Dynamic Fracture Model Generator
Petrel version
Installation and Use guide

Michael Welch, Mikael Lüthje and Simon Oldfield

20/06/2023

Contents

1 Installation

The current version of the Dynamic Fracture Model Generator (DFM Generator) Petrel plug-in should be installed directly using the .pip installation files provided. You will need to download the correct .pip file for your version of Petrel.

After starting Petrel, open the Ocean Plugin Manager (**File → Options → Plugin manager**), as shown in Figure 1. Then click on **Install plug-in**, locate and open the *.pip* file. It is advisable to copy the *.pip* file locally onto the C drive before installing, as installing over a network can cause problems. Please make sure to select the *.pip* file corresponding to the Petrel version being used such as *DFMGenerator_2020.pip* for Petrel 2020. When the plugin has been installed correctly, you will get a confirmation screen as shown in Figure 2.

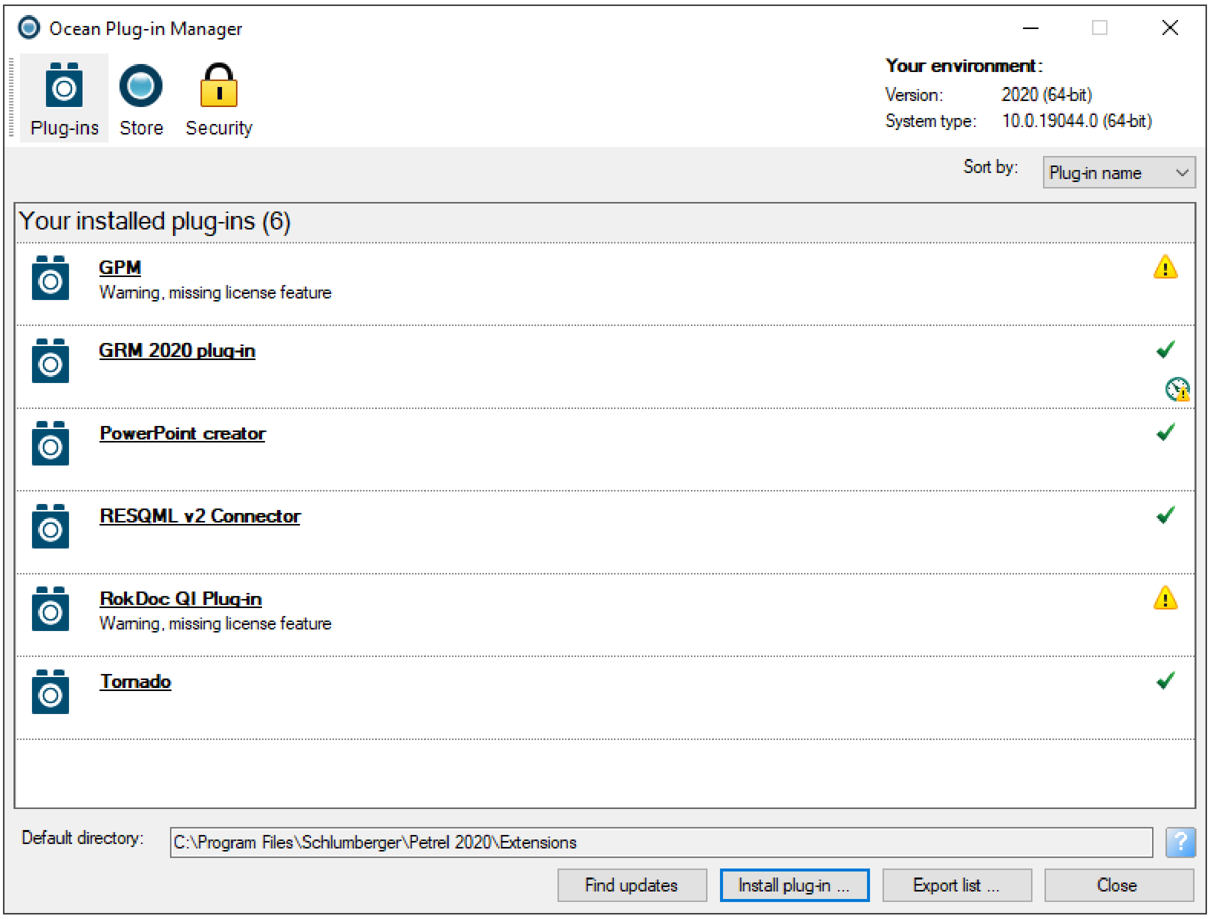

Figure 1: Ocean Plug-in Manager window.

You must then close the Plug-in Manager and restart Petrel before using the plugin. DFM Generator should now appear in the list of installed plug-ins in the Ocean Plugin Manager The **Build dynamic fracture model** launch button should now appear in the *Fracture Modelling* ribbon tab, as well as in the *Fracture Modelling* group in the *Process Pane*.

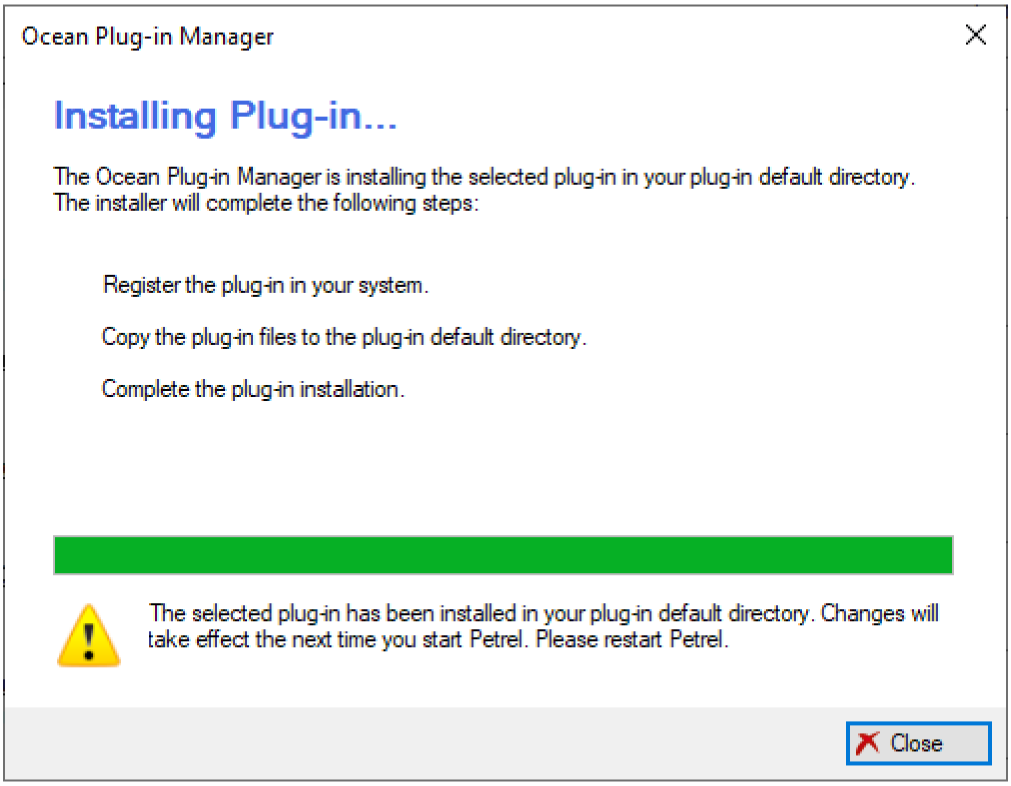

Figure 2: Confirmation message when the plugin has installed correctly.

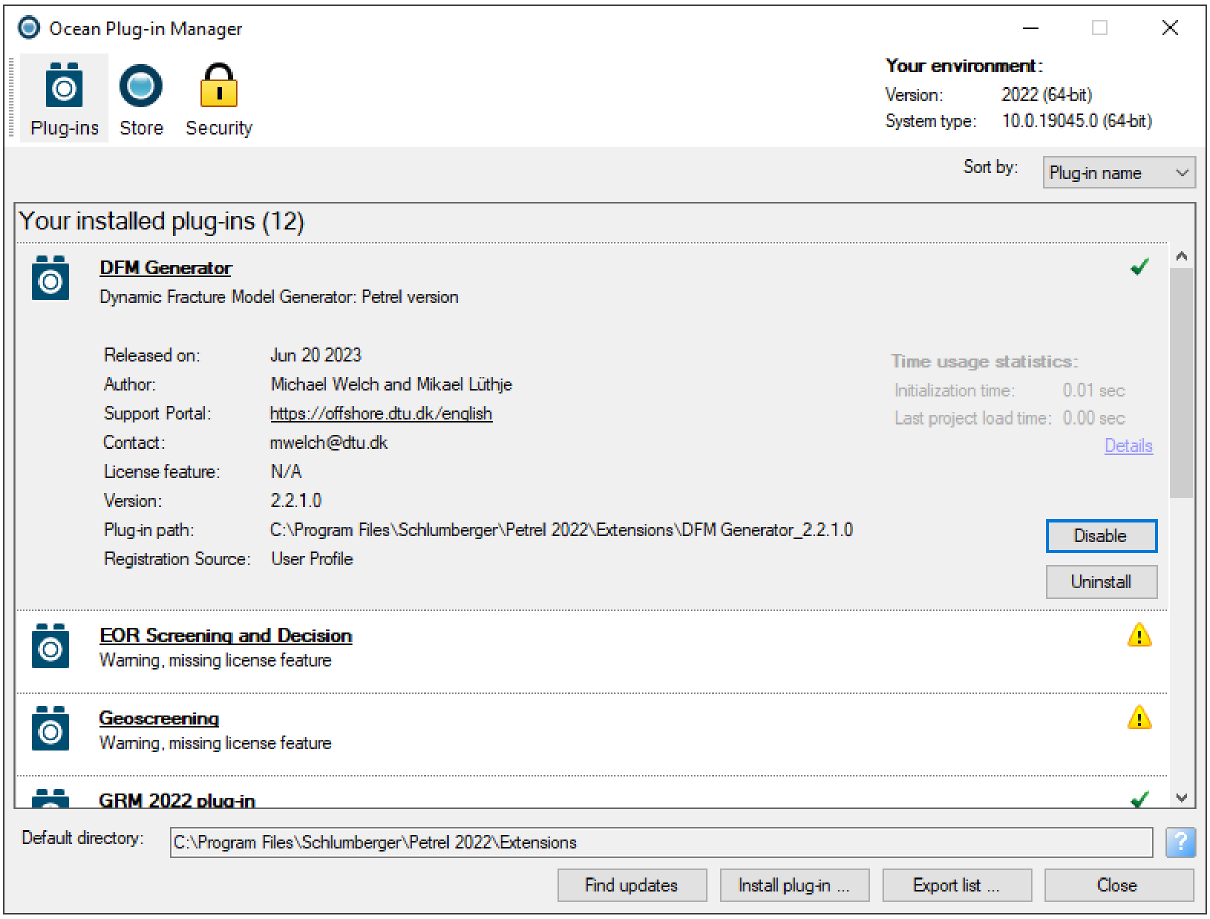

Figure 3: Ocean Plug-in Manager window showing DFM Generator has been correctly installed.

2 Using the DFM Generator plugin

The DFM Generator plugin is not intended to replace the current Petrel fracture modelling tools and workflow, but to complement and enhance them. It can be used to build either explicit Discrete Fracture Network models (DFNs), in which fractures are represented individually as geometric objects, or implicit fracture models, in which the fracture network is represented by continuum properties such as fracture density.

Conventional fracture modelling workflows typically have five stages, shown on Figure 4:

1. Log fractures in wells, from core or borehole images, to determine the fracture density and principal fracture orientations (or sets).
2. Derive “fracture drivers”, i.e. grid properties representing the inferred fracture distribution across the area of interest. These drivers may be purely empirical (e.g. generated by extrapolating and contouring the fracture density measured in different wellbores), semi-empirical (e.g. distance to the nearest fault, if faults are thought to be the main control on fracture density), or mechanical (e.g. curvature as a proxy for strain, or elastic stress calculated around major faults using numerical techniques).
3. Use the fracture drivers to generate a Discrete Fracture Network model (DFN) and/or an implicit fracture model (fracture properties) stochastically. In stochastic fracture modelling, fractures are simply placed at random locations and assigned arbitrary geometries in such a way that the fracture density is proportional to the driver property.
4. Calibrate the fracture model against observed fracture data from the boreholes, and outcrop analogues if available. If a good match is not obtained, reject and rebuild the model. Assess the uncertainty inherent in the fracture model.
5. Upscale the final model, and use it to generate the required fracture properties (e.g. permeability or stiffness tensors).

The stochastic modelling technique used in stage 3 has a number of drawbacks. Since it is not based on the physics of fracture nucleation and growth, the resulting models may not honour the geology or geomechanics. It is therefore often necessary to tweak the models to get a good match with observed data, which can be time consuming. Furthermore the models give no constraint on parameters that are important in controlling fluid flow through the fractures but difficult to measure in the subsurface, such as fracture size, connectivity and anisotropy.

The DFM Generator plugin provides a tool for building fracture models (either DFNs or implicit models) dynamically, by simulating the process of fracture nucleation and growth based on fundamental geomechanics. This has several advantages over stochastic models:

- **Accuracy:** Since it simulates the physical processes of fracture nucleation and propagation, the DFM Generator output will automatically honour the geology, geomechanical properties and structural evolution of the reservoir. The level of detail of the output will reflect the input data, but even using very simple input data the output will be geologically consistent.
- **Ease of use:** Since DFM Generator is a deterministic model it is easy to set up and run. It can typically be run with data already available in most models (e.g. curvature of top horizon, default mechanical properties for the known lithology) – a first pass model can be generated very quickly. However it is also easy to build more detailed models, e.g. taking into account lateral variability in mechanical properties due to facies variations.

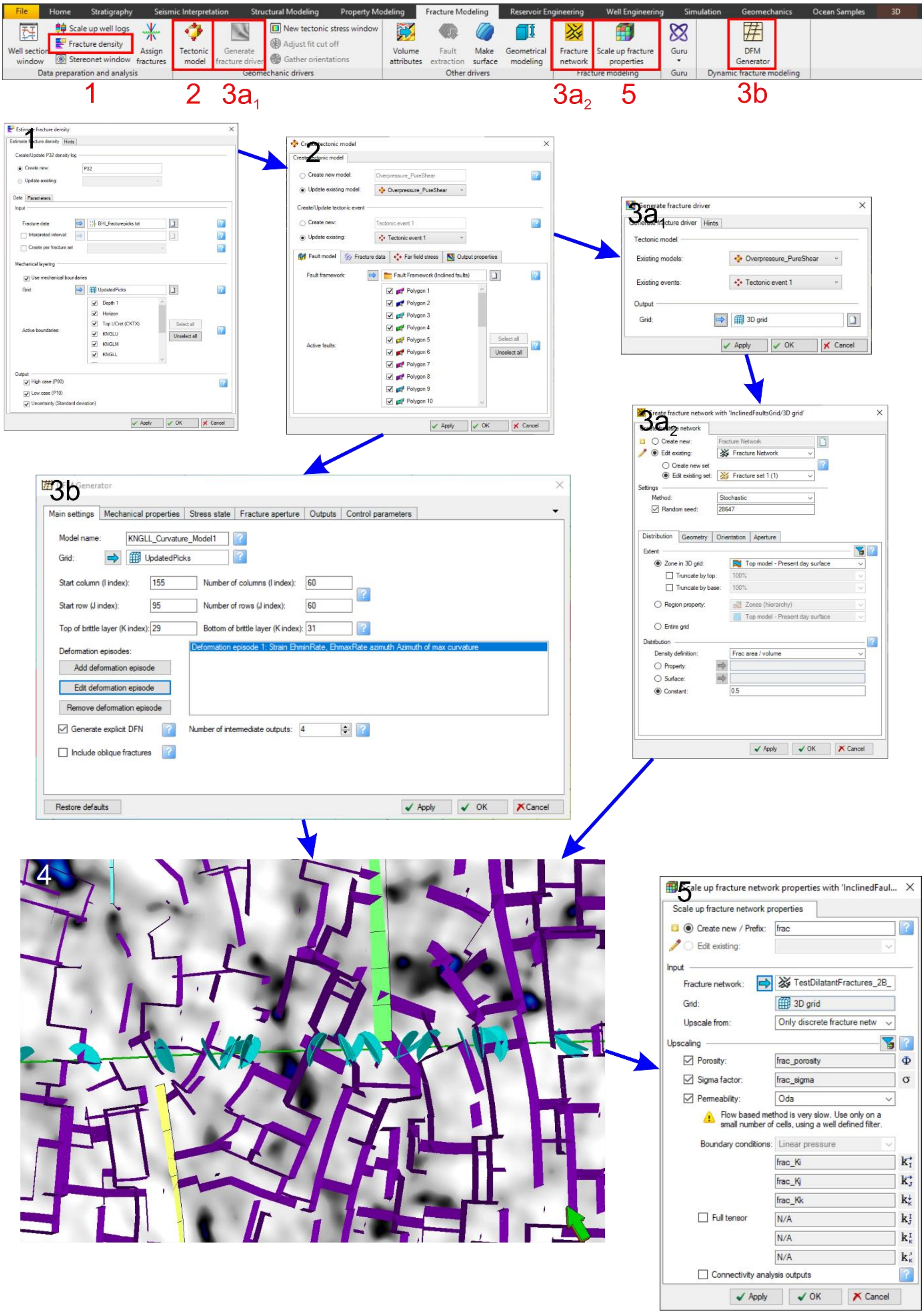

Figure 4: DFM Generator fits into the standard Petrel fracture modelling workflow and tools.

- **Uncertainty analysis:** Since it simulates fracture growth, DFM Generator can automatically generate multiple realisations representing different stages of fracture development, which will accurately reflect the distribution of fractures at low, moderate or high intensity of deformation.
- **Usefulness:** DFM Generator calculates many parameters (e.g. fracture size distributions, fracture connectivity) that are required for complex flow modelling calculations. These properties emerge from the simulation, and cannot be calculated directly from the input data.

The DFM Generator plugin therefore fits into stage 3 of the fracture modelling workflow, as a substitute for stochastic fracture modelling (see Figure 4). It can also assist with stages 4 and 5.

To get the most value from DFM Generator, it is helpful to have some understanding of the regional geological history and the origin of the fractures. However DFM Generator can be run, and give useful results, even where there is very limited input and calibration data, for example in an exploration setting (as will be demonstrated in Section 3). When more data becomes available to constrain the simulation, this can easily be applied to obtain more accurate and reliable results.

The three key inputs required to run DFM Generator are:

- Mechanical layering: DFM Generator models layer-bound fractures. The fractured layers can be identified from:
  - Fracture density logs from wells, showing the heavily fractured intervals.
  - Production or drilling data, indicating high permeability intervals or intervals of mud loss.
  - Wireline log data, indicating brittle lithologies (typically characterised by high density, low porosity and low clay content).

  If multiple fractured layers are present, DFM Generator can be run independently for each.

- The deformation load responsible for generating the fractures (e.g. horizontal extensional strain, fluid overpressure, uplift and decompaction). The best method for determining this will depend on the origin of the fractures:
  - For fractures resulting from a regional extensional strain, it may be sufficient to simply apply a regional azimuth of extension and typical tectonic strain rate (e.g. 0.01-0.001/ma).
  - For fractures resulting from folding and flexure, an estimate of the local strain rate and orientation can be derived from the curvature of the fractured layer (see Appendix 1). More complex structural restoration tools can also be used.
  - For fractures developed in the damage zones around larger faults, the Petrel Tectonic Modelling module can be used to calculate the local elastic strain magnitude and orientation.
  - For fractures resulting from fluid overpressure, an estimate of the rate of overpressure accumulation.
  - For fractures resulting from uplift and decompaction, the induced strain (including thermal and fluid pressure effects) can be calculated from the rate of uplift, the overburden density and the geothermal gradient.

- Mechanical properties of the fractured layers: The availability of mechanical property data can vary considerably:
    - For first pass models, it is usually sufficient to use standard mechanical property values for the lithology in question, which can be derived from published literature. DFM Generator will default to typical mechanical property values for a brittle limestone.
    - For more detailed modelling, mechanical properties (especially Young's Modulus and friction coefficient) can often be calculated from wireline log data, using standard or proprietary algorithms (e.g. Chang et al. 2006).
    - In some cases, mechanical properties may have been mapped out in 3D, e.g. from seismic inversion. If available, such data can easily be applied to the model in the form of a grid property.

An ideal fracture modelling workflow using the DFM Generator plugin is as follows:

1. Log fractures in wells, from core or borehole images, to determine the fracture density and principal fracture orientations, and to identify the fractured layers.
2. Calculate or estimate the input data required for the simulation, in particular:
    - the fractured layers,
    - the mechanical fracture drivers (e.g. horizontal strain rate and orientation), and
    - the mechanical properties.
3. Use the input data to generate a dynamic DFN and/or implicit fracture model for the fractured layer(s). Select to generate multiple intermediate outputs.
4. Calibrate the fracture model against observed fracture data from the boreholes. Determine which of the outputs give the best match for the observed fracture distribution, and use the others for uncertainty estimation. If none of the models give a good match, adjust the input parameters (especially strain rate and orientation) and rerun.
5. Upscale the final model, and use it to generate the required fracture properties. Even if only an implicit fracture model is generated, this contains information on fracture length, connectivity and anisotropy as well as a breakdown of the distribution of fractures between different sets, which can be used to calculate anisotropic permeability tensors. DFM Generator can also output bulk rock stiffness and compliance tensors.

A full description of the forward modelling algorithm used by DFM Generator is presented in publication by Welch et al. 2019 and Welch et al. 2020. More information and tips on the input data for dynamic fracture models can also be obtained from these publications.

