Isoliner3D - a 3D window and volume tools for QGIS

Contents

What this is **3**
- Installation . . . 3
- Where to start . . . 4

The data the module works with **4**
- A surface . . . 4
- A bed grid: a multiband raster . . . 4
- A cube of values . . . 4
- An isosurface . . . 4
- Voxels . . . 4
- A block model . . . 5
- A shell, that is, a body . . . 5
- Drawings on sections . . . 5
- Holes and samples . . . 5
- A texture . . . 5
- An area mask . . . 5
- Result fields . . . 6

The viewer **6**
- The coordinate system . . . 6
- Scene properties . . . 6
- The centre of rotation . . . 7
- Fly to and orbit . . . 7
- Layer properties . . . 7
- Where the colour comes from . . . 7
- Lines as tubes . . . 7

Isosurfaces and voxels **8**
- Shells in a table . . . 8
- The cap at the edge of the cube . . . 8
- Cleanup . . . 8
- Shells into a project layer . . . 8

Exporting the scene **9**
- Spin and capture . . . 9

The tools **9**
- 1.01 Assemble a bed grid . . . 9
  - When there is no floor . . . 10
  - What to do with it next . . . 10
- 1.02 Bed calculator . . . 10
- 1.03 Bed grid to a block model . . . 11
  - A multi-bed grid . . . 11
- 1.04 Surfaces to 3D (meshes) . . . 11

1.05 Domains to a bed band . . . 12
1.06 Reserve difference (write-off) . . . 12
1.07 Create sample data (demo) . . . 12
1.08 A map for a texture (demo) . . . 13
2.01 Demonstration boreholes in three dimensions . . . 13
2.02 Interpolating points in 3D . . . 14
2.03 Cube to a block model . . . 16
2.04 Cube body as voxels . . . 16
2.05 Check of the interpolation . . . 17
2.06 Kriging in three dimensions . . . 18
2.07 MBA in volume . . . 19
2.08 Beds from sections . . . 21
2.09 A demonstration drift (demo) . . . 22
What it is for . . . 23
What is inside . . . 24
The numbers to expect . . . 24
2.10 Surfaces from cross sections . . . 24
What it is for . . . 25
How it counts . . . 26
Separate measurements . . . 26
What accuracy to expect . . . 26
2.11 Boolean operations on shells . . . 26
What it is for . . . 27
Why over cells . . . 27
The price . . . 27
What is checked on input . . . 27
2.12 Selection by a shell . . . 27
What it is for . . . 29
How it counts . . . 29
The check worth attention . . . 29
The limit of the method . . . 29
2.13 A shell from a bed grid . . . 29
What comes out . . . 30
About the volume . . . 30
2.14 Zonal statistics of shells . . . 30
What it is for . . . 31
How it counts . . . 31
Repeated runs . . . 31

Typical tasks **31**
2.15 Formation bodies from outlines . . . 31
2.16 A demonstration river network (demo) . . . 32
From boreholes to reserves . . . 33
From horizons to a bed . . . 33
A section on the drawing and in volume . . . 33
Where the coordinates come from . . . 34
How the roof and the floor are taken . . . 34
The contact of neighbouring beds . . . 34
A lens inside a bed . . . 34
How far to trust the surface . . . 35
A mask for one bed . . . 35
The order of the work . . . 35
Formations side by side, not one above another . . . 35
A surface from cross sections . . . 35
From a DEM to a body: a pit and spoil heaps . . . 36
Reserves left after mining . . . 36

What has changed between surveys 37
Showing it to someone who is not a geologist 37
Checking the data before the model 37

Typical situations **37**

Neighbouring plugins **38**

Licence and developer **38**
Where to look 38

What this is

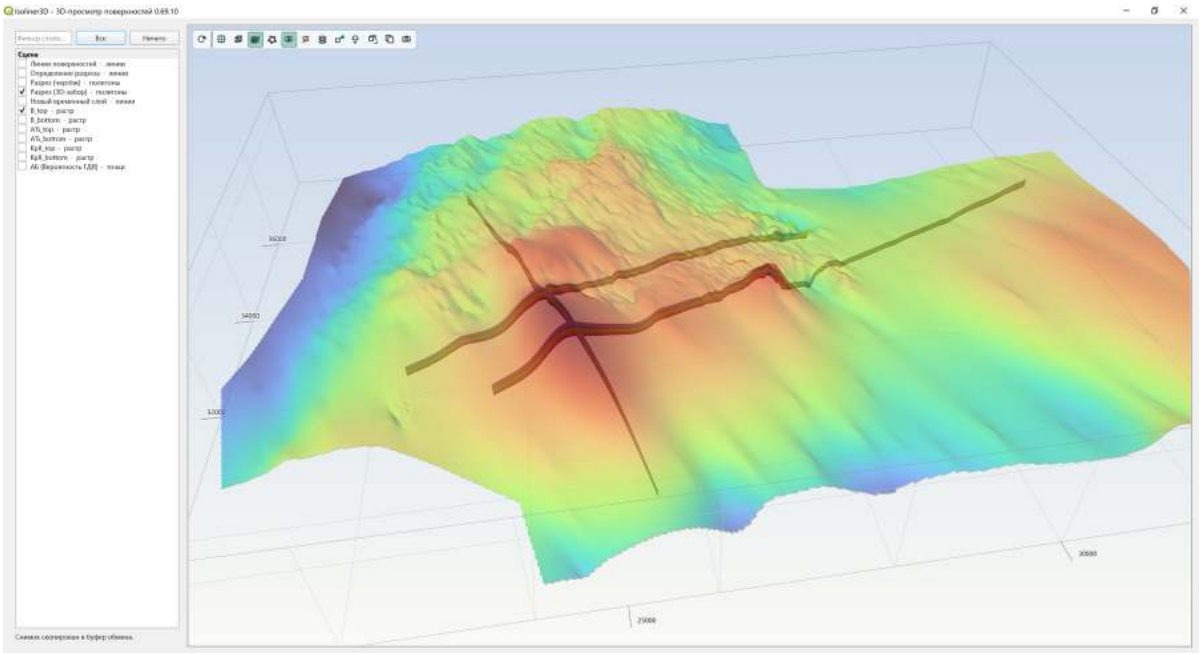


Figure 1: Relief with hillshading and three section fences across it, inside a coordinate box with elevation labels.

Isoliner3D shows the data of a project in three dimensions and computes from it. Rasters become surfaces, multiband grids become watertight beds, boreholes become coloured shafts, polygons with Z become bodies. Apart from that there is a set of tools in the Processing panel: assembling a bed grid, a block model, interpolation in volume, isosurfaces, volumes.

The module depends neither on the built-in 3D view of QGIS nor on Qt3D: the drawing goes through libraries bundled with it, nothing extra to install.

Interpolation in plan, contours, terrain and drawn sections stay with the Isoliner plugin, of which Isoliner3D is a companion.

Requirements: QGIS 3.16 or newer, Qt5 or Qt6.

Installation

Plugins - Manage and Install Plugins - Install from ZIP, choose `isoliner3d.zip`. QGIS need not be restarted.

Where to start

Without your own data: the tool **2.01 Demonstration boreholes in three dimensions** puts samples into the project, then **2.02 Interpolating points in 3D** makes a cube out of them, and the viewer shows an isosurface over that cube.

With your own: open the window from the toolbar, tick the layers in the list and press refresh. A raster shows as a surface, polygons with Z as a body.

The data the module works with

The module handles several kinds of data, and half of the misunderstandings come from mixing them up. Briefly: what each one is, how it is stored in QGIS, who makes it and who consumes it.

A surface

An ordinary raster where the value of a cell is an **elevation**: the roof of a bed, the floor, the terrain, a water table. A single surface has no volume: to get one a second surface is needed to bound the body from below.

A bed grid: a multiband raster

The same raster, but with several bands in an order that is the convention of the module: **band 1 the roof, band 2 the floor**, then a pair for every further bed. In a single-bed grid bands 3 and on are parameters: grade, thickness, anything numeric.

A pair of bands is a body: their difference is the thickness, the thickness times the cell area is the volume. That is why 1.02 can count reserves and the scene can show a body.

A common confusion: **a bed grid is not a cube**. In a cube the bands are levels along Z, in a bed grid they are the boundaries of a body. Mistaking one for the other, the scene would draw a box at the zero elevation, and it says so plainly.

A cube of values

A raster whose bands are **horizontal slices along Z**, from the bottom up. The Z layout - the elevation of the lowest slice and the step between them - is written into the metadata, so the cube knows where it stands in space.

A value in a cell of a cube is a parameter at a point in space: a grade, a probability, a density. A cube is not a body: it is filled with values everywhere, and a body appears once a cut-off is set.

An isosurface

The boundary drawn through a cube at a given level: “where the grade equals five per cent”. A closed isosurface is a body, and a volume is computed from it.

Voxels

The same cube shown not by a smooth boundary but by the **little boxes of the cells**. The difference is not in the data but in the honesty of the display: an isosurface interpolates between the nodes and looks smooth, a voxel shows exactly the resolution that exists.

Voxels also carry the boolean operations of 2.11: bodies are rasterised into cells because an operation over cells always gives a closed result.

A block model

A layer of **points** - the centroids of the blocks - each with its own fields: the volume of the block, the grade, the type of ore. This is the familiar form of counting reserves: sum the volumes, weight the grades, get the tonnage.

A block model is coarser than a cube by its nature: a block is either wholly inside a contour or wholly outside. The difference between the volume of the blocks and the volume of the shell is the measure of that coarseness, and 2.12 prints it.

A shell, that is, a body

A layer of **polygons with height** (MultiPolygonZ) where every polygon is a triangle of the surface of a body. A closed shell bounds a volume, and the volume is computed by an exact formula over the shell itself, without cells.

The key property is **closedness**. A shell that is not closed has no inside, no volume can be computed from it, and the tools refuse it. The `closed` field in the attributes shows this for every body.

