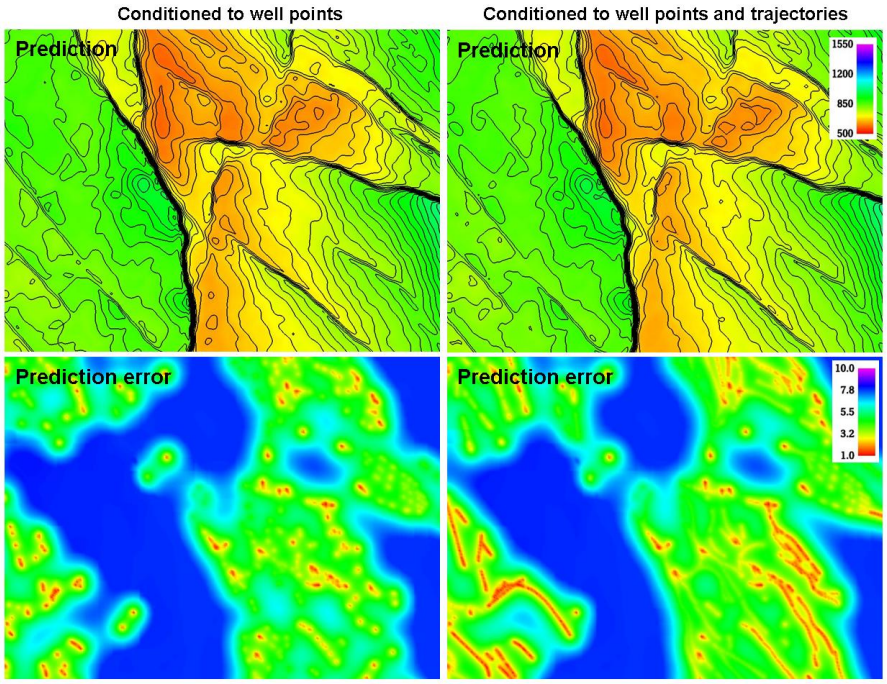
Norsk Regnesentral
NORWEGIAN COMPUTING CENTER

Note

COHIBA user manual version 1.4.1



Note no SAND/03/2009

Authors
Ariel Almendral Vazquez
Pål Dahle
Petter Abrahamsen
Arne Skorstad
Frode Georgsen

Date May 19, 2009

Norwegian Computing Center

Norsk Regnesentral (Norwegian Computing Center, NR) is a private, independent, non-profit foundation established in 1952. NR carries out contract research and development projects in the areas of information and communication technology and applied statistical modeling. The clients are a broad range of industrial, commercial and public service organizations in the national as well as the international market. Our scientific and technical capabilities are further developed in co-operation with The Research Council of Norway and key customers. The results of our projects may take the form of reports, software, prototypes, and short courses. A proof of the confidence and appreciation our clients have for us is given by the fact that most of our new contracts are signed with previous customers.

Norsk Regnesentral
Norwegian Computing Center
Postboks 114, Blindern
NO-0314 Oslo, Norway

Besøksadresse
Office address
Gaustadalléen 23
NO-0373 Oslo, Norway

Telefon · telephone
(+47) 22 85 25 00
Telefaks · telefax
(+47) 22 69 76 60

Internett · internet
www.nr.no
E-post · e-mail
nr@nr.no

Title	**COHIBA user manual version 1.4.1**
Authors	**Ariel Almendral Vazquez , Pål Dahle , Petter Abrahamsen , Arne Skorstad , Frode Georgsen**
Date	May 19, 2009
Publication number	SAND/03/2009

Abstract

This document describes the functionality of the COHIBA surface modelling software and how to run it.

Updated versions of this manual are available at: www.nr.no

Keywords	mapping, surfaces, kriging, depth conversion, conditional simulation
Target group	COHIBA users
Availability	Open
Project	
Project number	
Research field	
Number of pages	97

Contents

1 Introduction

COHIBA provides a fast and accurate method for generating a set of surfaces. COHIBA can use information from:

- Surface observations in wells (well points).
- Horizontal well paths with zone logs.
- Travel time maps.
- Interval velocity models.
- Geological interpretation of depositional thickness.

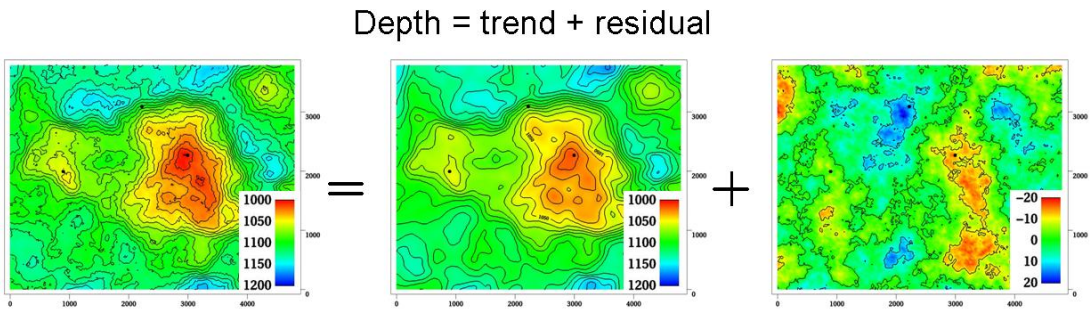
COHIBA uses the available data in a consistent manner to minimise the uncertainty. The accuracy is further improved by linking together all surfaces in a multi-layered model.

COHIBA provides two ways of evaluating uncertainty:

- A local depth uncertainty at every surface location can be calculated.
- Simulated (Monte Carlo) surface realizations can be generated. A set of these span the uncertainty range.

1.1 COHIBA basic idea

Surfaces are modeled as a sum of a *trend* and a *residual*. The trend captures the large scale (deterministic) shape of the surface and the residual captures deviations between the trend and the unknown *true* surface. This is illustrated here:

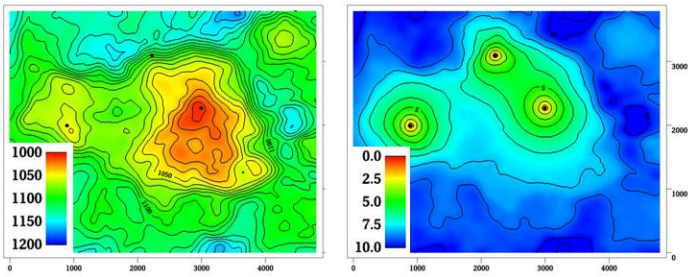


The trend can be a simple map or a linear combination of many maps. The trend determines the main shape of the surface so the choice of trend maps is very important. The residual amplitude is usually small compared to the trend. The residual is specified by an uncertainty (standard deviation) and a spatial continuity determined by a variogram. The residual is in principle random.

COHIBA will adjust the specified trends to match well data. The remaining difference will be described by the residual. The residual will be irregular and random for simulation, just like the figure above. The residual is a smooth interpolation of the known residuals at the wells for prediction. This is illustrated on page 23.

