvers are available. For meshes with more than 500 nodes, we recommend the iterative solver with respect to computing time and storage space. It does not compute an exact result, but the deviation will usually be negligible. The accuracy can be evaluated using the mass balance

6.3.5 Results of the calibration

The iterated horizontal and vertical K values in the unit [m/s] are assigned via *Attributes →Import data/Computation results ...* directly to the attribute KWER and KWEV.

Besides, the FE computation provides the following results:

- Potential heads at each node,
- Distance between ground level and groundwater level, as long as the surface data has been provided for,
- Mass flow rates (in or outflow rates) for each nodes with fixed potential heads,
- Leakage amounts for nodes and elements. The leakage coefficients and the receiving water course levels (LERA, LEKN, LEEL) and VORF) are essential information within the input data and
- Mass balance for the cumulative amounts for in- and exfiltration. They are compared to the total mass flows. The sum of the cumulative amounts for in- and exfiltration is supposed to be close to or equal to zero. Besides this, the infiltration (positive) and the exfiltration (negative) are listed separately and the areas, which have been balanced in BILK and BILE.

6.3.6 Batch file model calibration

It is possible to execute the *calibration* directly in the command line without using the dialog box, if a batch-file *. bei* exists in the directory. For this you have to change to the directory in which the calculation is to be executed. The input of *eichen* and confirming with the Enter key starts the interpolation. The default file name is *eichen.bei*. This file is searched in the directory.

To use another batch file, type *eichen file name* (the extension ".bei" is automatically attached). Batch files can be created, viewed and modified with the help of any text editor.

A detailed batch file can be viewed in the online help. Similarly, there is a batch file of the model calibration for further processing provided.

6.4 Inverse Model Calibration

Call via: *SPRING → Calculation → Inverse calibration…* A distinction is made according to the method of calculation:

→ *Steady state flow…*

→ *Transient flow…*

→ *Mass transport (density dependent)…*

→ *Heat transport computation…*

After creating the groundwater model and input of all data follows the calibration. SPRING supports the Calibration with the gradient method or the method of the inverse modelling.

The inverse modelling is an iterative process in which the system will be explored step by step and this will be judged over and over again. The user must constantly review the sense of the calculated values. After correcting false measurements or other incompatibilities of the input data remain the following model parameters as uncertain variables:

- Permeabilities (K-values)
- Leakage coefficients (LERA, LEKN)
- Boundary in- and outflow (VORF and LERA)
- Storage coefficients (transient calculation, KSPE)

The unknown model parameters p are normally determined by measured potentials on groundwater wells. For transient calculations hydrographs of potentials should also be present for some measuring points. In addition, e.g. discharge measurements give information about the magnitude of leakage rates, and the volume of boundary in- and outflow may be known. These are measurements or observations data (m).

In the calculation of the direct problem the potential heads and the quantities (measures m) are calculated from estimated, but unsecured model parameters p. In the inverse method, however, the inverse problem is to be solved, in which the unsecured model parameters p are determined from the backed measures m.

Method	Input data	Calculated data
direct	Model parameters p	measured data m
inverse	Measured data m	Model parameters p

The model parameters are in the inverse problem the dependent variables which should be determined by the independent observations. The aim of the calibration is to estimate the unknown model parameters so that a measured state is reproduced by the numerical model in the best way possible.

In the formulation of the inverse problem is to be taken into account that the hydrogeological concept might be faulty. Through a statistical formulation of the inverse problem, errors in the subsequent interpretation of the measured data can be compensated.

6.4.1 Model parameters as unknown sizes

The following input data are the unknown, to be determined model parameters which should be optimized using the inverse modelling:

- Permeabilities:

  In a 3D model the horizontal (KWER) and vertical (KWEV) K-values can be optimized simultaneously. It is possible to optimize the horizontal K-values and simultaneously hold the vertical K-values constant or the other way round.

  In these cases, there is no way to influence the anisotropy factor between horizontal and vertical K-values. However, it is possible to optimize the horizontal K-values and hold the ratio between horizontal and vertical K-values constant by entering KFVH (*Attributes → Special Features → Adjust factor vertical/horizontal K-value (KFVH) …*, default: global KFVH = 0.1).

- Storage coefficients:

The storage coefficient (attribute KSPE) in a horizontal model or the porosity (PORO) for calculating the storage coefficient in 3D or vertical models can be optimized (see explanations to the storage coefficient in chapter: "Transient calculation").

■ Leakage coefficients:

The leakage coefficients for polylines (attribute LERA, per m) can be optimized.

About these parameters p, there is usually prior knowledge, for example in the form of laboratory tests, or interpretations of field tests or subjective preliminary estimations. The starting values for the iteration of the inverse problem are based on this previous knowledge.

They must be defined for an optimization in a separate file. This file is mentioned in the following as "parameter file". The structure of the parameter file is described in chapter: "Realization in SPRING".

6.4.2 Observation data and measured data

The observation data are used to judge the quality of the current parameter set p. Their analysis provides usually the main share of the current error f (p) after a calculation of the direct problem. Currently, the following observation data are possible:

■ Potential heads at single nodes of the model (attribute POTE)

■ Leakage rates (i.e. from discharge measurements) as sum of a defined section of a water course, which is defined by the attribute LERA (attribute LKNO, only inverse modelling),

■ Mass flow rates as sum for a defined number of nodes, which are assigned to the attribute POTE in the model file (attribute KNOT).

All data can be available at different times in transient calculations.

Which observation data are used, is regardless of the choice of model parameters to be optimized. For example, K-values can be optimized by observed potential and leakage rates, without optimizing leakage coefficients simultaneously. In contrast, for the optimization of leakage rates the existence of observed leakage data is not of necessity.

The number of available observation data has a significant influence on the success of the inverse modelling. The more observation data are available, the greater the prospect of finding an "optimum" parameter set. A successful optimization is achieved only if the zoning of the parameters to be optimized is reasonable. To define a reasonable zoning is the main task in the inverse modelling.

The observation data must be defined for an optimization in a separate file. This file is mentioned in the following as "observation file". The structure of the observation file is described in chapter: "Realization in SPRING".

6.4.3 Zoning of the parameters

Because the computational effort growth disproportionately with the number of parameters to be optimized, it is necessary to zone the model parameters to be determined with the inverse modelling. That means, the parameters are summarized in zones of equal values. In SPRING it is possible to zone the K-values, the storage coefficients and the leakage coefficients. For this the attributes described in the following chapters are to be assigned.

6.4.3.1 Zoning the K-values and the storage coefficients

K-values (horizontal and vertical), storage coefficients in the horizontal model and the porosity for the calculation of the storage coefficient in the vertical and 3D model are data that must be defined for all elements. However, due to the computational cost it is not possible to optimize for each element or node its own parameter. Therefore, these data are zoned.

The following model data types are provided to zone these parameters:

■ Parameter Z-KW for zoning the K-values (KWER, KWEV).

■ Parameter Z-SP for zoning the storage coefficients (KSPE).

The zones are established in the model file. The zone numbers must be integers > 0. If some elements or nodes are not defined in any Z-KW and Z-SP zone, an additional zone with the zone number 0 is automatically generated during the model checking, in which all non-covered elements or nodes are listed.

Within a zone exist three possibilities:

- If only a corresponding initial parameter is defined for only one element/node of the zone in the parameter file, all the elements/nodes of that zone obtain this constant parameter value. In this case it does not matter, for which element/node of the zone this parameter was defined.

  → Only one parameter is optimized in this zone.

- If initial parameters are defined for several elements/nodes of a zone, values are interpolated for all elements/nodes of this zone from these parameters (the interpolation algorithm must be selected). Parameters of other zones are not taken into account in this interpolation.

  → There are several parameters optimized within this zone, namely the parameters applied for the interpolation.

- If no initial parameter is defined for an element/node in a zone the corresponding data of the model file are used for the calculation.

  → Within this zone no parameters are optimized. The calculation uses the constant values for the K-values or the storage coefficients of the model file (attributes KWER, KWEV or KSPE).

Should therefore the values of the model file be constant for elements or nodes in a zone, initial parameters for this zone must not be entered in the parameter file.

If one or more initial parameters are defined for certain zones in the parameter file the data for the elements/nodes of the zones entered in the model file is of no significance, because during the optimization they are replaced by the calculated values.

If all the elements/nodes are zoned and initial parameters are defined for all zones the input of K-values and storage coefficients in the model file is not required anymore.

6.4.3.2 Zoning of leakage coefficients

The attribute Z-LR is available for zoning the optimization parameter LERA. It defines a node-polyline for which a polyline leakage coefficient (attribute LERA) should be determined.

Attention: Formally, it is necessary for all polygons for which a leakage coefficient is to be optimized using the inverse modelling, additionally to assign a leakage coefficient in the model files with LERA.

- If an initial parameter for the leakage coefficient of a Z-LR-zone is specified in the parameter file, this parameter is optimized. From the initial parameter a node-based leakage coefficient is derived for the calculation of the direct problem by integrating the path length between the neighbouring polyline nodes. The additionally (and formally) in the model file entered leakage coefficient is in this case without meaning.
- If no initial parameter for the leakage coefficient of a Z-LR zone is entered it is calculated with the leakage coefficient of the model file for this polyline.

6.4.4 Formulation of the inverse problem

The following remarks are intended to provide guidance for the practical application of inverse modelling. They do not give a correct mathematical formulation of the inverse problem. Here we refer to additional literature.

6.4.4.1 Minimization of the target function

The objective of inverse modelling is to minimize the target function. The target function consists of two main components:

$$ZF = ZF_{obs} + ZF_{par}$$

Both the measured values (ZF_{obs}, i.e. potential heads, quantities) and the estimated values of the parameters (ZF_{par}, for example K-values, leakage coefficients, storage coefficients) enter into the target function. The shares are further divided

$ZF_{obs} = ZF_{POTE} + ZF_{LEAK} + ZF_{KNOT}$

$ZF_{par} = ZF_{KWER} + ZF_{LERA} + ZF_{KSPE}$

To determine the target function, a standard deviation σ is assigned to each parameter and measured value. This corresponds to the expected error of the parameter.

Each measurement value m, and each parameter value p is evaluated after each step of the inverse calibration as follows (method of type 'least squares'):

$$\mathbf{ZF_m} = \left(\frac{\mathbf{m_{gem}} - \mathbf{m_{ber}}}{\mathbf{\sigma_m}} \right)^2 \quad \text{or} \quad \mathbf{ZF_p} = \left(\frac{\mathbf{p_{gesch}} - \mathbf{p_{ber}}}{\mathbf{\sigma_p}} \right)^2$$

For the numerical solution of this nonlinear minimization problem, SPRING offers two different mathematical algorithms in the inverse modelling:

- a Gauss-Newton-method by Levenberg-Marquard and
- a gradient method by the Quasi-Newton-method

The shares of the target function of each measurement and of each parameter are summed. Chosing a small standard deviation σ results in a large proportion of the target function and thus in a high weighting. From a large standard deviation σ results a small proportion of the target function and thus a small weighting.

The following table shows the influence of weighting.

1. First, the two potentials H1 and H2 are listed. The residual is the same for both measurement points, however, a hundred-fold greater proportion of the target function results for H2, as this value is weighted with 10 cm standard deviation more strongly as H1 with 1 m standard deviation.

2. Another effect can be illustrated with the quantities Q1 and Q2. Depending on which unit is used a different weighting results by the same standard deviation. Therefore the standard deviations for quantities should always be in the range of 10-25% of the measurements (depending on accuracy).

3. Q1* and Q2* have a standard deviation of 10% - the shares of the target function are in the same order of magnitude.

4. As a final sample size is a parameter value (K-value) shown. Since this value is processed logarithmically, the standard deviation should be given in orders of magnitude from 0.1,…, 1.

		calculated	measured	Residual (calculated-measured)	Standard dev. (Sigma)	Proportion ZF (Residual²/sigma²)
1.	H1	65.285	66.040	-0.755	1	0.57
	H2	66.985	66.230	0.755	0.1	57.0
2.	Q1 [m³/d]	-10000	-5000	-5000	1	25000000
	Q2 [m³/d]	-0.116	-0.058	-0.058	1	0.0033
3.	Q1* [m³/d]	-10000	-5000	-5000	500	100
	Q2* [m³/d]	-0.116	-0.058	-0.058	0.006	93
4.	KWER [m/s]	1.5e-3	1.0e-03	Log(1.5e-3)-log(1.e-3)=-0.699	0.3	5.43

The following simplified illustration shows, why the initial values of the parameters can be crucial. The target function differs in size for different parameter values. There is an absolute minimum, as well as various local minima.

If you start the inverse modelling for example with the start parameter 1, the target function seeks to the associated local minimum. The target function for the start parameter 2 is on a local maximum and seeks to one of the two local minima.

The target function for the starting parameter 3 is at the highest, but the algorithm can reach the absolute minimum from this starting point. It is therefore quite reasonable to start the optimization process several times with different parameter values.
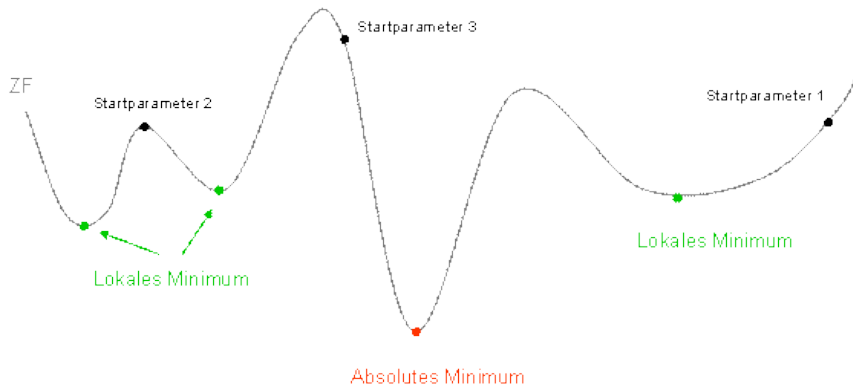
Fig. 13: Formulation of the target function

6.4.5 Realization in SPRING

Preparation in the model file

The parameters to be calibrated must first be defined in the model file. Thus, for example, water course sections are combined or the hydraulic permeabilities are divided into zones.

Within a zone, either a single K-value or storage coefficient (number of parameter values within the zone = 1) or a field of K-values or storage coefficients (number of parameter values within the zone > 1) can be calibrated.

For the calibration of leakage coefficients, the surface waters must be defined as LERA. A river or stream can be divided into several sections, these sections must be equivalent in LERA and Z-LR. Within a section only one value can be specified and optimized.

Structure of the parameter file

The parameter file must be in *.csv format. The columns are separated by a semicolon, as decimal the point is predetermined. The structure of the file looks like this:

	A	B	C	D	E	F	G	H	I	J
1	KWER	10	1102	6.26E-02	3.00E-01	7.00E-03	1.00E-20	1.00E+01	0	KWER1102
2	KWER	10	1429	1.00E-01	3.00E-01	8.70E-03	1.00E-20	1.00E+01	0	KWER1429
3	KWER	10	1432	7.46E-04	3.00E-01	3.00E-03	1.00E-20	1.00E+01	0	KWER1432
4	KWER	5	1995	1.02E-03	3.00E-01	1.02E-03	1.00E-20	1.00E+01	1	KWER1995
5	KWER	5	2789	1.10E-03	3.00E-01	1.10E-03	1.00E-20	1.00E+01	1	KWER2789
6	KWER	5	3650	5.09E-03	5.00E-01	5.09E-03	1.00E-20	1.00E+01	1	KWER3650
7	KWER	5	3689	8.60E-04	3.00E-01	8.60E-04	1.00E-20	1.00E+01	1	KWER3689
8	LERA	0	1	2.89E-05	5.00E-01	4.32E-05	1.00E-20	1.00E+01	0	LERA1
9	LERA	0	2	5.90E-06	5.00E-01	1.10E-05	1.00E-20	1.00E+01	0	LERA2
10	LERA	0	3	6.82E-05	5.00E-01	1.50E-04	1.00E-20	1.00E+01	0	LERA3
11	LERA	0	4	2.74E-05	5.00E-01	3.97E-05	1.00E-20	1.00E+01	0	LERA4
12	LERA	0	5	6.34E-05	5.00E-01	6.29E-05	1.00E-20	1.00E+01	0	LERA5
13	LERA	0	6	6.38E-06	5.00E-01	1.04E-05	1.00E-20	1.00E+01	0	LERA6

Fig. 14: Example of a parameter file

- In column A the type of the parameter is defined. The following types of parameters are currently implemented in the inverse modelling: KWER, KSPE, LERA, FLAE.

  For the areal calibration of KWER and KSPE zones can be defined in the model file (Z-KW, Z-SP). In the best case they should be consistent with the geological formations in which uniform parameters are to be expected. If only one formation exists, a single zone is defined.

- In column B the zones of Z-KW and Z-SP are taken over. The parameter LERA in column B is always 0. The assignment of Z-LR is made in column C.
- In column C is the element number of the base defined.
- Column D is the starting value for each parameter. In the first inverse calculation, this value is read.
- Column E contains the standard deviation. This corresponds approximately to the expected errors of the size. For potential measurements, this can be 5-10 cm, for quantities are 10% good values. In logarithmic variables such as KWER the standard deviation is given in orders of magnitude (eg K-value = 1e-03 + / - 0.5 magnitude).
- Column F contains the estimated value of the parameter for the generation of the residual.
- Columns G / H contain the lower and upper limit of the estimated value of column F.
- Column I: 0 for variable / 1 for fixed (making sure that the desired value is defined in column D!) → Shown in the following figure.
- Column J contains the name of the parameter that appears in the output.
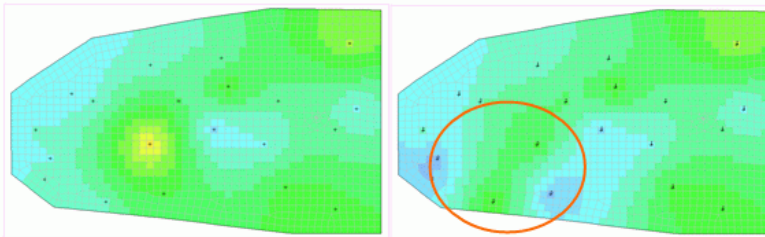
Fig. 15: Left: Bases and initial K-values. Right: The variable K-values are circled (column I=0). The K-values at the other bases are constant (column I=1), but have to stay in the parameter file for the interpolation.

Structure of the observation file

The existing measurements are saved in a file without a special format (e.g. text file). The measured values can be present steady state or transient. The structure of the file is as follows:

```
Line 1-3 comments
Line 4 empty
Line 5 ZEITEINHEIT MENG SEKUNDE (MINU STUN TAG_ JAHR)
Line 6 ZEITEINHEIT ZEIT SEKUNDE (MINU STUN TAG_ JAHR) OR
Line 6 BEZUGSDATUM XX.YY.ZZZZ
Line 7 empty
Line 8-x measured values:


POTE  Name_1  // comment (Attention: no blancs in the name!)
x-coordinate y-coordinate [z-coordinate in 3D]
Zeitpunkt Measurement Standard deviation OR
DATUM     Measurement Standard deviation


KNOT Name_2
Number of nodes n
n node numbers (10 per line)
Zeitpunkt Measurement (Sum over all n nodes) St.dev. OR
DATUM     Measurement (sum over all n nodes) St.dev


LKNO Name_3
Number of nodes n
n node numbers (10 per line)
Zeitpunkt Measurement (Sum over all n nodes) -St.dev. OR
DATUM     Measurement (Sum over all n nodes) -St.dev.
```

In line 5 and 6, the time units for the quantity and the time unit for the time scale are set. For steady state measurements, the time unit is defined as ZEIT with the time 0.0 in the file. For transient measurements can

be selected if an absolute time scale in a time unit (e.g. hour: 0.0, 1.0, 1.5, 2, 20) is used or if it is worked with a reference date and a date time scale.

From line 8 the measured values follow. There are 3 possible categories: POTE for potential measurements, KNOT for point withdrawal / infiltration amounts and LEKN for measured infiltration and exfiltration rates at a water course section. The control word for the category is followed by the name of the measuring point. It must be ensured that there are no blanks in the name.

A tab is not allowed between the x- and y-coordinate.

After the point of time follow the measured value and its standard deviation. The standard deviation is used for the weighting of this value. This is described in detail in chapter "Minimization of the target functions" on page 275 .

Examples of a steady-state and transient observation file follow:

```
 1 Messwerte stationäre Kalibrierung
 2 Datenquelle: xy.csv
 3 Region
 4
 5 ZEITEINHEIT MENG SEKUNDE
 6 ZEITEINHEIT ZEIT SEKUNDE
 7
 8 POTE 1-GW-                   //
 9  12407.135  20571.265
10 0.0   469.43   0.10      // Datum: 10/06/04
11
12 KNOT
13 3
14 28 29 30
15 0.0 -0.050 0.005
16
17 LKNO 62·
18 24
19 317 318 319 320 321 322 324 323 325 326
20 327 328 329 330 331 332 333 209 210 211
21 212 213 214 216
22  0.000 -0.037 -0.1
23
```

Fig. 16: Observation data for a steady state calibration

```
 1 Wasseramt
 2 instationaere Kalibrierung
 3 Jahr 2002
 4
 5 ZEITEINHEIT MENG SEKUNDE
 6 BEZUGSDATUM 31.12.2001
 7
 8 POTE   407
 9   11260.00    22405.00
10 01.01.2002    462.21      0.50
11 02.01.2002     462.2      0.50
12 03.01.2002    462.18      0.50
13 04.01.2002    462.17      0.05
14 05.01.2002    462.16      0.05
15 .
16 .
17 .
18 27.12.2002    462.51      0.05
19 28.12.2002     462.5      0.05
20 29.12.2002    462.49      0.05
21 30.12.2002    462.49      0.05
22 31.12.2002    462.48      0.05
23
24 POTE   410
25   11483.00    22617.00
26 01.01.2002    461.13      0.50
27 02.01.2002    461.12      0.50
28 03.01.2002    461.11      0.50
29 04.01.2002     461.1      0.50
30 05.01.2002     461.1      0.05
31 .
32 .
33 .
34 27.12.2002    461.25      0.05
35 28.12.2002    461.25      0.05
36 29.12.2002    461.25      0.05
37 30.12.2002    461.25      0.05
38 31.12.2002    461.25      0.05
39
```

Fig. 17: Observation data for a transient calibration

6.4.5.1 Input parameters

Via *Calculation → Inverse Calibration* and selection of the calculation
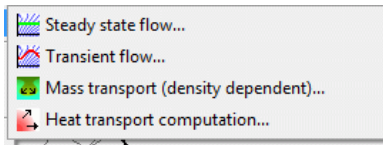
appears the input window of the selected calculation module.

The description of the input parameters, especially those of the extended settings can be found in the chapters: "Input parameters" or "Extended Settings" of each calculation module. In addition to the parameters described there the inverse calibration requires more information. The dialog for entering the parameters for the inverse calibration is another notebook side in the extended settings (button "Extended"):
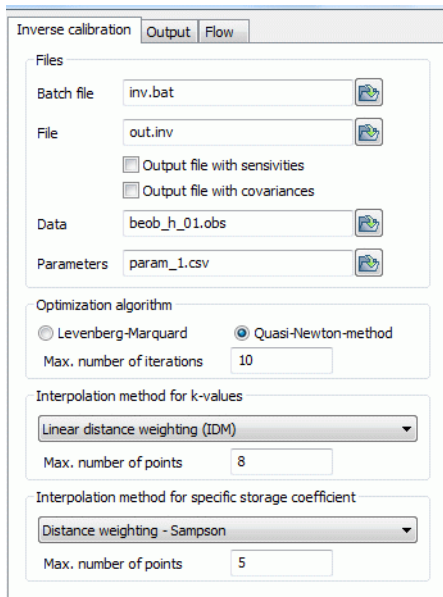
Files

An inverse calculation requires two batch files: The usual batch file with the parameters of the direct problem (*sitra.bsi)* and a batch file which contains the parameters of the inverse problem (default: *inv.biv*). The latter is defined here.

An inverse calculation creates two output files: The first output file contains the results of the latest direct calculation (*out.s*). Its name is defined in the main dialogue box. The second output file is named in this dialogue (default: *out.inv*). It contains the protocol of the inverse calculation with error tables for each iteration step, as well as the optimized parameters.

If the check boxes "output file with..." are enabled, the sensitivities (files *param_name.sns, out.inv_tab.csv* and *out.inv.sens*) and covariances (files *kovarianz.csv*) are written in additional output files.

The sensitivities provide a statement about how a change of the i.th parameter p_i in the current set of parameters p affects the calculation result of the j. th observation date b_j.

The covariance is a measure of the (linear) relationship between two random variables a and b with common distribution. The covariance is positive if a and b tend to having a linear relationship in the same direction, i.e. high values of a are associated with high values of b and low with low. On the other hand is the covariance negative, if a and b have a linear relationship in opposite directions, i.e. high values of a variable are associated with low values of the other variable. If the result is 0, then no linear relationship between the two variables a and b exists (non-linear relationships are possible) (from Wikipedia).

In the inverse modelling the covariances of the parameter values are calculated with respect to each other and written tabulated in the file *kovarianz.csv.*

The files containing the observation data and parameters are also specified here.

Optimization algorithm

Here you can select the optimization algorithm. Available are the Gauss-Newton algorithm based on Levenberg-Marquard and the Quasi-Newton algorithm.

The maximum number of iterations after which the optimization algorithm will be terminated regardless of the state of convergence has to be inserted.

Interpolation algorithm

This part defines the parameters for the interpolation of hydraulic conductivity or storage parameters within each zones (Z-KW or Z-SP), should such an interpolation be necessary. The available interpolation algorithms are:
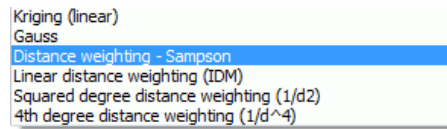
Using the distance weighting, the definition of a maximum search radius is not required. This is in contrast to the module *Interpolation*. The interpolation always uses maximally the number of points with the smallest distance given in the data file.

6.4.5.2 Results

It is recommended to perform a calculation with 0 iterations first. So without much effort, the target function can be controlled. The results are written in the output file (here: *out.inv*). The shares for each parameter/measurement group are listed individually.

It is important to balance the shares of the target function. After the first inverse calculation (0 iterations), the shares of the target function are compared. The best-known sizes (usually the potential heads) should have the largest share of the target function (ZF).

Possible sources of errors:

■ K-values are weighted too heavily. This forbids the algorithm to calculate the K-values far away from the estimated value, so there are almost no changes in the K-values.

   Remedy: Increasing the standard deviation

■ Quantities are weighted too heavily. Thus the measured potential heads lose their weight and the algorithm tries (in vain!) to bring the volumes to match. In most cases, the rule applies: If the potentials and the K-values fit, the amounts are also set in.

   Remedy: Weighting the shares of the target function balanced (using well-chosen standard deviations)

■ Transient hydrographs: The piezometer x is measured every day for over a year, in contrast, piezometer y is measured only once per month. When all measurements have the same σ (i.e. equally weighted), the well x is weighted about 30 times stronger than y (365 measurements compared to 12 measurements).

   Remedy: Choosing a less σ for well y

If the shares of the target function are balanced, the number of iterations is set to 10 and the inverse calculation is started again.

Output files

■ out.inv

   File with the protocol of the calibration (measured / calculated value / target function) and the K-values and leakage factors in the format of the model file. It is possible to assign the iterated K-values directly to the attribute KWER via the menu item *Attributes → Import data/computational results → iterated K-values.*

■ Invpar.csv:

   File in the format of the parameter file (Note: semicolon as the column separator, point as decimal). In column D, the calculated K values are listed. This file can be renamed and then used as input file for a next calculation (e.g. with partially constant values).

■ Invpar_xy.csv:

   Analogous to the file invpar.csv, but additionally with coordinates (not for LERA!). The file can be overlayed, for example, to visualize the results.

■ resan_pote.csv, resan_lkno.csv, resan_knot.csv (steady state only):

   Contain all data of the measured values

	A	B	C	D	E	F	G	H	I	J
1	Name	X	Y	Z	Calculation	Measured	Residual	Std.dev	W-Res	ZF
2	1	257	1482	0	102.854	103	-0.146	1.00E-01	-1.46E+00	2.12E+00
3	2	1046	1075	0	102.751	103.2	-0.449	1.00E-01	-4.49E+00	2.02E+01
4	3	2228	884.12	0	102.09	102	0.09	1.00E-01	8.96E-01	8.03E-01
5	4	2850.67	1940.99	0	101.806	102	-0.194	1.00E-01	-1.94E+00	3.76E+00
6	5	4285.02	2694.89	0	100.858	101	-0.142	1.00E-01	-1.42E+00	2.02E+00
7	6	4856.81	482.17	0	100.251	99.8	0.451	1.00E-01	4.51E+00	2.04E+01
8	7	3843.76	1455.22	0	100.915	101.04	-0.125	1.00E-01	-1.25E+00	1.57E+00
9	8	1652.76	1000.22	0	102.378	103.04	-0.662	1.00E-01	-6.62E+00	4.38E+01
10	9	3768	81.7	0	101.296	101.49	-0.194	1.00E-01	-1.94E+00	3.75E+00
11	10	3632.74	1870	0	101.141	101	0.141	1.00E-01	1.41E+00	2.00E+00

Fig. 18: Example of the file resan_pote.csv

■ name.csv (transient only):

For each measurement point (POTE, LKNO, KNOT) a *.csv file is stored with time series data. There will be no difference according to data types, so it is important that the names in the observation file (*.obs) are clearly defined (not a potential head called X1 and a quantity measurement called X1 (better: h_x1 and q_x1).

	A	B	C	D	E	F	G	H	I	J
1	Time	Calculation	Measured	Residual	Std.dev	W-Res	ZF			
2	1	446.613	446.59	2.31E-02	2.00E-01	1.15E-01	1.33E-02			
3	2	446.616	446.57	4.61E-02	2.00E-01	2.30E-01	5.30E-02			
4	3	446.617	446.55	6.68E-02	2.00E-01	3.34E-01	1.11E-01			
5	4	446.616	446.53	8.62E-02	2.00E-01	4.31E-01	1.86E-01			
6	5	446.614	446.51	1.04E-01	5.00E-02	2.08E+00	4.32E+00			
7	6	446.611	446.5	1.11E-01	5.00E-02	2.21E+00	4.89E+00			
8	7	446.607	446.49	1.17E-01	5.00E-02	2.34E+00	5.46E+00			
9	8	446.603	446.48	1.23E-01	5.00E-02	2.45E+00	6.02E+00			
10	9	446.598	446.46	1.38E-01	5.00E-02	2.76E+00	7.60E+00			
11	10	446.593	446.45	1.43E-01	5.00E-02	2.85E+00	8.14E+00			
12	11	446.588	446.44	1.48E-01	5.00E-02	2.95E+00	8.71E+00			
13	12	446.582	446.43	1.53E-01	5.00E-02	3.05E+00	9.30E+00			
14	13	446.577	446.43	1.48E-01	5.00E-02	2.95E+00	8.70E+00			
15	14	446.573	446.42	1.53E-01	5.00E-02	3.05E+00	9.31E+00			
16	15	446.568	446.41	1.58E-01	5.00E-02	3.16E+00	9.96E+00			

Fig. 19: Example of a transient output file name.csv

6.4.5.3 Batch files

It is possible to execute the *inverse calibration* directly in the command line without using the dialog box, if a batch-file *. bsi* exists in the directory. For this you have to change to the directory in which the calculation is to be executed. The input of *sitra* and confirming with the Enter key starts the calculation. The default file name is *sitra.bsi*. This file is searched in the directory.

To use another batch file, type *sitra file name* (the extension ".bsi" is automatically attached). Batch files can be created, viewed and modified with the help of any text editor

■ Example of a batch file *sitra.bsi* (steady state calculation):

```
out.s # Output file
1 # detailed protocol (1-yes/0-no)
1 0 # Equation solver P/U (1-iterative, 0-direct)
1 0 # Calculation of control lines (1-yes, 0-no), IMITTEL
0 0 # Peclet numbers, hydr. cond. from EICHEN (1-yes, 0-no)STROEMUNG+INVERS inv.biv  #
inverse modelling without transport
1 # satur./unsat. computation (1-yes,0-no)
0 0 0 # equal conc., density-related calc., p or h (1-yes/0-no)
5 0.5 # Number of iteration steps, factor of attenuation
0 0 # Flow/transport steady st./0 trans./1 trans.+conf./unconf./2
0 0 # Initial values POTE/KONZ from ASAT(3) null(2) EICH/AKON(1) =0(0
```

■ Example of a batch file *inv.biv* (inverse calculation), called from the steady state calculation:

```
out.inv          # output file for inverse modelling record
invers.obs       # observation data file
invers.csv       # parameter file
1                # optimization with Levenberg-Marquard / 1: Quasi-Newton
10               # maximum number of iterations
```

```
KWER 1 0 # interpolation flags(if necessary) of KWER parameters / parameter d for distance
weighting
SPEI 10 3 # interpolation (if necessary) of KSPE/parameter d for distance weighting
output = SENS;KOVAR
```

Meaning of the value of the interpolation flag:

```
    0: Kriging
 1: Gaussian
 10: distance weighting – Sampson
 11: distance weighting 1/d
 12: distance weighting 1/d^2
 13: distance weighting 1/d^4
```

6.4.6 General procedure

If model parameters are estimated by inverse modelling, the following procedures are generally recommended:

■ The inverse modelling should at first be started with a relatively coarse zoning of the data, i.e., only a few global parameters are optimized. The evaluation of the inverse modelling with a coarse zoning usually provides good information about whether the general zoning is appropriate. If this is the case, detailed classifications can be made.

■ If the "coarse" inverse modelling is completed individual parameters can be further refined in some areas. It makes sense to take out the areas with sufficiently accurate results from the inverse optimization. The optimized model data are required in the model file and removed from the parameter file. This reduces the number of parameters to be optimized and limits the computational cost for an inverse calculation.

■ Often there are areas of the model in which there are only a few observation data. Then it is usually difficult to find meaningful parameters using the inverse modelling. Even if good results in the rest of the model area are achieved, there are often "runaways" in the optimized parameters of these critical areas. There are essentially two options, to get a grip on the parameters in areas with such little information:

1. The parameters are removed from the inverse modelling and specified with an estimated plausible value in the model file.

2. There is also the possibility to assign "dummy" observation data in this area: the user can, for example, introduce estimated additional potential heads as observation data. These should then get smaller weighting parameters - as opposed to the actual measured potentials - because they are less credible than typically measured values.

■ For the transition from a coarse to a finer differentiation of the zoning parameters, there are two different approaches:

A zone can be divided into smaller zones, or

The number of parameters within the zone can be increased.

In general, the results of the optimization are to be interpreted more easily when using zones with only one parameter (→ constant parameters within the zones). If zones with multiple parameters are used, the results of the optimization have to be evaluated according to the used interpolation algorithm. The optimization is then additionally vulnerable to weaknesses in the used interpolation algorithm. These are for example over- or under oscillations in Gauss and Kriging, or the tendency to local extrema in the interpolation using the distance weighting.

Evaluation of the sensitivities

The inverse modelling requests constantly a review by the user of the conceptual model. An evaluation of the sensitivities at the end of each inverse calculation provides this important information:

■ Model areas where are not enough information can be detected:

If all parameters in a model area show only small sensitivities, this area should either provided with additional observational data or be removed from the calibration.

■ Critical and non-critical parameters:

When comparing the sensitivities it is obtained, which of the parameters has the greatest influence on the error in a defined observation point.

■   Wrong model assumptions:

    If the iteration diverges or there is only a minimal reduction of errors a general review of the model assumptions (e.g. the distribution of the zones) is required.

6.4.7 Restrictions for the inverse modelling

For the calculation of the direct problem control parameters must be defined during the inverse modelling. Here are some limitations to be considered:

■   Generally, a coupled flow and transport model can be calculated. Dispersivities and other model data only required for the transport are currently not optimized and measured concentrations are not included as observation data. Therefore, the transport calculation is currently not scheduled in an inverse modelling. An exception is the optimization of flow parameters for a density-dependent model. In this case, the transport problem must be calculated as well.

■   In general, it is possible in a transient calculation, to save the calculated potentials for certain time steps or time series of individual nodes in order to reduce the required storage space. In the inverse modelling, however, the potential heads are needed for each time step of the transient calculation. Therefore the potential heads must be stored for all time steps in a transient inverse modelling.

■   The continuation of a transient calculation (a "warmstart") with the null-file is in an inverse modelling calculation not possible. Thus, storing the data of the transient calculation for a continuation of the iteration (out66-file) does not make sense.

■   In a transient three-dimensional or vertical flow model, which is calculated saturated/unsaturated, the relative K-values can be calculated iteratively within a time step. This is in an inverse modelling not possible. When such problems occur it must be worked with an update of the relative K-values from one time step to the next (i.e. only one iteration per time step and without attenuating the iteration: attenuation factor = ca. 1.0).

■   As a further limitation with respect to a normal flow calculation the inverse modelling does not work with minimum or maximum in- or exfiltration rates (MXKI, MXKE, MXEI, MXEE). To keep leakage rates still in a certain size, the observation data LKNO can be used.

6.5 Steady state flow

Call via: *SPRING → Calculation → Steady state flow…*

Currently are two calculation modules available to calculate the steady state flow. These are the modules GEONEU and SITRA. The module GEONEU has features that are missing in the module SITRA:

■   Calculation of a 2D model combined with a partially 3D model

■   additional potential value in the element centre used in the streamline calculation based on a triangular or quadrilateral decomposition of the individual elements

■   path line calculation

The notes of the following chapters, however, refer only to the steady state flow calculation with the module SITRA.

6.5.1 Theory of the steady state flow

Formulation of the potential

Assuming a saturated aquifer, constant density and steady state conditions, the flow equation is based on Darcy's Law:

$$\mathbf{v_f} = -\mathbf{K}\nabla\mathbf{h}$$

and the fluid mass balance

$$\nabla\mathbf{v_f} = \mathbf{q}$$

that can be combined to

$$-\nabla(\mathbf{K}\nabla\mathbf{h}) = \mathbf{q}$$

With:

h = Potential head [m],
v_f = Darcy velocity [m/s],
K = symmetric tensor for the permeability [m/s],
q = source/sink term [1/s].

For the flow in a homogeneous, isotropic medium, the K-value can be regarded as a scalar variable. In an anisotropic aquifer the tensor property of the K-value must be considered, since the permeabilities in one point are different in size in different directions.

Besides the mathematical formulation of the steady state flow equation with the potential h, the pressure p or the pressure head ψ can be used as the primary variable. This is particularly necessary when density-dependent flow conditions are considered, or a saturated/unsaturated calculation should be carried out. The sizes of "saturation" and "relative K-value" of saturated/unsaturated conditions are formulated in the literature as well as from the pressure or the pressure head dependent variables.

6.5.1.1 Formulation of the pressure equation

Hubbert's Law is a relationship between the hydrostatic pressure p and potential head h:

$$\mathbf{h} = \mathbf{z} + \frac{\mathbf{p}}{\rho\mathbf{g}}$$

with:
h = potential head [m]
z = elevation [m]
p = pressure [N/m²] = [kg/(s²m)]
ρ = Density [kg/m³]
g = gravitational acceleration [m/s²]

One of the flowing medium independent parameter for the permeability K is the permeability K_{perm}. The relationship between the permeability K_{perm} of a porous medium and the permeability K of the porous medium in terms of a flowing fluid is:

$$\mathbf{K} = \mathbf{K_{perm}} * \frac{\rho\mathbf{g}}{\eta}$$

with:

K = symmetric tensor for the K_f-Wert [m/s]

K_{perm} = Permeability [m²]

ρ = Density [kg/m³]

g = gravitational acceleration [m/s²]

η = dynamic viscosity [kg/(ms)]

Inserting these two equations in the from the potential dependent steady state flow equation, we obtain the steady state flow equation as a function of pressure:

$$-\nabla(K\nabla h) = q = -\nabla(\frac{K_{perm}}{\eta}(\rho g \nabla z + \nabla p))$$

The unit of the equation is that of the source term q = [1/s].

Since unsaturated conditions are to be taken into account, the K_f-value has to be scaled with a relative k-value kr_{el} ($0 < kr_{el} < 1$).

$$-\nabla(K\nabla h) = q = -\nabla(\frac{K_{perm} * k_{rel}}{\eta}(\rho g \nabla z + \nabla p))$$

The factor k_{rel} [-] is defined as a function of the saturation: $k_{rel} = k_{rel}(S_r)$. In the saturated zone is $k_{rel} = 1.0$.

The relative saturation is a variable, dependent of the pressure $S_r = S_r(p)$, which is usually defined with a function according to van Genuchten.

6.5.1.2 Pressure-saturation function by van Genuchten

The non-linear relationship between capillary pressure and saturation is primarily determined by the nature of the soil (grain shape and grain size distribution). The following figure shows schematically the pressure-saturation relationship in the soil:
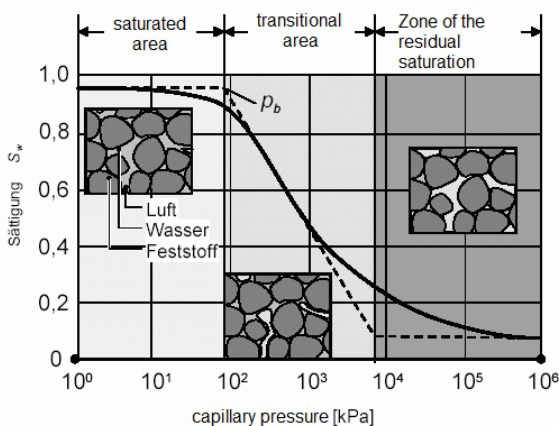
Fig. 189: Capillary pressure-saturation relationship (p_b means the water inflow pressure)

Van Genuchten defined the following relationship between the proportion of unsaturated from the saturated hydraulic conductivity:

$$\mathbf{k_{rel}(S_e) = (S_e)^l \left[1 - (1 - (S_e)^{\frac{1}{m}})^m \right]^2}$$

Meaning:

S_e = effective saturation

l = unknown parameter, determined by van Genuchten with the value 0,5

m = 1-1/n = constant, n = pore size index (> 1)

Van Genuchten [VAN GENUCHTEN, M. TH. (1980): A Closed-Form Equation for Predicting the Hydraulic Conductivity of Unsaturated Soils, Soil Science Society of America Journal. 44: Pages 892-898. 1980] defined the values 1,5 (clayey) to 4,5 (sandy) for n.

The effective saturation is defined by the following formula:

$$S_e = \frac{S_r(p) - S_{res}}{S_s - S_{res}}$$

The parameters are:

S_r = relative saturation $S_r(p)$

S_{res} = Residual saturation = minimum saturation level, which exists as a result of flow processes and depends on the type of soil

S_s = maximum saturation = saturation level which is depends on the soil type and is at maximum attainable ($S_s \sim 1.0$)

The effective saturation is related to the capillary pressure p_c and the water inflow pressure p_e as follows:

$$S_e = \frac{S_r(p) - S_{res}}{S_s - S_{res}} = \left[1 + (\frac{p_c}{p_e})^n \right]^{\frac{1-n}{n}}$$

Solving this equation for the pressure-dependent relative saturation $S_r(p)$, you obtained the pressure-saturation function according to van Genuchten on which the calculation in SPRING is based.

$$S_r(p_c) = S_{res} + (S_s - S_{res}) * \left[1 + (\frac{p_c}{p_e})^n \right]^{\frac{1-n}{n}}$$

With:

p_c = capillary pressure [N/m²] = [kg/(s²m)]

p_e = water inflow pressure [N/m²] = [kg/(s²m)]

The water inflow pressure is a soil specific parameter. It is defined as the inverse of $a = 1/p_e$ in the Saturation parameters. The following figure shows examples of the pressure-saturation functions according to van Genuchten depending on various soils:
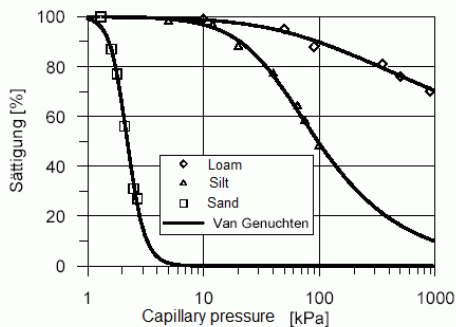
Fig. 190: Pressure-saturation functions according to van Genuchten

6.5.2 Calculation of the free surface (2D)

Determination of the free surface in a 2D horizontal model is equivalent to the computation of the unknown thickness. The computed thickness (m) of an unconfined aquifer must be such that it ensures that the resulting potential head (h) fulfils the equation

$$m = h - u$$

in which u represents the lower boundary of the aquifer. The iteration of the thickness is carried out for each single element. In each step, the thickness is determined with:

$$m_{i+1} = h_i - u.$$

with: i = iteration step

To avoid oscillations, the change of the thickness for the next step of iteration is multiplied by an attenuation factor $0 < w < 1$:

$$m_{i+1} = m_i + w * ((h_i - u) - m_i).$$

For w=1.0, the iteration is not attenuated at all; for w=0, the attenuation is maximal. The default value is 0.5. The computed thickness values from the model checking are used as initial thickness for the iteration.

The iteration process is continued until the difference between the two computed potential heads is less than an internal limit, or until the user-defined maximum number of iteration steps is reached.

For the case of a confined aquifer, the groundwater surface cannot move upwards. The iteration takes this into account, and the computed thickness is not allowed to be larger than the maximum (possible) thickness computed in the model checking.

The definition of the lower boundary of the aquifer (UNTE) is absolutely necessary in order to carry out an iteration of the thickness in a 2D model.

Example:

For instance, a calibrated state is to be verified where the model file contains initial potential heads (EICH) and the lower boundary of the aquifer (UNTE). Hence it follows that the initial thickness is actually very close the thickness to be found. Therefore, an iteration using an attenuation factor w=1.0 will converge rapidly and approach the final thickness.

Generally, further calculations consider a new groundwater situation, e. g. changing exfiltration rates. Hence the iterated thickness in the corresponding area will be different from the initial thickness. Rapidly changing boundary conditions or a "bad" initial thickness makes the attenuation necessary in order to avoid possible oscillations.

Special feature of the module SITRA:

In the module SITRA the thicknesses are stored as node values. Therefore the initial and maximum thicknesses computed in the model checking have to be averaged on node values, which may eventually lead to a loss of accuracy.

6.5.3 Calculation of the free surface (vertical- or 3D model)

In a vertical or rather a 3D model a numerical modelling of a partly unsaturated groundwater aquifer can be carried out. This is carried out in SITRA at basis of the pressure-saturation functions and the function of the relative K-value for van Genuchten.

In SITRA three sets of parameters for the parameters residual saturation (S_{res}), inverse of the water inflow pressure (a), pore size index (n) and specific parameters of the relative K-value function (l) are predefined. Depending on K-values defined the in the model file they are automatically assigned to the appropriate soil types and van Genuchten parameters.

The exact allocation of the K-values and the soil types is given in a table in the chapter: "Data structure of a groundwater model, saturation" p. **44**.

6.5.3.1 Calculation with van Genuchten-functions

This time the unsaturated parts of the aquifer are fully included in the finite element calculation. The algorithm uses the pressure equation. At each node, the saturation $S_r = S_r(p)$ is obtained from the actual pressure p. Based on the saturation at all nodes, the relative K-value $k_r = k_r(S_r)$ for each element is computed:

pressure p_i → Saturation $S_{r,\,i+1}$ and relative K-value $k_{r,\,i+1}$

with: i = iteration step

The change of the relative permeability from one step to the other is attenuated with the attenuation factor $0 < w < 1$ to avoid oscillations:

$k_{r,\,i+1} = k_{r,\,i}) + w * (k_{r,\,i+1} - k_{r,\,i})$

with: i = iteration step

For w=1.0, the iteration is not attenuated at all; for w=0, the attenuation is maximum. The default value is 0.5. The initial state for the iteration is either obtained by relative permeabilities computed from initial potential heads (EICH), from the potential heads stored in the null-file p. **28** or from the attribute ASAT (initial saturation). Else, the relative permeabilities must have the initial value 1.0.

The iteration process continues until the difference between the two computed potential heads is less than the user-defined limit (limit of iterations.

Remark:

Poorly-chosen parameters can lead to numerical problems in the iteration of the free surface. In the worst case, the iteration may not converge to a solution.

The main reason for this is high areal recharge rates in unsaturated areas. These can cause oscillations.

In the case that the elements are "open" (k = 1 or rather k_r = 1), the free surface drops beneath the elevation of the elements. Hence it follows that the elements are unsaturated in the next iteration step, thus made "impermeable" (k → 0 or rather. kr -→ 0). Then, high areal recharge rates cause a "potential head mound" above these impermeable elements. Therefore, in the next step they will have full saturation again.

A strong attenuation coefficient can, in most cases, prevent the oscillations. However, the attenuated iteration takes longer to converge, so the number of iteration steps must be increased.

6.5.4 Input parameters

Via *Calculation → Steady state flow* the following input window appears:
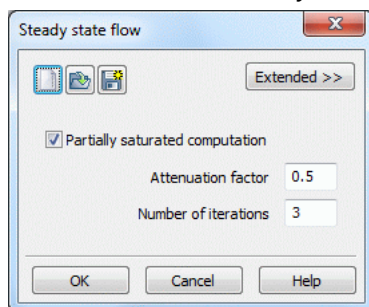
Fig. 191: Input for the steady state flow calculation (3D model)

When calling the module, the program reads the batch file with the default name (here: sitra.bsi), the module SITRA is predefined as well. For the calculation with the module GEONEU the desired batch file has to be explicitly defined.

Iteration of thickness (2D model) / partially saturated computation (3D model)

This part is related to the iteration of the thickness in a 2D horizontal model or the partially saturated computation of the free surface in vertical or 3D models. The user must decide whether the iteration is performed. Furthermore, the maximum number of iterations and the attenuation factor must be defined.

Attenuation factor

In order to avoid oscillations in the iteration, the change of the thickness of one iteration to the next is provided with an attenuation factor 0 <w <1.

The buttons at the top of the input window allow resetting the input parameters ( ), opening an existing batch file ( ) or saving the current batch file under another name ( ).

The selection "Extended" leads to further settings.

6.5.5 Extended settings in the module SITRA

By activating the "Extended" button other parameters can be set. First, the inputs of the module SITRA are described:
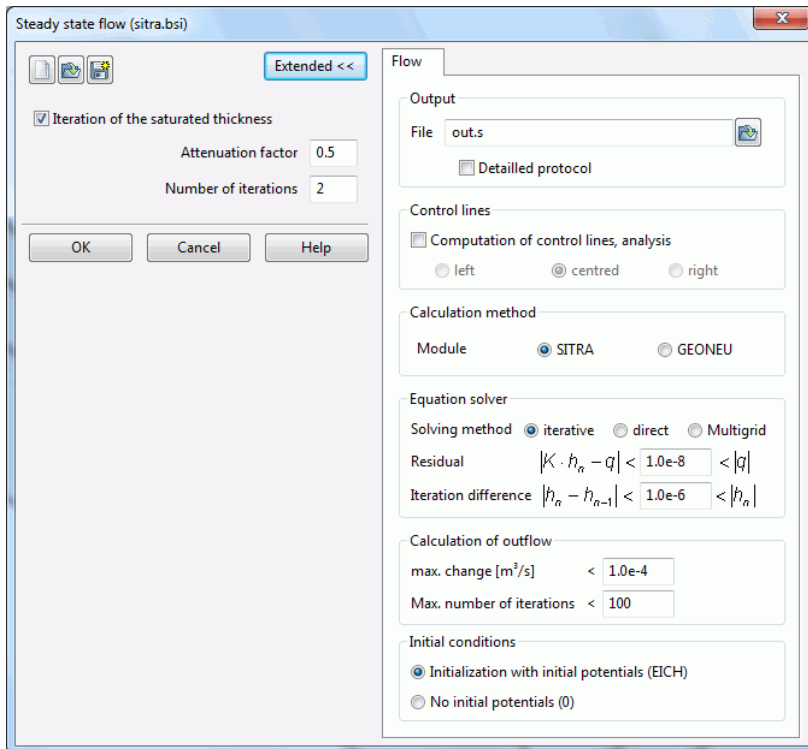
Output

The name of the output file is entered, default is the name *out.s* in the module SITRA and *out.g* in the module GEONEU.

In a detailed protocol (only SITRA) the intermediate time steps of a transient calculation are stored as well as the mass balances (for each node), calculated volumes of control lines (for each node), calculated potentials, concentrations or temperatures, saturations (by a saturated / unsaturated calculation), and velocities.

Control lines

(only if the attribute KONT is defined in the model file)

After checking the box the computation of the control lines is done. For each segment of the control line, a velocity is required. The velocity can be determined as the mean value over the two adjacent elements. Otherwise, the velocity of the left or the right element is chosen.

Calculation method

Possible are the modules GEONEU and SITRA. Default is the calculation with SITRA.

Equation solver

As an equation solver either the iterative PCG solver, the direct Cholesky solver (LU decomposition) or an algebraic multigrid solver can be selected. For meshes of about 500 nodes or more, you should always choose the iterative one, because it is much faster. The error due to the iterative solution is always much

smaller (less than 3%) than the other errors (e.g. measurement error). This can be verified with the help of the mass balance printed to the output file. The multigrid solver is particularly recommended for models with large local differences in the permeabilities.

- Iterative PCG solver: General CG method with preconditioning (Preconditioned Conjugate Gradient Solver) with block-SSOR relaxation (Symmetric Successive Over Relaxation Method)
- Direct Cholesky solver: Gaussian elimination method
- Multigrid solver: CG method with preconditioning by multigrid methods

Residual and iteration difference (iterative and multigrid solver only)

The basis of the iterative solution method is the boundary value problem defined as variational problem from which by discretization a quadratic function F (v) results whose minimum gives the desired solution. The partial differentiation of F produces the known symmetric and positive definite system of equations as a necessary condition for the minimization:

$$\mathbf{K\,h^* = q}$$

In the following the exact solution vector is named **h*** is named, now to be determined by an iterative process. With **h** as so-called trial vector, the quadratic function can be written as:

$$F(h) = \frac{1}{2}h^T K h - q^T h$$

Or in index notation:

$$F(h) = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}k_{ij}h_i h_j - \sum_{i=1}^{n}q_i h_j$$

The partial differentiation to $\mathbf{h_i}$ leads to:

$$\frac{\delta F}{\delta h_i} = \sum_{j=1}^{n}k_{ij}h_j - q_i$$

δF / δh$_i$ is the i-th component of the gradient or for the residual vector **g** for the trial vector **h,**

$$g = \frac{\delta F}{\delta h_i} = K h - q \longrightarrow 0$$

Which now will be lead toward the value zero within the iterative process (or in this case: 1.0e-08).

The iteration process that leads the residual vector **g** to zero, contains a systematic correction of a sequence of approximation vectors **h** in the following form:

$$\mathbf{h_{i+1} = h_i + \eta_i\,d_i}\ \ ,(i = 0, 1, 2, \ldots)$$

Therein the trial vector **h$_i$** of the i-th iteration is improved to the new approximation vector **h$_{i+1}$** using the iteration vector **d$_i$** and a scalar step size **η$_i$.**. The vector **d$_i$** gives the direction of the relaxation which is oriented to the minimum of the function.

The iteration process is terminated when a pre-set convergence limit (iteration difference) ε is reached or exceeded, so that the accuracy of the result to be achieved is determined by the choice of this convergence measure (iteration difference) ε.

The various vectors which are involved in determining the termination criterion are related by forming their Euclidean norm. This results in the following termination criterion:

$$\left| h_n - h_{n-1} \right| < \varepsilon$$

Calculation of outflow

The input data for the flow calculation is described in detail in chapter: Input for water course system calculation.

Initial conditions

When selecting the module SITRA, the initial conditions for the steady state flow can additionally be determined. It is possible, to take over the initial potential heads from the attribute EICH or from the null-file, or the initial potentials can be set to 0.