Drawings on sections

Polygons and lines with real Z lying in the plane of a section. A polygon is a body: the ring runs along the roof one way and along the floor back, and two surfaces are taken from it (2.08). A line is one surface (2.10).

The position of the sections is nowhere set: it is entirely in the geometry, in the vertices. **A flat drawn section is no good**: its X and Y are coordinates on the sheet, and there are no elevations at all. The geometry type must be `PolygonZ` or `LineStringZ`.

Holes and samples

Points with height and numeric fields. For interpolation what matters is not the point as such but that the samples of one hole lie along a stem: neighbours half a metre away along it and tens of metres away to the side. That is why the check 2.05 can leave out a whole hole - an estimate over a single sample flatters the model several times over.

A texture

A picture of a map stretched over a surface in the scene: a plan, a satellite image, a drawing. It is display only - nothing is computed from a texture - but it shows where the model falls against a familiar picture.

An area mask

A polygon the result of the interpolation is clipped by. Between the sections and beyond the data the surface goes where the interpolation drew it, and the mask says how far to trust that. In 2.08 and 2.10 a mask can be tied to a particular bed or surface, or the tool can be asked to take the area from the data themselves - the convex hull of their own samples.

A mask only clips the surface, and at its edge the body of the bed drops down as a vertical wall. If the bed has to pinch out towards a boundary known on the plan together with its elevations, 2.08 takes it as the outline on the plan.

Result fields

The fields of the layers the module creates have Latin names that do not depend on the language. Expressions, labels and styles refer to them, and a project made in a Russian QGIS opens in an English one. The attribute table, the forms and the identify window show field captions in the interface language: **Volume, m³**, **Bed roof, m**, **Grade**.

The captions are written into the GeoPackage itself as well. A result file opened on its own, without a project, shows the same captions. The file keeps the language it was made in.

A field that came from the input is neither captioned nor changed by the module. If the input field has a caption, the result gets the same one.

Numbers in the fields the module creates are rounded. Coordinates, elevations, lengths and sizes go to the centimetre. Volumes, masses, grades and residuals keep four significant digits, and the integer part is never rounded: a shell volume of 39524.12 m³ is written as 39524, a grade of 25.3745 as 25.37, a grade of 0.0045678 as 0.004568. The fields of the input layer are not rounded.

The viewer

The list of project layers with checkboxes is on the left, the scene on the right. A tick says what to show. The scene is built by the refresh button, not by the tick itself: on a large project a rebuild takes seconds, and doing it on every click is pointless.

Mouse: the left button rotates, the wheel zooms, the middle one pans.

The coordinate system

The scene works in the coordinates of the project. The origin is shifted to the middle of the extent of all visible layers: without that, numbers of six digits lose precision on the graphics card and the model starts to shiver.

The vertical scale is one for the whole scene. A common origin and a common factor are computed over all visible layers at once and applied alike to rasters, polygons, lines and points. That is why a section fence lies on the surface it was built from at any exaggeration.

Vertical exaggeration stretches the elevations. It is a way of looking, not a property of the data.

Scene properties

Opened by a button on the toolbar. The sections are **View**, **Clipping**, **Appearance** and **The coordinate box**. All the settings are saved into the project.

Hillshading is not decoration: a surface coloured by a ramp is drawn with vertex colours and no light at all, so the relief within one shade is lost. The gradient background is on by default.

For a bed grid the **height band** is read as a place in the stack: a roof band shows that bed and everything below it, a floor band shows that bed and everything above it.

Bodies are lit by three sources. With a single one everything turned away from it falls into shadow completely, and half of a bed body is just that - the side walls and the floor - so the scene comes out dark. The main source draws the shape, the second lights from the other side at half strength, the third from below keeps the bottom from sinking. The hillshading keeps a single source: with three it goes flatter, and the relief within one shade of the ramp reads worse.

The centre of rotation

The scene turns about the centre of the extent of the data, and that is no way to look at a single detail: it leaves the frame faster than it turns. A **right click** on the scene moves the centre of rotation to the point you pointed at.

The picture does **not** change: the place of the camera is derived from the centre, the distance and two angles, so a plain move of the centre would carry the view away. Here the distance and the angles are recomputed back, and the camera stays exactly where it was. Only the point the rotation goes about changes. A right click **beyond the area of the data** puts the centre back on the whole scene - that is the “show me everything”. A click that misses the object but falls within the area leaves the centre alone: that is an ordinary near miss on a thin body, and a reset there would look like the view walking off on its own. The boundary is drawn with a quarter of the extent to spare.

If the scene fills the frame and there is no way to click beyond the area, use **Centre of rotation - the whole scene** in the menu of the scene list.

To look at a detail: fly to the layer, right-click the place, switch the orbit on. The rotation will go about the chosen point.

The right button is taken only on a click without movement: on a drag the camera still keeps it. In drawing mode the right button undoes the last vertex, and the centre is not moved there.

Fly to and orbit

The right button on a row of the scene list offers **Fly to** and **Orbit the current centre**. On a large site there is no way to find a small object: it takes up a pixel, and spinning about the common centre of the scene only leads away from it. Flying moves the centre of rotation onto the layer itself and takes the distance of the camera from its extent. Orbiting aims at nothing, it turns about the current centre. Were it to aim at the layer, the centre chosen by a click would be wiped out.

The extent is taken from the layer and converted into the coordinates of the scene: the scene lives shifted, its centre stands at zero, and an extent taken as it is would lead the camera away by the whole shift.

Layer properties

Opened by a double click on a layer in the list. Only the fields that work in the chosen mode are shown.

Where the colour comes from

The scene asks the data itself, in this order: the **`color`** field if the layer has one, then the layer symbology asked per feature, then a single colour for the whole layer.

Bodies take the opacity together with the colour: the alpha of the symbol colour, the symbol opacity and the layer opacity from its properties, as on the map. So the voxel classes from 2.04 can be made translucent right in the layer style. Solid bodies are drawn first and translucent ones over them, so what lies inside shows through a transparent shell. The layer transparency in the scene properties is multiplied by the transparency of the style.

Lines as tubes

A screen line is drawn a few pixels wide and looks the same at any distance from the camera, so its thickness shows nothing. A line layer has the **As a tube** display in its properties: a tube has a diameter in metres, and the field value reads by size, as a line width does on the map.

Tube thickness field is a numeric field. The smallest value gives **Diameter from**, the largest **Diameter up to**. Without a field all tubes of the layer get **Diameter up to**. **Thickness scale** by cross-section area makes the area, not the diameter, proportional to the value: the eye reads a tube by its area, and large values do not swamp the picture. The scale by diameter makes the diameter proportional.

The colour and transparency of a tube come from the layer style, as for bodies, so one value can be shown by thickness and another by colour. The tubes of a layer are built into one mesh, the wall keeps its thickness at bends and the joints have no gaps. On layers with hundreds of thousands of vertices the number of faces around the tube drops, so the scene keeps rotating smoothly.

Tool 2.16 makes an example to try: a river network where the discharge grows downstream tens of times.

Isosurfaces and voxels

A cube of values is a multiband grid where a band is a horizontal level. Tools 2.02, 2.06 and 2.07 write such grids.

Shells in a table

A row per level: the level, the colour, the opacity, the tick that shows it. An empty cell takes the automatic value, the colour from the number of the shell and the opacity growing outwards. An empty table means the cut-off and a single shell.

The colour is chosen by a double click, a row is deleted by the right button, and a new one appears by itself once the last is filled in.

The inner shells are drawn before the outer ones. What is drawn first hides whatever lies behind it.

The cap at the edge of the cube

Cap where the body meets the edge of the cube puts a cap where the body runs into the boundary of the cube or into an area switched off by the clipping. Without it the shell is not closed, the body looks opened up, and no volume can be computed from it. Off by default.

Cleanup

Smoothing settles the steps left by the cells of the cube, and **discard pieces smaller than** removes the small scraps. The scraps go before the smoothing, otherwise a scrap draws its neighbours towards itself and a dent is left where it was.

Smoothing shrinks the body a little, so take the unsmoothed one to count a volume.

Shells into a project layer

The button on the toolbar puts the shells of the selected layer into a project layer, taking the levels and the cleanup from the properties of the layer. The shell is split into connected bodies at once, and the volume of each is computed there and then.

The button takes two modes: an isosurface over a cube and a bed body. For a bed body every pair of bands gives its own body, and the level field holds the number of the roof band. The body is built afresh rather than taken from the scene. What is shown has its vertical stretched by the exaggeration, the layers spaced down Z and the mesh thinned to a vertex budget, so the volume of such a body is wrong by exactly the factor the vertical is stretched by. The clipping, however, is the one set in the scene.

A body per feature. The fields are the number, the level, the colour, the count of faces, the closedness, the volume, the range of elevations, and two more about the quality. **holes** counts torn edges, and with them the volume is meaningless and is not computed. **pinch** counts the places where the body touches itself, and there the volume is exact.

The volume is computed by an exact formula over a closed shell, not by summing cells: the shell cuts the edge cells in half, and a sum over whole cells overstates the volume by some five per cent. For the same reason the volume of a bed body here and the volume from 1.02 will not agree exactly. The mesh runs through the centres of the cells, 1.02 counts whole cells, and the difference is the half-cells around the perimeter.

The layer is temporary. To keep it beyond the session, save it to a file.

Exporting the scene

A button on the toolbar writes the scene to a file, the format chosen by the extension.

GLB carries colour and transparency and suits viewing in a browser, in Blender and in Windows. The vertical exaggeration is asked for: the true heights are the ones to compute with, the model as it looks on the screen is the one to show. The coordinate box with its labels goes out together with the scene.

STL and **OBJ** are for CAD, where a closed shell is needed to make a solid from. The box and the labels do not go there, and the elevations are the true ones. At the end the count of bodies that came out closed is reported: a shell that is not closed will show in CAD as a mesh, but no solid can be made from it.

Spin and capture

The spin button walks the camera around the model, and pressing it again stops it. The scene is not rebuilt while it turns.