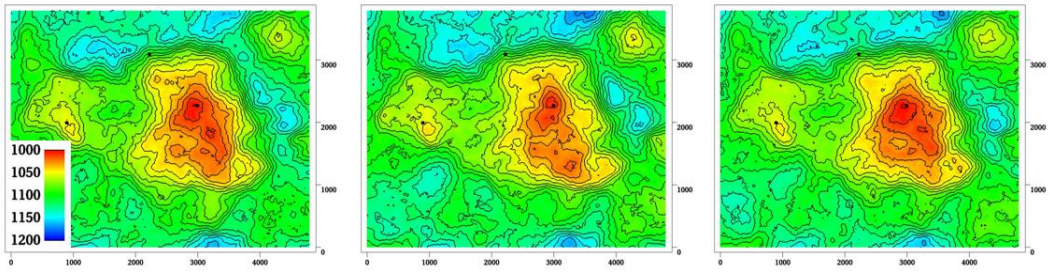
1.2 COHIBA main features

Prediction of surfaces. COHIBA finds the most likely surface given all available data. This is typically used for well planning. The uncertainty (prediction error) can be calculated and displayed as a map. A prediction with corresponding prediction error is illustrated here:



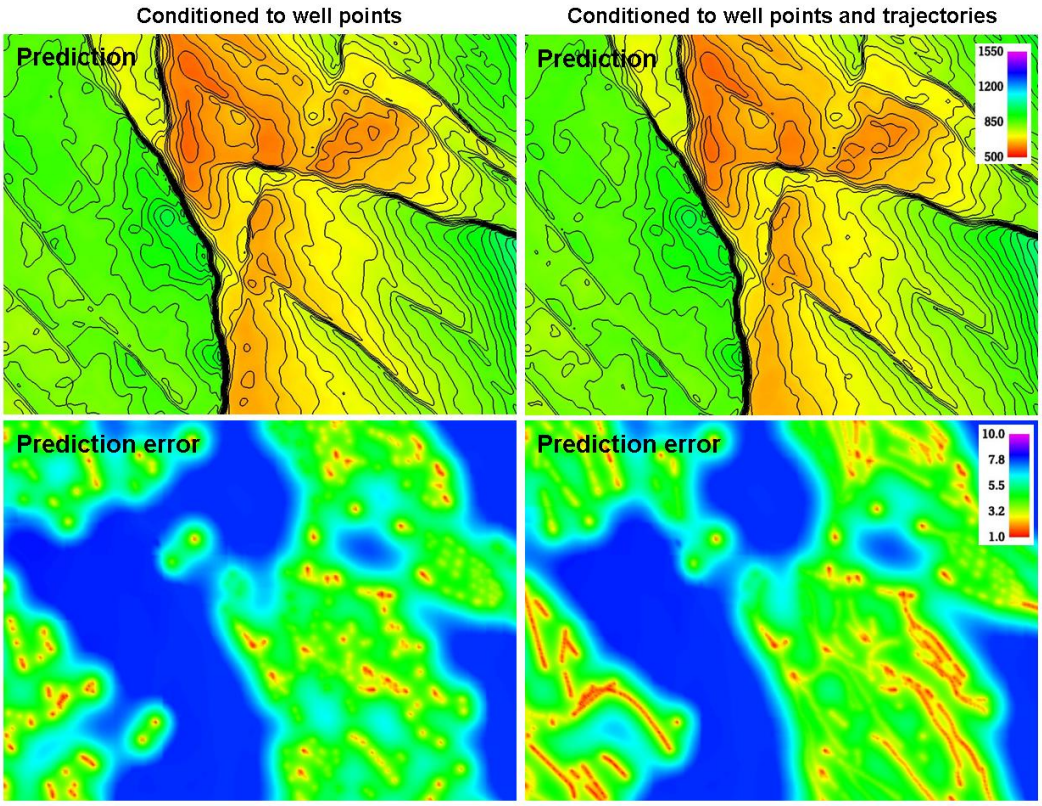
Note how the uncertainty (prediction error) drops to zero at the well points.

Simulation of surfaces. COHIBA can simulate (Monte Carlo) alternative realizations that are consistent with all available data. This is typically used for sensitivity and uncertainty studies. Three simulations are illustrated here:



The average of a large set of simulations will coincide with a prediction. The standard deviation of the large set will coincide with the prediction error.

Horizontal wells. COHIBA ensures that surfaces are correctly positioned according to zone logs in horizontal wells. COHIBA uses an advanced technique where surfaces are conditioned to inequality constraints along the well paths. This ensure correct location of surfaces and reduction in the surface uncertainty along the wells. Here is an example to show the effect of using information from well paths:

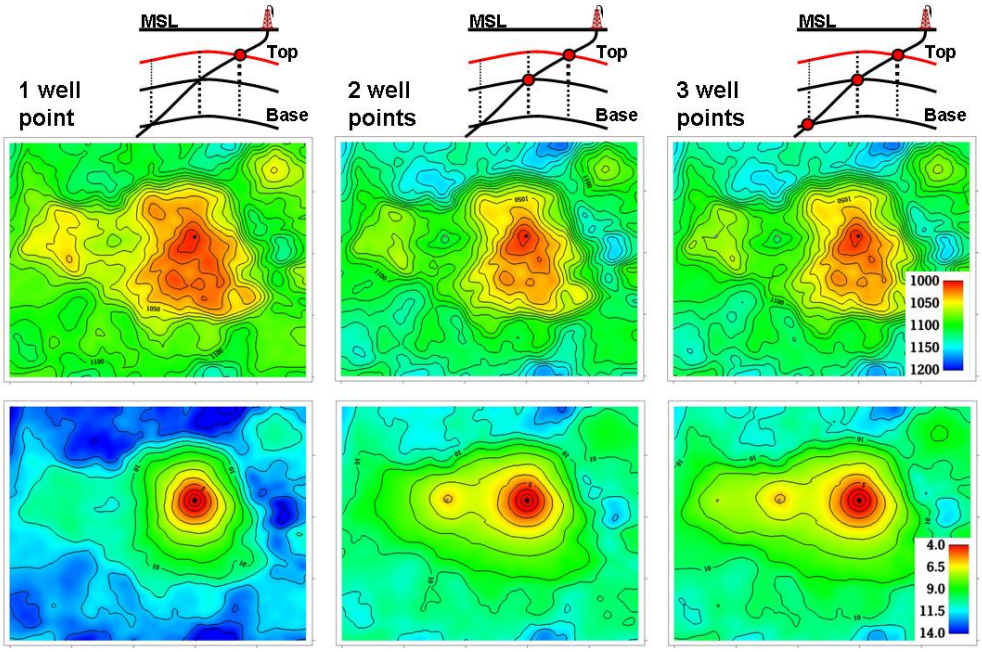


The two upper figures are the predicted depth to the surface and the two lower figures are the corresponding prediction errors. The change in the predicted depths are subtle in this case but the prediction errors has been significantly reduced along the well paths.

This approach works in complex situations with many surfaces and layers and will ensure correct results even in when well paths follow very thin layers.