6.5.5.1 Input for water course system calculation

By using the water course systems in SPRING it is ensured that the upstream quantities of the groundwater are considered downstream in the exchange balance. This ensures that when a water section dries up only so much water can trickle away, as the existing upstream amount, for example seasonally, allowed.
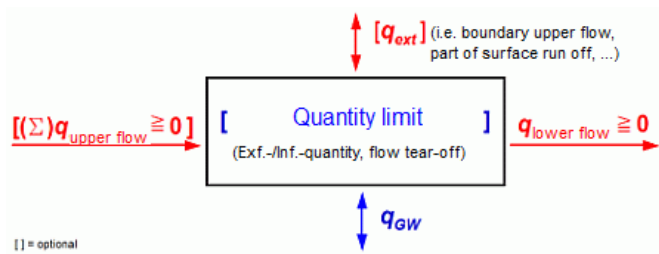
The basic formula for the simplified derivation of the water level in the water course is:

$$h = \left(\frac{q}{b \times k_{St} \times \sqrt{I}} \right)^{\frac{3}{5}}$$

Quelle: *U.S. Geological Survey: Open-File Report 88-729; Formel 5; 1989*

The formula is valid only at a ratio of b >> h.

With:

- Water level (over bottom oft he water course) h
- outflow q
- bottom slope I
- Manning/Strickler coefficient k_{St}
- Wetted perimeter b

If the attributes (VINZ, VGRD, VSYS) for the water course system calculation are assigned in the model file, an additional input block appears when selecting the module SITRA:

Maximum change [m³/s]

In determining the groundwater runoff at a node, the quantities which flow from the upper section to this node are taken into account. These leakage rates are at each iteration added together from the upper course to the considered node, to evaluate how much water can trickle away at the appropriate potential (< water level).

This in turn causes a change in the leakage rates and thus a change in the groundwater runoff. This iteration is limited by specifying a maximum volume change.

Maximum number of iterations

The iteration of the leakage rates can lead to a very high number of calculation steps before the max. (volume) change is achieved. In extreme cases, no convergence is achieved at all. Therefore, at this point, the number of iterations is limited.

Once a restriction (max. volume rate or max. number of iterations) is reached, the iteration of the groundwater runoff stops.

6.5.6 Extended settings in the module GEONEU

In the module GEONEU additional calculations and entries are possible:

Initial values

(3D model only, module GEONEU only)

This determines on which the potential initial values for the calculation of the relative K-values are based.

Calculation method

For a horizontal model with a partially 3D model and for the pathline calculation the module GEONEU has to be selected because SITRA is not able to calculate this type of model and doesn't do a pathline calculation. The theory of the pathline calculation is explained in the chapter "How To" on p. 397.

Equation solver

When selecting the module GEONEU only the iterative solver or the direct solver are available. The different solvers as well as the termination criteria have been described in the chapter "Extended settings in the module SITRA" (p. 290).

6.5.7 Results of the steady state flow computation

The finite element calculation provides the following results (partly on request)

- Potential head at each node and the centre of each element (primary result)
- Velocity for each element: the module SITRA outputs the field velocity (v_a), the module GEONEU outputs the Darcy velocity (v_f), ($v_a = v_f/n$)
- The velocity field is saved in the background files and can be visualized with the software STRING (p. 304).
- Distance between ground level and groundwater level at each node where the input file contains data for the ground level
- Mass flow rates (inflow and outflow rates) at each node where the potential head (POTE) is defined as a boundary condition

- Leakage rates at nodes and elements. The leakage coefficients and the receiving water course levels must be part of the input data. Leakage rates at nodes and elements where the input file contains leakage coefficients (LERA, LEKN, LEEL) and the receiving water course level (VORF

- Mass balance for the total of infiltration and exfiltration rates. It is compared to the total of mass flow rates. The total of the defined and the calculated rates must be close to zero. Besides, the infiltration (positive) is compared to the exfiltration (negative) and the areas with BILK and BILE are balanced.

- Mass flow rates at control lines defined in the input file (see control line calculation)

- Aquifer thickness for each element (in the case of a horizontal model which was calculated with an iteration of thickness)

- Path lines on the basis of the calculated velocities

6.5.8 Batch file steady state flow

It is possible to execute the *steady state flow* directly in the command line without using the dialog box, if a batch-file **.bge* (for GEONEU) or **.bsi* (for SITRA) exists in the directory. For this you have to change to the directory in which the calculation is to be executed. The input of *geoneu* (or *sitra*) and confirming with the Enter key starts the flow calculation. The default file name is *geoneu.bge* or *sitra.bsi*. This file is searched in the directory.

To use another batch file, type *sitra* (or *geoneu) file name* (the extension ".bge" or ".bsi" is automatically attached). Batch files can be created, viewed and modified with the help of any text editor.

A detailed batch file can be viewed in the online help. Similarly, there is a batch file of the steady state flow for further processing provided.

6.6 Transient Flow

Call via: *SPRING → Calculation → Transient flow*

In the transient flow calculation, the change in the flow condition over the time is taken into account.

The change of state takes place through time-varying boundary conditions. Classical applications are, for example, flood waves in the main receiving water course or time-varying well withdrawals.

Currently are two calculation modules available to calculate the transient flow. These are the modules INSTAT and SITRA. The module INSTAT has two features that are currently missing in the module SITRA.

- Calculation of a 2D model combined with a partially 3D model

- Additional potential value in the element centre used in the streamline calculation based on a triangular or quadrilateral decomposition of the individual elements

The notes of the following chapters, however, refer only to the transient flow calculation with the module SITRA.

The computation of the thickness and the free surface is already described in chapter "Steady state flow", p. 284.

6.6.1 Theory of the transient flow

In the chapter "steady state flow" the differential equation for a steady state flow with sources and sinks term is derived, as potential and as a pressure equation.

In the saturated, density independent case the transient flow equation as a function of pressure is formulated like this:

$$S_0 \frac{\partial h}{\partial t} - \nabla \left(\frac{K_{perm} * k_{rel}}{\eta} (\rho g \nabla z + \nabla p) \right) = q$$

That means:

S_0 = spezific storage coefficient [1/m]

h = potential [m], (sum of pressure- and potential height)

t = time [s]

K_{perm} = permeability [m²]

k_{rel} = scaling factor of the relative k-value [-]

η = dynamic viscosity [kg/(ms)]

ρ= density [kg/m³]

g = gravity [m/s²]

z = layer height [m]

p = pressure [N/m²],

q = sources/sinks term [1/s].

The transient pressure flow equation requires the specific storativity S_{0p}. It depends on the flow effective porosity n of the aquifer, the matrix compressibility α and the fluid compressibility β.

$$\mathbf{S_{0p} = \alpha(1-n) + \beta n}$$

with:

S_{0p} = specific pressure storage coefficient [m²/N] = [ms²/kg]

n = flow effective porosity of the aquifer [-]

α = matrix compressibility [m²/N] = [ms²/kg]

β = fluid compressibility [m²/N] = [ms²/kg]

Note: If the compressibility of the entire system (attribute KOMP) is defined in the model file, the specific pressure storage coefficient (S_{0p}) is substituted by the compressibility of the entire system.

After including the density ρ, the specific pressure storage coefficient S_{0p} and a - caused by changes of the density - transient source/sink term you get the transient saturated/unsaturated density-dependent flow equation as function of the pressure:

$$\mathbf{(S_r \rho S_{0p} + n\rho \frac{\partial S_r}{\partial p}) \frac{\partial p}{\partial t} + nS_r \frac{\partial \rho}{\partial t} - \nabla(\rho \frac{K_{perm}k_{rel}}{\eta} (\nabla p + \rho g \nabla z)) = q}$$

With:

$S_r = S_r(p)$ = relative saturation depending on the pressure [-]

S_{0p} = specific pressure storage coefficient [m²/N] = [ms²/kg]

n = flow effective porosity of the aquifer [-]

The unit of the equation corresponds in this case to q = [kg/m³s]

6.6.2 Computation of the storage coefficient

Computation in **horizontal models**:

For each element (SPEI) or for each node (KSPE) in transient calculations, a storage coefficient S [-] has to be entered for an unconfined aquifer (S = S_0 * thickness m).

In confined aquifers, the set specific storage coefficient S_0 = S / thickness = 3.3 10^{-6} [1/m] is used. This set value can be changed during the input dialog of the transient calculation.

In **vertical- and 3D models** the flow equation as a function of the pressure is used and the storage coefficient is computed using these parameters:

- Pressure p,
- Saturation S_r,
- Density ρ,

- flow effective porosity n,
- Matrix and fluid compressibility α, β

The default compressibilities of the matrix (α) and the fluid (β) are defined in the input dialog of the transient calculation.

Normally, the assigned porosity n (attribute PORO) is used for the calculation of the specific storage coefficient.

If the module INSTAT is used, the attribute SPEI should be used, since an internal conversion of node (PORO) to element values (SPEI) is not necessary.

If no storage coefficient and porosity are specified, the value 0.2 is used as default value.

6.6.3 Structure of the transient input file

For the transient calculation is a transient input file required, which includes the changing boundary conditions.

The easiest and safest way to create a transient input file is using transient structures.

The structure of the transient input file is described in chapter: "Data structure, transient input file". The transient attributes are described also in chapter: "Transient data types".

6.6.4 Input parameters

Via *Calculation → Transient flow* the following input window appears (2D model):
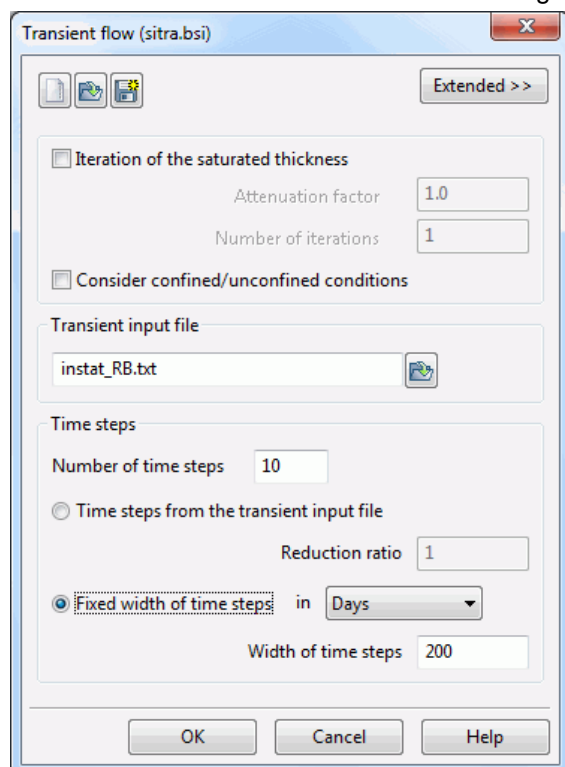
When calling the module, the program reads the batch file with the default name (here: sitra.bsi), the module SITRA is predefined as well. For the calculation with the module INSTAT the desired batch file has to be explicitly defined.

Iteration of thickness (2D model) / partially saturated computation (3D model)

This part is related to the iteration of the thickness in a 2D horizontal model or the partially saturated computation of the free surface in vertical or 3D models. The user must decide whether the iteration is performed. Furthermore, the maximum number of iterations and the attenuation factor must be defined. If a calculation has already taken place, it is possible to continue the iteration.

For 2D horizontal models, the thickness is only updated in a transient flow calculation and not changed itera-tively. Therefore, the number of iterations is automatically set to "1" and the checkbox is disabled. In an in-verse model calculation, the number of iterations is generally set to "1".

Attenuation factor

In order to avoid oscillations in the iteration, the change of the thickness of one iteration to the next is provided with an attenuation factor 0 <w <1. For 2D horizontal models, the attenuation factor is set to "1", the selection box does not appear.

Considering confined/unconfined conditions

This checkbox appears **only** in a 2D horizontal model with a thickness limited through the attributes OBER or UNDU. If there are confined conditions, they can be taken into account in the calculation of the storage coeffi-cient by checking the checkbox. Otherwise, the storage coefficient is determined for unconfined conditions (see chapter: "Computation of the storage coefficient").

Transient input file

Selecting the file with the transient data (file selection box)

Time steps

Number of time steps:

Input of the number of time steps to be calculated

Definition of time steps

When entering the parameters for the transient calculation, the dates at which the calculation is done can be affected in different ways.

Only dates from the transient input file are used. Here, a reduction ratio can be specified.

Alternatively, you can enter a fixed time step size (e.g. every 2 days) regardless of the transient input file. If these fixed time steps match with those of the input file, then the appropriate boundary conditions are set. Dates of the transient input file, which do not match the time steps are ignored with a warning issued.

Example:

Number of time steps: 10

Fixed width of time steps n = 200 days

This is an transient flow calculation carried out on 10 * 200 = 2000 days.

The buttons at the top of the input window allow resetting the input parameters ( ), opening an existing batch file ( ) or saving the current batch file under another name ( ). The selection "Extended" leads to further settings.

6.6.5 Extended

By activating the button "Extended" more parameters can be defined for *Output* and *Flow*.

6.6.5.1 Output

In the following window the input parameters for the output are defined:
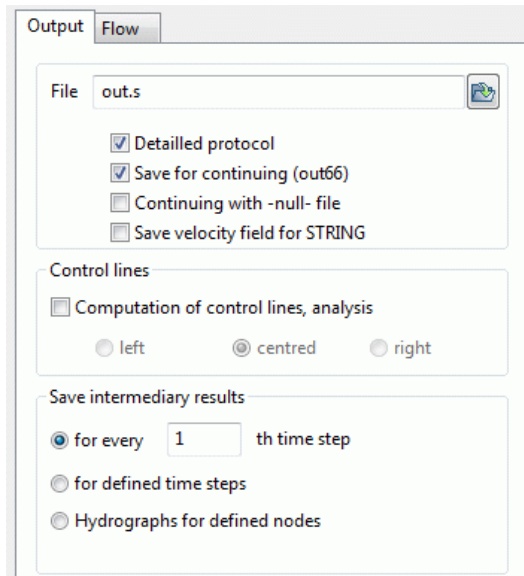
Output

The name of the output file is entered, default is the name *out*.s

Checking the first box enables the output of a *detailed protocol*.

Checking the third box enables the output of the file *out66* in **detailed form**. If this file is stored, copied and renamed in *null*, it is possible to continue the iteration with the previously calculated data (2nd check box).

Save velocity field for STRING

In a transient flow model it is possible to store the calculated velocity field for each time step to visualize it with the program STRING. The data is written into the background files and can be read directly with STRING.

A detailed explanation of the software STRING is found in the chapter: "Visualization of velocity distributions (Software STRING)" (p. 304).

If the checkbox is not checked, only the velocity field of the last time step is saved. This can also be visualized with STRING.

Control lines

(only if the attribute KONT is defined in the model file)

After checking the box the computation of the control lines is done. For each segment of the control line, a velocity is required. The velocity can be determined as the mean value over the two adjacent elements. Otherwise, the velocity of the left or the right element is chosen.

Saving of intermediary results

It is possible to store the intermediate results only at certain time steps so that the background files for large models would not be unnecessarily large.

Either every nth time step can be saved or max. 50 fixed dates can be specified in a separate window.

The intermediary results are necessary to plot contour lines for special dates or to display hydrographs (final state is always possible).

Alternatively to the selection of specific dates hydrographs of the potential values and concentrations / temperatures at all calculated intermediate dates can be saved for single nodes.

6.6.5.2   Flow

In the following window the input parameters for the flow are defined:

Calculation method

The choice of the calculation modules will remain until the module SITRA is adapted as described above. Until then, for a horizontal model with a partially 3D model and for the stream- and path line calculation the module INSTAT has to be selected (the modul XTRA will be available in the near future).

Equation solver and termination criteria

The equation solver and the termination criteria have been described in the calculation of steady-state flow.

Storage coefficient

The following parameters can be changed here:

The specific storage coefficient in horizontal models (confined aquifer) or the fluid and matrix compressibility in 3D models (saturated/unsaturated calculation).

Calculation of the storage term

The storage term calculation is based on the van Genuchten parameters.

Boundary conditions at intermediate points in time

If it is calculated with a fixed time step size or smaller time steps of the transient input file, it is possible for the intermediate time steps, which are not regulated by the transient input file to interpolate the transient boundary conditions! In principle he following is assumed:

- ■ Transient terrain settings are always linear interpolated!

- ■ Is a point of time listed in the transient input file (without data as well), the boundary conditions (except BERG) from that date are not interpolated, i.e. only the data that are listed below are changed. All other conditions remain those of the previous time step! The following example explains this:

Example 1:

The model file contains: POTE   10.0 1,….

The transient input file contains:

```
ZEITEINHEIT ZEIT TAG
```

```
ZEIT (first date)
2.0
POTE
     1      12.0
```

In case a constant time step of 1 day is defined the fixed potential heads are:

a) **with** interpolation of the boundary conditions:
10.0 for the 0th time step (not interpreted)
11.0 for the 1st time step (i.e. after 1 day)
12.0 for the 2nd time step (i.e. after 2 days)

b) **without** interpolation of the boundary conditions:
10.0 for the 0th time step (not interpreted)
10.0 for the 1st time step (i.e. after 1 day)
12.0 for the 2nd time step (i.e. after 2 days)

Example 2:
The model file contains: POTE   10.0 1,…
The transient input file contains:

```
ZEITEINHEIT ZEIT TAG
ZEIT (first time step)
1.0 (without data)
ZEIT (2nd time step)
2.0
POTE
     1      12.0
```

So, the boundary conditions for node 1 do not change on the first day. The potential head is fixed at 10.0 m on the first day. A requested change must be defined explicitly! The fixed potential head for node 1 is:

■   for the case in which a constant time step of 1 day is defined or the time steps defined in the transient input file are used with and without interpolation of boundary conditions:

10.0 for the 0th time step (not interpreted)
10.0 for the 1st time step (i.e. after 1 day)
12.0 for the 2nd time step (i.e. after 2 days)

■   For the case in which a *constant time step of 0.5 day* is defined or the time steps defined in the transient input file are *multiplied by 0.5*:
a) **with** interpolation of the boundary conditions:
10.0 for the 0th time step (not interpreted)
10.0 for the 1st time step (i.e. after 0.5 days)
10.0 for the 2nd time step (i.e. after 1 day)
11.0 for the 3rd time step (i.e. after 1.5 days)
12.0 for the 4th time step (i.e. after 2 days)

b) **without** interpolation of the boundary conditions:
10.0 for the 0th time step (not interpreted)
10.0 for the 1st time step (i.e. after 0.5 days)
10.0 for the 2nd time step (i.e. after 1 day)
10.0 for the 3rd time step (i.e. after 1.5 days)
12.0 for the 4th time step (i.e. after 2 days)

Remark: Volume boundary conditions are always handled explicitly:

For the calculation of the state at the point in time t_{n+1} the boundary conditions of the point in time t_n are used.

Initial conditions

Here is defined with which initial potential heads the transient flow computation is started.

■ Initialization with initial potentials (EICH): The in the steady state flow computation calculated resulting potentials are used as the initial potential. This requires that the resulting potentials are assigned to the attribute EICH via the menu item *Attributes → Import data/computation results → Potentials*.

■ The transient calculation can be continued, if the corresponding null-file from the previous calculation exists (warm start). If the user wants to continue the actual calculation, the program has to create the corresponding file called out66. This is specified in the choice of output parameters (Save for continuing (out66)).

■ No initial potentials: The iteration is started with initial potentials = 0.0.

6.6.6 Monitoring the time progress

After starting the calculation the log window of the calculation run appears, the SPRING Process Launcher.
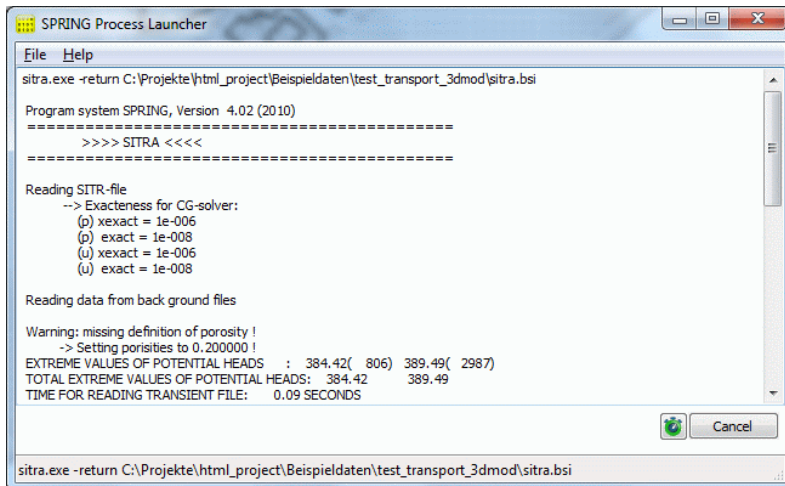
During a transient calculation, it is possible to track the time progress of the calculated results live by starting the SPRING Process Observer (Pressing the button).

The following window appears:
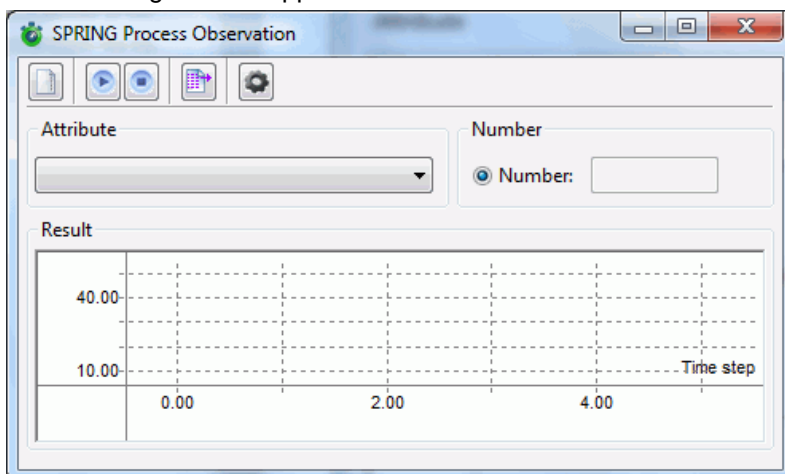
The program can be started directly from the command line with the command *obs*.

In that case is at first the directory to be chosen (pressing the button: ( ), in which a current calculation should be observed. With several simultaneously running calculations you can hereby change to a different working directory, in which a hydrograph should be monitored.

First, over the fields *Attribute* and *Number* the data type and the number of the node to be observed (or element) can be selected. The process is started by pressing the 2nd button:

The hydrograph is shown successively:
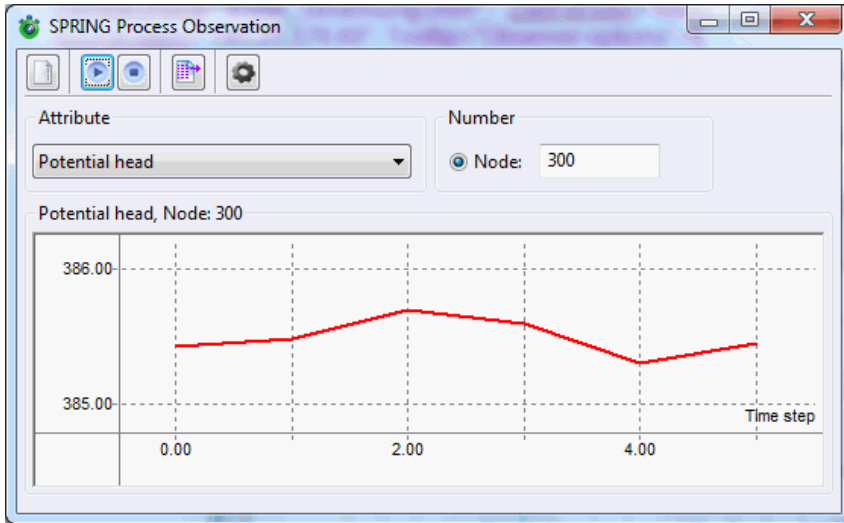
The ongoing process can be stopped by pressing the 3rd button: . After that, an attribute and a node or element can be selected and observed again.

By pressing the 4th button ( ) the currently observed hydrograph can be saved in a csv-file.

By pressing the button the properties for saving the hydrograph can be modified. The following input window appears:
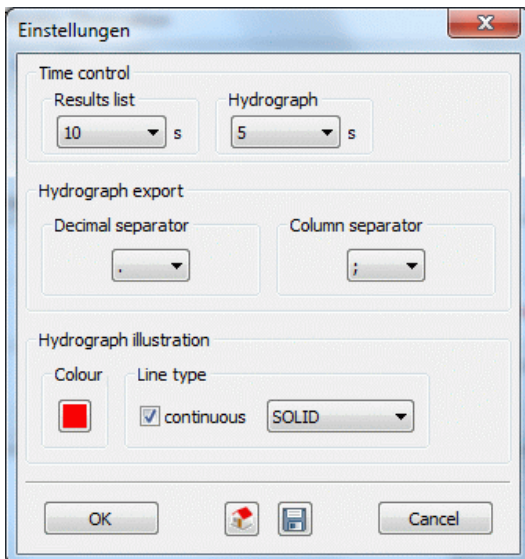
Time control

In large models the computation of a time step can take some time.

The specification of seconds defines, at what time the process shall review the present progress of time steps (*hydrograph*) and the change in the results (*result list*) in the background files. The review of the results is useful primarily for data types, which "start" only at a particular event (i.e. attribute SICK).

Hydrograph Export

The user can select which columns and decimal separators shall be used in the time series file.

Hydrograph illustration

The colour and the line type of the hydrograph illustration can be chosen freely.

6.6.7 Visualization of velocity distributions (Software STRING)

For the visualization of velocity distributions of flow simulations, various methods have been established. These include the presentation of information by means of small arrows, color-coded stream and path lines and Schlieren. All these methods have their pros and cons. Above all, without the knowledge of the observed data they are difficult to interpret. For this reason, the software STRING was developed in a cooperation between the delta h engineering company ltd. and the Fraunhofer Institute for Technical and Industrial Mathematics (ITWM) in Kaiserslautern. STRING provides intuitive access to the results of flow simulations. The program allows you to visualize steady state and transient flows in two spatial dimensions as a video. The relevant direction and velocity information are displayed by the moving flow streamlets. A short line segment of a streamline is called streamlet. The moving direction of a streamlet is the direction of the flow and the length correlates with the velocity.
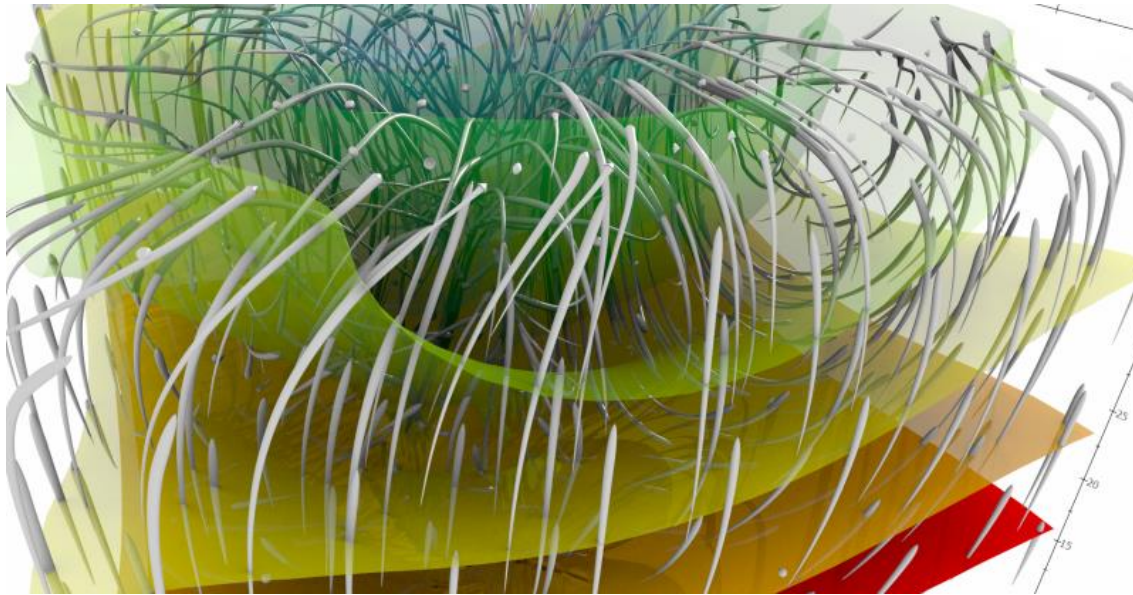
Fig. 192: Example of Visualization

The software allows the import of flow data from SPRING and storing the calculated images for a single video. These images can also be provided with a transparent background, so you can show in a subsequent step, any additional data in the video. To facilitate this, a world file is stored together with the single picture that allows the georeferencing of the video in any geographic information system (GIS). The generation of a video requires from SPRING information of geometry and of the velocities of flow within this area. For the initial distribution of the streamlets a proven method is used, which is applied within the Finite Pointset Method (FPM) and has long been developed on ITWM. The equations of motion of the streamlets are solved using an adaptive Runge-Kutta method of higher order. The error tolerance is adjustable. For interpolating the velocities from the data calculated in SPRING the method of Moving Least Squares is used. It also takes into account special cases related to complicated geometries.

Within the animation it can happen, that very dense accumulations of streamlets are formed in depressions of the flow area or that the streamlets flow away from all sources. In these areas no assessment of the prevailing flow is possible. To avoid this, a management of the local distribution of the streamlets is necessary.

The ITWM uses years of experience in the area of the FPM to solve such problems. Various algorithms from this environment were implemented in STRING.

The program STRING is controllable by a graphical user interface and the output of each image of the video can be adjusted freely. Parameters are for example given by the resolution and the running time of the finished video. But the appearance of each streamlet can be customized in great detail.

In this way, clear impressions of the flow direction, course, speed and temporal evolution of the water flow can be obtained. In combination with SPRING 5, STRING unfolds its full potential.

6.6.8 Results of the intermediate points of time of the transient flow calculation

The following intermediate results are delivered from the finite element calculation (partly on request) and saved in the output file "out.s":

- Potential heads at each node (detailed protocol selected)
- Saturation at each node (saturated/unsaturated flow and detailed protocol selected),
- Field velocity at each element (saved in the background files only when detailed protocol is selected),
- Mass balance with the total of inflow and outflow rates. It is compared to the total of the mass flow rates. The total of the input and the calculated mass flow rates must be close to zero. Additionally, the inflow rates (positive sign) are compared to the outflow rates (negative sign). The areas with BILK and BILE are also balanced.
- Mass flow rates at control lines, if defined in the model file.

6.6.9 Results of the transient flow calculation

At the end of a transient calculation the following results (partly on request) are delivered:

- Potential heads at each node
- Peak values of the calculated data (e.g. maximum potential heads) and the corresponding time steps

   They are imported by *Attributes → Import model data/computation results...*:
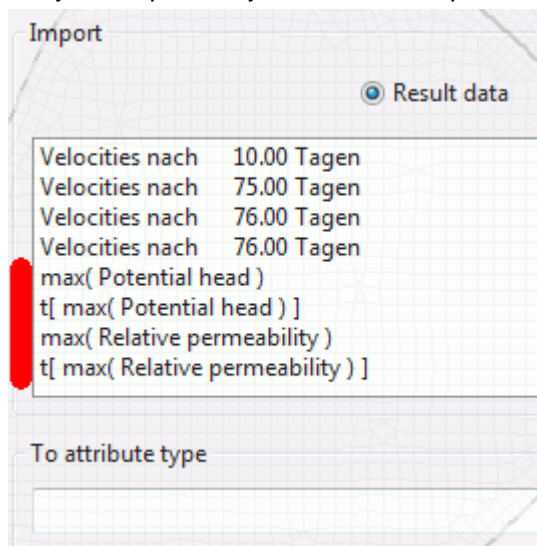
- Saturation at each node (saturated/unsaturated flow)
- Iterated thickness at each node (horizontal model, calculation including an iteration of the thickness)
- Field velocity at each element
- Velocity field for the last time step for visualization with STRING
- Velocity at each fracture element (if existing)
- Distance between ground level and ground water level at each node (if the input file contains ground levels GELA
- Mass flow rates (inflow and outflow rates) at each node with a boundary condition for the potential heads (POTE)
- Leakage rates at nodes and elements, if the input file contains leakage coefficients (LERA, LEKN, LEEL) and receiving water course levels (VORF
- Mass balance with the total of inflow and outflow rates. It is compared to the total of the mass flow rates (see before). The total of the input and the calculated mass flow rates must be approximately zero. Additionally, the inflow rates (positive sign) are compared to the outflow rates (negative sign). The areas with BILK and BILE are balanced, too.
- Mass flow rates at control lines, if defined in the model file.

6.6.10   Batch file transient flow

It is possible to execute the *transient flow* directly without using the dialog box. The default file name depends on the calculation module (*instat.bin or sitra.bsi*). To use another batch file, type *instat/sitra file name* (the extensions ".bin" or ".bsi" are automatically attached). Batch files can be created, viewed and modified with the help of any text editor

 detailed batch file can be viewed in the online help. Similarly, there is a batch file of the transient flow for further processing provided.

6.7 Mass transport

Call via: *SPRING →Calculation → Mass transport computation*

The spread of soluble substances (mass transport) in a porous medium is influenced by the processes of advection, diffusion and dispersion. Dissolved substances are generally also the process of adsorption, and chemical and biological degradation reactions (production and decay). Substances which are transported exclusively in the pores of the porous medium and stored (without adsorption, and chemical or biological degradation) there, are referred to as an ideal tracer.
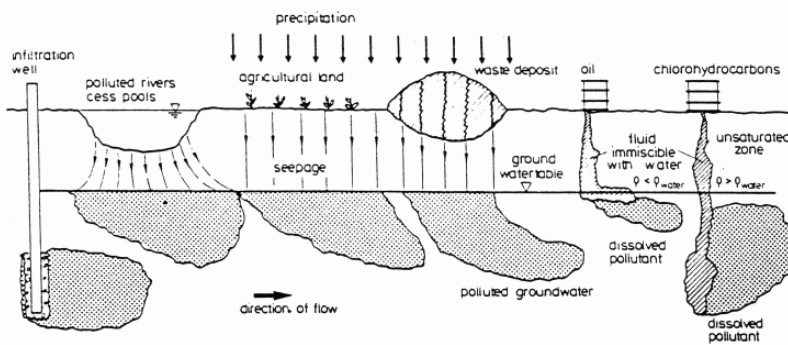
Fig. 193: Possible modes of substance flow into the groundwater

Each transport model is based on a flow model, so that both the parameters describing the material behaviour in the groundwater and the information for the flow modelling are required.

6.7.1 Components of the mass transport

- Advection
- Diffusion
- Hydromechanical dispersion
- Adsorption
- Degradation and production
- Radioactive or biological decay (half-life calculation)

At minimum the definition of one of these attributes KONZ, 1KON or AKON is required for the mass transport calculation.

6.7.1.1 Advection

Advection describes the passive movement of a substance (or Heat) with the moving fluid. The purely advective transport leads to a shift of material particles with the field velocity v of the groundwater in the direction of the flow vector.
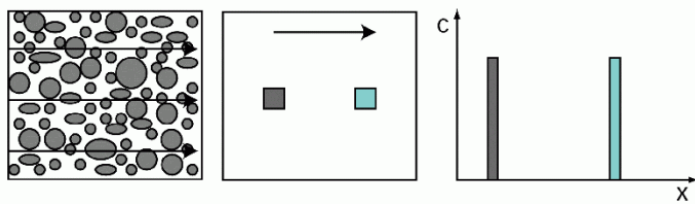
Fig. 194: Purely advective mass transport

The equation for the advective mass flux is:

$$\mathbf{j_{adv}} = \mathbf{vc}$$

The field velocity v is the quotient of the Darcy velocity v_f and the porosity n:

$$\mathbf{v} = \mathbf{v_f} \big/ \mathbf{n}$$

The concentration of the dissolved substance is **c.**

6.7.1.2 Diffusion

The molecular diffusion is a process that occurs independently of the groundwater movement. It is based on the thermal motion of the molecules, the Brownian movement, and leads to a concentration compensation. Only when the concentration of a substance is equal at all points in the groundwater, the diffusion stops.
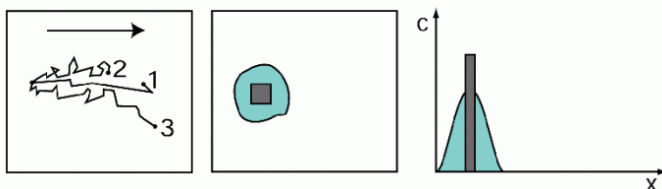
Fig. 195: Diffuse mass transport

The mass flux due to molecular diffusion j_{diff} is described by the 1st Fick's law:

$$\mathbf{j_{diff}} = -\mathbf{d_m} \nabla \mathbf{c}$$

with
d_m = molecular diffusion coefficient [m²/s].

Here, the mass flux due to the Brownian movement is proportional to the concentration gradient. Similar to Darcy´s law the negative sign results from the fact that the mass flows from high to low concentrations against the concentration gradient ∇**c.**

6.7.1.3 Hydromechanical dispersion

The hydromechanical dispersion describes the distribution and the mixing of the same materials (or heat) in the moving pore water. This process is due to the different flow rates in the pore, the pore size distribution and the path which each particle covers.
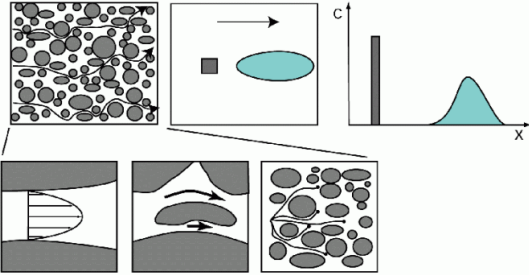
Fig. 196: Dispersive mass transport

The hydromechanical dispersion provides a further flow-dependent component to the mass or energy flow besides the advective component. The mass or energy flow is described with the 1st Fick's law, similar to the diffusion:

$$\mathbf{j_{disp}} = -\mathbf{D_d}\nabla\mathbf{c}$$

With the symmetrical dispersion tensor D_d [m²/s]:

$$\mathbf{D_d} = \begin{vmatrix} \mathbf{D_{11}} & \mathbf{D_{12}} & \mathbf{D_{13}} \\ \mathbf{D_{21}} & \mathbf{D_{22}} & \mathbf{D_{23}} \\ \mathbf{D_{31}} & \mathbf{D_{32}} & \mathbf{D_{33}} \end{vmatrix}$$

Its coefficients in the three-dimensional analysis are defined as follows by Frind (from: FRIND, E.O.; BURNETT, R. D. (1987): Simulation of contaminant transport in three dimensions, 2. Dimensionality effects. Water Resour. Res., 23(4), pp. 695-705):

$$\mathbf{D_{11}} = \alpha_L \frac{v_1^2}{v} + \alpha_{TH} \frac{v_2^2}{v} + \alpha_{TV} \frac{v_3^2}{v}$$

$$\mathbf{D_{12}} = \mathbf{D_{21}} = (\alpha_L - \alpha_{TH})\frac{v_1 v_2}{v}$$

$$\mathbf{D_{22}} = \alpha_{TH} \frac{v_1^2}{v} + \alpha_L \frac{v_2^2}{v} + \alpha_{TV} \frac{v_3^2}{v}$$

$$\mathbf{D_{23}} = \mathbf{D_{32}} = (\alpha_L - \alpha_{TH})\frac{v_2 v_3}{v}$$

$$\mathbf{D_{33}} = \alpha_{TV} \frac{v_1^2}{v} + \alpha_{TV} \frac{v_2^2}{v} + \alpha_L \frac{v_3^2}{v}$$

$$\mathbf{D_{13}} = \mathbf{D_{31}} = (\alpha_L - \alpha_{TH})\frac{v_1 v_3}{v}$$

with:

$\mathbf{v}$ = (v_1, v_2, v_3) = velocities in the carthesian coordinate system [m/s],

v = Amount of the field velocity v [m/s],

α_L = longitudinal dispersivity [m],

α_{TH} = transversal-horizontal dispersivity [m],

α_{TV} = transversal-vertical dispersivity [m].

In the two-dimensional, the coefficients of the dispersion tensor D_d are defined by Scheidegger:

$$\mathbf{D_d} = \begin{vmatrix} \mathbf{D_{11}} & \mathbf{D_{12}} \\ \mathbf{D_{21}} & \mathbf{D_{22}} \end{vmatrix}$$

$$\mathbf{D_{11}} = \alpha_L \frac{v_1^2}{v} + \alpha_T \frac{v_2^2}{v}$$

$$\mathbf{D_{12}} = \mathbf{D_{21}} = (\alpha_L - \alpha_T) \frac{v_1 v_2}{v}$$

$$\mathbf{D_{22}} = \alpha_T \frac{v_1^2}{v} + \alpha_L \frac{v_2^2}{v}$$

with:

α_T = transversal dispersivity [m],

The dispersivity is a characteristic length that can be measured in the laboratory or in the field. It is a geometric measure of the permeability and storage heterogeneity of the aquifer. The magnitude of the dispersion coefficient α is dependent on the scale as well as of the porosity, grain shape and grain size.
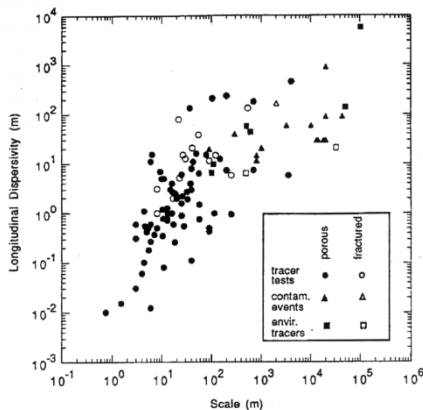
Fig. 197: Longitudinal dispersivity depending on the propagation length from different field trials (from: GELHAR, L. W.; WELTY, D.; REHFELDT, K. R. (1992): A critical review of data on field-scale dispersion in aquifers. Water Resour. Res., 28(7), pp. 1955-1974)

The hydrodynamic (total) dispersion results from the addition of the hydromechanical dispersion and the diffusion.

Substances which obey only the previously described so-called conservative transport processes are known as an **ideal tracer**.

In the following the so-called non-conservative mechanisms are described which cause a retention (retardation) and a reduction in the mass content in the groundwater.

6.7.1.4 Adsorption

Adsorption processes, which include its inverse (desorption) are generated due to the physical or chemical binding of tracer particles to the surface of the grains of the aquifer.

While the chemical bond to the matrix is usually irreversible, a physical bond is usually reversible, that is, depending on the relation between adsorption and transport velocity an equilibrium between adsorption and desorption appears.

Such equilibria can be described for constant temperature conditions through so-called adsorption isotherms with both linear (Henry isotherm) and non-linear character (Freundlich and Langmuir isotherms) and considered in the mass transport calculation.
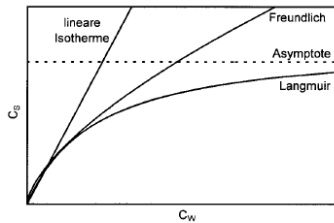
Fig. 198: Comparison of the different adsorption isotherms

The adsorption rate is described by a function $c_s = f(c_w)$:

with:

c_s = concentration of substances adsorbed to the matrix

c (or c_w) = concentration of substances dissolved in the fluid.

All three approaches (Henry, Freundlich and Langmuir) assume a constant fluid density $\rho_0 = \rho(c = 0)$.

6.7.1.4.1   Linear adsorption based on Henry

$$c_s = (\chi_1 \rho_0)c$$

with

χ_1 = Henry isotherm [m³(fluid)/kg(matrix)].
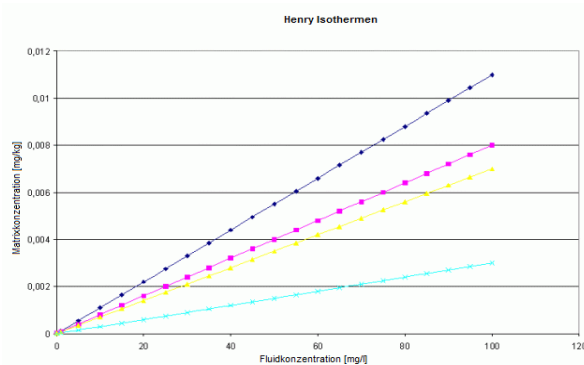
Fig. 199: Parameter variation for the adsorption based on Henry

6.7.1.4.2   Adsorption based on Freundlich

$$c_s = \chi_1(\rho_0 c)^{\left(\frac{1}{\chi_2}\right)}$$

with

χ_1 = 1. Freundlich isotherm [m³ (fluid)/kg (matrix)] and

χ_2 = 2. Freundlich isotherm [-].
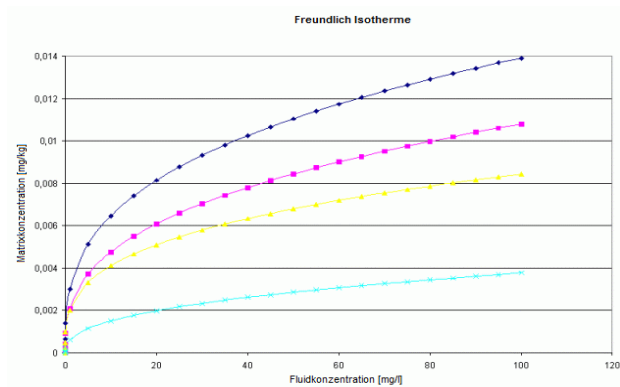
Fig. 200: Parameter variation for the adsorption based on Freundlich

6.7.1.4.3   Adsorption based on Langmuir

$$c_s = \frac{\chi_1(\rho_0 c)}{1 + \chi_2(\rho_0 c)}$$

with

χ_1 = 1. Langmuir isotherm [m³ (fluid)/kg (matrix)] and

χ_2 = 2. Langmuir isotherm [-].
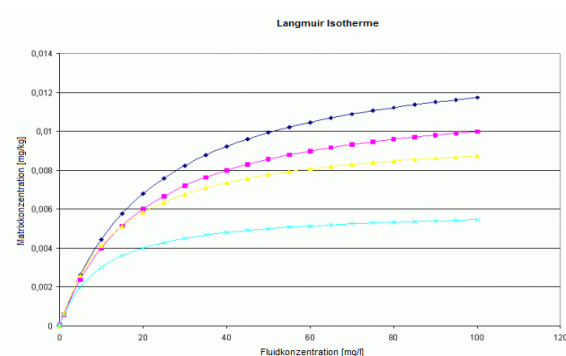
Fig. 201: Parameter variation for the adsorption based on Langmuir

The type of adsorption is set with the corresponding isotherms χ_1 and χ_2 during the input dialog of the mass transport calculation. Default is a transport calculation without adsorption.

6.7.1.5 Degradation and production

Decay and production of substances in the groundwater can be both of chemical and biological nature.

In the mass transport calculation decay or production processes can be taken into account. In theory, a distinction is made between production and decay rates of the solution in the fluid and the adsorbate in the matrix. In nature, these processes are not clearly separated from each other. Because of this, the mass transport calculation is done without taking the processes within the matrix into account. Instead, all production, degradation or decay processes are considered only in the fluid.

For the **degradation and production in the fluid**, different Ansatz functions are available:

■   **No** production and decay processes are considered (default).
■   **Constant** behaviour of production and degradation rates:
    f(c) = c = const.
    The parameter c corresponds to the intercept of the function, example: c = 1.0.
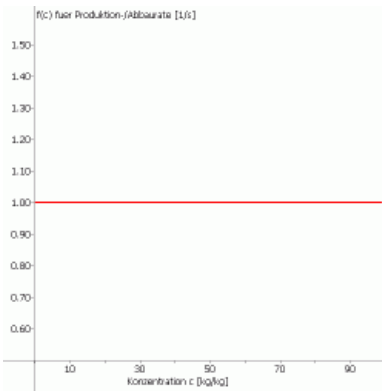
■ **Linear** behaviour of production and degradation rates:

f(c) = Intercept + gradient * c

In the example is the intercept = 1.0, and the gradient = 1.0.
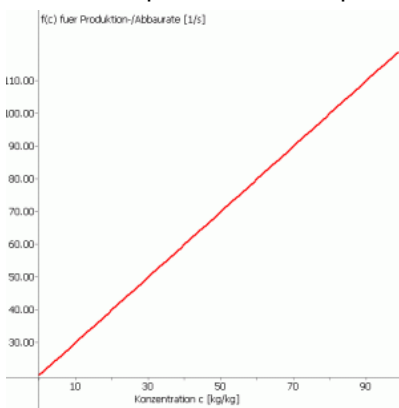
■ A **step function** can be entered for the decay and production rate (starting with f(c = 0) = 0), which is limited by a maximum rate (maximum) when reaching a certain concentration (lower limit). When reaching an entered a maximum concentration (upper limit) the decay or the production rate is set back to 0:

f(c) = (c_{max}/$c_{untere\ Grenze}$) * c, für: 0 <= c < lower limit

f(c) = c_{max}, for: lower limit <= c < upper limit

f(c) = 0.0, for c >= upper limit

In the example the lower limit is = 10.0, the upper limit is = 20.0 and the maximum is 10. 0.
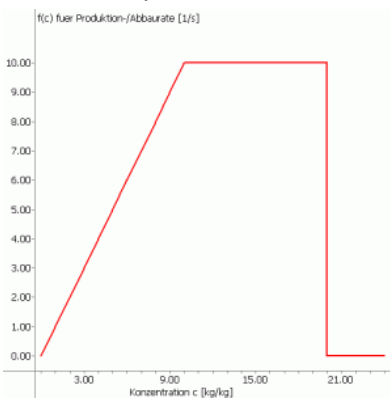
■ If a zoning of the material parameters was made (attribute Z-AP), a 4th Ansatz function for the decay and production rate is additionally available. The **tent function** drops after reaching a maximum until the upper limit is reached. After that the decay or the production rate is set back to 0:

f(c) = (c_{max}/$c_{untere\ Grenze}$) * c, for: 0 <= c < lower limit

f(c) = - (c_{max}/(c_{og} – c_{ug})), for: lower limit <= c < upper limit

f(c) = 0.0, für c >= upper limit

In the example the lower limit is = 20.0, the upper limit = 30.0 and the maximum = 200.0.
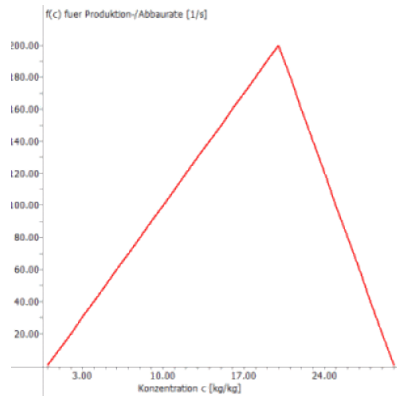
The individual parameters intercept, maximum, lower and upper limit of the function are defined in the extended definitions – decay and production.

6.7.1.6   Radioactive or biological decay (half-life calculation)

The radioactive or biological decay (e.g. by bacteria) is usually an exponential decay rate, which is described by the half-life.

The half-life $T_{1/2}$ is the time in which an exponentially decreasing value has itself divided in half.

The **half-life calculation** is based on an exponential shape function.

$$c(t) = c_0 * \left(\frac{1}{2}\right)^{\frac{t}{T_{1/2}}}$$

With:

c_0 = initial concentration

In the example the half-life is 5000 seconds.
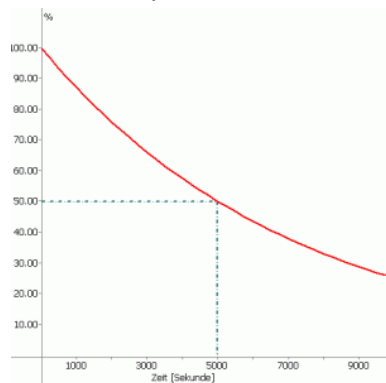
The parameter t of the function is defined in the extended definitions – half-life (decay and production).

6.7.2 Equation of the mass transport

For a saturated aquifer with **constant density** and steady state flow the mass flux **j** for the mass transport can be mathematically formulated as follows:

$$j = j_{adv} + j_{diff} + j_{disp}$$

Since the transport of the ingredient takes only place in the effective pore volume (n) of the aquifer, it must be additionally introduced for a volume based consideration.

$$\mathbf{j_v = nj}$$

By substituting the components we obtain the equation:

$$\mathbf{j_v = n(vc - d_m\nabla c - D_d\nabla c)}$$

The proportion of the molecular diffusion and hydromechanical dispersion can be summarized to the hydrodynamic dispersion:

$$\mathbf{D = d_m I + D_d}$$

With the standard matrix **I**:

$$\mathbf{I = \begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{vmatrix}}$$

Results the equation for the steady state mass flow:

$$\mathbf{j_v = n(vc - (d_m I + D_d)\nabla c)}$$

The mass balance allows the consideration of the change in concentration over time, referred to in a control volume in- and outflowing mass quantities and taking into account sources and sinks (**σ**), that means, points at which the system is supplied with substances or at which substances are removed from the system.

$$\mathbf{n\frac{\partial c}{\partial t} + \nabla n(vc - (d_m I + D_d)\nabla c) - \sigma = 0}$$

The term **σ** represents all mass in- and outputs, and can be broken down as follows:

$$\mathbf{\sigma = qc^* + R_i} \text{ with:}$$

qc* = volume-related addition or removal of water with the concentration c*
R_i = Share of all non-conservative mass transport processes such as sorption, chemical or biological degradation reactions

Using the product rule for the advection term:

$$\mathbf{\nabla(nvc) = nv\nabla c + c\nabla nv} \text{ ,}$$

of the continuity condition

$$\mathbf{\nabla(nv) = q} \text{ and the summary } \mathbf{D = d_m I + D_d}$$

results the transient transport equation for ideal tracer at a constant density of the aquifer to:

$$\mathbf{n}\frac{\partial \mathbf{c}}{\partial \mathbf{t}} + \mathbf{nv}\nabla\mathbf{c} - \nabla(\mathbf{nD}\nabla\mathbf{c}) = \mathbf{qc}*$$

[Example: q corresponds e.g. to the attribute node removal/addition KNOT or to a reaction rate, unit [m³/TU] and c corresponds to the attribute KONZ]*

If saturated/unsaturated conditions exist in the aquifer the saturation **S$_r$** must be taken into account in the transport equation:

$$\mathbf{nS_r}\frac{\partial \mathbf{c}}{\partial \mathbf{t}} + \mathbf{nS_r v}\nabla\mathbf{c} - \nabla(\mathbf{nS_r D}\nabla\mathbf{c}) = \mathbf{qc}*$$

The unit of this equation results in [kg mass / (kg solution *s)] or [kg mass / (m³ solution *s)], dependent on the unit of the concentration boundary condition.

6.7.2.1 Non-conservative processes

The following figure gives an overview of the non-conservative mass transfer processes:
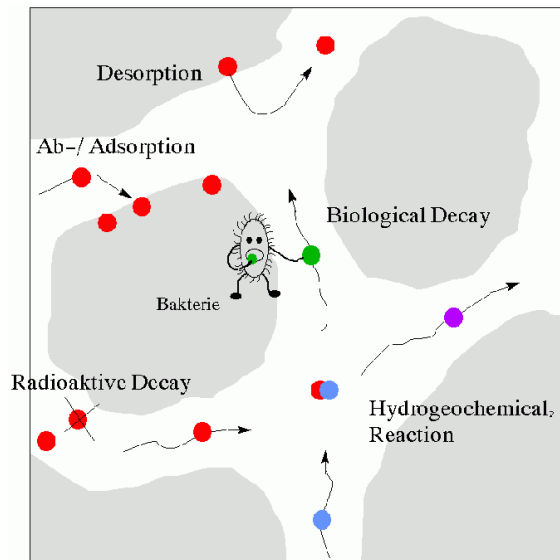
Fig. 202: Non-conservative mass transport processes

When including the **adsorption processes** in the transport equation is to be noted that the total mass of the considered ingredient in a reference volume not only consists of the dissolved and transported amount of material (nc), but also of the amount of adsorbed substances $(c_s(1-n)\rho_s)$. The concentration of the adsorbed substance (c_s) is referred to the solids fraction (1-n), where the density of the matrix (ρ_s) must additionally be taken into account.

In this case, the term

$$\mathbf{R_1} = \frac{\partial \mathbf{c_s}}{\partial \mathbf{t}}((1-n)\rho_s)$$

Has to be taken into the transport equation with:

c_s = adsorbed concentration in the matrix [kg/kg]

ρ_s = Density of the matrix [kg/m³]

The adsorption rate is described by a function c_s = f (c), either after the adsorption isotherm based on Henry, Freundlichor Langmuir.

By the integration of **production and decay** is important to note that this theoretically occurs both in the fluid and in the matrix. The corresponding term R therefore consists of two parts with different decay and production functions for the solution in the fluid f(c) and the adsorbed materials in the matrix $f_s(c_s)$:

$$R_2 = nS_r f(c) + (1-n)f_s(c_s)$$

During the mass transfer calculation in SPRING production and decay processes will be considered only in the fluid, because separation of these processes is impossible in nature.