The capture button writes a full turn as PNG frames into a chosen folder, thirty six frames of ten degrees each. Assembling them into a video is not something the module takes on, there are ready means for that.

The tools

All of them live in the Processing panel, in the groups **1. Bed and block model** and **2. 3D interpolation**.

1.01 Assemble a bed grid

Field	What it sets
Roof (raster)	The grid of the bed roof. Elevations in metres, and the step and extent must match the floor, or there is nothing to compute the thickness from.
Floor (raster, optional)	The grid of the floor. Where the floor is above the roof the thickness comes out negative and the cell becomes a gap.

Field	What it sets
Floor at the elevation, m	Build the surface downwards to a constant elevation. That turns a DEM into a body that can be subtracted and intersected: mined-out parts of a pit, spoil heaps. Where the surface is NOT ABOVE the elevation there is no body, and such cells are cleared - otherwise the volume would be counted with a negative thickness.
Or a thickness downwards, m	Build downwards by a constant thickness from the surface itself. It suits a layer of even thickness: overburden, topsoil, fill.
Parameters (rasters, band 1 is taken)	Extra grids that will become separate bands: grade, density, domain. The first band of each is taken.
Bed grid	A multiband grid: band 1 the roof, band 2 the floor, then the parameters. Every other tool and the viewer read this order.
Roof band	The roof band in the source grid. Needed when the roof is not the first band but sits inside a multiband one.
Bottom band	The floor band in the source grid. Needed when the floor sits inside a multiband one.

When there is no floor

A survey gives a surface, and only bodies can be subtracted and intersected. That is why the floor need not be given as a raster: it is laid at a constant elevation or built downwards by a constant thickness.

An elevation suits counting above a level: the wall of a pit, a spoil heap, the volume above a horizon. A constant thickness suits a layer of even thickness: overburden, topsoil, fill.

Where the surface is **not above** the given elevation there is no body at all. Such cells are cleared and their count is printed to the log. Counting them with a negative thickness is not allowed, the volume would shrink from it, while in truth there is simply nothing there.

What to do with it next

The assembled grid is taken by 1.02 for the volume and the reserves, by the scene in the **Bed body** mode, and by 2.13 for a closed shell for the boolean operations and the selection.

1.02 Bed calculator

Field	What it sets
Bed grid (band 1 roof, band 2 bottom)	The bed grid from 1.01. The first band is the roof, the second the floor, and the thickness follows from them.
Roof band	The roof band. The next one counts as the floor: that is how a bed grid is built. In a multi-bed grid this is the choice of bed - the first one used to be taken in silence.
Content band (empty - no content)	The grade band. Empty means computing only volume and thickness, without reserves.

Field	What it sets
Ore density, t/m³	The ore density. The volume is multiplied by it to get mass: without it the report is in cubic metres rather than tonnes.
Reserve contour (polygons, optional)	Polygons outside which cells do not enter the computation: a computation block, a licence area.
Bed grid with thickness and reserves	The same bed grid with added bands of thickness and reserves per cell.
Report (HTML)	A summary over the contour: area, volume, mass, mean grade. Opens in a browser.

1.03 Bed grid to a block model

Field	What it sets
Bed grid (band 1 roof, band 2 bottom)	The bed grid from 1.01. The column between the roof and the floor is split into blocks.
Ore density, t/m³	The ore density, when there is no band for it. The block volume is multiplied by it.
Reserve contour (polygons, optional)	Polygons outside which blocks are not written out: a computation block, a licence area.
Block model (centroids)	A centroid point per block with its size, volume and mass. The usual QGIS vector machinery works from there.
Density band (empty - use the value above)	The density band in the grid. Empty means taking the single value set above for the whole bed.
Vertical layers (column split)	How many blocks to split the column into vertically. One block gives a model without vertical division, and the whole thickness then falls into a single layer.

A multi-bed grid

A grid from 2.08 carries as many pairs of bands as there are beds. Tools 1.02 and 1.03 count ONE bed, the one whose roof is chosen by the band. The first one used to be taken in silence: the numbers came out right, but about a different bed from the one the person had in mind.

The grade band cannot be a roof or a floor. In a multi-bed grid the third band is the roof of the next bed, and the grade would come out equal to an elevation: minus two hundred and forty seven “per cent” and negative reserves of metal. Such a band is now refused with an explanation.

1.04 Surfaces to 3D (meshes)

Field	What it sets
Surface grids	The grids to be given out as meshes. Each becomes a separate 2DM file.
Z scale (vertical exaggeration)	Vertical exaggeration. A bed of one metre over a kilometre cannot be seen without it, but volume can no longer be computed on such a mesh.

Field	What it sets
Z offset	A shift of all elevations along the vertical. Needed to separate the beds of a pile and see them apart.
Z spacing (step per next grid)	The spread of surfaces along the vertical. Otherwise the beds of a pile lie flush and fight for depth.
Folder for meshes (2DM)	Where to put the files. The file name is taken from the grid name.
Node thinning (every Nth)	Thinning of the nodes. Every second node is four times fewer triangles, and the shape of the bed looks the same.
Elevation band (Z)	The elevation band in the grid. For a bed grid it is the roof or the floor, depending on what to show.

1.05 Domains to a bed band

Field	What it sets
Bed grid	The bed grid a domain band will be added to.
Domain polygons	Domain polygons: ore types, sites, zones. A cell takes the code of the domain it falls into.
Bed grid with a domain band	The same grid with an added domain band. Filters for the computation and colouring of the scene work from it.
Domain code field (numeric, optional)	The numeric field with the domain code. Empty means numbering the polygons in order.

1.06 Reserve difference (write-off)

Field	What it sets
The “before” model (centroids)	The block model at the start of the period. Comparison goes by matching blocks, so both models must be built on one grid.
The “after” model (centroids)	The block model at the end of the period. A block absent from it counts as mined out entirely.
Difference (centroids)	Centroids with the difference per block. The sum of the field over the layer is the write-off for the period.
Reserve field	The reserve field whose difference is computed: mass, volume, metal.

1.07 Create sample data (demo)

Field	What it sets
Example	What exactly to create: a bed body, a pile of folded beds, a cube or a tetrahedron. The map for a texture has moved to the separate tool 1.08.
Extent (map view) - placement and size	Where to put the example and of what size. Empty means taking the extent of the view window.
Thickness, map units	The bed thickness in map units. Whether the body is visible at the usual vertical scale depends on it.
Bed body resolution (cells per side)	How many cells the side of the body is split into. Finer means a smoother shape and more triangles.
Output as TIN (triangulate)	Triangulate the body. Without it the faces come out quadrilateral, and not every viewer shows those.
Body (demo)	A layer with bodies: polygons with Z, fit for the scene and for volume computation.
Base elevation (floor), map units	The elevation of the floor. The pile is built upwards from it.
Beds in the suite	How many beds in the pile. Each lies as its own body with its own grade.

1.08 A map for a texture (demo)

Field	What it sets
Extent (when no grid is set)	The bounds of the map when no grid is set. Empty means taking the extent of the view window.
Map: by the extent of a grid (raster)	The raster whose extent the map is made to. Needed so the texture lands exactly on an existing grid.
Map (demo)	A picture for a texture: it can be stretched over a surface in the viewer.
Map: image side, pixels	The side of the picture in pixels. Larger means a sharper texture and a heavier file.
Map: graticule cells	How many cells of the coordinate grid to draw on the map.
Map: bed fields	How many bed fields to draw on the map.

2.01 Demonstration boreholes in three dimensions

Field	What it sets
Deposit type	A folded and dipping bed shows the main point: cube levels cut the deposit across. A lens is isotropic and the simplest, a vein is the opposite extreme.
Holes	The grid is jittered rather than regular: a regular one gives interpolation too easy a task.

Field	What it sets
Sample length, m	A sample longer than the thickness of the deposit will miss it between the readings. On a bed of twenty six metres, ten metres is already a lot.
Site extent	The site the boreholes are placed on. Empty means a kilometre from the origin, which is written to the log.
Surface elevation, m	The mean elevation of the ground surface. Collars are placed on a gentle relief around it.
Drilling depth, m	The drilling depth down from the surface. The proportions of the body are taken from it as well.
Sampling noise, a share	The share of lognormal sampling noise. Zero gives data without noise, where the model itself is visible.
The random seed	The same seed gives the same data: with it methods can be compared on an unchanged sample set.
Samples with grades	Samples with the fields hole, from_m, to_m, grade, truth, zone.
Core grade above background	The core grade above background. The boundary of the body is where the grade falls to half of it.
Background in the host rock	The background in the host rock. The cutoff that separates the body is computed from it and from the core grade.
Overall grade trend, a share	The overall grade trend across the site. It is there so the data does not reduce to a single body.
Share of holes stopped short	The share of holes stopped short. It is there so the cube has places without data, as on real exploration.
Hole inclination, degrees	The tilt of the holes from the vertical. Zero gives vertical boreholes.

2.02 Interpolating points in 3D

Computes a value at the nodes of a volume grid from points with elevation.

Anisotropy is the ratio of the vertical scale to the horizontal one. Without it the nearest point turns out to be the neighbouring borehole rather than the neighbouring sample at the same place in plan.

Nodes with fewer points within the radius than needed stay a gap: emptiness beats an invented value.

Neighbours are gathered by sectors: the circle around the node is split into equal parts and each gives its share of the nearest points. Without this, under anisotropy all the neighbours land in one borehole, because a sample in the hole is hundreds of times closer than the neighbouring hole, and inverse distances degenerate into nearest neighbour. One sector switches the split off.

The elevation source is set separately. A flat layer gives a zero Z at every point, so taking it from the geometry will not do: all the samples would land in one plane. An elevation field suits a computed elevation, and depth below a surface suits samples that record a depth rather than an

elevation. A point whose elevation could not be obtained is left out, and the number of those is written to the log.