Multi layer models. COHIBA consider models for all surfaces (reflecting and non-reflecting) in a stratigraphic model *simultaneously*. This has two nice implications:

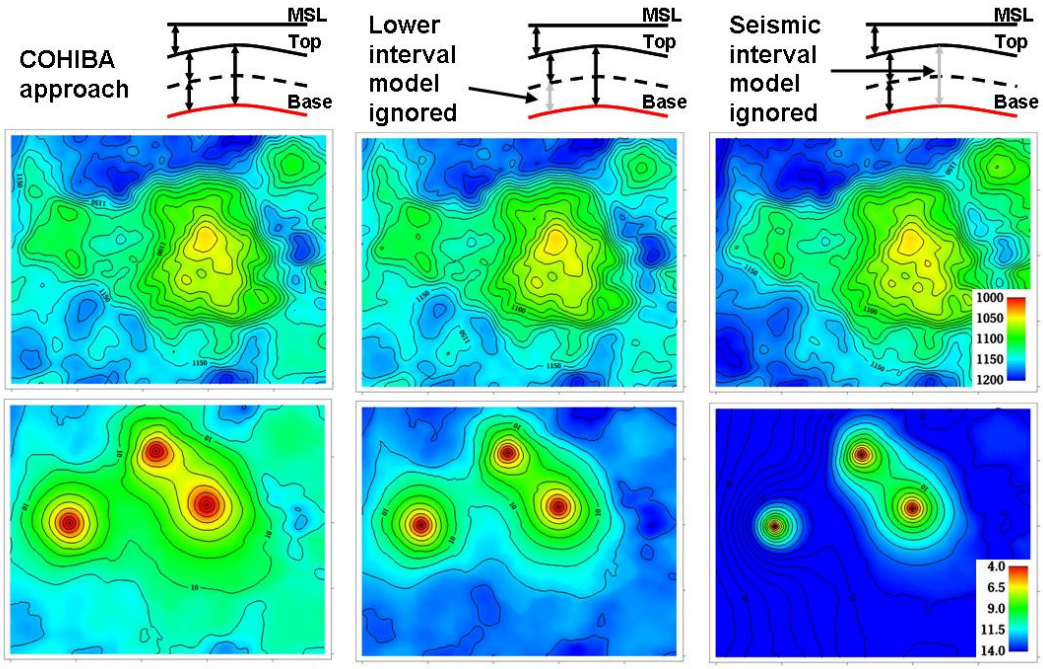
1. A particular surface is constrained by observations from surfaces above and below. This improves precision and reduces the uncertainty. The impact of observations below a surface is illustrated here:



   The illustration shows the top surface (and its prediction errors) from a model with three surfaces. This is the red surface in the schematic cross sections above the contoured maps. The three columns show how the predicted surface and prediction errors are modified as a deviated well is drilled into the two lower surfaces. The lateral location of the well points are easily seen on the prediction error maps. It is clearly seen that prediction errors are reduced by the observations of deeper surfaces.

2. Sufficient well data are well points and zone logs. Unobserved interval thicknesses in deviating or horisontal wells are not needed.

Ambiguous multi layer models. It is sometimes unclear when it is best to add layers from above or from below. COHIBA automatically combines alternative building sequences for multi layer models so that the precision is optimized. A typical example is a layered reservoir outlined by two accurate seismic reflectors. COHIBA solves this by automatically fitting the layers inside the envelope of seismic reflectors. This is illustrated here:

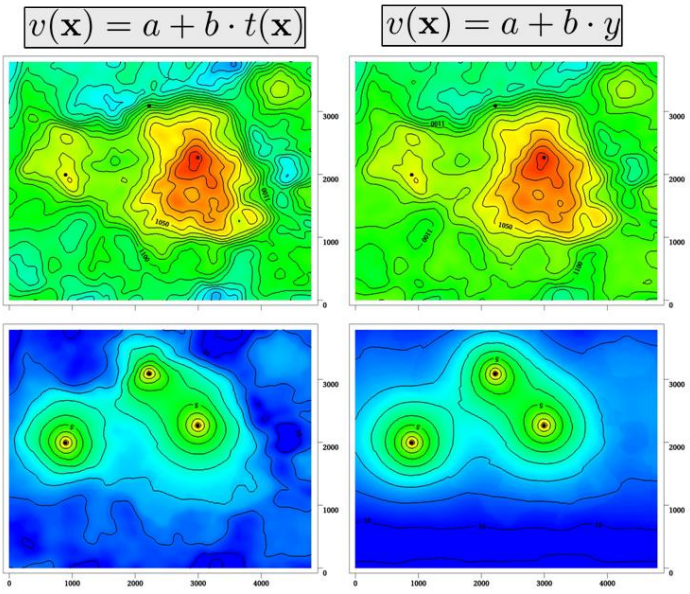


The illustration shows the base surface (and its prediction errors) from a model with three surfaces. This is the red surface in the schematic cross sections above the contoured maps. The top and base surfaces are seismic reflectors. The intermediate surface (dashed line) is not a seismic reflector. The left figures show how COHIBA handles this situation. The two figures in the middle show a typical solution where the model for the lowest layer is ignored. The figures to the right shows what happens if the information from the seismic travel times between the top and base surfaces is ignored.

Isochores. Reservoir layers are typically described by smooth isochores or trends. Isochores are fitted to depth observations taking prior information and spatial correlation into account.

Depth conversion. Seismic reflectors are described using interpreted travel time maps and an interval velocity model for each interval. The interval velocity model can be fitted to depth observations taking prior information and spatial correlation into account.

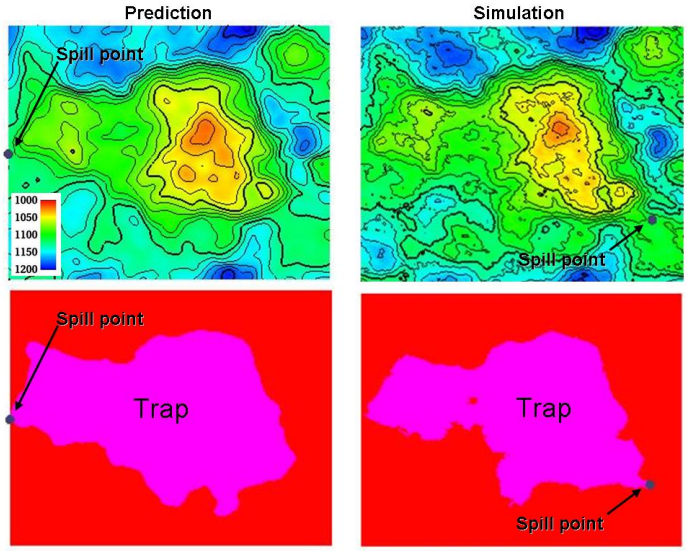
Example of two alternative depth conversion models:



The left prediction and prediction errors are obtained using an interval velocity that increase with the travel time, $t(x, y)$. The right figure shows a result using a north-south interval velocity trend (depend on the y-coordinate). Both interval velocity trends are fitted to the three well points.

Interval velocity maps. Interval velocities from e.g. stacking velocities can be used as the interval velocity model in the depth conversion.

Spill point detection. Spill point and the corresponding trapped area can be found for any surface. Spill points can be detected in a single imported surface or on any surface generated by COHIBA. Here is an example of a spill point and the corresponding trap:



The spill point is detected and written to file (x-, y-, and z-coordinate). Also the trapped area can be written to an output grid file. Note that the spill point for the predicted and the simulated surfaces can be quite different.

1.3 COHIBA methodology

Kriging is used for prediction, and conditional simulation (Monte Carlo) is used for creating alternative realizations. The conditioning data are the well points while seismic data and isochores can be included as trend maps. Kriging with trend maps fitted to well points are called *universal kriging* or *kriging with trends*. COHIBA also offers *Bayesian kriging* which is more robust when few well points are available.

COHIBA uses multiple linear regression to fit the trends to well points. Spatial correlations are taken into account so that the information from clusters of data are given less importance than well points scattered evenly in the area.