Due to the irreversibility of these processes here, there is no storing!

Instead of production and decay it is possible to realize a **half-life calculation**. The corresponding term is then:

$$R_3 = nS_r c(t)$$

with:

$$c(t) = c_0 * \left(\frac{1}{2}\right)^{\frac{t}{T_{1/2}}}$$

6.7.3 Boundary and initial conditions

The differential equations described above are supplemented by the boundary conditions:

- Boundary condition 1st order: Specification of fixed concentrations (attribute 1KON):

  $$c = c*$$

- Boundary condition 2nd order: Specification of a diffusive-dispersive mass flux on the boundary. The almost exclusive practical application, however, is to make the edge impermeable for diffusion and dispersion, so that:

  $$-(D\nabla c)N = j = 0$$

  (N denotes the outward facing normal to the boundary of the model).

The system of equations can only be solved if at least at one node a boundary condition of the 1st order is defined.

For a transient mass transport equation initial conditions for the start time have to be defined for the entire model domain.

- Initial condition: Initial concentrations for the start time t_0 of the transient calculation (attribute AKON):

  $$c(t_0) = c_0^*$$

6.7.4 Input parameters

Via *Calculation → Mass transport computation* the following input window appears:
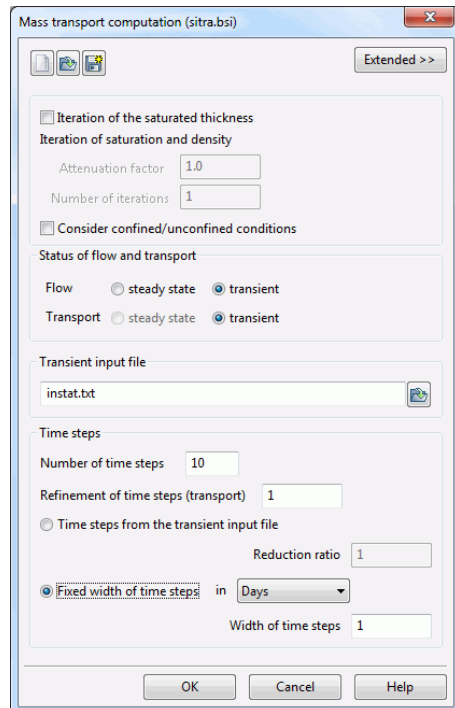
When calling the module, the program reads the batch file with the default name (here: sitra.bsi).

Iteration of thickness (2D) / partially saturated computation (3D)

This part is related to the iteration of the thickness in a 2D horizontal model or the partially saturated computation of the free surface in vertical or 3D models. The user must decide whether the iteration is performed. Furthermore, the maximum number of iterations and the attenuation factor must be defined. If a calculation has already taken place, it is possible to continue the iteration.

In a transient calculation of 2D horizontal models, the number of iterations and the attenuation factor are set to "1", the selection fields are disabled. In an inverse model calculation, the number of iterations is generally set to "1", the attenuation factor can be entered (except for 2-D horizontal models).

Considering confined/unconfined conditions

This checkbox appears **only** in a 2D horizontal model with a thickness limited through the attributes OBER or UNDU. If there are confined conditions, they can be taken into account in the calculation of the storage coefficient by checking the checkbox. Otherwise, the storage coefficient is determined for unconfined conditions (see chapter: "Computation of the storage coefficient").

Status of flow and transport

Here the choice is made whether flow and transport are calculated steady state or transient. If the mass transfer is steady state, a calculation with adsorption or production and decay is not possible.

Transient input file

Selecting the file with the transient data (file selection box)

Time steps

An explanation of the defined time steps is already given in chapter „ Compilation of different model types – creating a transport model" .

The buttons at the top of the input window allow resetting the input parameters ( ), opening an existing batch file ( ) or saving the current batch file under another name ( ).
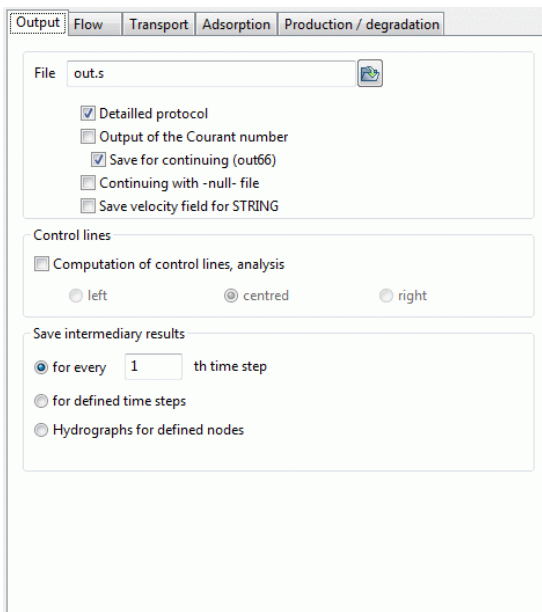
6.7.5 Extended

By activating the button "Extended" more parameters can be defined for:

■ Output
■ Flow
■ Transport
■ Adsorption
■ Degradation/Production
■ Half-life calculation

6.7.5.1   Output

After calling the dialog window "Output" the following input window appears:

Output of the Courant number

The Courant number is a measure of the "Stability criterion of the time discretization"

The other entries are already"Transient flow – Output".

6.7.5.2   Flow

The entries of notebook page "flow" are already described in chapter "Transient flow – Flow" .

6.7.5.3   Transport

After calling the dialog window "Transport" the following input window appears:

Equation solver

For solving the equation system there are an iterative PCG solver and a direct Cholesky solver. For meshes of about 500 nodes or more, you should always choose the iterative one, because it is much faster. The error due to the iterative solution is always much smaller (less than 3%) than the other errors (e.g. measurement error). This can be verified with the help of the mass balance printed to the output file. Further notes to the equation solvers and termination criteria, see "Steady State Flow - Extended Settings in the module SITRA" (p. 290).

Equal concentrations for GLEI

This determines whether at nodes with the attribute GLEI also the equal concentrations should be calculated.

For nodes at which the equal potential heads have been defined, in transport calculations can be selected if the concentrations at this node are to be equated as well or if this is not desired. Equal concentrations at equal potential heads are useful at perfect wells (potential heads are equated over the height).

If there are large open water areas (lakes, with the equal potential heads) in the transport model, then the equating of the associated concentrations are interpreted as follows:

The incoming concentrations at the inflow boundary of the lake are transported directly to the outflow boundary of the lake without delay. If the concentrations are not equated, a transport takes place inside the lake. Since the velocities calculated in the elements within the lake is close to 0, this approach will lead to large numerical problems.

Basically, a transport model with a large open water area in the range of an expected mass spreading is only calculable with the present calculation methods under simplifying assumptions. That is, a flow and transport model for an aquifer is not suitable for the calculation of flow in surface waters.

Boundary conditions at intermediate point in time

If the calculation is done with a fixed time step size or diminished time steps of the transient input file, it is possible at the intermediate time steps, which are not regulated by the transient input file to interpolate transient boundary conditions for the concentrations.

Initial conditions

Here is defined with which initial concentrations the transient transport calculation is started:

- No initial concentrations: The iteration is started with initial concentrations $c_0 = 0.0$.
- Using initial concentrations: The initial concentrations of the model file (attribute AKON) are used.
- The transient calculation can be continued, if the corresponding null-file from the previous calculation exists (warm start). If the user wants to continue the actual calculation, the program has to create the corresponding file called out66. This is specified in the choice of output parameters (Save for continuing (out66)).

Diffusion constant

Defining the molecular diffusion coefficient d_m in [m²/s].

Dispersivities

Defining transversal-horizontal (α_{TH}) and transversal-vertical (α_{TV}) dispersivities [m].
The scale refers to the longitudinal dispersivity, which is defined in the model file using the attribute DISP.

6.7.5.4 Adsorption

After calling the dialog window "Adsorption" the following input window appears:

Ansatz functions

Selection of the adsorption isotherms: It is distinguished by:

- Henry
- Freundlich
- Langmuir

The adsorption can be calculated zoned with different parameters for the respective zones (attribute Z-KD).
Each isotherms with their corresponding parameters are described in chapter "Adsorption".

Matrix parameters

Definition of the density of the matrix (soil parameter) in [kg/m³]

Zoning

If a zoning file already exists, it can be selected via the appearing file selection window. For a new zoning a preformatted file in *.txt format can be created by clicking the button 🗋. The format of the file is described in chapter "Structure of the zoning files" (data structure of the groundwater model).

Adsorption parameters

Depending on the chosen adsorption isotherm the required coefficients $\chi_{1,2}$ are entered here, unit [m³/kg].

Diagram

Based on the input parameters and the underlying isotherm a preview graph is created. This can be saved as an image (save diagram).

6.7.5.5 Degradation and production

After calling the dialog window "Degradation/production" the following input window appears:

Zoning

The decay and production can be calculated zoned with different parameters for the respective zones (attribute Z-AP).

If a zoning file already exists, it can be selected via the appearing file selection window. For a new zoning a preformatted file in *.txt format can be created by clicking the button 🗋. The format of the file is described in chapter "Structure of the zoning files" (data structure of the groundwater model).

Fluid

Here, the functions for decay and production are selected and the decay or production parameters of the fluid are given.

The Ansatz functions and their corresponding parameters are described in chapter: "Degradation and production".

Diagram

Based on the input parameters and the underlying Ansatz function a preview graph is created. This can be saved as an image (save diagram).

6.7.5.6 Half-life calculation

After calling the dialog window "Degradation/production" and if no zoning parameters are defined in the model file the following input window appears when selecting "half-life period":
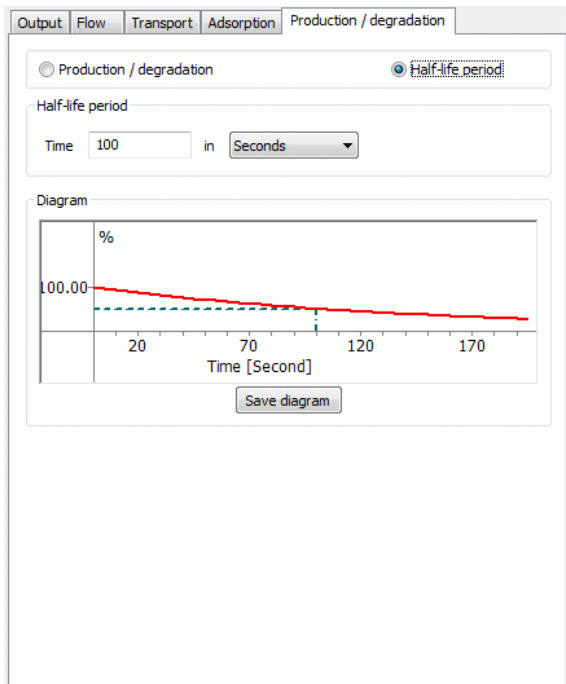
Half-life period

Input of the half-life period of the radioactive or biological decay

Diagram

Based on the input parameters and the underlying isotherm a preview graph is created. This can be saved as an image (save diagram).

6.7.6 Results of the intermediate points of time of the transient mass transport calculation

As intermediate results of the transient mass transport calculation the temporary concentration at all nodes are saved in the output file „out.s" additionally to the pure flow relevant data.

6.7.7 Results of the transient mass transport calculation

The calculation provides at the end of the transient mass transport the final concentrations of all nodes, additionally to the pure flow-relevant data.

6.7.8 Batch file transient mass transport

It is possible to execute the *mass transport* directly without using the dialog box. The default file name is *sitra.bsi*. To use another batch file, type *sitra file name* (the extension ".bsi" is automatically attached). Batch files can be created, viewed and modified with the help of any text editor

A detailed batch file can be viewed in the online help. Similarly, there is a batch file of the transient mass transport for further processing provided.

6.8 Density-dependent mass transport

Call via: *SPRING → Calculation → Mass transport (density-dependent)*

The density-dependent flow is only possible in vertical or 3D models!

In the density-dependent calculation the density of the ground water depends on the concentration of the solute:

$$\rho_w = \rho_w(c)$$

The density ρ has to be taken into account into the flow and transport equation to calculate a density-dependent flow. The full transport equation is:

$$nS_r\rho_w \frac{\partial c}{\partial t} + nS_r\rho_w v\nabla c - \nabla(nS_r\rho_w D\nabla c) = q\rho_w c * + R_i\rho_w$$

By including the density the unit of the density-dependent mass transport equation results in:

[kg mass / (m³ solution *s)]

6.8.1 Realization in SPRING

The calculation in SITRA is based on a linear function of the density ρ to the concentration c. The general equation for a concentration-dependent density ρ(c) is:

$$\rho(c) = \rho_0 + \alpha(c - c_0)$$

With:

ρ(c) = concentration dependent density of the solution [kg/m³]

c = concentration [kg mass / m³ solution]

c_0 = reference concentration [kg mass / m³ solution]

ρ_0 = density of the reference concentration c_0 in [kg/m³]: $\rho_0 = \rho(c = c_0)$

α = constant density gradient

These parameters are defined within the window of the Input parameters of the density-dependent mass transport calculation. The following has to be considered:

The calculation module uses internally the concentrations (c_{kg}) in the unit [kg mass/kg solution]. In practice, however, often a unit in [g/l] = [kg/m³] is usual.

The following relationship applies to the two sizes:

$$c_{kg}\left[\frac{kg(mass)}{kg(solution)}\right] = c\left[\frac{kg(mass)}{m^3(solution)}\right]/\rho(c)$$

ρ(c) = 1000 kg/m³ for water.

All inputs of the model data files (*.net, *.3d, additional files and transient input files) (KONZ, AKON and 1KON) are interpreted in a density-dependent calculation as inputs in the unit [kg/kg]. However, the results (= calculated concentrations) are converted during a density-dependent calculation in the unit [kg/m³] = [g/l], so the resulting concentrations and the input values are not directly comparable. This also applies to the plot generation!

Since the calculation module internally uses the pressure equation, some attributes have to be converted in other sizes:

■ Water course levels (VORF), initial potential heads (EICH) and fixed potential heads (POTE) of the model files and the transient input file are converted in pressure sizes. If the corresponding node has a concen-

tration c ≠ 0 (AKON), the starting values (EICH, VORF, POTE in the model file) can initially be corrected with the entered parameters via the menu item *Attributes - Computation - Density-corrected initial potentials*. For transient calculations is it possible, to correct the water levels with the density parameters during the time steps. For this purpose the button "Pressure" has to be activated in the input dialog of the density-dependent mass transport ("Boundary condition"). If the potential heights should not be corrected with the density during the transient calculation, they can be kept constant by activating the button "potential".

- All quantities of the model files and transient input file (affects KNOT, FLAE, RAND/Q/X, EFLA) are always converted from m³ to kg (solution) using the density ρ(c = 0). This should be considered especially when a quantity should be entered with an inflow concentration $c_{in} > 0$. In this case, the used density is wrong (for the conversion actually the density ρ(c_{in}) should be used). This uncertainty may be circumvented by converting the actual quantity (Q_{Start}) in the input value (Q_{korr}).

$Q_{korr} = Q_{Start} * ρ(c)/ρ_0$

with:

Q_{korr} [m³/TU] = by the density-parameters corrected quantity for KNOT, FLAE, RAND/Q/X, EFLA

Q_{Start} [m³/TU] = KNOT, FLAE, RAND/Q/X, EFLA at a node/element with given KONZ or AKON

$ρ_0$ [kg/m³] = Density of the reference concentration $c_0 = 0$

$ρ$ [kg/m³] = known/desired density ρ(c = c_{max}), c_{max} is usually the as KONZ or AKON entered maximum concentration

- The K-values defined in the model files are converted into permeabilities. In this conversion is always the density ρ(c = 0) used.

6.8.2 Input parameters

Call via: *Calculation → Mass transport (density-dependent)* appears the following input window:

When calling the module, the program reads the batch file with the default name (here: sitra.bsi).

Besides the already known input parameters „Status of Flow and Transport" and „Time steps" (see "Input parameters Mass transport"), in the density-dependent calculation the parameters for the change of density depending on the mass concentrations in the fluid are defined.

Iteration of saturation and density

The required number of iteration steps and the attenuation factor are determined.

Density parameters

Density ρ_0 of the reference concentration c_0 in [kg/m³]: $\rho_0 = \rho(c = c_0)$, default is $\rho_0 = 1000$ kg/m³

Density gradient α = constant, default is $\alpha = 0.0$

Reference concentration c_0 [kg/kg], default is $c_0 = 0$ kg/kg

The corresponding diagram can be represented and saved.

In general, the density $\rho_0(c_0 = 0)$ of the reference concentration is known as well as the desired density ρ ($c = C_{max}$), C_{max} is typically the maximum concentration entered as 1KON or AKON.

With this the required density gradient α can be calculated by changing the linear density function that is programmed in the program module SITRA:

$\rho(c) = \rho_0 + \alpha(c - c_0)$

to:

$\alpha = (\rho(c_{max}) - \rho_0) / (c_{max} - c_0)$, if $c = c_{max}$.

ρ_0 [kg/m³] = Density of the reference concentration $c_0 = 0$

ρ [kg/m³] = known/desired density $\rho(c = c_{max})$,

c_0 and c_{max} have to be entered in the unit kg/kg for the calculation of α .

Numerical example for SPRING:

Density of the saltwater $\rho(c) = 1021$ kg/m³

Density of the fresh water $\rho_0 = 997$ kg/m³

Salt concentration $c = 35$ kg/m³,

For the calculation in SPRING the salt concentration c [kg/m³] has to be converted from the unit [kg/m³] into the unit [kg/kg]:

$c_{kg} = c / 1000 = 0,035$ [kg/kg]

Substituting into the formula obtained:

$\alpha = (1021-997)/0,035 = 685,7$ [(kg/m³)/(kg/kg)]

The values are put in the dialog as follows:

Parameters for density-dependent calculation		
Density of the reference concentration	997	[kg / m³]
Density gradient	685.71	[(kg/m³)/(kg/kg)]
Reference concentration	0	[kg/kg]
	Diagram	

The corresponding density function looks like this:
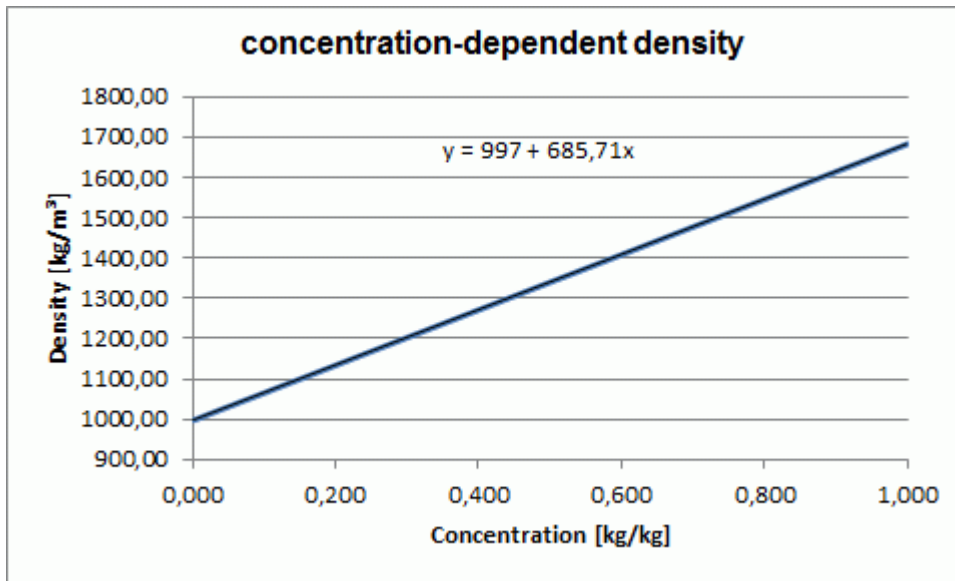
Fig. 203: Density depending on the salt concentration in [kg/kg]

These numbers can be found in the sitr file in the 5th row:

4th parameter of row 5: $\rho_0(c_0)$ = 997.0 (kg/m³)

5th parameter of row 5: c_0 = 0.0 (kg/kg)

6th parameter of row 5: α = 685.70 (kg/kg)/(kg/m³)

Specific density function for salt (sea water)

If the 5th and 6th parameter of the sitr file are set to -999.9, SITRA calculates with a specific density function for salt (T = 20 ° C):

$$\rho(c)= 998{,}5 *(1+0{,}000765 * c)= 998{,}5 + 0{,}7639 * c$$

with:

$\rho(c)$ = concentration-dependent density of the saltwater [kg/m³]

Density of the reference concentration (fresh water): $\rho_0 = \rho(T=20°, c_0=0)$ = 998,5 [kg/m³]

Reference concentration: $c = c_0 = 0$ [kg/m³]

Regardless of the input for ρ_0 it is calculated with ρ_0 = 998,5 kg/m³.

General density function for salt

The concentration-dependent density function for salt water (sea water) is in general:

$$\rho(T,c)= \rho(T,c = 0)*(1+\alpha_s * c)= \rho_0 +\alpha * c$$

$\rho(T, c=0) = \rho_0$ = density of the reference concentration [kg/m³]

α_s: The factor α_s depends on the proportion and type of dissolved salts and is for sea water (see above) by a temperature of T = 20°C: α_s = 0,000765 [m³/kg]

c = Reference concentration [kg/m³]

α = density gradient, $\alpha = \alpha_s * \rho_0$ [-]

Fluid viscosity

The fluid viscosity (or dynamic viscosity) η (for water: η(T=20°) = 0.001 [kg/(m s)]) is needed, because in the density-dependent mass transport calculation the pressure equation is used.

Boundary Condition

While the density-dependent calculation, the input potentials or water levels (EICH, VORF, POTE) are internally converted in the unit pressure with the known formula:

$p = (h - z) * \rho * g \ [N/m^2]$

If the water levels should be kept despite a change in density, the button "potential" (default) is enabled.

$p = (h - z) * \rho_0 * g \ [N/m^2]$ with $\rho_0 = \rho(c=c_0)$

If the water levels should be corrected by the change in density, the button "pressure" has to be activated.

$p = (h - z) * \rho * g \ [N/m^2]$ with $\rho = \rho(c)$

The buttons at the top of the input window allow resetting the input parameters ( ), opening an existing batch file ( ) or saving the current batch file under another name ( ).

6.8.3 Extended

By activating the button "Extended" more parameters can be defined for:

- Output
- Flow
- Transport
- Adsorption
- Degradation and production
- Half-life calculation

Since the input of these notebook pages are identical to those that were already described in the mass transport, see chapter: Mass transport - Extended.

6.8.4 Results of the density-dependent mass transport calculation

Intermediate results

As intermediate results of the density-dependent transient mass transport calculation the temporary concentration as well as the density at all nodes are saved in the output file „out.s" additionally to the pure flow relevant data.

Results

The calculation provides at the end of the density-dependent transient mass transport the final concentrations and the associated density of all nodes, additionally to the pure flow-relevant data.

If the resulting concentrations are required as starting values for a continuing calculation, they have to be converted from the unit kg/m³ into the unit kg/kg for the initial concentrations AKON. This can be done in three simple steps with SPRING:

- First, the calculated concentrations are imported via the menu *Attributes → Import model data/Computation results* to an arbitrary attribute, e.g. CCCC.
- Then, the associated density is imported as well via the menu to another arbitrary attribute, e.g. RRRR.
- In the last step the new AKON is calculated via *Attributes → Computation → Intercalculation...*: Concentrations CCCC / density RRRR = AKON (kg/kg)

6.8.5 Batch file density-dependent mass transport

It is possible to execute the *density-dependent flow* directly without using the dialog box. The default file name is *sitra.bsi*. To use another batch file, type *sitra file name* (the extension ".bsi" is automatically attached). Batch files can be created, viewed and modified with the help of any text editor.

A detailed batch file can be viewed in the online help. Similarly, there is a batch file of the density-dependent transient mass transport for further processing provided.

6.9 Heat transport

Call via: *SPRING →Calculation → Heat transport computation*

Analogous to the spread of solute (mass transport) in a porous medium the heat transport is also affected (energy spreading) through the processes of advection, diffusion and dispersion. However, when one speaks of heat transport the term convection is used instead of advection and heat conduction instead of diffusion.

In contrast to pure mass transport energy storage and transport take place both in the pores and in the solid matrix itself. Also a heat transport calculation is always a density-dependent calculation, because the density of the fluid and the solid matrix depends on the temperature.

The components of the mass transport are already described in the identically named chapter. Here, only the energy-specific processes are listed in abbreviated form.

6.9.1 Convection

The convective heat transport describes the by the movement of the groundwater itself transported heat energy. Thus, the convective heat flow j_k is the heat energy which is transported via a considered cross-sectional area A transverse to the direction of flow per time unit. Taking into account the velocity v the convective heat flow can be described as follows:

$$j_k = c_w * \rho_w * A * v * T$$

With:

j_k = convective heat flow [W]

c_w = specific heat capacity of the fluid [Ws/(kg K) = J/(kg K)]

ρ_w = density of the fluid [kg/m³]

A = cross-section of flow [m²]

v = distance velocity [m/s]

T = temperature [K]

6.9.2 Heat conduction

The heat flow due to both the thermal conductivity of the fluid $j_{m,w}$ and the matrix $j_{m,s}$ is described by Fourier's law (1822):

$$j_{m,w} = -\lambda_w * A * \nabla T \quad \text{and} \quad j_{m,s} = -\lambda_s * A * \nabla T$$

with

$j_{m,w}$ = heat flow in the fluid [W]

$j_{m,s}$ = heat flow in the matrix [W]

λ_w = thermal conductivity of the fluid [W/(m K)]

λ_s = thermal conductivity of the matrix [W/(m K)]

A = cross-section of flow [m²]

∇T = temperature gradient [K/m]

This part of the energy flow is independent of the movement of water. Similar to Darcy`s law, the negative sign results from the fact that the energy flows from high to low temperatures and the temperature gradient is reversed to this.

6.9.3 Hydromechanic dispersion

The hydromechanical dispersion describes the distribution and the mixing of heat in the moving pore water. The heat flow due to dispersion is also described by Fourier's law based on the heat conduction:

$$j_{disp} = -D_d * A * c_w * \rho_w * \nabla T$$

With:

j_{disp} = heat flow caused by dispersion [W]

D_d = symmetric dispersion tensor [m²/s]

A = cross-section of flow [m²]

c_w = specific heat capacity of the fluid [Ws/(kg K) = J/(kg K)]

ρ_w = density of the fluid [kg/m³]

∇T = temperature gradient [K/m]

6.9.4 Energy transport equation

Using the two variables:

c_w = specific heat capacity of the fluids [J/(kg K)]

c_s = specific heat capacity of the matrix [J/(kg K)]

and considering the pore volume n the law of conservation of energy can be mathematically described as follows:

$$\frac{\partial(n\rho_w S_r c_w T)}{\partial t} + \frac{\partial((1-n)\rho_s c_s T)}{\partial t} + \nabla(n\rho_w S_r c_w(j_k + j_{diffw} + j_{disp})) + \dots$$
$$\dots + (1-n)\rho_s c_s j_{diffs}) = qc_w \rho_w T*$$

In this case, the right side of the equation presents the sum of all energy sources and sinks and T* the temperature of incoming and outgoing amounts.

In this equation, the heat fluxes for convection, dispersion and heat conduction are inserted. Using the product rule for the convective term and the continuity condition (see Mass transport):

$$\nabla(nv) = q$$

the transient energy equation results:

$$nS_r\rho_w c_w)\frac{\partial T}{\partial t} + ((1-n)\rho_s c_s)\frac{\partial T}{\partial t} + (nS_r\rho_w c_w)v\nabla T - \nabla(nS_r\rho_w c_w(\sigma_w I + D_d)\nabla T) - \dots$$

$$\ldots - \nabla((\quad 1-n)\rho_s c_s \sigma_s I \nabla T) = q \rho_w c_w T \, ^*$$

The heat diffusion parameters σ_w [m²/s] and σ_s [m²/s] are defined as:

$$\sigma_w = \lambda_w /(\rho_w c_w) \quad \text{and} \quad \sigma_s = \lambda_s /(\rho_s c_s)$$

Unit of the parameters:

$\rho_{w,s}$ = density of the fluid (w) or the matrix (s) [kg/m³]

$c_{w,s}$ = specific heat capacity [J /(kg K)]

$\lambda_{w,s}$ = heat conductivity [W / (m K)]

$\sigma_{w,s}$ = diffusion parameter = $\lambda_{w,s}$ / ($\rho_{w,s}$ $c_{w,s}$) = [m²/s]

I = identity matrix [-]

D = dispersion tensor [m²/s]

Sr = saturation rate [-]

n = effective pore volume [-]

v = distance velocity [m/s]

q = div v (from continuity condition) [1/s]

T = temperature [K]

∇T = temperature gradient [K/m]

T* = variable temperature flow/discharge [K]

The energy transport equation and the heat production rate (QKON) have the unit [W/m³] = [J/(m³s)].

6.9.5 Boundary and initial conditions

The energy transport equation described above is supplemented by the boundary conditions:

■   Boundary condition 1st order: Specification of fixed temperatures (attribute 1KON):

$$T = T \, ^*$$

■   Boundary condition 2nd order: Specification of a diffusive-dispersive heat flow on the boundary. The almost exclusive practical application, however, is to make the edge impermeable for diffusion and dispersion, so that:

$$-(D\nabla T)N = j = 0$$

(N denotes the outward facing normal to the boundary of the model).

The system of equations can only be solved if at at least one node a boundary condition of the 1st order is defined.

For a transient heat transport equation initial conditions for the start time have to be defined for the entire model domain.

■   Initial condition: Initial concentrations for the start time t_0 of the transient calculation (attribute AKON):

$$T(t_0) = T_0^*$$

6.9.6 Temperature dependent parameters of the flow equation

Viscosity

Viscosity is a property that is due to the inner friction of the fluid. The unit of the viscosity is [kg/m s)] or Pascal-second [Pa s].

The dynamic fluid viscosity η [kg/(m s)] and, with a large temperature gradient, the density ρ [kg/m³] are temperature-dependent variables. Their change due to the temperature has the consequence that the differential equations (the transient flow and the energy transport equation) represent a coupled differential equation system.

An experimental formula for describing the relationship between dynamic fluid viscosity and temperature is used in the calculation:

$$\eta = \eta(T) = (239{,}4 * 10^{-7}) * 10^{\frac{248{,}7}{T+133{,}15}}$$
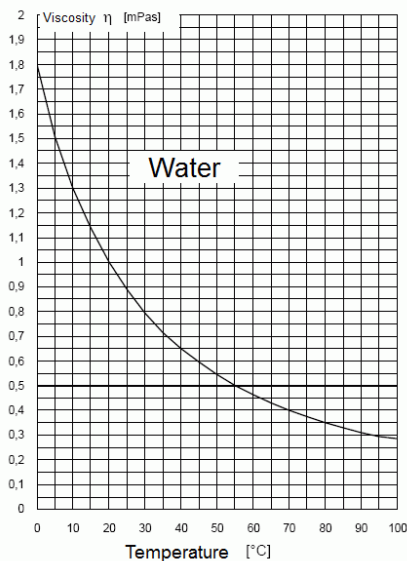
Fig. 204: Dynamic viscosity η of water dependent on the temperature at a density of ρ= 1000.0 [kg/m³]

The unit millipascal second [mPa s] corresponds to the unit 1000*[kg/(m s)].

The following relationship applies for dynamic and kinematic viscosity:
dynamic viscosity: $\eta = \mu * \rho$ [kg/(m s)]
kinematic viscosity: $\mu = \eta / \rho$ [m²/s]

Temperature-dependent density

The linear dependency of the density [kg/m3] from the temperature T [°C] is considered in SPRING:

$$\rho(T) = \rho_0 + \alpha(T - T_0)$$

With:
$\rho(T)$ = temperature dependent density [kg/m³]

ρ_0 = density of the reference temperature T_0 in [kg/m³], default is $\rho_0 = 1000.0$ [kg/m³]:

$$\rho_0 = \rho(T = T_0)$$

T_0 = Reference temperature [°C], default is $T_0 = 20°C$
α = constant density gradient, default is $\alpha = 0.0$

The following figure shows the dependency of the density from the temperature:
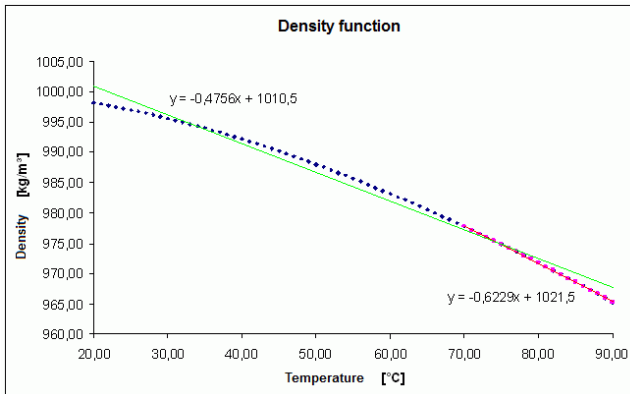
Fig. 205: Temperature-dependent density function

In this case is:

$\rho_0 = \rho(T_0 = 20\ °C) = 1010{,}5\ [kg/m^3]$

The density function is approximated by a linear equation (green line) in the considered temperature interval (here: about 32 – 75 °C), by this the density gradient α is defined.

Here, the density gradient α = constant = -0.4756. (This corresponds to the slope (= 1st derivative) of the line):

$$\alpha = \frac{\partial \rho}{\partial T} = \textbf{const.}$$

If the initial or computed temperatures lie at a different temperature range, another fit line is to be defined (red line).

6.9.7 Realization in SPRING

All model data in the input file and the transient file which are usually related to concentration are, in the case of a heat transport calculation, related to the temperature:

Data	meaning in the transport calculation	meaning in the heat transport calculation
KONZ	concentration of inflow rates	temperature of inflow rates
1KON	fixed concentration (first order boundary condition)	fixed temperatures (first order boundary condition)
AKON	Initial concentrations	Initial temperatures
QKON	-	Heat production rate

Analogously, the results from the heat transport calculation in the output files and the postprocessing modules (plot generation, data export, ...) are "temperatures" but called "concentrations".

6.9.8 Input parameters

Via *Calculation → Heat transport computation* the following input window appears:
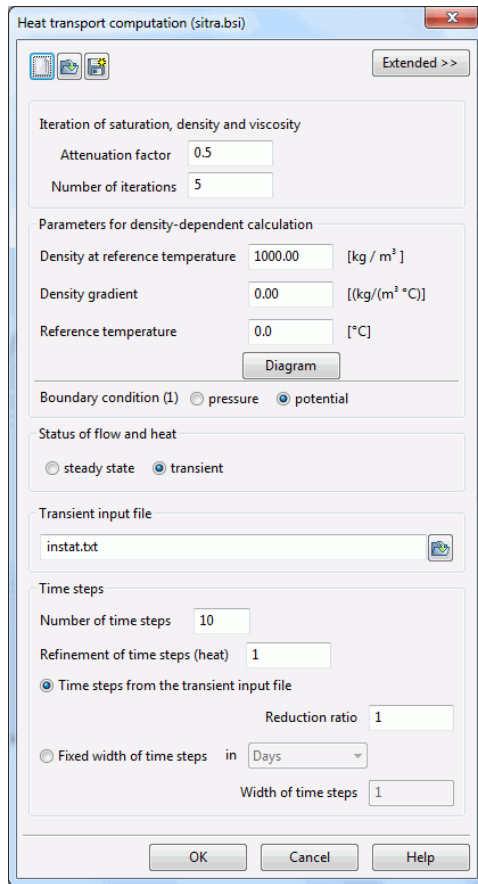
When calling the module, the program reads the batch file with the default name (here: sitra.bsi).

Besides the already known input parameters „Status of Flow and Heat" and „Time steps" (see „Input parameters Mass transport"), in the heat transport calculation the parameters for the change of density depending on the temperatures in the fluid are defined.

Iteration of saturation, density and viscosity

The required number of iteration steps and the attenuation factor are determined.

Density parameters

Density ρ_0 at the reference temperature T_0 in [kg/m³]: $\rho_0 = \rho(T = T_0)$

Density gradient α = constant, default is $\alpha = 0.0$

Reference temperature T_0 in [°C]

The corresponding diagram can be represented and saved:
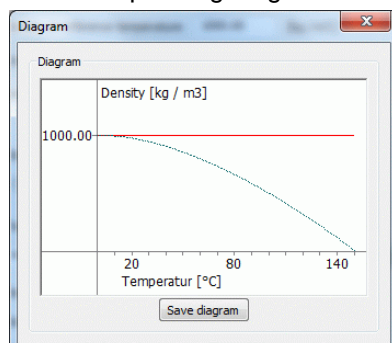
Fig. 206: Density dependent on the temperature

Boundary Condition

While the heat transport calculation, the input potentials or water levels (EICH, VORF, POTE) are internally converted in the unit pressure with the known formula:

$p = (h - z) * \rho * g \ [N/m^2]$

If the water levels should be kept despite a change in density, the button "potential" (default) is enabled.

$\qquad p = (h - z) * \rho_0 * g \ [N/m^2]$ with $\rho_0 = \rho(T=T_0)$

If the water levels should be corrected by the change in density, the button "pressure" has to be activated.

$\qquad p = (h - z) * \rho * g \ [N/m^2]$ with $\rho = \rho(T)$

Status of flow and heat

In the heat transfer calculation, it is only possible to calculate the flow **and** transport either steady state or transient.

The buttons at the top of the input window allow resetting the input parameters ( ), opening an existing batch file ( ) or saving the current batch file under another name ( ).
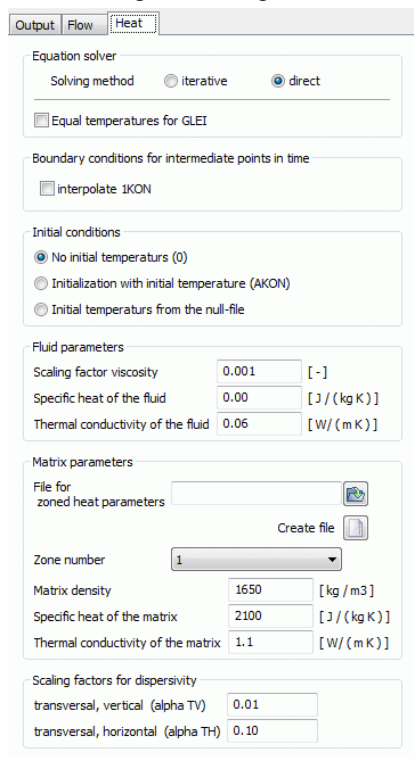
6.9.9 Extended

By activating the button "Extended" more parameters can be defined for:

- Output
- Flow
- Heat Since the input of these notebook pages are identical to those that were already described in the mass transport, see chapter: Mass transport - Extended .

6.9.9.1 Heat

After calling the dialog window "Heat" the following input window appears.

Equation solver

For solving the equation system there are an iterative PCG solver and a direct Cholesky solver. For meshes of about 500 nodes or more, you should always choose the iterative one, because it is much faster. The error due to the iterative solution is always much smaller (less than 3%) than the other errors (e.g. measurement error). This can be verified with the help of the mass balance printed to the output file.

Equal temperatures for GLEI

This determines whether at nodes with the attribute GLEI also the equal temperatures should be calculated.

Boundary conditions at intermediate point in time

If the calculation is done with a fixed time step size or diminished time steps of the transient input file, it is possible at the intermediate time steps, which are not regulated by the transient input file to interpolate transient boundary conditions for the temperatures.

Initial conditions

Here is defined with which initial temperatures the transient heat transport calculation is started:

- No initial temperatures: The iteration is started with initial temperatures = 0.0 (not recommended).
- Using initial temperatures: The initial temperatures of the model file (attribute AKON) are used.
- The transient calculation can be continued, if the corresponding null-file from the previous calculation exists (warm start). If the user wants to continue the actual calculation, the program has to create the corresponding file called out66. This is specified in the choice of output parameters (Save for continuing (out66)).

Fluid parameters

The scaling factor viscosity (η_0) scales the underlying formula for the calculation of the dynamic viscosity as a function of the model species:

$$\eta = \eta(T) = (239,4 * 10^{-7}) * 10^{\frac{248,7}{T+133,15}}$$

by:

$$\eta = \eta_0 * \eta(T)$$

For vertical- or 3D models is $\eta_0 = 1.0$ [-] (default), because in the calculation of these model types, the pressure equation is used. In horizontal models is $\eta_0 = 1000.0$ [-], because in the calculation of this model type, the potential equation is used.

The specific heat capacity of the fluids in [J/(kg K)] has to be entered.

The heat conductivity of the fluid in [W/(m K)] has to be entered.

Matrix parameters

If the zoning attribute Z-KD is available in the model file, the associated file with the zoning data can be selected for the matrix parameters.

The density of the matrix in [kg/m³] has to be entered.

The specific heat capacity of the matrix in [J/(kg K)] has to be entered.

The heat conductivity of the matrix in [W/(m K)] has to be entered.

Creating a new file with zoning data is explained in the chapter "Structure of the zoning files", p. 30.

Saving the dialogue above results in the following entry in the file with the zoned heat parameters:

```
Zone nr spec. heat capacity (matrix) heat conductivity (matrix) density (matrix)
      1      2100       1.1      1650      #Zone number CS SIGMAS RHOS
```

Dispersivities

Defining transversal-horizontal (α_{TH}) and transversal-vertical (α_{TV}) dispersivities [m].

The scale refers to the longitudinal dispersivity, which is defined in the model file using the attribute DISP.

6.9.10   Results of the heat transport calculation

Intermediate results

As intermediate results of the transient heat transport calculation the temporary temperatures at all nodes are saved in the output file „out.s" additionally to the pure flow relevant data.

Results

The calculation provides at the end of the transient heat transport the final temperatures of all nodes, additionally to the pure flow relevant data.

6.9.11   Batch file of the transient heat transport calculation

It is possible to execute the *heat transport* directly without using the dialog box. The default file name is *sitra.bsi*. To use another batch file, type *sitra file name* (the extension ".bsi" is automatically attached). Batch files can be created, viewed and modified with the help of any text editor.

A detailed batch file can be viewed in the online help. Similarly, there is a batch file of the transient heat transport for further processing provided.

6.10 Transport with inverse flow

Call via: *SPRING → Calculation → Transport with inverse flow*

It is possible to represent catchment areas considering advection and dispersion and diffusion processes. Thus, even local heterogeneities are shown using the dispersion term which cannot be determined with the traditional methods of velocity tracking.

The procedure and the basics are already described in chapter:"How To – Determination of catchment areas".

The input parameters correspond to the parameters of the mass transport and are described in detail there.

7 Data export

It is possible to export different data types from the groundwater model via *File → Export*:
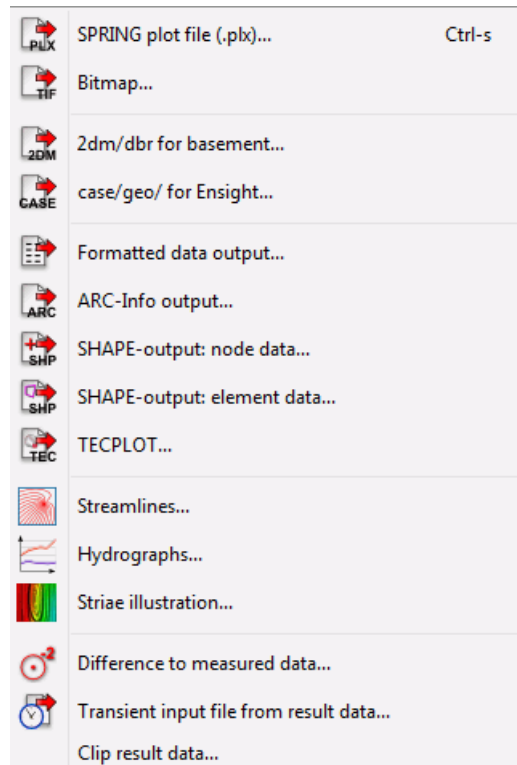
The first four menu items make it possible to save the following without further input:

■   the current display (*.*plx*)

■   the current display as geo referenced file (*.tif, *jpg, *.bmp, *png)

■   the mesh in 2dm/dbr-format

■   the mesh in case/geo-format for Ensight (2D model only)

When you save the current view as geo referenced raster image the image's width and the number of dpi are then set in the window that opens.
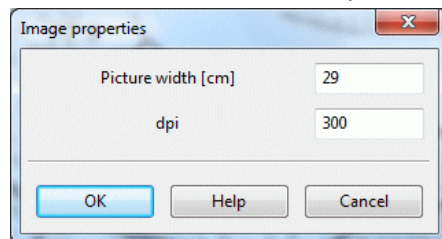
The following items are from the post-processing module NACHLAUF. They are described in detail in the following chapters.

7.1 Formatted data output

Call via *File → Export → Formatted data output*

The data export can transform almost any input or output data in a desired format to an ASCII file. For example, data can be used to create tables and plots with the help of spread sheet programs.

7.1.1 Input parameters

The following input window appears in a 3D model:
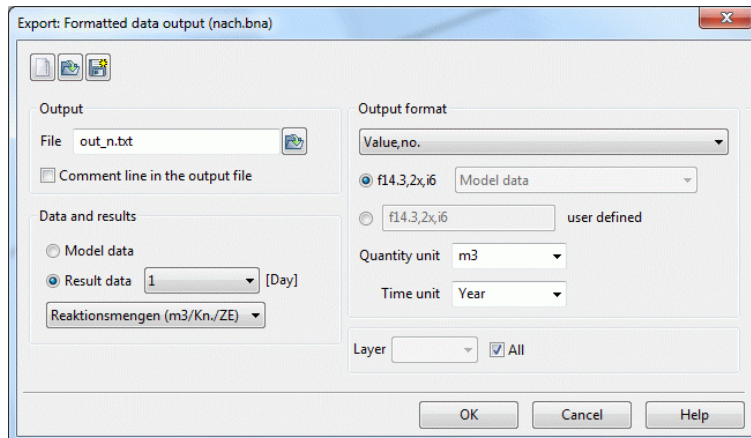
Fig. 207: Input for the formatted data output

Output

The text field requires the name of the output file. If the box "Comment line in the output file" is checked the output file contains a comment line. A file without comment line can be overlaid in SPRING directly as "txt"-file.

Data and results

If result data is selected after a transient calculation another selection box appears where the desired time step is defined.

Output format

You can choose from four output formats. The meaning is:

- No.: node or element number
- X: easting/X-coordinate
- Y: northing/Y-coordinate
- value: input or calculated value for each node or element

By selecting the output format, for example "Value, no" appears at first a SPRING-specific format, here: the model file format. This can be replaced by a self-defined format.

The self-defined format shall be equal to the parent: If the parent format is "Value, Number ", the self-defined format must have the order "Floating point number, Integer", too. The number of digits or the format description tion can be chosen freely. The following formats are allowed:

- F or f
- x or X
- I or i
- E or e

Example:

- F14.3: Floating-point number with 14 points, including three decimal places
- 2x: two empty places
- i6: six-digit integer

If any other character instead of a comma should be used as column separator, this can be defined as follows:
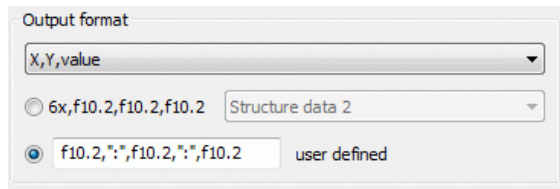
Remark: If "asterisks" (*) appear in the output file instead of numbers, the format does not match with the actual existing digits.

Example:

For the node 123 a value of -1000000 m³/s (attribute KNOT) should be written in the format of the transient input file (no. with format I6, value with format F10.2). With decimal point and two decimal digits results for the value a total number of 11 digits.

→ There appear asterisks in the output file, because the input output format only allows a total of 10 digits! This problem can be solved by entering a user-defined format "I6, E10.2". With this a readable value is written in the output file.

Quantity unit

If a quantity-based (model or results data) data type (e.g. mass flow rates or groundwater recharge rates) is chosen another check box appears where you can set the unit of quantity/time itself.

Layer (3D model)

If the project is a 3D model the corresponding layer or all layers can be selected.

The buttons at the top of the input window allow resetting the input parameters ( ), opening an existing batch file ( ) or saving the current batch file under another name ( ).

7.1.2 Batch file formatted data output

It is possible to execute the *formatted data ouput* directly without using the dialog box. The default file name is *nachlauf.bna*. To use another batch file, type *nachlauf file name* (the extension ".bna" is automatically attached). Batch files can be created, viewed and modified with the help of any text editor

A detailed batch file can be viewed in the online help. Similarly, there is a batch file of the formatted data output for further processing provided.

7.2 Data output for ArcInfo

Call via *File → Export → ARC-Info output*

Most of the input data and the results can be written to ASCII files using the data export interface with ARC/INFO. The GIS program can load the resulting files and use the data. The files contain all the necessary information to prevent misinterpretation in ARC/INFO (e.g. smearing via interpolation). Hence, the program can intersect and blend the data with other data, create equal value contours, etc.

7.2.1 Input parameters

The following input window appears:
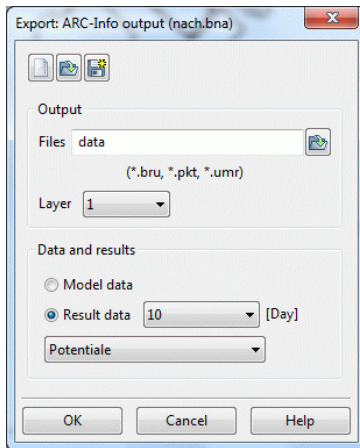
Fig. 208: Input for the ARCInfo data output

Output

Here the name of the output file is determined to which the ArcInfo-specific extensions are attached. Additionally, the layer number for the data to be exported in a 3D model is specified.

Data and results

If result data is selected after a transient calculation another selection box appears where the desired time step is defined.

If the export data are node data, three output files are created:

File	Contents
<name>.**bru**	element edges with nodewise data at the element corner nodes
<name>.**pkt**	element centres with nodewise data
<name>.**umr**	mesh boundary polygons (outer and inner boundaries) with nodewise data at the corner nodes

If the export data are element data, four output files are created:

File	Contents
<name>.**bru**	element edges *without* any data (this file has the same appearance for all data types!)
<name>.**pkt**	element centres with element numbers *without* any data (this file has the same appearance for all data types!)
<name>.**umr**	mesh boundary polygons (outer and inner boundaries) *without* any data (this file has the same appearance for all data types!)
<name>.**tab**	table with element numbers and element data

The buttons at the top of the input window allow resetting the input parameters ( ), opening an existing batch file ( ) or saving the current batch file under another name ( ).

7.2.2 Batch file for ArcInfo output

It is possible to execute the *ArcInfo ouput* directly without using the dialog box. The default file name is *nachlauf.bna*. To use another batch file, type *nachlauf file name* (the extension ".bna" is automatically attached). Batch files can be created, viewed and modified with the help of any text editor

A detailed batch file can be viewed in the online help. Similarly, there is a batch file of the ArcInfo output for further processing provided.

7.3 Data output in SHAPE format

Call via *File → Export → SHAPE-output: node/element data*

Many programs, such as geographic information systems, have an interface for the SHAPE format.

It is decided in advance what data (node-or element-based) is to be exported. The input window for both species, apart from the offered data types, is identical.

A SHAPE block consists of two components in three files: the geometry information (*.shx- and *.shp-files) and the attributes in the database file (*.dbf, dBase-format). Because some programs limit the notation of the attributes to 11 characters SPRING creates a *.alias file in ASCII-format which contains the exact notations of the data sets and if necessary the information about the layer number and/or the time step.

7.3.1 Input parameters

The following input window appears:
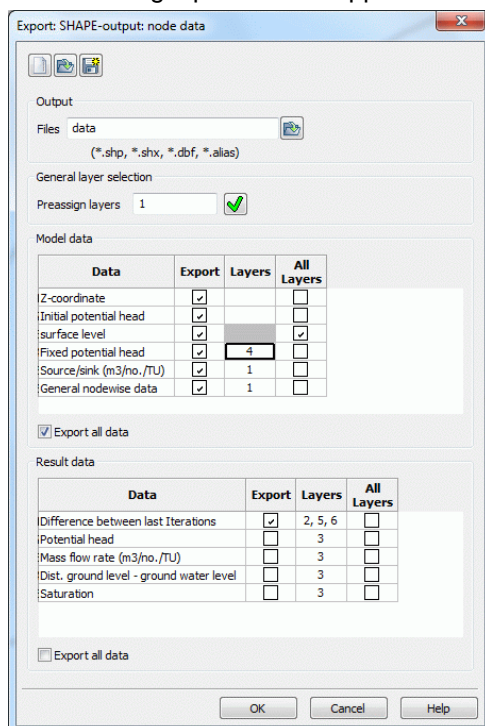
Fig. 209: Input for the output in SHAPE format

Output

Here the name of the output file is determined to which the SHAPE-specific extensions are attached.

File	Contents
<name>.**shx**	Indices of the graphic-objects
<name>.**shp**	Coordinates of the graphic-objects
<name>.**dbf**	Attributes in the dBase-format
<name>.**alias**	Table with the exact notations of the attributes if necessary with layer and time information (no rather SHAPE-file)

General layer selection

(3D models only)

For the selection of the layers where an output is desired, two possibilities are available:

General layer selection: In the input box the desired layers are entered. After confirming the selected layers are assigned to all available records. This does not mean that all records have been selected for output. This type of layer selection is appropriate when multiple data types should be exported for the same layers. Then, the layer must be re-entered only once and not for any type of data.

Individual layer selection (for model and result data): For each record displayed, the layers can be issued by entering the desired number(s) in the field "Layers". This will change the preassigned numbers using the action "General layer selection. Activating the input field involves automatically the selection of the correspondent data set. This type of layer selection is appropriate if only a few data types or very different layers should be exported.

Data and results

The desired data types and the layers are selected (see above).

Go to …

By entering a point of time the result data at this point of time is presented (after a transient calculation).

The buttons at the top of the input window allow resetting the input parameters ( ), opening an existing batch file ( ) or saving the current batch file under another name ( ).

7.3.2 Batch file for the output in SHAPE format

It is possible to execute the *output in SHAPE format* directly without using the dialog box. The default file name is *nachlauf.bna*. To use another batch file, type *nachlauf file name* (the extension ".bna" is automatically attached). Batch files can be created, viewed and modified with the help of any text editor

A detailed batch file can be viewed in the online help. Similarly, there is a batch file of the output in SHAPE format for further processing provided.

7.3.3 Export of contour lines

In SPRING the export of contour lines with assigned values is possible by using the following command in the batch file of the plot generation.

```
gela.plx
HORI    ## Horizontalschnitt
1                                              # oberste Knoten/Elementschicht
TFLD    ## Textfeldtyp
1 0                                            # Textfeldtyp II, Anzahl Folgezeilen
KOOR    ## Koordinatenrahmentyp
1                                              # Typ=0/1
SKAL    ## Skalierungsfaktoren für Plot
1.0 1 1                                        # global, in x-Richtung, in y-Richtung
ISOL    ## Isolinienplot
SHP
85 0 0 0                                       # Geländeoberfläche
-51 0.25 1                                     # Farbe (<0->keine Beschr.), Strichstärke, Linientyp
1 10                                           # äquidistante Unterteilung -> Anzahl
RAND    ## Netzrandplot
1 0.25 1                                       # Farbe, Strichstärke, Linientyp
# ---> Plot Ende!
```

By adding "SHP" (red arrow) in the representation block of the desired data type (as contour lines) the corresponding shape-files are exported after performing the batch file. Input and performing in the dialog of the plot generation is currently not possible.

The contours lines should be displayed WITHOUT labels ("-" in front of the color number, blue arrow), as otherwise gaps in the exported contour lines arise.

If the interval between the contour lines to be exported is less than 1.0, the user should adjust the following settings before performing the plot:

In the dialog box *SPRING → Edit → Options: Plot → Contour lines* the number of decimal places has to be one at minimum (depending on the interval). Otherwise multiple contour lines are of equal value.

7.4 Data output for TECPLOT

Call via *File → Export → TECPLOT …*

TECPLOT is a commercial visualization program, which helps to generate three-dimensional graphics of model and result data from 2D or 3D models. Most model and result data can be exported with this menu item TECPLOT-compatible to ASCII files.