3 Building a first pass model

This section will run through how to build a first pass fracture model for a geological structure where there is limited input data. We will use the Drenthe salt diapir, onshore Netherlands as an example. This structure has been studied as a potential geothermal energy prospect, producing from the fracture chalk layer overlying the diapir, and as a result a 3D seismic survey has been undertaken across the diapir, and several wells drilled. This data is publicly available from NLog (https://www.nlog.nl/en/).

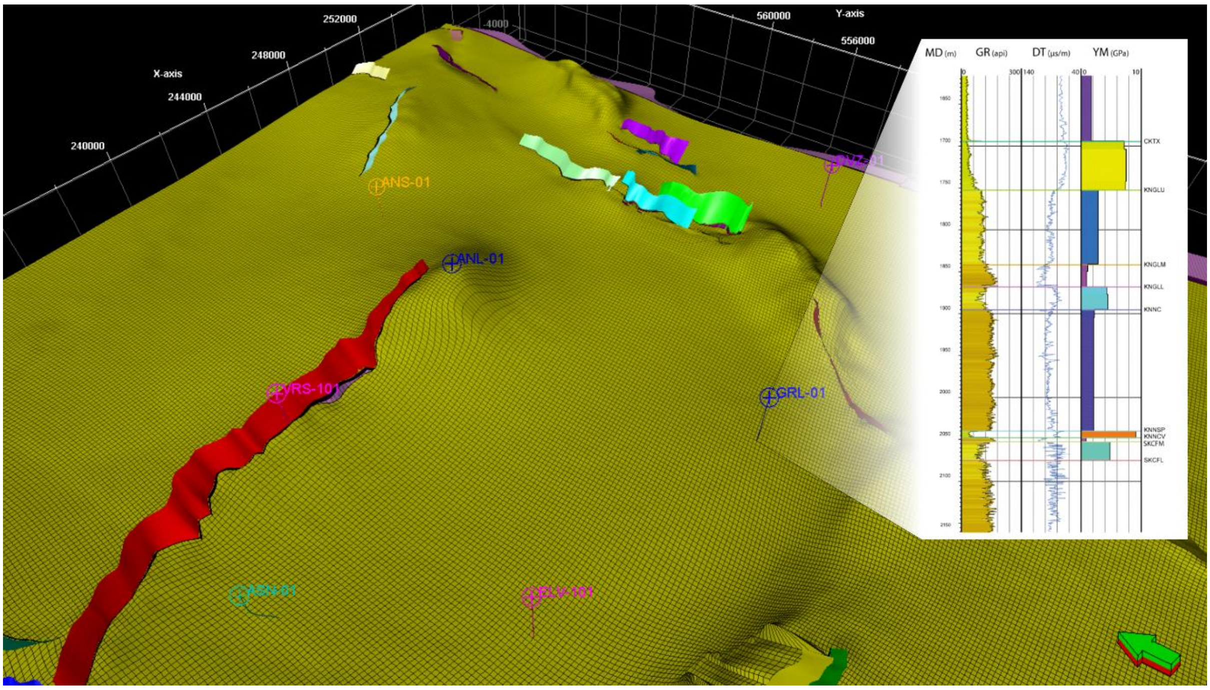

Figure 5: A basic 3D structural model and grid of the Drenthe salt diapir (left), and a well log identifying the brittle KNGLL layer (right).

We will assume that a basic 3D structural model and grid have already been built in Petrel, as shown in Figure 5. From the well logs, we have identified the Lower Holland Member (KNGLL) as a potential fractured layer. This corresponds with K-layers 29-31 in our grid. The fractures in this layer are most likely formed by flexural strain, in response to growth of the diapir. For a first pass model, the Petrel curvature modelling tools can be used to generate grid properties representing the minimum and maximum horizontal strain rate, and the minimum strain azimuth, as described in Appendix 1.

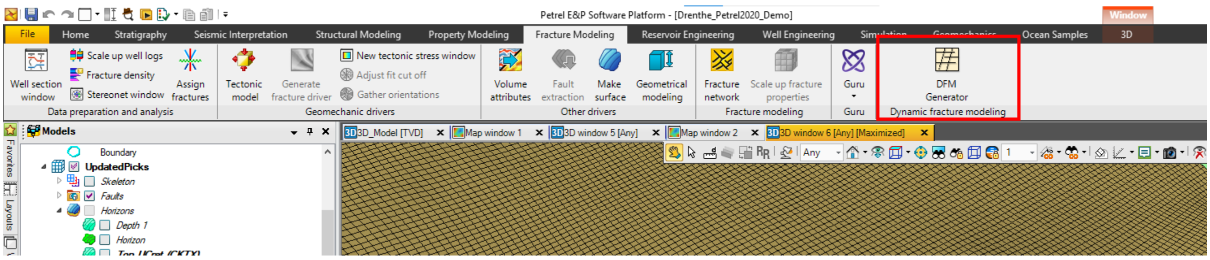

Figure 6: The DFM Generator launch button.

To launch DFM Generator, click on the **DFM Generator** button on the *Fracture Modelling* ribbon tab (Figure 6). This will open the *DFM Generator dialog* (Figure 7). The *DFM Generator dialog* contains several tabs of user-adjustable settings; however most of these can be left at their default values. For the first pass model we only need to use the *Main settings* tab.

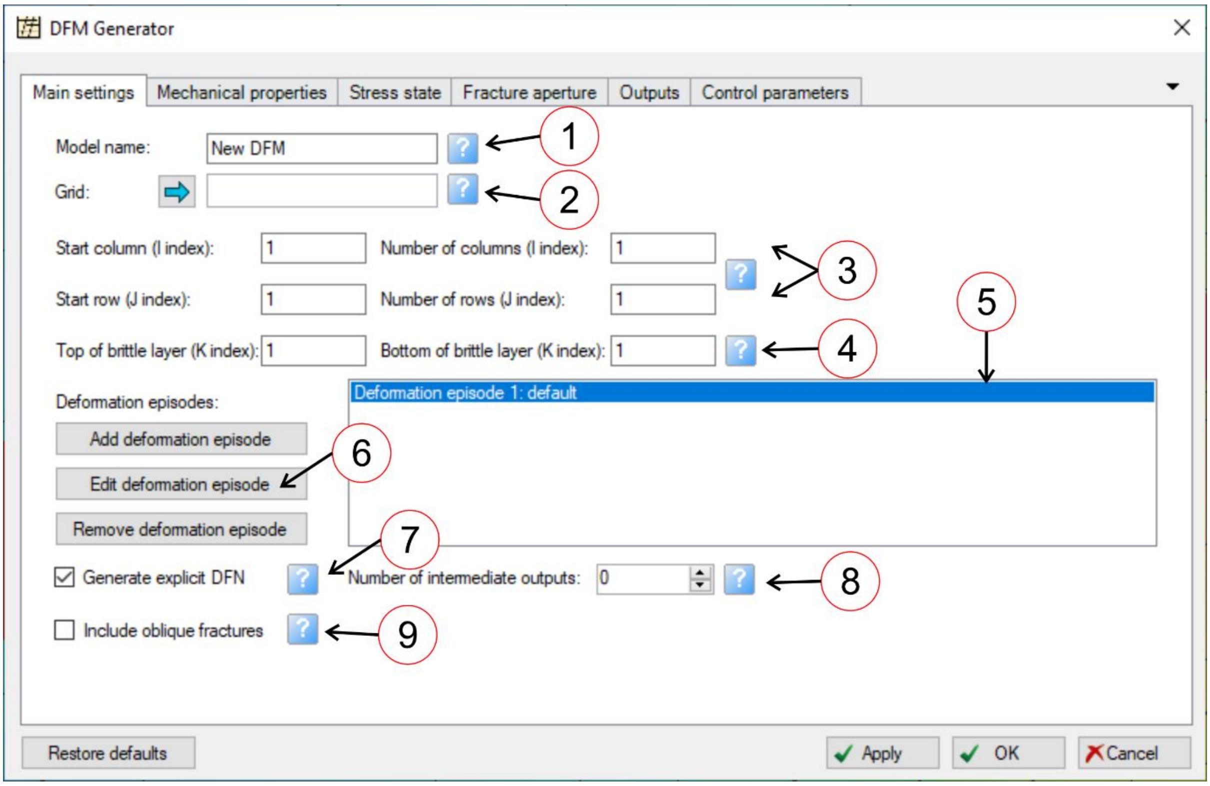

Figure 7: The Main Settings tab of the DFM Generator dialog, populated with typical properties and values.

3.1 Input parameters

The only input parameter that *must* be specified to run DFM Generator is the Grid object; all other parameters can be left at their default values. However you will usually specify values or properties for the other input parameters on the *Main settings* tab before running a model, as shown on Figure 7:

- **Model name (1):** Assigning each fracture model a unique name will help to identify the model output. This name will be applied to the output folders containing the implicit fracture properties as well as to the DFN objects. For additional trackability, the settings used to generate each fracture model are recorded on the comments tab of the output folders and DFNs.
- **Grid (2):** This is the Petrel grid object that represents the model geometry, and contains the properties used for input parameters. Output properties will also be written to this grid object. Select the required grid object in the Models pane and click the blue arrow to drop it in the presentation box (Figure 8).
- **Rows and columns (3):** Use the column and row parameters to limit the model to a part of the overall grid. Leave these blank to run the model on the entire grid. In this example, the Drenthe salt diapir comprises only a small part of the total grid, approximately 60x60 cells starting from column 155, row 95.

  Since it may take several hours for a model to run on a large grid, it is recommended to run a test on a small part of the grid beforehand (e.g. 5x5 cells) to check that it is working correctly, before running it on the full grid.

- **Brittle layer (4):** Use these parameters to specify the brittle layer containing the layer-bound fractures, by specifying the K-index of grid layers at the top and bottom of the brittle layer. Note that the brittle layer can span multiple grid layers, and need not correspond to stratigraphic layers or zones. If the brittle layer spans multiple grid layers, the grid cells will be amalgamated into "cell stacks" for the purpose of calculation. In the output properties, the same values will be applied to every cell in the stack; these represent average values for the property across the entire brittle layer. In this example, the top of the brittle layer we are modelling corresponds to grid layer 29, and the bottom of the brittle layer corresponds to grid layer 31 (Figure 11).

  It is only possible to model one brittle layer at a time. However it is of course possible to rerun DFM Generator several times to generate multiple fracture models for different brittle layers.

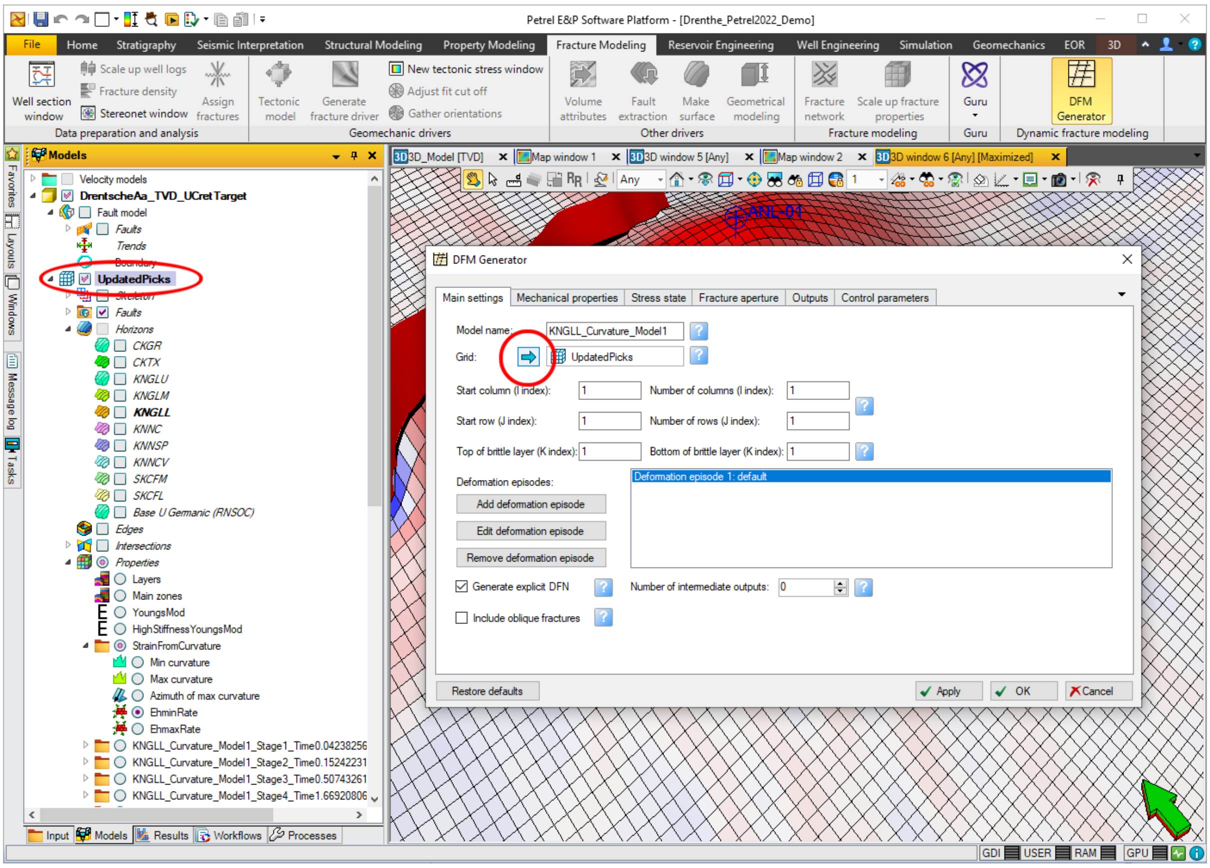

Figure 8: Specify a grid object by selecting it on the Models pane and clicking on the blue arrow to drop it into the DFM Generator dialog.

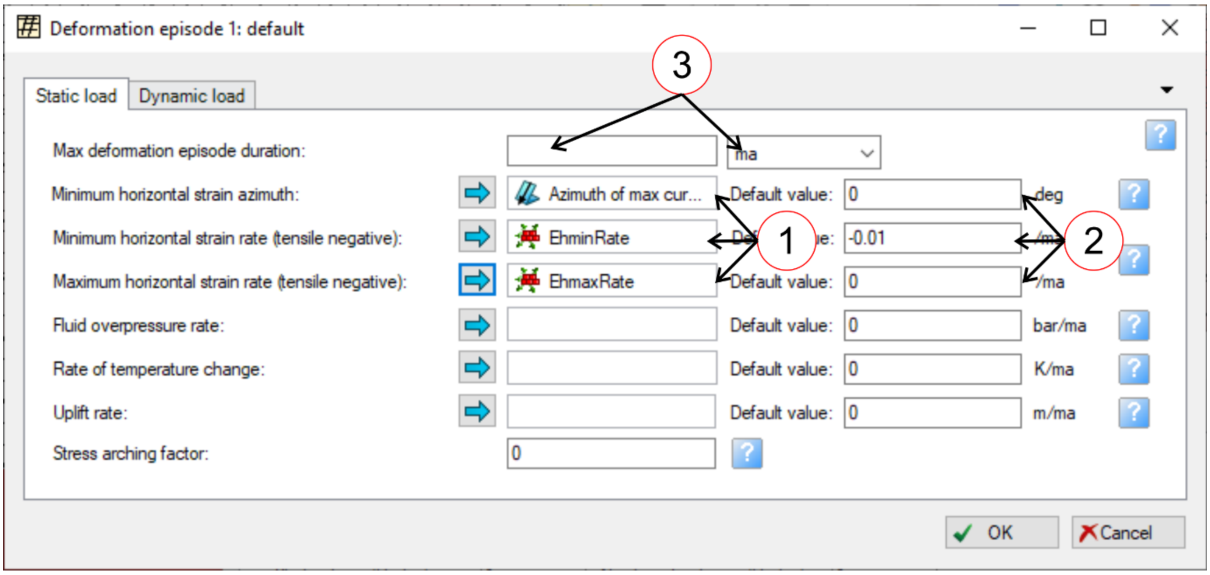

Figure 9: The Deformation episode dialog for the first (default) deformation episode.

- **Deformation load (5):** In this example we model a single deformation episode in which fracturing is driven by a biaxial applied horizontal strain. For this we can use the default deformation episode that is generated when DFM Generator is launched. Click on the **Edit deformation episode (6)** button to open the *Deformation episode dialog* for this episode (Figure 9).

  A horizontal strain deformation load comprises three components: a minimum and maximum strain rate and an orientation. Each component can be specified either as a grid property or as a default value applicable to all cells. Specifying a strain component as a grid property allows lateral variations in the strain orientation and rate to be modelled. To do this, select the required grid property in the Models pane and click the blue arrow to drop it in the appropriate presentation box, as shown on Figure 9 (1) and Figure 10. If no grid property is specified (i.e. the grid property presentation box is blank), the default value specified in the corresponding text box (2) will be applied to all cells; this value will also be applied to any cells where the specified grid property is null or undefined.

  Following geomechanical convention, extensional strain is represented by negative values and compressional strain by positive values. The minimum horizontal strain rate therefore represents the maximum extension rate, and should have predominantly negative values; typical values are in the range -0.001 to -0.1/ma. The maximum horizontal strain rate may be negative, zero or positive. The minimum horizontal strain azimuth represents the direction of maximum extension, and controls the orientation of the fractures in the DFN: the primary fracture set will strike perpendicular and the secondary fracture set will strike parallel to this. If a grid property is specified, the fractures will bend at cell boundaries to reflect changes in the strain orientation.

  Horizontal strain data can be obtained by many different methods, and the most appropriate method will depend on the origin of the fractures and available data. In this example we use the strain data calculated from horizon curvature (Figure 11). Horizon curvature is often an easy way to generate strain data for a first pass model; the procedure for this is described in Appendix 1.

  It is possible to define a duration for the deformation episode (3); however if this is not specified the deformation will continue until the fracture network has either reached saturation or stopped growing. We will therefore leave this box blank. When all the data has been entered, click the **OK** button to close the *Deformation episode dialog* and return to the *DFM Generator dialog*.

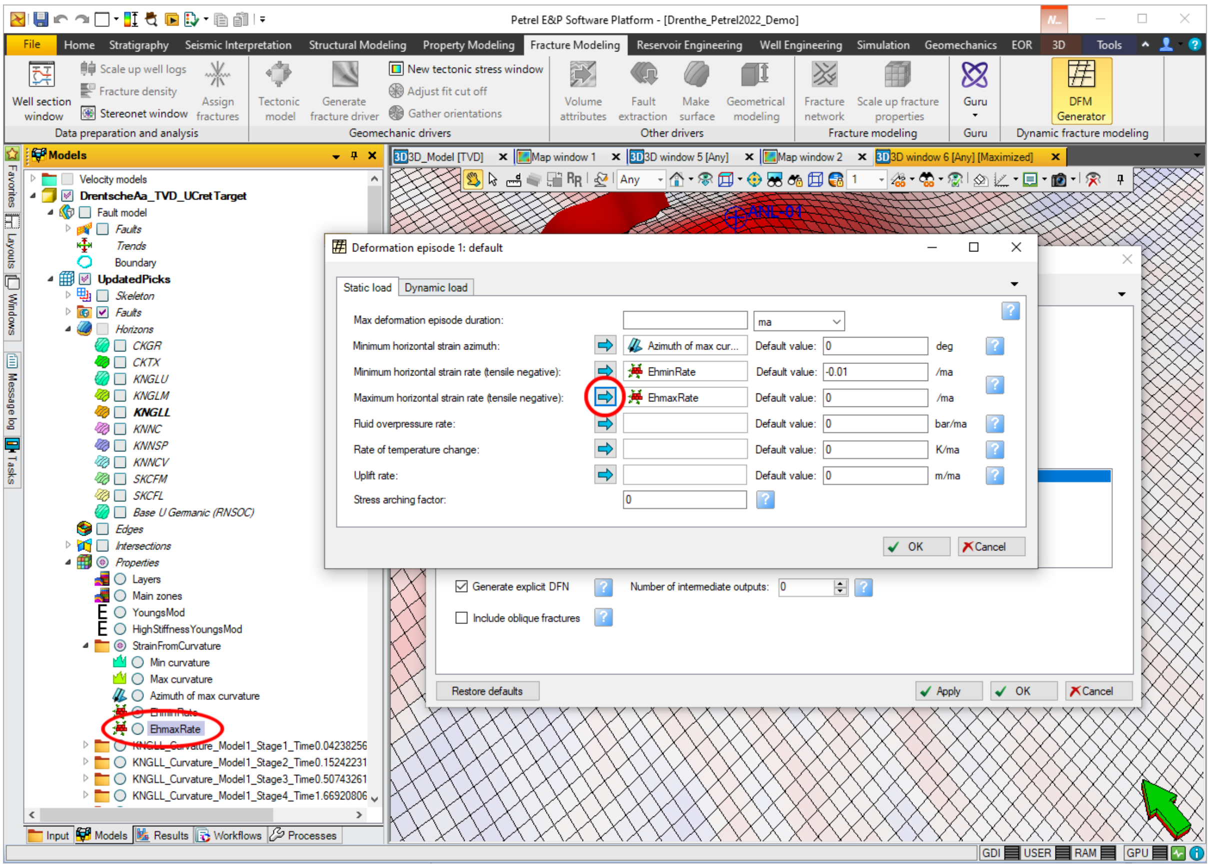

Figure 10: Specify the horizontal strain by selecting grid properties on the Models pane to represent the minimum horizontal strain orientation and the minimum and maximum strain rates, and then clicking on the blue arrows to drop them into the appropriate presentation boxes on the Deformation Episode dialog.