Different surfaces are *covariates* so cokriging is used if more than one surface is considered. The interval velocities are also correlated to the surfaces. Therefore surfaces and interval velocities are also covariates.

The residuals (differences between data and trends) are assumed to be *Gaussian random fields* allowing a fast and accurate Fast Fourier Transform (FFT) simulation technique. Experience has shown that the Gaussian assumption is reasonable as long as well points are consistent with trends for isochores and interval velocities.

The basic methods for handling several surfaces and intervals simultaneously were developed back in 1993 [1]. The extension to handling alternative ways of combining layers to obtain a surface was published in 2004 [2]. The original idea on how to constrain the surfaces to zone logs in well paths was published in 2001 [3]. COHIBA combines all these ideas into a consistent and robust approach for accurate surface prediction.

Spill point detection can be applied to any surface in the model or to a single imported surface. The algorithm is described in [4]. The algorithm requires a starting point known to belong to the trap. The starting point is by default the shallowest point on the surface.

1.4 COHIBA input data

Depending on the situation, the user can choose to use the following input data:

Well points. Locations of intersections between well paths and surfaces.

Well paths with zone logs. Well points are extracted and surfaces will be adjusted according to the zone log information. This includes ensuring consistency with horizontal wells.

Travel time maps. These are interpreted travel time maps for seismic reflectors.

- Uncertainty in the interpreted travel time maps either as a constant value (say 4ms) or as a map describing the interpretation in different areas.
- Variograms describing the spatial correlation of the uncertainty for each travel time map.

Interval velocity models:

- Interval velocity trend. Typical example is $v(x, y) = a+b\times t(x, y)$, where a and b are trend coefficients, and $t(x, y)$ is the seismic travel time at location (x, y). (Any linear function of maps can be used.)
- Interval velocity maps.
- Uncertainty in interval velocities either as a constant value or as a map.
- Variograms describing the spatial correlation of the uncertainty for each interval velocity.

Interval thickness models:

- Interval thickness trend. Any linear trends of coordinates and known maps are possible. For instance: $\Delta z(x, y) = a + b \times x + c \times y + d \times m(x, y)$, where a, b, c, and d are trend coefficients and $m(x, y)$ is any known map.
- Interval thickness trend using an isochore map: $\Delta z(x, y) = a \times m(x, y)$, where $m(x, y)$ is a user specified isochore map. Note that pinch outs are possible so $m(x, y)$ can be zero in large areas.
- Uncertainty in interval thickness either as a constant value or as a map.
- Variograms describing the spatial correlation of the uncertainty for each interval thickness.

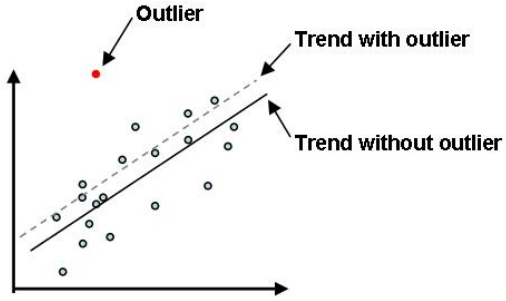
Ordering of surfaces. The surfaces in the model must be ordered according to their depth. This makes it possible to check if well observations are in correct order. Moreover, this ensures that erosion, pinch out, and onlaps can be handled properly.

1.5 Quality control

COHIBA has methods for performing quality control of the input data. Conflicts and ambiguities in the data are handled automatically and reported, so that actions can be taken when necessary. Quality control includes:

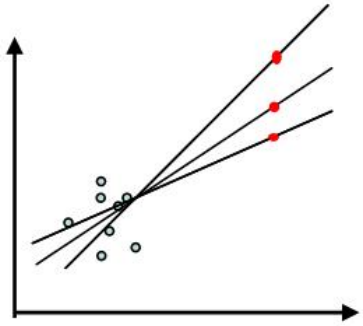
Model conflicts. Well points that are in conflict with the model are flagged. Well points are removed if the conflict may corrupt the results.

- Detection of outliers. These are well points that fit the model assumptions poorly. This can be caused by errors in the well data or by mismatch between well data and model assumptions. The red data in the following figure could be an outlier:



Outliers will often have significant impact on the trends.

- Detection of leverage points. These are data that are potentially outliers but influence the model so strongly that they appear to fit nicely. The following picture shows how a leverage point (the red dot) will influence the regression line:



Conflicting observations. A common problem is that two or more observations are at the same location. There are two ways to handle this, depending on the situation:

1. If the data are identical (same well, same surface, same depth), the best approach is to collapse them to a single observation.
2. If the data have multiple values (different depth) they are retained but given a measurement uncertainty (error) determined by their difference.

Detected outliers, leverage points, and conflicts are flagged for inspection. In severe cases COHIBA removes data that otherwise would corrupt the surfaces.

1.6 Robust estimation of trends

The trends are fitted to well points using linear regression. Outliers and leverage points may corrupt the fitted trends. Therefore, severe outliers are removed prior to the fitting of the trend. This is done in an iterative way: First the trends are fitted using all well points and the mismatch between the trend and well points is analysed. Then, the severe outliers are removed and the trends refitted to the remaining well points. This is repeated until there are no severe outliers left.

The iterative method of fitting the trends ensures a complete removal of inconsistent data so that trends becomes reasonable. Note that these data are also removed from the kriging step where the trends are locally fitted to the well points.

Well points removed from the estimation are flagged and reported. It is recommended to inspect the log-files provided by COHIBA to check for errors and inconsistencies. The log-files can be read into and analyzed in e.g. Excel. This is illustrated in Figure 1.