Field	What it sets
Points with an elevation	The layer of samples. The elevation comes from the geometry, from a field or from a surface, which is set below.
The value field	The numeric field whose value is spread through the cube: grade, concentration, moisture. On the demonstration data from 2.01 it is grade rather than hole: otherwise the cube comes out of borehole numbers.
Elevation source	A flat layer gives a zero Z at every point. Taking it from the geometry would put every sample in one plane and the cube would be meaningless.
Elevation or depth field	For elevation from a field this is the elevation itself, for depth it is the depth measured down from the surface.
Surface the depth is measured from	The grid the depth is measured from. Needed by soil and similar samples, which record a depth rather than an elevation.
The method	Nearest neighbour gives steps and suits checking the data. Inverse distances give a smoothed field.
Step in plan, m (0 means from the data)	Zero takes a fifth of the distance between places in plan. Finer is pointless: there is no data in between anyway, and the node count grows as a square.
Vertical step, m (0 means from the data)	Zero takes half the sampling step. Coarser means merging neighbouring samples and losing the difference with depth.
Largest number of points (0 means from the data)	Zero takes one more than the number of samples at one place in plan. More means mixing all the levels at once and smoothing an anomaly with depth away.
A cube of values	A multiband grid: a band is a horizontal level, and the elevation of the first level and the step go into the metadata.
Anisotropy (0 means from the data)	Zero measures the variogram on the data and takes the ratio of the vertical range to the plan one. This is the case where no guessing is needed. A value of your own sets the scale by hand: a large one smooths along the vertical, a small one keeps the difference with depth.
The search radius, m (0 is automatic)	Zero takes a quarter of the data extent. A node with too few points within the radius stays a gap.
The power of the inverse distances	The larger the power, the more a near point outweighs the far ones. Two is the usual choice.
The smallest number of points	A node with fewer points within the radius than this stays a gap: emptiness beats an invented value.

Field	What it sets
Search sectors (0 means from the data)	Zero takes it from the data: boreholes need the split, or all the neighbours land in one hole, while for samples in plan it only tears the field. A sector boundary runs as a ray from the node, and the set of neighbours changes across it in a jump: hence the stars on soil samples.

2.03 Cube to a block model

Field	What it sets
Cube of values (bands are levels)	A cube of values from 2.02: bands are levels, the elevation of the first level and the step live in the metadata.
Apply the cutoff	Without a cutoff every cell with data is written out, with one only those not below it.
Cutoff	The value below which a cell does not reach the model.
Computation contour	Polygons outside which cells are not written out: a computation block, a licence area.
Clip above (surface)	The surface above: the points below it stay. That is how everything above the ground surface or above the bed roof is cut off.
Clip below (surface)	The surface below: the points above it stay. Together with the upper one only the bed is left.
Block model	A centroid point per occupied cell with the block size, volume and value.
Colour intervals (0 means no classes)	How many intervals to split the value into. The interval number goes into the cls field and suits colouring.
Density, t/m3 (0 means no conversion)	With a density given, mass is added to every block in the dens and ore_t fields.

2.04 Cube body as voxels

Field	What it sets
Cube of values (bands are levels)	A cube of values from 2.02: bands are levels, the elevation of the first level and the step live in the metadata.
Cutoff	A cell not below the cutoff counts as a body. For the demonstration data the cutoff is printed by 2.01.
Computation contour	Polygons outside which cells do not reach the body: a computation block, a licence area.
Colour intervals (0 means a single body)	Zero builds one body. Several intervals give a feature each, and the body can be coloured by grade. Not read when own bounds are given.

Field	What it sets
Own interval bounds, space separated	Own interval bounds, space separated: 0 5 10 15. They set the split instead of equal shares. A comma inside a number is a decimal mark. Order and repeats do not matter. A cell above the last bound stays in the last interval, it must not be lost.
Interval names, comma separated	Interval names, comma separated: low, medium, high. They go into the name field. Missing ones stay empty, extra ones are dropped.
Merge neighbouring faces	Merging makes the layer many times lighter but tears the boundary of the body with T-junctions. Such a body can neither be measured by volume nor cut in the scene: the cut stays open, there is nothing to cap it with. For volumes and cuts clear the flag.
Remove edge pinches	A pinch is two cells touching along a single diagonal. It is not a hole, but its edge belongs to four faces and a watertightness check rejects such a body.
Body as voxels	MULTIPOLYGON Z, one feature per colour interval. Fields cls, vmin, vmax, faces and shell.

2.05 Check of the interpolation

Field	What it sets
Points with an elevation	The same layer of samples that goes into 2.02. The check runs on the samples themselves, no cube is needed for it.
The value field	The value field the cube is built from. That is what we check. On the demonstration data from 2.01 it is grade.
Elevation source	The elevation source must match the one set in 2.02, otherwise a different arrangement of points is checked.
Elevation or depth field	For elevation from a field this is the elevation itself, for depth it is the depth measured down from the surface.
Surface the depth is measured from	The grid the depth is measured from. The same one as in 2.02.
Borehole field (empty means one sample at a time)	The borehole number. With it the whole hole is removed from the set and the check measures the ability to hit between holes. Without it one sample is removed, the neighbours come from the same hole, and the error comes out several times smaller than the real one.
The method	The method you intend to use in 2.02. The check is there to choose between them by numbers rather than by eye.

Field	What it sets
Largest number of points (0 means from the data)	Zero takes one more than the number of samples at one place in plan. This is the same choice as in 2.02.
Check residuals	Samples with the fields value, model, resid and aresid. They show not only how large the miss is but where it happened.
The anisotropy (vertical to horizontal)	The ratio of the vertical scale to the horizontal one. It is picked exactly by the error of this check.
The search radius, m (0 is automatic)	Zero takes a quarter of the data extent. A sample with no points around it stays unchecked.
The power of the inverse distances	The power of inverse distances. It is picked by the error of this check together with the anisotropy.
The smallest number of points	A sample with fewer points around it than this stays unchecked and does not enter the error.
Search sectors (0 means from the data)	Zero takes it from the data: boreholes need the split, or all the neighbours land in one hole, while for single samples in plan it only tears the field.

2.06 Kriging in three dimensions

Field	What it sets
Points with an elevation	The same layer of samples that goes into 2.02. The elevation is set below in the same way.
The value field	The numeric field whose value is spread through the cube. On the demonstration data from 2.01 it is grade.
Elevation source	A flat layer gives a zero Z at every point. Taking it from the geometry would put every sample in one plane and the cube would be meaningless.
Elevation or depth field	For elevation from a field this is the elevation itself, for depth it is the depth measured down from the surface.
Surface the depth is measured from	The grid the depth is measured from. Needed by samples that record a depth rather than an elevation.
Reference surface (flattening)	The roof or the floor of the bed. With it the vertical is counted from the surface and the computation runs along the bedding. The variogram is measured in the flattened coordinates already: measuring it in absolute ones and computing in flattened ones would give a model that does not belong to this data.
Floor for the thickness fraction	The second surface. With it the elevation becomes a fraction of the thickness: zero at the roof, one at the floor.

Field	What it sets
Step in plan, m (0 means from the data)	Zero takes a fifth of the distance between places in plan. Finer is pointless: there is no data in between anyway, and the node count grows as a square.
Vertical step, m (0 means from the data)	Zero takes half the sampling step. Coarser means merging neighbouring samples and losing the difference with depth.
Neighbours per node	Neighbours per node. Kriging solves a system the size of their number, so the cost grows as a cube: sixteen is the usual choice, thirty two is already noticeably dearer.
Measure the variogram on the data	The variogram is measured on the data itself: the range from the plan measurement, the nugget from the vertical one. Setting three numbers by eye is pointless, measuring them was the whole idea.
A cube of values	A cube of values: a band is a horizontal level.
Cube of the estimation variance	A cube of the estimation variance, of the same size. Zero at a sample, growing away from the data. It is a map of trust, and it is the one thing kriging always gives, whatever the density of the grid.
Variogram model	The kind of model. The difference between them is small, what matters more is the behaviour near zero: the gaussian one gives too smooth a field where the data is noisy.
Nugget effect	The scatter that does not fall even between neighbouring samples: sampling error and variability finer than the grid. Read only when the automatic measurement is off.
Sill	The overall scatter of the data the variogram reaches at large distances.
Range, m	The distance beyond which samples know nothing about each other.
Anisotropy (0 means from the data)	Zero takes the ratio of the vertical range to the plan one, measured on the data. This is the case where no guessing is needed.
The search radius, m (0 is automatic)	Zero takes a quarter of the data extent. A node with too few points within the radius stays a gap.
Search sectors (0 means from the data)	Zero takes it from the data: boreholes need the split, or all the neighbours land in one hole, while for samples in plan it only tears the field. A sector boundary runs as a ray from the node, and the set of neighbours changes across it in a jump: hence the stars on soil samples.

2.07 MBA in volume

Builds a cube of values from scattered points with multilevel B-splines.

A coarse lattice approximates the data, the residual is approximated by a lattice twice as fine, and so on level by level. No system of equations is solved: the work is linear in the number of points, and on hundreds of thousands of measurements the method computes where kriging stalls.

The method approximates rather than estimates. It does not hit the measured values exactly and gives no estimation error. Next to kriging it is good as a trend, which is then refined by kriging the residuals.

Beyond the cloud of points the surface goes anywhere: the edge coefficients have no data. Clip the result by a contour or by surfaces.