- **Generate explicit DFN (7):** By default, DFM Generator generates both implicit fracture data (e.g. fracture density and porosity, output as grid properties) and explicit DFNs (geometric representations of the fracture network, output as Petrel DFN objects). However if the explicit DFN is not required, it is possible to reduce the runtime by generating only the implicit fracture data.
- **Number of intermediate outputs (8):** One advantage of dynamically simulating fracture growth is that it is easy to generate a series of models representing intermediate stages in the evolution of the fracture network. These can be useful for uncertainty analysis, as they provide multiple, geologically realistic fracture models representing low, mid and high case scenarios. Use this parameter to set the number of intermediate models to be output. A separate and clearly labelled folder containing implicit fracture data, and a clearly labelled Petrel DFN object, will be output for each intermediate stage. If this parameter is set to 0, output will only be generated for the final, fully developed fracture network. However in this example we will output 4 intermediate models (Figure 11).
- We will not activate the option to **Include oblique fractures (9)** in this example. More details of this option are given in Section 5.1.

3.2 Running the model

For this first pass model, we will leave the mechanical properties, stress state and other input parameters at their default values. The defaults have been chosen to give reasonable results for a typical fractured carbonate or tight sandstone layer. Information on how to use these input parameters to build more accurate fracture models is given Section 5, and a full list of input parameters is given in Appendix 2.

Once the input data has been entered, the DFM Generator dialog should look as shown in Figure 11. Click the **Apply** or **OK** buttons to run the model (both buttons have the same effect, but **OK** closes the module after running). Updates on the different stages of the calculation will be given in the *Message Log* window and progress will be shown on the Petrel Progress Bar. It is possible to abort the calculation by clicking the **Stop** button, although any results generated up to that point will be lost. A full model on a large grid may take several hours to run, but a model of a small subsection of a grid (e.g. 5x5 cells) should run in a minute or less.

When the model has finished running, the output data should be visible in the *Models* pane, below the selected Grid object.

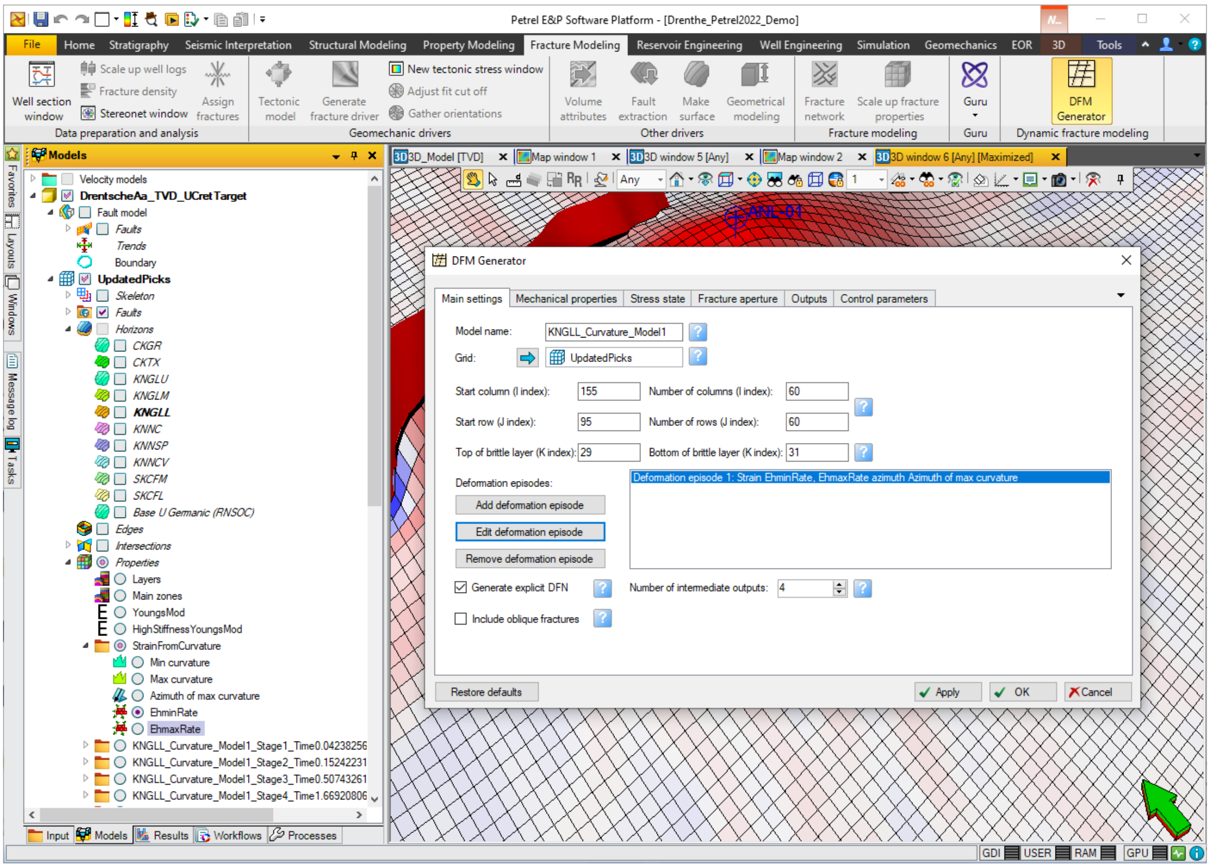

Figure 11: Ready to run the first pass model. The model has been set to run on a 60x60 cell portion of the grid covering the Drenthe salt diapir, with a brittle layer spanning grid layers 12 to 20 inclusive. There is a single deformation episode, and grid properties have been selected to represent the minimum (most tensile) horizontal strain orientation, and the minimum and maximum horizontal strain rates.

4 Output Data

By default, DFM Generator generates both implicit fracture data (e.g. fracture density and porosity, as grid properties) and explicit DFNs (geometric representations of the fracture network, as Petrel DFN objects). A model run with default settings will contain four fracture sets: a vertical Mode 1 set and an inclined Mode 2 set striking perpendicular to the minimum horizontal strain (labelled *HMin_Vertical* and *HMin_Inclined* repsectively), and a vertical Mode 1 set and an inclined Mode 2 set striking perpendicular to the maximum horizontal strain (labelled *HMax_Vertical* and *HMax_Inclined*). However not all of these sets will necessarily contain fractures. Within each set, the fractures are subdivided into two types: Layer-bound Fractures, which are elongated rectangular fractures spanning the entire brittle layer, and microfractures, which are small circular fractures contained within the brittle layer. Fracture density data will be generated individually for each set and type, as well as for the combined fracture network.

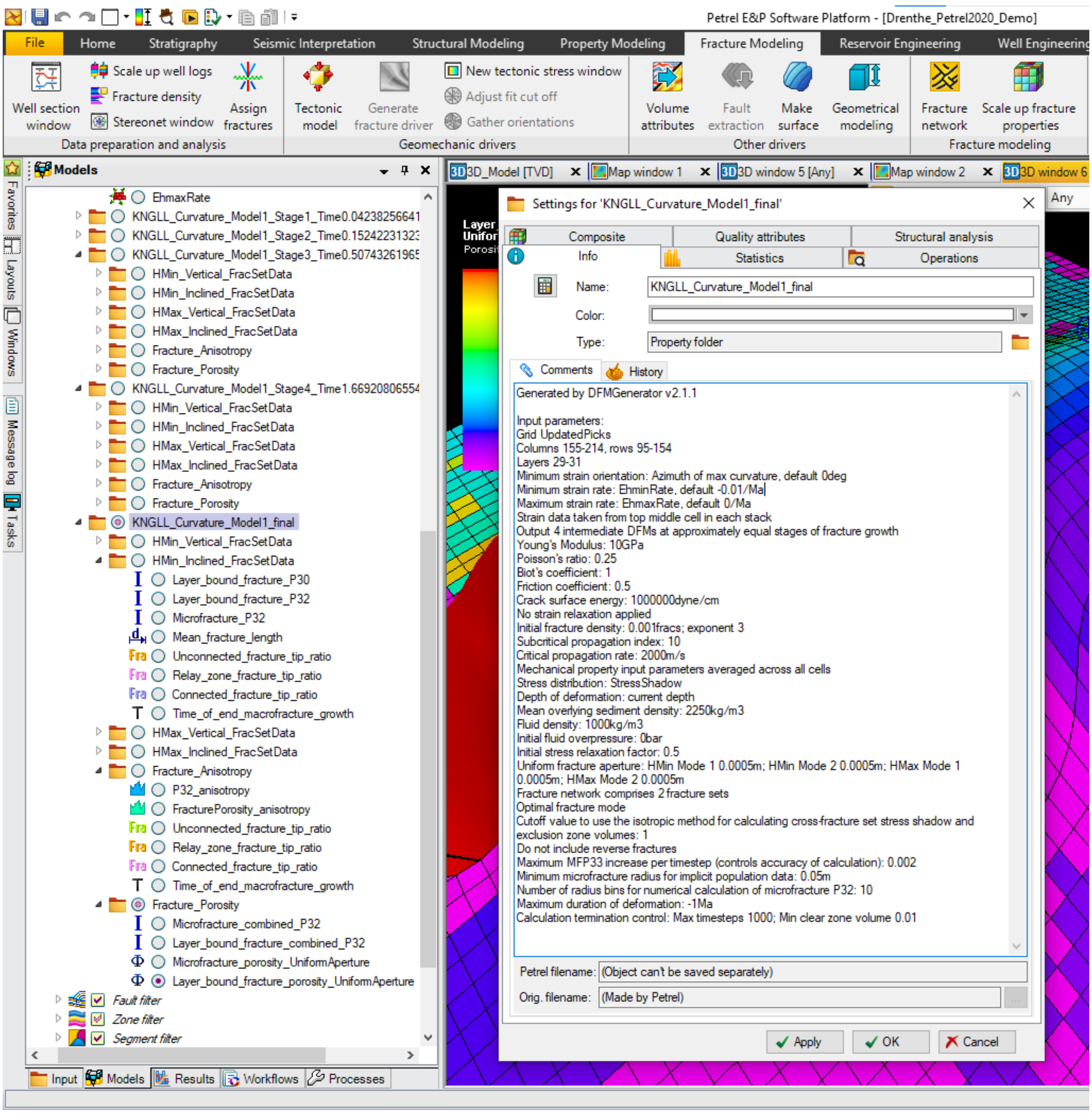

Figure 12: Output folders for the implicit fracture data, and the comments tab of the folder Settings dialog, showing the input parameters used to run the model.

4.1 Implicit output data

The implicit fracture data is output as grid properties, which are placed in folders underneath the *Grid* object in the *Models* pane. One folder is generated for each intermediate stage model that is output, as well as a folder for the final stage representing the fully developed fracture network. All folders will be clearly labelled with the model name and stage, and the input parameters used to generate the model will be listed on the *Comments* tab of the folder's *Settings dialog* (Figure 12).

Each folder contains multiple subfolders: one subfolder containing data for each of the fracture sets, and two subfolders containing data representing the entire network (Figure 13). These folders will contain the following grid properties:

- **In the fracture set folders:**
    - Total density of layer-bound fractures in the set, expressed as the number of fractures per unit volume (P30).
    - Total density of layer-bound fractures in the set, expressed as the total fracture area per unit volume (P32).
    - Total density of microfractures in the set, expressed as the total fracture area per unit volume (P32).
    - Mean length of layer-bound fractures, given by P32/(layer thickness*P30).
    - Proportion of layer-bound fracture tips from this fracture set that are not connected to other fractures (I-nodes).
    - Proportion of layer-bound fracture tips from this fracture set that are linked to other fractures by relay zones (R-nodes). These may be either hard-linked or soft-linked relay zones (see Section 5.8).
    - Proportion of layer-bound fracture tips from this fracture set that terminate against other perpendicular or oblique fractures (Y-nodes).
    - The time taken for the fracture set to reach saturation and stop growing.
- **In the Fracture Anisotropy folder:**
    - P32 anisotropy, defined as (P32(HMin)–P32(HMax)) / (P32(HMin)+P32(HMax)), where P32(HMin) and P32(HMax) are the total area of fractures perpendicular to the minimum and maximum horizontal strain directions respectively. P32 anisotropy is 0 for a fully isotropic fracture set, where P32(HMin)=P32(HMax), and 1 for a fully anisotropic fracture set, where P32(HMax)=0.
    - Porosity anisotropy, defined as (ϕ(HMin)–ϕ(HMax)) / (ϕ(HMin)+ϕ(HMax)), where ϕ(HMin) and ϕ(HMax) are the total porosity of fractures perpendicular to the minimum and maximum horizontal strain directions respectively.
    - Proportion of layer-bound fracture tips from the entire fracture network that are not connected to other fractures (I-nodes).
    - Proportion of layer-bound fracture tips from the entire fracture network that are linked to other fractures by relay zones (R-nodes).
    - Proportion of layer-bound fracture tips from the entire fracture network that terminate against other fractures (Y-nodes).
    - The time taken for the entire fracture network to reach saturation and stop growing.
- **In the Fracture Porosity folder:**
    - Total density of all layer-bound fractures in the entire fracture network, expressed as the total fracture area per unit volume (P32).
    - Total density of all microfractures in the entire fracture network, expressed as the total fracture area per unit volume (P32).

- Total porosity of all layer-bound fractures in the entire fracture network. Fracture porosity is calculated using the parameters specified on the *Fracture aperture* tab of the *DFM Generator dialog* (see Section 5.6).
- Total porosity of all microfractures in the entire fracture network.

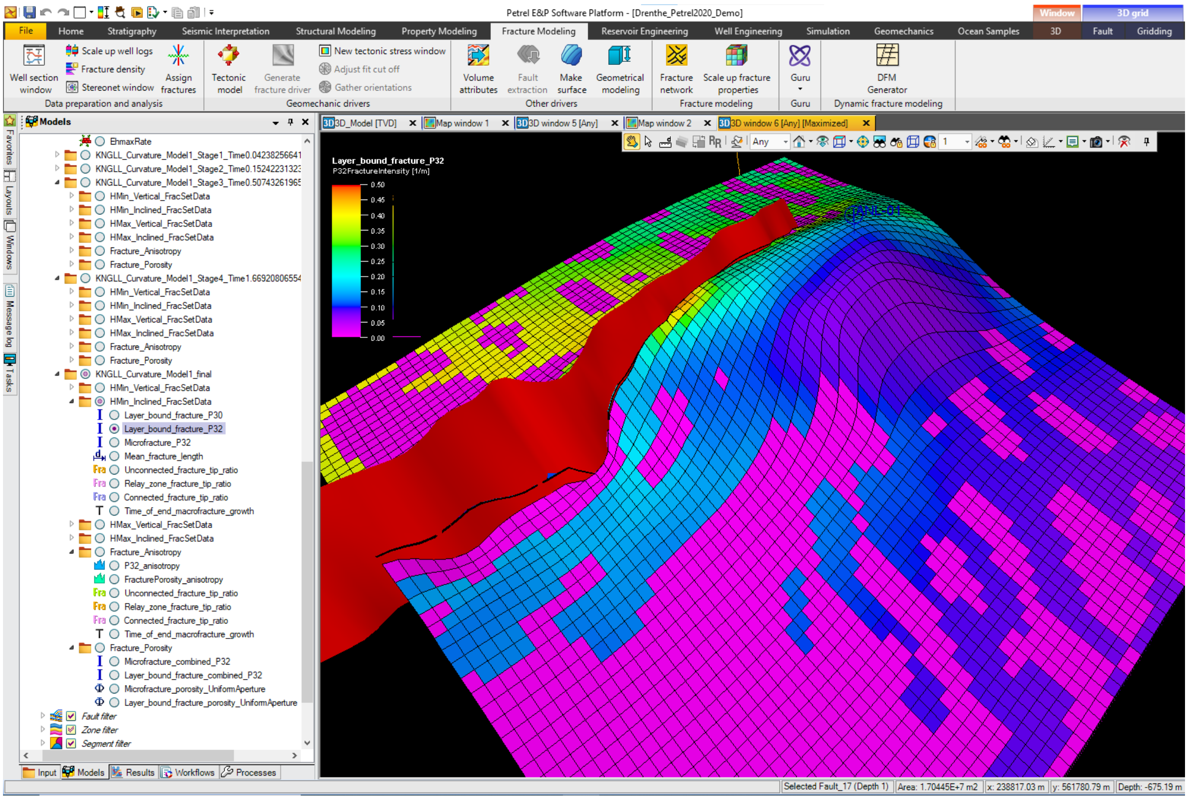

Figure 13: Implicit fracture output data. In this example, the total area (P32) of Mode 2 layer-bound fractures striking perpendicular to the direction of maximum extension, P32(HMin), is displayed as a grid property.

4.2 Explicit Petrel DFNs

If the **Generate explicit DFN** option is selected, a Petrel DFN object will also be generated for each intermediate stage, and for the final fracture network. These will be placed at the bottom of the Petrel *Model* folder containing the selected *Grid* object. They will likewise be clearly labelled with the model name and stage, and the input parameters used to generate the model will be listed on the *Comments* tab of the DFN object's *Settings dialog*.

Each DFN comprises a collection of planar geometric objects representing individual fracture segments (Figure 14). Long layer-bound fractures may comprise multiple segments, allowing them to bend across cell boundaries, following lateral variations in the strain orientation. The fractures are classified into sets based on strike, but not dip (so that the sets contain both vertical and inclined fractures). Each fracture segment is also assigned an aperture and an associated permeability, calculated as defined on the *Fracture aperture* tab of the *DFM Generator dialog* (see Section 5.6).

It is possible to see the growth of individual fractures by comparing the intermediate stage DFNs (Figure 15). This will also demonstrate the linking of initially independent fractures to form long fractures connected by relay zones, and the late growth of secondary fractures connecting the long primary fractures.

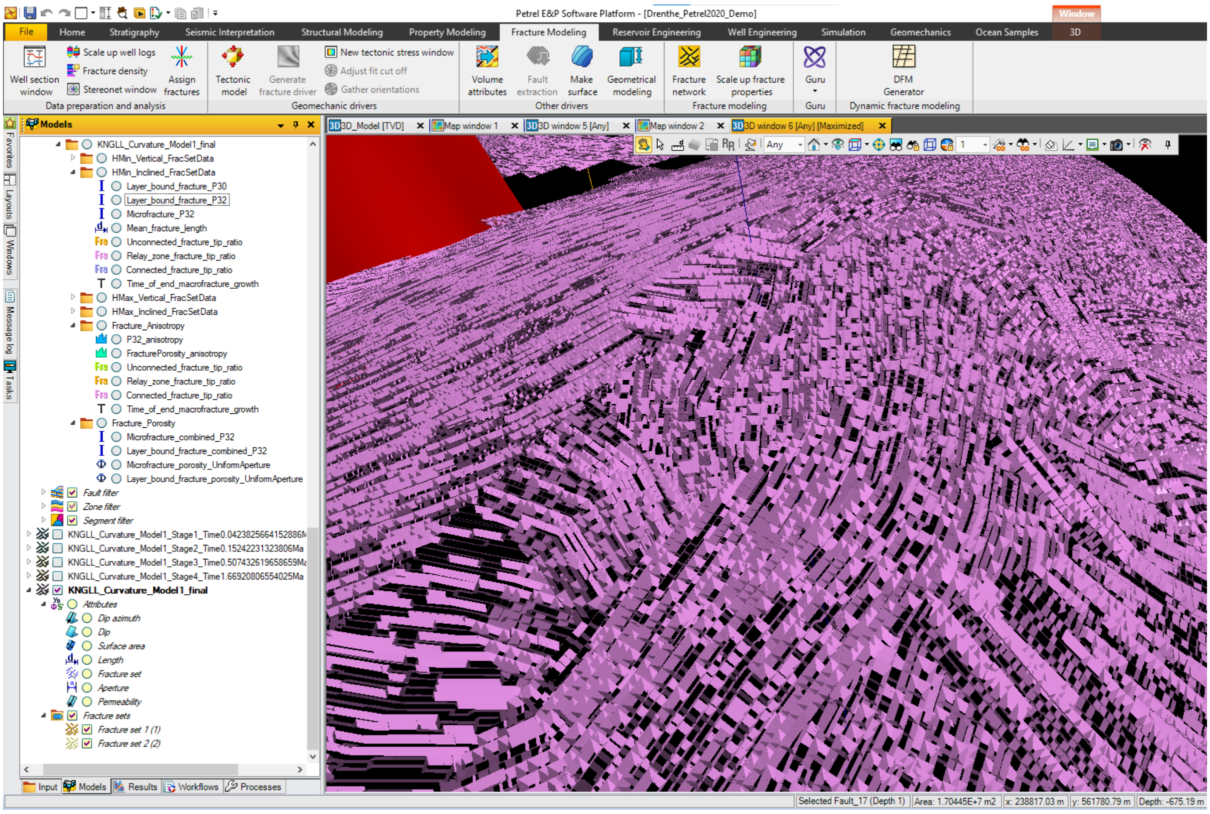

Figure 14: Explicit DFN representing the final fracture network as a collection of planar objects. Individual fractures may comprise multiple planar segments, allowing them to bend between cell stacks.