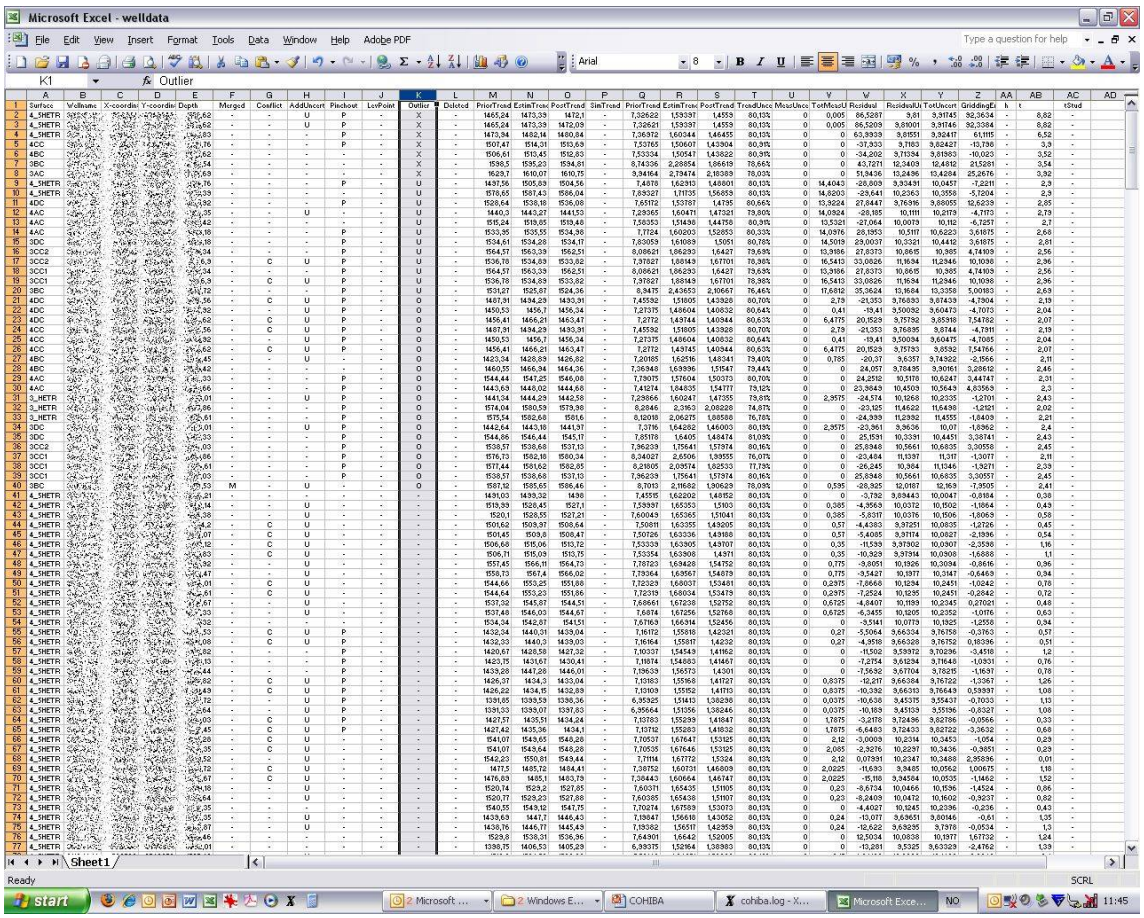


Figure 1. Log-file for well points imported into Excel. Outlier status is reported in column K. The 7 first observations has been removed (row 2–8), the 12 next (row 9–20) are kept but an observation error has been added, and another 20 well points (row 21–40) are marked as outliers.

1.7 Platforms and portability

COHIBA has been developed and tested on Linux and Windows operating systems. In this manual, the directories use the slash: /. On Windows systems the slash must be replaced by a backslash: \.

1.8 The name

COHIBA is an acronyme for **C**orrelated **H**orizon **I**ntervals and **B**ayesian **A**nalysis. But more important it is a nice word that is easy to remember. It is also known to be a famous Cuban cigar.

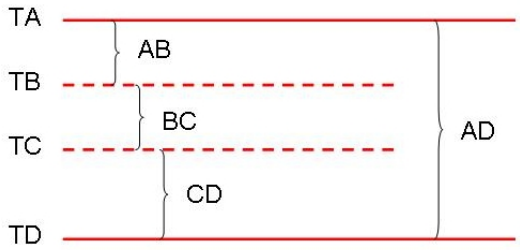
The previous versions of COHIBA were called HORIZON.

2 Terminology and concepts

This manual contains a lot of terminology. Some of these concepts, such as *interval*, are commonly used words but they have specific meaning in this document. This section tries to define exactly what is meant when certain words are used in this manual.

Surface. The surfaces in COHIBA are a set of depth surfaces in a specific depth order. The depth may coincide in pinch out situations. Every surface has a unique name used to identify the surface and data that belongs to it. A surface can be a *seismic reflector* with an associated travel time map. Surfaces can be *erosive* to surfaces below, and *onlapped* by surfaces above.

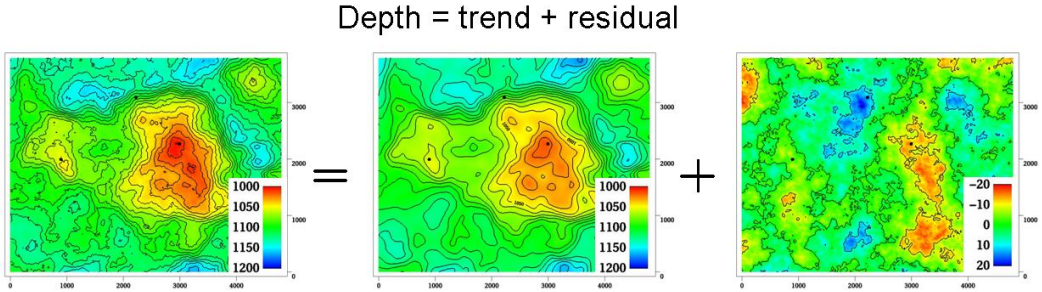
Interval. An interval in COHIBA has a top surface and a base surface. The top and base surfaces are two arbitrary surfaces but they must be in correct depth order (top above base). Any surface can be both a top surface and a base surface in several intervals. A common example is to have one interval spanning the whole reservoir and a set of smaller intervals spanning each individual reservoir zone. This is illustrated here:



The figure shows a schematic cross section of surfaces (`TA`, `TB`, `TC`, `TD`) and intervals (`AB`, `BC`, `CD`, `AD`). Note that surface `TA` is the top for intervals `AB` and `AD`, and surface `TD` is the base of intervals `AD` and `CD`.

Intervals are the most important building blocks in COHIBA. Surfaces are obtained by adding (or subtracting) the relevant intervals above (or below). The depth to any surface is the sum of interval thicknesses. The user must specify the depth sequence of the surfaces, the top and base surface for each interval, and how to build the interval thickness models. From this information, COHIBA automatically figures out how to assemble the intervals to obtain the surfaces in the best possible way.

Model. A model in COHIBA means a *stochastic model* for the intervals and the surfaces. The model is made up of two parts: The trend and the residual. Trends describe large scale behaviour whereas the residual describes variations around the trend on a smaller lateral scale (typically 1000m). This is illustrated by the following picture:



The trend is obtained from a seismic travel time map and a model for the interval velocity trend. The residual is a simulation of a possible realization of the mismatch between the trend and the real depth to the surface.

Given the model *and* the available data, COHIBA can predict or simulate the depth to surfaces.

Trend. The trend for a thickness can be constructed as a linear combination of known trend maps such as $\Delta z = a \times 1 + b \times x + c \times y + d \times m(x, y)$. The three first trend maps, 1, x, and y, are simple, but the last trend map, $m(x, y)$, could for instance include a sedimentological interpretation of the depositional thickness. The values a, b, c, and d are called trend coefficients.

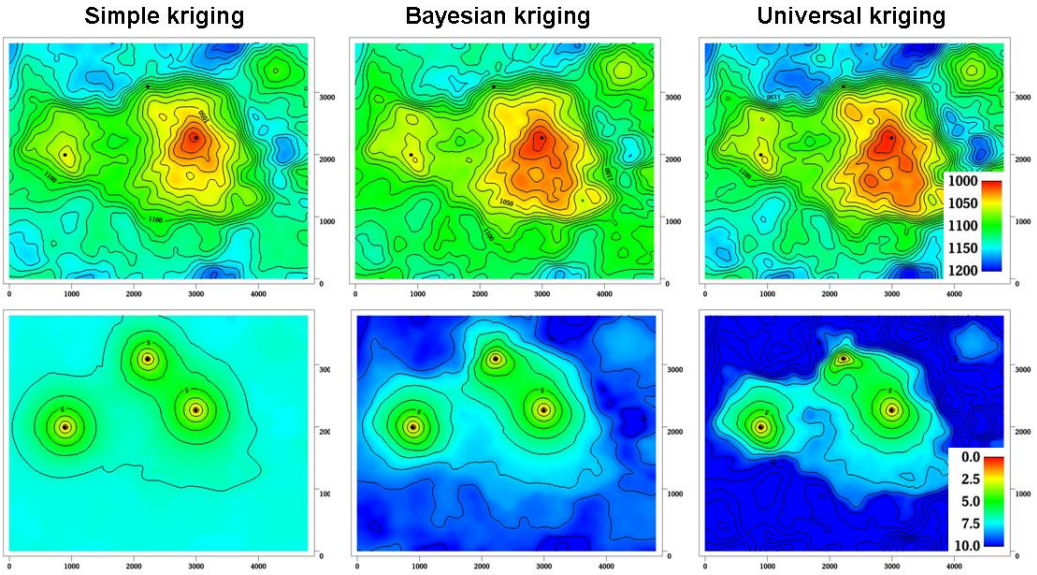
Similarly, interval velocity trends are used in combination with travel time maps for seismic depth conversion.