A subsequent visualization of path lines and creating 3D animations is possible with TECPLOT. Because TECPLOT only manages node data, all the element data is first averaged to node data.

In case of 3D models with 2D fractures, the data export for TECPLOT supports the separated data output on the fracture planes. Otherwise, the node or element data of the whole mesh are printed to the output file.

7.4.1 Input parameters

The following input window appears:
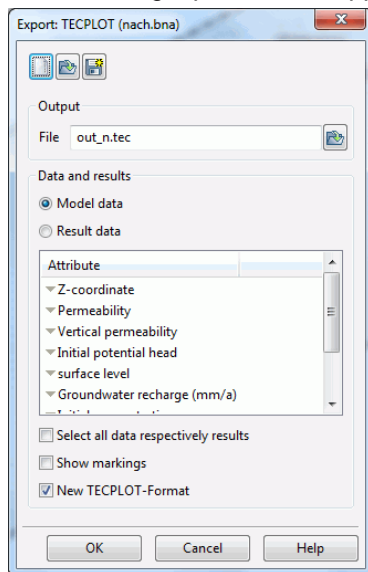
Fig. 210: Input for the TECPLOT output

Output

Here the name of the output file is determined to which the specific extensions are attached.

Data and results

The desired data types are selected here.

By entering a point in time at "Go to ..." the presentation jumps through the results (after a transient calculation) to the entered date.

The output for TECPLOT generally writes off the model or results data for all layers. In the program TECPLOT itself the user can then select the desired layer. By holding down the Ctrl key multiple data types can be selected. If all model and results data should be exported, there is a checkbox "Select all data respectively results" below the list.

Furthermore, there is the possibility to export the in the model existing markers for TECPLOT. For this, the checkbox "Show markings" should be activated.

To save in the new block format of TECPLOT the corresponding checkbox has to be enabled. This format exports the layer number as an element-centric attribute in 3D models, which allows the selective hiding or showing of element layers in TECPLOT.

The buttons at the top of the input window allow resetting the input parameters ( ), opening an existing batch file ( ) or saving the current batch file under another name ( ).

7.4.2 Batch file for the TECPLOT output

It is possible to execute the *TECPLOT ouput* directly without using the dialog box. The default file name is *nachlauf.bna*. To use another batch file, type *nachlauf file name* (the extension ".bna" is automatically attached). Batch files can be created, viewed and modified with the help of any text editor

A detailed batch file can be viewed in the online help. Similarly, there is a batch file of the TECPLOT output for further processing provided.

7.5 Output of streamlines

Call via *File → Export → Streamlines*

After the steady state flow calculation of a 2D horizontal model streamlines can be computed. The resulting streamlines can be different, depending on the program module used (GEONEU or SITRA). Firstly, the streamline calculation is highly sensitive with respect to the numerics. Even very small changes in potential heads or velocities cause large differences in the mass flow rates. Secondly, the different FE calculation affects the results (see chapter: Steady state flow).

GEONEU divides each element into triangles. This type of discretization, in contrast to that used by the linear or bilinear finite element shape functions in SITRA, calculates an additional value at the element centre.

7.5.1 Input parameters

The following input window appears:
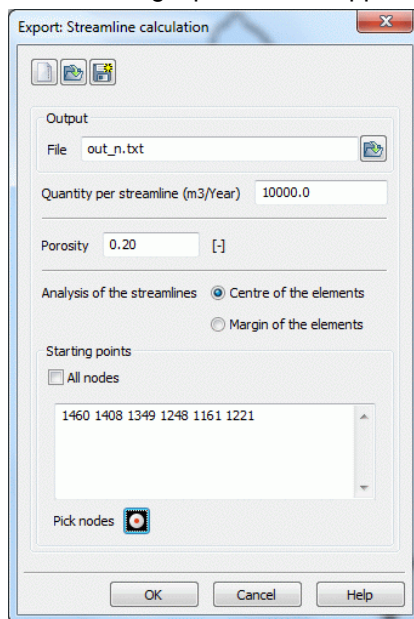
Fig. 211: Input for the streamline output

Output

The name of the output file is defined.

Quantity per streamline

Since the streamline calculation is volume-based here is entered for what amount the streamline will be calculated. The entry is made in the unit [m³/year].

In the plot presentation flows the entered amount between two streamlines.

Porosity

The necessary porosity used to convert the filter velocity is provided (between 0 and 1). It applies uniformly to all elements. An analysis of the data type as porosity is currently not possible.

Analysis of the streamlines

The evaluation of the streamline calculation can be done either from the centre or at the margin of the elements. In general, the evaluation done from the centre provides the more pleasant, smoother streamlines.

Starting points

The streamline calculation can be performed either for all or for specific nodes. Single nodes are to be entered in a further input field which appears after unchecking the box "*all nodes*". It is also possible to define the nodes interactively by selecting them directly in the SPRING surface (activating "*capture nodes*").

The starting points must be nodes to which a withdrawal is assigned. If the streamline calculation is done globally, an appropriate number of lines is started at the mesh boundary, at line markers and in all withdrawal points that amount is at least equal to the current size of quantity.

The buttons at the top of the input window allow resetting the input parameters ( ), opening an existing batch file ( ) or saving the current batch file under another name ( ).

7.5.2 Batch file for the output of streamlines

It is possible to execute the *output for streamlines* directly without using the dialog box. The default file name is *nachlauf.bna*. To use another batch file, type *nachlauf file name* (the extension ".bna" is automatically attached). Batch files can be created, viewed and modified with the help of any text editor.

A detailed batch file can be viewed in the online help. Similarly, there is a batch file for the output of streamlines for further processing provided.

7.6 Output of hydrographs

Call via *File → Export → Hydrographs*

Transient results that may be issued in the form of hydrograph curves, can be exported in csv format (comma separated) using this interface. The csv-format is supported from database and spread sheet programs.

7.6.1 Input parameters

The following input window appears:
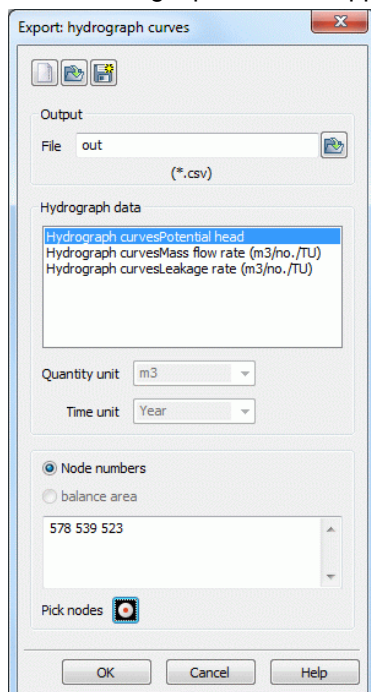
Fig. 212: Input for the hydrograph output

Output

Here the name of the output file is determined to which the extension *.csv is attached.

Hydrograph data

The type of data is determined which should be exported for the hydrograph. If it is a quantity-based data type the unit can be determined explicitly [quantity per unit of time].

Selecting the nodes, elements or balance areas

Depending on the selected data type here the node or element numbers are set for which the hydrographs should be generated. They can also be captured interactively on the screen (capture nodes/elements).

Were balance areas (attributes BILK or BILE) defined, it must be specified if the hydrographs are created for individual nodes or elements or for the whole areas.

Node or element numbers from deeper layers in 3D models need to be increased by the corresponding 3D-number offset

The buttons at the top of the input window allow resetting the input parameters ( ), opening an existing batch file ( ) or saving the current batch file under another name ( ).

7.6.2 Batch file for the output of hydrographs

It is possible to execute the *output for hydrographs* directly without using the dialog box. The default file name is *nachlauf.bna*. To use another batch file, type *nachlauf file name* (the extension ".bna" is automatically attached). Batch files can be created, viewed and modified with the help of any text editor.

A detailed batch file can be viewed in the online help. Similarly, there is a batch file for the output of hydrographs for further processing provided.

7.7 Schlieren illustration

Call via *File → Export → Schlieren illustration*

The program FLIC (Konrad-Zuse-Centre for Information Technology Berlin) can visualize the velocities computed in a flow calculation by the help of Schlieren. The underlying method is called "Line Integral Convolution (LIC)". This visualization provides a much better image of flow paths than the velocity arrows used in the plot generation. This is especially important for transient and 3D models where a streamline calculation is not possible.
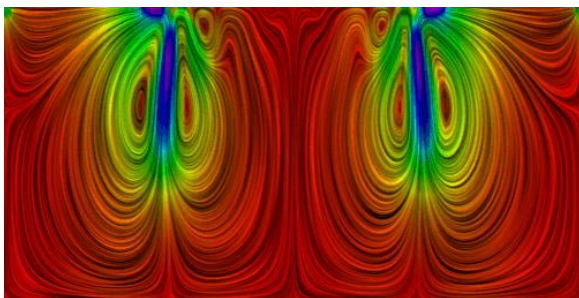
Fig. 213: Example of a Schlieren illustration

The data export creates an ASCII input file with the computed velocities for the FLIC program. The ASCII interface to the program FLIC is based on grid data. That's why the computed velocities must be determined at grid nodes beforehand. Additionally, it is possible to colour the Schlieren by using values from the input or the resulting data which exist at every node or element. These values (i.e. concentrations) are stored in an ASCII file. Markings are issued as a possible overlay file (*.mar).

Calling FLIC is done automatically during the export of data or can be started manually from a terminal on the command line:

   flic cmd=<command file> (*.scr)

FLIC then generates a bitmap image in jpg format, which can be read using a suitable graphics program. In 2D horizontal models and horizontal sections of 3D models the corresponding world file (extension *. jgw) is also created with the information required for georeferencing. Below the FLIC commands are listed, which can also be entered directly into the command line.

```
cmd=<filename>      read more commands from a file
field=circle        use a circular vector field
field=spiral        use a spiral-type vector field (default)
field=<filename>    read a vector field from a file
out=<filename>      name of the resulting LIC image (required)
size=<nx>x<ny>      size of noise input image (default 256x256)
in=bwnoise          create black-white noise image
in=colornoise       create color noise image
in=<filename>       use given image file as input
resize=<nx>x<ny>    resize current input image
kernel=box          select box filter kernel
kernel=triangle     select triangle-shaped filter kernel (default)
kernel=cubic        select third-order B-spline filter kernel
length=<n>          length of filter kernel in pixels (default=20)
subpix=<n>          enable nxn super-sampling to reduce aliasing
contrast=raw        no contrast adjustment (default for file input)
contrast=auto       contrast adjustment (default for noise input)
enhance=0           disable post-filtering of LIC image (default)
enhance=sharpen     compute directional derivative of LIC image
enhance=emboss,<a>  apply emboss filter, angle arg is optional
integrator=euler    use Euler method for field integration
integrator=rk43     use adaptive Runge-Kutta integrator
interpol=const      nearest-neighbour interpolation of input image
interpol=bilinear   bilinear interpolation of input input (default)
colorfield=dir      color denotes vector orientation (fixed colormap)
colorfield=mag      compute color from vector field magnitude
colorfield=<file>   compute color from scalar field stored in file
colormap=hueramp    use hueramp for pseudocoloring
colormap=hotiron    use rot-yellow-white colormap
colormap=<file>     read colormap from file (icol format)
colorrange=auto     colormap min max adapted to colorfield (default)
colorrange=min,max  specifies colormap min max values
overlay=<file>      render graphics on top of output image
verbose=0|1         enable verbose mode (default is off)
```

7.7.1 Input parameters

The following input window appears:
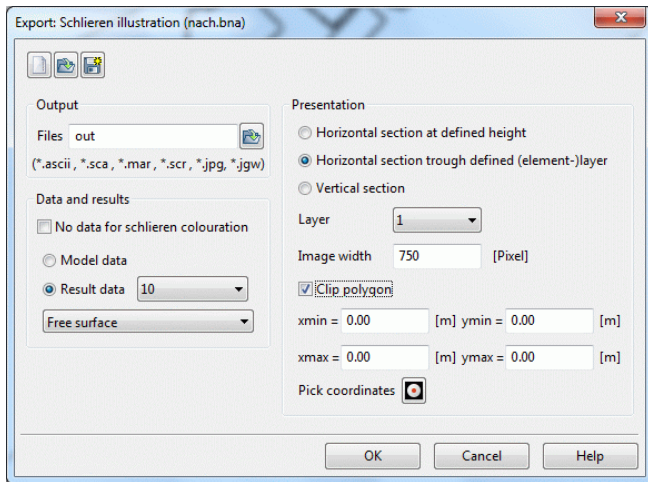
Fig. 214: Input for Schlieren illustration

Output

Here the name of the output file is determined to which the specific extensions are attached.

File	Contents
<name>.**ascii**	This file contains the for FLIC necessary information about the velocities at the nodes and element centres in a 2D model or a horizontal section through a 3D model. It also contains the incidence of the triangle mesh (SPRING-squares are subdivided in 4 triangles, SPRING-triangles are subdivided in 3 triangles) used in FLIC. In a vertical section the file contains the for FLIC necessary information about the velocities at the grid points.
<name>.**sca**	This file is only created when scalar data are selected for colouring the Schlieren. It contains the necessary data at the nodes and element centres in a 2D model or a horizontal section through a 3D model, supplemented by the triangle incidence used in FLIC. In a vertical section through a 3D model, the file contains the values of the data at the grid points.
<name>.**mar**	This file is only created when markings are present (though not in a vertical section through a 3D model). Markers are realized here as "asterisk". If these markers are not desired, the corresponding part in the file (after the comment "point markers") has to be deleted.
<name>.**scr**	This file is used as a command file for FLIC (called: "flic cmd = <output_file> scr").
	The filename of the bitmap generated with FLIC is preset to <output_file>.jpg (command line "*out* =..."). Alternatively, a pgm-format is available by changing the suffix to .pgm.
	If markings were issued as an overlay file, the file is overlaid with the command "*overlay* =...".
	If scalar data were output on the grid, these are used for colouring using the command "*colorfield* =...". The extreme values of the scalar data in the command "*colorrange* =..." play a crucial role in colouring. The used colour palette can be shifted by modifying these extreme values.
	If no scalar data are used for colouring, the velocities are crucial for the colouring ("*colorfield* =mag"). Here are also the extreme velocities essential for the colouring in the command "*colorrange* =...".
<name>.**jpg**	Image file
<name>.**jgw**	To the image file corresponding georeferencing file

Data and results

The default setting is displaying the velocities. Only by unchecking the box data for the colouring of the Schlieren can be optionally selected.

By selecting the model or result data appears a submenu in which the data types are listed, which have values at all nodes or elements. If there are transient result data the point of time must additionally be selected.

Presentation

Image width in pixel

The horizontal resolution of the Schlieren image can be specified in pixels. The resulting vertical resolution is determined in the calculation and passed on to FLIC. Thus, unintended distortions in particular during the clipping are prevented. The default size of the bitmap generated with FLIC amounts to 750 pixels in x-direction for 2D models or horizontal sections through 3D models and is rounded in the y-direction according to the model size. In vertical sections, the default size is based on the defined grid (10 times the number of grid points) (command line "*size = x...* ").

Clip polygon

At this point, a rectangle can be defined, in which the Schlieren image is displayed.

By activating the button *Pick coordinates* the points are captured interactively in the SPRING-surface.

For **3D models** has to be selected additionally whether a horizontal or vertical section should be exported as presentation.

The position of the **horizontal section** can be defined in two ways:

- Horizontal section in a defined height:
  Input of the height (in m) is required.

- Horizontal section through a defined (element-) layer:
  Input of the (element-) layer number is required, in that case the dialog changes from „height" to "layer".

The **vertical section** of the Schlieren illustration is not performed on the basis of the original mesh, but on a grid of the cut surface. The Schlieren are presented based on the calculated velocities at the grid points. The finer the grid is, the better is the colour representation. However, the reduction of the grid width under the existing FE mesh element size improves the representation only up to a certain point, after that only the computational cost increases. To define the position of the vertical section and the grid the following input is required:

Data for rasterisation

■   Extreme Z-coordinates:

The minimum and maximum height as the bottom and topmost boundaries of the vertical section (zmin and zmax) are required.

■   Location of the vertical section in the horizontal:

The location of the vertical section is defined by a minimum of two points in the horizontal plane.

At least two points are indicated in the horizontal plane to determine the section. This can be done manually by entering the coordinates or interactively in the SPRING-surface ("*pick coordinates*" 🔘).

■   Definition of the grid distances:

The grid spacing is determined by the input of the grid sizes in the horizontal and vertical directions.

During the calculation the velocity vector and eventually the value of the 'scalar' data are calculated at all co-ordinates (x, y, (z)) defined by the grid spacing and grid edges.

The buttons at the top of the input window allow resetting the input parameters (▯), opening an existing batch file (▯) or saving the current batch file under another name (▯).

7.7.2 Batch file for the Schlieren illustration

It is possible to execute the *Schlieren illustration* directly without using the dialog box. The default file name is *nachlauf.bna*. To use another batch file, type *nachlauf file name* (the extension ".bna" is automatically attached). Batch files can be created, viewed and modified with the help of any text editor.

A detailed batch file can be viewed in the online help. Similarly, there is a batch file of the Schlieren illustration for further processing provided.

7.8 Output of differences to measured data

Call via *File → Export → Differences to measured data*

With this tool, differences between calculated values (e.g. depth to water table) or model data (i.e. initial potential heads) and measured data (i.e. measured basement heights) can be calculated and stored in an ASCII-file.

7.8.1 Input parameters
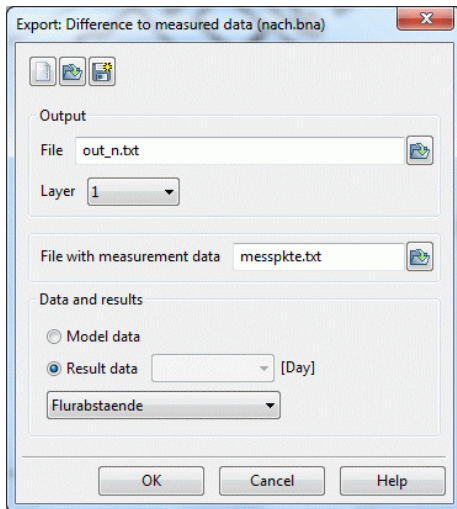
The following input window appears:

Fig. 215: Input for the differences to measured data

Output

The name of the output file is defined. The calculated differences are stored in the structure data format.

Layer

The desired layer number is defined in a 3D model.

File with measured data

The file which contains the measured values for the intercalculation is selected. The file must contain the measurement data (x_i y_i, value$_i$) in the structure data format (I6, F10, F10, F10).

Data and results

By selecting the model or result data appears a submenu in which the data types are listed, which have values at **all** nodes or elements. If there are transient result data the point of time must additionally be selected.

The buttons at the top of the input window allow resetting the input parameters ( ), opening an existing batch file ( ) or saving the current batch file under another name ( ).

7.8.2 Batch file for the output of differences to measured data

It is possible to execute the *output for the differences* directly without using the dialog box. The default file name is *nachlauf.bna*. To use another batch file, type *nachlauf file name* (the extension ".bna" is automatically attached). Batch files can be created, viewed and modified with the help of any text editor.

A detailed batch file can be viewed in the online help. Similarly, there is a batch file for the output of the differences for further processing provided.

7.9 Output of a transient input file from result data

Call via *File → Export → Transient input file from result data...*

With this tool, transient data can be saved after a calculation in a (new) transient input file. If, for example, a sub-model requires a transient boundary structure this menu item writes calculated potential heads (or any other transient node data) into a new transient input file. The user can specify the nodes and the time steps.

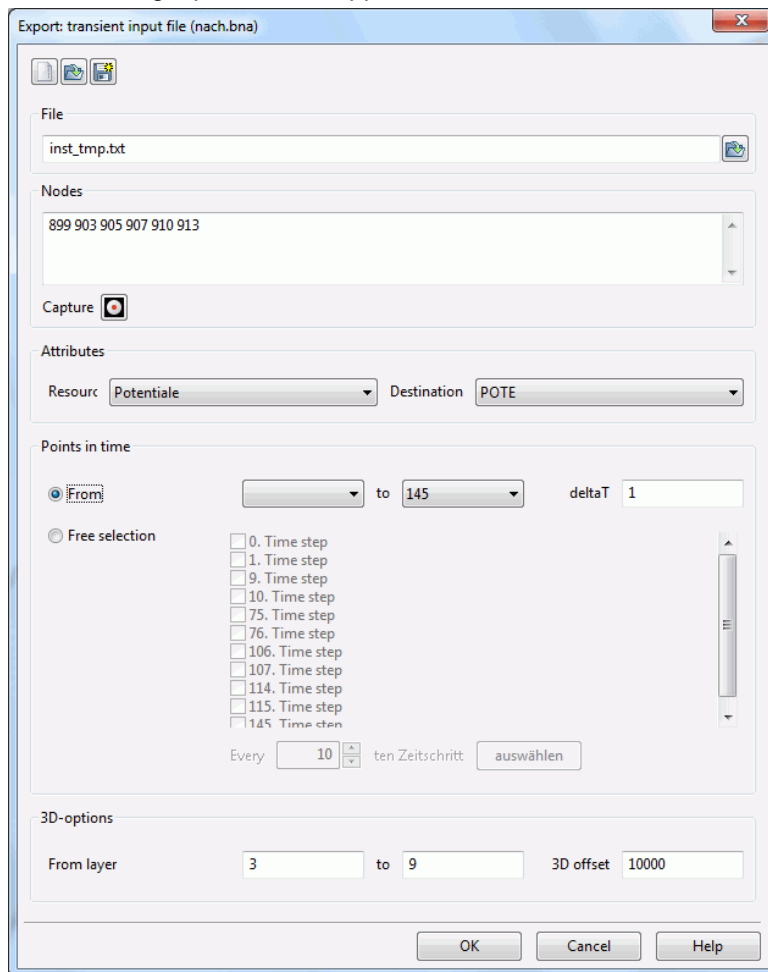## 7.9.1 Input parameters

The following input window appears:

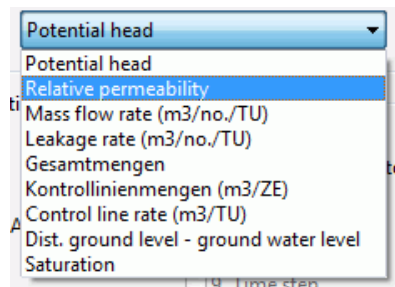Fig. 216: Input for saving transient result data

File

The name of the new transient input file is specified.
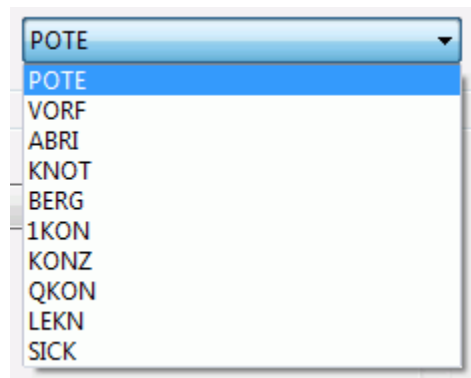
Nodes

The node numbers can be defined via the number list or selected interactively directly in the model by activating the button.

Attributes

First, the source data are set. These correspond to the output data after a transient calculation:

Then the source data is assigned to the target attribute. The user can choose from the possible transient node attributes:

Points in Time

It can be selected whether certain points of time from - to with a deltaT (e.g. delta T = 2, it is only the 1st, 3rd, 5th, etc. time step stored) are stored, or there is a free choice of the points in time to be stored.

3D-Options

In the case of a 3D model a selection of layers and a corresponding offset can be specified.

The buttons at the top of the input window allow resetting the input parameters ( ), opening an existing batch file ( ) or saving the current batch file under another name ( ).

7.9.2 Batch file for the output of transient data

It is possible to execute the *output of transient data* directly without using the dialog box. The default file name is *nachlauf.bna*. To use another batch file, type *nachlauf file name* (the extension ".bna" is automatically attached). Batch files can be created, viewed and modified with the help of any text editor.

A detailed batch file can be viewed in the online help. Similarly, there is a batch file for the output of transient data for further processing provided.

7.10 Output of result data for defined areas

Call via *File → Export → Clip result data....*

With this tool, the result data in the background files can be clipped for a particular model area. This is particularly important for using data in STRING.

7.10.1   Input parameters

The following input window appears:
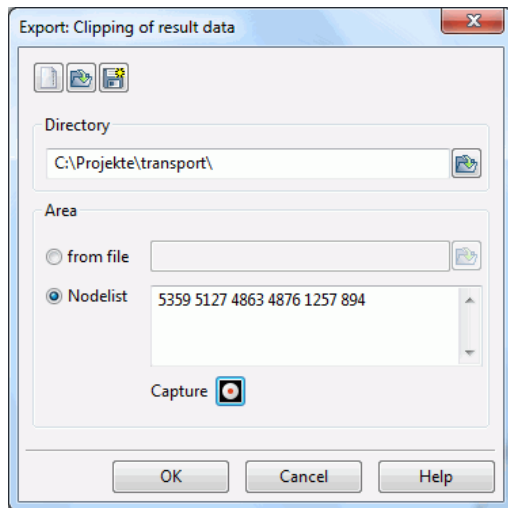
Directory

Here the directory is specified, in which the clipped background files will be stored. It may not be the original directory of the calculation. Thus, anytime there is a possibility to vary the clipping range on the basis of the original data.

Area

The clipping area can be determined either by choosing a shape file or interactively via a node selection.

The buttons at the top of the input window allow resetting the input parameters ( ), opening an existing batch file ( ) or saving the current batch file under another name ( ).

8 Plot Generation

In this chapter the different ways of the plot generation are described. The module is called via *File → Plot generation*.

Already after a *model checking*, it is possible to represent almost all the model data. Thus, a graphical control of input errors or a documentation of the data can be carried out.

After a successful calculation almost all modelling results are presented in different forms. Intercalculations of data are possible (also with the results of simulations in other directories) or the displaying of and intercalculating with measured data.

The generated plots are stored in *.plx-format, the individual information (contour lines, values, markings, etc.) is stored as a single layer and can thus be processed independently.

The following menu items are selectable:
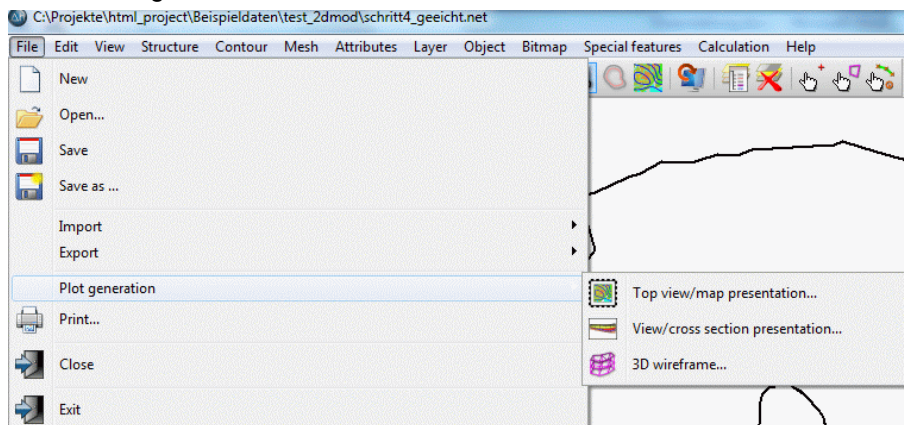
In contrast to the batch file structure of the other calculation modules the batch file of the plot generation is divided into command packets which are carried out one after another. Occurs for any reason during the plot generation an error tries the program to proceed with the next batch command. The batch commands are dependent on the type of data and the chosen representation.

They are described in chapter: "Batch file of the plot generation" (p. 381).

8.1 Top view/map presentation

After selecting this menu item the following input window appears:
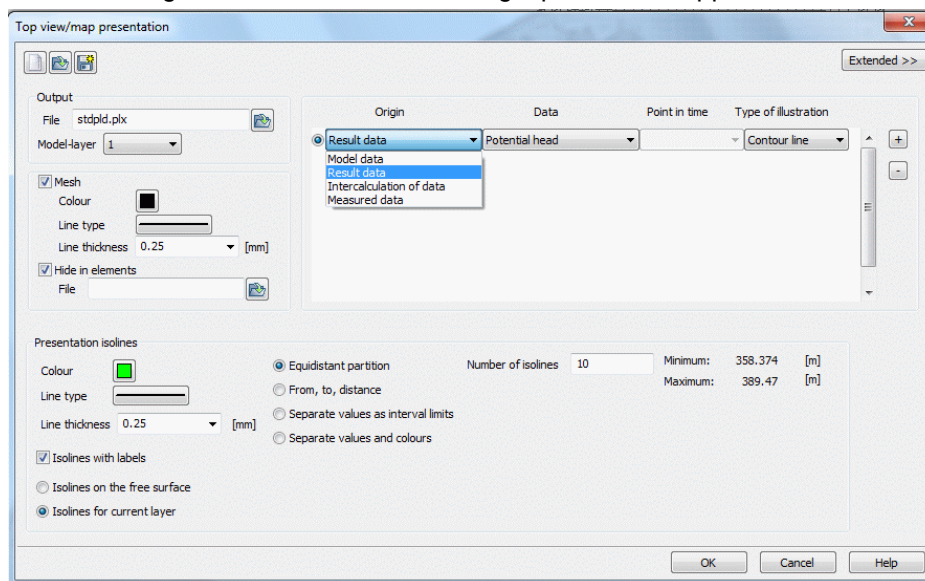
The buttons at the top of the input window allow resetting the input parameters ( ), opening an existing batch file ( ) or saving the current batch file under another name ( ).

Output

(batch command DATE)

The output file name is specified. Using the file selection window the name of an existing *.plx file can be loaded.

In a 3D model, the layer can be selected for which a top view should be created (horizontal section).

Mesh

(batch command NETZ)

By checking the box *Mesh*, the element mesh is shown in the plot. The colour, the line type and the line width can be selected.

Hide elements

In some models it may be useful not to present the results (eg, arrows or speed isolines of the potential) in certain areas, i.e. in a 3D model is the selected layer unsaturated or the layer is an aquitard. By specifying the appropriate element numbers in a separate ASCII file (Ausblend) the area to be hided is defined. The format of this file is equivalent to the input format of the model file (16x, 9 (I6, A1), A1) beginning in the first line without data identifier or comment line. The data value is arbitrary or may be absent (batch command AUSB).

Origin, data type, point in time, type of illustration

In this area of the input window, the individual layers of the plot are selected.
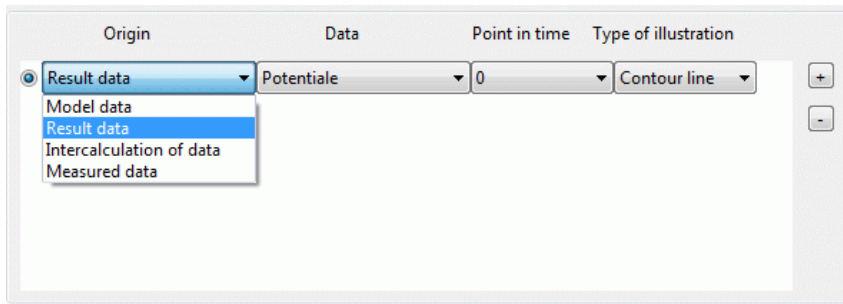
The order of the selection is not relevant. Before the plot file (*.plx) is created, the layers are sorted, so that surface representations are behind point or linear representations.

Represent several types of data simultaneously

The buttons at the right of the illustration type
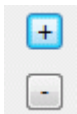
allow to add another layer (+ button) or remove a layer from the list (- button). In this case the active data is removed from the list. The radio button at the top of the line indicates which record is active.

In the column origin is the following data selection available:
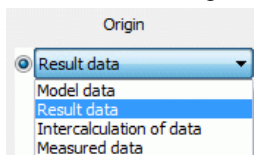
Depending on the selection in this field there are more input options for the other columns (data, point in time, type of illustration).

Origin: Model or result data

By selecting the model or result data as origin the data types stored in the background files are provided in the column data.

The types of model data result from the attributes existing in the model file, the types of result data result from the previously performed calculations.

After a steady state flow calculation this outcome of data is available:
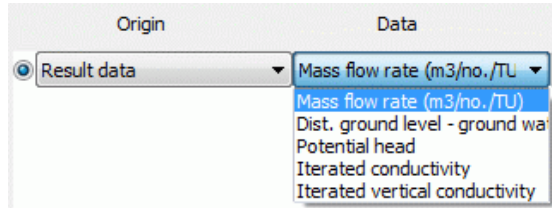
In a transient calculation the date is additionally set for an illustration.
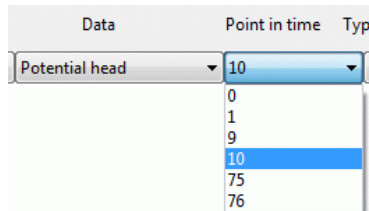
The selection of the data type results in appropriate display options.

For element data the following data illustration types are available:
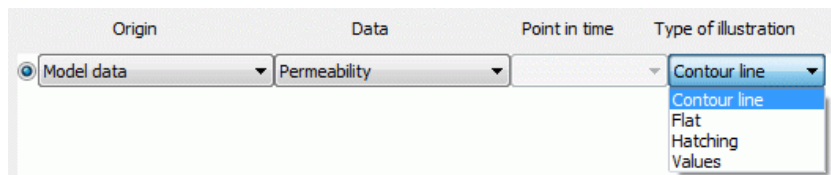
The illustration type *Hatching* is only possible for element data.

For node data, which are often defined globally, the following data illustration types are available:
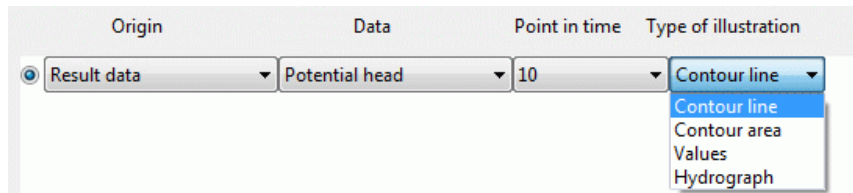
The illustration type *Hydrograph* is only possible after a transient calculation.

In contrast, for data at certain points such as sources/sinks or fixed potential heads are offered only the following illustrations:
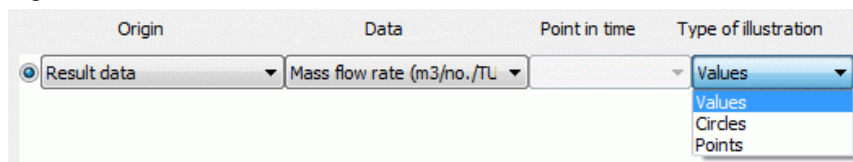
For specific data such as element-/ node numbers, mesh boundary, velocities or markings only the illustration type line is offered. It can be modified in the input block *Presentation*.

Details are described in chapter: "Types of illustration" (p. 361).

Origin: Intercalculation of data

Here it is possible, to offset data from the same or other simulation runs or from other layers against one another, and then to present the result (batch command RECH).

Additionally to the input block *Presentation* appear the following input possibilities:
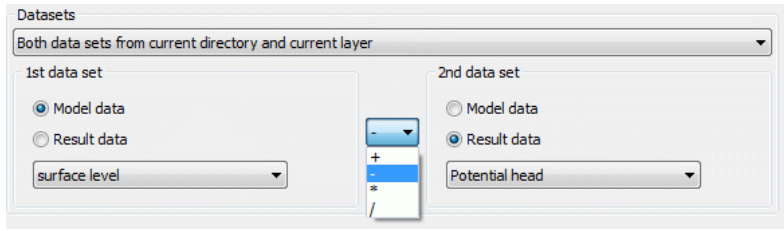
First thing is to decide on which data base the data are to be charged. The following combinations are possible:
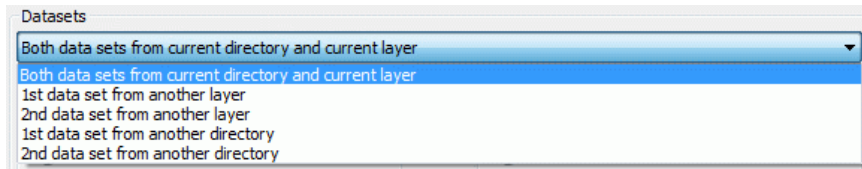
- Both datasets are found in the current directory, i.e. they result from the same calculation, and (in case of a 3D horizontal section) they both refer to the current layer.
- Both datasets are found in the current directory, i.e. they result from the same calculation, but one of the datasets (in case of a 3D horizontal section) can refer to another layer.
- Both datasets (in case of a 3D horizontal section) can refer to the current layer, but one of the datasets is found in another directory, i.e. results from another calculation.

Using a dataset from another directory, you must define the path from the current to the other directory in the corresponding text box (the directory selection can also be done in a file selection box).

Using a dataset from another layer (3D horizontal section), you must define the layer number in the corresponding text box.
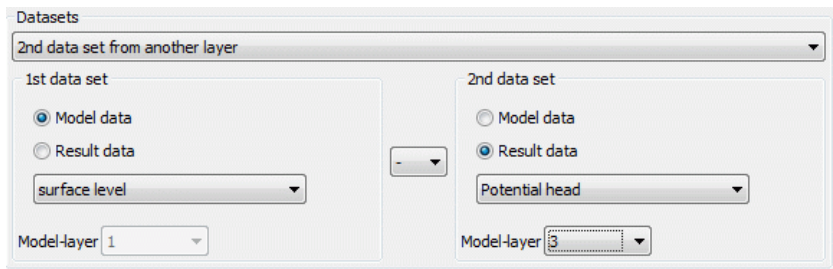
Selecting *Intercalculation of data* the fields *data* and *point in time* cannot be edited. This necessary information is specified in the input data block *datasets* for both species.

The left part of the shown dialog box defines the first data set, the right part defines the second one for the intercalculation. First, you must decide whether to use model data, results or transient results. Then you can select the appropriate data type. In case of a 3D model, all data refer to the layer which was defined in the input block *Output*, except when data is offset against data from another layer.

Available are the four fundamentals operations of arithmetic +, -, * and /.

Possible illustration types are: contour line, flat or values.
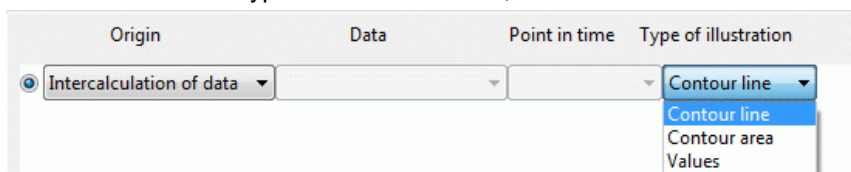
The illustration parameters are defined in the input block *Presentation*.

Origin: measured data

If measured data should be presented additionally to the input block *Presentation* appear the following input options: (batch command EICH):

At first, the file with measured data has to be entered or selected in a file selection box. The file must be formatted in the structure file format.

The following possibilities are available:

■ The values of the dataset are represented as circles:

The relevant nodes are marked by a + or another marker (see plogeo.ini commands EHGT nd ETYP). The corresponding value is printed at each node.

■ The values of the dataset are represented as bars:

The values of the measured data can be represented as bars. Among the colour and the line width the maximum size of the bar is defined:

■ The difference between the current data and any kind of existing data (input or result) is represented:

The differences are represented as bars (see figure above). The colour and the maximum bar size are determined by the user using the plogeo.ini command RADI.

The differences are represented as circles:

Without activating the checkbox „manual input" the maximum circle size is defined in the plogeo.ini-file (*Edit → Options*). In the batch file of the plot generation the following appears for EICH:

```
EICH   ## Measured data differences as circles
201 0 0 0         # Potentials
1 0.25 0 0        # Colour, line width, circle plot
gwmesspkte.txt    # file with measured data (structure file format)
```

If the manual input is activated, the user defines that a difference of e.g. 0.5 m/s corresponds to a circle radius of 1 cm. In the batch file of the plot generation the following appears for EICH:

```
EICH   ## Measured data differences as circles
201 0 0 0 1 0.5  # Potentials
1 0.25 0 0        # Colour, line width, circle plot
gwmesspkte.txt    # file with measured data (structure file format)
```

The values plotted at the measuring nodes are not plotted as real text but as poly lines. The default font-size for numbers is 0.18 cm. It can be changed using the menu *Edit → Options → Plot option → Size* (plogeo.ini command HWER).

During the offset of the data against other data an ASCII file *DiffEich.txt* is automatically created in which the x and y coordinates, the difference, the measured value and the calculated value in the format (6X, F10.2, F10.2 , F10.3, F10.3, F10.3) are saved. This file can be used to create a scatter plot (measured/calculated values as x/y graph) with Excel.

8.2 View/profile presentation

By selecting this menu item a vertical section can be created. The following input window appears:
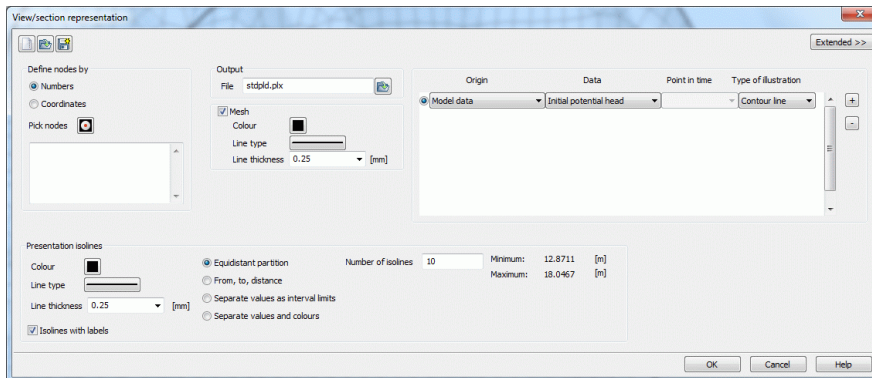
If data should be displayed in a vertical section, first the position of the section must be defined (batch command VERT).

Two possibilities are available:

■   Numbers

The position of the section is defined by node numbers. These can be selected interactively in the SPRING-surface (*Pick nodes*).

■   Coordinates

The position of the section is defined in x- and y-coordinates. The coordinate pairs can be entered directly in the appearing table or selected interactively in the SPRING-surface.
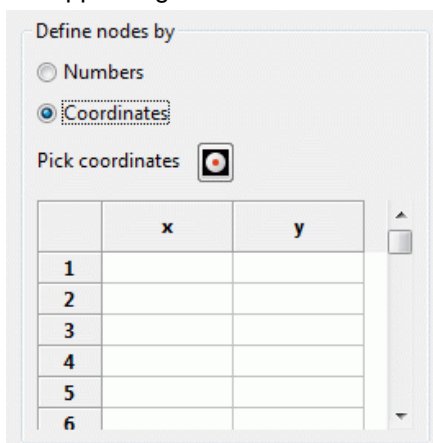
Remark:

If coordinates are used instead of node numbers SPRING looks for the nodes which are nearest to the selected coordinates. So the vertical section is placed through these nodes and NOT through the coordinates!

Furthermore, the input blocks correspond essentially to those described in the chapter "top view/map representation". There are minor restrictions on the various display modes observed. Thus, in node data, only the following illustration types:
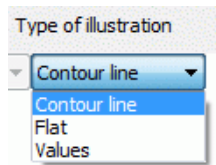
For element data, the illustration types are limited to these options:
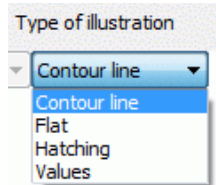
The illustration of measurement data is in a vertical section not possible:
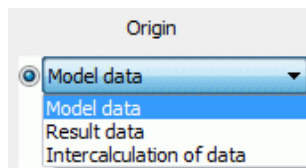
After creating a vertical section (*name.plx*) a structure file (*name.str*) is automatically created that contains the coordinates of the track. By importing the file or overlaying the structure in a top view the track of the vertical section can be marked.

8.3 Types of illustration

The following types of illustration are available:

- Line, p. 361
- Contour line, p. 363
- Flat, p. 367
- Hatching (only element data), p. **367**
- Values, p. 369
- Circles, p. **369**
- Points, p. 370
- Hydrographs (only node data), p. 370
- Data dependent features, p. 373

The specific input options for each illustration type, which depend in part on the selected data type are described in detail in the subsequent chapters. The "Data dependent features" are described in the chapter of the same name (p. 373).

8.3.1 Line

The illustration type *line* is not selectable, but set by default for the presentation of the mesh boundary, the node and element numbers, the markings, velocities and 2D fractures.

- Mesh boundary (batch command RAND)
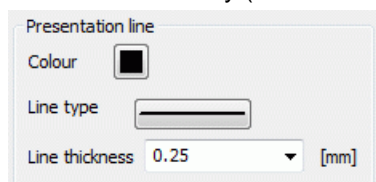
The colour, the line type and the line thickness are freely defined.

■   Element numbers, node numbers (batch command KNNR, ELNR)
The colour and the size can be defined.

■   Markers (batch command MARK)

☐ Only markings of current layer

This check box is only available in 3D models.

■   Velocities (batch command GESC)
When displaying the velocity, the colour, line type and line width can be freely determined.
If velocities of a 3D model should be presented in a vertical section the following input block appears:

Presentation line
Colour   ■ in the plane
Colour   ■ out off the plane
Colour   ■ into the plane
Line type   ────────
Line thickness   0.25   ▼   [mm]
☑ Hide above free surface
☐ Exaggerate vertical velocities

The vertical section through a 3D model presents the velocity arrows always with three colours. If the direction of the resultant of the velocity components deviates not more than 20 degrees from the sectional plane the first colour is used (in the plane). Points the resultant out of the sectional plane, the second colour is used. Points the resultant into the plane, the third colour is used.

■   2D fractures (batch command KL2D)

Presentation line
Colour   ■
Line type   ────────
Line thickness 1.1   ▼   [mm]

The colour, the line type and the line thickness are freely defined. Example of 2D fractures in a 3D model:
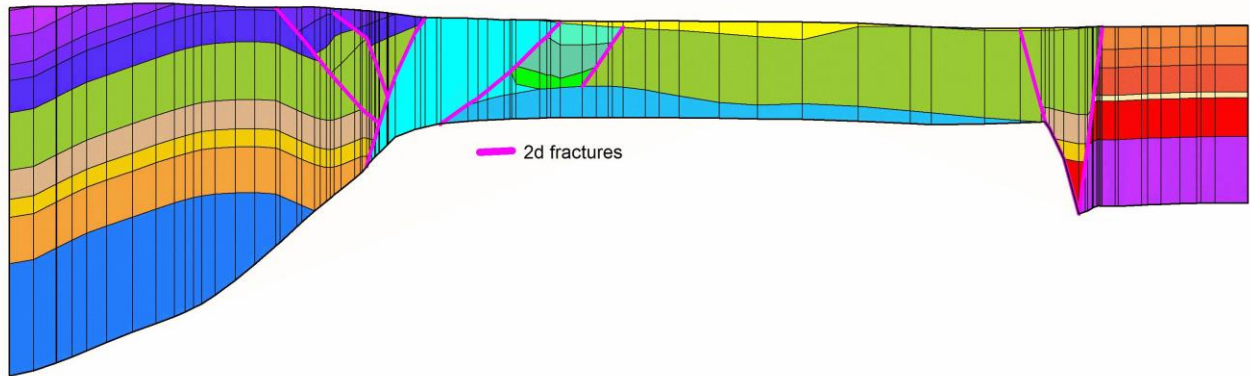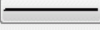
Fig. 217: 2D fractures in a vertical section

8.3.2 Contour line

Basically, the illustration of element data is offered as contour lines, but it is not always useful (eg K-values).

The left part of the input block the default settings for colour, line type and line thickness of the contour lines are changed arbitrarily. The labelling of the contour lines can be turned off by unchecking the box.

Otherwise, the contour lines are automatically labelled with the values.

The labels of the contour lines are not plotted as real text but as poly lines. The default font-size for these labels can be changed using the menu *Edit → Options → Plot option → Contour lines* (plogeo.ini command HISO).

Should data be plotted as contour lines (batch command ISOL), there are four possibilities to divide the values:

- Equidistant, p. 363
- From, to, distance, p. 364
- Separate values as interval limits, p. 364
- Separate values and colours, p. 364

In the right part of the input block, the extreme values of the selected data type are recorded for a better orientation.

8.3.2.1 Equidistant classification

The input block looks as follows:
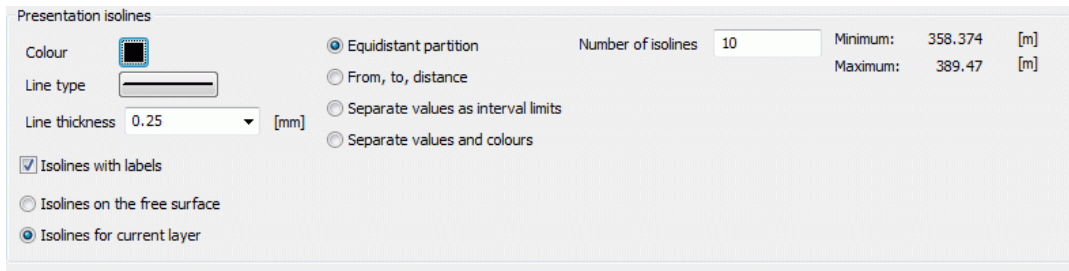
The number of desired isolines has to be entered. The interval from minimum to maximum is equally divided.

8.3.2.2 Classification from, to, distance

The input block looks as follows:
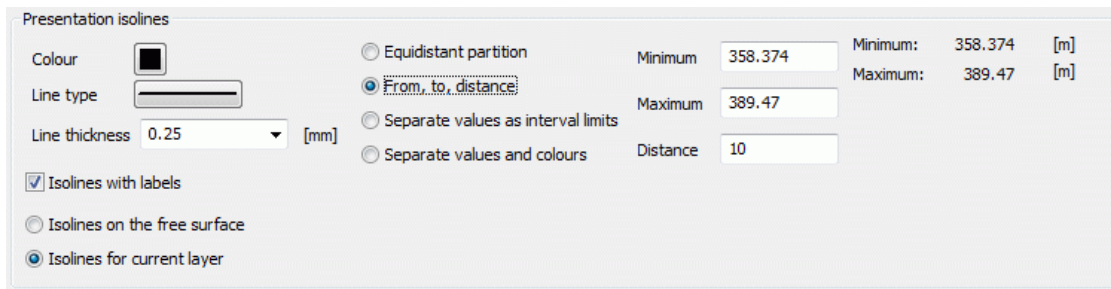
The classification from, to, distance requires the input of the minimum and maximum values and the desired distance of the contour lines in m.

8.3.2.3 Classification with separate values as interval limits

The input block looks as follows:
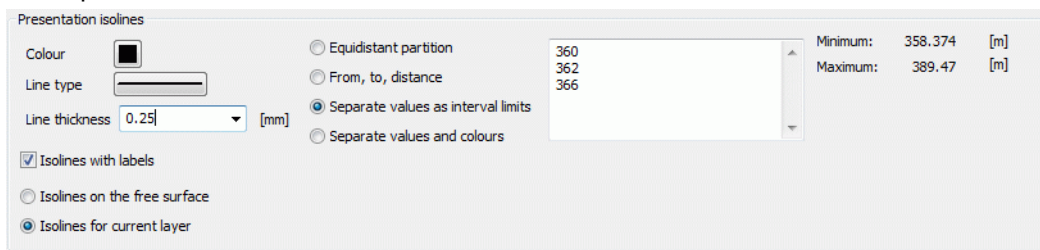
This classification requires the input of separate values as interval limits.

8.3.2.4 Classification with separate values and colours

The input block looks as follows:
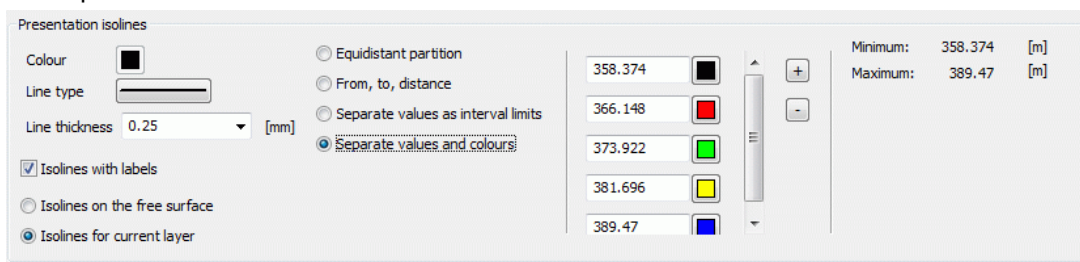
The classification with separate values and colours requires the input of the values for which the desired colour is then selected. The plus button allows to add an interval, the minus button allows to delete the last interval.

8.3.2.5 Special features

■ Hide isolines above free groundwater surface

Contour lines can be hidden in a vertical cross section or vertical model above the free surface.

☐ Hide isolines above free groundwater surface

■ Contour lines for potential heads in a 3D model

When representing contour lines of the potential heads in a 3D model it is is not necessarily known, for which horizontal layer the potential heads lie in the saturated zone.

Independent of that, in a horizontal section of a 3D model the potential heads can be represented at the free surface (first from above if there are more than one).

○ Isolines on the free surface

◉ Isolines for current layer
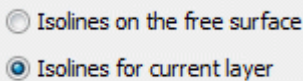
■ Presentation with a digital altitude model

Is a digital altitude model available the contour lines of the ground level (model data) and the distances between ground and water level (result data) can be plotted on the basis of the digital altitude model. The altitude model must have the structure format (ASCII).

When plotting the ground level instead of the attribute GELA the digital altitude model is plotted.

For the presentation of the distances between ground and water level (attribute FLUR) they are determined by offsetting the altitude model against the calculated potential heads. In both cases, an auxiliary mesh based on the altitude model and the FE nodes is created, so that the generation of such a plot takes some time, depending on the size of the data records and the available computing power.

It is possible to present the resulting auxiliary mesh (checking the box *Plot mesh*):

Digital altitude model

File: berg.txt ☑ Plot mesh Colour ▢
☑ Consider mining subsidence Line type ▭
Line thickness 0.05 ▼ [mm]
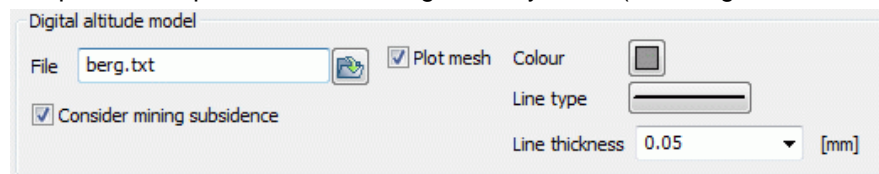
When terrain settings are present in the model file (attribute BERG), it is possible, to lower the digital altitude model with the subsidences of the model.

The advantage of calculation the distance between ground and water level based on the altitude model can be illustrated by the following example:

The presented water course (attributes VORF and LERA) is a water whose water bed should be raised within a regeneration. In the area of influence of the water course a rise of groundwater is expected. To preserve evidence therefore an assessment of the situation of the distance between the ground and water level is carried out for the current state (state of the model file).

Built-up areas are regarded as critical, which distance between ground and water level is already below 1.5 m in the current state.

To illustrate the improved accuracy obtained by calculating the distance between ground and water level with the altitude model, a plot of the distances was created after the flow calculation with and without use of the digital altitude model.
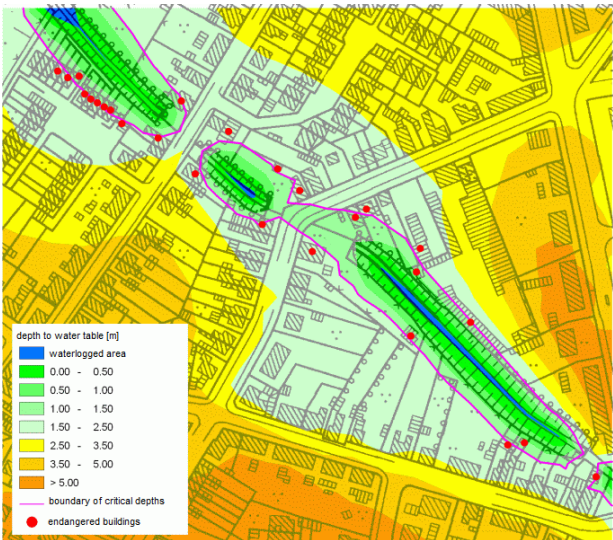
Fig. 218: Result without using the digital altitude model (26 buildings are affected)
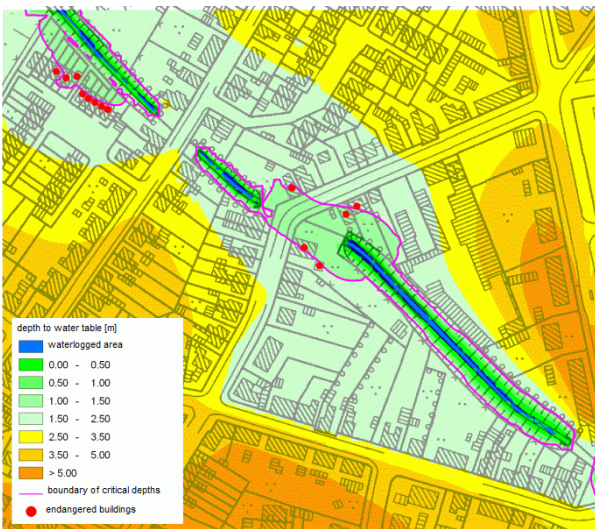
Fig. 219: Result with using the digital altitude model (13 buildings are affected)

On the basis of the digital altitude model only half of the buildings are classified as critical.