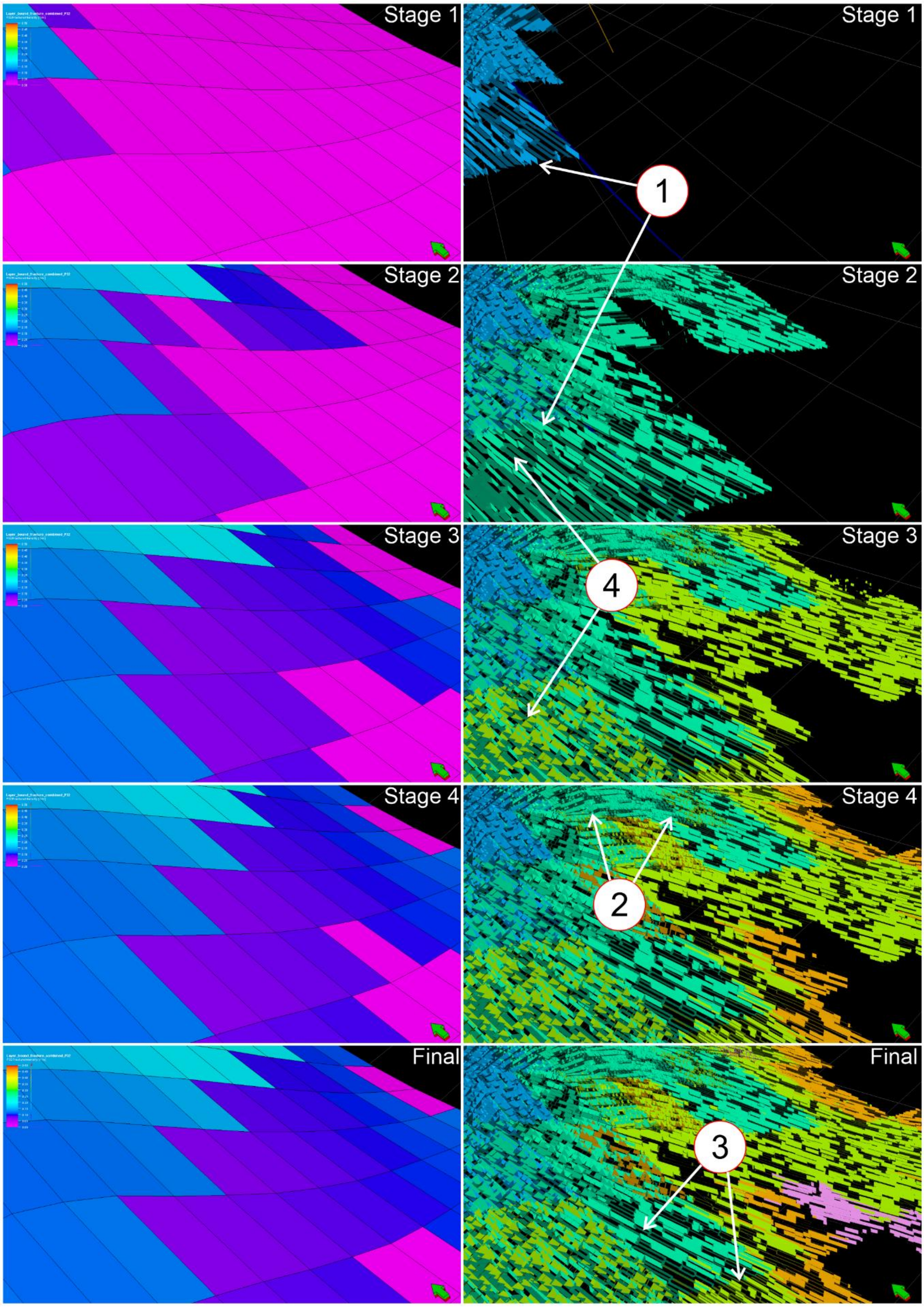

Figure 15: Implicit and explicit output representing the intermediate and final stages of fracture growth. On the explicit DFNs, note: 1) the growth of individual fractures between intermediate stages, 2) the curved profile of the long primary fractures across cell boundaries, 3) the relay zones along these primary fractures, where initially independent fractures have linked up due to stress shadow interaction, and 4) the short secondary fractures connecting the primary fractures.

Microfractures are not represented in the explicit DFN unless specified on the *Control Parameters* tab of the *DFM Generator dialog* (see Section 5.8). It is recommended that this is only done for small models (e.g. near-wellbore models), to avoid excessive runtimes. If microfractures are included in the explicit DFN, they will appear as polygonal representations of planar circular objects, contained within the brittle layer (Figure 16)

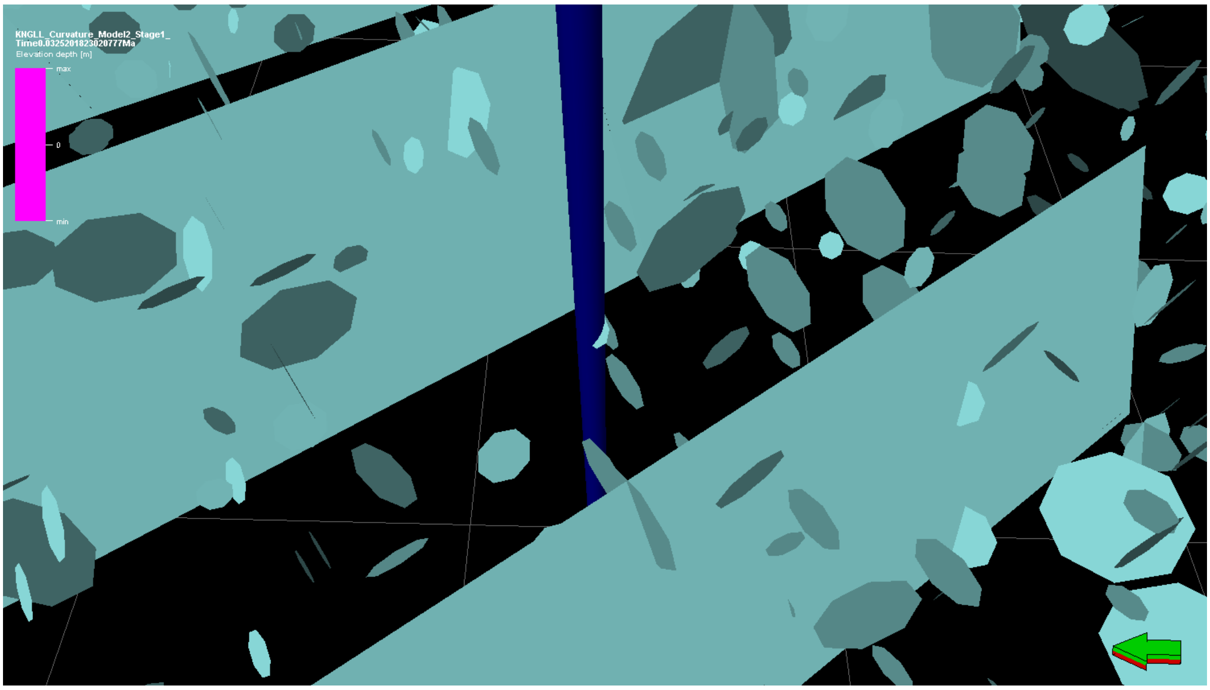

Figure 16: Near wellbore explicit DFN including circular microfractures, represented here as 8-sided polygons.

5 Advanced modelling

Section 3 demonstrated how to use DFM Generator to build a first pass fracture model with very limited data. However as more data becomes available, for example detailed mechanical property data from seismic inversion, or better information about the timing and conditions of fracture growth, we will want to integrate this into the model to better constrain the fracture network. In this section we will look at some of the more common input parameters and options that can be used to do this. A full list of all input parameters and options is given in Appendix 2.

5.1 Oblique and strike-slip fractures

By default the fracture networks include 4 fracture sets, comprising vertical and inclined fractures striking perpendicular to the minimum and maximum horizontal stresses. Displacement will therefore always be perpendicular to fracture strike. This is generally sufficient for modelling deformation induced by extensional strain in intact rock. However certain scenarios (e.g. isotropic extensional strain generating polygonal fractures, compressional maximum strain generating strike-slip fractures, or if there are pre-existing fractures oblique to the principal horizontal stresses), it is necessary to include more fracture sets in the model and to allow strike-slip displacement on those that are oblique to the principal stresses.

To do this, select the **Include oblique fractures** option on the *Main Settings* tab (8 on Figure 7). This will generate a model with 6 fracture sets striking at 30° intervals. The number of fracture sets can be modified further (see Section 5.7), but 6 sets are recommended because the oblique fracture sets will be close to the optimal orientation for strike-slip displacement, while the fracture sets striking perpendicular to the principal strains can accommodate orthogonal displacement. Increasing the number of fracture sets will increase the runtime of the model and the amount of output data, so it is recommended that this option is only activated if oblique fractures are necessary to accurately represent the fracture network geometry.

5.2 Deformation episodes and geological loads

It is possible to model fracture networks developed over multiple deformation episodes. To add additional deformation episodes to the model, click the **Add deformation episode** button on the *Main settings* tab. This will add an additional deformation episode to the list and open the *Deformation episode dialog* to allow you to define the duration and load for the new episode (Figure 17). It is also possible to edit the load for an existing episode by selecting it in the list and clicking the **Edit deformation episode** button, or to remove it from the list by clicking the **Remove deformation episode** button.

When multiple deformation episodes are defined, the loads for each episode will be applied sequentially to the same fracture network. Thus fractures that nucleated in response to one deformation load may grow further in response to a subsequent load. A finite duration should therefore be specified for all deformation episodes except for the final one, which can be left undefined so that the fracture network will grow to saturation (1). The time units for each deformation episode can also be selected; typically you would use ma for geological deformation, years for deformation on production timescales, and seconds to model events such as well tests.

It is possible to output the fracture network at the end of each deformation episode, by setting the **Interval between intermediate outputs** on the *Outputs* tab to **Deformation episodes** (see Appendix 2.5).

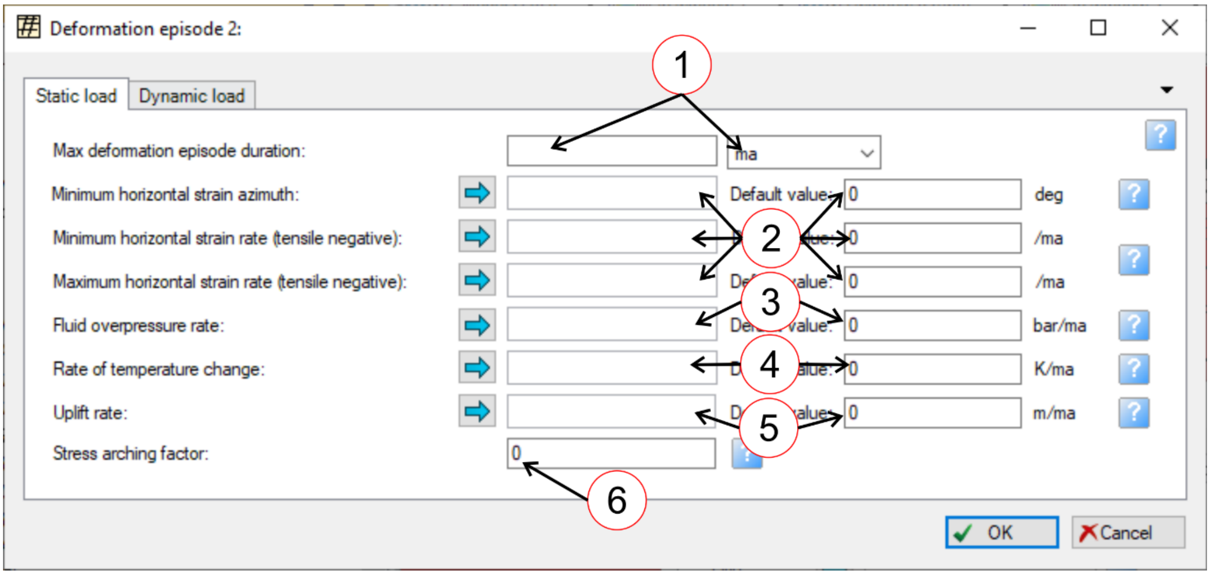

Figure 17: The Deformation episode dialog for an additional deformation episode, showing the input parameters for a geological deformation load on the Static load tab.

In addition to horizontal strain (2), geological deformation can also be driven by fluid overpressure, uplift and decompaction and thermal effects, and it is also possible to define these components of the deformation load on the *Static load* tab of the *Deformation episode dialog*, as shown on Figure 17:

- To generate deformation driven by fluid overpressure, specify the rate of increase in fluid overpressure, either as a uniform default value or as a grid property (3). This will increment the fluid overpressure defined on the *Stress state* tab (Section 5.5), or at the end of the previous deformation episode. The internal strain and resulting stress generated by the change in fluid pressure is calculated as described in Miller (1995), and will be dependent on the elastic properties of the rock (the Young's Modulus and Poisson's ratio), the Biot coefficient and the initial porosity.
- Internal strain can also be generated by thermal contraction, for example if the heatflow changes over time or if a cold fluid is injected into the rock. The rate of temperature change can also be specified either as a uniform default value or as a grid property (4). The internal strain and resulting stress generated by the change in fluid pressure is also calculated as described in Miller (1995), and will be dependent on the thermal expansion coefficient of the rock, defined on the *Mechanical properties* tab (Section 5.4). Note that temperature change resulting from uplift should not be specified here, as this will be calculated automatically.
- To generate deformation by uplift and decompaction, specify the uplift rate either as a uniform default value or as a grid property (5). It is assumed that the erosion rate equals the uplift rate so that the lithostatic stress will decrease at a rate determined by the uplift rate and the mean sediment density, defined on the *Stress state* tab (Section 5.5). The effects of thermal contraction and hydrostatic fluid pressure change due to uplift are also included in the calculation of stress change, which follows the algorithm of Miller (1995). The temperature and hydrostatic fluid pressure changes resulting from uplift are calculated from the geothermal gradient and fluid density specified on the *Stress state* tab (Section 5.5), so should not be specified directly; however any temperature change not proportional to the geothermal gradient, and any fluid overpressure change not proportional to the hydrostatic pressure, can be specified in addition.

Multiple load components can be specified, in which case the in situ stress state will reflect the combined effect of all of them.

Internal strain resulting from fluid overpressure or thermal contraction can sometimes be partially supported by stress arching, in which the overburden is wholly or partially supported from the sides so that the vertical effective stress is reduced. This can be incorporated into the model by specifying a **Stress arching factor (6)** between 0 and 1:

- A stress arching factor of 0 implies no stress arching: the absolute vertical stress remains constant (equal to the lithostatic load) and internal strain is accommodated only by change in the horizontal stress.
- A stress arching factor of 1 implies complete stress arching: there is no vertical strain, so the internal strain is accommodated by change in the horizontal and vertical stress. The absolute vertical stress will therefore deviate from the lithostatic load.

Note that stress arching does not apply to horizontal strain or uplift loads, as these cannot be supported from the side.

5.3 Dynamic loads

It is also possible to model fracture networks developed as a result of production- or injection-related pressure and stress changes, by using the output from Eclipse, Intersect or Visage simulations to define a deformation load. This is done on the Dynamic load tab of the tab of the *Deformation episode dialog*, as shown on Figure 18.

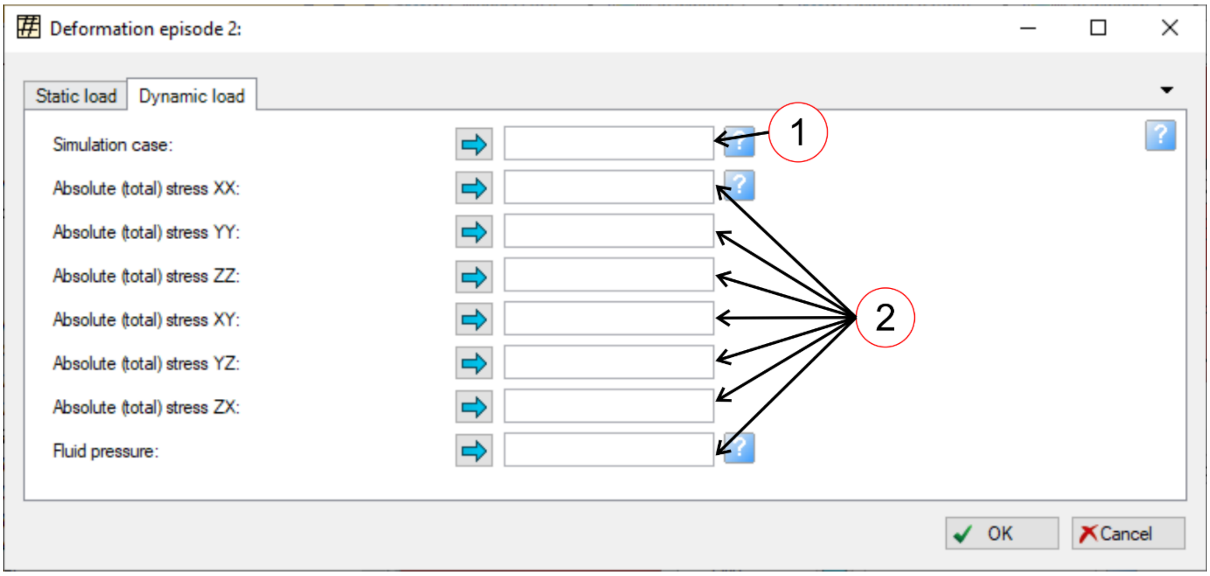

Figure 18: The Deformation episode dialog for an additional deformation episode, showing the input parameters for a production- or injection-related deformation load on the Dynamic load tab.

To create a production- or injection-related deformation episode, either create a new deformation episode, or edit an existing deformation episode to remove all the geological load components defined on the *Static load* tab. Then click on the *Dynamic load* tab. Firstly it is necessary to define the simulation case from which the stress and pressure values should be taken; to do this, select a simulation case from the *Cases* pane in Petrel and click on the blue arrow to drop it into the Simulation case presentation box (1). You should ensure that the simulation case was run on the same grid as specified for the DFM Generator model. Then go to the Results pane in Petrel, select the required absolute stress and fluid pressure components (these will usually be found in the Dynamic properties folder) and use the blue arrows to drop these into the respective presentation boxes on the *Dynamic load* tab (2). It is not necessary to define all components; it is possible to define only a fluid pressure load, only a vertical stress load, or an orthotropic stress load (σ_{yz}=σ_{zx}=0 and are not specified). Note that the stress components should represent absolute stress; effective stress will then be calculated from the combined absolute stress and fluid pressure data, taking into account the Biot coefficient defined on the *Mechanical properties* tab (Section 5.4).

The duration of the deformation episode is not specified but is taken from the simulation case results. The deformation episode will be split into multiple sub-episodes, each representing one timestep in the simulation case, except for the first timestep will define the initial conditions. It is possible to output the fracture network at the end of each timestep, by setting the **Interval between intermediate outputs** on the *Outputs* tab to **Deformation episodes (**see Section 5.7).

5.4 Mechanical properties

By default, the mechanical properties are set to values typical for brittle limestone, chalk or tight sandstone. However if more specific mechanical data is available, this can be used instead. This is done on the *Mechanical Properties* tab (Figure 19).

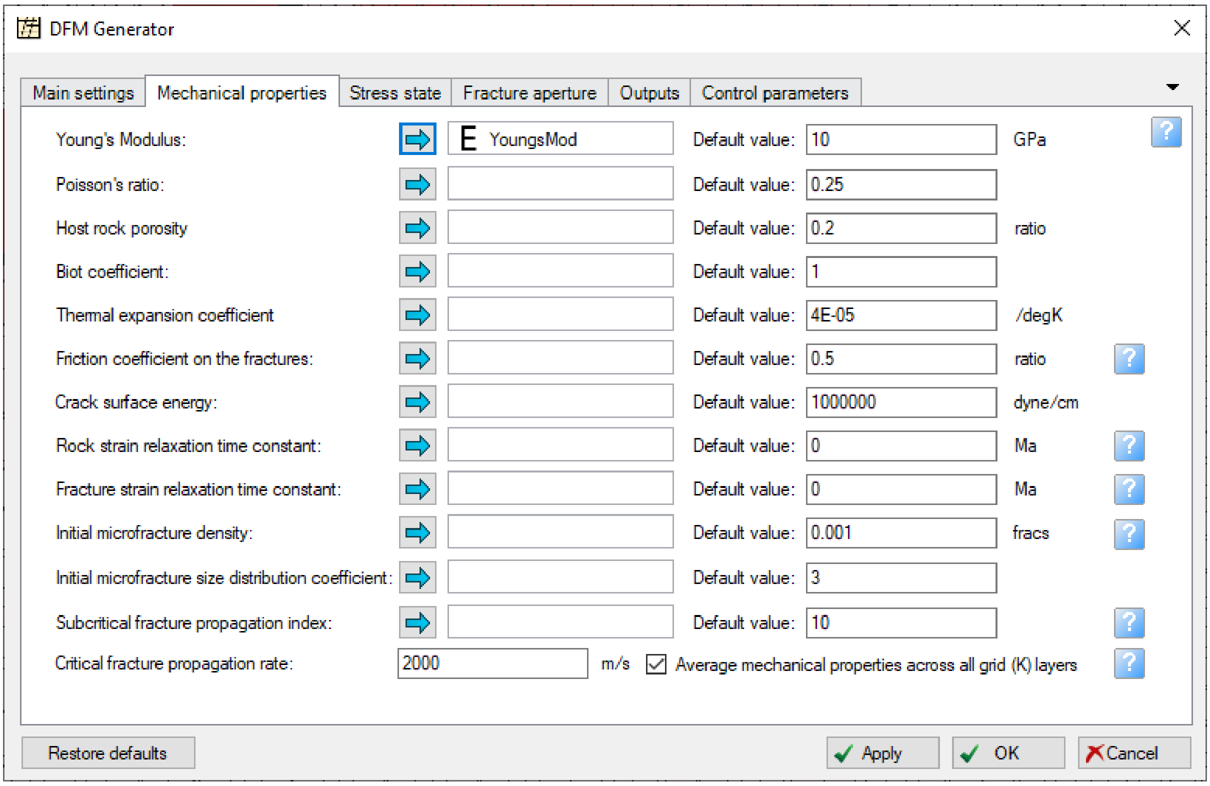

Figure 19: The Mechanical Properties tab, with a grid property specified for the Young's Modulus.