COHIBA supports any number of trend maps in the linear combination. But, most trends are quite simple and usually contains one or two trend maps.

Trend coefficient. The trend coefficients are assumed to have a Gaussian distribution. COHIBA offers three options of determining the values of the trend coefficients:

1. COHIBA can adjust the trend coefficients to the well points using linear regression[1]. This approach requires that there is enough well points. Otherwise, the estimation might fail or give very uncertain results. This option amounts to using *universal kriging*.
2. The user can specify the trend coefficients. This amounts to *simple kriging*. [2]
3. COHIBA can make a balance between a *prior specification* by the user and the estimates obtained from adapting the trend coefficients to data using linear regression. This is the recommended approach since it handles few (even zero) well points and will approach universal kriging if there is a lot of well points. This approach is called *Bayesian kriging*.

The differences between the three ways of specifying the coefficients and the corresponding kriging types are illustrated here:



1. The linear regression method minimises the generalised least squares (GLS). This is similar to ordinary least squares but it takes into account the variation in precision of the well observations and their spatial clustering.
2. Simple kriging will ignore any uncertainty on the trend coefficients.

The upper row contains the surface depths and the lower row the corresponding prediction errors. There are two trend coefficients in this example that corresponds to surface depth and surface curvature respectively. There are only three well points in this example and that makes it difficult to estimate the coefficents accurately. In particular the trend coefficient determining the curvature is very uncertain. This is clearly seen on the prediction error for universal kriging. The opposite extreme is simple kriging where the trend coefficents are assumed to be known. Bayesian kriging is a balance between the two extremes. Universal kriging is unbiased and will perform better with more well points. Simple kriging will adapt locally to additional well points but the trend coefficients initial prior guess will not be modified by additional well points.

The user must specify a prior distribution for the trend coefficients. This amounts to specifying the prior mean values (expectation) and the prior standard deviation for each trend coefficient. The prior specifications are not used by the universal kriging approach. However, it is very important to always specify reasonable values for the prior mean values because these are used when evaluating depth uncertainties caused by travel time uncertainty and interval velocity uncertainty.

Trend map. The individual trend map(s) must be chosen by the user. This is where the geophysical and geological knowledge enters. Intelligent choice of trend maps will reduce the residual error and therefore improve precision. The individual trend maps can be a simple constant (say 200 meters) or complex maps related to geophysical measurements and geological interpretations.

Residual. The residual is added to the trend to capture the difference between the trend and reality. The residual is assumed to be a *Gaussian random field* with known properties that must be specified by the user. The properties include a spatial correlation function (variogram) and an uncertainty (standard deviation). The spatial correlation function determines the continuity and smoothness of the residual whereas the uncertainty is a measure of how much we believe in our trend. The uncertainty can be specified by a single value or by a map.

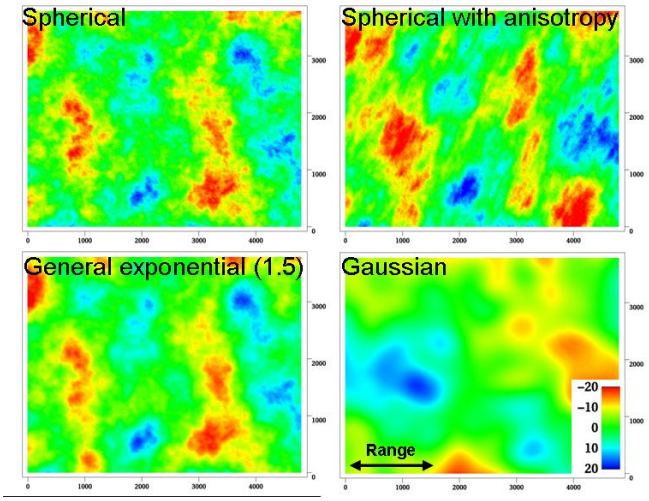
There are three types of residuals used by COHIBA:

1. For seismic reflectors, a residual for the travel time must be specified. This describes the interpretation error. Typical uncertainties are in the range 2–20 msec.

2. For a velocity interval, the interval velocity residual must be specified. Typical uncertainties are in the range 10–200 m/s.

3. For a thickness interval, the thickness residual must be specified. Typical uncertainties are 0–50 meters. (0 is used for pinch outs.)

Here is four examples of simulated residuals with different spatial smoothness:

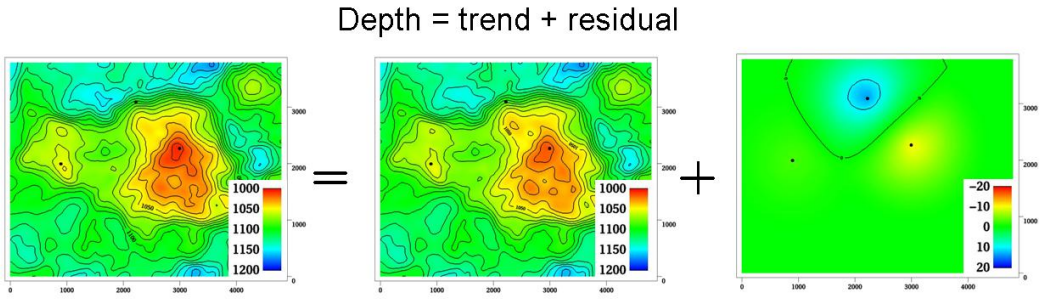


Depth conversion. The top and base surfaces of some intervals can be seismic reflectors. Then COHIBA offers the possibility of specifying interval velocities models rather than interval thickness models. COHIBA will depth convert the travel times using the specified interval velocity model for these intervals. Any mixture of interval thickness and interval velocity models is allowed.

An interval velocity trend can typically look like $v = a \times 1 + b \times t(x, y)$. The thickness is then obtained by multiplying with the travel times: $\Delta z = [a \times 1 + b \times t(x, y)] \times \Delta t(x, y)$.

Prediction. Prediction means to find the *most likely* set of surfaces given the model specifications *and* the data. This is a unique set of surfaces.

A surface model is made of two parts, the trend and the residual. Prediction uses the fitted trend and the interpolated residuals at the well points. This is illustrated here:



Simulation. Simulation is a Monte Carlo approach where trend coefficients and residuals are sampled from probability distributions to obtain a *realization*. A realization is not unique — using a new random seed provides a new set of surfaces. The average of a large set of realizations will coincide with the prediction.

The simulation can be thought of as a two step procedure. First, the trend coefficients are drawn from their estimated distribution and multiplied with the trend maps to fo m the simulated trend. Then the residuals are drawn according to their spatially correlated distribution and added to the trends. Note that the uncertainty from the trends have a global impact whereas the uncertainty in the residual is more local. During this process COHIBA ensures that the result is in correspondence with the well data.