The enormous difference in accuracy is due to the by a multiple higher grid density of the altitude model (here: 5 x 5 m - grid) compared with the density of the FE mesh nodes whose approximate distance in this model is about 30-50 meters.
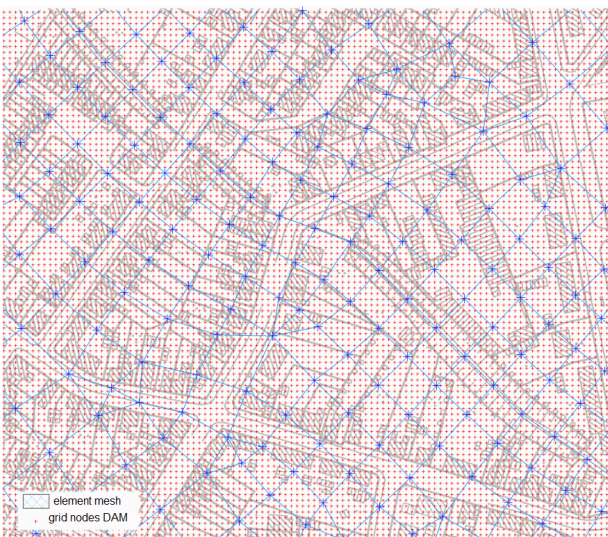
Fig. 220: Comparing the density of the mesh nodes (blue) to the density of the grid nodes of the altutude model (red)

8.3.3 Flat

A representation of flats means the colouring of flats between imaginary contours lines.

By clicking on the colour selection, different colour scales can be chosen to colour the intervals:

The classification of values is analogously to the intervals described in chapter: „Contour lines".

8.3.4 Hatching (only element data)

In the first sub-menu for setting the parameters for hatching (batch command SCHR) (see input block form below) is determined whether the hatch parameters should be entered precisely, or whether a simplified choice of parameters in the form of equidistant or by specifying a lower and upper value is desired.
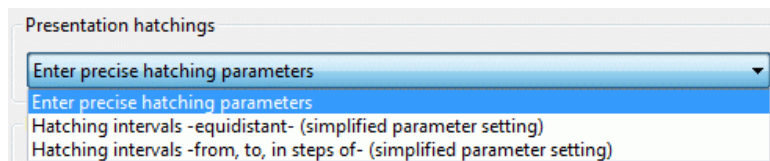
■   Precise settings of the hatching parameters

The selection of the exact input parameters offers the most opportunities to influence the hatching:
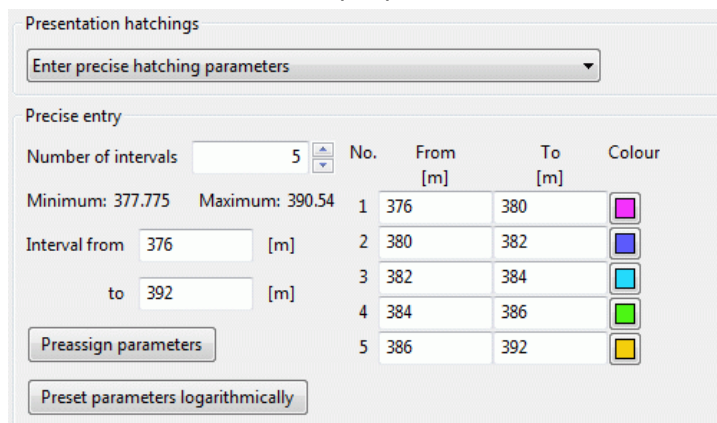
At first the number of intervals and the lower and upper boundary (from, to) are entered. After pressing one of the buttons *Preassign parameters* or Preset parameters logarithmically the values of the interval classes are set by default. They can be adopted or changed. It is possible to define up to 30 intervals.

With the SPRING menu item *Edit → Options → Plot options → In general* (plogeo.ini command FLAE) is defined, if the hatching is illustrated really as line hatching or if each element is coloured globally. Default is the areal colouring (parameter FLAE = 1).

In case of the line hatching the input of the hatching distance, the angle and the line thickness are additionally required.

- Equidistant (simplified parameter settings) choice of parameters

In the input block for the equidistant division the desired number of intervals is being queried. The interval between the minimum and maximum is evenly divided.

Line type, line thickness and the hatching distance are only relevant if the hatching is illustrated really as line (plogeo.ini command FLAE).

- From, to (simplified) choice of parameters

For the input from, to, a minimum and a maximum value are required as well as the distance of the hatching lines.

Line type, line thickness and the hatching distance are only relevant if the hatching is illustrated really as line (plogeo.ini command FLAE).

The following figures show the representation of a model lower boundary as line hatching (plogeo.ini parameter FLAE = 0, left) and areal hatching (plogeo.ini parameter FLAE = 1, right)
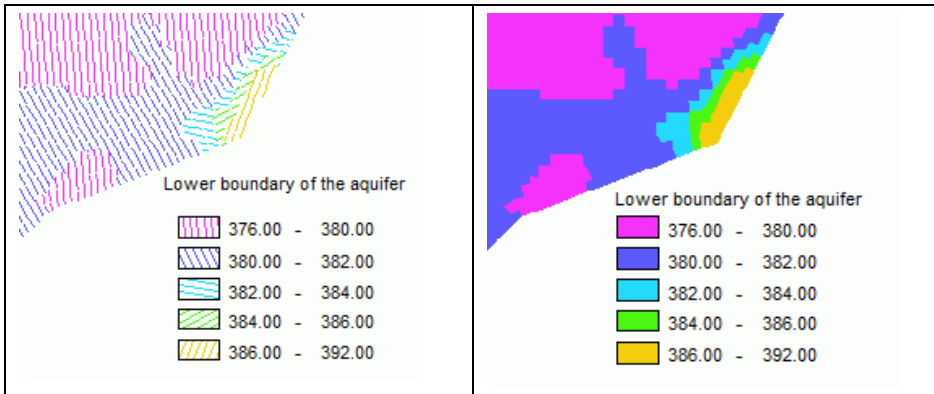
8.3.5 Values

The values plotted at the node and element centres are not plotted as real text but as poly lines. The default font-size for numbers is 0.18 cm. It can be changed using the menu *Edit → Options → Plot option → Size* (plogeo.ini command HWER).

In the presentation of quantitative data, stored in the background file in the unit m³/?/TU (TU = time unit of the model file), the conversion of data in different amounts and time units is offered.
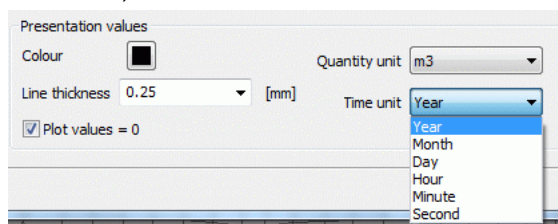
By checking the box *Plot values = 0* these values are explicitly shown, otherwise values equal zero are ignored.

8.3.6 Circles

The values plotted at the node and element centres are not plotted as real text but as poly lines. The default font-size for numbers is 0.18 cm. It can be changed using the menu *Edit → Options → Plot option → Size* (plogeo.ini command HWER).

In the representation of circles, the circle scale can be set manually defined by entering an amount for the circle size of 1.0 cm. This keeps e.g. representations of reaction volumes in different horizontal layers of a 3D model comparable.

If the circle scale is not defined manually SPRING calculates a maximum circle size for the largest value in the current plot. This maximum circle size can alternatively be defined in the menu *Edit → Options → Plot options → Sizes* (plogeo.ini command RADI).
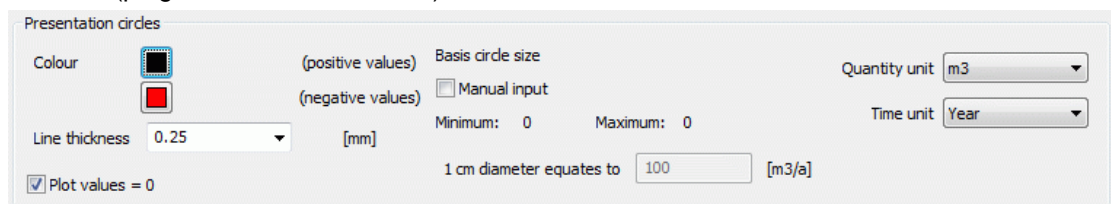
By checking the box *Plot values = 0* these values are explicitly shown, otherwise values equal zero are ignored.

8.3.7 Points

Instead of values or circles markers can be displayed at nodes by selecting this illustration type. Depending on the size of their attribute value they have a particular colour. This is especially useful for data at certain points e.g. withdrawals at wells (attribute KNOT), or leakage coefficients (attributes LERA, LEKN).
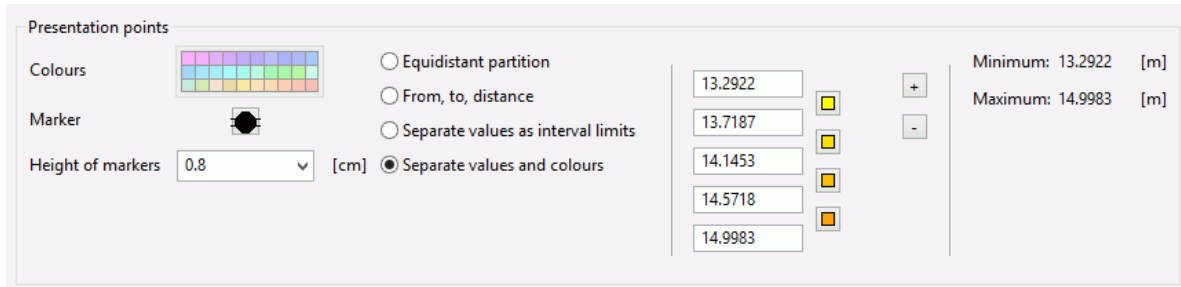
The following marker types are available:
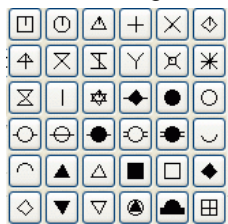
The classification of values is analogously to the intervals described in chapter: „Contour lines" (p. 363).

8.3.8 Hydrograph

A plot of hydrographs (batch command GANG) is possible only after a transient calculation.

After selecting the illustration type hydrograph the following input block appears:
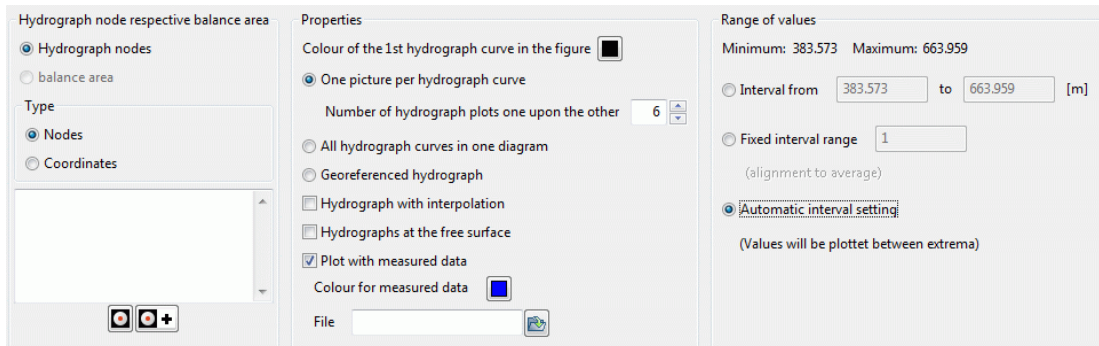
Hydrograph nodes respective balance area

If you select the *hydrograph nodes* the numbers of the nodes or coordinates are specified of which hydrographs should be plotted. If coordinates are chosen, the hydrograph for the nearest node is created.

The selection of the balance area is possible if nodes or elements have been grouped to balance areas (attributes BILK, BILE). Volume hydrographs are then given for the balance area numbers.

The input is unformatted with any number of values per line. The individual numbers must be separated by a blank or a comma. There is the possibility of activating the button 🔴 , to capture the nodes in the main window of SPRING. If more nodes or coordinates should be captured, the button 🔴 ✚ has to be selected.

In case of a3D model, only the nodes or elements of the top layer are captured. Node numbers of deeper layers are selected by adding the appropriate 3D offset.

Properties

◼ Colour

The colour of the first hydrograph is determined.

The hydrographs can be represented in three different ways.

◼ One picture per hydrograph curve

The specification defines how many individual hydrographs are presented in a column above each other in the resulting plot. For example, if there are 36 individual hydrographs and the number 9 is specified in this field there will be 4 columns with 9 hydrograph plots, one on top of the other.

◼ All hydrograph curves in one diagram

Should the hydrographs be plotted all in one diagram after checking the box appears the following input block:

As a horizontal line, for example, a (steady state) initial potential value of the first selected node is shown. For this, the *model data* is activated and the type of data: *initial potential heads* is selected.

The selected colour is used for the first hydrograph. For the following hydrographs the next colour in the colour palette is used.

◼ Georeferenced hydrographs

When this presentation is selected, at the selected nodes or elements are georeferenced hydrograph diagrams created. In addition, the background colour and a scaling factor can be specified. If "black" is chosen as background colour, the hydrograph gets a transparent background, if "white" is chosen as background colour, the hydrograph gets a solid background. The scaling factor affects the image size of the hydrograph diagram and should be no larger than 10.

Remark:

If a *.plx file that contains multiple georeferenced hydrograph curves is created this way, some processing steps have to be performed before the file can be overlaid. Each individual hydrograph is a separate *.plx map whose lower left corner corresponds to the position of the reference node (or the centre of the reference element) for which the hydrograph was created.

Such *.plx file can be transformed by the following procedure in a single map (in the SPRING mode plot file), then the *.plx file can be laid over a topographical map:

- Via the menu item *Map → Reduce → To a single map* all maps (= hydrograph curves) are reduced to one map. It does not matter which map is chosen as base.
- Via the menu item *Map → Edit → Adjust scale* both scale factors are set to the one hundred-fold scaling factor of the input window.
- Via the menu item *Map → Edit → Scale → Scale with attributes*, the map is scaled to the target scale, ie on the scale of the map, on which this file is overlaid later. Then the file must be saved.
- The so manipulated hydrograph curves can then be laid over the target map (*File → Import → Overlay*), that their left lower corner is georeferenced with respect to the target map.

- Interpolation

The individual points of the hydrograph are connected either in linear fashion (deactivated button) or with the help of a spline-interpolation (activated button). The usage of the interpolation can cause over- and underestimations in the case of large value changes.

- Hydrographs at the free surface

By activating this button the hydrographs of potential heads can be presented at the (upper) free surface instead of the selected layer (only 3D model).

- Plot with measured data
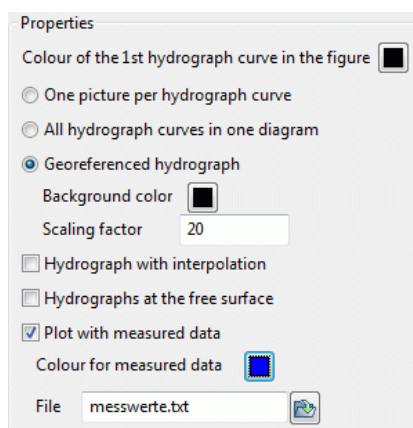
In the representation of the calculated curves measured nominal values (eg for a transient calibration) can be shown as additional hydrographs. The colour can be chosen freely.

The name of the file with nominal values can be entered manually or chosen from the file selection window. The format of the file is described in chapter: "File with hydrograph data".

Range of values

For the formatted representation of the curves the ranges of the values between which the hydrograph should be displayed are defined.

- Interval from, to

Here it is possible to determine the exact upper and lower limits of the representation.

If some values of the selected hydrograph nodes exceed or fall below these limits the program changes these limits automatically.

- Fixed interval range

It is possible to define a fixed interval, which is aligned at the mean value of the hydrograph.

- Automatic interval setting

Setting an automatic interval, each hydrograph is accurately represented between its extremes.

8.3.9 Data dependent features

For some data types a particular representation is given, which cannot be changed. These are nodes and element texts (KTXT, ETXT), equal potentials (GLEI) of the model data, calculated amounts of control lines and stream- and path lines of the result data.

8.3.9.1   Illustration of node and element texts

The data type node or element text specified by the attributes KTXT or ETXT can be selected for plotting in the model data. It appears the following input block:

The graphical display options for node and element texts are identical.

Besides the colour, line type and line thickness can be selected if the centre of the node or element is to be provided with a marker.

By entering an X and Y offsets, the distance of the text in cm from the centre of the node or element can be selected. The text size is arbitrary and, upon request, the text can be provided with a frame.

8.3.9.2   Marker at equal potential heads

The nodes to which the attribute GLEI (equal unknown potential heads) is assigned can be marked. The following input block appears:

Additionally to the markers the node numbers can be plotted. Their height can be entered regardless of the marker height.

8.3.9.3 Illustration of the groundwater recharge

If the groundwater recharge should be presented, appears, regardless of the illustration type an additional input block in which the unit of the amounts is defined.

Using the arrow keys, the unit of the quantities (cubic meters or litres) and the time unit (year, month, day, hour, minute, second) are determined arbitrarily. By activating one of the two check boxes, the default unit for the groundwater recharge can be selected. In that case the other options are disabled (grayed out).

8.3.9.4 Results of the control line calculation

If a calculation of control lines was done, the following input block appears when selecting control lines as plot data:

The results of a control line calculation are shown as a polygon with directional arrows and the corresponding amount of water (batch command KONT). The unit can be freely converted. The default is the unit m³ per time unit of the model file.

8.3.9.5 Illustration of stream- or path lines

After the calculation of path lines (flow calculation) or stream lines (data export) appears the following input block when selecting the appropriate type of plot data:

In addition to the graphical properties of colour, line type and line thickness can be selected if only the lines are shown (without markings) or markers (+) are put on these lines.

The markings can either be set after a distance [m] or time period. Here, the time can be specified in seconds, minutes, hours, days, months or years (batch command BAHN bzw. STRO).

8.3.9.6 Water course level

In a vertical section it is possible to present the height of the water course level (if the attribute VORF was defined in the model file) (similar to the presentation of the free surface).

Because the water course level is not defined at all nodes its presentation is only possible where the section leads through elements which have to all four nodes the attribute VORF assigned or where it leads along element edges or diagonal elements which have the attribute VORF assigned to these two nodes (batch command VORF). The colour, line type and line thickness can be arbitrarily determined.

8.4 Extended settings

By clicking on the button „Extended" an additional window appears in which more parameters can be set for the *Layout* and the *Text box*.

8.4.1 Layout

Some initialization parameters for the plot generation are set through the layout window:

Clip polygon

In case of a 2D model or a horizontal section of a 3D model, you can define a Clipping (batch command CLIP). Hence, the program uses for the plot the new window instead of the automatically computed extreme model coordinates. The coordinate frame will be adjusted. Parts of the plot which are outside the new frame are not plotted!

By selecting the button *Pick coordinates* two diagonally opposite points can interactively be caught in the SPRING-surface to determine the section.

Measuring unit

To each horizontal plot a scale bar (batch command MASS) can be added. The scale bar is printed centred above the text box. At different scales in x and y direction, the scale bar is based on the scale in x direction.

The default scale of the plot depends on the scale factor in the model file (attributeMASS).

To change the representation scale, use the text boxes. You can only super elevate the representation in a vertical section after having switched off the mode "x=y". Only then you can define the scale in y-direction as different from the scale in x-direction. The scale values are realized in the batch file by using the batch command SKAL (transfer into scaling factors).

Coordinate frame

Each horizontal section, vertical section or 2D plot has a coordinate frame. The user can choose between two different representations of the coordinate frame (batch command KOOR):

■   Presentation type „topographical map"

This coordinate frame type presents itself to topographic maps because they are usually presented in Gauss-Krüger coordinates.

■   Presentation type „basic coordinates"

This type is a simple frame with coordinate axes.
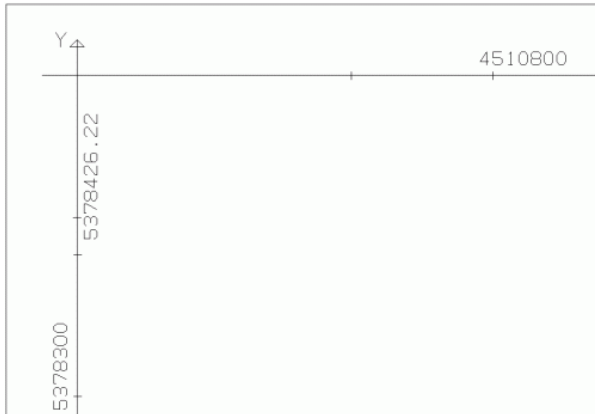
North arrow

The north arrow only makes sense in horizontal sections and 2D plots of a horizontal model. Rotated coordinate systems are not considered here.
(batch command NORD).

The north arrow is printed centred above the text box.

Fitting points

By checking the box *fitting points* coordinate crosses are represented in the coordinate frame. The default colour is black, it can be changed only directly in the plot via the menu item *Layer → Modify colours.*

Example for the representation of fitting points (green) in a plot:
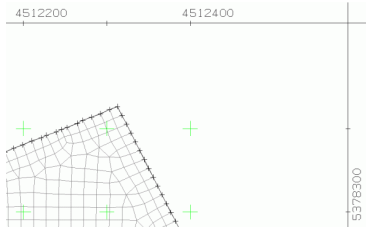
8.4.2 Text box

In this input window the text box is defined which appears in the lower right corner of the plot.

Choice of the text box type

Three text box types are available:

Selecting the **text box type 1** (12 cm x 11 cm) there is no further entry. It will be shown a pre-defined text field in the plot. If the attribute TEXT is defined in the model file, the text in it will be assumed in the text box type 1.

Example:

The model file contains:

```
TEXT : Label in the text box
    .50  3D Groundwater model
    .5     SPRING 4.1
    .5
    .5   Lower Boundaries
    .5     Hatching
    .3                      APPENDIX 2
```

The text box in the plot will look as follows:

```
┌─────────────────────────────────┐
│                                 │
├─────────────────────────────────┤
│                                 │
├─────────────────────────────────┤
│                                 │
├─────────────────────────────────┤
│    3D Groundwater model         │
│       SPRING 4.1                │
│                                 │
│    Lower Boundaries             │
│       Hatching                  │
│                                 │
│               APPENDIX 2        │
└─────────────────────────────────┘
```

In the **text box type 2** (default, 16 cm x 11 cm), the following text box is set by the user according to his ideas:

Text box 16X11

A (12cmx3cm)		B (2cmx1cm)	C (2cmx1cm)
		D (4cmx1cm)	
E (12cmx1cm)		F (4cmx1cm)	

First, in the text box area the field to edit (A-F) is selected:

Text box area

A (12cm x 3cm)
A (12cm x 3cm)
B (2cm x 1cm)
C (2cm x 1cm)
D (4cm x 1cm)
E (12cm x 1cm)
F (4cm x 1 cm)

Then, the type of data is associated with the area:

Data selection

Date
Date
filename without path
filename with path
PLX-file
Text

By the plus and minus button, all text fields can be edited. In the lower input range, the font, size, style, the angle and the offset for the label are set. Depending on the choice of data type a text input field or a file selection window appears.

X-Offset	0	[cm]	Y-Offset	0	[cm]
Text height	0.25	[cm]	Angle	0	[Â°]
Font	Helvetica		Type styles	Normal	
Colour	■				
Text	ground level				

Example:

The following input in the dialogue:

leads to this presentation in the plot:
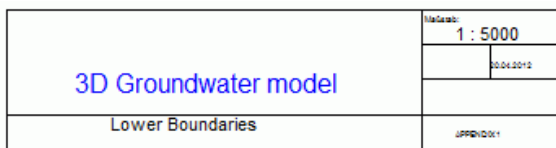
In the **text box type 3** (12 cm x 11 cm) the user has to enter the width and height of a placeholder for a (not yet created) text field. By specifying a width of for example 12 cm increases the distance between the coordinate frame of the representation to the outer plot border to 12 cm.

8.5 3D wireframe

The FE mesh of a 3D model can be represented three dimensional (batch command NE3D).

The following input window appears:
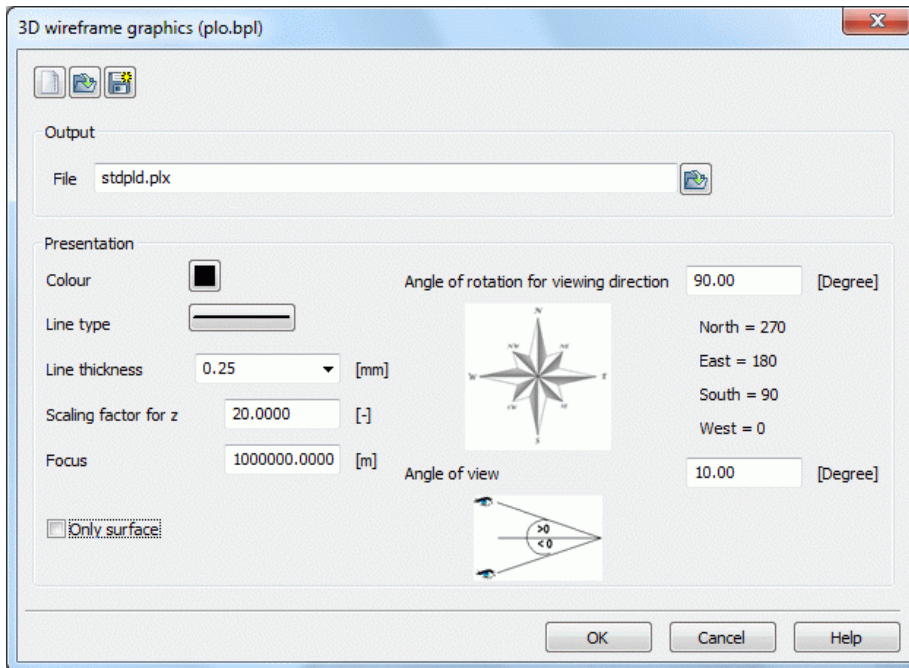
The buttons at the top of the input window allow resetting the input parameters ( ), opening an existing batch file ( ) or saving the current batch file under another name ( ).

Output

At this point the output file name is specified. By using the file selection window an existing *.plx file can be loaded.

Presentation

The mesh is presented in one colour: The plot parameters colour, line type and line thickness are to be specified.

■   Scaling factor for z

In general, the plot must be somewhat exaggerated, in order to compensate the differences between scales (extension in z-direction to expansion in (x, y)-direction).

■   Focus

The focus determines the view-coordinates, which look out on the model, based firmly on the centre of the area. If the distance is too small it can cause strange images (e.g. fish-eye perspective).

■   Only surface

By checking this box only the upper layer of the mesh is three dimensionally presented. By default the "sides" of the mesh are additionally presented. This is possible even for a 2D model with 3D subregion.

The focus point is defined by the following parameters:

■   Angle of rotation for viewing direction

Definition from which direction (0 to 360 degrees) one looks.

■   Angle of view

Angle in degrees from the horizontal (positive = view from above)

The following figure shows a horizontal model with a three-dimensional part as a 3D wireframe model:
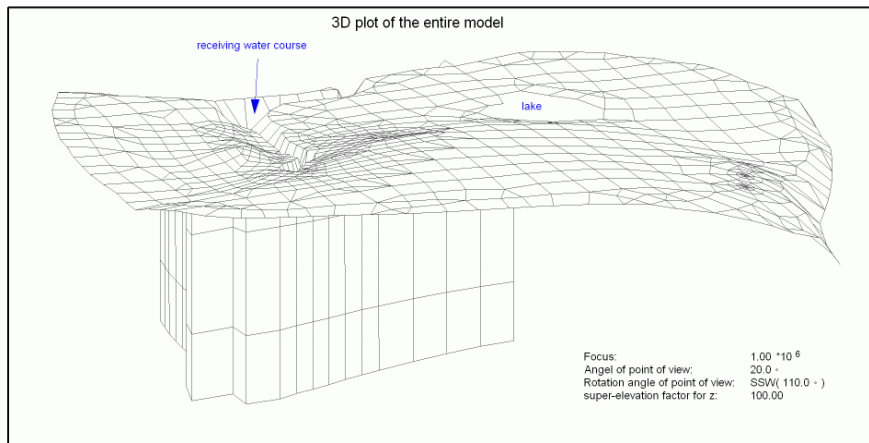
Fig. 221: 3D wire frame model

8.6 Batch file of the plot generation

It is possible to execute the plot generation directly without using the dialog box. The command *plogeo* starts the program using the default batch file *plo.bpl*. To use another batch file, type *plogeo file name* (the extension "*.bpl*" will be added automatically). Batch files can be created, viewed and changed with the help of any text editor.

The batch file of the plot generation is generally based on commands. The file contains command packages, which always start with a command identification line (the first 4 characters = name of the command in the line). The first four characters represent the command. If required, one or more input parameter lines in a defined format follow.

Basically, the order of command packets is not fixed. Only the first two command packets are equal in each batch file:

- Name of the output file *.plx (DATE)
- Type of presentation (i.e. HORI-zontal section)

The following chapters provide an overview of the various batch commands.

8.6.1 General remarks to the batch commands

Since some entries for several command blocks in the batch file are identical, they are described in advance in this chapter.

- Identification numbers

The data (input data and results) are specified using identification numbers (see → Appendix) and, if necessary, the time step (for transient data).

Knowing the identification number is not required in the processing of the user interface. If the batch file, however, is created or edited with an editor, the ID number is relevant for the user.

The ID number -99 is used as a flag for intercalculated data. The transformation flag in other units and the time step are not be interpreted (batch command RECH, p. 395).

- Transformation of data into other units:

For the representation of some data (especially rates), the transformation into other units is possible in the plot generation.

Mass rates [m³ /? /TU] are converted to other units for the plot by specifying a transformation flag. The question mark within the unit means "per node or element". The transformation flags are defined as follows:

Transformation flag	Unit in the plot
10	[m³/?/a]
11	[m³/?/m]
12	[m³/?/d]
13	[m³/?/h]
14	[m³/?/min]
15	[m³/?/s]
20	[l/?/a]
21	[l/?/d]
22	[l/?/m]
23	[l/?/h]
24	[l/?/min]
25	[l/?/s]

Additionally, recharge rates (identification 111) can be transformed from [m³/element/TU] to:

[mm/a] (transformation flag 1) i.e.

[l/s/km²] (transformation flag 2).

In the user interface corresponding controls are available for specifying the unit.

■   Potential heads in 3D models at the free surface

It is difficult to represent equal-value lines/domains for potential heads in a horizontal section of a 3D model, because you'll never know the correct layer. Using the transformation flag 9 automatically creates the potential heads at the free surface, independent of the specified layer.

■   Coordinates instead of node numbers:

Some commands require a list of node or element numbers. This is the case for:

Definition of a vertical section in a 3D model (VERT),

Definition of nodes/elements for hydrographs (GANG),

Definition of equal-value lines/domains through nodes (ISOL / FLAE).

The input of numbers can be substituted by the input of coordinates:

First, you must add a negative sign to the number of input nodes. Then, the program reads coordinates (x, y) and if necessary a layer number (nr) instead of the node numbers (maximum 10 per line). Each line requires one node (x, y and possible nr), the single values separated by a blank or a tab. As a result, the program searches for the node or element centre which lies next to the coordinates. Finally, the program reports the selected node or element numbers and their distance to the coordinates.

8.6.2 Name of the output file, new map

■   DATE

The command DATE sets the name for the output file (extension *. plx).

Input:

1. Line: filename

If no name for the output file is defined, a file with the name *stdpld.plx* is created.

■   NEUE

 A new map is initialized in the same plot. Further inputs are not required.

8.6.3 Definition of the general representation level

■ 2DMO

The command defines a horizontal plot in a horizontal model or a 2D plot in a vertical model. Further inputs are not required. The command is required only in combination with a different scaling (batch command SKAL).

This command in the batch file will be automatically inserted during the user's selection of the appropriate plot and model type in the user interface.

■ HORI

Specification of the layer number for a horizontal section through a 3D or a 2D/3D model.

Input:

1. Line: layer number (integer) greater or equal than 0

Layer numbers 0 and 1 provide node/element data representation for the 1st layer. Layer number 0 for a 2D/3D model represents only the 3D part. Layer number = 0 represents the whole 2D / 3D area. (Attention: There is one node layer more than there are element layers!)

This command in the batch file will be automatically inserted during the user's selection of the appropriate plot and model type in the user interface.

■ NE3D

Creates a 3D mesh plot

Input:

1st line: colour number, line thickness, line type

2nd line: type of representation: 1 = only 2D part of the mesh, 2 = 2D and 3D parts with boundary, 3 = only 3D part of the mesh with boundary

3rd line: direction of view (0 degree = W, 90 degree = S, 180 degree = O, 270 degree = N), point of view (>0 from above, <0 from bottom), scaling factor in z-direction and focus (>0, parallel projection for distance = infinite)

A fully 3D model makes no difference between representation 2 and 3!

■ VERT

Input of node numbers to create a vertical section through a horizontal model, a 3D model or the three-dimensional part of a 2D/3D model. This command in the batch file will be automatically inserted during the user's selection of the appropriate plot and model type in the user interface.

Input:

1st Line: Number of nodes (at least 2!)

From 2nd line:

　　Number > 0: node numbers (maximum 10 per line, separated by a blank or a tab)

　　Number < 0: number of following lines with x- and y- coordinates of one and if necessary layer number

In case of a 2D/3D model, the vertical section is allowed only through the three-dimensional part of the model! If the section does not lead along element edges but along straight connections between two nodes, some functions are not available anymore.

The vertical section through a horizontal model is only possible for the models where the aquifer bottom (UNTE) and the topmost boundary of the aquifer (OBER) (e.g. from GELA or EICH or OBER) are defined. If applicable, the ground level at the top (GELA) and the additional thickness for an impermeable layer (UNDU) are taken into account for the vertical aquifer structure in the mesh plot (batch command NETZ).

The following figure shows the vertical section of a horizontal model with an impermeable layer, whose lower edge (green) is shown in the mesh plot.

8.6.4 Plot parameters

■   AUSB

To hide parts of the presentation (contour lines, equal-value domains, etc) in certain elements in case of a horizontal plot.

Input:

1st line: name of the file containing the corresponding elements

The format of the file is described in chapter "file with elements to hide" , p. **31** . The program requires in this file only element numbers of the topmost layer (even in case of a section through deeper layers (3D)). You can create the list of elements with the help of SPRING and copy it from the input file into a new file. The command relates to all following commands where it is possible. It is switched off only with a new section (batch commands HORI,VERT,2DMO)!

■   CLIP

Clips a defined window in a horizontal plot (that means for plots of 2D models or horizontal sections through a 3D model)

Input:

1st line: xmin, ymin, xmax, ymax (separated by a blank or a tab) - corners at the bottom left and at the top right using global coordinates

All parts inside the coordinate frame are plotted, all parts outside the frame are clipped. Hence, the coordinate frame refers to the clip window coordinates.

■   SKAL

Edits the scales, globally and in directions x and y for horizontal and vertical plots

Input:

1st line: sca, scx, scy (separated by a blank or a tab)

(Input 0 causes the corresponding factor to be set on the default value 1.)

If you want to change the scales specified in the mesh file (identification MASS), you have to insert this command before the first plot command. Otherwise the scale command is ignored!

For a 3D model, if the definition of any section (using the batch command HORI or VERT) is missing, SPRING automatically executes the command HORI = 0 before the first plot command. In a similar fashion, the command 2DMO is executed for the two-dimensional case. Each of the commands HORI, VERT and 2DMO causes the scaling factors to be set back to the default value 1. Consequently, the scale command SKAL has only an effect on the scales after use of the commands HORI, VERT or 2DMO. So, if you want to use the scale command, you must insert the command HORI (= 0) or 2DMO before the scale command!

8.6.5 Text box types, coordinate frames, scale bar, north arrow

■   TFLD

Set the text box type and input of the labels

Input:

1st line Type number of the textbox (0 = type I, 1 = type II, 2 = type III), number of following lines in type II, width (xdist) and height (ydist) in type III (separated by a blank or a tab)

2nd line (only type II): description of the data field (TF3A-F)

3rd line:

in case of text input: xoffset, yoffset, font size, angle, colour number, two-digit flag for font and font style

    available fonts: Helvetica (1), Roman (2), Courier (3), Symbol (4)

    available font styles: normal (0), bold (1), italic (2), bold italic (3)

In the font "Symbol (4)" only the choice of font style "normal (0)" is possible.

in case of date input: DATUM, xoffset, yoffset, font size, angle, colour number, two-digit flag for font and font style

in case of file input (with or without path): DATEI, font size, angle, colour number, two-digit flag for font and font style

in case of input of PLX-file: PLX, xoffset, yoffset

The input of file (with or without the path) and PLX-file is simultaneously not possible.

By the following information for the text box in a PLOGEO batch file (extract):

```
TFLD   ## Text box type
1 8                             # Text box type II, number of following lines
TF3A                            # Text box position
    2 0.5 0.75 0 1 10 3D flow model            # #
TF3E                                           # Text box position
    3 0.5 0.5 0 1 10 lower boundaries          # #
TF3C                                           # Text box position
DATUM 0 0.3 0.25 0 1 10                        # #
TF3F                                           # Text box position
DATEI 1 0.3 0.4 0 5 12                         # #
```

this textbox is created:

■ KOOR

Defines the coordinate frame type:

Input:

1st line: Type number 0 or 1

Type = 0 means the simple coordinate frame

Type = 1 means the topographical coordinate frame

Examples for this are given in the "Extended settings", p. 375 . Hence, the command has to be located in the file before the first command which plots data. Each of the commands HORI,VERT,DATE and NEUE automatically disables the command KOOR and activates the default type II. This explains why KOOR, if it is to be used, should always be used after these commands!

■ MASS

Plots a scale bar above the text box:

No further input required!

The command defines a flag whether a scale bar is plotted together with the frame or not. This frame is plotted together with the first data representation. Hence, the command has to be located in the file before the first command which plots data. Each of the commands HORI, VERT, DATE and NEUE automatically disables the command MASS. This explains why MASS, if it is to be used, should always be used after these commands!

■ NORD

Plots a north arrow above the text box

An example for this is given in the "Extended settings", p. 375 . Hence, the command has to be located in the file before the first command which plots data. Each of the commands HORI, VERT, DATE and NEUE automatically disables the command NORD. This explains why NORD, if it is to be used, should always be used after these commands!

8.6.6 Mesh data plots

■   ELNR

Plots element numbers.

Input:

1st line: Colour number, line width

In case of a vertical section the numbers of the 2 neighbouring elements are plotted into the section element. Hence, if the section runs along element edges, the two numbers are equal.

If element numbers are to be plotted, they can be selected as a layer in the model data.

■   KL2D

Plots 2d fractures

Input:

1st line: colour number, line thickness, line type

A representation of fractures is only possible in the view/profile presentation.

■   KNNR

Plots node numbers

Input:

1st line: Colour number, line width

If the vertical section does not run along element edges, the plot contains two node numbers at the corners of the section elements. The section runs between these two nodes.

If node numbers are to be plotted, they can be selected as a layer in the model data.

■   KREU

Plots fitting points together with the coordinate frame

Without further input

Each of the commands HORI, VERT, DATEand NEUE automatically disables the command KREU. This explains why KREU, if it is to be used, should always be used after these commands! In the user interface is it possible under "Extended settings", p. 375 to choose if fitting points should be displayed.

■   MARK

Plots markers (identification MARKin the model file).

Input:

1st line: colour number, line thickness, line type

In the case of horizontal sections through 3D models, by default every marking is plotted independent on different layers. A negative colour number plots only the markings of the current layer. This function is not available for vertical sections of 3D models which do not run along element edges. In the case of a vertical section along element edges, the existing node markings are plotted.

The marking values 1.0, 2.0 and 3.0 create 'marking lines' instead of PLX markers. Their heights have constant values: MARK =1.0 (square) = 0.28 cm, MARK = 2.0 (octagon) = 0.32 cm and MARK = 3.0 (triangle) = 0.35 cm. Point markers where the value is greater than 100 are plotted as PLX markers with the default height of 0.32 cm. The height can be changed in the file plogeo.ini with the help of the initialization parameter HMAR, p. **82** .

■   NETZ

Plots the mesh.

Input:

1st line: colour number, line thickness, line type

■   RAND

Plots the mesh boundary

Input:

1st line: colour number, line thickness, line type

8.6.7 Data plots

- BAHN

Plots path lines

Input:

1st line: colour number, line thickness, line type

2nd line: marker type, z, ize (only marker type = 1)

Marker type = 0: no marking

Marker type = 1: markings after each 'z' time units, z = number of time units

Marker type = 2: markings after each 'z' meters, z = distance in meters

The selected time unit for the marker type = 1 is defined by the integer number ize:

ize = 0: seconds

ize = 1: minutes

ize = 2: hours

ize = 3: days

ize = 4: months

ize = 5: years

The program always plots all computed path lines which are stored in the binary files.

- ETXT

Plots text at elements (attribute ETXT)

Input:

1st line: colour number, line thickness, line type

2nd line: marker type, (x, y)-offset in cm, text height in cm, frame flag (in case of flag = 1)

In case of horizontal sections through 3D models, the program plots by default the element texts of all layers. A negative colour number limits the text plots to the current layer. The marker height is equal to the text height. The frame height is 1.5*text height.

The available marker types are equivalent to the regular PLX markers

Inputting '-1' causes no markers to be plotted.

This function is not available for vertical sections in 3D models.

- FLAE

Plots element wise or node wise data as equal-value domains

Input:

1st line: data identification number, transformation flag and time step

2nd line: first colour number, line thickness, line type

3rd line: which type of interval and further input depending on type:

   1 (= equidistant intervals) and the number of intervals

   2 (= from, to, distance) and min, max values, 4th line: distance

   3 (= input of certain values) and the number of values, 4th line: values

   4 (= contour lines through nodes) and the number of nodes, 4th line: node numbers. If the number of nodes is set negative, a pair of coordinates and the layer number (3D) can be entered from the 4th line on.

   5 (= input of certain values and colour numbers) and the number of nodes, 4th line: values, in 5th line: corresponding colour numbers

The input type 4 can only be entered via the batch file.

If more intervals are defined as colour shades are available the colours are repeated.

Area plots can only be created from data that are defined for all nodes or for all elements.

- FROB

Plots the free surface in vertical sections (3D models) and vertical models, manages the areas where equal-value lines (ISOL), equal-value domains (FLAE) and velocities (GESC) are not plotted above the free surface.

Input:

1st line: colour number, line thickness, line type and time step

The free surface is only plotted for a colour number > 0. Colour numbers = 0 switches on the flag for equal-value lines, equal-value domains and velocities. This ensures that they are not plotted above the free surface. A colour number < 0 switches the flag off again.

■   GESC

Plots velocities

Input:

1st line: colour number, line thickness, line type, time step and vertical flag (for vertical section through a 3D model or a vertical model)

In case of vertical sections through a 3D model or vertical models, the vertical flag defines whether the vertical velocity components are super elevated as the whole plot (vertical flag = 0) or not (vertical flag = 1).

Furthermore, please notice that the velocity representation in vertical sections of 3D models requires a total of three colours. The specified colour is used for velocities where the angle between the resulting velocity and the section plane is less than 20 degrees. The additional colour number +1 is used for velocities pointing out of the section plane; the colour number +2 represents the velocities pointing into the section plane.

■   GLEI

Plots markers at nodes with equal potential heads (GLEI-nodes)

Input:

1st line: colour number

2nd line: marker type, marker height

3rd line: Flag for the node number (1 = with, 0 = without), height of the node numbers

■   ISOL

Plots equal-value lines of node wise or element wise data.

Input:

1st line: data identification number, transformation flag and time step

2nd line: colour number, line thickness, line type

3rd line: which type of interval and further input depending on type:

  1 (= equidistant intervals) and the number of intervals

  2 (= from, to, distance) and min, max values, 4th line: distance

  3 (= input of certain values) and the number of values, 4th line: values

  4 (= contour lines through nodes) and the number of nodes, 4th line: node numbers. If the number of nodes is set negative, a pair of coordinates and the layer number (3D) can be entered from the 4th line on.

  5 (= input of certain values and colour numbers) and the number of nodes, 4th line: values, in 5th line: corresponding colour numbers (number of colours has to be the same as the number of values)

  6 (= input of certain values, colour numbers, line thicknesses and line types) and the number of values, in the 4th line: value, colour number, line thickness, line type

The input types 4 and 6 can only be entered via the batch file.

The plotting of equal-value lines is only available for data which exist for all nodes or elements!

■   KONT

Plots the computed rates of control lines

Input:

1st line: colour number, line thickness, line type, transformation flag and time step

This function is not available for vertical sections!

■   KREI

Plots circles at nodes or in elements

Input:

1st line: data identification number, transformation flag and time step

2nd line: colour number, line thickness, scaling flag and scaling value

The circles can be created for either node or element data. A circle plot is only useful for data which do not exist for all nodes or elements. The positive values are represented by the specified colour. The colour number +1 is used for the negative values.

A negative colour number plots only values which are not equal zero.

Selection of the scaling flag 1 requires a scaling value > 0 which corresponds to a circle radius of 1.0 cm.

With no scaling flag or selection of 0 the maximum radius is automatically determined from the plot size. Otherwise you can explicitly define the maximum radius using the initialization parameter RADI in the file plogeo.ini.

Identification number -99: for calculated data. The transformation flag and the time step are ignored. See batch command RECH!

- **PUNK**

For plotting of points (or markers) at nodes or in elements

Input:

1st line: data identification number, 0, time step, 0

2nd line: Marker type, marker height

3rd line: Dummy, number of intervals

from 4th line: interval boundaries (at maximum 5 per line)

from 5th line: colours for the intervals (at maximum 5 per line)

At the moment the values equal zero are not interpreted. Only node or element data can be plotted as points.

- **KTXT**

Plots node texts (attribute KTXT)

Input:

1st line: colour number, line thickness, line type

2nd line: marker type, (x, y)-offset in cm, text height in cm, frame flag (in case of flag = 1)

In case of horizontal sections through 3D models, the program plots by default the node texts of all layers. A negative colour number limits the text plots to the current layer. The marker height is equal to the text height. The frame height is 1.5*text height.

The available marker types are equivalent to the regular PLX markers

Inputting markertype = -1 causes no markers to be plotted.

This function is not available for vertical sections in 3D models.

- **SCHR**

Creates hatching (see p. 367) or rather an area plot of element data

Input:

1st line: data identification number, transformation flag and time step

2nd line: type and definition of the hatching intervals:

   0 (= detailed input) and number of intervals, in the following lines: For each hatching interval a line is read with min, max, distance of the lines, angles, colour, line thickness and line type

   1 (= short input: equidistant division) and, in the following line: number of the colour palette, line thickness, line type, distance between the lines

   2 (= short input: from, to, distance), in the 3rd line: from, to, distance, in the following line: number of the colour palette, line thickness, line type, distance between the lines

If more intervals are defined as colour shades are available the colours are repeated.

If no hatching distance or the distance = 0 defined, the hatching distance is automatically set to 0.2 cm.

Hatching plots are only available for element plots!

- **STRO**

Plots stream lines.

Input:

1st line: colour number, line thickness, line type

2nd line: marker type, z, ize (only marker type = 1)

Marker type = 0: no marking

Marker type = 1: markings after each 'z' time units, z = number of time units

Marker type = 2: markings after each 'z' meters, z = distance in meters

The selected time unit for the marker type = 1 is defined by the integer number ize:

ize = 0: seconds

ize = 1: minutes

ize = 2: hours

ize = 3: days

ize = 4: months

ize = 5: years

The program always plots all computed stream lines which are stored in the binary files

- VORF

Plots the water course level (VORF) of a receiving water course in vertical sections through horizontal or 3D models.

Input: colour number, line thickness, line type

- WERT

Plots values at nodes or in elements.

Input:

1st line: data identification number, transformation flag and time step

2nd line: colour number

The function is only available for node wise and element wise data.

A negative colour number causes only values not equal 0 to be plotted.

8.6.8 Transient data

- GANG

Creates hydrograph plots

Input:

1st line: data identification number, flag for index, free surface and transformation

2nd line: number of elements/nodes for the hydrograph plot

3rd line: node or rather element numbers (maximum 10 per line, separated by a blank or a tab) or rather (number < 0) one node per line (x- and y-coordinates, layer number nr). If more than one line is needed, the next lines shift accordingly.

4th line: colour number for each first curve in a hydrograph plot, number of plots over each other

5th line: minimum/maximum values, size of interval (if min=max, the program uses the global values)

6th line: flag for spline-interpolation (1= with interpolation, 0 = without interpolation)

7th line: flag for all hydrograph curves into one diagram? (1= yes, 0 = no)

8th line: flag for nominal values (1 = with nominal values, 0 = without nominal values), background colour, scaling factor

With nominal values:

9th line: colour number for nominal values

10th line: name of the file which contains the nominal values

Generally, you can represent only transient data (identification numbers between 400 and 500) as a hydrograph plot.

If curves are presented with nominal values the calculated and the nominal values are recorded in an ASCII file (C <Node/element number>. txt) when the plogeo.ini command EDIF = 1 is set.

Curves with an additional horizontal line for reference data:

For the case in which each curve is plotted into its own map (7th line of the GANG-command "all time-series curves into one diagram?"), an additional horizontal line can be plotted into each map (colour number +1). The level for this line is taken from a second identifier at this node/element. Hence, the second identifier must refer to steady-state data. Possibilities are input data (e. g. ground level - identification 85, z-coordinates - identification 13) as well as results at the end of the transient calculation (e. g. potential head - identification 306).

8.6.9 3D Data

- PL3D

The command PL3D creates a 3D plot, when the user writes himself the corresponding batch file. Below are several examples of such files.

Input (separated by a blank or a tab):

1st line: direction of view (0 degree = W, 90 degree = S, 180 degree = O, 270 degree = N), point of view (> 0 from top, < 0 from bottom), scaling factor in z-direction and the focus (> 0, parallel projection for focus = infinite)

2nd line: nx, ny (grid division: number of intervals in x and y direction)

3rd line: data identification number for the basic level of the 3D representation, time step and layer number

4th line: grid level ($\leq$ data minimum), colour number, line thickness, line type

5th line: data identification number for grid plot, time step, layer number:

Flag = 0: no grid!

Flag = -1: monochrome grid (above the basic level), the following line contains colour number, line thickness, line type

Flag > 0: multi-coloured grid for the defined values (at the basic level). The following lines contain information on the colour intervals:

Colour number (see FLAE), line thickness, line type

Which and how many equal-value lines:

1 (= equidistant intervals) and number of intervals

2 (= from, to, distance): the following line contains min, max, distance

3 (= input of certain values) and the number of values, the following line contains values (maximum 10 per line, separated by a blank or a tab)

4 (= equal-value lines through nodes) and the number of nodes, the following line contains node numbers (maximum 10 per line, separated by a blank or a tab) or rather (number < 0) one node per line (x- and y-coordinates, layer nr for 3D)

6th line: Add equal-value lines to the grid? Data identification number, time step, layer number

Flag = 0: no equal-value lines!

Flag > 0: coloured equal-value lines for the defined values (at the basic level), the following lines contain: first colour number, line thickness, line type

Flag < 0: monochrome equal-value lines

Flag > 0: multi-coloured equal-value lines (starting colour number: see FLAE)

Which and how many equal-value lines:

1 (= equidistant intervals) and the number of intervals

2 (= from, to, distance): the following line contains min, max, distance

3 (= input of certain values) and the number of values, the following line contains values (maximum 10 per line)

4 (= equal-value lines through nodes) and the number of nodes, the following line contains node numbers (maximum 10 per line, separated by a blank or a tab) or rather (number < 0) one node per line (x- and y-coordinates, layer nr for 3D)

7th line: Colour number, line thickness, line type

Plot linear markings into the grid?

Colour number = 0: no markings in the plot (value=10)

Colour number < 0: plot only linear markings of the current layer

Colour number > 0: plot all linear markings

Examples:

To create 3D plot with the following PL3D blocks, the corresponding *.bpl file has to contain the two lines

```
DATE   ## file name of the plot file
bild.plx
```

at the beginning of the file and the line

```
# ---> Plot Ende!
```

at the end of the file.