Like the strain data, the mechanical properties can be specified either as grid properties or as default values applicable to all cells. Specifying a mechanical property as a grid property allows modelling of lateral variation in the property, for example if Young's Modulus has been calculated by inversion of a seismic cube. To do this, select the required grid property in the *Models* pane and click the blue arrow to drop it in the appropriate presentation box (Figure 19). If no grid property is specified (i.e. the grid property presentation box is blank), the specified default value will be applied to all cells in the model. This may be more appropriate if limited mechanical property data is available, for example if mechanical property measurements are available from lab tests on a few core samples. The default property value will also be applied to any cells where a specified grid property is null or undefined.

Mechanical property data should be supplied in project units. It is advisable to use the appropriate Petrel template for the grid properties, to ensure correct unit conversions, although DFM Generator will also accept grid properties with other templates. A full list of mechanical properties is given in Appendix 2.2.

5.5 Stress state at the time of deformation

By default, the in situ stress at the start of the deformation episode is set to a mixed elastic-viscoelastic equilibrium stress state for hydrostatic fluid pressure at the current depth of burial of the brittle layer. The vertical effective stress is therefore set to the current lithostatic stress minus the hydrostatic fluid pressure, and the horizontal effective stress is equal to the 0.5/(1-ν) times the vertical effective stress. This stress state can be adjusted by changing the following input parameters on the *Stress state* tab (Figure 20):

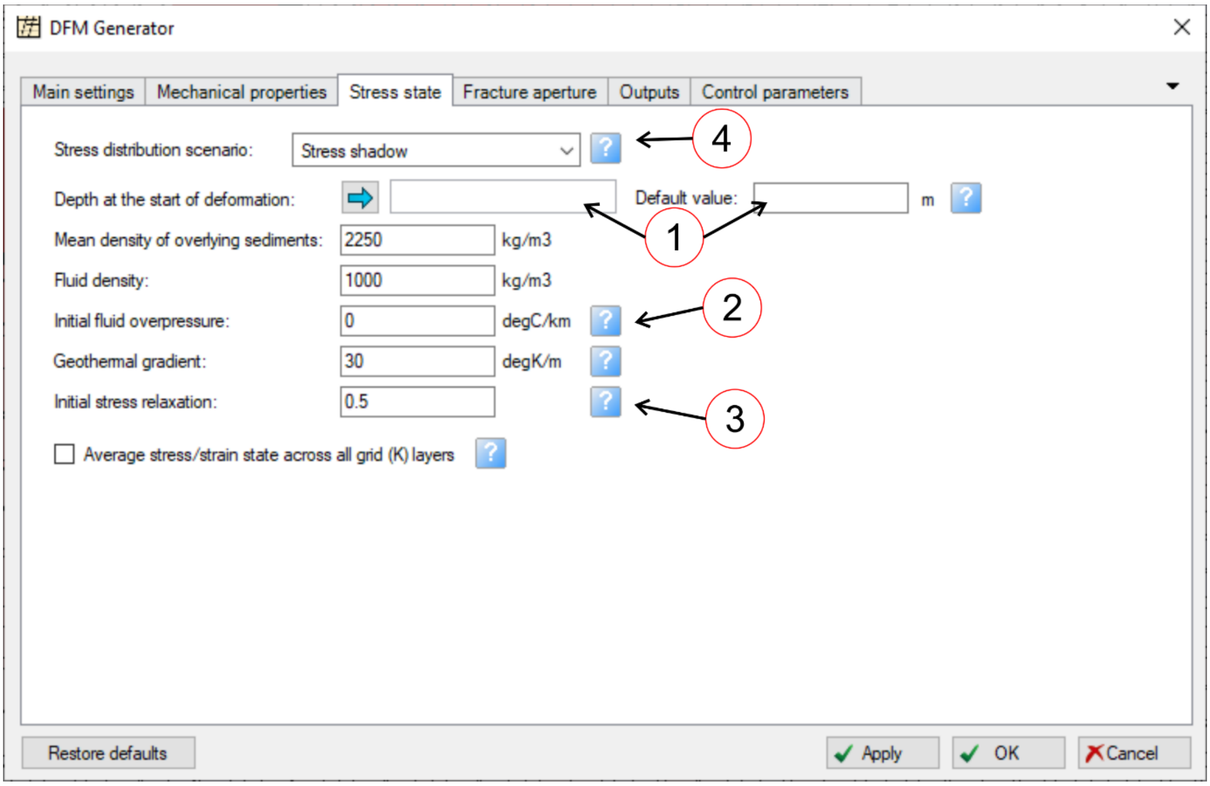

Figure 20: The Stress State tab.

- **Depth at time of deformation (1):** If the fractures developed at a more shallow or deeper depth of burial than the current depth of the brittle layer, this depth should be specified here, either as a uniform default value or as a grid property (in project units, positive downwards). The initial in situ stress in each cell will be adjusted to match the equilibrium stress state at the specified depth of burial. If both input boxes are blank, the current depth of burial will be used to calculate vertical effective stress.
- **Initial fluid overpressure (2):** If the brittle layer was overpressured at the start of deformation specify the overpressure here (the overpressure represents the excess fluid pressure above the hydrostatic gradient). A high fluid overpressure will promote the development of Mode 1 dilatant fractures rather than Mode 2 shear fractures. If the fluid overpressure develops during the deformation episode, this should be specified as a load component (Section5.2).

- **Initial stress relaxation (3):** This controls the initial horizontal stress σ_{h0}', prior to deformation. A value of 1 represents viscoelastic equilibrium, where horizontal stress is equal to vertical stress ($\sigma_{h0}'=\sigma_v'$); a value of 0 represents elastic equilibrium, where $\sigma_{h0}'=\nu/(1-\nu)*\sigma_v'$. For values between 0 and 1, σ_{h0}' will be calculated by linear interpolation; by default it is set to 0.5, where $\sigma_{h0}'=0.5/(1-\nu)*\sigma_v'$, a typical value for brittle layers in the subsurface. If this input box is left blank, the initial horizontal stress will be set to the critical stress state (i.e. the Mohr-Coulomb failure stress) for each cell.

It is also possible to modify or turn off the stress shadow effect by changing the stress distribution scenario (4). This controls the spacing of the fractures:

- In the **Stress shadow** scenario, fractures are surrounded by a stress shadow, in which the applied strain is accommodated by displacement on the fractures rather than elastic strain in the rockmass. Stress shadows of adjacent fractures may not overlap, and propagating fractures will terminate if they intersect the stress shadow around another fracture. This creates a minimum spacing for the layer-bound fractures, which is proportional to the layer thickness.
- In the **Evenly distributed stress** scenario, there are no stress shadows around the fractures and therefore no minimum fracture spacing; instead, displacement on the fractures reduces the elastic strain equally throughout the rockmass. This can cause a large number of fractures to be generated.

Most outcrop studies show a fracture spacing proportional to the layer thickness (e.g. Bai & Pollard 2000, Bai et al. 2000), so it is recommended that the stress shadow scenario is used, unless the evenly distributed stress scenario is necessary to accurately represent the fracture network geometry.

Other parameters on the *Stress state* tab are described in Appendix 2.3.

5.6 Fracture aperture

In order to calculate fracture porosity, it is necessary to specify a method for determining the fracture aperture. This will also determine the aperture property assigned to individual fractures in the explicit DFNs. This is done on the *Fracture aperture* tab.

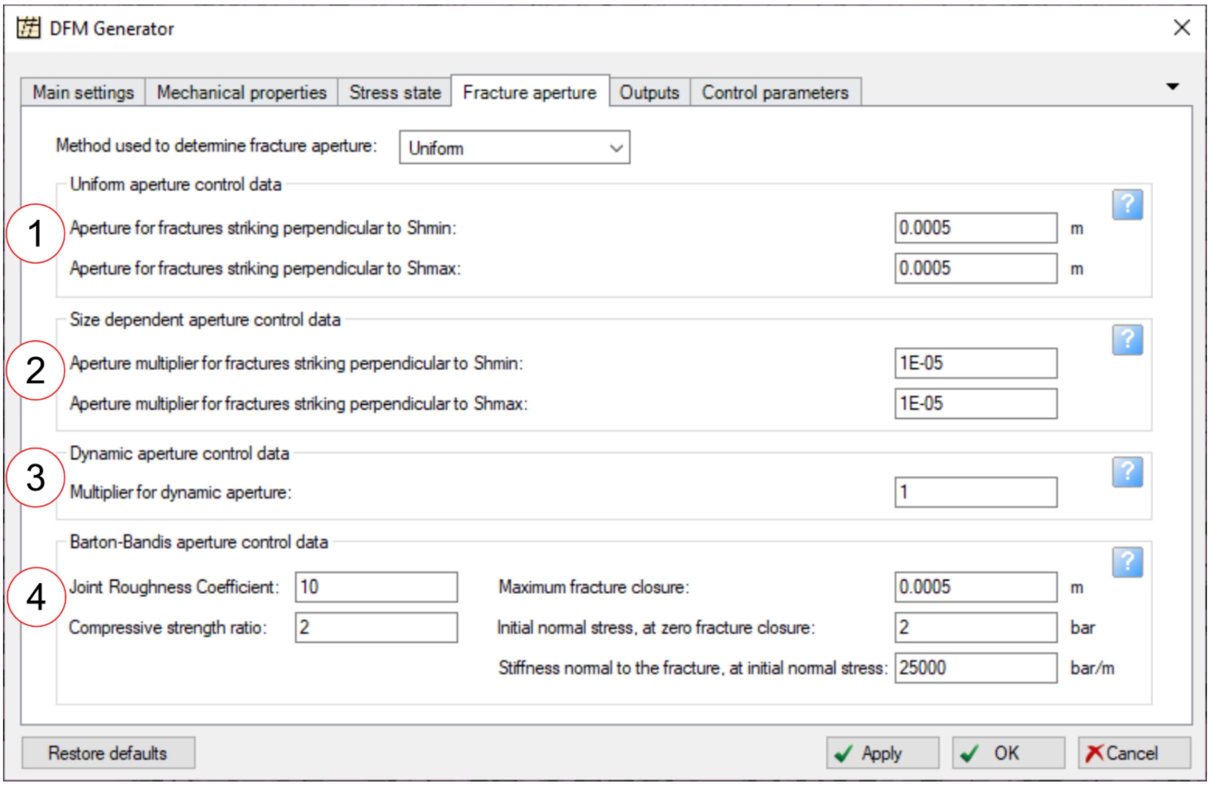

Figure 21: The Fracture aperture tab, showing the four methods available for determining fracture aperture. Normally, only the options relating to the selected aperture determination method will be active; the others will be greyed out.

Four methods are available for determining fracture aperture:

- **Uniform aperture (1):** All fractures are assigned an arbitrary user-specified aperture. Different apertures can be specified for fractures striking perpendicular to the minimum and maximum horizontal stress.
- **Size-dependent aperture (2):** Fracture aperture is proportional to the minimum fracture dimension (i.e. fracture diameter for microfractures, brittle layer thickness for layer-bound fractures). Different scaling factors can be defined for fractures striking perpendicular to the minimum and maximum horizontal stress.
- **Dynamic aperture (3):** This method calculates the equilibrium elastic aperture for dilatant fractures subject to a tensile normal stress at the end of the final stage of deformation. An arbitrary user-defined multiplier can also be applied. If the normal stress acting on the fractures is compressive (as will generally be the case for Mode 2 shear fractures), the fractures will have zero aperture. Note that the stress at end of deformation may not be the same as the current stress.
- **Barton-Bandis aperture (4):** This method calculates the aperture of shear fractures subject to a compressive normal stress using the Barton-Bandis formula (Bandis et al. 1983). The calculation is based on the in situ stress state at the end of the final stage of deformation and various parameters related to the fracture morphology and compressibility. The default values (as defined in Bandis et al. 1983) are typical for shear fractures in sedimentary rocks; more information is

given in Appendix 2.4. Note that the stress at end of deformation may not be the same as the current stress.

5.7 Additional outputs

By default, DFM Generator will output the key parameters required for flow modelling in fractured rocks, including fracture intensity, porosity, connectivity and anisotropy. However it is possible to output additional data using the options on the *Outputs* tab (Figure 22). It is also possible to deselect some output data, e.g. porosity (1) or fracture connectivity and anisotropy (2), if it is not required.

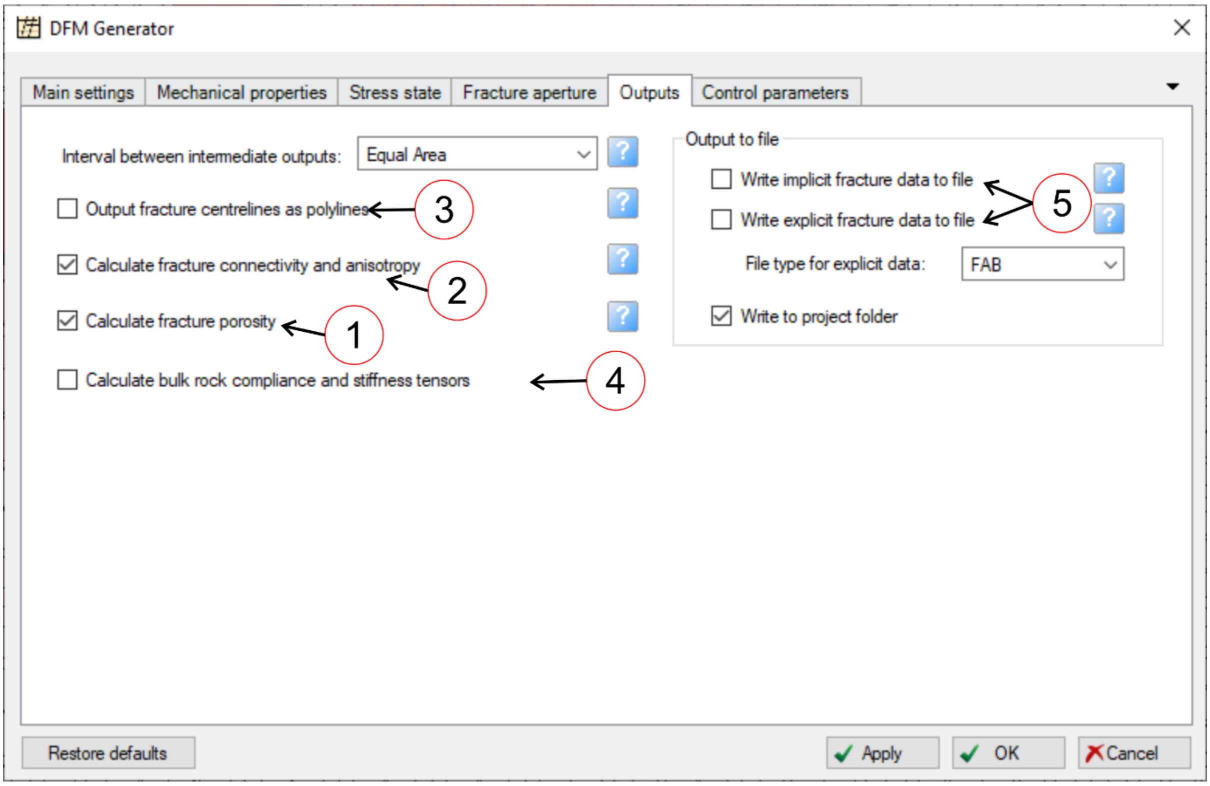

Figure 22: The Outputs tab.

Useful additional outputs include:

- **Fracture centrelines (3):** Selecting this option will generate a set of Petrel polylines representing the horizontal centrelines of the fractures in the DFNs. It is often easier to visualize the geometry and connectivity of the fracture network from the 2D network of centrelines than from the 3D network of fracture segments, especially in Map view. The centrelines are placed in a separate folder on the *Inputs* pane.
- **Bulk rock elastic tensors (4):** This option will output the bulk rock compliance and stiffness tensors (i.e. including both the host rock and the fractures) at the end of the final stage of deformation. Since they are fourth order tensors, the bulk rock compliance and stiffness tensors each comprise 36 components, relating all combinations of the XX, YY, ZZ, XY, YZ and ZX stress and strain components. Each tensor component is output as a grid property and placed in subfolders labelled *Bulk rock compliance tensor* and *Bulk rock stiffness tensor* respectively. The bulk rock compliance and stiffness tensors are only generated for the fully developed fracture network, and not for the intermediate stages.

It is also possible to write the explicit or implicit output data directly to file (5). The implicit data output file will include fracture density, stress and strain data for every timestep in the model (there will typically be many more timesteps than output stages), and also the full cumulative density distribution functions for the final fracture network (which give a breakdown of the fracture size distribution). It can therefore be useful for statistical analysis and for understanding the evolution of the fracture network. However one output file will be generated for each cell stack in the model, so it is recommended that this option is only set when running small models.

5.8 Calculation control parameters and options

The *Control parameters* tab contains advanced parameters and options that control the calculation (Figure 23). For most models these should not be changed, as the default values have been selected to give realistic results. However some of these parameters may be helpful for replicating unconventional fracture geometries:

- **Number of fracture sets (1):** Selecting **Include oblique fractures** on the *Main settings* tab (Section 5.1) will by default generate a model with 6 fracture sets striking at 30° intervals. This is the optimal geometry for modelling strike-slip fractures. However it is possible to increase or decrease the number of fracture sets in the model, in order to match the observed fracture geometry. The fracture sets will always strike at equal intervals, with the first set striking perpendicular to the minimum horizontal strain orientation. If the **Include oblique fractures** option is not selected, this parameter will be deactivated and only 2 fracture sets will be generated.
- **Fracture mode (2):** By default the fracture model will include both vertical Mode 1 dilatant and inclined Mode 2 shear fractures, although typically one of these modes will predominate, depending on which is energetically optimal. However it is possible to force the model to contain only vertical Mode 1 or inclined Mode 2 fractures by specifying a fracture mode.
- **Create relay segments (3):** By default, when two parallel layer-bound fractures are deactivated due to stress shadow interaction, they will be connected by a small relay segment to generate a single long fracture. If this option is deselected, fractures deactivated due to stress shadow interaction will not be connected, leaving a “soft-linked” relay zone (note that this will still count as an R-node when calculating fracture connectivity indices).
- **Layer thickness cutoff (4):** Because the fracture spacing is proportional to the thickness of the brittle layer, then where the brittle layer becomes very thin (e.g. if there is stratigraphic pinch-out), an excessive number of fractures may be generated in the explicit DFN. This will cause the model runtime to increase. To avoid this, a minimum layer thickness cutoff is implemented; the explicit DFN will not be generated in cell stacks thinner than this cutoff. By default this is set to 1m, which is suitable for full-field models; however for small-scale models it may prevent the explicit DFN from being generated in thin layers, and should therefore be adjusted accordingly. Note that the thickness cutoff applies only to the explicit DFN; implicit fracture data will still be generated in all cell stacks, since the calculation time for the implicit model is independent of the number of fractures.

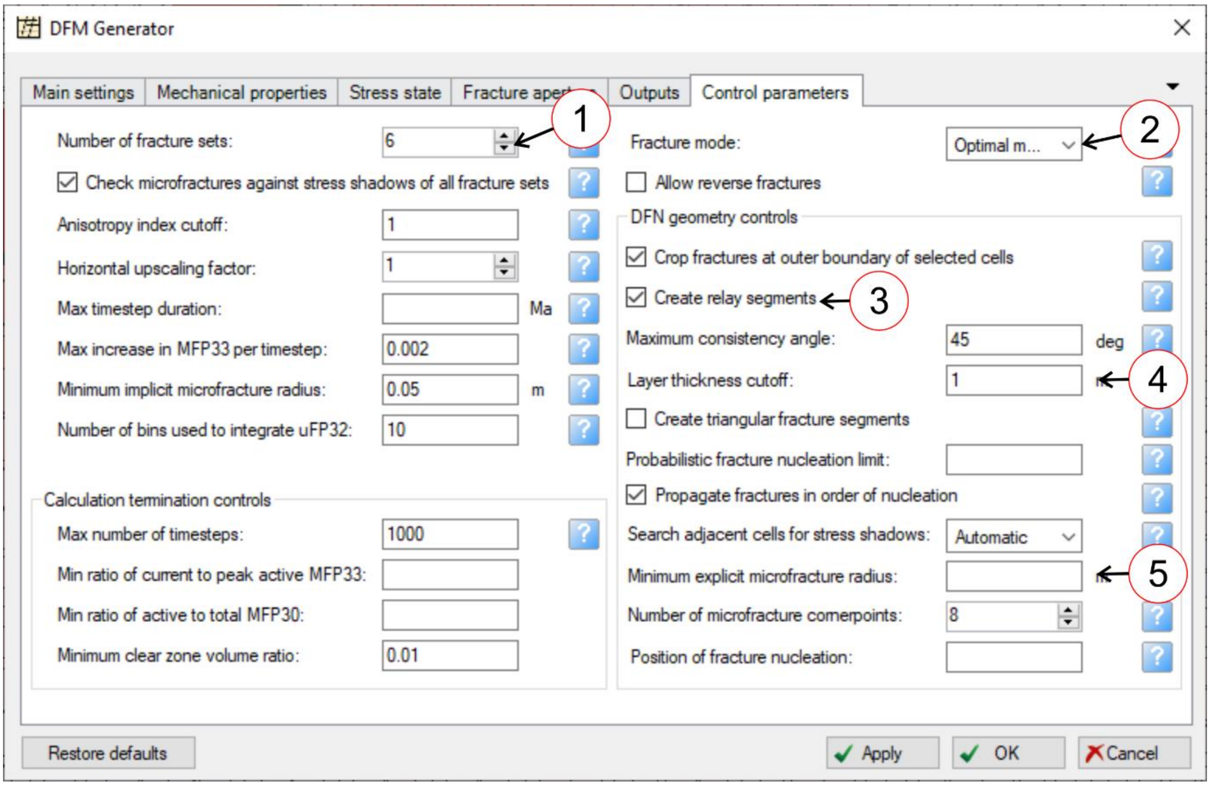

Figure 23: The Control Parameters tab.