Field	What it sets
Points with an elevation	Measured points: borehole samples, intervals, anything with an elevation and a value.
The value field	The field with the value spread through the volume.
Elevation source	Where the elevation of a sample comes from: the geometry height, a field, or a depth below a surface.
Elevation or depth field	The field of elevation or depth, when it is not in the geometry.
Surface the depth is measured from	The surface the depth is measured from.
Initial lattice in plan	The initial lattice in plan: the method starts from it and doubles it at every level. A finer start gives a better first approximation but takes more memory.
Initial lattice down the vertical	The initial lattice down the vertical. Exploration data is elongated and the lattice need not be cubic: kilometres in plan and metres in thickness are different things. A smaller number here both stretches the influence along the bed and saves memory.
Levels	How many times to double the lattice. Every level picks up what the previous one could not: the residual falls fast, while the memory of the last level grows as a cube.
Cube step in plan (0 means from the data)	The cube step in plan. Zero takes it from the sampling net.
Cube step down the vertical (0 means from the data)	The cube step down the vertical. Zero takes it from the net.
A cube of values	A multiband grid: a band is a horizontal level of the cube. It is read by 2.03, 2.04 and the scene.
Stop by residual (0 means all levels)	Stopping by residual: once the largest deviation from the measured values falls below it, no further levels are built. Zero means build them all.
Smallest value (empty means no bound)	The smallest possible value: a grade is never below zero, while the method does leave the range. What leaves is pressed to the bound, and a plateau appears where the overshoot was - the shape is lost there. Leave it empty if there is no bound.
Largest value (empty means no bound)	The largest possible value. The number of clamped nodes goes to the log: it shows whether the model is any good.

2.08 Beds from sections

Builds a grid of beds from the outlines drawn on sections.

Every ring runs along the roof one way and along the floor back, so two surfaces are taken from it. They are interpolated over the area with multilevel B-splines, and the space between the sections is filled.

The position of the sections is taken from the geometry itself, from the vertices of the outlines. The lines of the sections need not be given separately. A flat drawn section is no good: it has no real elevations.

The output is a multiband grid: a roof and a floor per bed. The scene shows such a grid as a bed body, and 1.02 and 1.03 compute the thickness, the blocks and the volumes from it.

The floor of the upper bed and the roof of the lower one are one and the same boundary if the geologist drew them as one line. Such a boundary is built once from both sets of points: built separately, they drift apart between the sections, and the model gets a gap or an overlap that the section does not have. The threshold of the gluing is set by the tolerance.

The area mask clips the result: between the sections there is no data, and the mask says how far to trust the surfaces.

If the boundary of the bed on the plan is known together with its elevations, it is given as the outline on the plan. The roof and the floor meet on it, and the bed pinches out towards it. The mask cannot do that: at its edge the body drops down as a vertical wall.

Between the sections the surface goes where the interpolation put it: there is no data there. Where the sections cross, the elevations on them must agree. The disagreements are counted and go to the log together with the coordinates of the place where they are largest: with a number alone there is nowhere to look for the vertex that slipped.

Field	What it sets
Outlines on sections (polygons with Z)	A layer of outlines on sections: polygons with real Z. The position of the sections is taken from the geometry itself, from the vertices of the outline, and is never asked for. A flat drawn section is no good: its X and Y are coordinates on the sheet and there are no elevations at all. The geometry type must be PolygonZ or MultiPolygonZ.
The field of the bed number	The field of the bed number. Every bed gives two bands in the grid: a roof and a floor.
This bed only (empty means all)	The number of the bed, if only one is wanted. Empty means all.
Grid step, m (0 means from the data)	The step of the grid over the area. Zero takes a two-hundredth of the extent.
Area mask (polygons, optional)	A polygon layer the result is clipped by. Outside it there is no grid at all. Between the sections there are no data, and the surface goes where the interpolation drew it: the mask says how far to trust that. Usually it is the outline of a working, a pit or a block. Empty - the grid covers the whole extent of the contours.
Field of the bed number in the mask (empty - one mask for all)	The field that ties a mask polygon to a bed. It helps when one bed has shorter sections than its neighbours and has to be clipped differently. A polygon with an empty value clips every bed.

Field	What it sets
Outline of the bed on the plan (lines or polygons with Z, optional)	The boundary of the bed on the plan: lines or polygons with real Z. The roof and the floor meet on it, and the bed pinches out towards it. The mask cannot do that: it clips the finished surfaces, and at its edge the body drops down as a vertical wall. The bed does not go beyond the outline, no separate mask is needed for that. An outline without elevations works as a mask only. Where the outline crosses a section, the elevations must agree, the disagreement goes to the log.
Field of the bed number in the outline (empty - one outline for all)	The field that ties an outline to a bed. An outline with an empty value applies to all beds.
Grid of the beds	A multiband grid: a roof and a floor per bed, in order of the numbers. The scene shows such a grid as a bed body, 1.02 computes the thickness from it and 1.03 the blocks and the volumes.
Levels	Levels in the multilevel approximation. Few levels give a smooth surface, many bring it closer to the elevations on the sections.
Contact gluing tolerance, m	The tolerance within which the floor of the upper bed and the roof of the lower one count as one surface and are built once. Two surfaces built independently drift apart between the sections, and the model gets a gap or an overlap that the section does not have. Five centimetres by default is not modesty: the samples of two beds land in different places along the wall, and on a dipping bed that alone gives a centimetre and a half of disagreement. That is below any parting and above the sampling noise. Zero switches the gluing off: then every surface is its own.
Clip every bed by its own sections	Clip every bed by the area of its OWN sections: the convex hull of its samples plus the margin. Beyond it the bed was not observed, and drawing it there is invention. A bed met on three walls out of four will stop stretching over the whole area.
Margin outwards from the mask, m	A margin outwards from the mask. The bed usually continues beyond the outline of a working, and clipping exactly along it would cut away what the data do hold.

2.09 A demonstration drift (demo)

Creates a potash working with drawings of the walls, fan holes and grooves.

The beds and the grades are set by formulas, the sampling noise is added separately, so the set has a known answer. The true volumes of the beds are printed to the log: build the body with 2.08 and compare.

The working is a drift with crosscuts across it. Two walls of the drift give parallel sections, the crosscuts give intersections where the elevations must agree, and between them an area is left for the interpolation to fill.

There are three beds: KrII, AB and V. The set deliberately holds the cases the model stumbles on. The floor of KrII and the roof of AB are drawn as one line - the contact gluing must recognise them and build one surface. Between AB and V there is a parting, and no contact there. Inside AB a lens is drawn under the same bed number - it must not touch the boundary of the body. Bed V pinches out and does not reach the far crosscut at all, so without clipping by its own sections it will be stretched over the whole area.

There are two grades, KCl and the insoluble residue. They are tied in reverse: where there is more sylvite there is less residue. On a pair of bands it is visible how several parameters live in one grid at once.

Field	What it sets
Extent (where to put the drift)	Where to put the drift. Empty means the extent of the map view: the example lands where you are looking.
Length of the drift, m	The length of the drift. It decides how much area is left between the walls and the crosscuts - the very place where there is no data and the interpolation works.
Width of the working, m	The width of the working: the distance between the walls. Two walls give parallel sections a couple of metres apart.
Crosscuts across	Crosscuts across the drift. Each gives another pair of walls, and where they cross the walls of the drift the elevations must agree. Without crosscuts there is nothing to check that on.
Sample length, m	The length of a sample in a hole and the step of a groove across the thickness. A sample longer than the bed will miss it between the measurements.
Sampling noise, a share	The share of lognormal sampling noise. Zero gives data without noise: on them the model itself is visible rather than the scatter.
Wall drawings (polygons with Z)	Wall drawings: polygons with real Z, the input for 2.08.
Samples of the holes	Samples of the fan holes: points with grades, the input for 2.06 and 2.07.
Grooves across the thickness	Grooves across the thickness of a bed on the walls: intervals with grades.
The random seed	The same seed gives the same set. With it the methods are compared on unchanged data.
Draw a lens inside bed AB	Draw a lens inside bed AB under the same bed number. It must not touch the boundary of the body, and that is exactly what is being checked.

What it is for

There is usually nothing to check a model against: nobody has seen the real bedding, and to the eye an honest model and an invention look equally convincing. This set **has a known answer**. The beds and the grades are set by formulas, the sampling noise is added separately, the true

volumes go to the log, and beside every measurement lies the value without noise. So what you build shows not “close enough” but the size of the miss.

What is inside

A drift with two crosscuts. The two walls give parallel sections, the crosscuts give intersections where the elevations must agree, and between them an area is left for the interpolation to fill.

Three beds, with the awkward cases planted on purpose:

- the floor of KrII and the roof of AB are drawn as **one line**, and the gluing of contacts must recognise them and build one surface,
- between AB and V there is a parting of about a metre and no contact at all, so there must be no gluing,
- inside AB a **lens** is drawn under the same bed number, and it must not touch the boundary of the body,
- bed V **pinches out** and does not reach the far crosscut.

The numbers to expect

With the default settings the true volumes are 5662, 7764 and 1390 cubic metres. A body built from the drawings by 2.08, clipped by its own sections, agrees with them to tenths of a per cent. The contact of KrII and AB differs by 0.016 m and is glued, the contact of AB and V by 0.83 m and is not.

2.10 Surfaces from cross sections

Builds surfaces from the lines drawn on sections.

A line on a section is one surface, not a body: it need not be split into a roof and a floor, every vertex is already a point in space. Rings and bodies are the business of 2.08.

This is how a designed surface is put together from cross profiles, a fault plane from its traces on a series of sections, a horizon from the boundary of a member. The difference of two bands gives the volume of the works: where too much was filled and where too little.

The position of the sections is taken from the geometry itself, from the vertices of the lines. They need not be given separately. A flat drawn section is no good: it has no real marks.

Before the fitting a flat trend is removed: without it the error of the method grows with the mark itself rather than with its spread.

Where the sections cross, the marks on them must agree. The disagreement is counted and goes to the log together with the coordinates of the place where it is largest.

Field	What it sets
Lines on sections (with Z)	A layer of lines on sections: lines with real Z. A cross profile of an embankment, the trace of a fault, the boundary of a member - anything drawn on a section as one line rather than a ring. The position of the sections is taken from the geometry itself, from the vertices, and is asked for nowhere. A flat drawn section is no good: its X and Y are coordinates on the sheet, and there are no marks at all.