- 3D plot with a monochrome grid at the level of the potential head and coloured equal-value lines for the concentration:

```
PL3D                 # 3D-plot
135. 15. 10. 1.e+6 #direction of sight: SW, angle of direction, elevation factor z,focus
100 100              # 100x100 raster
201                  # Raster from height of potential heads
0. 3 0.25 1          # Raster level at potential=0. in colour 3
-1                   # monocoloured raster
1 0.25 1             # colour 1, line width, line type ...
203                  # Isolines for concentartions (203),
0 0.25 1             # different colours! Colour to begin with=0
2                    # Isolines
4. 16. 1.            # from, to distance
1 0.5 1              # Markers in colour 1, line width 0.5, line type 1
```
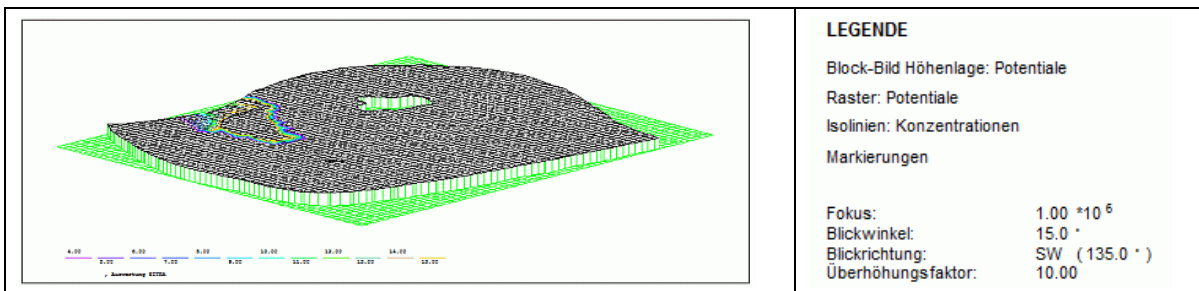

- 3D plot at the level of the potential head with a multi-coloured grid for the potential head and monochrome equal-value lines for the concentration:

```
PL3D                 # 3D-plot
135. 15. 10. 1.e+6 # direction of sight: SW, angle of direction, elevation factor z,focus
100 100              # 100x100 raster
201                  # Raster from height of potential heads
0. 8 0.25 1          # Raster level at potential=0. in colour 8
201                  # Raster for potentials
4 0.25 1             # multi-coloured raster, starting colour 4 ...
2                    # Raster colour interval for potentials (201)
13. 18. 1.           # from, to, distance
-203                 # Isolines for concentrations (203),
2 0.25 1             # monochrome (colour 2)
2                    # Isolines
4. 16. 1.            # from, to distance
1 0.5 1              # Markers in colour 1, line width 0.5, line type 1
```
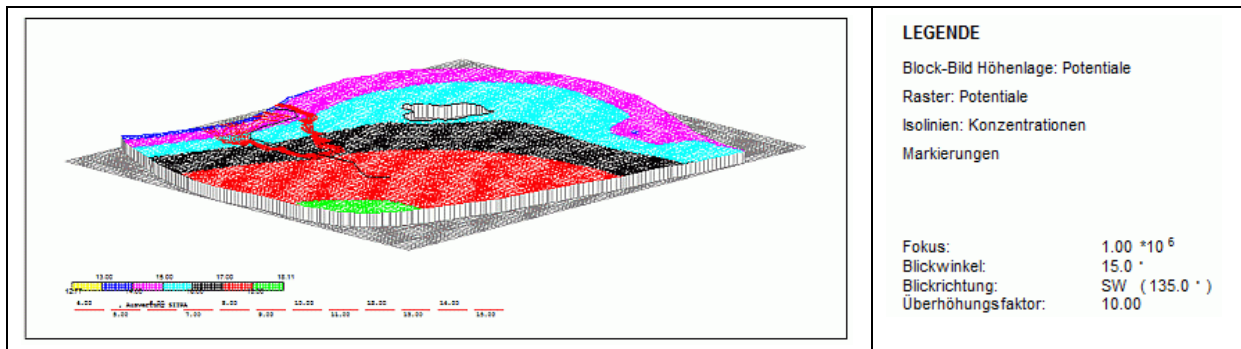
- 3D plot (without grid) with coloured equal-value lines for the concentration at the level of the potential head:

```
PL3D                # 3D-plot
135. 15. 10. 1.e+6  #direction of sight: SW, angle of direction, elevation factor z,focus
100 100             # 100x100 raster
201                 # Raster from height of potential heads
0. 8 0.25 1         # Raster level at potential=0. in colour 8
0                   # no raster
203                 # Isolines for concentrations (203),
0 0.25 1            # different colours! Starting colour=0
2                   # Isolines
4. 16. 1.           # from, to, distance
1 0.5 1             # Markers in colour 1, line width 0.5
```
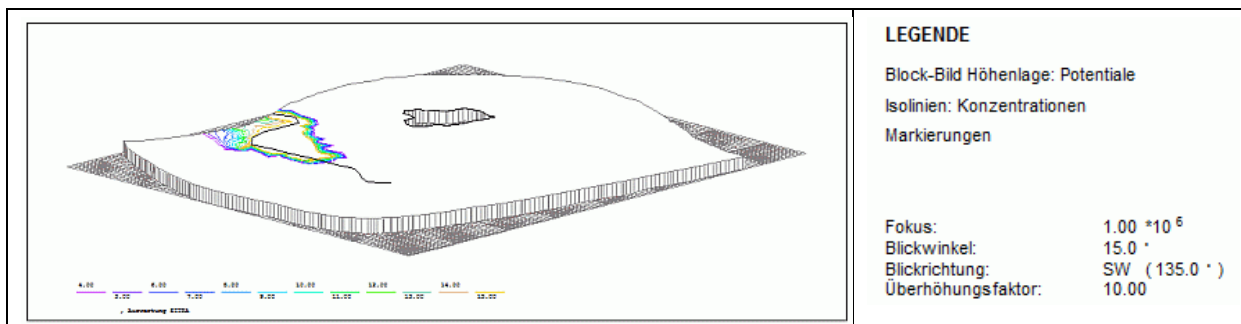

- 3D plot at the level of the potential head with a multi-coloured grid for the potential head, without additional equal-value lines:

```
PL3D                # 3D-plot
135. 15. 10. 1.e+6  # direction of sight: SW, angle of direction, elevation factor z,focus
100 100             # 100x100 raster
201                 # Raster from height of potential heads
0. 8 0.25 1         # Raster level at potential=0. in colour 8
201                 # Raster for potentials
-1 0.25 1           # multi-coloured raster, in gray (starting colour -1 ...)
2                   # Raster colour interval for potentials (201)
13. 18. 1.          # from, to distance
0                   # no isolines
1 0.5 1             # Markers in colour 1, line width 0.5
```
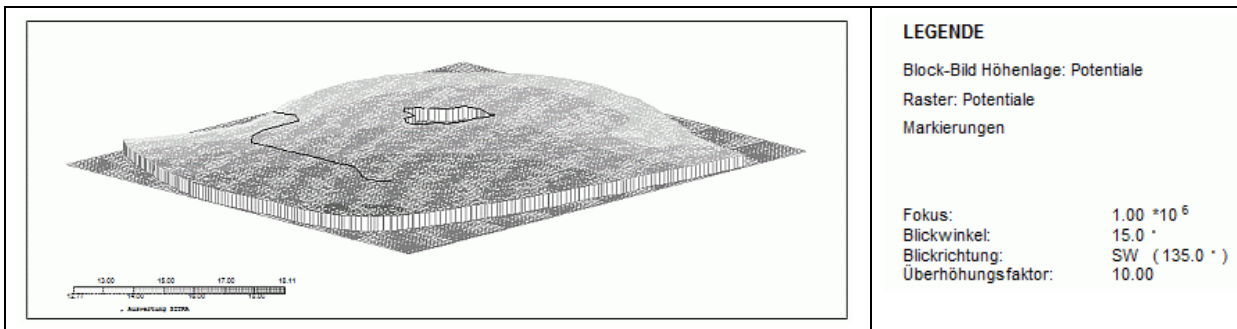
- 3D plot at the level of the potential head after 10 days with a multi-coloured grid for the concentrations and multi-coloured equal-value lines for the concentrations:

```
PL3D ## 3d-Schaubild
80.0 25.0 50.0 1000000.0  # direction of sight: SW, angle of direction, elevation factor
z,focus
60   60                   # 60x60 raster
301  10 1 0               # Height: Potentials (time step 10), layer 1, Index
1.00 8 0.25 1             # Raster level at potential=1., colour 8, ...
303 10 1                  # multi-coloured raster: conc. (time step 10), layer 1
0 0.25 1                  # Starting colour 0, line width, line type
3  8                      # single values -> number of values
1.00000  3.00000   5.00   8.0000  10.0000 #
12.0000 15.00000 20.00000                 #
303  10 1                  # multicoloured isolines: conc. (ts 10), layer 1
0 0.30 1                   # starting colour, line width, line type
3 6                        # single values -> number of values
1.00000  3.00000  5.00000 10.00000 12.00000 #
20.00000                                   #
1 0.25 1                   # Markers in colour 1, line width 0.25
```
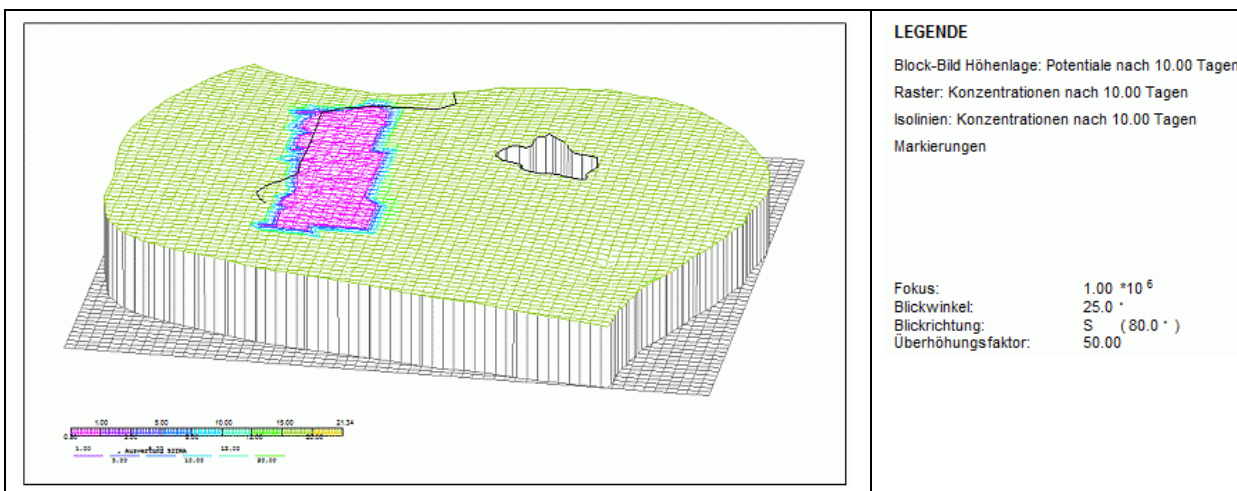

8.6.10   1D Fracture data

The following commands are accessible only via the batch file parameters.

- KL1D

Plots 1D fractures

Input:

1st line: colour number, line thickness, line type

- NR1D

Plots fracture numbers at 1D fractures

Input:

1st line: Colour number

- PL1D

Creates a coloured representation of the 1D fractures depending on the 1D fracture data

Input (separated by a blank or a tab):

1st line: identification number for the 1D fracture data

2nd line: colour number, line thickness, line type

3rd line: definition of values for the colour intervals:

> 1 (= equidistant intervals) and the number of intervals
>
> 2 (= from, to, distance) and min, max values, 4th line: distance
>
> 3 (= input of certain values) and the number of values, 4th line: values (maximum 10 per line)

- WE1D

Plots values at 1D fractures

Input:

1st line: identification number for the 1D fracture data

2nd line: colour number

Negative colour number causes only values not equal 0 to be plotted.

8.6.11   Various

- EICH

Plots measuring data or differences to measuring data

Input:

1st line: data identification number for differences, transformation flag and time step

2nd line: colour number, line thickness, flag for the representation of the differences:

> 0 = circle representation
>
> 1 = bar representation, scaling parameter for the bar height

3rd line: name of the file which contains the measuring data

The file with the measuring data has to be in the ASCII structure data format .

An identification number = 0 causes the measuring data to be plotted as values.

An identification number > 0 requires node wise data for all nodes or element wise data for elements. Then the differences between these data (current layer) and the measuring data at all measuring nodes inside the modelling area are computed. The differences are plotted as circles or bars. You can adjust the marker type and the marker height with the help of the initialization parameter ETYPund EHGT. Additionally, the program takes down the computed differences into an ASCII file (DiffEich.tmp) if the initialization parameter EDIF is set to 1.

Note: the function for measuring data and differences is not available for vertical sections!

- FUNK

Representation of any data as a function curve, also possible with corresponding 'nominal values'

Input:

1st line: title for the x-axis

2nd line: title for the y-axis

3rd line: name of the file which contains the function values

4th line: colour number for the function curve

5th line: minimum / maximum values (if min = max, the global values are used)

6th line: interpolation flag (1= with interpolation, 0 = without interpolation)

7th line: flag for nominal values (1= with nominal values, 0 = without nominal values)

With nominal values:

8th line: colour number for the nominal values

9th line: name of the file which contains the nominal values

In principle, the command FUNK requires no binary files and no prior program execution. But there should exist a file aaa and the corresponding further binary files (bbb, ccc, ...). Otherwise, the plot generation cannot be executed. The binary files can result from any execution and any model, but they must be present in the current directory.

■   RECH

For intercalculation of two data types

Input:

1st line: identification number of the first data set, transformation flag, time step, layer number

   Layer number > 0: data result from another layer (3D model)

   Negative identification: data result from another calculation (→ another directory), in that case contains the following line the directory, where the data can be found (absolute or relative path from the current working directory, the last character must be a slash '/'!)

2nd line: identification number of the second data set, transformation flag, time step

   Layer number > 0: Data from another layer (3D-Model)

   Negative identification: data result from another calculation (→ another directory) in that case contains the following line the directory, where the data can be found (absolute or relative path from the current working directory, the last character must be a slash '/'!)

   Identification = 0: (only allowed in case the first data set results from another directory or another layer): In this case, the first data set is not transformed. This makes it possible to plot data directly from another calculation/directory or another layer (without any transformations).

3rd line: intercalculation type: '+', '-' , '*' or '/' (1st character in the line!)

(The line with the intercalculation type is not required when the identification of the 2nd data set is equal 0).

Note: You can use a maximum of one data set from another directory or layer in the intercalculation algorithm! Otherwise, you can do the intercalculation directly in the corresponding directory or layer.

The plotting of the intercalculation results is done with the help of one of the commands WERT, KREI, ISOL, FLAE or SCHR and the identification number-99. Afterwards, the intercalculation result is deleted.

■   UGSC

Hatching of elements in a horizontal section of a 3D model where the difference between the potential head and the z-coordinate is less than 0. This command is accessible only via the batch file parameters.

Input:

1st line: colour number, line thickness, line type and time step (for potential heads)

This function is only available for horizontal sections of 3D models. The program computes the difference between potential head and the z-coordinate at the element centre. The plot gives the following information:

In a horizontal section at the layer 0 or 1, usually the z-coordinates of the first node layer are equivalent to the ground level. Then the hatched elements represent the areas where the distance between ground level and ground water level (= ground level - potential head) is positive.

If the section represents the bottom of an aquifer, the elements where the potential head lies beneath the bottom (z-coordinate) are hatched.

9 How To

In this chapter, special procedures are described that occur over and over during creation of groundwater models.

9.1 Determination of catchment areas

A catchment area is the whole area, from which the flow from the surface and subsurface flows at some point together. This point may be a withdrawal well or the mouth of a water in a receiving water course. The edge of the catchment area is the water divide.

The area is mainly determined by the topographic and geological conditions. The catchment area in a groundwater model can be determined by stream- or path lines or Schlieren or in a transport model by the inverse flow. The procedures are explained in the following.
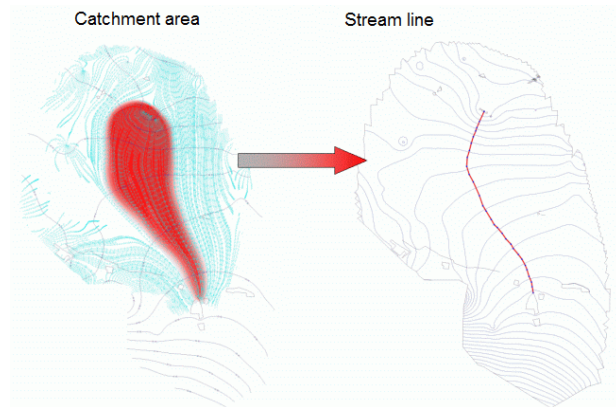
9.1.1 Theory of the path line calculation

A path line is the line resulting from following the movement of a single water particle over a certain flow time t. It is computed by following the flow path of an imaginary water particle through integration of the flow field. Path lines are calculated using the field velocity v_a, but not the Darcy velocity v_f. The field velocity is the actual interstitial velocity referring to the mesh flow area. It is obtained by:

$$v_a = v_f / n_d$$

with

v_a = field velocity

v_f = Darcy velocity

n_d = pore volume relevant to flow (effective porosity)

Two factors strongly influence the accuracy of path lines:

- First, since each element has a constant calculated velocity as a result of the linear shape function for potential heads, areas with changing potential gradients require a fine discretization.
- Second, the transition from one element to the next must be smooth.

To achieve this, the path line is first followed a certain distance (s) with the initial velocity; then in the next step, a new velocity is interpolated and the line is continued. The distances chosen depend on the homogeneity of the flow field.
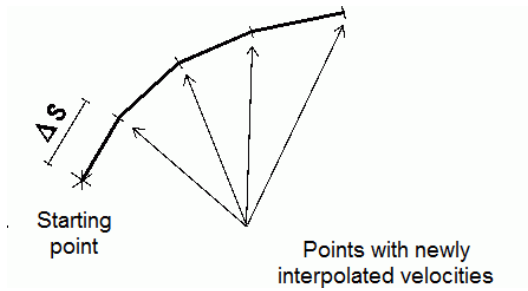
Fig. 222: Computing a path line

Exfiltration wells are difficult points for computing a path line. They must be considered as a secondary condition. Otherwise, the changing flow directions result in zigzag lines:
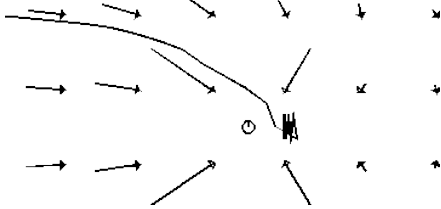

Fig. 223: Computed path lines at a well without any secondary condition

Better results are obtained by checking whether the computed path line has reached the "capture radius" of the well. Then the path line will end directly at the exfiltration node:
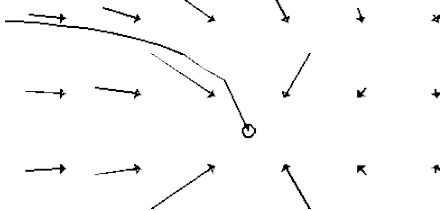

Fig. 224: Computed path lines at a well with "capture radius" depending on the exfiltration rate and the distance

If one of the following criteria is fulfilled, the computing of the path line ends:

■   the defined time or distance is reached,

■   the outer boundary of the model or a free water surface is reached,

■   the path line is captured by an exfiltration well,

■   the distance between two computed path line points is less than the defined limit.

The implementation and the necessary entries in SPRING are described in chapter "Calculating and plotting pathlines" (p. 399) .

The starting point of a path line can be a node, the centre of an element or any point with the coordinates x, y, (z). The direction for following the path line is either the flow direction or the direction opposite to the flow. The latter one is useful for determining the protection area around a well (e. g. 50-days-zone). The *plot generation* (p. **374**) can mark time intervals along the path lines. Finally, isochrones can be manually constructed with them.

The computation of transient path lines differs from the above procedure only in that the velocity field changes continually. Therefore, in addition to the distance s, the particle travel time t must be checked. It cannot be greater than the time-step length T in the transient flow calculation.

9.1.2 Calculating and plotting path lines

For the presentation of path lines the checkbox "Path lines" has to be enabled in the flow calculation with the module GEONEU (steady-state flow) or INSTAT (transient flow). The following window appears in GEONEU in which the criteria for the path lines are established:
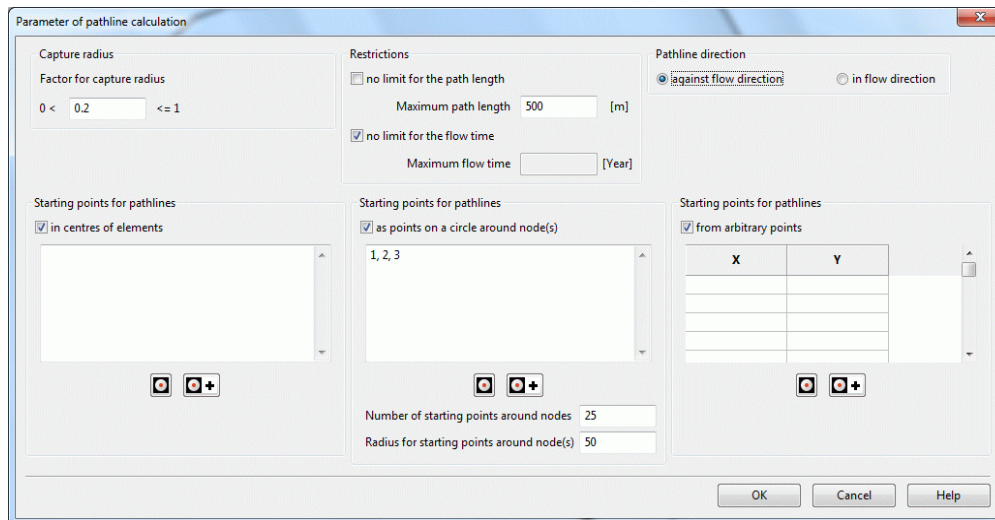

Fig. 225: Input for path lines

The input window of the module INSTAT differs by the lack of the input blocks "restrictions" and "pathline direction".

Capture radius

While tracking path lines on the basis of the calculated velocities near a well, the path line will run directly toward the well only in exceptional cases. To avoid oscillation of the path line around the well, a check is performed to determine if the path line passes within the area of influence of the well. If yes, the path line will be captured by the well. Only nodes lying on elements having a well as one of its corner nodes are considered. The "capture radius" is actually not a radius but a scale factor for the path line coordinates. The capture criterion is therefore dependent on the element size; in addition it is weighted according to the pumping/injection rate at the well. If necessary, the default value can be changed (always > 0).

Restrictions

To limit the path lines, two parameters can be used. You can define the maximum length (m) and/or the maximum flow time (unit of time defined in the *.net-file).

Pathline directions

The direction of the path line can be either in or against the flow direction.

Starting points for pathlines

When selecting the menu items "*Starting points for path lines*" open windows in which the starting points are established. Using the button 🔘 or 🔘➕ an interactive selection in the user interface is possible. A combination of all three types of starting points is allowed.

In the path line calculation is a porosity of 0.2 considered. It can be changed via the batch file of the flow computation (p. **295**).

A typical application of the path line representation is the issue of catchment areas of withdrawal wells, e.g. which area in the vicinity of the well needs to be shown as a water protection zone (e.g. 50-day protection zone). The representation contrary to the direction of flow provides this.

After the flow calculation the plot representing the path lines has to be created with the plot generation. This is done via *File → Plot generation →Top View/Map presentation* and then selecting *Result data → Path lines*. The following window opens:
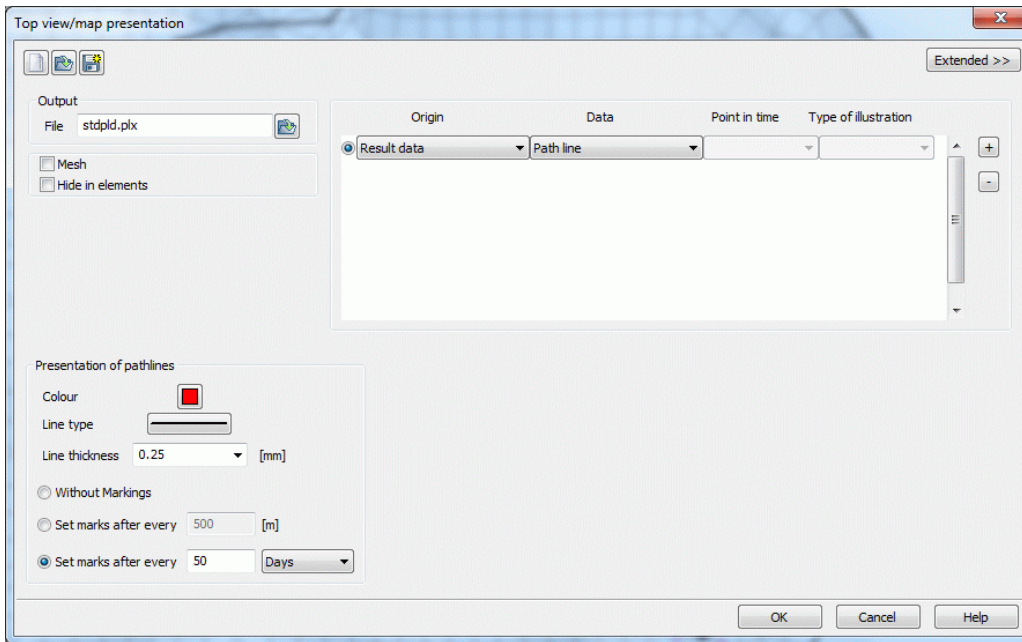
Fig. 226: Criteria of path lines in the plot generation

The statement "Set marks after every 50 Days" means that on the path line a marker after every 50 days is set. To identify the 50-day protection zone of a well by the time markers, the corresponding 50-day isochrone can be constructed manually. For the following image, the above parameters and the node number of the wells (1, 2, 3) were used:

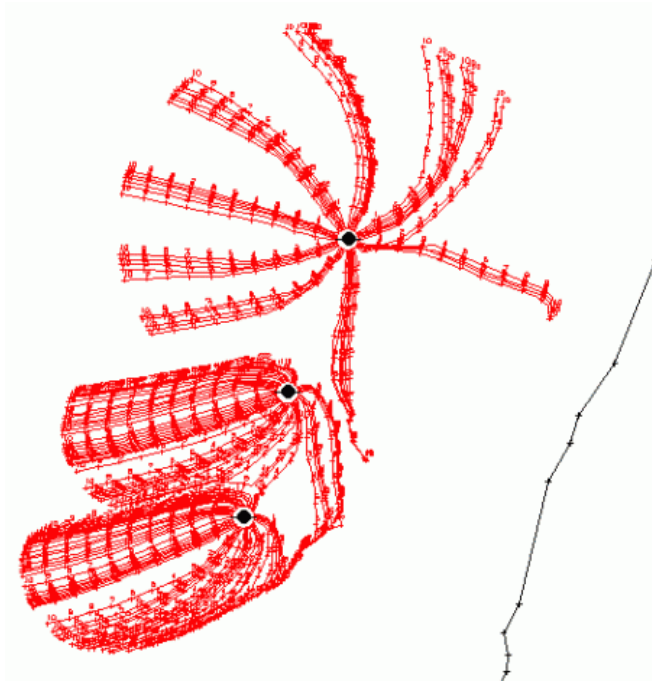Fig. 227: Path lines with a length of 500 m and a marker after every 50 days

On this picture the parameters set in the flow calculation are clearly visible. The path line ends after reaching the 500-m limit, the distance corresponds to the well "radius for the starting points around node(s) = 25 m, and there are 50 path lines.

However, it requires a little practice, to obtain a meaningful picture of this problem.

9.1.3 Theory of the stream line calculation

In the flow field, a streamline represents a dividing line through which no water passes. Therefore, the mass flow rate of water between two streamlines is constant (assuming no infiltration from the top). These theoretical basics limit the calculation of streamlines related to mass flow rates in **2D horizontal models**.

An important aspect for streamlines, which is not taken into account for path lines, is the transition between two elements with different thicknesses. Mass balance requires closer streamlines in larger thicknesses than it does in smaller thicknesses. The streamline results from an equation system at each node. The mass balance defines that the inflow rate must always be equal to the outflow rate (plus possible infiltration rates).

The starting points are also determined by the mass balance for the nodes. First, a limit mass rate per streamline must be defined. The number of streamlines at each well node results from dividing the mass flow rates at the nodes by the limit rate. The streamlines are distributed to the adjacent element sides depending on the mass flow rates passing them. Starting at an arbitrary point, the rates are summed until the sum reaches the limit rate. At this point, the next streamline has its starting point. Starting at the computed points, the streamlines are followed through the next element and again over a side, and so on. In the same way, starting points along the outer boundary or receiving water courses are determined. The inflow and outflow rates are summed until the limit is reached.

The backwards-computed streamlines end at the side of an element adjacent to a node which has an inflow. This can be either a boundary node with a computed inflow rate or any node of the model with an infiltration rate (even from areal recharge).

In general, a horizontal model must take into account the areal recharge rate. It is assigned to all nodes as a small infiltration rate. Therefore, each node can be the endpoint of a backwards-computed streamline. This results in less fewer lines, e.g. along the drainage area of an exfiltration well, and represents the distribution of areal recharge. However, this effect does become apparent until a large number of streamlines per well has been generated.

The computed streamlines not only show the flow paths of the groundwater, but give an overview of the flow rates. Very close streamlines represent large mass flow rates, while widely separated ones represent small mass flow rates. In addition, the quality of the calibration can be checked with the help of streamlines. Individual elements with a much higher or lower permeability than the adjacent elements cause sharp bends of the stream lines.

The graphic representation of streamlines strongly depends on the quality of the finite element mesh, because large discontinuities at the element boundaries cause lateral displacements of the lines.

9.1.4 Calculating and plotting stream lines

The calculation of stream lines is possible only after a two-dimensional horizontal flow calculation (steady state or transient). To illustrate the stream lines (= quantity-based path lines) they have to be exported: *File → Export → Stream lines*. You will get this input window:
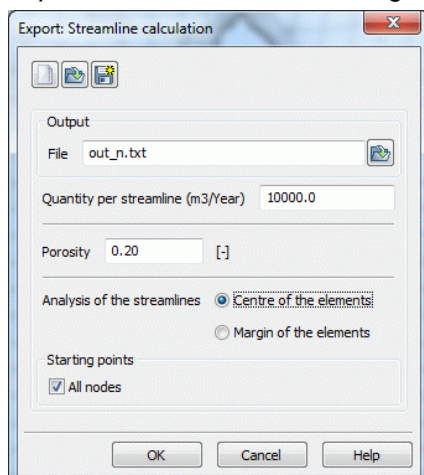
Fig. 228: Streamline calculation

The declaration of the "quantity per streamline (m³/year)" requires a bit of sensitivity for the available water quantities in the model. The example is calculated with 10,000 cubic meters per year, and there are stream lines run in all nodes (specifying individual node numbers is allowed only at withdrawal wells). After the data export the result data "path lines" can be selected in the *plot generation*. The input window is the same as for

the path lines and requires the same parameters. Here, "Set marks after every 100 m", the result is the following picture:
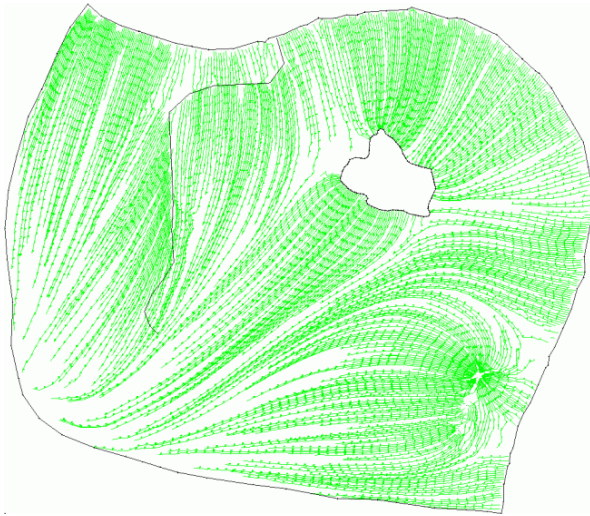

Fig. 229: Quantity of streamlines: 10.000 m³/year

Streamlines close together mean large quantities of water, in areas with sporadic stream lines flows only few water. Sharp bends in the flow curve are an indication of leaps in the K-values of the adjacent elements.

9.1.5 Determining a catchment area with Schlieren (FLIC)

Often the question of the catchment areas of water courses or wells arises, e.g. in the determination of drinking water protection zones or the influence area of a withdrawal well.

The program FLIC (Konrad-Zuse-Centre for Information Technology Berlin) can visualize the velocities computed in a flow calculation by the help of Schlieren. This visualization provides a much better image of flow paths than the velocity arrows used in the plot generation. This is especially important for transient and 3D models where a streamline calculation is not possible.

In the example model, there are two catchment areas of wells, which can be distinguished by the presentation of Schlieren clearly from each other. The procedure for presenting the Schlieren is this:

After a successful flow calculation, the data has to be exported via *File → Export→ Schlieren illustration*. The following input window appears:
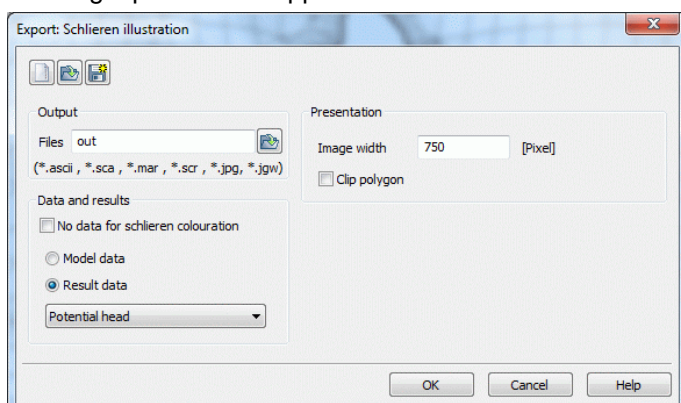
Fig. 230: Input for the Schlieren illustration

The output files are called "out.*" by default, the name can be changed by the user. Attached are the six output extensions .ascii, .sca, .mar, .scr, .jpg and .jwg.

The presentation of the Schlieren is done in the format. ".jpg" with the necessary geo-referencing file ".jwg", which allows a faithful representation of coordinates. The colour of the Schlieren can be changed by selecting a type of data (model or result data). In the example, the potential heads were chosen as a colour sensor. The following picture results:

Fig. 231: Schlieren illustration of the potential heads

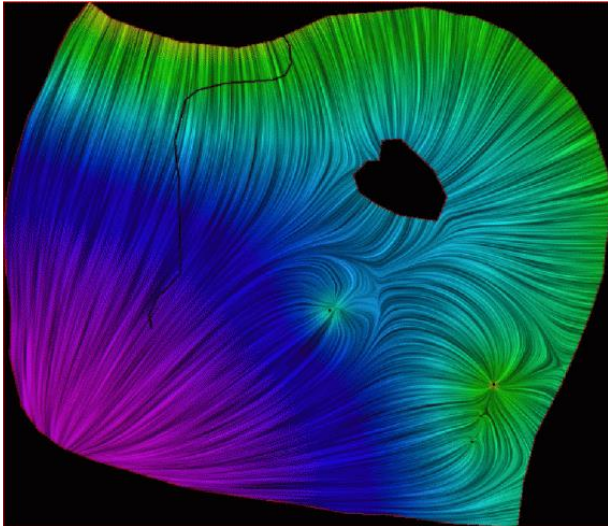The image is called via *File → Import File → Overlay file: *.jpg*. You can see clearly that the course of the Schlieren corresponds to the course of the path and stream lines. The other files can also be used in other geographic information systems (e.g. ARC INFO).

By using the structure menu of SPRING the catchment areas of the (line of -) wells can be digitized along the course of the Schlieren and saved as area structure.
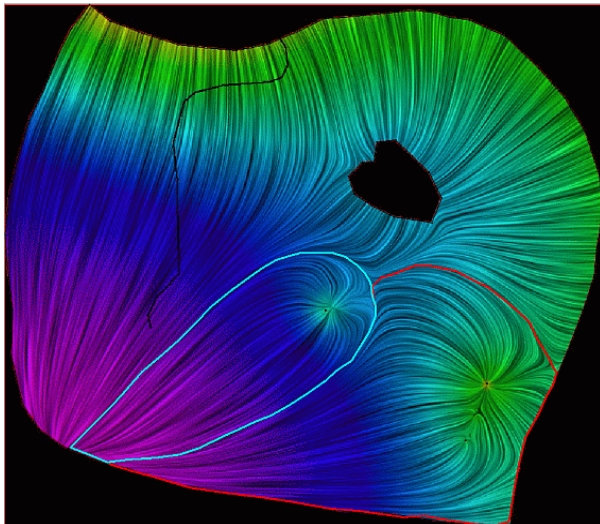

Fig. 232: Digitized edges of the catchment areas of the well (cyan) and the line of wells (red)

After deleting the Schlieren representation and overlaying the topographic map the catchment areas are referred to their geographical site. Eventually you have to change the display of the structures from a polygon into a filled area.
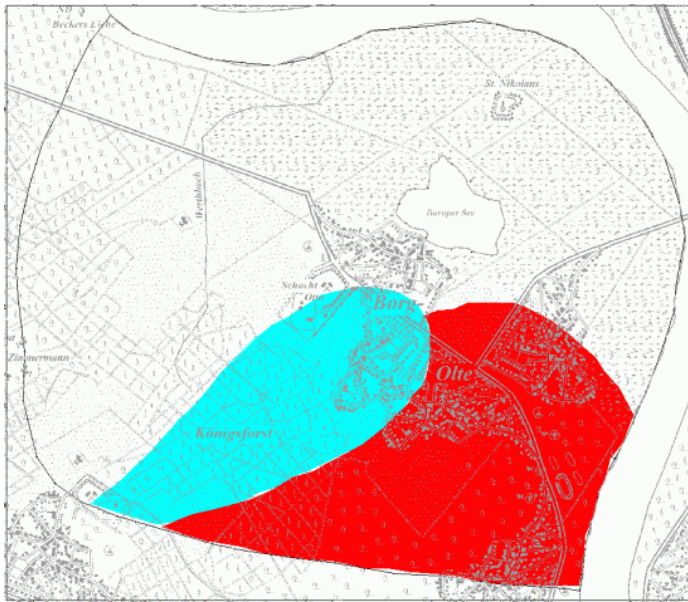
Fig. 233: Catchment areas of the wells

9.1.6 Geo referencing of Schlieren illustrations

For geo-referencing of files in *.jpg or *.jpeg format, a file of the type *.jgw has to be created. For this, the extreme coordinates of the model or the model boundary xmin/ymin and xmax/ymax in world coordinates (usually Gauss-Krüger coordinates) and the display size of the Schlieren illustration in pixels (sizeX/sizeY) is needed.

The coordinates are for example determined using the menu item *View → Show coordinates*. The extension of the Schlieren illustration can be seen in the output file *.scr, after the data was exported for the Schlieren illustration (*File → Export → Schlieren illustration*). The file can be opened with any editor. The line "size" refers to the pixel size of the plot.

If the required numbers are determined, an ASCII file is created with an editor. It contains the following numbers (upper suspension point) and scaling:

```
1st line: (xmax-xmin)/sizeX
2nd line: 0.0
3rd line: 0.0
4th line: (ymin-ymax)/sizeY
5th line: xmin
6th line: ymax
```

The file will be renamed by the name of the Schlieren image with the extension *.jwg. This geo referencing file is automatically evaluated when opening the Schlieren image with SPRING.

Tip: By increasing the pixel value in the Input window of the data export the resolution increases as well.

In the current SPRING-release, the Schlieren images are automatically geo referenced in horizontal and 3D models (only when representing defined layers).

9.1.7 Transport with inverse flow

It is possible to describe a catchment area in consideration of advection- and dispersion-/ diffusion processes. Thus also the local heterogeneities can be simulated with the dispersion term which are not or only limited ascertainable with conventional methods of velocity tracing.

The base of this approach is described in detail in *Uffink, F., Application of Kolmogorovs backward equation in random walk simulations of groundwater contaminant transport; Contaminat Transport in Groundwater, Kobus & Kinzelbach, 1986 and Wilson, J.L. and J. Liu, Backward particle tracking to find the source of pollution in waste management. Waste Management: From Risk to Reduction, 1995.*

If the concentration in the well is given with 1.0 the probabilities of the arriving particle at the well will be calculated as a concentration value. With the presentation of isoconcentration areas from 0.05 to 1.0 the 95%-propability of the arrival of a particle at the withdrawal well will be calculated.

The figures show a 3D example. Here the two aquifers are separated by a groundwater aquitard. The withdrawal wells deliver from the bottom aquifer. The 50 day area is shown in the vertical and horizontal section:
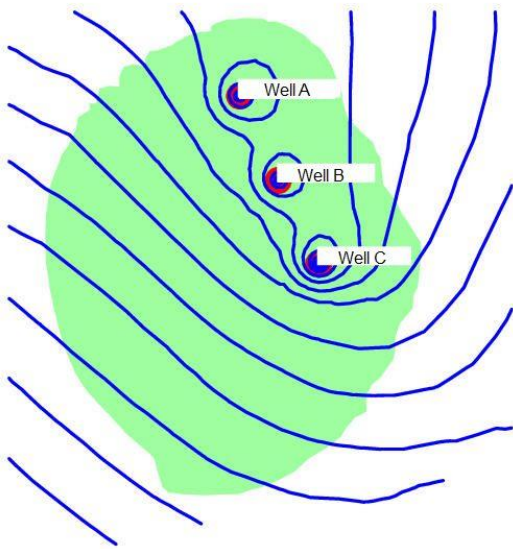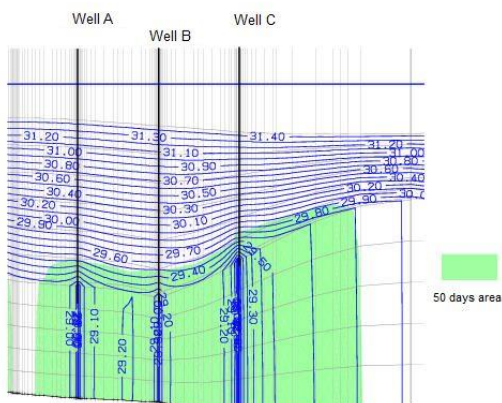

Fig. 234: Horizontal flow

Fig. 235: Vertical flow

A detailed description of the procedure of inverted flow is beyond the scope of this manual. During a training session the calculation of transport processes with inverted flow can be discussed by request.

9.2 Mass balances

A mass balance is the volume-based coverage of the amount of water which flows during a certain period of time in and out of an area. Inflows are covered with a positive sign outflows are covered with a negative sign. Within the model area control lines define lines or surfaces on which the flow can be calculated.

9.2.1 Theory of control lines

Streamlines and path lines show where the water reaching the well comes from. With the help of control lines, one can estimate the mass flow rate coming from a certain direction. The rate is obtained by the flow velocity and the corresponding percolated area. Three-dimensional calculation determines the rates over the depth depending on the layers.
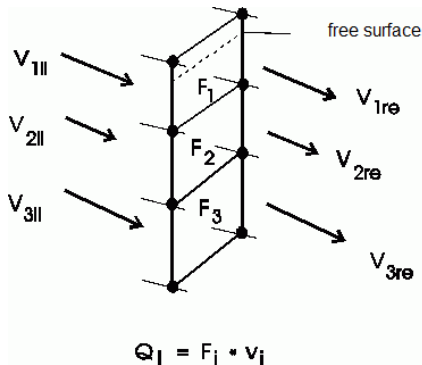

Fig. 236: Calculation of the quantities flowing through a vertical section

9.2.2 General hints for using control lines

The control lines are defined in the model file (*.net) by assigning the attribute KONT. The attribute KONT is independent of the model type (2D or 3D).

The lines can range from a 2D area into a 3D area however they must not run along the 2D/3D-boundary.

In the case of a fixed potential head boundary, mass flow rates crossing the boundary should be obtained by summing the nodal quantities, as these are more accurate (attribute BILK).

The control lines must always follow element edges. Therefore, velocities and flow-through areas of two elements are available. The adjoining areas are averaged, since they are generally of similar size. In the case of velocity, the user can choose between the average value, or the value of the left or right adjoining element. Obviously, the accuracy of the mass flow rate calculation strongly depends on the discretization and the discontinuity of the elements.

Using the attribute KONT the following items have to be considered:

- The control lines are calculated when the appropriate button is enabled in the main dialog box of the flow module.
- Results for each node and the whole line are written in the output file of the flow calculation (out.*). The value of the control line matches the number of the line in the output file.
- If you define a closed control line, e.g. around a well, you must insert the last node (= first node) manually in the model file, otherwise the line is not closed.
- In the module GEONEU you can assign the attribute KONT to nodes along an element diagonal. But if later on a mesh refinement is done, the sequence of the nodes can be incorrect and a warning is given. Then a new assignment is required.
- In the module SITRA it is impossible to calculate a control line along an element diagonal (Warning nr. 34)!
- The sign of the calculated rates depends on the assignment direction:
- Flow over the control line from RIGHT to LEFT is defined positive in the direction of assignment flow over the control line from LEFT to RIGHT in assignment direction is defined negative.
- Three-dimensional calculation determines the rates over the depth from the layer where KONT was assigned down to the model base.

The following chapters describe concrete examples of control lines.

9.2.3 Determining the flow rate to a withdrawal well (KNOT)

To calculate the flow to a withdrawal well a closed control line is placed around the well. In the example, a well (KNOT) is defined in the centre of the model with an amount of withdrawal of -800000.0 m³ / year.

The control line (red) has a distance of about 400 m from the well. The area of the control volume amounts to 560505 m². This results in a groundwater recharge (FLAE = 0.200 m³/m²/a) of 112101 m³/a in this area. In the current state the FE-mesh around the well looks like this:
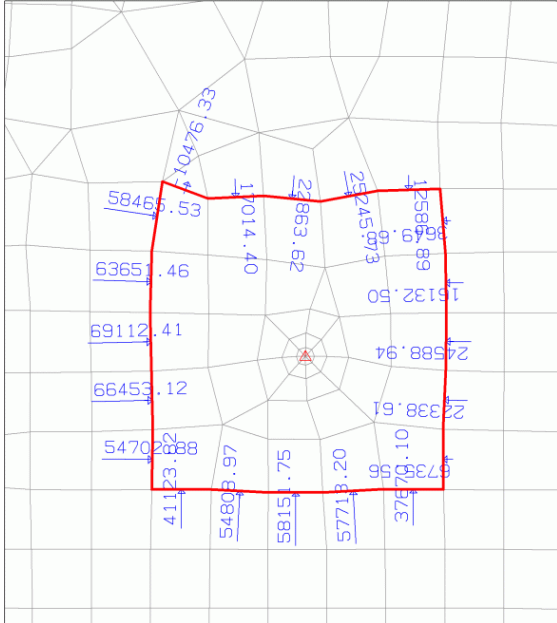
Fig. 237: Initial state of the well calculation

After running SITRA (partial saturation, 5 iterations) a flow into the control volume of 702533 m³/a results. From adding the groundwater recharge of 112101 m³/a you get a total amount of 814634 m³/a respectively 1.8 % deviation.

The next step is a local mesh refinement around the control volume. The following flow computation presents a total inflow of 112101 + 690024 = 802125 m³/a. So the refinement minimizes the deviation to 0.3%.

Fig. 238: Mesh refinement around the well

9.2.4 Determining the quantity of bank-filtered water of a main receiving stream (POTE)

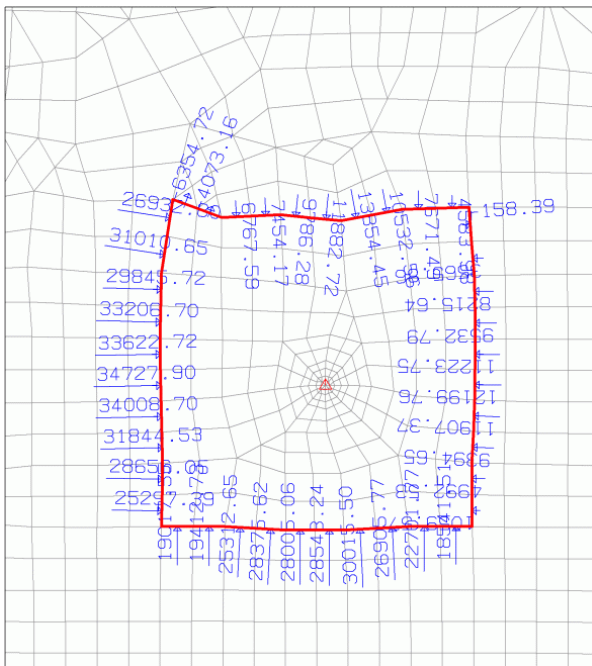If wells are positioned near a main receiving stream an undefined percentage of bank filtrate flows to the wells. To determine this part of the mass balance the attribute KONT is assigned to a node line parallel to the main receiving stream and a flow computation is started. After that the calculated control line quantities are visualized and you can read the flow rates directed to the wells, as shown in the following figure (section):
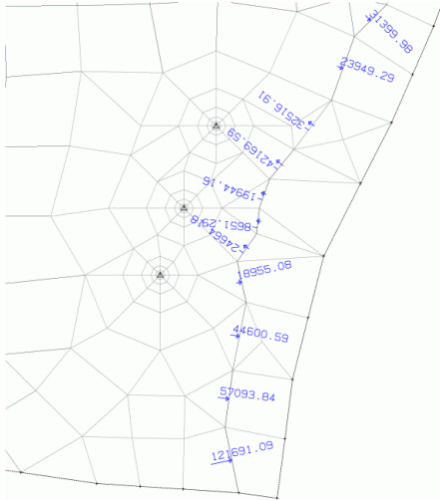

Fig. 239: Bank filtrate rates of the main receiving water course

An amount of 127946 m³/a results when adding the filtrate rates. This corresponds to 9% of the total withdrawal quantities (1450000 m³/a) of the wells.

9.2.5 Groundwater exfiltration into a receiving water course (VORF)

If a receiving water course exists in the investigation area which is defined by the attributes VORF and LERA (or LEKN) you can balance the in- and outflow to the water course with the help of control lines. In the known example file the local area around the receiving water course is refined one respectively two times. Then two control lines are created more or less parallel to the receiving water course. In this control volume, which is limited on the western and eastern side by the control lines, in the north by the main receiving water course and in the south by an imaginary line between the ends of the control lines, the in- and outflow rates are balanced. The mass balance in the steady state must have identical in-and outflows.

The following parameters are necessary:

- Inflow (+) and outflow (-) over the control lines in the control volume (KONT)
- Groundwater recharge in the control volume (FLAE)
- Leakage quantity of the water course (leakage coefficients LERA, LEKN)
- Outflow through the fixed potential heads at the embouchure (Mass flow rates)

Creating the control lines

The amount of water which is calculated by the control line computation corresponds to the groundwater run-off of the receiving water course. Looking at the Schlieren shows clearly that the water course has only a minor connection to the groundwater because the Schlieren run mainly parallel and not **to** the water course.
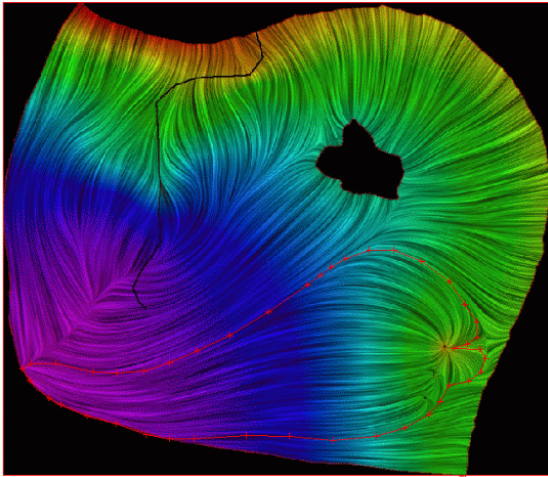
Fig. 240: Schlieren image of the model

The control lines are created in SPRING with *Attributes → Assign → Direct → KONT*. After running the flow computation the quantities flowing over each element side and of the whole control line can be read in the output file (out.* after GEONEU or SITRA).

The signs of the given quantities depend on the assignment direction of the nodes: Looking **IN** assignment direction, the flow from right to left has a positive sign! Otherwise the sign must be negative.
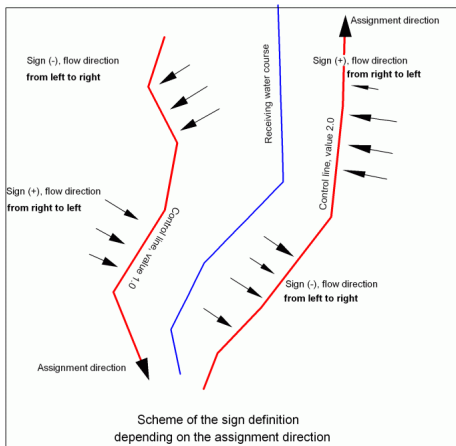
Fig. 241: Scheme of the flow direction depending on the assignment direction
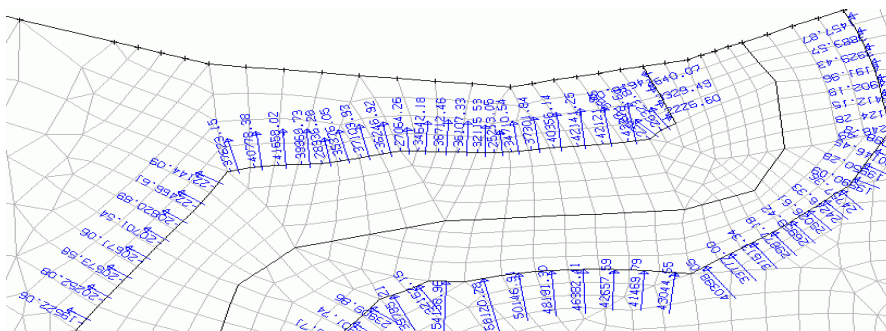
Fig. 242: Control lines and quantities (clipping)

An inflow (+) arises from the eastern side in the control volume and an outflow (-) results at the western side of the control volume.

Although the total sums of both control lines are positive, the value of the western line must be put negative because it is an outflow of the control volume. After the flow computation (SITRA, partial saturation, 5 iterations) the following amounts result: western line: (-) 740284 m³/a, eastern line: (+) 808193 m³/a.

Calculation of the groundwater recharge

After creating the control lines an area structure is created which covers the entire control volume (*Structure → New → Flat*). With *View → Calculate area* (or the corresponding button ◁?) and selecting the control volume the area of it is shown in the status bar of the main window. It amounts to 1975284 m². Multiplicated with the groundwater recharge rate of 0.200 m³/m²/a a groundwater recharge of 395057 m³/a results.

Fig. 243: Flat structure of the groundwater recharge calculation

Determining the leakage rates

For calculating the leakage rate the attribute BILK has to be assigned to the nodes of the water course. Then the total leakage quantity is given in the output file: it amounts to - 209405 m³/a.
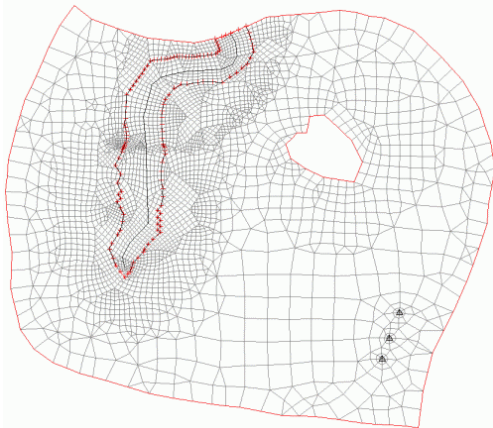
Determining the mass flow rates

The water course flows into a main receiving stream, so the outflow through the fixed potential heads has to be considered as well. The quantities are read in SPRING with *Attributes → Import model data/computation results... → aaa → mass flow rates → KKKK*. With *Attributes → Computation → Sum attributes... → in polygon* you can sum up the mass flow rates inside the control volume. The mass flow rates of the left and right "boundary nodes" of the control volume are averaged and added up only one time. So the total mass flow quantity in the control volume amounts to - 260292 m³/a.

If all required values are determined you can balance them [m³/a]:

395057	+ 808193	- 740284	- 209405	- 260292	=	-6731 [m³/year]
Groundwater recharge	Inflow	Outflow	Leakage	Mass flow	=!	0

The deviation of - 6731 [m³/a] corresponds to 0.6% of the negative water quantity.

The mass balance is a useful tool to check the position of the control lines. The exactness of a control line depends considerably on the discretization and on the flow direction: the more upright the flow the more accurate the from the velocity derived quantity (Q = V * K * I).

9.2.6 Quantity of water flowing through a lake (GLEI)

When creating a groundwater model, some nodes have the same potential head, but the value is unknown. This is valid for:

- surface waters in horizontal and three-dimensional models,
- along the wetted boundary of a vertical well in a vertical model,
- as a boundary condition for the border line between a two-dimensional and a three-dimensional area,
- the nodes of a well below each other in a three-dimensional model.

For all nodes defined with GLEI, the calculation obtains the same value for potential head. Numerically, the equations for these nodes are summed to one equation. They are considered as one unknown value.

Therefore, in three-dimensional models, no different exfiltration rates over the depth of a vertical well are required. The complete exfiltration rate can be assigned to one node. This is not valid for the case of several pumps with different extraction rates over the depth. Only in case of a constant pumping rate can the total extraction rate be assigned to any one of the nodes.

9.2.6.1   Realization in SPRING

In a horizontal model the attribute GLEI (equal potential heads) is assigned to the nodes of a lake. If a lake is characterized like that, it is impossible to read the in- and outflow rates directly. First a flow (steady state or transient) calculation is done. Then you assign in SPRING the resulting potential heads to a user-defined attribute with *Attributes → Import model data/computation results... → aaa → result data: potential head → e.g. KKKK*. With *Attributes → Edit nodes* and picking a boundary node of the lake, you get the calculated potential head of the lake.