Uncertainty. Uncertainty is a somewhat loose term. It can almost without exceptions be translated into *standard deviation* in this manual. All COHIBA input concerning uncertainty and

measurement uncertainty are given as standard deviations. All output concerning uncertainty is also given as standard deviations.

Well data. There are two types of well data used by COHIBA: *Well points* and *well paths* with a *zone log*. Well points are the intersections between a well path and a surface. Well points can be imported directly from a file. Alternatively, well points can be extracted from a well path (well trajectory) using a zone log. COHIBA will detect all zone transitions and record a well point if the transition is associated with a surface. COHIBA can also ensure that the surfaces are located correctly according to the zone log. This is convenient for horizontal wells.

3 Getting started

COHIBA is started from the command line with the COHIBA model file as an argument:

```
> cohiba.exe model_file.xml
```

A COHIBA project consists of a COHIBA model file and a set of trend maps and well data. We start by running a very simple example, with a constant surface and one well point. We follow the steps:

1. Create a directory `test/` and copy `cohiba.exe` into this directory.
2. Create a file `model_file.xml` using the following XML COHIBA model file:

```
<cohiba>
<project-settings>
  <messages>
    <screen>
      <detail-level>
        <overall> 5 </overall>
      </detail-level>
    </screen>
  </messages>
  <output-grid format="irapclassic">
    <xstart> 448300.0  </xstart>
    <xend>   449300.0  </xend>
    <ystart> 6736000.0 </ystart>
    <yend>   6739000.0 </yend>
    <nx>     30        </nx>
    <ny>     40        </ny>
  </output-grid>
</project-settings>
<modelling-settings>
  <mode>           prediction </mode>
  <kriging-method> bayesian   </kriging-method>
</modelling-settings>
<well-data>
  <surface-points>
    <files format="cohiba"> wellpoints.coh </files>
  </surface-points>
</well-data>
<surfaces>
  <!--Surfaces must be listed in stratigraphic order -->
  <surface>
    <name>                          top    </name>
    <travel-time>
      <value format="constant">     1.0    </value>
      <variogram>
        <type>                      spherical </type>
        <range>                     300       </range>
        <stddev format="constant">  0.01      </stddev>
      </variogram>
    </travel-time>
    <output>
```

```
      <depth>  yes  </depth>
    </output>
  </surface>
</surfaces>
<interval-models>
  <interval>
    <name>          MSL-to-top    </name>
    <top>           MSL           </top>
    <base>          top           </base>
    <interval-type> velocity      </interval-type>
    <trend>
      <coefficient-mean>        1         </coefficient-mean>
      <coefficient-stddev>      0.1       </coefficient-stddev>
      <value format="constant"> 1000      </value>
    </trend>
    <variogram>
      <type>                            spherical </type>
      <range>                           500       </range>
      <stddev format="constant">        5         </stddev>
    </variogram>
  </interval>
</interval-models>
</cohiba>
```

3. Create a file `wellpoints.coh` with the following content:

   ```
   top well 448800 6737500 1220 0.0
   ```

4. From the command line type: `cohiba.exe model_file.xml`

COHIBA will start by reading the COHIBA model file and generate output and surfaces according to the specifications in the COHIBA model file `model_file.xml`. In this case, COHIBA is asked to run a bayesian prediction on the surface `top` using the well point with coordinates $(448800, 6737500, 1220)$. Three directories are created under `test/`: `logfiles/`, `surfaces/` and `welldata/`. The Irap Classic trend surface `dt_top.irap` and depth surface `d_top.irap` are written in the directory `surfaces/`. Logfiles are written under `logfiles/` and `welldata/` contains IRAP RMS well point files.

Almost at the end of the screen output you may find the following table:

```
-------------------------------------------------------------------------------
Interval     Coefficient   PriorMean   PostMean   PriorStd    PostStd    SimMean
-------------------------------------------------------------------------------
MSL-to-top             1      1.0000     1.2173     0.1000     0.0111          -
-------------------------------------------------------------------------------
```

This shows prior and posterior mean and standard deviations for the trend coefficient. The single data at 1220m has moved the prior guess at 1000m to 1217.3m and the uncertainty has dropped from $0.1 \times 1000m = 100m$ to $0.0111 \times 1000m = 11.1m$. This uncertainty should be compared to the residual uncertainties: $0.01s \times 1000m/s = 10m$ (travel time) and $5m/s \times 1s = 5m$ (interval velocity). Combining these gives a total uncertainty of $(10^2m^2 + 5^2m^2)^{1/2} = 11.2m$. This is not a coincidence. A larger residual uncertainty would imply less impact from the well point and a larger posterior standard deviation.

4 COHIBA command line options

The normal way of running COHIBA is with a single COHIBA model file as the argument:

```
> cohiba.exe modelfile.xml
```

The COHIBA model file can of course have any name.

COHIBA offers an alterantive way of setting the seed for the simulation:

```
> cohiba.exe -s 123 modelfile.xml
```

where 123 is the random seed. Normally the seed is either read from a file or set in the COHIBA model file.

4.1 Spill point detection

COHIBA has a special mode where it reads a single surface and reports the spill point:

```
> cohiba.exe -g spillpointgrid.irap
```

The output will look like:

```
*****************************************************************
***                     COHIBA version 1.3.1                    ***
*****************************************************************

Irap classic grid: spillpoint.irap imported.

*********************************************************
***              Spill points summary                 ***
*********************************************************

Xstart          Ystart          XSpill          YSpill          ZSpill
-------------------------------------------------------------------------
349542.094      6426545.500     349292.094      6428595.500     1278.000
-------------------------------------------------------------------------

Irap classic grid: trap_spillpoint.irap exported.

Total CPU  time used in COHIBA:     0.80
Total Wall time used in COHIBA:     1.00
```

The input file with an added prefix `trap_` is written to file. This file contains ones inside the trap and zeroes outside the trap.

The following options are available:

Flag: `-g`
Description: Input grid file. Trigger the spill point detection algorithm for a single surface.
Argument: A string specifying the input grid's file name.
Default: No default.

Flag: `-f`
Description: Format of the input grid.

Argument: `irapclassic / storm`.
Default: `irapclassic`

Flag: `-x`
Description: Start x-coordinate for the spill point searching algorithm.
Argument: Real
Default: The x-coordinate where the surface has a minimum (highest point in surface).

Flag: `-y`
Description: Start y-coordinate for the spill point searching algorithm.
Argument: Real
Default: The y-coordinate where the surface has a minimum (highest point in surface).