Field	What it sets
Field of the surface name	Field of the surface name. Every surface gives its own band in the grid: that way a roof and a floor, a design and a survey, or several horizons are built in one run. Empty - all the lines count as one surface.
This surface only (empty - all)	The name of the surface if only one is wanted. Empty - all of them.
Points with elevations (optional)	Separate measurements in addition to the sections: hole elevations, points where a fault was met, benchmarks. They are taken where there are no sections, and there a point holds the surface alone - it has nobody to argue with. Next to a section an argument is possible, and its size goes to the log.
Field of the elevation on the points	Field of the elevation on the points. It is needed when the points have no height in the geometry itself: a flat layer has nowhere to take it from, and putting a zero in silence is not allowed.
Field of the surface name on the points	Field of the surface name on the points. Empty - all the points go into the same surface as the lines without a name.
Grid step, m (0 means from the data)	The step of the grid over the area. Zero takes a two-hundredth of the extent.
Area mask (polygons, optional)	A polygon layer the result is clipped by. Between the sections there is no data, and the surface goes where the interpolation drew it: the mask says how far to trust that.
Field of the surface name in the mask	The field that ties a mask polygon to a surface.
(empty - one mask for all)	A polygon with an empty value clips them all.
Surfaces (grid)	A multiband grid: a band per surface, in the order of the names. The difference of two bands is the volume of the works: where too much was filled and where too little.
Levels	Levels in the multilevel approximation. Few levels give a smooth surface, many bring it closer to the marks on the sections.
Clip every surface by its own sections	Clip every surface by the area of its OWN sections: the convex hull of its points plus the margin. The designed embankment is not defined beyond the last profile, and continuing it there is invention.
Margin outwards from the mask, m	A margin outwards from the mask. A surface usually continues beyond the outermost section, and clipping exactly along it would cut away what the data do hold.

What it is for

A ring on a section is a body with a roof and a floor, and 2.08 deals with it. A **line** is one surface, and it needs no splitting: every vertex is already a point in space.

Hence the uses. Cross profiles of an embankment every hundred metres give the designed surface, and its difference with the survey is the volume of the works. The trace of a fault on a series

of sections gives its plane, and the boundary of a member gives a horizon.

How it counts

The vertices of the lines go into the interpolation as they are, with a flat trend removed. The field of the surface name sorts the lines into bands: the design and the survey, a roof and a floor, several horizons, all in one run. The difference of two bands is the volume of the works.

Where the sections cross, the elevations on them must agree. The disagreement is counted and printed with the coordinates of the place where it is largest, because a single number gives nowhere to go and look.

Separate measurements

Besides the lines of sections the tool takes **points with elevations**: holes, points where a fault was met, benchmarks. They are used where there are no sections.

The elevation is read from the geometry when it is three-dimensional. A flat layer has nowhere to take it from, so a field of the elevation is given. The tool will not put a zero in silence.

There is no weighting, and none is needed. Where there are no sections a point holds the surface alone: on a test embankment with one profile missing a measurement raised by a metre and a half raised the surface, and the departure came to 0.06 m. Next to a section it is different: a hundred vertices of the profile outweigh one point, and the departure there was 1.09 m. That is right - a hundred measurements are more reliable than one - but it has to be known.

That is why the departure from the measurements is always printed, the mean and the largest, with coordinates. A measurement ignored in silence looks like a fault of the tool, while a number shows at once who argues with whom.

What accuracy to expect

The test embankment: a crest of six metres, slopes of one in two, profiles every hundred metres. The restored surface departed from the designed one by 0.18 m at the worst place and by 0.03 m on average. The volume of the fill agreed with the true one to three per cent.

The accuracy is decided by how often the sections are spaced, and no computation mends that. Beyond the outermost section the surface is not defined, and continuing it there is invention: that is what the clipping by own sections is for.

2.11 Boolean operations on shells

Subtracts, unites and intersects two bodies.

This is how mining is accounted for: the shell of a mined-out chamber is subtracted from the shell of an ore body and the remaining reserves come out. An intersection gives what fell inside a zone, a union takes two bodies as one.

The work goes not over meshes but over cells: both shells are turned into the occupancy of a common cube, the operation is done over the occupancy, and the result is turned back into a body. An exact operation over meshes cuts triangles against each other and on touches regularly gives a result that is not closed, and no volume can be taken from it.

The price of this decision: the body comes out stepped, and the accuracy is limited by the cell. The error follows the surface area and falls with the cell: on a ten-metre cube a cell of 0.5 m gave a volume error of five per cent, 0.25 m two and a half, 0.1 m landed exactly. The memory meanwhile grows as a cube.

The shells on input must be closed: one that is not has no inside, and there is nothing to tell what is within. A shell that is not closed on input is refused with the object named.

Field	What it sets
The first body (polygons with Z)	The first body: a closed shell from a layer of polygons with Z. The shells button in the viewer and the tool 2.04 give such layers.
The second body (polygons with Z)	The second body. For a difference this is what is cut out of the first: a mined-out chamber, a flooded zone.
Action	A difference leaves of the first body what the second does not hold. A union takes both, an intersection only the common part.
Side of the cell, m	The side of the cell. It sets the accuracy: the error follows the surface area and halves with every halving of the cell. On a ten-metre cube a cell of 0.5 m gave a volume error of 5 per cent, 0.25 m two and a half, 0.1 m landed exactly. The memory grows as a cube, so do not take it finer than needed.
The result (bodies)	The bodies of the result: polygons with Z, the volume of each in the attributes. The shell is always closed, so the volume can be computed from it.

What it is for

This is how mining is accounted for: the shell of a mined-out chamber is subtracted from the shell of an ore body and the remaining reserves come out. An intersection gives what fell inside a zone, and a union takes two bodies as one.

Why over cells

An exact operation over meshes cuts triangles against each other. On touches - and in geology bodies touch all the time - it regularly gives degenerate faces and a result that is not closed, and no volume can be taken from that. Here both shells are turned into the occupancy of a common cube, the operation is done over the occupancy, and the result becomes a body again. It is stepped, but always closed.

The price

The accuracy is limited by the cell, and the error follows the **surface area** rather than the volume. On two ten-metre cubes with a shift (intersection 500, union 1500, difference 500 cubic metres) a cell of 0.5 m gave five per cent, 0.25 m two and a half, 0.1 m landed exactly. The memory meanwhile grows as a cube.

What is checked on input

The shells must be closed, and that is checked by counting edges rather than by the fill having worked. A hole in a **side** wall is invisible to an upward ray: a vertical face is never crossed by it, and the volume would be computed over a torn body in silence.

2.12 Selection by a shell

Selects the features that fell into a closed shell and computes a summary over them.

That is how reserves are obtained: a block model, the shell of an ore body or of a mined-out zone, and out come the volume, the tonnage and the volume-weighted grade of the selection. Selecting the outside gives the remainder without building a difference of bodies.

Not only points can be selected. Lines and polygons with height go the same way: holes that pierced the body, traces of workings, shells of other bodies.

A point counts as inside by the parity rule: a ray is sent up from it, and if the shell is crossed an odd number of times the point is inside. No cells are needed for that, and the answer comes out exact rather than to within a cell, as with the boolean operations.

For lines and polygons the vertices are looked at. A segment that goes right through the body between two of its own vertices is not caught by vertices - the sampling step along is there for that.

The volume of the selected blocks is checked against the volume of the shell itself: the difference shows how coarse the model is for this body.

The shell must be closed: one that is not has no inside.

Field	What it sets
What to select (features with Z)	What to select: points, lines or polygons with height. A block model from 2.03 and 1.03, samples, traces of workings, shells of other bodies - as long as the geometry has Z.
The shell (polygons with Z)	The closed shell the selection is made by: an ore body, a mined-out zone, a block of reserves. The shells button in the viewer, 2.04 and 2.11 give such shells.
What to keep	Entirely inside - every vertex of the feature is inside the shell. Crosses - at least one is: that is how holes and workings going right through the body are caught. Outside - the remainder, everything that does not cross.
Field of the block volume	The field of the block volume, if a block model is being selected. With it the volume of the selection is computed and checked against the volume of the shell itself: the difference shows how coarse the model is for this body.
Field of the grade	The field of the grade. The mean over the selection is weighted by volume rather than plain: blocks come in different sizes, and a plain mean gives the small ones too much say.
Selected features	The selected features with all their own fields. The summary goes to the log.
Sampling step along lines, m (0 - vertices)	The sampling step along lines and around the edge of polygons. Without it only the vertices are looked at, and a segment that goes right through the body between two of its own vertices will not be found. Zero - vertices only.
Density, t/m3 (0 - no tonnage)	The density of the ore. With it the volume turns into tonnes and the grade into the tonnage of metal.

What it is for

This is the last link of the chain: a cube of values - a block model - a selection by a shell - the volume, the tonnage and the volume-weighted grade. Selecting the outside gives the remainder after mining without building a difference of bodies.

Not only points can be selected: lines and polygons with height go the same way.

How it counts

A point is inside if a ray sent up from it crosses the shell an odd number of times. No cells are needed for that, and the answer is **exact**, unlike the boolean operations, where the accuracy runs into the cell.

The mean grade is weighted by the volume of the block. A plain mean overstates the small blocks, and in a model those usually lie along the edge.

The check worth attention

The volume of the selected blocks is compared with the volume of the shell itself, and the difference is printed. That is not a check of correctness but a measure of how **coarse the block model** is for this body: a block is either wholly inside or wholly outside, there is no third case.

The limit of the method

For lines and polygons the vertices are looked at. A segment that goes right through the body between two of its own vertices is not caught by vertices - the sampling step along is there for that.

2.13 A shell from a bed grid

Builds a closed shell of a bed body from a grid: the roof, the floor and the skirt between them.

The shells button in the viewer does the same, but there it is a manual step. As a tool it becomes part of a processing model, and the chain “surface - body - subtract what is mined out - count the reserves” runs as a whole.

The shell is closed, so 2.11 and 2.12 take it, and the volume is computed by an exact formula over the shell itself.

The volume here and the volume from 1.02 differ by about one per cent: the shell runs through the centres of the cells, while 1.02 counts whole cells. These are different boundaries of one area, not a dispute of methods.