- **Minimum radius for microfractures to be included in explicit DFN (5):** By default the explicit DFN contains only layer-bound fractures; microfractures are represented only in the implicit fracture model. However it is possible to include the larger microfractures in the explicit DFN by specifying a minimum microfracture radius. This should be between 0 and half the brittle layer thickness. DFNs that include microfractures can become very large, so it is only recommended to do this only in small models (e.g. near wellbore models) comprising a few cells (see Section 4 and Figure 16). If this input box is blank, no microfractures will be included in the DFN.

Other control settings are described in Appendix 2.6.

6 Using DFM Generator with the Petrel Workflow Editor

The Petrel Workflow Editor provides a means to automate elements of model generation. This is particularly useful when running multiple models for uncertainty analysis, as it can be used to automate the input property assignment and run models overnight.

The Petrel Workflow Editor is available from the *Workflows* pane, normally found at the bottom left of the Petrel window. To launch the *Workflow Editor window*, right-click in the blank space below the existing workflows and select **New workflow** in the context menu (Figure 24):

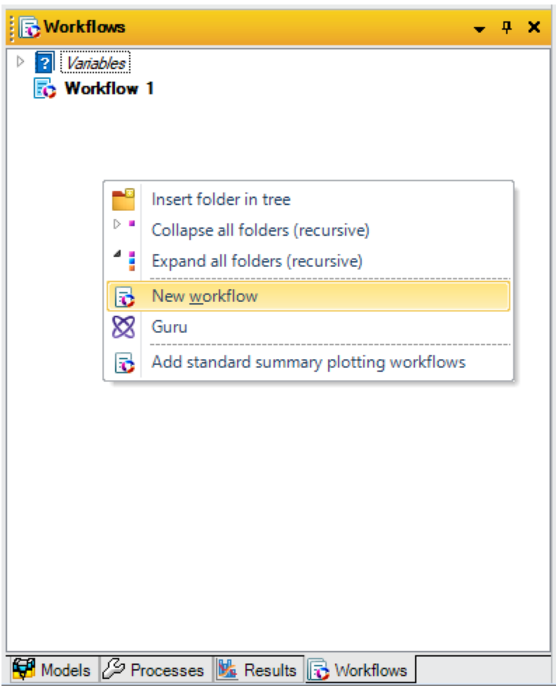

Figure 24: The Workflows pane and context menu showing the New workflow command.

The *Workflow Editor window* is shown in Figure 25. You can assign a more description name to the workflow in the top left corner; then click **OK** to close the *Workflow Editor window*. The new workflow, with the new name, should now appear in the *Workflows* pane.

If you right-click on the new workflow, you can add new workflow variables, which can be named as appropriate and subsequently called in the workflow itself (Figure 26). These can be assigned to represent Petrel objects such as grid properties. Note that variables representing numerical and string expressions are created within the workflow editor itself.

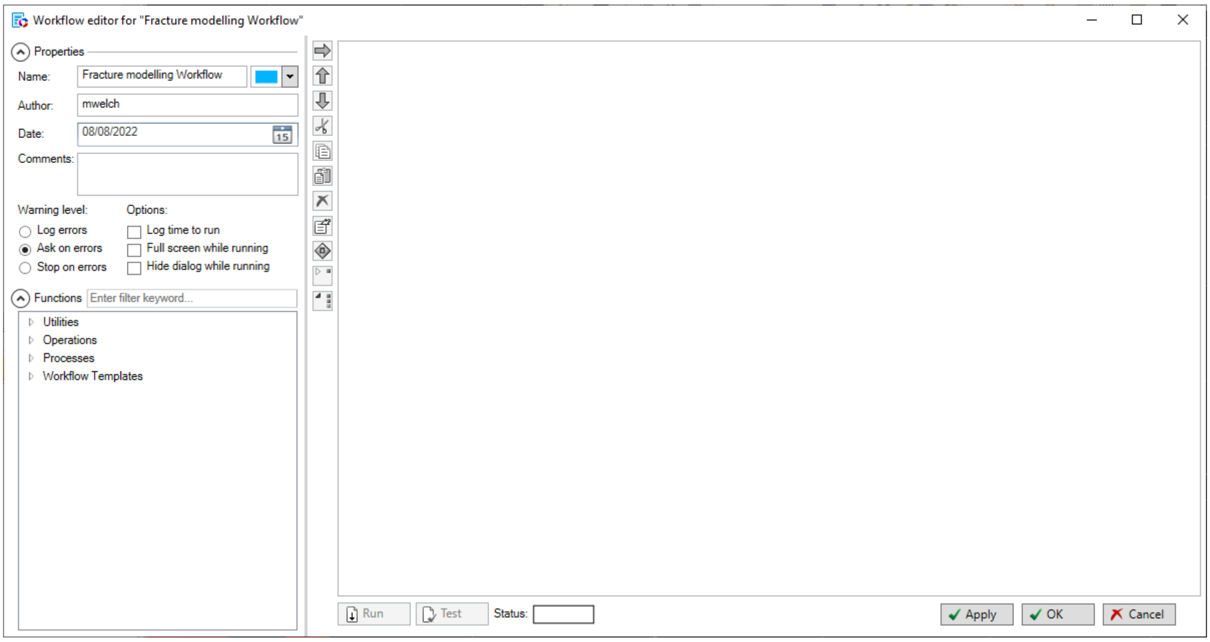

Figure 25: Workflow Editor window.

In this example, we will create workflow variables for the horizontal strain components and the Young's Modulus. Right-click on the new workflow and select **New workflow variable** from the context menu (Figure 26). Double-click on the new variable that has been generated and rename it *EhminAzi*. Repeat this procedure to create workflow variable for the minimum and maximum horizontal strain rates and the Young's Modulus.

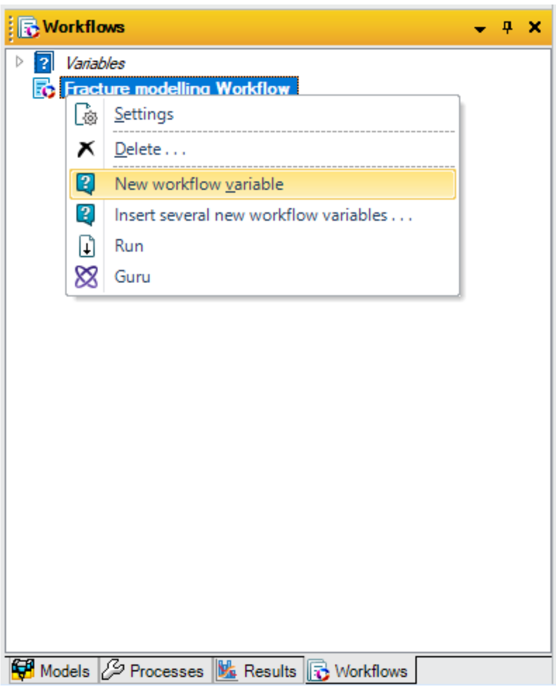

Figure 26: Workflow context menu showing the New workflow variable command.

Now double-click on the new workflow to re-open the *Workflow Editor window*. This window allows you to build the workflow. From the *Functions* pane (on the bottom left in Figure 25) you can access most of the functionality available within Petrel. Open the **Processes → Ocean Plug-ins** folder and select **DFM Generator**, then click the blue right arrow (at the top of the left sidebar to the *Workflow* pane) to add a DFM Generator workstep to the workflow (Figure 27).

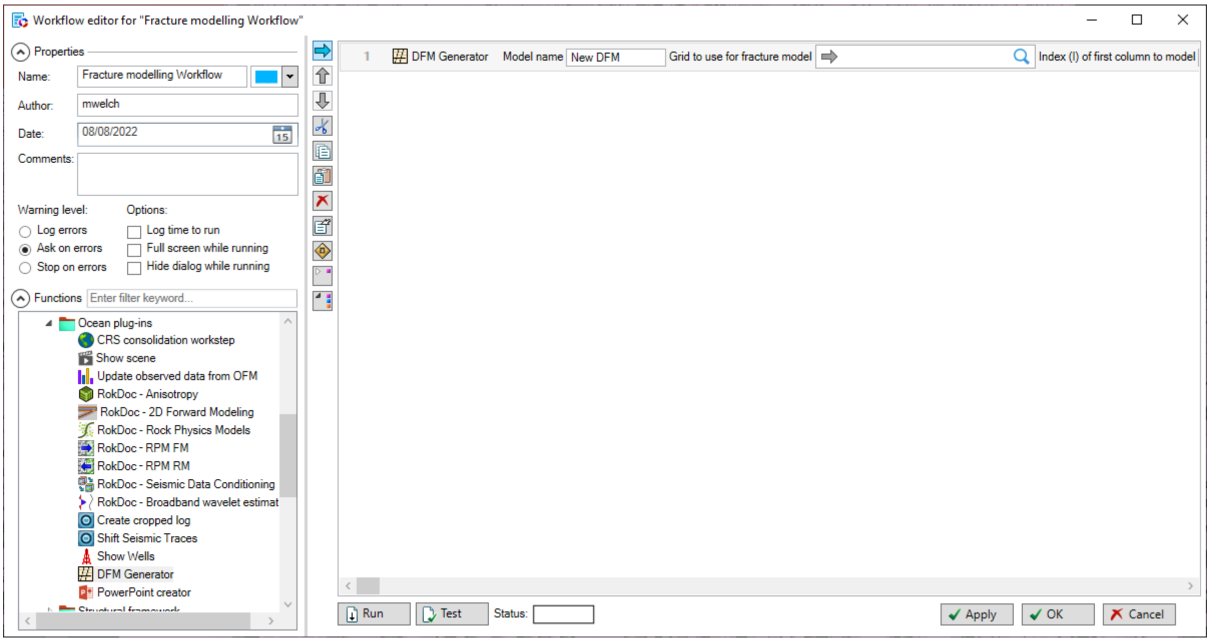

Figure 27: Adding DFM Generator to a workflow.

The DFM Generator input parameters can now be set in the *Workflow* pane (the large pane occupying the right hand side of the *Workflow Editor window*). They will initially be set to the default values, but can be reassigned either to constant values or to variables.

In the example shown in Figure 28, we use the workflow editor to create a series of numeric variables and assign them to selected DFM Generator input parameters. Open the **Utilities** folder in the *Functions* pane and select **Numerical expression**, then click the blue right arrow to add a numeric variable assignment workstep to the workflow. It may be necessary to use the blue up arrow (below the blue right arrow) to move this workstep above the DFM Generator workstep. Define the variable name, beginning with a dollar sign, in the left hand text box and assign it a value in the right hand text box. Repeat this to generate numerical variables for the the start column, the start row, number of columns, the number of rows, and the friction coefficient (lines 1-5 in Figure 28).

We will also create a string expression for the model name, based on the variables we have previously defined. Open the **Utilities** folder in the *Functions* pane and select **String expression**, then click the blue right arrow (top middle) to add a string variable assignment workstep to the workflow. It may again be necessary to use the blue up arrow to move this workstep above the DFM Generator workstep. Set the variable name to *$ModelName* in the left hand text box, and enter the string expression *Drenthe StartCell $StartCol $StartRow Size $NumberOfCols $NumberOfRows* in the right hand text box (line 6 in Figure 28). Note that it is possible to include previously defined variables in the string expression.

Finally we assign grid properties to the workflow variables. Note that, unlike numeric and string variables, workflow variables must be created outside the *Workflow Editor window*, in the *Workflows* pane; however they are assigned values within the workflow. Open the **Utilities** folder in the *Functions* pane and select **Set reference**, then click the blue right arrow to add a workflow variable assignment workstep to the workflow. If necessary, use the blue up arrow to move this workstep above the DFM Generator workstep. Select the new *EhminAzi* variable in the *Workflows* pane, and click the blue arrow in the first input box of the *Set reference* workstep to drop it into the *Workflow Editor window*. Then select the *Azimuth of maximum curvature* grid property on the *Models* pane and click the blue arrow in the second input box of the *Set reference* workstep to assign this property to the *EhminAzi* variable (line 7 in Figure 28). Repeat this procedure for the *EhminRate*, *EhmaxRate* and *YoungsMod* properties.

The DFM Generator workstep should now be on line 11 of the workflow. The DFM Generator input parameters can be accessed by selecting line 11 and scrolling to the right.

We can assign numerical or string variables as input parameters by simply entering the variable name (starting with $) in the appropriate input text box. To assign the workflow variables to grid property inputs, select the required variable and click on the blue arrow in the appropriate grid property presentation box. It is also possible to assign numeric or string values directly to the DFM Generator input parameters, or to drop properties directly into the DFM Generator property input presentation boxes from the *Models* pane. If you have assigned variable and properties using the *DFM Generator dialog* interface, click OK on this interface to close the dialog and store the assigments in the workflow.

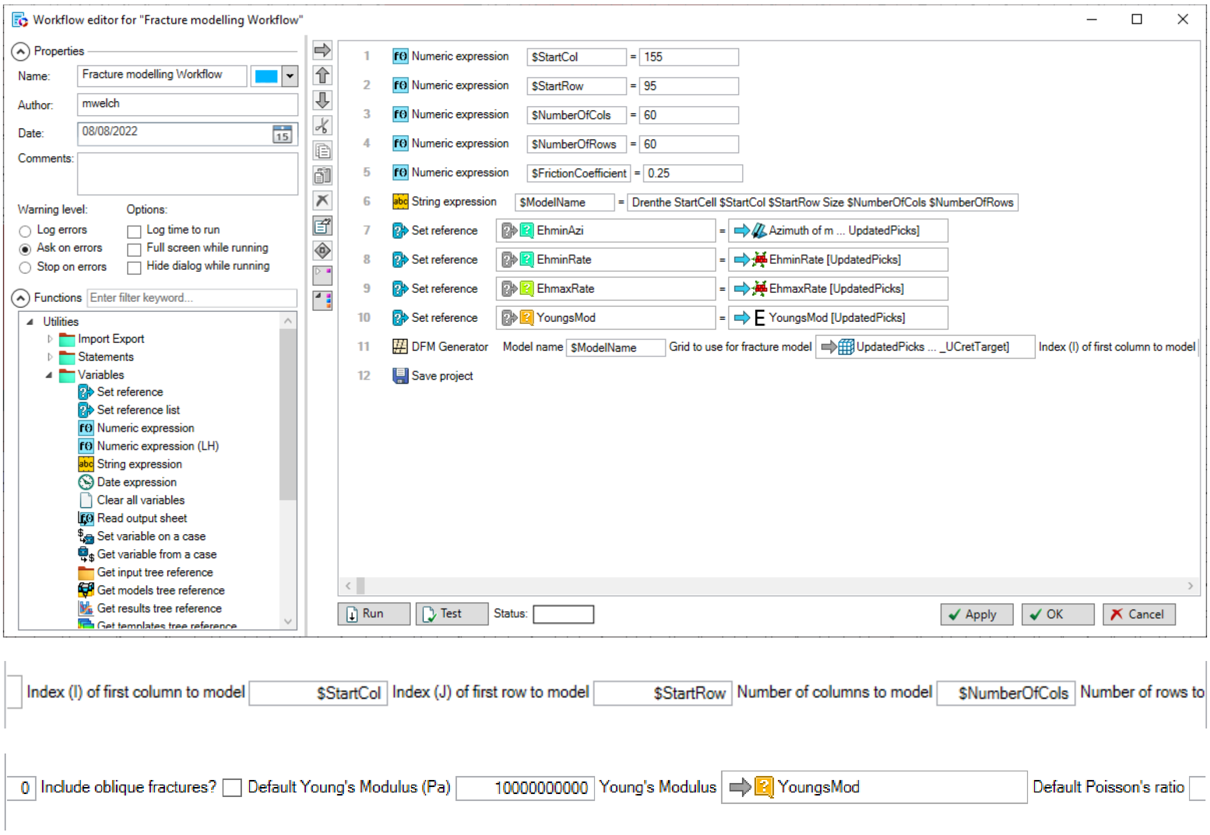

Figure 28: Creating variables and assigning them to the DFM Generator inputs. The close-ups show assignment of numerical and workflow variables.

Finally, we will add a **Save project** command (from the **Utilities – Housekeeping** folder) at the end of the workflow. At this point it is advisable to click the **Apply** button in the *Workflow Editor window*; the input parameters will not be saved in the Petrel project until you have done this.

To run the workflow, click the **Run** button on the *Workflow Editor window*. A syntax error will be thrown if any of the required input parameters are unassigned (e.g. the Grid object). It is therefore advisable to click the **Test** button before running the workflow, especially if the workflow includes multiple large models. This will check the workflow for syntax errors and report accordingly, allowing problems to be fixed quickly.

It is possible to use the loop, flow control and logic statements available in the workflow editor to automatically generate multiple fracture models, changing the value of one or more of the input variables each time to test different scenarios. It is also possible to generate multiple versions of specific grid parameters to use in different iterations. An example of this is illustrated in Figure 29, where nested loops are used to run multiple models, testing the effects of varying Young's Modulus and friction coefficient values.

To better understand how to use these elements of the workflow editor, we recommend consulting the Petrel help pages on general usage of the workflow editor.

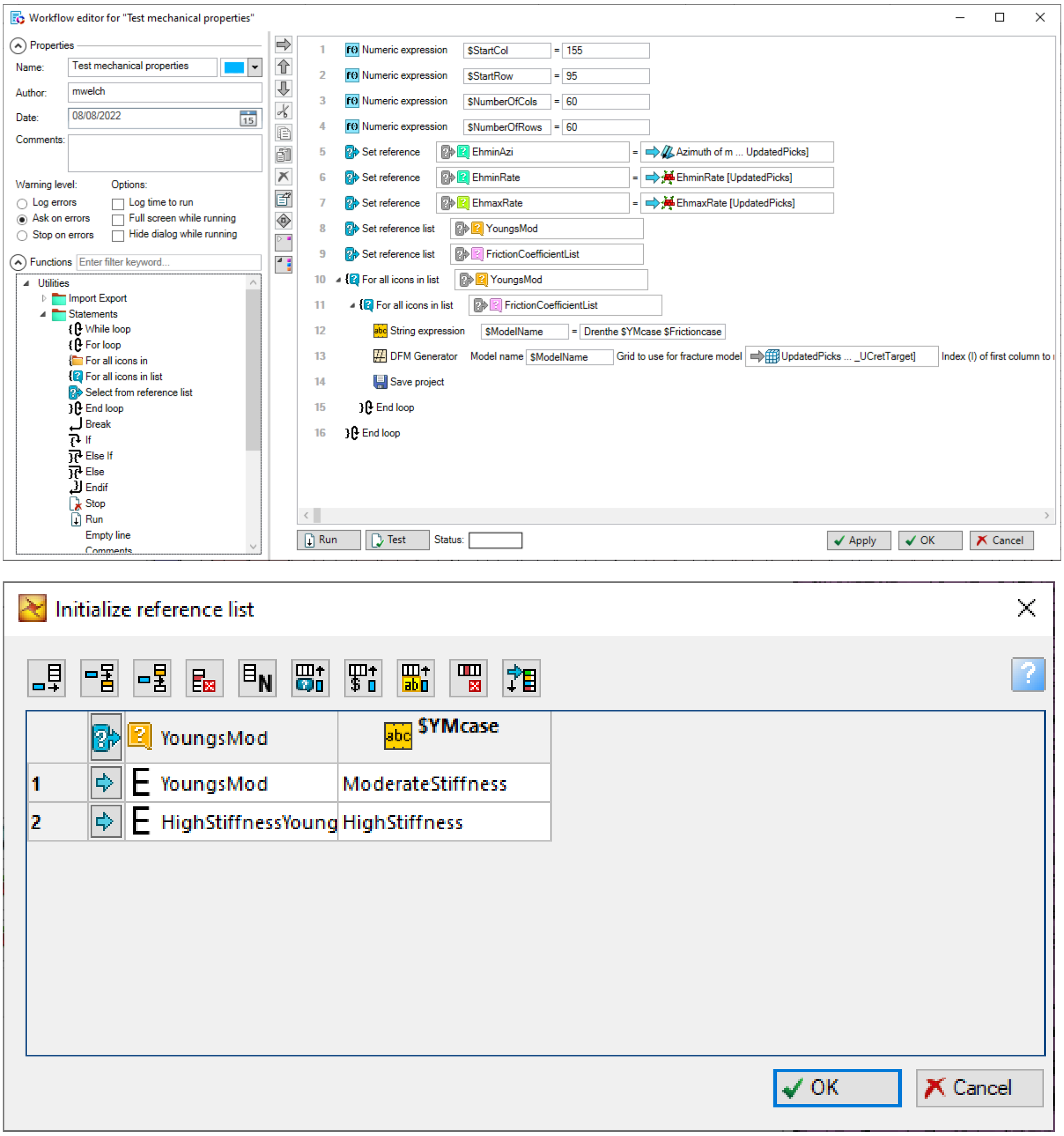

Figure 29: Using a workflow loop to run multiple models, testing the effects of different Youngs Modulus and friction coefficient values (top). Multiple values can be assigned to numerical, string or workflow variables using the ***Set reference list*** *command (bottom).*

7 Further information and contact details

More information about the algorithm used in DFM Generator, as well as analysis of the key geological and geomechanical controls on the development of fracture networks, can be found in

> Welch, M. J., Lüthje, M., & Glad, A. C. 2019. Influence of fracture nucleation and propagation rates on fracture geometry: insights from geomechanical modelling. Petroleum Geoscience, 25(4), 470-489.

and in more detail, in the book

> M. Welch, M. Lüthje and S. Oldfield. Modelling the Evolution of Natural Fracture Networks - Methods for Simulating the Nucleation, Propagation and Interaction of Layer-Bound Fractures. Springer. 2020

The latter also contains examples of the application of DFM Generator to outcrop and subsurface examples of fractured layers. It can be ordered direct from the publisher, at https://www.springer.com/gp/book/9783030524135.