This value is assigned as attribute POTE to all boundary nodes of the lake, the attribute GLEI is deleted. At last the attribute BILK (integer value, identifies the balanced group) is assigned to these nodes to get the mass flow rates later on. After the flow computation you can read the mass flow rates of the boundary nodes in the output file. Or you display the resulting mass flow rate with the plot generation, where you can choose the quantity unit to be recalculated:
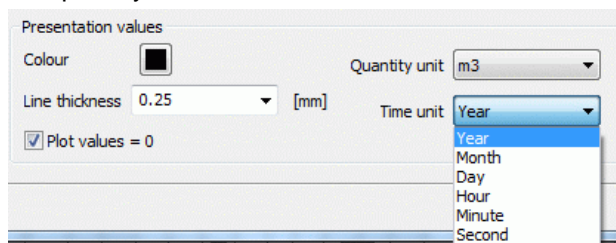
The unit m³/a is chosen. The following figure shows the lake with the calculated mass flow rates.
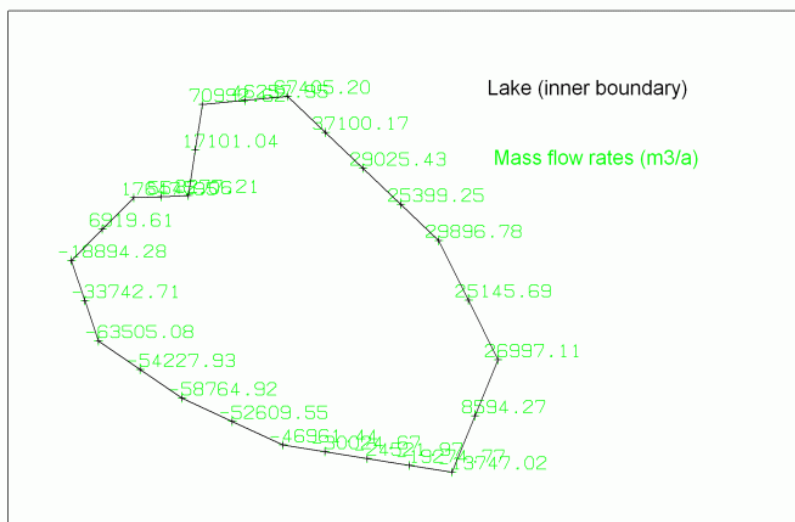
Fig. 244: In- and outflow of a lake

The quantity of water flowing through the lake $Q_{in} = Q_{out}$ is calculated by adding the negative and positive values. Depending on the discretization the values differ from each other in a marginal percentage:

From adding the inflow (= negative sign → it is an outflow of the model because of the inner boundary) results a mass flow rate of 430646 m³/a. From adding the outflow of the lake (= positive sign → inflow into the model) results a mass flow rate of 422458 m³/a. The difference between the two rates amounts to - 8188 m³/a, which means a deviation of 0.5% of the water quantity flowing through the lake. A mesh refinement minimizes the deviation to -1032 m³/a respectively 0.2%, the water quantity flowing through the lake amounts to ca. 416790 m³/a.

Fig. 245: In- and outflow of a lake after a mesh refinement

9.3 Calculating groundwater recharge rates

For calculating the average groundwater recharge rates in SPRING there are three, partially for Nordrhein-Westfalen developed, methods available:
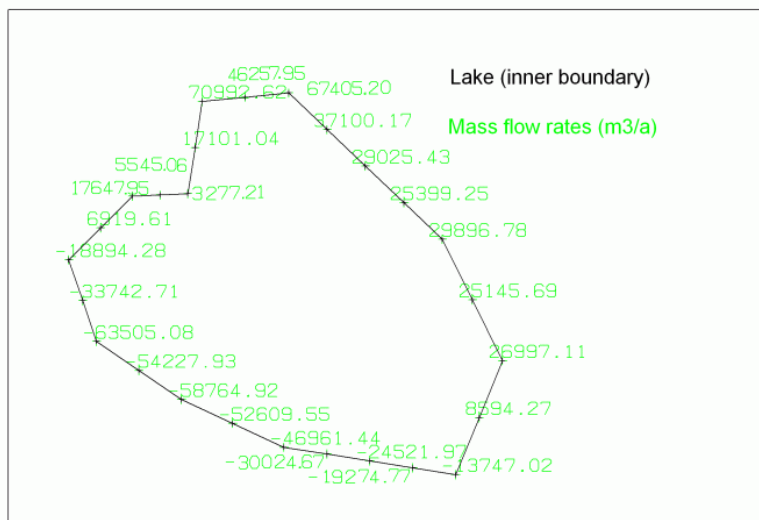
■ Recharge rate calculation by Schroeder and Wyrwich, originally developed for the region near Münster [*Schroeder und Wyrwich (1990): "Eine in Nordrheinwestfalen angewendete Methode zur flächendifferenzierten Ermittlung der Grundwasserneubildung" - Deutsche Gewässerkundliche Mitteilungen, 34, Koblenz 1990.*]

■ Method of calculating the groundwater recharge by Meßer, originally developed for the Emscherregion [*Meßer (2008): „Ein vereinfachtes Verfahren zur Berechnung der flächendifferenzierten Grundwasserneu-bildung in Mitteleuropa"*]

■ Method of the soil water balance

The element-wise calculation of the recharge rate takes place in SPRING via the menu *Attributes → Computation... → Groundwater recharge…*

9.3.1 Recharge rate calculation by Schroeder and Wyrwich

The method according to Schroeder and Wyrwich is based on the converted water balance equation:

Groundwater recharge = precipitation – total evaporation – direct runoff.

■ **Groundwater recharge**: It will be calculated in SPRING from all input data and assigned to the attribute FLAE (column 16: "*") in the unit m³/m²/a. The time unit "year" is required in the model file.

■ **Precipitation**: Precipitation in m/year for all elements or m³/m²/TU element-wise (column 16: „*") (attribute NIED).

■ **Total evaporation**: The average total evaporation rate is taken into account depending on the soil type and land use class of the following table (values in mm/year, N means precipitation):

Soil type ＼ Land use	Farmland, grass land	Deciduous forest	Mixed forest	Coniferous forest	Build-up areas	Water areas
Terrestrial sand soils	380	480	540	600	0.2*N	1.0*N
Terrestrial clay soils	440	540	600	660	0.2*N	1.0*N
Semi terrestrial soils	550	650	700	750	0.2*N	1.0*N

- **Direct runoff**: The still available water from the difference between precipitation and evaporation is then reduced by the direct runoff. Besides the parameters soil type and land use on farm- and grassland the relief energy or the slope is needed. The definitive slope is the maximum slope between two nodes of an element. The program computes this value automatically from the ground level (attribute GELA).

The direct runoff depends on the land use, for **farm- and grassland** it is given by:

Soil type ＼	Relief energy [m/km²]	Slope [%]	Direct runoff [%]
Terrestrial sand soil	0 - 20	0 - 2	0
	21 - 90	2 -9	0 - 100
	> 90	> 9	100
Terrestrial clay soil	0 - 60	0 - 6	0 - 100
	> 60	> 6	100
Semi terrestrial soil	Without considering	Without considering	50

Other land uses are independent of the soil type and the slope:

Land use	Direct runoff [%]
Decidous, mixed, coniferous forest	0
Build-up areas	90*
Water areas	0

*: In contrast to Schroeder and Wyrwich the part of the direct runoff on build-up areas is set to 90% instead of 100%.

9.3.1.1 Input data

The following data must be available for all elements or nodes in the model file:

- Ground level (attribute GELA). To calculate the relief energy and the slope.
- Precipitation (attribute NIED).
- Soil type (attribute NSBT). The soil type must be specified in the model file for all elements according to the following classification:

Soil type	Classification (= NSBT)
Terrestrial sand soils	0
Terrestrial clay soils	10
Semi terrestrial soils	20, 21, 22

- Land use (attribute NSFN): The land use must be specified in the model file for all elements according to the following classification:

Land use	Classification (= NSFN)
Farm-, grassland	0
Decidous forest	1
Mixed forest	2
Coniferous forest	3
Build-up areas	4
Water areas	5

For the determination of the attribute NSFN the digital land use map of the Regionalverband Ruhr (regional association Ruhr) can be used.

Depending on the format, the data can be imported as a structure and assigned in SPRING.

The transformation of the RVR-Code (State: 07/2002) into the classification of the method by Schroeder and Wyrwich is possible without additional model data via the menu item *Attributes → Computation → Groundwater recharge → According to Schroeder Wyrwich → Determine land use from RVR-Code (KVRN → NSFN).*

With this method the groundwater recharge rate can be determined for each finite element according to the above-described water balance equation via: *SPRING → Attributes → Computation → Groundwater recharge → According to Schroeder Wyrwich → Calculate groundwater recharge (NSFN+NSBT+GELA+NIED → FLAE).*

9.3.2 Recharge calculation by Meßer

The method according to Meßer is based on the converted water balance equation:

Groundwater recharge = precipitation – total evaporation – direct runoff.

- **Groundwater recharge**: It will be calculated in SPRING from all input data and assigned to the attribute FLAE (column 16: "*") in the unit m³/m²/a. The time unit "year" is required in the model file.

- **Precipitation**: Precipitation in m/year for all elements or m³/m²/TU element-wise (column 16: „*") (attribute NIED).

- **Total evaporation**: The total evaporation rate is taken into account depending on the climate zone, the land use, the soil type and the class of the depth to water table.

- **Direct runoff**: The still available water from the difference between precipitation and evaporation is then reduced by the direct runoff. Besides the parameters soil type and land use on farm- and grassland the relief energy or the slope is needed. The definitive slope is the maximum slope between two nodes of an element. The program computes this value automatically from the ground level (attribute GELA).

A detailed description of this method is given on this website:

Ein vereinfachtes Verfahren zur Berechnung der flächendifferenzierten Grundwasserneubildung in Mitteleuropa

9.3.2.1 Input data

The following data must be available for all elements or nodes in the model file:

- Ground level (attribute GELA). To calculate the relief energy and the slope.

- Depth to water table (attribute FLUR): To determine the evaporation and the runoff. The attribute FLUR can be calculated in SPRING as difference between GELA – EICH (*Attributes → Computation → Intercalculation → GELA – EICH*).

- Precipitation (attribute NIED).

- Soil type (attribute NSBT). The soil type must be specified in the model file for all elements according to the following classification (nFK = usable field capacity):

Soil type	Classification (= NMBT)
Sand, low nFK	1
loamy sand, middle nFK	2
Loamy clay, high nFK	3
Loess soils	4
Pseudo gley	5

- Land use (attribute NMFN): The land use must be specified in the model file for all elements according to the following classification:

Land use	Classification (= NMFN)
Agricultural use	11
Farmland	12
Grassland	13
Fallow land	14
Without vegetation	15
Decidous forest	21
Mixed forest	31
Coniferous forest	41
Mixed vegetation, share of sealing 0%	51
Mixed vegetation, share of sealing 1-20%	52
Mixed vegetation, share of sealing 21-40%	53
Mixed vegetation, share of sealing 41-60%	54
Mixed vegetation, share of sealing 61-80%	55
Mixed vegetation, share of sealing 81-100%	56
Water areas with discharge	61
Water areas without discharge	62
Dumps, indifferent deposit	71
Deposit in use (without vegetation), low nFK	72
Deposit recultivated (mixed vegetation), low nFK	73
Dump in use (without vegetation), low nFK	74
Dump recultivated (Mischvegetation), low nFK	75

Mud pool without vegetation, high nFK	76
Roof areas drained	81
Asphalt, concrete drained	82
Compund concrete (new), drained	83
Mosaic-/small paving stones, drained	84
Grass gid stones, drained	85
Compressed, drained	86
Sealed, not drained	91
Asphalt, concrete not drained	92
Compounf concrete (new), not drained	93
Mosaic-/small paiving stones, not drained	94

To determine the data type NMFN standardized land use codes as RVR-code (attribute KVNR), CORINE land use data (attribute NCLC) or ATKIS data (attribute NATK) can be used. Depending on the format, the data can be imported as a structure and assigned in SPRING.

- Slope class (attribute NMGK): The definitive slope is the maximum slope between two nodes of an element. The program computes this value automatically from the ground level (attribute GELA). The slope class must be specified in the model file for all elements according to the following classification:

Slope class (NMGK)	Relief energy [m/km²] or slope [%]
1	0 - 20 or 0 - 2
2	21 - 40 or 2 - 4
3	41 – 100 or 4 -10
4	> 100 or > 10

Via the menu item *Attributes → Computation → Groundwater recharge → According to Meßer 2008 → Determine classes of incline of ground (GELA → NMGK)* the slope class can be determined automatically in SPRING.

- Class of depth to water table (attribute NMFK): The class of the depth to water table must be specified in the model file for all elements according to the following classification:
-

Class of depth to water table [m]	Classification (= NMFK)
Depth to water table 0 -1.0	1
Depth to water table 1.0 – 2.0	2
Depth to water table 2.0 - 3.0	3
Depth to water table > 3.0	4

Via the menu item *Attributes → Computation → Groundwater recharge → According to Meßer 2008 → Determine depth to water table (FLUR → NMFK)* the classes of the depth to water table can be determined automatically in SPRING.

- Climate zones (attribute NMKL): The climate zones must be specified in the model file for all elements according to the following classification:

Climate zones, depending on the pot. evapotranspiration (ETpot)	Classification (= NMKL)
ETpot 400 – 480 mm/a	1
ETpot 480 – 500 mm/a	2
ETpot 500 – 520 mm/a	3
ETpot 520 – 540 mm/a	4
ETpot 540 – 580 mm/a	5
ETpot 580 – 640 mm/a	6

Via the menu item *Attributes → Computation → Groundwater recharge → According to Meßer 2008 → Determine climate zones from ETpot (NETP → NMKL)* the climate zones can be determined automatically in SPRING if the potential evopotranspiration is assigned (attribute NETP, unit [m/TU]) to all elements.

9.3.2.2 Determining the evaporation

If all input data for the determination of the real evaporation rate are available (attributes: NMKL, NMBT, NMFK, NMFN), the real evaporation (attribute NMET) is calculated automatically in SPRING via the menu item *Attributes → Computation → Groundwater recharge → According to Meßer 2008 → Calculate evaporation (…).*

9.3.2.3 Determining the direct runoff

If all input data for the determination of the direct runoff are available (attributes: NMGK, NMBT, NMFK, NMFN), the direct runoff (attribute NMAD) is calculated automatically in SPRING via the menu item *Attributes → Computation → Groundwater recharge → According to Meßer 2008 → Calculate direct runoff (…).*

With this method the groundwater recharge rate can be determined for each finite element according to the above-described water balance equation via: *SPRING → Attributes → Computation → Groundwater recharge → According to Meßer 2008 → Calculate groundwater recharge (...).*

9.3.3 Land use classes from RVR-Code

For large parts of the Ruhr district detailed digital land use data of the Regionalverband Ruhr (RVR) are available as a so-called use catalogue of the land use map. This land use key can be transferred into the in SPRING required land use identification NSFN, for the recharge calculation by Schroeder and Wyrwich or NMFN for the recharge calculation by Meßer. For this the RVR key has to be provided about the element attribute KVRN.

It is emphasized that the transformation between the individual use classifications is based on experience. In individual cases this may differ. A review on plausible rates of groundwater recharge with the help of the RVR-code matching topographical map is recommended.

9.3.3.1 RVR-Code and land use by Schroeder and Wyrwich

- RVR-Code 0 – 99, land use (NSFN) by Schroeder and Wyrwich

RVR-Code	RVR description	NSFN-Classification	NSFN-land use
10	built-up areas, serve to living, up to 3 floors	4	Development
20	built-up areas, serve to living, up to 5 floors	4	Development
30	built-up areas, serve to living, more than 5 floors	4	Development
40	Mixed build-up areas	4	Development
50	Trade areas	4	Development
51	Buildings/Plants	4	Development
52	Storage areas	4	Development
53	Empty industrial areas (possible spare areas)	4	Development
54	Parking areas in trade areas	4	Development
55	Urban building yard/ fleet of vehicles, public services, agency in charge of turnpike maintenance, Technical Inspection Authority (TÜV)	4	Development
56	Bus and tram depot	4	Development
57	Exhibition and fair grounds	4	Development
58	Telecommunications	4	Development
59	Other areas	4	Development
60	Industrial areas	4	Development
61	Buildings/Plants	4	Development
62	Storage areas	4	Development
63	Empty industrial areas (possible spare areas)	4	Development
64	Parking areas	4	Development
65	Storage areas for raw materials (coal, ore etc.)	4	Development
66	Other areas	4	Development
70	Build-up areas for sport and relaxation	4	Development
71	Indoor swimming pools	4	Development
72	Gyms, tennis-, ice sports, riding halls	4	Development
73	Multipurpose and event halls	4	Development
74	Stadium	4	Development
75	Structural plants for sports and leisure places	4	Development
76	Other areas	4	Development
80	Municipality need areas	4	Development
81	Public administrations, penal system	4	Development
82	Health service (hospitals, clinical complexes)	4	Development
83	Public, private educational institutions, education centers, libraries	4	Development
84	Play schools, after-school care club, youth and old-age homes/housing departments	4	Development

85	Churches and parish roomses, cloisters	4	Development
86	police, fire department, rescue centers, bunker plants	4	Development
87	Post office	4	Development
88	Culture places (museums, theaters)	4	Development
89	Other areas (i.e. animal homes)	4	Development
90	Agricultural yard and building areas	4	Development
91	Buildings/Plants	4	Development
92	Cultivation enterprises, poultry farms, hen batteries	4	Development
93	Other areas	4	Development

■ RVR-Code 100 – 199, land use (NSFN) by Schroeder und Wyrwich

RVR-Code	RVR description	NSFN-classification	NSFN-land use
100	Other edificial areas (military area)	4	Development
101	Buildings/Plants	4	Development
102	Open spaces within the area	4	Development
103	Other areas	4	Development
110	Freeway and freeway similar streets	4	Development
140	Primary streets and main streets	4	Development
150	Life estate and opening up streets	4	Development
151	Life estate and opening up streets	4	Development
152	Other ways/streets	4	Development
160	Pedestrian zones	4	Development
170	Parking places	4	Development
171	Parking places	4	Development
172	Parking garage	4	Development
173	Bus stations	4	Development
174	Other areas (e.g. garage court not belong to 10-40)	4	Development
180	Railroad traffic areas and side plants	4	Development
181	Stations	4	Development
182	Premises (goods stations, repair shops, engine shacks)	4	Development
183	Track system (city railway, tram)	0	Farm-, grassland
184	Other areas	4	Development
190	Flight and landing places	4	Development
191	Buildings/Plants	4	Development
192	Landing places	4	Development
193	Open spaces	0	Farm-, grassland
194	Other areas	4	Development

■ RVR-Code 200 – 299, land use (NSFN) by Schroeder und Wyrwich

RVR-Code	RVR description	NSFN-classification	NSFN-land use
200	Other public places	4	Development
210	Energy supply	4	Development
211	Buildings/Plants	4	Development
212	Storage areas	4	Development
213	Transformation plants	4	Development
214	Parking places	4	Development
215	Other areas, i.e.open spaces	0	Farm-, grassland
220	Water supply	4	Development
221	Buildings/Plants, i.e. pumping stations, water-towers	4	Development
222	Water extraction plants (wells, enrichment basins, well galleries)	5	Water areas
223	Other areas, i.e.open spaces	0	Farm-, grassland
230	Sewage disposal	4	Development
231	Buildings/Plants	4	Development
232	Sewage ponds and basins	5	Water areas
233	Rain storage reservoir	5	Water areas
234	Other areas, i.e.open spaces	0	Farm-, grassland
240	Waste disposal	4	Development
241	Waste incinerating plant, recovery plants	4	Development
242	Buildings/Plants	4	Development
243	Dumpsite areas	4	Development
244	Recultivated dumpsite areas	4	Development
245	Mud deposition	4	Development
246	Gathering points for recycling/biological degradation areas	4	Development
247	Other areas	0	Farm-, grassland
250	Embankment areas for soil, rubble	0	Farm-, grassland
260	Exploitation areas	0	Farm-, grassland
261	Buildings/Plants (conveyers, transshipping station)	4	Development
262	Excavation areas including the safety stripe and edge green	0	Farm-, grassland
263	Exploited, at present resting, not filled areas	0	Farm-, grassland
264	Exploited, filled areas, without any use	0	Farm-, grassland
270	Public and private public parks and parkways	0	Farm-, grassland
271	Arranged green spaces near the house area (criteria 10, 20, 30, 40)	0	Farm-, grassland

272	Parks (parks, botanic gardens, zoo)	0	Farm-, grassland
273	Other green areas	0	Farm-, grassland
280	Cemetry	0	Farm-, grassland
281	Structural plants	0	Farm-, grassland
282	Occupancy and green	0	Farm-, grassland
283	Expansion areas	0	Farm-, grassland
284	Other areas	0	Farm-, grassland
290	Allotments	0	Farm-, grassland
291	Open spaces near the house area (criteria 10, 20, 30, 40, 91)	0	Farm-, grassland
292	Duration allotments, little gardens, allotments	0	Farm-, grassland
293	Digging land	0	Farm-, grassland
294	Other areas	0	Farm-, grassland

- RVR-Code 300 – 399, land use (NSFN) by Schroeder und Wyrwich

RVR-Code	RVR description	NSFN-Classification	NSFN-land use
300	Game and sports facilities	0	Farm-, grassland
301	Sports fields	0	Farm-, grassland
302	Free and beach baths	4	Development
303	Tennis courts	4	Development
304	Plants for water sport (sport boat anchorage, footbridge)	4	Development
305	Dog training places	0	Farm-, grassland
306	Ride and racecourses	0	Farm-, grassland
307	Golf courses	0	Farm-, grassland
308	Game and kicking places	0	Farm-, grassland
309	Other leisure plants (crazy golf, shooting range, amusement park, drive-in movies, motocross, traffic drill ground, model airfield)	4	Development
310	Camping sites	0	Farm-, grassland
311	Regular camping, trailer parking spaces	0	Farm-, grassland
312	Tent pitch	0	Farm-, grassland
313	Other areas	0	Farm-, grassland
320	Accompanying green	1	Decidous forest
321	Copses	1	Decidous forest
322	Lawn, small copses	1	Decidous forest
323	Other areas	0	Farm-, grassland
330	Running waters	5	Water areas

331	Flowing waters	5	Water areas
332	Fully developed waters	5	Water areas
333	Other waters	5	Water areas
340	Channels and ports	5	Water areas
341	Sluice plants	5	Water areas
342	Channels	5	Water areas
343	Port waters	5	Water areas
344	Other areas	5	Water areas
350	Lakes and ponds	5	Water areas
351	Stationary waters close to nature	5	Water areas
352	Fully developed, stationary waters (bath, boat and surfing waters)	5	Water areas
353	Exploitation waters	5	Water areas
354	Ponds in parks	5	Water areas
355	Fish farming waters	5	Water areas
356	Sports ports	5	Water areas
357	Other areas	5	Water areas
360	Duration meadows and pastures	0	Farm-, grassland
361	Meadows and pastures	0	Farm-, grassland
362	Fruit meadows and fruit pastures	0	Farm-, grassland
363	Other areas	0	Farm-, grassland
370	Farm land	0	Farm-, grassland
380	Acquisition horticulture	0	Farm-, grassland
381	Structural plants (hothouses)	4	Development
382	Acreages, special cultures, tree nurseries	0	Farm-, grassland
383	Other areas	0	Farm-, grassland

■ RVR-Code 400 – 499, land use (NSFN) by Schroeder and Wyrwich

RVR-Code	RVR description	NSFN-Classification	NSFN-Land use
400	Decidous forest	1	Decidous forest
410	Coniferous forest	3	Coniferous forest
420	Mixed forest	2	Mixed forest
430	Copse inventory	1	Decidous forest
431	Ccopse inventory	1	Decidous forest
432	Clusters of trees and tree rows	1	Decidous forest
433	Other areas	1	Decidous forest

440	Reafforestation areas	2	Mixed forest
441	Reafforestations and cultivated areas	2	Mixed forest
442	Deforestation	0	Farm-, grassland
450	Living fallow	0	Farm-, grassland
451	Unused areas within living quarters (empty sites)	0	Farm-, grassland
452	At present unused areas with recognizable opening up measures	0	Farm-, grassland
453	At present unused areas which are provided for living buildings in the land development plan	0	Farm-, grassland
454	Other areas, which are provided e.g. for other uses in the land development plan	0	Farm-, grassland
460	Ccommercial and industrial fallow areas	0	Farm-, grassland
461	At present unused areas with recognizable opening up measures	0	Farm-, grassland
462	At present unused areas which are provided for trade and industry settlements in the land development plan	0	Farm-, grassland
463	Other areas which are provided e.g. for other uses in the land development plan	0	Farm-, grassland
470	Unused areas of farming and forestry	0	Farm-, grassland
471	Agricultural fallows	0	Farm-, grassland
472	Scrubed fallows	2	Mixed forest
473	Other areas which are provided e.g. for green spaces in the land development plan	0	Farm-, grassland
480	Coalmine fallows	0	Farm-, grassland
481	Buildings and plants	4	Development
482	Vacated, unused business areas	0	Farm-, grassland
483	Other areas	0	Farm-, grassland
490	Unused traffic areas	4	Development

■ RVR-Code 500 – 510, land use (NSFN) by Schroeder and Wyrwich

RVR-Code	RVR description	NSFN-classification	NSFN-land use
500	Dumps	4	Development
501	Dumps situated in embankments or material losses	4	Development
502	Recultivated dumps, also parts of a dump	4	Development
503	Other areas	4	Development

Literature: [*Regionalverband Ruhr, Abt. Karten-/Luftbildwesen und Stadtklimatologie (Hrsgb., 2002): " Nutzungsartenkatalog der Flächennutzungskartierung / FNK", Stand der Daten: 10.07.2002, RVR, Essen*]

9.3.3.2 RVR-Code with land use by Meßer

For the method by Meßer, version 2008 a sealing component for the conversion of the RVR-code (as of 07/2002) into the land use by Meßer is not required anymore.

For the transformation of land use data from the RVR-code or from other area codes such as CORINE land use data or ATKIS data into the attribute NMFN Meßer has developed a special calculation routine.

This conversion is based on dbase files which are stored during the installation of SPRING in the directory "C:\Users\Public\Documents\SPRING\Konfig" (Windows 7 or Windows Vista). This path varies depending on the operating system or the hard drive partitioning (e.g. on Windows XP: "C:\Documents and Settings\All Users\Documents\SPRING\Konfig").

Land use data in the format of RVR code are imported as structure data and assigned to the attribute KVRN.

Then, using *Attributes → Computation → Groundwater recharge → According to Meßer 2008 → Determine land use from RVR code (KVRN → NMFN)* with the help of the dbase file the attribute NMFN, that is necessary for the calculation of the evapotranspiration and the direct runoff, is created.

If there are land use data in the ATKIS format, the proceeding is the same as for the RVR code, only that the data must be assigned to the attribute NATK.

9.3.4 Recharge rate calculation on soil water balance

The process of the (transient) calculation of recharge rates to land on the soil water balance is based on the theoretical foundations that are described in DVWK 238/1996 "Determination of evaporation from land and water areas" in detail.

For the calculation of the recharge rate on the soil water balance is the crucial role of the soil as a water reservoir by including the following components considered:

- Potential evapotranspiration (ET_P): evaporation from surfaces under given meteorological conditions and an unlimited mass of water available. The potential evaporation is an operand, which is determined from measured meteorological values (from: DVWK 238/1996)

- Field capacity (FK): The field capacity is the water-holding capacity of the soil against the force of gravity. The field capacity is the maximum water content, that a soil can hold under natural conditions.

- Permanent wilting point (PWP): The permanent wilting point describes the water content of the soil (depending on the plant), below which available water for plants no longer exists (see: Müller, Tibor: "Wörterbuch und Lexikon der Hydrogeologie"). It is therefore the lowest water content, which reached a soil under natural conditions. It depends on the soil and plant.

- Usable field capacity (nFK): The part of the field capacity, which can be absorbed by plants through the roots is called the usable field capacity (see: Müller, Tibor: "Wörterbuch und Lexikon der Hydrogeologie"). The usable field capacity is thus obtained from the difference between field capacity - permanent wilting point (nFK = FK - PWP).

- Effective root depth (W_e): The effective root depth is about 50 - 60% of the maximum root depth. It features the soil volume in which the soil water balance is affected by intensive plant water consumption.

The recharge rate calculation on to the soil water balance applies in the first place to natural undeveloped areas, but a groundwater model usually also includes built-up areas, these areas are in SPRING still considered by the method of recharge calculation by Schroeder and Wyrwich.

The recharge rate calculation on soil water balance, as it is currently implemented in SPRING, is in a development phase. Therefore, in future, certainly changes in required data or input files are possible. The current program description is always available on our hompage under the menu item "Download & Support: Manual SPRING 4.1".

9.3.4.1 Input data in the model file

The following data must be available for all elements and nodes for the top element or node layer in the model file:

■ Ground level (attribute GELA). To calculate the relief energy or the slope to take into account the surface runoff based on Schroeder and Wyrwich.

■ Initial potential heads: To calculate the distance to water level. The conversion of node in element values is done automatically within the recharge calculation.

■ Soil type (attribute NSBT): The soil type must be specified in the model file according to the following classification as an element attribute:

Soil type	Classification (= NSBT)
Terrestrial sand soils	0
Terrestrial clay soils	10
Semi terrestrial soils	20, 21, 22

■ Degree of sealing (attribute VERS): The degree of sealing is specified for each element in the unit [%]. It is necessary to calculate the conditional sealing surface runoff.

■ Land use class (attribute NSFN): The land use class must be specified according to the following classification for all elements in the model file:

Land use	Classification (= NSFN)
Farm-, grassland	0
Decidous forest	1
Mixed forest	2
Coniferous forest	3
Build-up areas	4
Water areas	5

9.3.4.2 Transient climate data

First, a file containing daily values of precipitation, potential evaporation (unit [mm]) and temperature must be provided. The column with the temperature is not currently used, but is expected by the program. The file with the name "niedevap.prn" must be provided in the directory of the model file in the format "formatted text (space delimited)". Each column of the file must be 10 characters wide.

Example of a file "niedevap.prn":

```
         1         2         3         4
12345678901234567890123456789012345678 90
----------------------------------------
Datum     N [mm]    ETp(Haude)T [14:30h]
01.11.2004       0.0   0.67170       11.2
02.11.2004       0.0   0.36714        9.8
03.11.2004       1.4   0.66568       13.5
04.11.2004       3.3   0.18652       13.2
05.11.2004       1.2   0.81364       10.7
06.11.2004       2.2   0.62583       10.6
07.11.2004       0.0   0.81942        9.8
08.11.2004       0.0   1.03164        9.6
----------------------------------------
         1         2         3         4
12345678901234567890123456789012345678 90
```

9.3.4.3 Global soil parameters

Global soil parameters are defined in the file "bopa.txt". The file must be provided in the directory of the model file. If such a file exists, the default parameters are replaced by this. If there exists no file "bopa.txt", the following default values are used.

Structure of the soil parameter file "bopa.txt" with default values:

Column	Meaning	Standardwerte
1	Medium field capacity in [%]: 1st value: sand soils, 2nd value: clay soils	10., 15.
2	Effective root depth in [dm]: 1st value: farm/grass land, build up areas, 2nd value: forest	3., 6.
3	Permanent wilting point in[%]: 1st value: sand soils, 2nd value: clay soils	3., 4.
4	Correction factor of possible over-saturated field capacity	1., 1.

Example of a soil parameter file "bopa.txt":

```
20.,40
3.,6
4.,10
1.0,1.2
```

The correction factor takes an over-saturation of the soil by the field capacity into account, which is reduced in heavy soils after 1 or 2 days. In sand, the correction factor is 1, in heavy soils it is up to 1.2 (from: DVWK 238/1996).

The rate of capillary rise is not considered.

9.3.4.4 Input dialog in SPRING

In contrast to the calculation according to Schroeder and Wyrwich or Meßer 2008 after clicking on the menu item "recharge rate calculation":

Transform land use (NMFN->NSFN)

Compute groundwater recharge (NSFN+NSBT+VERS+GELA+EICH->FLAE)
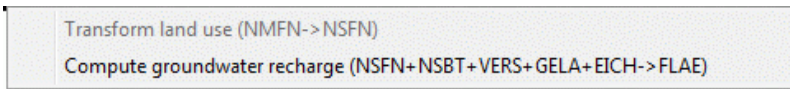
a further input window opens:

Calculation of groundwater recharge, soil water balance

Climatic curve   niedevap.prn

Calculation method

○ Average values elementwise

● Hydrographs averaged over   30 ▲▼   day(s)

☑ Soil parameters   bopa.txt

OK   Cancel   Help
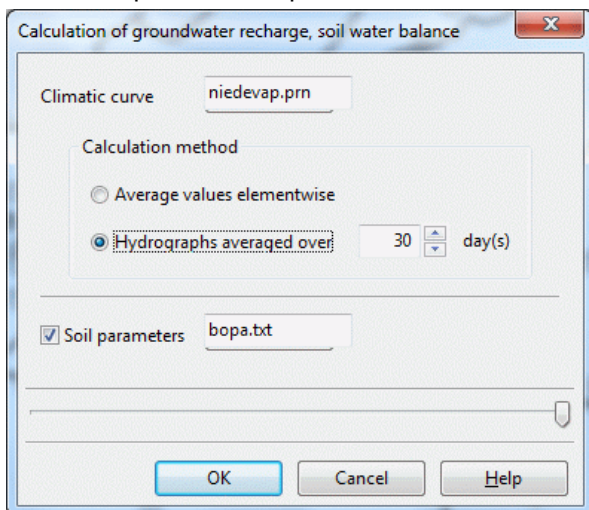
Climatic curve

In the input window the name of the file with the transient climate data has to be entered.

Calculation method

When activating the button "Average values, elementwise" the middle ground water recharge (FLAE) is calculated for each element.

When activating the button "hydrographs, averaged over" the transient data for every x days is generated, averaged over the interval groundwater recharge rate for each element.

The transient values are stored internally during the computation process, and can be exported to a transient input file via *Special Features → Transient → Export transient input file....*

Soil parameters

When activating the check box, a file with user-specific soil parameters can be specified. Otherwise, the file "bopa.txt" with the default values is considered in the calculation.

9.4 Calculating a transient groundwater recharge rate

(The application described below is a preliminary test version!)

As part of the research paper "Development of the groundwater level as a result of climate change on the city of Dusseldorf" (delta h Ingenieurgesellschaft mbH / Institute of Applied Physical Geography RUB, 2014) a method for calculating transient space-differentiated rates of groundwater recharge has been developed which is based on the basis of a soil water balance model (Internet link later ....). The method is integrated in SPRING via the menu RUBINFLUX.

The following figure shows the relevant balance components to determine the daily leachate quantity depending on the precipitation, which is recognized as "groundwater recharge" in the groundwater model.
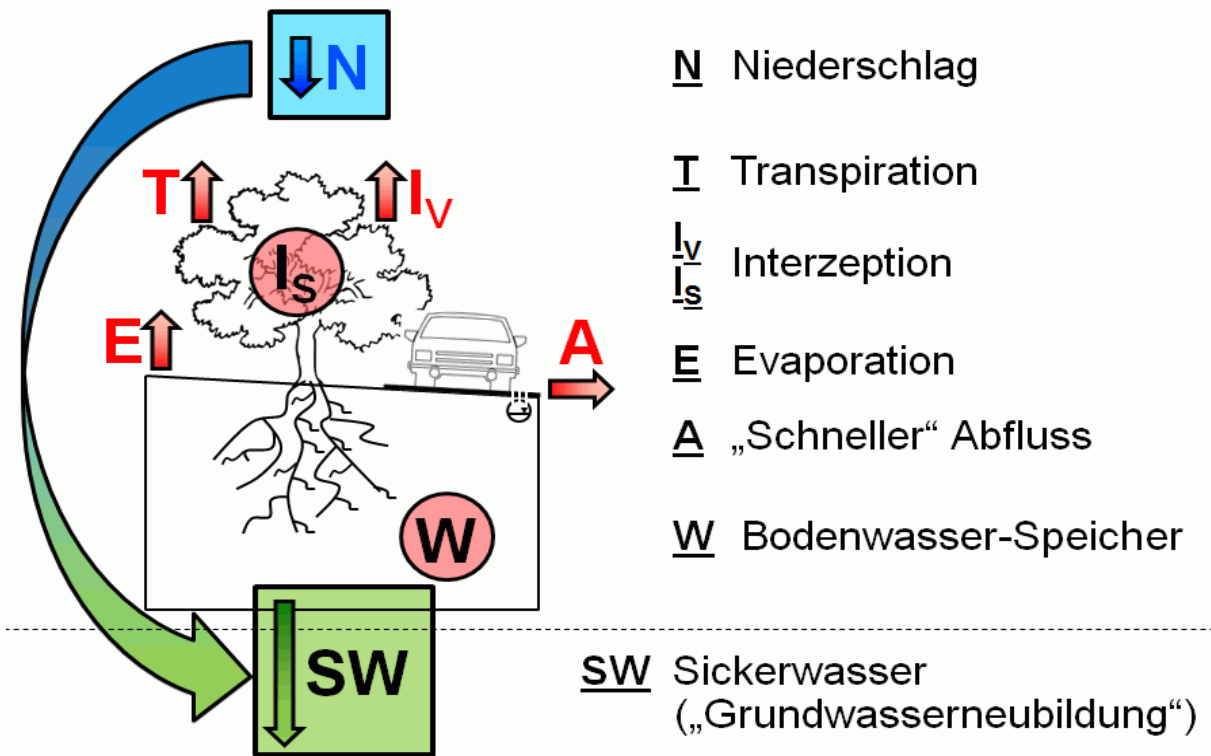
Fig. 246: Balance items for the determination of groundwater recharge rates

- Transpiration, interception and evaporation are the components of the actual ("current") evaporation. They are limited by physical (e.g. heat radiation), astronomical (e.g. the sun) and climatic (e.g. current temperature) constraints and site-related restrictions (e.g. use, vegetation, soil).
- "Rapid" runoff components are determined by their slope, land use, soil class and humidity (part A_{SCS}) and the sealing degree at the observed location (part A_{VERS}). The precipitation N must be reduced by the amount of rapid runoff from soil-dependent parameters and slope (A_{SCS}) if the attribute GGRD is assigned.
- Soil water storage: Its size is defined by the shares of (useful) field capacity and permanent wilting point of soil.
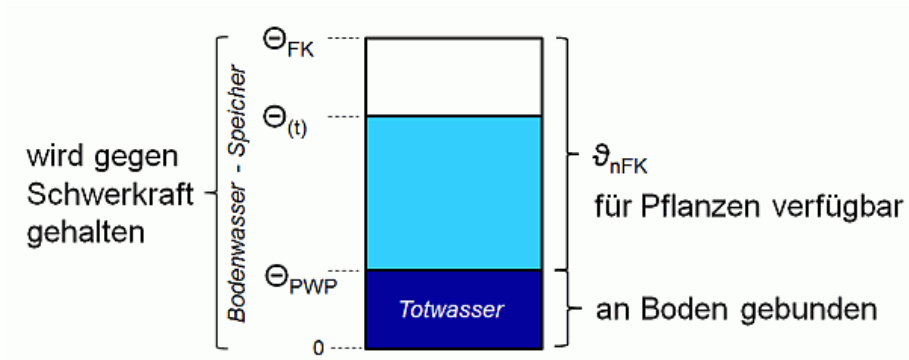
Fig. 247: Water shares of the soil water storage

9.4.1 Input data in the model file

The following data must be available for all elements in the model file.

- Surface sealing (attribute VERS, optional): The degree of sealing is specified in the unit [%]. It is necessary to calculate the sealing dependent "rapid" runoff.
- Slope (attribute GGRD): For elements to which the attribute GGRD is not assigned the rapid runoff component A_{SCS} is not disconnected from the precipitation.
- Layer height (attribute NKLH): NKLH is required for the calculation of the reference grass evaporation and refers to the sea level (e.g. m NN, m NHN, mamsl).
- Latitude (attribute NKBR): NKBR is required for the calculation of the reference grass evaporation. The unit is [°].
- Land use (attribute NKFN) [nr.]: The specification of the land use is made via the ID assignment according to the definitions in the files for the global location parameters (see below). Unless own definitions are made the IDs correspond to the following table:

ID	Land use
1	Grass land
2	Farm land
3	Garden
4	Tree nursery / orchard
5	Water course
6	Standing water
7	Residential, industrial, commercial, maritime transport, airport, public buildings, other land with buildings
8	Mixed use (Sealed + associated open space, for example, farm), fallow land
9	Landfill
10	Mining, mine, quarry
11	Sports, leisure and recreation area, cemetery, playgrounds,
12	Traffic areas
13	Coniferous forest, trees / tree row of softwood
14	Deciduous forest, trees / tree number from hardwood
15	Hedges, bushes, orchard
16	Mixed forest, trees / tree row of deciduous and coniferous or without differentiation
17	Reeds
18	Unknown use
19	vegetation-free area
20	heath

- Soil class (attribute NKBT) [nr.]: The specification of the soil class is made via the ID assignment according to the definitions in the files for the global location parameters (see below). Unless own definitions are made, the encryption is based on the BK50.

Bodenartengruppe der
obersten Bodenartenschicht

0 = Torf, Feinhumus, künstliches Material
1 = lehmig-tonig
2 = tonig-lehmig
3 = tonig-schluffig
4 = sandig-lehmig
5 = stark lehmig-sandig
6 = sandig-schluffig
7 = lehmig-sandig
8 = sandig
9 = feinbodenarm

Fig. 248: Soil classes by BK50

- Climate data ID (attribute NKID) [nr.]: For the classification of climate data time series a climate data ID is assigned to the elements. The climate data ID is part of the file name that contains the associated climate data for the corresponding element. Example: Climate zone 1008 → NKID = 1008 → file name = input_id1008.csv. Example of a file:

```
Datum;P;T;S;ea;u2
01.01.2004;0.00;-0.10;0.00;4.50;3.15
02.01.2004;0.00;-1.40;6.30;3.90;3.07
03.01.2004;0.00;-2.70;0.00;3.80;1.61
04.01.2004;2.80;0.00;0.00;5.30;3.39
05.01.2004;5.86;3.40;0.00;6.90;2.58
06.01.2004;7.76;5.30;0.00;8.00;4.28
```
with:

P = precipitation [mm/d]

T = Temperature [C°]

S = Sunshine duration [h/d]

ea = Vapor pressure [hPa]

u2 = Wind speed at 2 m height above ground level [m/s]

These time series must be provided by the user. All climate time series must begin with the same date and be dissolved in daily steps. The file directory is defined within the input menu.

- Water content at field capacity (attribute NKFK): The water content at field capacity can, for example, be determined by the BK50. The unit of the attribute NKFK is [vol.-%].
- Water content at permanent wilting point (attribute NKWP): The water content at permanent wilting point can, for example, be determined by the BK50. The unit of the attribute NKWP is [vol.-%].

9.4.2 Input dialog in SPRING

The calculation is performed in SPRING currently via the menu Special Features → *RUBINFLUX*. (später *Attribute → Berechnen → Neubildung → RUBINFLUX*(später!!!!).

The following input window appears ("Main settings"):

Climate time series data

This determines in which local folder the climate time series files are stored. The file prefix must be the same for all climate files. Example:

Prefix = klima, then the file names are: klima_id10001.csv, klima_id10001.csv (for elements with NKID=10001), klima_id50.csv (for elements with NKID=50) etc. All climate time series have to begin with the same date.

Climate time series - transient analysis

Here, the portion of a time series is set for which a recharge calculation is to be made. If the time series runs for several years, the number of time steps determines the initialization date from which the data is to be exported as transient boundary condition. With the given time steps after the initialization is determined until the data should be calculated and exported.

Output

The name of the output file with the transient boundary conditions as well as the reference date in the transient input file are defined. The user can specify on the indication of averaging whether the transient recharge rate is written for each day (average = 1) or, for example, only for every 10th day (average = 10, corresponding to the average over 10 days) in the file with the transient boundary conditions.

By activating the checkbox the element averages of the groundwater recharge (attribute FLAE) and the rapid runoff components are calculated. The rapid runoff component of soil-dependent parameters A_{SCS} is assigned to the attribute NKAG and the sealing-dependent run off component A_{VERS} is assigned to the attribute NKAV.

9.4.2.1 Global location parameters

After selecting the tab "Global location parameters" the following input window appears:
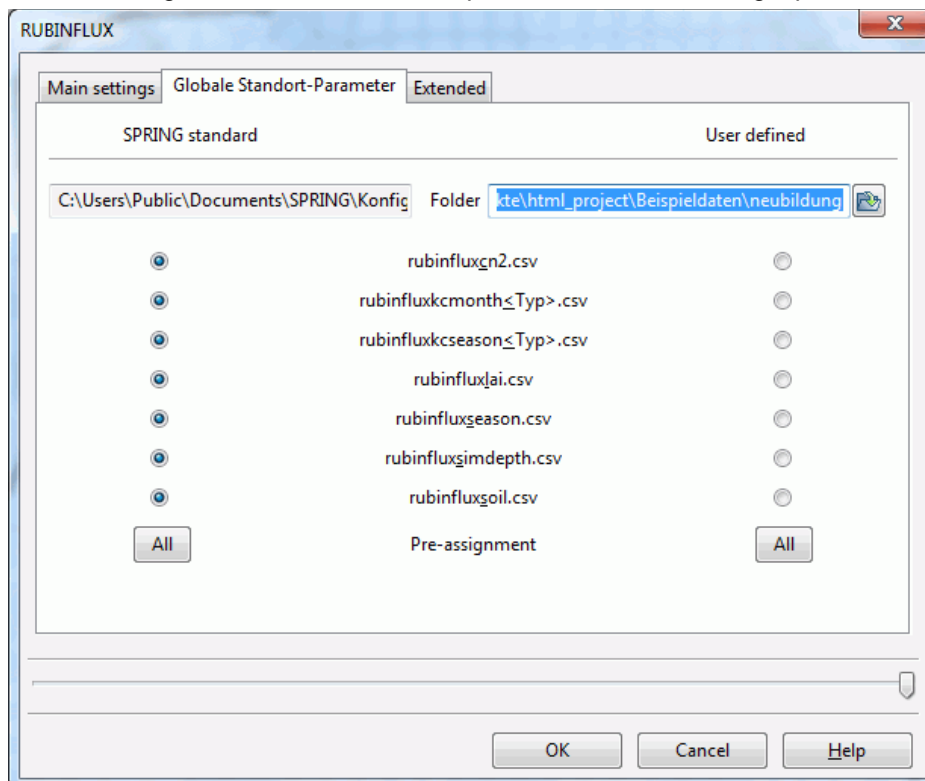
The files with the default parameters are stored in the directory "C:\Users\Public\Documents\SPRING\Konfig" (Windows 7). They contain data related to soil type (_soil), vegetation period (_season), leaf area index (_lai), SCS parameters (_cn2), depth of simulation (_simdepth) and inventory coefficients (_kc_season, _kc_month). These files can also be custom made available and stored in a different directory.

For the format of the *.csv files applies: Column separator = semicolon, decimal separator = point or comma.

During the calculation of the transient recharge an "on-the-fly" -linking of the basic tables happens based on the existing data of the model elements. Thus, different parameter combinations can be tested without interrupting the SPRING session.

9.4.2.2 Extended

Program control by command line parameters is currently used exclusively for further development purposes. Usage is only for the developers.

9.5 Leakage relations and receiving water interactions

If a large receiving water course exists in the model area or at the field edge, it is in most cases directly connected to the groundwater aquifer. The water level can be assigned as a fixed potential head (POTE) to the corresponding nodes. The resulting inflow or rather outflow rates are represented by mass flow rates. The process with water flowing from the aquifer into the surface water is called exfiltration. The opposite process, with water flowing from the surface water into the aquifer, is known as infiltration. In the case of smaller water courses, there is often no direct connection with the aquifer.

Because of colmation or the formation of an unsaturated zone between the surface water and the groundwater the exchange of flow is disturbed.

The following figures illustrate the various terms related to the aquifer:

Exfiltration of the groundwater into the water course	Infiltration of the water course into the groundwater	Infiltration with tear-off height
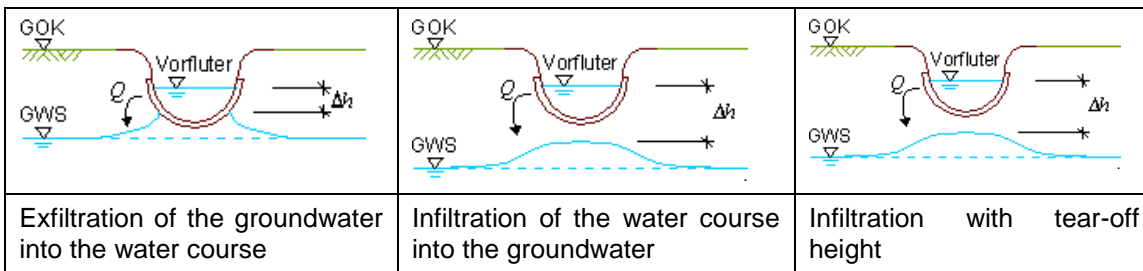
The direction of the flow and the flow rate is determined from the mostly linear relation between the water level in the surface water and the groundwater level. The corresponding function is called leakage function. The leakage coefficient [1/s] represents the quotient between the inflow / outflow rate Q (referring to the width of the surface water and the proportional length) and the average potential difference. Especially for infiltration, there exists a maximum rate. Independent of further increase of the potential difference, the current rate will never exceed this maximum. For certain situations, limiting the exfiltration rate can also be useful.
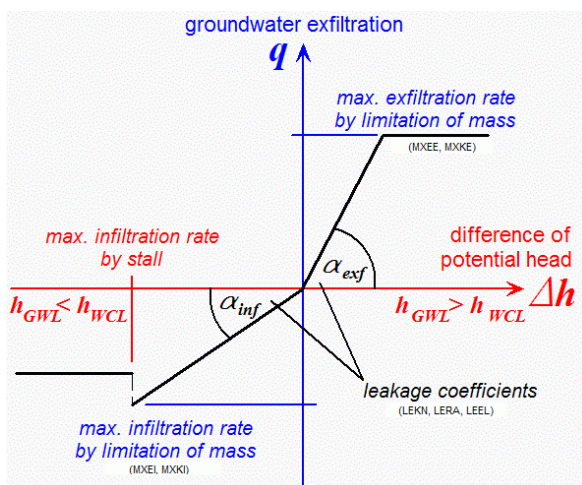
Fig. 249: Schematic representation of the leakage function

Predominantly, leakage functions are defined along lines. But surface leakage relationships are useful as well. SPRING provides the following input data for the definition of leakage processes

- VORF: node wise height of the water level in the surface water

- ABRI: Potential difference between water course and groundwater level from which the infiltration rate does not increase any more (tear-off height)
- LEKN: Leakage coefficient of a node
- LERA: polyline wise leakage coefficient
- LEEL: Leakage coefficient for elements
- LKFA: Ratio between exfiltration und infiltration at nodes (LEKN) or polylines (LERA)
- LEFA: Ratio of exfiltration and infiltration in elements (LEEL)
- MXKE: max. exfiltration rate of a polyline
- MXEE: max. exfiltration rate for elements
- MXKI: max. infiltration rate for a polyline
- MXEI: max. infiltration rate for elements

Absolutely necessary data for a leakage calculation are VORF and values for one of the data sets LEKN, LERA or LEEL.

Leakage in case of impermeable layers

The water exchange through an 'impermeable' layer is a special case for surface leakage. The relatively tight layer slows down the rising of the groundwater level (see max. thickness of the groundwater aquifer), but enables water exchange processes, especially for steady state.

Leakage through an 'impermeable layer' requires the following data:

- UNDU: Thickness of the impermeable layer (requires the input of the ground level GELA)
- LEEL: The meaning of this value depends on the data type UNDU: UNDU>0: LEEL = vertical K-value of the impermeable layer in m/s; UNDU=0 or no input: LEEL = direct leakage coefficient in 1/time unit.
- VORF: Water level of the surface water (if no surface water is existing, VORF is equal to the ground level)

In the following chapters, the practical implementation of individual water course interactions in the ground water model is explained using typical examples.

9.5.1 Controlling the depth to water table with a withdrawal well

The mining industry has caused an increase of the groundwater level in many cultivated areas, so that the groundwater level is often too near to the surface. To avoid damage at existing buildings controlling the groundwater level with withdrawal wells is necessary. In contrast to wells, used for groundwater winning, the pumping rate of local drain wells is often unknown. So they cannot be considered as boundary condition KNOT. More important than the pumping rate is the altitude of the groundwater subsidence. The following describes the modelling of such a situation with SPRING.

The figure shows schematically the problem:
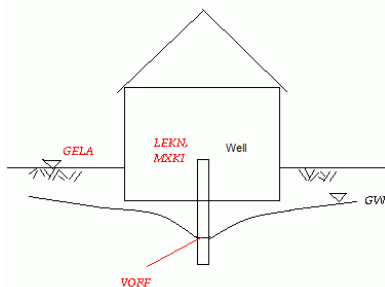
Fig. 250: Schematic representation of a polder measurement in the basement of a building

- VORF

The FE node of the drain well gets the attribute VORF, its value corresponds to the maximum admissible altitude of the groundwater level (e.g. basement foundations - 0.8 m). In a 3D model it is sometimes necessary

that the FE nodes in the filter area of the drain well get the attribute GLEI. This depends on the vertical discretization of the FE net. In this case the attribute VORF has to be assigned to one of these nodes.

- LEKN

Additionally to the attribute "VORF" the leakage coefficient LEKN must be assigned to the FE node of the drain well. A very great value of the attribute "LEKN" acts like the fixed potential head "POTE". This guarantees keeping the requested altitude of the groundwater level. If pumping rates of the drain well exist the leakage coefficient can be calibrated by comparing the measured and calculated rates.

In SPRING you can assign the calculated pumping rates to any attribute (e.g. KKKK) with *Attributes → Import result data*, selection of the background file "aaa" and selection of "Leakage rate (m³/node/TU)". With *Attributes → Edit nodes → Capture* all attributes of the node are listed, including the pumping rate.

- MXKI

If the groundwater level is below the defined value of "VORF", the drain well infiltrates into the groundwater. To prevent that the attribute "MXKI = 0.0" has to be assigned to the FE node of the well.

- MXKE

Sometimes it is necessary to limit the pumping rate to a maximum, e.g. to consider the capacity of the drain well. In this case the attribute "MXKE" [m³/TU/m] defines the maximum possible pumping rate. Attention: Taking water out of the model requires a negative value!

9.5.2 Modelling of a drainage

In areas with a groundwater level near the surface this level has to be controlled. Especially in cultivated regions you can do this with drain pipes. The groundwater exfiltrates in these drainages and they drain the water to the next receiving water course.

The following figures show the problem:
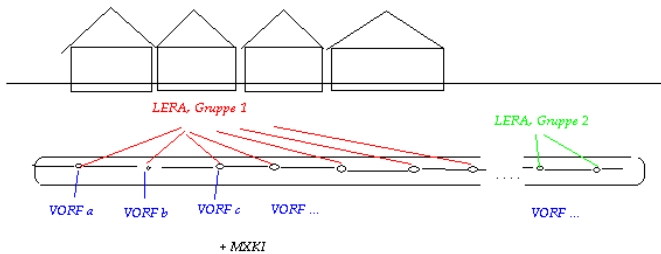
Fig. 251: Longitudinal section through a drain pipe in the street
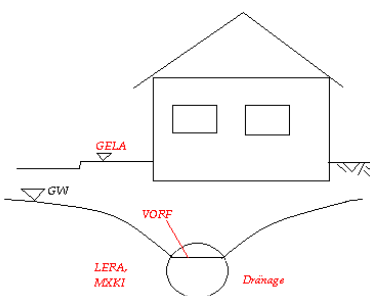
Fig. 252: Cross section through a drain pipe in the street

The realization in SPRING looks like this:

- VORF

The attribute VORF is assigned to the nodes which represent the drain pipe. The values correspond to the requested level of the groundwater surface (e.g. basement foundations - 0.8 m).

- LERA

When modelling a drain pipe the attribute LERA ([m/TU]) has to be assigned to the nodes. It is very important to keep the nodes in order and to separate each drain pipe in different node groups. If discharge rates of the drain pipes exist the leakage coefficient of the drain pipe can be calibrated while comparing the calculated leakage rates with the measured leakage rates. In SPRING you can assign the calculated leakage rates to any attribute (e.g. KKKK) with *Attributes → Import result data*, selecting the background file "aaa" and selecting "Leakage rate (m³/node/TU)". With *Attributes → Edit nodes → capture* all node attributes are listed, including the leakage rate "KKKK". The summation of the single node rates gets you the total quantity of the calculated runoff of the drain pipe.

- MXKI

If the groundwater level is below the defined value of "VORF", the (empty) drain pipe infiltrates into the groundwater. To prevent that the attribute "MXKI = 0.0" has to be assigned to the nodes of the drain pipe.

- MXKE

Sometimes it is necessary to limit the runoff rate to a maximum, e.g. to consider the capacity of the drain pipe. In this case the attribute "MXKE" defines the maximum possible runoff rate. Attention: Taking water out of the model requires a negative value! "MXKE" is defined as m³/TU/m, so a total quantity is given to each drain pipe. The sequence of the nodes must be kept! The maximum value for each node is calculated through the multiplication with the proportionate section between the left and right node.

9.5.3 Seepage surface in a dam

A calculation of the seepage surface in a dam needs precise discretization. Inside the dam is a permeability which is about tenfold lower than the permeability of the outer material. This leads to oscillation problems in the flow calculation, if the element size is not adjusted stepwise.