Field	What it sets
Bed grid	A bed grid: a roof band and a floor band. 1.01 gives one from two surfaces or from a single one built downwards, and 2.08 from the drawings on sections.
Roof band	The roof band. The next one counts as the floor: that is how a bed grid is built. In a multi-bed grid this is the choice of bed.
Bodies of the bed	The bodies of the bed: polygons with Z, the volume of each in the attributes. 2.11 and 2.12 take them next.

Field	What it sets
Thinning of the mesh	Thinning of the mesh. One means as it is, two takes every second cell and gives four times fewer triangles. The volume changes little, but fine detail is cut away.

What comes out

A closed shell of the body: the roof, the floor and the skirt between them. The bodies are split into connected parts, each with its own volume in the attributes. Closedness is checked there and then, bodies that are not closed are counted separately and named in the log, because 2.11 and 2.12 will not take them.

About the volume

On a test DEM of 101 by 101 with a cut-off elevation the shell came to 39524 cubic metres against 40309 over the cells. The difference came to two per cent, and all of it is around the edge.

2.14 Zonal statistics of shells

Counts for every body how many features fell into it and writes the numbers into the attributes of the body.

Points: the number inside and the sum of an attribute. Lines: the number of lines that touched the body and the length of their parts inside. Polygons: the number of polygons with at least one vertex inside.

A line is cut by the body: a segment is intersected with the surface, the points are sorted along it, and only the part inside goes into the length. A segment that goes right through the body between its own vertices is not lost.

Zero and empty are different things. A body with nothing inside gets zero: that is a measured fact. A body that is not closed gets empty and goes into a warning: it has no inside.

Repeated runs over one layer of bodies add up: the result of the previous run is given as the layer of bodies to the next, and every run adds its own fields. A clash of names is refused unless overwriting is allowed.

The sum over the bodies agrees: the length of lines inside all the bodies plus the length outside equals the total length, and that is printed to the log.

Field	What it sets
Bodies (polygons with Z)	A layer of bodies: closed shells, one per feature. Every body gets its own count fields. A body that is not closed gets empty rather than zero: it has no inside, and there is nothing to count.
What to count (features with Z)	What to count: points, lines or polygons with height. A block model, holes, traces of workings, faults.

Field	What it sets
Name of the result fields (empty - from the layer)	The name for the result fields. Empty - from the name of the layer, turned into Latin letters. Suffixes are added to it: _n the number of features inside, _len the length of lines inside, _sum the sum of the attribute. Give your own when you run the tool over one layer of bodies several times with different data.
Attribute to sum	A numeric attribute to sum: the volume of a block, a thickness, a grade. Empty - count only.
Allow overwriting of fields	Allow overwriting fields of the same name. Without it a repeated run with the same name is refused: otherwise the second result would silently wipe out the first.
Bodies with statistics	The bodies with all their previous fields and the new count fields. To accumulate several runs, give the result of the previous run as the layer of bodies to the next.

What it is for

This is 2.12 from the other side. There the features are selected by one body. Here every body gets written how many features fell into it, and the numbers go into the attributes.

How it counts

Points - by the parity of an upward ray, as in 2.12. Lines are cut by the body: a segment is intersected with the triangles of the shell, the points are sorted along it, and only the part inside goes into the length. A segment that goes right through the body between its own vertices is not lost here, unlike the vertex-based selection of 2.12.

Checked by two independent routes: the exact cut and a dense sampling of points along the same lines differed by 0.002 per cent.

Repeated runs

The tool is meant to accumulate: blocks of a model, then faults, then workings, all in one layer of bodies. The result of a run is given as the layer of bodies to the next, and every run adds its own fields. A clash of names is refused unless overwriting is allowed.

Zero and empty are different things. A body with nothing inside gets zero. A body that is not closed gets empty: it has no inside.

Typical tasks

Ten chains that come round again and again in the work. Numbers with a dot are Isoliner3D tools, names without a number are the neighbouring Isoliner.

2.15 Formation bodies from outlines

Builds bodies of formations from outlines in any planes: vertical sections, the map on the ground surface, inclined cuts. There may be several input layers.

For formations that lie side by side and meet along steep contacts. 2.08 builds beds lying one above another, and on such formations it stretches each over the common area.

Every formation gets a field: the signed distance to its contacts, plus inside, minus outside. The outer edge of an outline is not a contact: the bottom of a section and the edge of the map are drawn where the data ended. The field is interpolated in the volume with multilevel B-splines, and every voxel goes to the formation with the largest field. There are no gaps or overlaps between the bodies by construction.

If the sections are parallel, the interpolation lattice is dense in their plane and sparse across. Then a contact between the sections runs from one to the other instead of blurring. The map holds a contact near the surface, the sections carry it into depth.

From above the bodies are clipped by the relief built from the map elevations and the tops of the sections, from below by the lowest elevation of the data, in plan by the convex hull of the data.

The log shows the misfit at the contacts for every formation. A large misfit means that outlines in different planes disagree.

The code of a formation must match on the sections and on the map: that is how the tool knows it is one formation. A contact is found by the shared edges of neighbouring outlines in one plane, so neighbouring outlines should be drawn with snapped vertices, without gaps.

Field	What it sets
Outline layers (polygons with Z)	Outline layers: polygons with real Z, in any planes. Vertical sections, a map of the formations on the ground surface with the relief elevations, inclined cuts, all together. Every outline is sampled in its own plane, and the plane is taken from its vertices. A flat drawn section is no good: it has no real elevations.
Formation code field (empty - layer name)	The formation code field. The same code on a section and on the map means one formation. Empty or no such field in a layer - the layer name serves as the code: handy when every formation lies in its own layer.
Area grid step, m (0 - from the data)	The grid step over the area. Zero takes a hundredth of the extent. A contact lands within half a cell, and the time grows with the number of cells.
Vertical step, m (0 - half the area step)	The vertical grid step. Zero takes half the area step: formations are usually steeper vertically than along the strike.
Formation cube	Formation cube: every voxel holds the number of its formation, above the relief and outside the data it is empty. The scene shows it as a cube, 2.03 turns it into a block model.
Formation bodies	Formation bodies: closed shells, one per connected piece of a formation, with the volume in the attributes. 2.11, 2.12 and 2.14 take them.

2.16 A demonstration river network (demo)

Creates a relief and a river network over it, with a discharge on every link.

The relief is set by a formula: a general slope to the south, ridges and hills. Closed depressions are filled, water flows to the steepest of the eight neighbours, the catchment area accumulates

downstream. A channel starts where the catchment exceeds the threshold, the discharge of a link is proportional to its catchment area.

The set is meant to show lines as tubes. The discharge grows downstream tens of times, and the thickness of a tube reads at once. In the layer properties in the scene choose the tube display and the thickness field q, and colour by the Strahler order.

Field	What it sets
Extent (where to put the example)	Where to put the example. Empty means the extent of the map view: the network lands where you are looking.
Relief grid step, m	The relief grid step, metres. The grid is 140 by 110 cells, so the step also sets the size of the area: at 50 m it is 7 by 5.5 kilometres.
Catchment where a channel starts, km2	The catchment area where a channel starts. A smaller threshold gives a denser network and more small tributaries.
Runoff modulus, l/s per km2	The runoff modulus: litres per second from a square kilometre of catchment. The discharge of a link is its catchment area times this modulus.
River network	Links of the network: lines with relief elevations, with fields of discharge, catchment area, Strahler order and length. For showing as tubes the thickness field is q.
Relief	The relief the network runs over, with the depressions filled. In the scene it is the surface the channels lie on.
The random seed	The same seed gives the same relief and the same network. Another seed gives another relief.

From boreholes to reserves

1. Samples with grades in a point layer, the elevation in the geometry or in a field.
2. **2.05 Check of the interpolation** to choose the parameters. Leave out a whole hole, not a single sample: a neighbouring sample in the same hole almost repeats the one left out, and the error comes out lower than the real one.
3. **2.02 Interpolating points in 3D** or **2.07 MBA in volume** for the cube of values.
4. The cube into the scene, isosurface mode. Clipping by the terrain and by the licence outline is in the scene properties.
5. The **shells into a project layer** button: bodies with the volume of each.

From horizons to a bed

1. In Isoliner the **roof** and the **floor** are interpolated in plan.
2. **1.01 Assemble a bed grid** - a multiband grid.
3. **1.03 Bed grid to a block model** - blocks with volume and mass.
4. The same grid shows in the scene as a bed body, without any conversion.

A section on the drawing and in volume

The section fence built by Isoliner is polygons with real Z, and it goes into the scene as it is, and the colour of the beds comes from the `color` field. The point is a check by eye: the fence lies on the surfaces it was built from. ## From drawings on sections to a body

The tool **2.08 Beds from sections**, for when the holes are few and the sections are drawn.

Where the coordinates come from

From the geometry of the outlines, and from nowhere else. An outline on a three-dimensional section is a polygon whose every vertex has X, Y and Z. X and Y run along the line of the section, Z is the elevation. The tool works out the plane of the outline from those vertices, taking the direction from the largest spread of the points in plan. The lines of the sections need not be given separately: they are already in the data.

A flat drawn section is no good: its X and Y are coordinates on the sheet, not on the ground, and there are no elevations at all. The geometry type must be `PolygonZ` or `MultiPolygonZ`.

How the roof and the floor are taken

The ring of the outline is sampled across, along the line of the section: at every step the upper point of the ring and the lower one. Splitting the ring in half is not allowed - only a fence is built that way, with the top running forward and the bottom back, while a hand-drawn outline has its vertices in any order.

It is these two surfaces that are interpolated, not the body as a whole. A roof and a floor are ordinary surfaces, and the problem about them has a solution for any layout of sections.

Before the fitting a flat trend is removed from the elevations. Without that the error of the method grows with the elevation itself rather than with its spread. On a roof around minus two hundred and fifty metres with a spread of eighty centimetres the surface went off by fourteen metres, and no number of levels removed that defect.