Other articles referenced in the text are:

Atkinson, B.K., 1984. Subcritical crack growth in geological materials. Journal of Geophysical Research 89, B6, 4077-4114.

Bai, T., Pollard, D.D. 2000. Fracture spacing in layered rocks: a new explanation based on the stress transition. Journal of Structural Geology 22, 43-57.

Bai, T., Pollard, D.D., Gao, H. 2000a. Explanation for fracture spacing in layered materials. Nature 403, 753-756.

Bandis, S.C., Lumsden, A.C., Barton, N.R. 1983. Fundamentals of Rock Joint Deformation. International Journal of Rock Mechanics, Mining Science & Geomechanics Abstracts, 20, 249-268.

Chang, C., Zoback, M.D., Khaksar, A. 2006. Empirical relations between rock strength and physical properties in sedimentary rocks. Journal of Petroleum Science and Engineering 51, 223–237.

Miller, T.W. 1995. New insights on natural hydraulic fractures induced by abnormally high pore pressures. American Association of Petroleum Geologists Bulletin 79, 1005–1018.

Swanson, P.L. 1984. Subcritical crack growth and other time- and environment-dependent behaviour in crustal rocks. Journal of Geophysical Research, 89, 4137–4152.

DFM Generator was originally developed with funding from the Danish Offshore Technology Centre (DOTC) under the Advanced Water Flooding programme.

Please note therefore that DFM Generator comes with no warranty and DOTC and the authors accept no liability for any consequence arising from its use. There is also no formal support or service level agreement for the software. However if you encounter any problems, or have any comments or suggestions, please contact Michael Welch (mwelch@dtu.dk) and we will try to help you. Please also report any bugs that you encounter or requests for functionality enhancements in the same way.

Appendix 1 Generating strain data from curvature

The horizontal strain data required by DFM Generator can be obtained by many different methods. The most appropriate method will depend on the mechanism responsible for generating the fractures: for example a regional tectonic fracture set could be modelled using a uniform extensional strain with a typical tectonic strain rate (e.g. 0.01-0.001/ma), fractures caused by folding or diapirism could be modelled using a structural restoration or backstripping algorithm to calculate strain, while fault-related fractures could be modelled by using the Petrel Tectonic Model module to calculate the local elastic strain around the faults.

For a first-pass fracture model, however, estimates of horizontal strain can be generated quickly and easily from the horizon curvature, using existing Petrel functionality (this requires a license for the Petrel structural analysis package). It should however be noted that curvature gives only an approximation for strain, and a more mechanically-based approach should be used to determine strain for more detailed studies.

The procedure for generating strain data from curvature is as follows:

1. Select a horizon at or near the top of the brittle layer. Convert this to a *Regular Surface*: double-click on the horizon in the *Models* pane to open the *Settings* dialog, go to the *Operations* tab and click on **Make Surface**; the surface should then appear at the bottom of the Input pane.
2. It may be necessary to smooth the surface, to remove curvature resulting from noise or very localised structures. Double-click on the surface in the *Input* pane to open the *Settings* dialog, go to the *Structural Analysis* tab and select **Surface Smoothing** from the **Data Clean-up and Processing** folder. Good results can generally be obtained with I and J region half-widths of 2, the Combo Smooth averaging method, Smooth Edges, and 2 passes (Figure 30). Click on **Run** to carry out the smoothing operation. This will generate a new surface in the same folder as the original.
3. Double-click on the (smoothed) surface to open the *Settings* dialog, go to the *Operations* tab and open the **Curvature Operations** folder to find the algorithms for calculating surface curvature. The three required horizontal strain components can be generated using the **Min Curvature**, **Max Curvature** and **Azimuth of Max Curvature** algorithms. Select each icon in turn and then click on **Run**; the outputs should appear as three surface attributes, as shown on Figure 31.
4. The surface attributes can now be converted to grid properties using the *Geometric Modelling* module on the *Structural Modelling* tab. Open this module and select the **Constant or surface in segments or zones** method. Generally it is easiest to write the curvature data to all cells in the grid (set the **Constant or Surface** to be **Same for all Zones** and **Same for All Segments**); however it is also possible to write the data only to specific stratigraphic zones. Click on the check box in the **All Segments, All Zones** cell of the table to activate the blue arrow button, then select the **Min Curvature** attribute in the Input pane and click on the blue arrow button to drop this into the table (Figure 32). Set the *Property Template* to *General*, and then click on **Apply**. This should create a new grid property on the *Models* pane labelled *Min curvature*. Repeat this procedure to create grid properties for *Max curvature* and for *Azimuth of Max Curvature* (use the *Dip Azimuth Property Template* for the *Azimuth of Max Curvature*).
5. The maximum curvature is used as a proxy for the maximum extensional strain. It must therefore be inverted, to convert positive curvature values into negative strain values. It is usually also necessary to apply a multiplier to give geologically realistic strain rates; these are typically -0.001 to -0.1/ma, requiring a multiplier of c.10-100. This can be done using the *Property Calculator*. Right click on the *Max Curvature* property in the *Models* pane and select **Calculate**. Create a new property *EhminRate* from the *Max Curvature* property by typing `EhminRate=-10*` in the Calculator command line, selecting the *Max Curvature* property in the Models pane, clicking on

the blue arrow button in the calculator to drop it into the command line, selecting **Strain** from the *Property Template* drop-down menu, and pressing **Return** to run the calculation (Figure 33). Repeat the procedure to create a new *EhmaxRate* property from the *Min Curvature* property. Note that the maximum curvature is used to calculate the minimum horizontal strain rate and vice versa; this is because the minimum horizontal strain represents the most negative (i.e. most extensional) horizontal strain.

6. The *EhminRate*, *EhmaxRate* and *Azimuth of Max Curvature* properties can now be assigned to the Minimum horizontal strain rate, Maximum horizontal strain rate and Minimum horizontal strain azimuth input parameters on the DFM Generator *Main settings* tab.

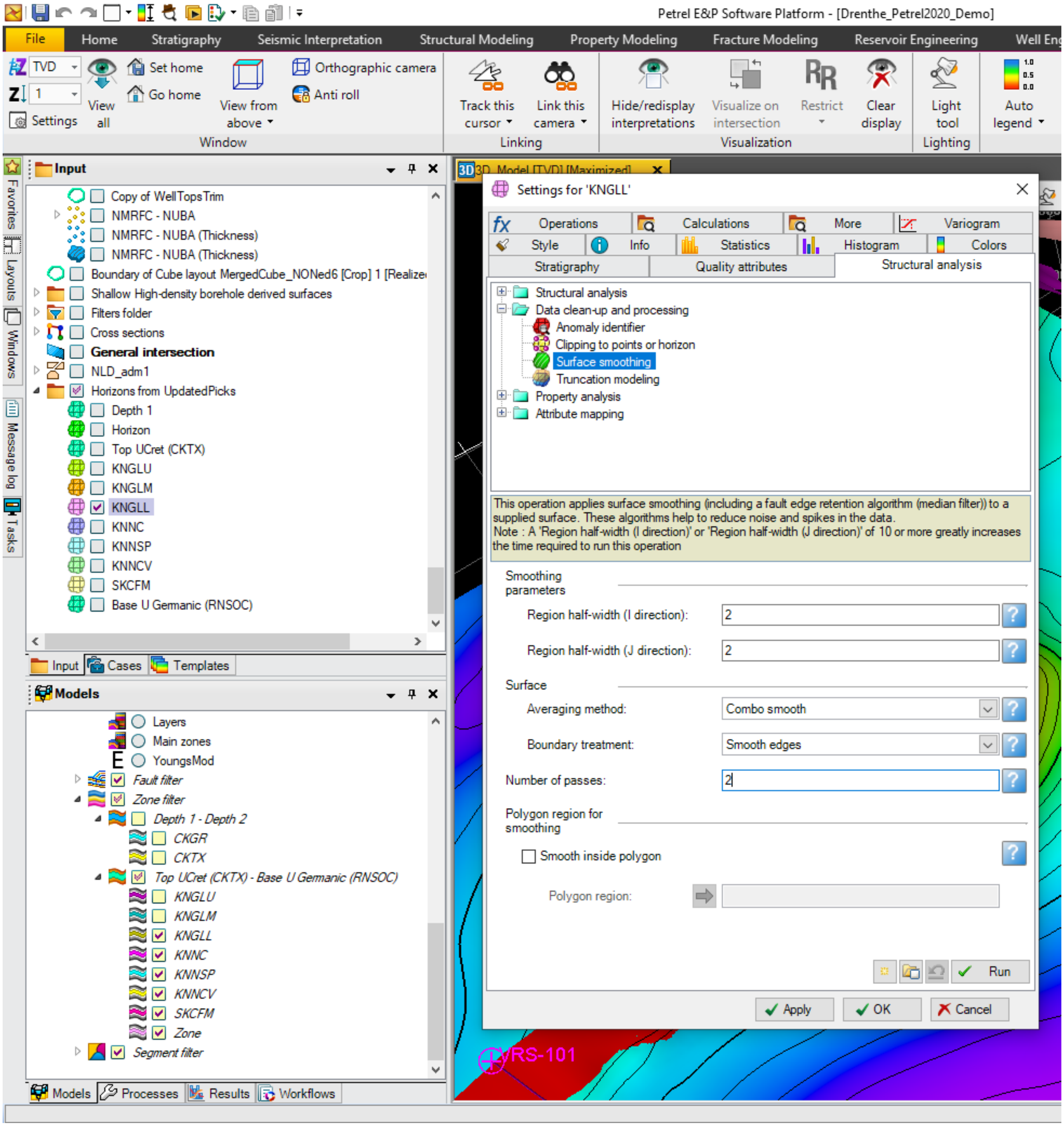

Figure 30: Using the Surface Smoothing algorithm to smooth a surface.

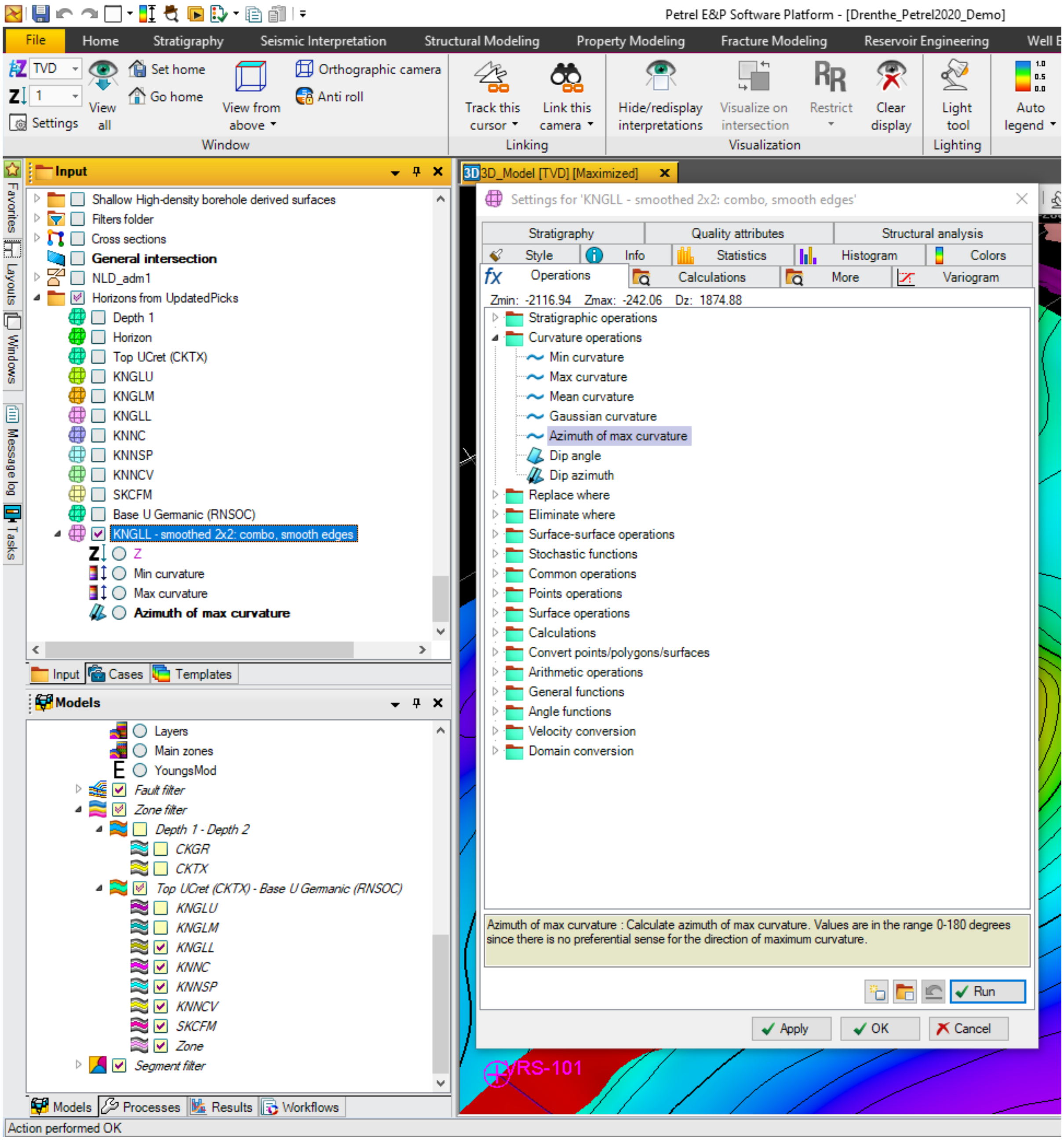

Figure 31: Calculating curvature as a surface attribute using the Curvature Operations algorithms.

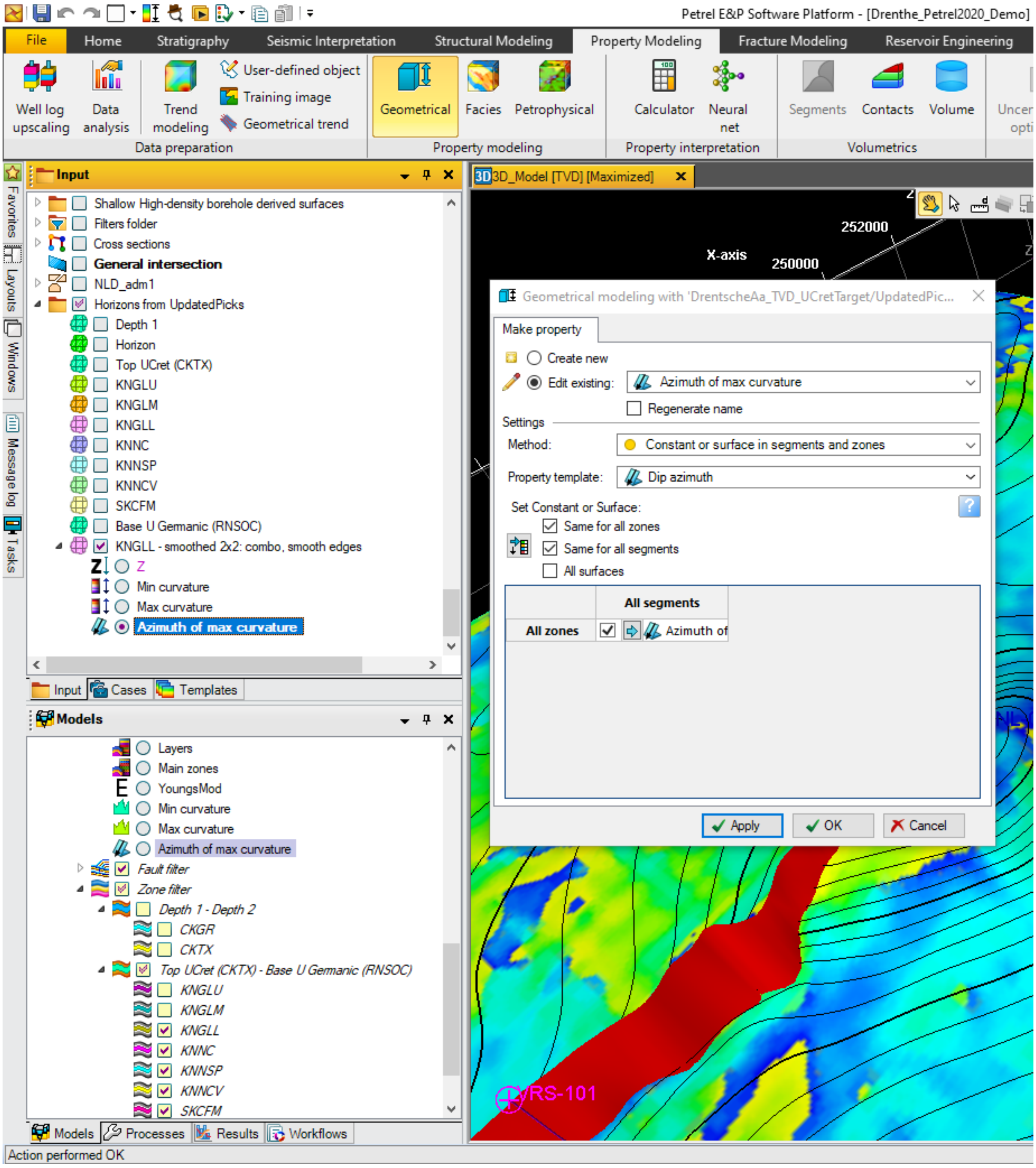

Figure 32: Converting the curvature surface attributes to grid properties using the Geometric Modelling module.

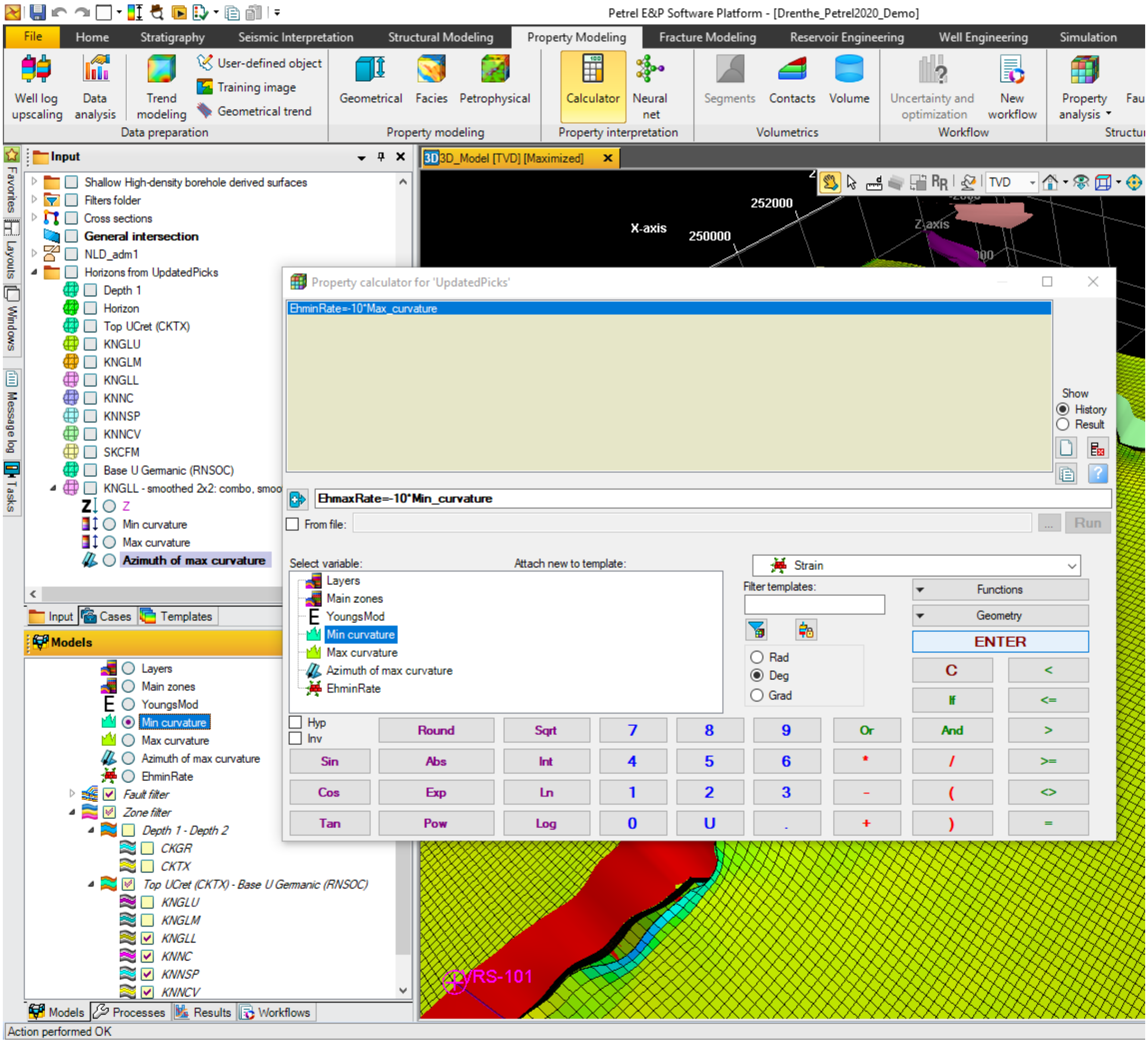

Figure 33: Using the Property Calculator to invert the curvature data to generate strain rate properties.