The following figure shows the essential model data for a dam:
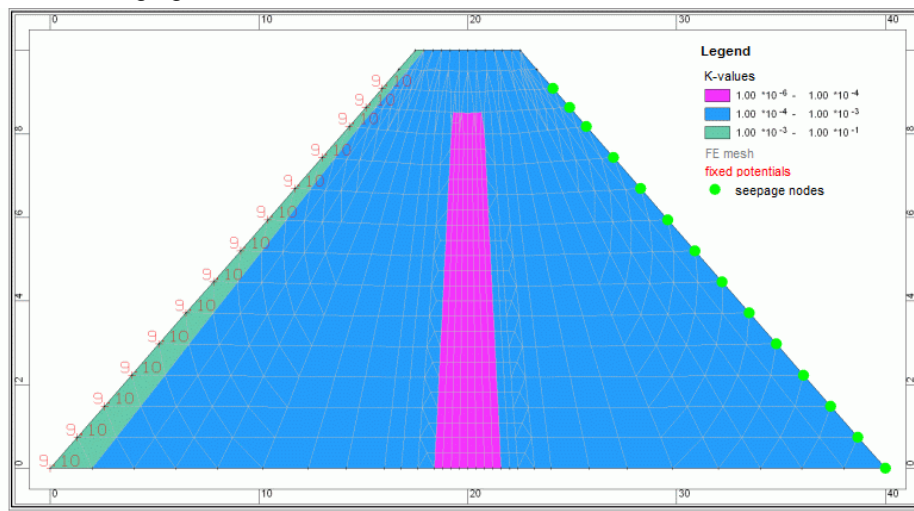
Fig. 253: Model data of a dam (double super elevation in y-direction)

These are the necessary parameters for a flow calculation:

- POTE: fixed potential heads at the boundary nodes of the water side in m NN (altitude of the water level)
- KWER: green = 0.001 m/s, blue = 0.0001 m/s, pink = 0.000001 m/s
- MAEC: horizontal expansion (user defined)
- SICK: at the land side boundary nodes up to the altitude of the water level (value = 0.0)

After a steady state flow computation (10 iterations) you get the following figure:
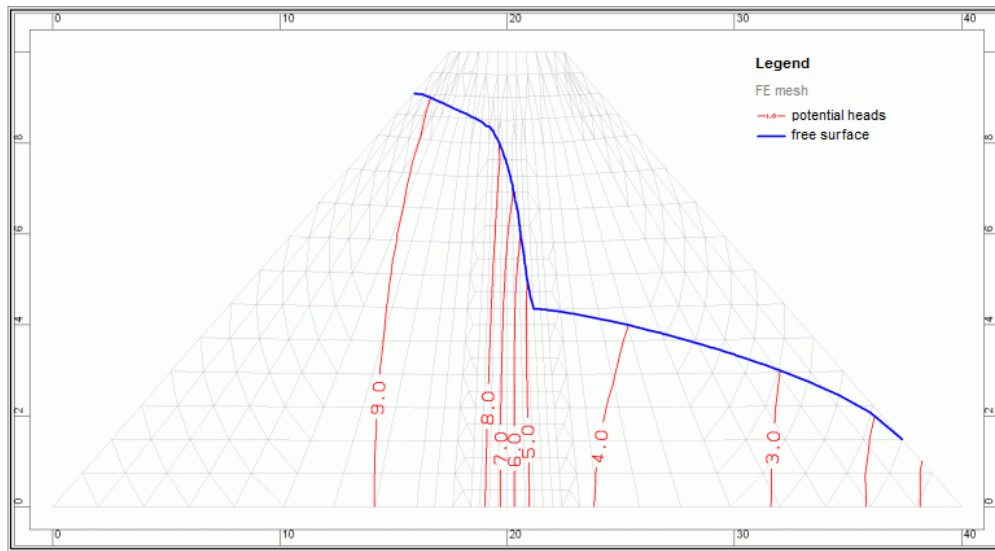
Fig. 254: Result data: potential heads and seepage surface of the dam

The location of the seepage surface (blue) corresponds to the calculated zone of the free surface.

Where the seepage line intersects the land side of the dam water percolates to the surface. In figure 6 you can see it at three nodes at the right bottom of the dam. In SPRING you can assign the percolation rate to any attribute with *Attributes → Import result data*, selecting the background file "aaa" and selecting "Seepage rate [m³/no./TU]", e.g. to "KKKK". With *View → Show attributes* you can then visualize the chosen attribute and with *Attributes → Edit nodes* all node attributes are listed, including the seepage rate. The total quantity of the seepage rate has to be calculated per running dam-meter.

9.6 Simulation of flooding processes

Agreements

A mine is at the beginning a total unsaturated 3D area which is totally or in parts under the compound surface. In addition:

- The nodes which belong to the mine are indicated by explicit zone numbers (data type GRUB, numbers > 0).
- For all FE-mesh-nodes of the considered mine where an inflow is possible the boundary conditions must be defined as "free off seepage" (data type SICK). If there are drainages for the mine pumping facilities which drain into the mine the affected nodes can get the boundary condition "water course level and leakage", additional the node can be supplied with the attribute for the zone number of the mine (data type GRUB, see above).
- For the time $t=t_0$ is no water in the mine located (no initial volume).
- For a transient flow calculation a steady ascending curve is available for the description of the ratio between water spreading volume (m³/s) and the water spreading height (m NN).

Temporal course of the water spreading in a mine

After each time step $t = t_i$ the flow rate is balanced which inflow happens from time $t = t_0$ to time $t = t_i$ into the mine. The volume is composed of the inflow considered at the point in time $t = t_i - t_{i-1}$ on the mine nodes and the spreading volume until to the point in time $t = t_{i-1}$. If there are drainages in the mine and marked in the way which is described above the inflow of the drainage will be added to the volume in the cavity.

With the relationship between the water spreading volume and the water spreading height the resulting water spreading height $h(V_i)$ can be calculated at the considered point in time from the volume $V(t_i)$.
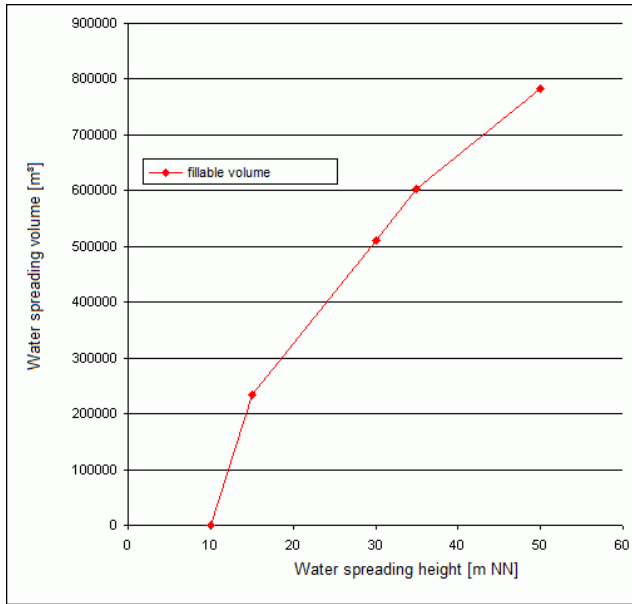
For each node with the boundary condition "free off seepage" the vertical location z estimates if the node lies over the free surface $h(V_i)$. For a node lying under the free surface the boundary condition "free infiltration boundary" is changed into "fixed potential" or alternatively in "water course with leakage" with the height $h(V_i)$. For all other nodes with the boundary condition "fixed potential" or "water course with leakage" the height $h(V_i)$ will be updated.

This procedure guarantees that the mass which is removed out of the model by the boundary conditions "free off seepage" will be completely added back through the definition of fixed potentials or water course nodes. So the mass balance in the model is ensured.

If the "off seepage nodes" change in "potential nodes" or in the damped boundary condition "water course node" is defined through flow parameters.
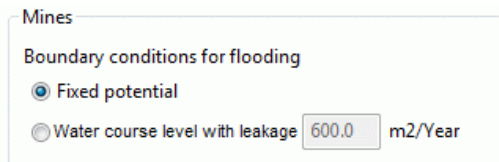
When choosing the "water course node" entering a leakage value for the nodes (LEKN) is required.

Pile-up of a mine

If all nodes of a mine have at one point in time the boundary condition "confirmed potential" the mine considered to be overflown. In this case all (potential-) boundary conditions are "deactivated" so that the mine will be considered in the flow calculation as a totally saturated subarea of the aquifer.

Lowering device of the water level in a mine

If the water level drops in the mine the nodes now lying over the free surface will change (inverted equal to the water spreading procedure) from the boundary condition "confirms potential" back to the boundary condition "free off seepage".

If the water level drops down after a pile-up in the mine all mine nodes get back their originally boundary conditions. That means, when the water volume in the mine is no more filling it up the mine nodes which lie below the surface get the boundary condition "confirmed potential" and the nodes lying above the surface get the boundary condition "free off seepage", always considering the relationship between water spreading volume and water spreading height.

Limitation of the water level

If for a mine a maximum water level is defined and achieved the volume which guarantees keeping the water on that level in each time step is removed from the model. The calculation of the shrinking volume is based on the water spreading volume-water spreading height-relationship. Thus the water level must be located in the codomain of the function.

A temporal changing of the water level is always possible using the transient input file. The data type is described through HMAX. The values correspond to the zone numbers of the mine.

Addition of water from outside the model

When water is added to the mine from outside the model during the flooding phase this can be defined as a flow rate. If such a volume is defined for a time step it is considered in the calculation of the water spreading height using the water spreading volume-water spreading height-relationship. A mathematically positive volume defines a water supply to the mine and a negative volume defines a water removal from the mine.

A temporal changing of the water supply is always possible using the transient input file. The type of data is described with MENG. The values correspond to the zone numbers of the mine (see "Limitation of the water level").

Definition of the flooding parameters in SPRING

First of all each FE-mesh-nodes which describe a mine get an explicit number at the data type GRUB before a flooding simulation with the above described method can be executed. In the following procedure this number is used as the zone number (X) of the mine.

If nodes indicate a mine the further flooding parameters "water level" and "addition volume" can be defined with the menu *Attributes → Special features → Flooding parameters* (GRUB). Should a set parameter be changed the entry in the relevant field has to be deleted (field remained blank).
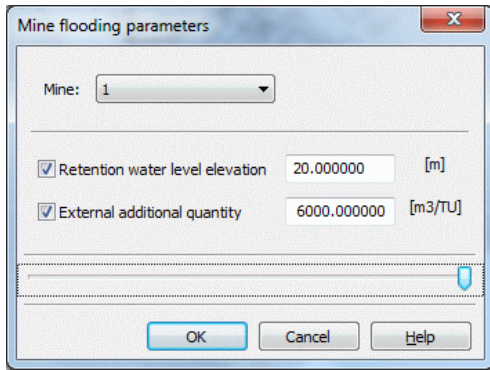
Definition of the water spreading volume - water spreading height-relationship

For each mine a continuous function must be defined which describes the dependence on the spreading volume and the water level in the mine:
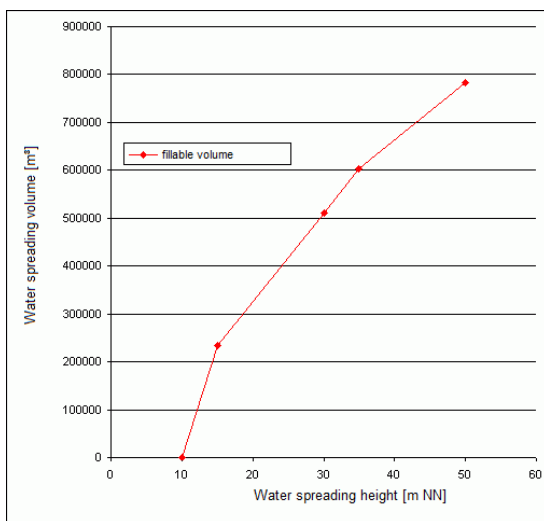
The function should be defined over the total interval of the water spreading procedure, that means, for all appearing values of the volume and the height during the simulation.

For the flooding calculation the water spreading volume-water spreading height-relationship for each mine must be prepared in the format CSV (separators disconnected) and placed in the working directory. The files have to be named after "grubeX.hv" where "X" is the corresponding zone number. This file is read automatically by the calculation module.

```
10.0,0.0
15.0,234900.0
30.0,511600.0
35.0,602800.0
50.0,783000.0
```

Results

Beside the known results (potentials, free surface, hydrograph, etc.) from the transient flow calculation SPRING creates for each cavity during the flooding simulation a volume hydrograph which describes for each time step the actual water spreading volume and by definition of the water level a volume hydrograph of the shrinking volume. Additionally the volumes for each boundary condition for each time step are saved. The file is created with the name "gruben.csv" (separators disconnected) in the working directory.

9.6.1 Example of a realization in SPRING

The software SPRING affords the simulation of flooding processes for mines and barrages. The following tutorial shows how to handle flooding processes of a mine.

The model files and the files needed for the computation are ready for download on our under "Tutorial (EN)" (

). The archive file "Tutorial_bsp_files.zip" includes a directory "HowTo_Grube" in which the files "fl0.zip", "fl1.zip" and "fl2.zip" are to be found.

The tutorial model is shown in the figure below. It contains three geological layers which describe two aquifers and one aquitard. In the second aquifer the mine is provided. The system is discretised up to nine node layers.
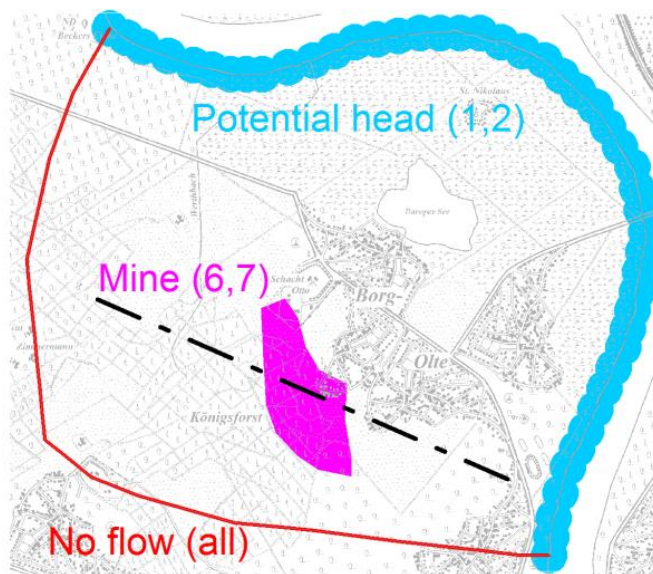
Fig. 255: Top view of the model with the boundary conditions of the individual layers
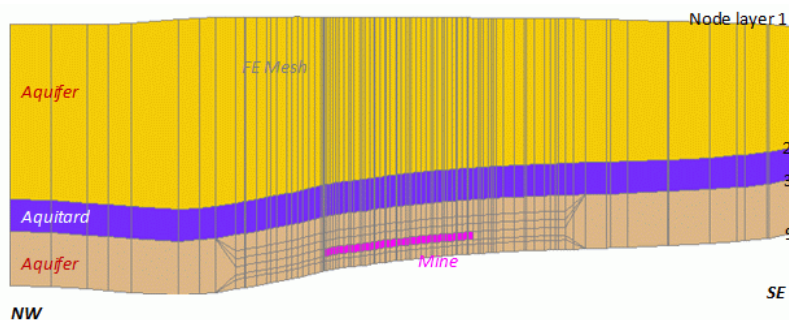
Fig. 256: Model basics (vertical cross section with 20x vertical exaggeration)

9.6.1.1   Initial conditions

Because flooding is always a transient simulation type you have to prepare the initial conditions first. Therefore it is necessary to define a seepage boundary condition at the nodes which describe the mine.

The example "fl0.zip" includes a structure (attribute type SICK) to set the boundary condition. Assign this structure to node layer 6 and 7.

Save the mesh and start the model checks with the item *Calculation → Model checking*. Continue with running a steady state flow simulation with the item *Calculation → Steady state flow.*
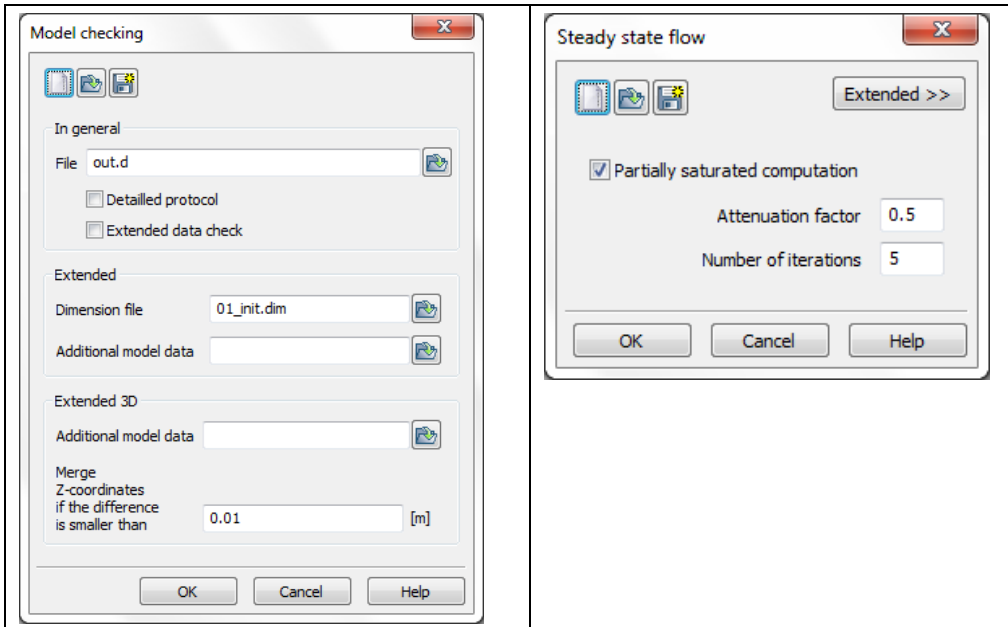
Fig. 257: Model checks (step 1) and steady state flow parameters (step 2)

The results of the computation are shown in the figure below. You can create this vertical cross section using the plot batch file "cs.bpl" (archive "fl0.zip").
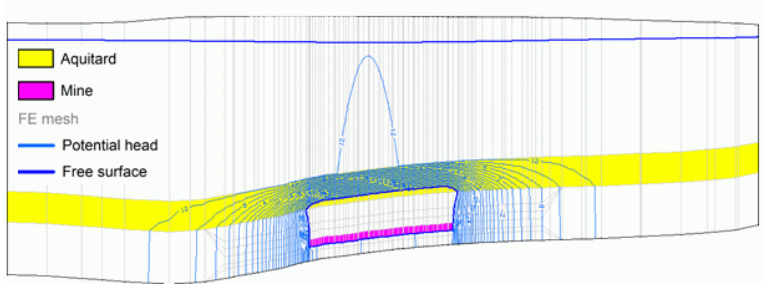
Fig. 258: Vertical cross section with results of steady state simulation

9.6.1.2   Simulation 1: Uncontrolled flooding

(Archive fl1.zip)

For a simulation of uncontrolled flooding processes it is necessary to mark the mesh nodes which represent the mine and to describe the correlation between mine water level and the water volume inside the mine. This correlation is called "h-V relation". Uncontrolled flooding means that the draining of the mine will be stopped and the mine will begin to fill with groundwater until reaching the equilibrium.

The attribute for defining the nodes of the mine is called GRUB. You have to assign this boundary condition at node layer 6 and 7 with the help of the structure with the attribute GRUB.

The attribute GRUB contains the zone number of the mine. In this example the number is "1". The corresponding h-V relation has to be defined in an ascii file which is called "grube1.hv" (file name = "grube"+zone number+".hv", see "fl1.zip"). The figure below shows the h-V relation of this example. For a better understanding during later simulation steps the gradient of the curve is displayed too.
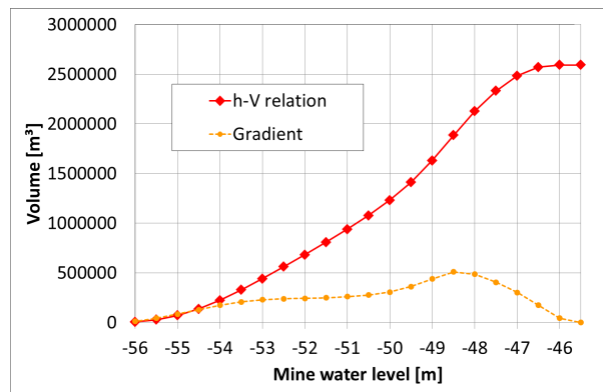
Fig. 259: Relation between mine water level and flooded volume of the mine

For transient simulations in SPRING a file including the transient boundary conditions is always needed. In this example no transient boundary conditions were changed. Therefore the file is "empty" (see also trans.txt in "fl1.zip").

File with transient boundary conditions (Flooding 1):

```
# SPRING tutorial
# Flooding
# Simulation 1

ZEITEINHEIT MENG JAHR
BEZUGSDATUM 01.01.2012

DATUM
02.01.2012
```

Before starting the simulation the definition of the initial values for the potential head of each node is needed. The necessary attribute is called EICH. Therefore import the results of the steady state flow computation (from directory fl0) using the item *Attributes → Import Data/Computation Results*:
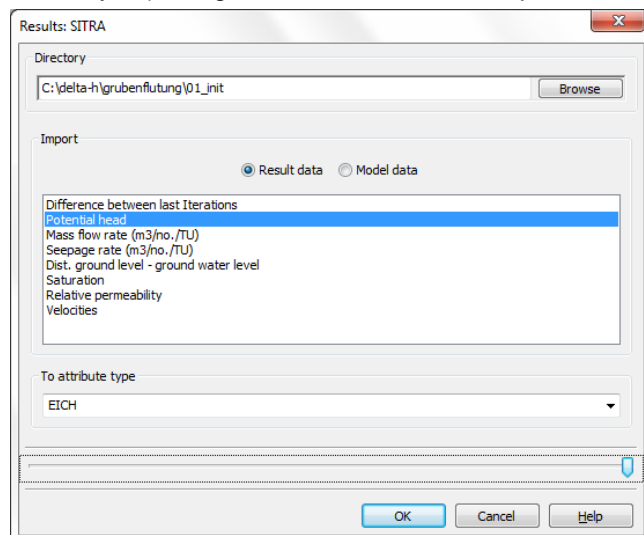
Fig. 5: Setting initial potential heads

After setting the initial potential heads save the mesh and start the model checks with the item *Calculation → Model checking*. Continue with running the transient flow simulation with the item *Calculation → Transient flow*:
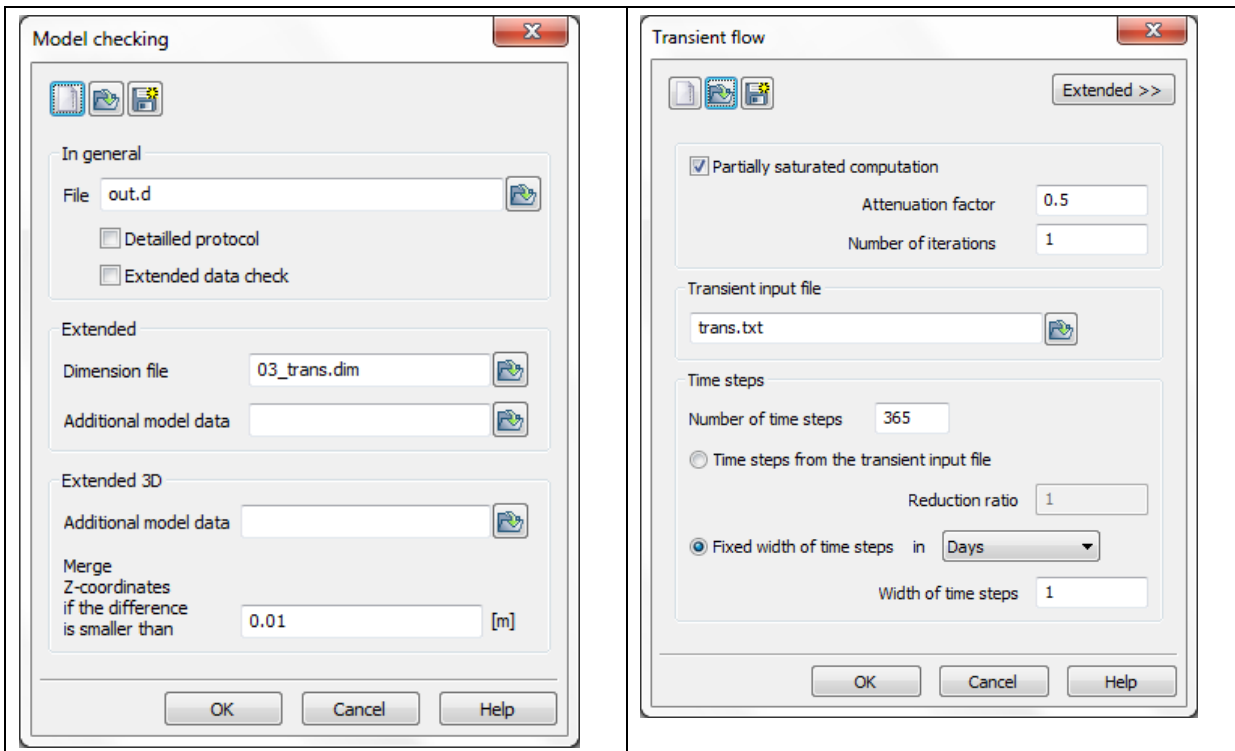
Fig. 6: Model checks (step 1) and transient flow parameters (step 2)

The resultant potential heads can be displayed by using the plot batch file "cs.bpl" for a vertical cross section and "pot56-7.bpl" for a horizontal cross section through layer 7 with the values of the 56th time step (see also figure below). The right figure also includes hydrographs of the resultant potential heads at selected nodes at the bottom of the mine. The dip in the curves is caused by reaching the end of the flooding process when the mine is filled completely with water. After this point the potential heads are rising up until reaching steady state conditions.
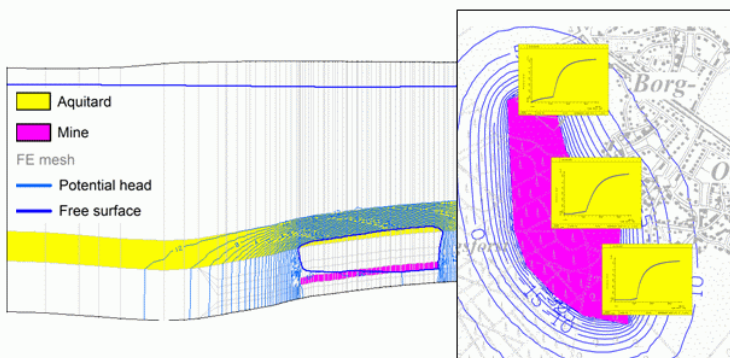
Fig. 260: Vertical cross section (left) and horizontal cross section through layer 7 (right) with results 56 days after begin of flooding (Flooding 1)

In addition to a standard transient computation simulating mine flooding creates a file called "gruben.csv". This file contains a table with a complete balance for the flooding box for each time step. This allows you to display the resultant balance curves with the help of a spreadsheet. For this example the evaluation was done with MS Excel (see file "gruben1.xls" in "fl1.zip") and is shown in the figures below.

The first figure presents the resultant mine water level and the ratios of the water balance depending on the several boundary conditions. For each time step the nodes which describe the mine were set to the boundary condition "seepage" or "potential head" depending on their current vertical position:

- "Seepage": The vertical position of the node (z-coordinate) is higher than the mine water level (unsaturated).
- "Potential head": The vertical position of the node (z-coordinate) is lower than the mine water level (saturated).

Both boundary conditions cause a mass flow rate respectively a mass flow which is exchanged between the mine and the aquifer. In the present example the sum of the mass flow from both boundary conditions is equivalent with the ground water inflow (see below).

The dip in the mass flow curves is caused by the gradient in the h-V relation. The mine water level reaches the top of the mine after 124 days.
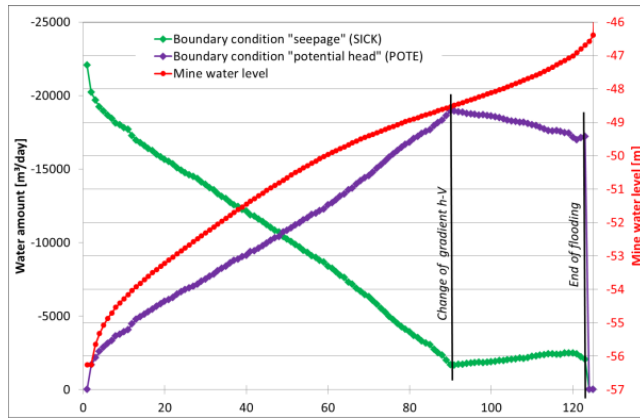
Fig. 261: Water level and water amount until reaching end of flooding (Flooding 1)
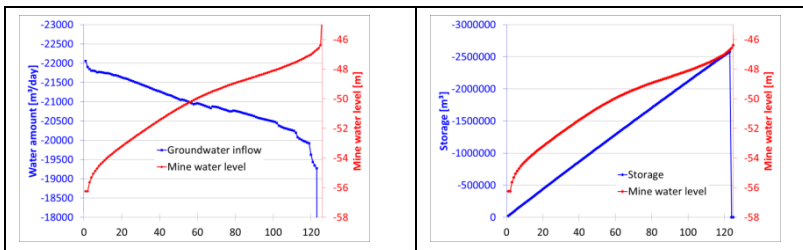
Fig. 262: Groundwater inflow (left) and storage (right) until reaching end of flooding (Flooding 1)

9.6.1.3 Simulation 2: Controlled flooding

(Archive fl2.zip)

In the second example the use of additional mine-specific transient boundary condition will be shown.

It is possible to define a retention water level elevation and external water supply for each mine. Both parameters can be changed in the transient input file. For external water supply a positive value has to be set if water should be pumped into the mine to make flooding faster. A negative value implies pumping water out of the mine. Both parameters can be initialized with dialog in *Attributes → Special features → Flooding parameters*. Note: If you want to vary one of these parameters during a simulation an initialisation is needed here.
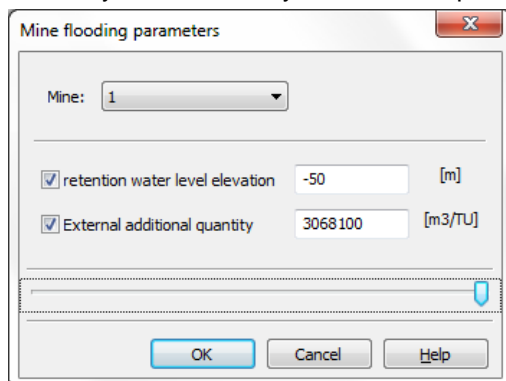
Fig. 263: SPRING dialog for managing flooding parameters

To show the working of these flooding parameters the following transient boundary conditions were set:

Table: Transient boundary conditions (Flooding 2)

Date	Time step	Boundary condition	Description
01.01.2012	0	HMAX = -50 m MENG = 3068100 m³/year	Begin of flooding up to mine water level -50 m with water supply of about 8400 m³/day
01.03.2012	60	MENG = -13149000 m³/year	Empty the mine by pumping 36000 m³/day (= 1500 m³/h)
01.07.2012	182	HMAX = --46.5 m MENG = 5259600 m³/year	Continue flooding up to mine water level -46.5 m with water supply of about 14400 m³/day
01.09.2012	244	MENG = 0 m³/year	Continue flooding without external water supply
31.12.2012	365		End of simulation

The file with the transient boundary conditions can be found in "fl2.zip" and looks as follows.

File with transient boundary conditions (Flooding 2):

```
# SPRING tutorial
# Flooding
# Simulation 2

ZEITEINHEIT MENG JAHR
BEZUGSDATUM 01.01.2012

DATUM
01.03.2012

MENG #empty mine with pumping rate 1500 m3/hour
     1-13149000.

DATUM
01.07.2012

HMAX #restart flooding upto mine water level -46.5m
     1    -46.5

MENG #add about 600 m3/hour
     1  5259600.

DATUM
01.09.2012

MENG #continue flooding without water supply
     1     0.5
```

The control of the simulation follows the steps of the example above.

The results of the present simulation are shown in the figures below. They base on the evaluation of the file "gruben.csv" and are included in the file "gruben2.xls" ("fl2.zip").

The given transient boundary condition for the water supply is shown by the curve "potential external water supply" (identical to definition of attribute MENG). The curve named "actual external water supply" shows the water volume which can be pumped in or out of the mine in a compatible way with the physics of the given system. E.g. the actual and potential water supply differs between reaching the end of emptying of the mine and continuing of the flooding. This is caused by the defined water supply (MENG = 36000 m³/day) which is not available after emptying the mine. Therefore the actual pumping rate after emptying the mine will be lower (about 22.000 m³/day).
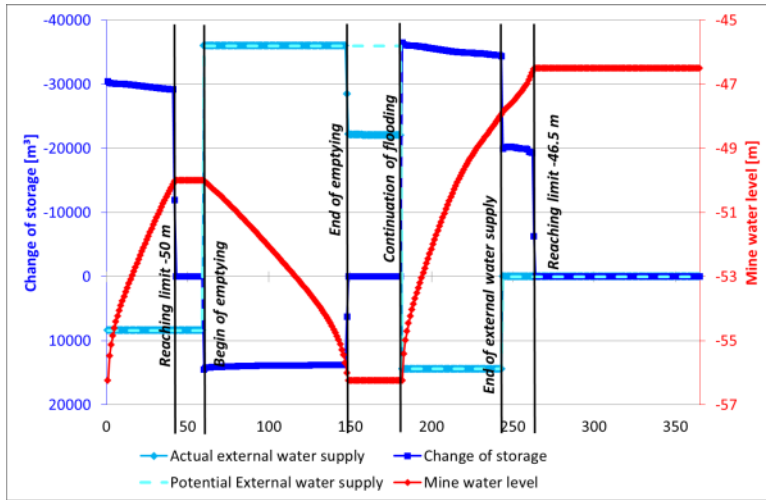
Fig. 264: Mine water level, change of storage and external water supply (Flooding 2)
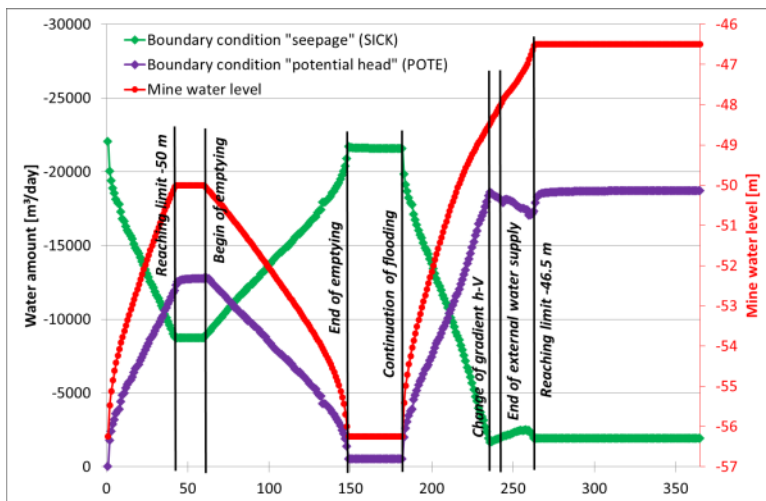
Fig. 265: Boundary condition dependent mass flow rates (Flooding 2)

10 Appendix

Here are explanations of the result files, background files, internal data identifiers and information about the file xsusi.kenn.

10.1 Structure and content of the result files

DiffEich.txt/csv

The file *DiffEich.txt* (or *DiffEich.csv*) is generated in the plot generation after a flow calculation if the corresponding menu item (*Origin - measured data*) is selected in the plot generation dialog. The file contains the following information:

x-coordinate, y-coordinate, difference (calculated - measured), measurement, calculated groundwater level

C<Node/Element-nr.>.csv

The file C<Knoten/Element-Nr.>.csv is generated in the plot generation after a transient flow calculation, if the corresponding menu item (*Type of illustration - Hydrograph*) is selected in the plot generation dialog. The file contains the following information:

calculated groundwater level, measurement, time step, name of the monitoring well (from file *sollpot*)

More detailed information on these two files can be found in chapter "SPRING-Menues" in "Plot - Measured data" (p. 83).

reaktion.csv

The file *reaktion.csv* is created at a flow calculation if the attribute BILK has been assigned at least to one node with the attribute POTE in the model file (*.net or *.3d).

The first line of the file contains the values of the attribute BILK in descending order. First, the balance areas of the first layer (if existing in the file *.net file) are listed in descending order. Then, the balance areas of the lower layers (from the file *.3d) follow in descending order.

The second line contains in case of a steady-state calculation:

The consecutive number of the time step (steady-state = 1.0), time (steady-state = 0.0), corresponding sum of the mass flow rate of the in the first line assigned balance areas.

After a transient calculation contain the second and the following lines:

The consecutive number of the time step, time of the transient input file, corresponding sum of the mass flow rate of the in the first line assigned balance areas

leakageBILK.csv

The file *leakageBILK.csv* is created at a flow calculation if the attribute BILK has been assigned at least to one node with the attribute LEKN or LERA in the model file (*.net or *.3d).

The first line of the file contains the values of the attribute BILK in descending order. First, the balance areas of the first layer (if existing in the file *.net file) are listed in descending order. Then, the balance areas of the lower layers (from the file *.3d) follow in descending order.

The second line contains in case of a steady-state calculation:

The consecutive number of the time step (steady-state = 1.0), time (steady-state = 0.0), corresponding sum of the leakage rate [m³/TU] of the in the first line assigned balance areas.

After a transient calculation contain the second and the following lines:

The consecutive number of the time step, time of the transient input file, corresponding sum of the leakage rate [m³/TU] of the in the first line assigned balance areas.

leakageBILE.csv

The file *leakageBILE.csv* is created at a flow calculation if the attribute BILE has been assigned at least to one node with the attribute LEEL in the model file (*.net or *.3d). The other structure is identical to the file *leakageBILK.csv*.

sick.csv

The file *sick.csv* is created at a flow calculation if the attribute BILK has been assigned at least to one node with the attribute SICK in the model file (*.net or *.3d).

The other structure is identical to the file *leakageBILK.csv,* the sum is the seepage rate in [m³/TU] for the respective balance area.

glei.csv

The file *glei.csv* is created at a flow calculation if the attribute BILK has been assigned at least to one node with the attribute GLEI in the model file (*.net or *.3d).

The other structure is identical to the file *leakageBILK.csv,* the value is the calculated groundwater level in [m] for the respective balance.

gruben.csv

The file *gruben.csv* is created in a flooding simulation.

For each cavity (attribute GRUB) a volume hydrograph is generated that describes the current water spreading volume for each time step and, if a water level is defined (attributes HMAX and MENG), a volume hydrograph of the shrinking volume.

In addition, the amounts of the individual types of boundary condition for each time step are stored.

The first line of the file contains the names of each column, sorted by cavity 1, cavity 2, etc.

The following lines contain the data:

Time step, current water spreading volume, shrinking volume (when defined), amount of the boundary condition seepage (node is unsaturated), amount of the boundary condition fixed potential head (node is saturated), amount of the boundary condition leakge (node is saturated).

10.2 Background Files

The different SPRING modules (calculation, data export and plot generation) communicate between each other using the binary background files. The modules automatically create the background files. They are binary and cannot be directly edited. The modules get access on the background files by using a general interface (SPRING-DTNLIB).

The background files have file names like aaa, bbb, ccc, ddd etc. The background file aaa (binary) is not managed in the DTNLIB interface.

For a faster access to the data, files are linked at least in pairs: each data file that contains individual data has an index file that contains a list of additional information (identifier, class, number, minimum and maximum values, time step etc.).

The current program version uses the following files:

- Input data:
  bbb (index)
  ccc (data)

- Steady-state results and interpolation results:
  ddd (index)
  eee (data)

- Transient results:
  fff (index)
  ggg (data 1: last time step)
  hhh (data 2: all time steps)
  iii (data 3: all time steps)
  jjj (data 4: all time steps)
  kkk (data 5: for calculations with transient terrain settings)

- Results from data export:
  lll (index)
  mmm (data)

- Data for inverse modelling:
  nnn (index)
  ooo (data)

10.3 Internal Identifiers

The data in the background files are identified by a so-called identifier number. With the help of the identifier number and the SPRING-interface DTNLIB, each module knows how to store and read the data, how to plot the data, and their meaning.

The user notices the internal data management with identifier numbers only in the batch files for the plot generation or the data export: These files define the data to be plotted with the help of identifiers. The following tables contain all identifying information.

One line of a table contains the following information:

- column 1: identifier number (< 1000)
- column 2: class: the data class manages the storage and reading procedures:
  k = node wise data
  K = node wise data for the nodes of the topmost layer (3D-model)
  e = element wise data
  E = data for the elements of the topmost layer (3D-model)
  r = data along a polyline or a group of nodes
  R = data for the nodes/elements of the topmost layer (3D-model)
  1 = 1D fracture data
  2 = 2D fracture data
  x = for nodal / elemental data
  without class = for data without direct meaning, e.g. index arrays
- column 3: description

Model data after model checking

Identifier number	class	description
2D mesh data:		
1	k	Mesh boundary
2	k	Node numbers
3	e	Element numbers
4	k	x-,y-coordinates
3D mesh data:		
11	k	3D mesh boundary
13	k	z-coordinates
Fracture description:		
22	1	1D fracture numbers
23	1	1D fracture aperture
24	1	1D fracture thickness
26	2	2D fracture numbers
27	2	2D fracture aperture
Plot data:		

32	x	Markers
33	k	text at nodes
34	e	text at elements
Internal data:		
45	k	Layer number for nodes
46	e	element numbers in the layers beneath
47	e	Layer number for elements
Analysis data:		
51	R	Control lines
52	k	PRAN-nodes
53	r	Mass balance nodes
54	E	Mass balance elements
55	r	Group of mass balance nodes with numbers
56	e	Group of mass balance elements with numbers
Parameters:		
61	e	permeabilities (m/time unit)
62	e	vertical permeabilities (m/time unit)
63	e	Storage coefficient
64	k	Porosities
65	e	Dispersivities
67	k	Storage coefficients
68	1	1D fracture dispersivities
69	2	2D fracture dispersivities
70	k	Zones for K_d (XTRA)
Calibration data:		
71	e	minimum permeability (m/time unit)
72	e	maximum permeability (m/time unit)
73	k	initial potential head
75	e	Permeability zones
76	k	Zones for the storage coefficient
77	r	Zones for the leakage at nodes
78	e	Van-Genuchten-zones
79	e	Van-Genuchten-parameters
80	e	Initial saturation
Geology:		
81	E	Initial thickness
82	E	Lower boundary
83	E	Upper boundary
84	K	Thickness of the impermeable layer
85	K	Ground level
86	K	Terrain settings as a result of subsurface mining

87	E	Maximum thickness
boundary conditions for potential equation:		
91	k	boundary potential head
92	r	equal potential head
93	k	Leakage coefficients (m²/nodes/time unit)
94	e	Leakage coefficients (m²/m²/time unit),(ms)
95	r	Leakage coefficients (m/time unit)
96	k	receiving water course level
97	r	maximum exfiltration rates (m³/m/time unit)
98	r	maximum infiltration rates (m³/m/time unit)
99	e	maximum exfiltration rates (m³/m²/time unit)
100	e	maximum infiltration rates (m³/m²/time unit)
101	r	seepage surface node
102	k	Leakage coefficients (m²/node/time unit)
103	k	maximum exfiltration rates (m³/node/time unit)
104	k	maximum infiltration rates (m³/node/time unit)
105	e	Leakage coefficients (m²/element/time unit)
106	e	maximum exfiltration rates (m³/el./time unit)
107	e	maximum infiltration rates (m³/el./time unit)
108	k	break hight
109	k	Ratio of the leakage coefficient EXF/INF
110	e	Ratio of the leakage coefficient EXF/INF
Source data:		
111	E	recharge rate (m³/El./time unit)
112	k	Sinks and sources (m³/node/time unit)
113	r	Boundary in- and outflow (m³/time unit)
boundary and initial conditions transport equation:		
120	k	Inflow rate (per s)
121	k	defined concentrations
122	k	Source concentrations
123	k	Initial concentrations
124	k	defined concentrations (XTRA)
125	k	Source concentrations (XTRA)
126	k	Initial concentrations (XTRA)
127	k	Inflow rate (per s) (XTRA)
General data:		
131	k	general nodewise data
132	e	general elementwise data
155	k	Cavity nodes
156	k	Cavity water level
157	k	Cavity addition volume

Steady state results or interpolation results after a calibration, calculation of a steady state flow or an interpolation:

Identifier number	class	description
results of the potential equation / concentration equation:		
201	k	Potential heads
202	e	Potential heads at element centers
203	k	Concentrations
204	e	Velocities
205	e	Velocities
206	k	Concentrations (XTRA)
208	1	Velocity of 1D fractures
209	2	Velocity of 2D fractures
further results:		
210	k	Seepage rates (m³/node/time unit)
211	k	Mass flow rates (m³/node/time unit)
212	R	Control line rates (m³/time unit)
213	k	Leakage rates (m³/node/time unit)
214	e	Leakage rates (m³/element/time unit)
215	K	distance between ground level and groundwater level
216	k	Saturation
218	E	Changed lower boundary
219	E	Iterated thickness
220	K	Iterated nodewise thickness
Path line data:		
221		Path lines (2D)
222		Path lines (3D)
Peclet numbers:		
225	e	Peclet number longitudinal
226	e	Peclet number transversal
Spezial results of the calibration:		
231	e	iterated permeability (m/time unit)
232	e	iterated vertical permeability (m/time unit)
Interpolation results:		
240	K	Interpolated node data
241	E	Interpolated element data

Transient results:

Identifier number	Class	Description
Time steps:		
300		Date/time, time unit, referred date
results of the potential equation / concentration equation: last time step:		
301	k	Potential heads
302	e	Potential heads at element centers
303	k	Concentrations
304	e	Velocities (2D)
305	e	Velocities (3D)
306	k	Concentrations (XTRA)
309	2	Velocities of 2D fractures
further results: last time step:		
310	k	Seepage rates (m³/node/time unit)
311	k	Mass flow rates (m³/node/time unit)
312	R	Control line rates (m³/time unit)
313	k	Leakage rates (m³/node/time unit)
314	e	Leakage rates (m³/element/time unit)
315	K	distance between ground level and water table
316	k	Saturation
Path line data:		
321		Path lines (2D)
322		Path lines (3D)
Peclet numbers/Courant numbers:		
325	e	Peclet numbers longitudinal
326	e	Peclet numbers transversal
327	e	Courant numbers
results of the potential equation / concentration equation: time-dependent:		
400	k	Drawdown z-coordinates
401	k	Potential heads
403	k	Concentrations
404	e	Velocities (2D)
405	e	Velocities (3D)
406	k	Concentrations (XTRA)
409	2	Velocities of 2D fractures
further results: time-dependent:		
410	k	Seepage rates (m³/node/time unit)
411	k	Mass flow rates (m³/node/time unit)
412	R	Control line rates (m³/time unit)
413	k	Leakage rates (m³/node/time unit)
414	e	Changed lower boundary

results as a function of time:		
430	k	Inflow at a water course node (VKNO, m³/node/time unit)
450	k	z-coordinates
451	k	Potential heads
453	k	Concentrations
456	k	Concentrations (XTRA)
Transient input data:		
470	K	terrain settings as a result of subsurface mining
471	k	Defined potential heads
472	k	Receiving water course level
473	E	Recharge rates (m³/element/time unit)
474	k	Sinks and sources (m³/node/time unit)
475	k	Defined concentrations
476	k	Source concentrations

Results of the data export

Identifier number	Class	Description
523		Stream lines

Data of the inverse modelling

Identifier number	Class	Description
steady state calculation:		
601	k	Potential heads / permeabilities (hori.)
602	k	Potential heads / permeabilities (vert.)
603	k	Potential heads / storage coefficients
604	k	Potential heads / nodewise leakage
611	k	Mass flow rate / permeabilities (hori.)
612	k	Mass flow rates / permeabilities (vert.)
613	k	Mass flow rates / storage coefficients
614	k	Mass flow rate / nodewise leakage
transient calculation:		
651	k	Potential heads / permeabilities (hori.)
652	k	Potential heads / permeabilities (vert.)
653	k	Potential heads / storage coefficients
654	k	Potential heads / nodewise leakage
661	k	Mass flow rates / permeabilities (hori.)
662	k	Mass flow rates / permeabilities (vert.)
663	k	Mass flow rates / storage coefficients
664	k	Mass flow rates / nodewise leakage

10.4 Information of the File xsusi.kenn

Definition of Data Identifiers

A complete model requires input data at the mesh nodes and elements. These data are known as mesh properties (permeability, thickness, geographic height, in- and outflow rates etc.). Each parameter has a data identifier (four letters) to manage the different kinds of data. They are described in Chapter: Data Structure of the Groundwater Model –Description of Data.

The data management in SPRING requires information on whether the data are related to nodes or elements and whether they have some special characteristic to be considered, e.g. data for polygonal lines or data which are limited to the topmost layer in a 3D model. With some exceptions, more information on the meanings is not necessary when creating the model. At this point, it does not matter whether the values of BERG are less than zero, or whether the geographic height at some nodes lies beneath the z-coordinate of the topmost layer. Hence, the data identifications are initialized directly with the help of the file xsusi.kenn. (This is different from the procedure in the model checking.)

The file is installed here:

"C:\Users\Public\Documents\SPRING\Konfig" (Windows 7 or Windows Vista). This path depends on the operating system or the partitioning of the computer (e.g. in Windows XP: "C:\Documents and Settings\All Users\Documents\SPRING\Konfig").

Besides the general classification of the reference nodes or elements, it is the distinction between:

■ Simply used data: each node or element can receive only one value per layer for this type of data (eg: a node of a layer can have only one potential head (attribute POTE) and an element of a layer can only have one k-value (attribute KWER)). If a node or element of such a data type has been for any reason assigned twice, a specific query dialog appears. In it, the user can determine whether the first value (no overwriting) or the last value in the model file (ALL overwriting) should be stored.
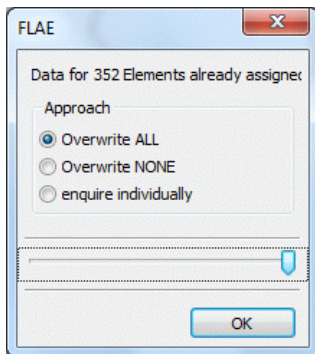
■ Group data: each node or element can receive multiple values per layer for this type of data (eg, a node can be assigned to several mass balance areas (eg. BILA).

■ Polyline data (at node data): The data value is defined for a polygon of nodes, ie the order of node numbers plays a crucial role in the type of data (eg. Attributes LERA, MXKI, MXKE,…).

■ 2D or 3D data: There are data types for which an allocation to deeper layers is not allowed. These have to be initialized as 2D data so that an allocation on layer numbers > 1 is not possible (eg: GELA, MAEC, UNTE, UNDU, ...).

In the file xsusi.kenn are the identifiers initialized as follows:

At first, the line ".IDENTIFIER." is sought in the file. Thereafter, the following data is read per line until the end of file or finding a line beginning with ".END.":

Column 1 - 4	KENN	Four letter identifier of the data type
Column 6	k, e, 1, 2	k = node related e = element related 1 = 1D fractures 2 = 2D fractures
Column 7	empty, g, s	empty = single used data g = group data s = polyline data (only in combination with k)
Column 8	2, 3	2 = 2D data 3 = 3D data
from Column 10	Any text as information for the identification	

Extract of the file xsusi.kenn as an example of the attribute description:

```
.IDENTIFIER.
ABRI k 3 Tear-off heights
AKON k 3 Initial concentrations
ASAT e 3 Initial saturation
BERG k 2 Terrain settings
BILE e 3 Bilance elements
BILK k 3 Bilance node
DISP e 3 Dispersitivity
EEEE e 3 General elementwise data
EICH k 3 Nominal potential head from calibration
FLAE e 2 Areal seepage (recharge rate)
FLUR k 2 Distance between ground level and water level
GELA k 2 Ground level
GLEI ks3 Equal potential head
GRUB k 3 Balance node for Flood Method
KKKK k 3 General nodewise data
KMAX e 3 Max. permeability (m/s)
KMIN e 3 Min. permeability (m/s)
KNOT k 3 Nodewise in-/outflow (+/-)
KONT ks3 Controll Lines
KONZ k 3 Inflow concentrations
KSPE k 3 Storage coefficient (nodewise)
KVRN e 2 Land use RVR
KWER e 3 Permeability (m/s)
KWEV e 3 Vertical permeability (m/s)
.END.
```

Unknown or user specified identifiers

The user can also specify any identifiers in the structure file. When opening the project with SPRING in this case, the following input window appears:
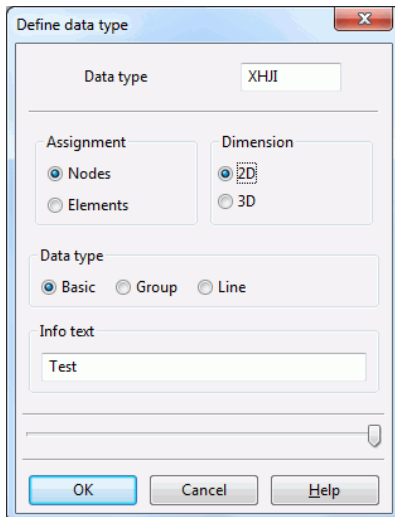
The user can then define whether it should be a node or element related data of the type "Basic", "Group" or "Line". If this new data type should be made permanent, it must be entered in the (user-specific) file xsusi.kenn.

Default parameters of the representation of the attributes

With the help of the file xsusi.kenn, you can modify the default parameters of the representation of the attributes. The default settings for displaying attributes are defined as:

- individual used node data (eg. GELA, EICH, POTE): 10 contour lines with equidistant partition
- individual used element data (eg. FLAE, KWER): an area plot with 10 colour intervals (equidistant partition)
- group data (eg. BILA): a plot of circles with 10 colour intervals (equidistant partition)
- polyline data (eg. LERA, MXKI): a plot of polylines with 10 colours at maximum (equidistant partition)

Especially for the single used data that are usually only defined locally at single nodes (eg POTE or KNOT) the default settings are not usable.

In the file xsusi.kenn the default parameters of the representation of the attributes are initialized as follows:

After the definition of the data identifiers the line beginning with „.DARSTELLUNG." is sought. Thereafter, the following parameters are read per line until the end of file or finding a line beginning with ".ENDE.":

Line 1:

Column 1: * : stands for the initialization string of a new identifier

Columns 2 – 5: KENN: stands for the four letter identifier of the data type

Column 7: layer number

Line 2:

Column 1:

    k = plot of circles

    f = area plot

    i = plot of contour lines

Thereafter follows the number of the colour palette (plot of areas or circles) or the colour number for the plot of contour lines.

For a plot of circles you can then define the height of the marker $r_{max} <= 6.0$ (for the maximum found value). Should all circles have the same height the chosen radius has to be $r_{max} = 6.0$.

Line 3:

These columns are without formatting. At first the parameter for the partition is defined:

0: equidistant partition

1: from/to/number
2: from/to/number (logarithmical)
3: from/to/distance
4: Single values

After the partition type 0, 1, 2 and 4 follows the number of intervals or the numbers of the single values.

In case of the partition type 1 to 4 follows:
Line 4:
Type 1 and 2: three values free formatted: from, to, number
Type 3: three values free formatted: from, to, distance
Type 4 with k or i: number of values for the plot of contour lines or for the interval limits of the circle plot (free formatted).
Type 4 with f: twice the number of values with the minimum and maximum interval limit (free formatted)

Extract of the file xsusi.kenn as an example of the representation description:

```
.DARSTELLUNG.
*AKON 1    /* Equal-value lines pen 5 */
i 5
0 10
*ASAT 1    /* Equal-value domains with palette 0 */
f 0
0 30
*BERG 1    /* Equal-value lines pen 5 */
i 5
0 10
*BILE 1    /* Equal-value domains with palette 0 */
f 0
0 10
*BILK 1    /* Circles with equal radius with palette 0 */
k 0 6.0
0 10
*DISP 1    /* Equal-value domains with palette 0 */
f 0
0 10
*EEEE 1    /* Equal-value domains with palette 0 */f 0
0 10.
```

11 Index