The contact of neighbouring beds

The floor of the upper bed and the roof of the lower one are one and the same boundary if the geologist drew them as one line. Built separately, they drift apart between the sections, and the model gets a gap or an overlap that the section does not have.

The tool brings the samples of neighbouring beds together and measures the spread of the elevations. If it is within the **contact gluing tolerance**, the boundary is built once from both sets of points and put into both bands: the gap is then zero by construction. If it is larger, the surfaces are built separately and a warning goes to the log with the coordinates of the place where the disagreement is largest. Go there and look in your own layer of outlines: usually one vertex has slipped.

For this the beds are ordered by their bedding, from the top down, rather than by the name of the field: numbers come as text, and then “10” stands between “1” and “2”.

A lens inside a bed

The outlines of one section plane are taken together, by their outer boundary: the roof follows the topmost of them, the floor the lowest. A lens or a parting drawn inside a bed under the same number therefore does not touch the boundary of the body. Their samples used to be pooled together, and the cloud of the roof held both the real roof and the top of the lens: the interpolation got two answers in one place and ran the surface between them.

If the lens is wanted as a body of its own, give it its own bed number.

Outlines of different sections are never merged this way. A disagreement at one place in plan between two sections is data, and it must be seen rather than hidden by taking the outermost.

How far to trust the surface

The grid is built over the whole extent of the outlines, and beyond the sections the surface goes where the interpolation put it. The **area mask** says how far to trust that: outside it there is no grid at all. Usually it is the outline of a working, a pit or a block. The **margin outwards from the mask** is left because the bed continues beyond the outline of a working, and clipping exactly along it would cut away what the data do hold.

A mask for one bed

The mask need not be common to all. If a **field of the bed number** is set on the mask layer, a polygon clips the bed whose number it carries, and a polygon with an empty value clips them all. This is what to use when one bed has shorter sections than its neighbours.

The checkbox **clip every bed by its own sections** does the same without drawing: the convex hull of the bed's own samples plus the margin. In a pit with four walls a bed met on three of them used to take the whole area. With its own hull it takes 39 per cent against 64 for its neighbours.

A hull is convex, and no dent can come out of it: for a bed on three walls of a rectangular pit it covers the whole rectangle, because the middle lies inside. That is what the per-bed mask polygon is for.

The order of the work

1. Outlines of the beds on sections: polygons with real Z.
2. **2.08 Beds from sections**. Set the field of the bed number and leave "This bed only" empty: all the beds are built in one run, on a common grid. In separate runs every bed gets its own extent and its own step, and the contacts between them no longer meet.
3. The log: the thickness over the area must lie within the thickness on the sections, and the lines about the contacts must show centimetres rather than decimetres.
4. The grid into the scene, the **Bed body** mode. The shells button puts the bodies into a project layer, with the volume in the attributes.
5. Thickness, volume and reserves come from **1.02 Bed calculator**, the block model from **1.03**.

What is worth knowing beforehand. The accuracy is decided by how often the sections are spaced, and no computation can mend that: between two sections there is no data.

The volume of a body from the shells button and the volume from 1.02 will not agree exactly. The mesh runs through the centres of the cells, 1.02 counts whole cells, and the difference is the half-cells around the perimeter: about one per cent on a grid of two hundred cells, and noticeably more on a coarse one.

Formations side by side, not one above another

2.08 builds beds: each has a roof and a floor over the plan, and the beds lie one above another. If formations lie side by side and meet along steep contacts, like volcanic formations on a section across the strike, 2.08 stretches each over the common area. For them there is **2.15 Formation bodies from outlines**: sections, a map of the formations with the relief elevations and inclined cuts go in together, and every point of the volume goes to one formation.

A surface from cross sections

The designers send cross profiles every hundred metres. A designed surface has to be built and compared with the survey, to see where too much was filled and where too little.

1. Digitise the profiles as lines with real Z and put them on the alignment. Every line is one section.

2. **2.10 Surfaces from cross sections**. Set the field of the surface name if the design and the survey lie in one layer, and each will then go into its own band.
3. The difference of the bands is the volume of the works. It is computed by the raster calculator, or by **1.02** if the surfaces lie as the roof and the floor of one grid.

The accuracy is decided by how often the sections are spaced, and beyond the last profile the surface is not defined. The measurement on the test embankment and the clipping by a surface's own sections are described in the section of tool 2.10.

From a DEM to a body: a pit and spoil heaps

A survey gives a surface, and only bodies can be subtracted and intersected. Building downwards is what turns one into the other.

1. **1.01 Assemble a bed grid**: the roof is the DEM, the floor is set by an elevation, which is the level to count from, or by a thickness downwards for a layer of even thickness such as overburden or fill.
2. **2.13 A shell from a bed grid** gives a closed body in a project layer.
3. The same for the second surface, the mined-out part or the spoil heap.
4. **2.11 Boolean operations on shells**: subtract what is mined out, add the heap, intersect with a zone. Or **2.12 Selection by a shell** when the counting has to go over a block model.

Where the surface is not above the given elevation there is no body at all. Such cells are cleared and their count is printed to the log, because counting them with a negative thickness is not allowed, the volume would shrink from it.

One reservation about accuracy. The difference of two surfaces is computed over the cells of the grid and carries no error from voxels. A boolean operation over bodies goes over the cells of a cube, and a thin layer of half a metre will come out badly with a cell of a metre. For thin layers take the difference of the surfaces through 1.01 and 1.02, and for a complicated shape take the boolean operations.

Reserves left after mining

There is the shell of an ore body and the shell of what was mined out over a month. The remainder and its volume have to be obtained.

1. Both shells must be closed. The shells button in the viewer and tool 2.04 give closed ones, and the **closed** field in the attributes shows it.
2. **2.11 Boolean operations on shells**, the action "subtraction", the first body the ore, the second the mining.
3. The volume of every body of the result lies in the attributes. There can be several bodies, because the mining may have cut the deposit into parts.

The same result comes without subtracting bodies: with **2.12 Selection by a shell** select the blocks of the model that lie outside the shell of the mining. The selection is exact, without cells, and the volume and the grade are computed at once. There is a way to check yourself. The volume of the selected blocks is compared with the volume of the shell itself, and the difference is printed to the log, and that difference is the coarseness of the block model for this body.

The side of the cell sets the accuracy. The error follows the surface area and falls with the cell: on the test cubes of ten metres a cell of 0.5 m gave four per cent and 0.1 m gave one. The memory meanwhile grows as a cube, so there is no sense in taking it finer than needed.

The result is stepped, because these are cells and not a smooth shell. In exchange it is always closed and a volume can be computed from it, while an exact operation over meshes regularly gives broken geometry on touches, and no volume can be taken from that.

What has changed between surveys

Two models of the same area from different times, then **1.06 Reserve difference (write-off)**.

Showing it to someone who is not a geologist

GLB for a browser, Blender or Windows, PNG frames of a turn for a video, **STL** or **OBJ** for CAD. For CAD switch on the cap at the cube edge: an open shell will show as a mesh but will not become a solid.

Checking the data before the model

2.05 leaving out one sample, and the same leaving out a whole hole. The difference between the two errors tells what the model rests on.

Measured on the demonstration deposit, samples every five metres down the hole: with a 100 m grid of holes the ratio is 0.9, with 200 m it is 1.6, with 400 m it is 2.6. The error per sample hardly changes with the grid - a neighbour down the hole is always close by. The error per hole grows with the distance between the holes, and that one is the real one.

The figures are for the demonstration data, and yours will differ. What matters is the ratio and how it changes as the grid gets denser.

Typical situations

What is seen	Why	What to do
The scene is empty	No layer is ticked, or refresh was not pressed	Tick a layer and refresh
A layer does not show	The layer has no height: flat geometry and no source of elevation set	Set the source of height in the properties of the layer
Only part of the features shows	The vertex limit of the scene came into play	Raise the limit in the scene properties or untick the layers not needed
The body is opened up at a cut	The shell is not closed	Switch on the cap at the edge of the cube
The volume is not computed	The body has holes, the holes field is above zero	Switch on the cap, and reduce the count of levels if need be
The cut-off built nothing	The cut-off lies outside the range of the values of the cube	Take a value inside the range, which is shown in the legend
Stars and rays across the field of interpolation	The split into sectors on samples that lie in plan	Leave zero in the field of sectors, it is taken from the data
The shell came out coarse	The cube has few levels down the vertical	Set the vertical step in 2.02 by hand
The layer is refused at the input of a tool	A mesh of triangles is never a valid multipolygon	Nothing, the tools of the module do not require that check

Neighbouring plugins

Isoliner - interpolation in plan, contours, terrain, drawn sections, borehole graphics. Isoliner3D reads what it builds.

Topoliner - topographic constructions.

Licence and developer

Isoliner3D is developed by **Inform++ LLC**, Perm - www.informpp.ru.

The module is distributed under the **GNU General Public License, version 2 or, at your option, any later version** (GPL-2.0-or-later). You are free to use, study, modify and pass it on. Passing on a modified version, you must keep the same licence and open the source. The full text of the licence is in the `LICENSE` file inside the module.

The program is distributed in the hope that it will be useful, but **with no warranty whatsoever**, including the implied warranty of fitness for a particular purpose. Responsibility for the results rests with whoever applies them: every model is an approximation, and decisions about reserves and mining are made by a specialist, not by a program.

The bundled pyqtgraph and PyOpenGL libraries carry their own licences, compatible with the GPL.

Where to look

The **About the plugin** button on the toolbar, and the menu item of the same name, show the version and the links, open the changelog and the log of the module.

The log is started when the module loads, next to the QGIS profile, as `isoliner3d.log`. Every step goes into it, not only failures. The lines the tools print go there as well, not only into the Processing window: a picture of the log used to show that a run took half a second and not a single number to judge the result by.

Where	What for
www.informpp.ru	about the developer and the other products
Source code	builds, history, texts
Report a bug	if something works wrong

When reporting a bug, attach the version from the window title and the lines from the log: with them the cause is found in one go.