Appendix 2 Complete list of input parameters and options

This appendix describes the effect of all input parameters and options not described in Section 3 or Section 5. The default values of these settings have been chosen to give reasonable results for a typical fractured carbonate or tight sandstone layer, and it is recommended they are not changed unless required to troubleshoot problems with specific models.

Appendix 2.1 Main settings

All parameters and options on the *Main settings* tab are described in Section 3 or Section 5.

Appendix 2.2 Mechanical properties

- **Young's Modulus:** specifies the Young's Modulus for intact rock.
- **Poissons ratio:** specifies Poisson's ratio for intact rock.
- **Host rock porosity:** specifies the host rock porosity at the start of deformation. This is used to calculate the internal strain and resulting stress generated by fluid pressure loads (either geological loads or dynamic loads), using the algorithm of Miller (1995).
- **Biot coefficient:** the Biot coefficient relates fluid pressure to effective stress, and is used to calculate the internal strain and resulting stress generated by fluid pressure loads (either geological loads or dynamic loads), using the algorithm of Miller (1995).
- **Thermal expansion coefficient:** the thermal expansion coefficient of the host rock is used to calculate the internal strain and resulting stress generated by thermal loads, or by cooling due to uplift.
- **Friction coefficient:** specifies the coefficient of friction on newly developed fractures. The default value for friction should be supplied as a friction coefficient, not a friction angle (the friction coefficient is the tangent of the friction angle). However friction angle can be supplied as a grid property, as long as it uses the Friction Angle template. Grid properties supplied with any other templates will be assumed to represent the friction coefficient.
- **Crack surface energy:** specifies the crack surface energy required to break apart the rock as the fractures propagate. This is equivalent to the critical energy release rate.
- **Rock strain relaxation:** Use this parameter to apply viscoelastic strain relaxation to the rock matrix. Strain relaxation controls the rate of stress accumulation: with no strain relaxation, the horizontal stress will increase at a constant rate proportional to the horizontal strain rate; with strain relaxation, the horizontal stress will increase initially but then level off at a magnitude proportional to the ratio of strain rate to strain relaxation time constant. Rock strain relaxation is specified as a time constant controlling the rate of viscoelastic strain relaxation in the rock matrix: elastic strain will reduce to 1/e of its initial value during this time. Leave this input box blank to turn off viscoelastic strain relaxation. Note: a low strain relaxation time constant corresponds to rapid strain relaxation, while a high strain relaxation time constant corresponds to slow strain relaxation.
- **Fracture strain relaxation:** Use this parameter to apply viscoelastic strain relaxation around the fractures only. Fracture strain relaxation is specified as a time constant controlling the rate of viscoelastic strain relaxation around the fractures only: elastic strain around the fractures will reduce to 1/e of its initial value during this time, but elastic strain in the rock matrix will not be affected. To apply viscoelastic strain relaxation around the fractures only, you must set **Rock strain relaxation** to 0 (or leave the input box blank) and **Fracture strain relaxation** to a value >0.
- **Initial microfracture density** and **Initial microfracture size distribution:** The initial seed microfractures are assumed to follow a fractal distribution. The **Initial microfracture density** specifies the density of initial seed microfractures, and **Initial microfracture size distribution**

specifies the size distribution of initial microfractures. Increasing the **Initial microfracture size distribution** gives a higher ratio of small:large initial microfractures

- **Subcritical fracture propagation index:** This controls the fracture propagation rate at different stress states (see Atkinson 1984). If the subcritical fracture propagation index is <5, fracture growth will be dominated by slow subcritical propagation; if it is >15, fracture growth will be dominated by rapid critical propagation. Experimental studies have shown that subcritical indices for rocks are typically 10 or greater (Swanson 1984).
- **Critical fracture propagation rate:** Specifies the critical fracture propagation rate. This is typically similar to the sonic velocity of the rock.
- **Average mechanical properties across all cells:** If the specified brittle layer contains multiple grid (K) layers and mechanical properties are specified as grid properties, select this option to calculate the average value for each property within each cell stack and apply it to the entire stack. If this option is not selected, the value of the top cell in the stack will be applied to the entire stack.

Appendix 2.3 Stress state

The effective vertical stress at the time of deformation, σ_v', is calculated from the depth at the time of deformation z, the mean overlying sediment density ρ_b, the fluid density ρ_f, the fluid overpressure P_o, the Biot coefficient α, and the gravitational constant g, using the formula $\sigma_v'=(\rho_b gz)-\alpha(\rho_f gz+P_o)$. The initial effective horizontal stress σ_{h0}' is determined from the effective vertical stress and the initial stress relaxation (see Section 5.5):

- **Depth at time of deformation:** Depth of burial at the time of deformation, specified either as a uniform default value or as a grid property (in project units, positive downwards). If both input boxes are blank, the current depth of the brittle layer will be used to calculate vertical effective stress.
- **Mean density of overlying sediments:** Mean bulk density of overlying sediments (including pore fluids), used to calculate lithostatic stress.
- **Fluid density:** Mean pore fluid density, used to calculate hydrostatic fluid pressure.
- **Initial fluid overpressure:** Fluid overpressure; this represents the excess fluid pressure, above normal hydrostatic pressure (given by $\rho_f gz$).
- **Geothermal gradient:** This is used to calculate the rate of cooling due to uplift and erosion.
- **Average stress/strain state across all cells:** If the specified brittle layer contains multiple grid (K) layers and the horizontal strain parameters are specified as grid properties, select this option to calculate the average value for each property within each cell stack and apply it to the entire stack. If this option is not selected, the value of the top cell in the stack will be applied to the entire stack. It is recommended to not select this option, as averaging the strain orientation can lead to anomalous results.

Appendix 2.4 Fracture aperture

Method to determine fracture aperture: Use this to select the method used to calculate fracture aperture, which is in turn used to calculate fracture porosity and permeability. Four methods are available: *Uniform*, *SizeDependent*, *Dynamic*, or *BartonBandis*. Depending on the selected method, the following parameters are used to calculate fracture aperture:

- **Uniform** fracture aperture: All fractures are assigned an arbitrary user-specified aperture. Different apertures can be specified for fractures with different orientations.
    - **Aperture for fractures striking perpendicular to Shmin:** Specifies the aperture for fractures striking perpendicular to the minimum horizontal strain. This applies to both vertical and inclined fractures.

  - **Aperture for fractures striking perpendicular to Shmax:** Specifies the aperture for fractures striking perpendicular to the maximum horizontal strain. This applies to both vertical and inclined fractures.

  If there are more than 2 fracture sets, the apertures of the oblique fractures are determined by trigonometric interpolation between the two specified values.

- **Size Dependent** fracture aperture: Fracture aperture is proportional to the minimum fracture dimension (i.e. fracture diameter for microfractures, brittle layer thickness for layer-bound fractures). Different scaling factors can be defined for fractures with different orientations.
  - **Aperture for fractures striking perpendicular to Shmin:** Specifies the size-dependent aperture multiplier for fractures striking perpendicular to the minimum horizontal strain. This applies to both vertical and inclined fractures.
  - **Aperture for fractures striking perpendicular to Shmax:** Specifies the size-dependent aperture multiplier for fractures striking perpendicular to the maximum horizontal strain. This applies to both vertical and inclined fractures.

  If there are more than 2 fracture sets, the size-dependent aperture multipliers for the oblique fractures are determined by trigonometric interpolation between the two specified values.

- **Dynamic** fracture aperture: Calculates the equilibrium elastic aperture for dilatant fractures subject to a tensile normal stress at the end of the final stage of deformation. If the normal stress acting on the fractures is compressive (as will generally be the case for Mode 2 shear fractures), the fractures will have zero aperture. Note that the stress at end of deformation may not be the same as the current stress.
  - **Multiplier for dynamic aperture:** Arbitrary multiplier applied to the calculated equilibrium elastic aperture for dilatant fractures. The same multiplier is applied to all fractures.
- **Barton Bandis** model for fracture aperture: Calculates the aperture for shear fractures subject to a compressive normal stress using the Barton-Bandis formula (Bandis et al. 1983). The calculation is based on the in situ stress state at the end of the final stage of deformation and various parameters related to the fracture morphology and compressibility, defined in the paper:
  - **Joint Roughness Coefficient:** Represents the morphology of the fracture surface: a higher value reflects a rougher surface, a lower value reflects a smoother surface.
  - **UCS Ratio:** Ratio of unconfined compressive strength of unfractured rock to fractured rock.
  - **Initial normal stress:** Reference value for the normal strength on fracture, at which the fracture normal stiffness was measured.
  - **Fracture normal stiffness:** Stiffness normal to the fracture, at initial normal stress. Units are [Pressure]x[Length].
  - **Maximum closure:** Maximum fracture closure, at very high compressive stress.

  The default values, which are taken from Bandis et al. 1983, are typical for shear fractures in sedimentary rocks.

Fracture aperture A and permeability k_f are also assigned as attributes to individual fractures in the explicit DFN. The fracture permeability attribute is defined as $k_f=A^2/12$, which is derived by equating the Darcy and the Pouseille parallel plate flow laws.

Appendix 2.5 Outputs

- **Output intermediate results by time:** This controls the interval between the intermediate stage fracture models specified on the *Main settings* tab (see Section 3.1). There are three options:
    - **Equal time** will output the specified number of intermediate fracture models at equal intervals of time.
    - **Equal area** (the default) will output the specified number of intermediate fracture models at approximately equal increments in the total fracture area.
    - **Deformation episodes** will output an intermediate fracture model at the end of each specified deformation episode, or at the end of each timestep in the simulation case if a dynamic load is specified. If this option is selected, the number of fracture models output will equal the number of deformation episodes, and the number of intermediate fracture models specified on the *Main settings* tab will be ignored.
- **Calculate fracture porosity:** This option is selected by default. If it is deselected, then fracture porosity data will not be calculated, and neither will total fracture density for the entire fracture network (fracture density data will still be calculated for the individual fracture sets). The *Fracture porosity* subfolder will not be generated for any of the implicit fracture models.
- **Calculate fracture connectivity and anisotropy:** This option is selected by default. If it is deselected, then fracture connectivity and fracture anisotropy indices, as time to fracture saturation, will not be calculated either for individual fracture sets or for the entire fracture network. The *Fracture anisotropy* subfolder will not be generated for any of the implicit fracture models.

In addition to generating output data in Petrel, it is also possible to write the explicit or implicit output directly to file:

- **Write implicit data to file:** Select this option to write implicit fracture data to file. The implicit data output file includes fracture density, stress and strain data for every timestep in the model (there will typically be many more timesteps than output stages), and also the full cumulative density distribution functions for the final fracture network (which give a breakdown of the fracture size distribution). It can therefore be useful for statistical analysis and for understanding the evolution of the fracture network. Excel spreadsheets to assist with this analysis can be downloaded from https://jointflow.eu/download.

  One output file will be generated for each cell stack in the model, so it is recommended that this option is only set when running small models.

- **Write explicit data to file:** Select this option to write the explicit DFN data to file. Two files will be generated for each intermediate stage, and two more for the final fracture network: one containing the microfractures and one containing the layer-bound fractures. These can be output in ASCII or FAB format, as determined by the **File type for explicit data** drop-down. ASCII files can be read by the Excel data analysis spreadsheets which can be downloaded from https://jointflow.eu/download, while FAB files can be loaded directly into Petrel and other geomodelling packages.
- **Write to project folder:** If this box is checked, all output files will be written to the Petrel project folder. If unchecked, a new folder $home\DFMFolder will be created and all output files will be written to this. Make sure you have permission to write files to the selected output folder.

Appendix 2.6 Control parameters

- Selecting **Include oblique fractures** on the *Main settings* tab (Section 5.1) will by default generate a model with 6 fracture sets striking at 30° intervals. This is the optimal geometry for modelling strike-slip fractures. However the **Number of fracture sets** parameter can be used to increase or decrease the number of fracture sets in the model, in order to match the observed fracture geometry. The fracture sets will always strike at equal intervals, with the first set striking perpendicular to the minimum horizontal strain orientation. If the **Include oblique fractures** option is not selected, this parameter will be deactivated and only 2 fracture sets will be generated.
- **Check microfractures against stress shadows of all fracture sets:** This controls whether microfractures will be affected by stress shadows of all layer-bound fractures, or only layer-bound fractures in the same set. If selected, microfractures may be deactivated if they lie within the stress shadows of oblique as well as parallel layer-bound fractures. The width of the stress shadow around a specific layer-bound fracture, as seen by a specific microfracture, depends on the relative orientations of the layer-bound fracture, the microfracture, and the in situ stress. It must therefore be calculated separately for every combination of fracture sets. It is recommended that this option is selected if there are more than 2 fracture sets, as it gives a more realistic placement of the layer-bound fracture seeds. However if the **Include oblique fractures** on the *Main settings* tab is not selected, this option will be deactivated as there is no stress shadow interaction between perpendicular dip slip or dilatant fractures.
- **Anisotropy index cutoff:** Specifies the anisotropy cutoff value which determines whether the isotropic or anisotropic method is used to calculate cross-fracture set stress shadow and exclusion zone volumes (this is only relevant when the option to Check microfractures against stress shadows of all fracture sets is selected). Both methods are approximations: the isotropic method takes account of overlapping fractures, while the anisotropic method takes account of the influence of the primary fracture set on the distribution of secondary sets. By default this is set to 1, so the isotropic method will always be used, since this tends to give more accurate results.
- **AllowReverseFractures:** Controls whether reverse fractures are allowed in the fracture network. If this option is not selected, fracture sets with a reverse displacement vector will not be allowed to accumulate displacement or grow.
- **Horizontal upscaling factor:** Use this control to amalgamate Petrel grid cells laterally as well as vertically, and thus reduce the number of cell stacks. In addition to reducing model runtime, this will reduce the height:width ratio of the cell stacks; for optimal results this should be less than 1/2. However faults will not be honoured in horizontally upscaled grids (i.e. fractures may cross the fault planes).
- **Max timestep duration:** Specifies a maximum duration for individual timesteps. If this input box is blank, no maximum timestep duration will be applied, so that the timestep duration will be controlled only by the **Max increase in MFP33 per timestep**. This is the recommended setting.
- **Max increase in MFP33 per timestep:** Specifies the maximum increase in layer-bound fracture volumetric ratio allowed in each timestep. The volumetric ratio (also known as the P33 fracture density) is proportional to the strain accommodated by the fractures. This parameter controls the timestep duration; if it is increased, the calculation will run faster, with fewer but longer timesteps.
- **Minimum implicit microfracture radius:** Specifies the minimum radius for microfractures to be included in implicit fracture density and porosity calculations. If this is set to zero (i.e. all microfractures are included) then it will not be possible to calculate the volumetric microfracture density (μFP30) as this will be infinite. If this input box is blank, the maximum radius of the smallest bin will be used (i.e. the smallest bin will be excluded from the microfracture population).

- **Number of bins used to integrate μFP32:** The total microfracture area (μFP32) and volume (μFP33) cannot be derived analytically, so must be calculated numerically, by subdividing the population into equally spaced "bins" across the range of possible microfracture radii (from zero to half the layer thickness). This parameter specifies the number of bins used for the numerical integration, which in turn controls the accuracy of numerical calculation of microfracture density, porosity and stress shadow volume. Increase the number of bins to increase accuracy of the numerical integration at expense of runtime.
- **Calculation termination controls:** The calculation is set to stop automatically when fractures stop growing. This can be defined in one of three ways:
    - When the total volumetric ratio of active (propagating) layer-bound fractures (aMFP33) drops below a proportion of the peak historic value, specified by the **Min ratio of current to peak active MFP33** parameter.
    - When the total volumetric density of active (propagating) layer-bound fractures (aMFP30) drops below a proportion of the total (propagating and non-propagating) volumetric density (MFP30), specified by the **Min ratio of active tot al MFP30** parameter.
    - When the total clear zone volume (the volume in which fractures can nucleate without falling within or overlapping a stress shadow) drops below a proportion of the total volume, specified by the **Minimum clear zone volume** parameter.

  Increase these cutoffs to reduce the sensitivity and stop the calculation earlier. This prevents a long calculation tail - i.e. late timesteps where fractures have stopped growing so they have no impact on fracture populations, but increase runtime. To stop calculation while fractures are still growing, set the deformation episode duration(s) on the *Deformation episode* dialog to define a termination point based on geological time, or use the **Max number of timesteps** parameter on this tab to define a termination point based on model runtime. If any of these input boxes are blank, the relevant termination criterion will not be applied. It is generally recommended to use the total clear zone volume as the primary termination criterion, but set a maximum number of timesteps (c.1000) to prevent excessively long runtimes in case the primary criterion fails.

- **DFN geometry controls:** these parameters and options apply only to the explicit DFNs:
    - **Crop fractures at outer boundary:** Select this option to crop the fractures at the outer boundary of the fracture model, as defined on the *Main settings* tab. If deselected, fractures will continue to propagate beyond the model boundary.
    - **Create relay segments:** By default, when two parallel layer-bound fractures are deactivated due to stress shadow interaction, they will be connected by a small relay segment to generate a single long fracture. If this option is deselected, fractures deactivated due to stress shadow interaction will not be connected, leaving a "soft-linked" relay zone (note that this will still count as an R-node when calculating fracture connectivity indices).
    - **Maximum consistency angle:** Specifies the maximum variation in fracture propagation azimuth allowed across a gridblock boundary. If the orientation of the fracture set varies by more than this value across a gridblock boundary, the algorithm will seek a better matching set (although if it does not find one, it will revert to the original set).
    - **Layer thickness cutoff:** Because the fracture spacing is proportional to the thickness of the brittle layer, then where the brittle layer becomes very thin (e.g. if there is stratigraphic pinch-out), an excessive number of fractures may be generated in the explicit DFN. This will cause the model runtime to increase. To avoid this, a minimum layer thickness cutoff is implemented; the explicit DFN will not be generated in cell stacks

thinner than this cutoff. By default this is set to 1m, which is suitable for full-field models; however for small-scale models it may prevent the explicit DFN from being generated in thin layers, and should therefore be adjusted accordingly. Note that implicit fracture data will still be generated in all cell stacks.

- **Create triangular fracture segments:** Select this option to create triangular instead of quadrilateral segments for the layer-bound fractures. Microfractures will comprise a series of coplanar triangles with vertices at the centre, rather than a single polygon. This may facilitate meshing for some applications.
- **Probabilistic fracture nucleation limit:** Specifies the minimum limit for fracture nucleation to be controlled probabilistically. By default, explicit fractures nucleate deterministically at regular intervals determined by the implicit microfracture density and growth rate. However if the cell widths are small relative to the fracture spacing, so that the average number of fractures per cell stack is less than 1, then the fracture nucleation interval will be greater than the model duration and no fractures will ever nucleate. Setting this parameter allows the timing of fracture nucleation to be determined probabilistically, if the number of fractures nucleating per timestep is less than the specified value. Probabilistic nucleation allows some explicit fractures to nucleate even when the cells are small. Set this parameter to zero to disable probabilistic fracture nucleation. Leave the input box blank to enable automatic probabilistic fracture nucleation: probabilistic fracture nucleation will be activated whenever the option to search adjacent cell stacks for stress shadows is active; if **Search adjacent cell stacks for stress shadows** is also set to automatic, this will be determined independently for each cell stacks based on the cell geometry. Note: if probabilistic fracture nucleation is required to generate fractures, it is often because the height:width ratio of the cell stacks is too high. Try using horizontal upscaling to reduce this instead.
- **Propagate fractures in order of nucleation:** This option controls the order in which fractures propagate within each timestep: if selected, fractures will propagate in order of nucleation time regardless of fracture set; if deselected they will propagate in order of fracture set. Propagating in strict order of nucleation time removes bias in fracture lengths between sets, but adds a small overhead to calculation time.
- **Search adjacent cell stacks for stress shadows:** By default, checking for stress shadow interaction is only carried out against other layer-bound fractures in the same cell stack. Some stress shadow interactions may therefore be missed, when the fractures are parallel to and close to the cell boundaries. Use this option to also search the adjacent cell stacks for stress shadow interaction. This will increase runtime. If this option set to automatic, the requirement to search adjacent cell stacks will be determined independently for each cell stack based on its geometry.
- **Minimum radius for microfractures to be included in explicit DFN:** By default the explicit DFN contains only layer-bound fractures; microfractures are represented only in the implicit fracture model. However it is possible to include the larger microfractures in the explicit DFN by specifying a minimum microfracture radius. This should be between 0 and half the brittle layer thickness. DFNs that include microfractures can become very large, so it is only recommended to do this only in small models (e.g. near wellbore models) comprising a few cells (see Section 4 and Figure 16). If this input box is blank, no microfractures will be included in the DFN.
- **Number of microfracture cornerpoints:** This parameter specifies the number of cornerpoints defining the microfracture polygons. Increase this to get a better approximation to a circle.

- **Position of fracture nucleation:** Set this parameter between 0 and 1 to force microfractures to nucleate at a specific height within the layer: if set to 0, fractures will nucleate at the base of the layer; if set to 1, fractures will nucleate at the top of the layer; if set to an intermediate value, fractures will nucleate at a proportional distance between the top and base of the layer. This can be used to model fractures nucleating at a mechanical interface or at a specific stratigraphic layer, e.g. a chert layer. Leave the input box blank to nucleate each microfractures at a random height within the layer (the default).