Flag: `-m`
Description: Missing as wall flag. See Section A.4.2.4.1.
Argument: `yes / no`.
Default: `yes`

5 The COHIBA model file

The COHIBA model file is a text file in XML (Extensible Markup Language) format . The basic XML syntax for one *element* is:

```
<name attribute = "attribute value">
 argument
</name>
```

where `name` is the key-word for the element and the `argument` contains a value (string, number, etc.) or other elements. The attribute is omitted in most cases. Often `arguments` contains further elements in a nested structure. The COHIBA model file is divided into 5 major elements:

1. `<project-settings>` ... `</project-settings>`
2. `<modelling-settings>` ... `/<modelling-settings>`
3. `<welldata>` ... `</welldata>`
4. `<surfaces>` ... `</surfaces>`
5. `<interval-models>` ... `<interval-models>`

There is a hierarchy of elements belonging to each of these 5 elements. The sequence of the elements within a hierarchy is arbitrary[3] but it is recommended to stick to the suggested order to avoid confusion. Here follows an example of the skeleton of a COHIBA model file:

```
<cohiba>
  <project-settings>
     ......
  </project-settings>
  <modelling-settings>
     ......
  </modelling-settings>
  <welldata>
     ......
  </welldata>
  <surfaces>
     <!--Surfaces MUST be listed in stratigraphic order -->
     <reference>
       ......
     </reference>
     <surface>
       ......
     </surface>
     <surface>
       ......
     </surface>
  </surfaces>
```

3. An important exception is the order of surfaces. These *must* be specified in correct stratigraphic order to obtain correct ordering of the generated surfaces.

```
  <interval-models>
     <interval>
        .....
     </interval>
     <interval>
       ......
     </interval>
   </interval-models>
</cohiba>
```

Note that many of the elements can be repeated. E.g. there will be one `<surface>...</surface>` element for every surface considered in the model.

An extensive example of a COHIBA model file is given in Section B.

A detailed description of each of these main elements and their sub-elements is provided in Appendix A. Most elements are optional and default values will be assigned if the element is omitted.

5.1 Visualizing and editing the COHIBA model file

An XML file is a simple ASCII text file and almost any text editor (WordPad, Word, Emacs,...) can edit the file. However, the structure of the XML file is easier to read in a more specialized XML editor. For instance the Microsoft XML Notepad 2007 editor may be freely downloaded from: http://msdn2.microsoft.com/en-us/xml/default.aspx. This editor is illustrated in Figure 2.

5.1.1 Creating a mapping into Microsoft Excel

It is possible to create a workbook using an XML file. As an example, consider the model file in Section B. Follow the steps:

- Start Excel.
- Choose the XML model file using the **File→Open...** menu.
- Select **Use the XML source task pane**. Excel will then create a tree from the XML source data. So far only the tree structure has been established and no data has been imported.
- Follow the instructions below the tree: **To map elements, drag the elements from the tree onto your worksheet**. We suggest that you choose the repeated `<surface>` or `<interval>` elements.
- Right click on the imported cells and select **XML->Import...**. Select the XML model file again.
- The values will be visible in a table format.

This procedure may be useful for inspection and comparison. It is possible to sort the input data according to variogram type, interval type, etc.

Modified data in the worksheet can be exported to XML format by right clicking on mapped cells and selecting `XML->Export`. Note that only the mapped section of the model file will be exported so this is of limited interest.

5.2 Errors reading the model file

When COHIBA reads the model file it will detect most spelling and syntax errors in the model file.

If COHIBA encounters an error corrupting the structure of the XML syntax a message will be issued to screen and the COHIBA log file. It can look like:

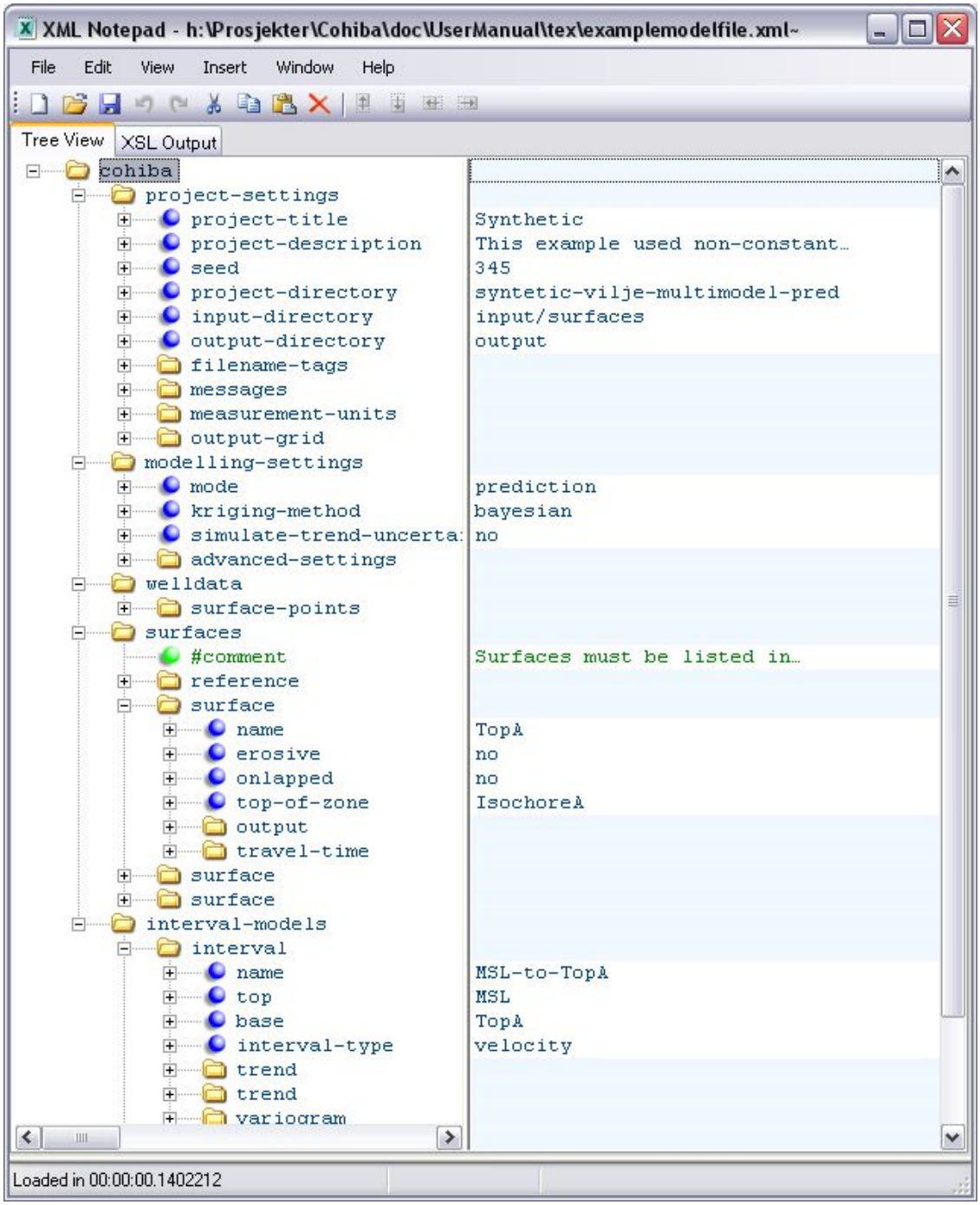


Figure 2. The COHIBA model file used in Section B visualised in the Microsoft XML NotePad 2007 editor. The entries in the right column can be edited.

```
XML error: ln 287, col 1
while parsing 'benchmark/modelfile.xml'.
Error reading end tag of <project-settings>: found </cohiba>

Fatal error: Could not read XML file
```

In this case the final tag `</project-settings>` was forgotten.

If COHIBA finds an unknown key-word a message will be issued. It can look like:

```
Error: Unknown tag <variograms> encountered at line 274, column 7.
Select one of: <name>, <top>, <base>, <interval-type>, <trend>, <variogram>,
               <correlations> or <output>
Parsing error on line 264. Missing <variogram> entry under <interval>

Fatal error: Invalid model file.
```

In this case the `<variogram>` element has been misprinted as `<variograms>`. Note that COHIBA suggests the possible key-words.

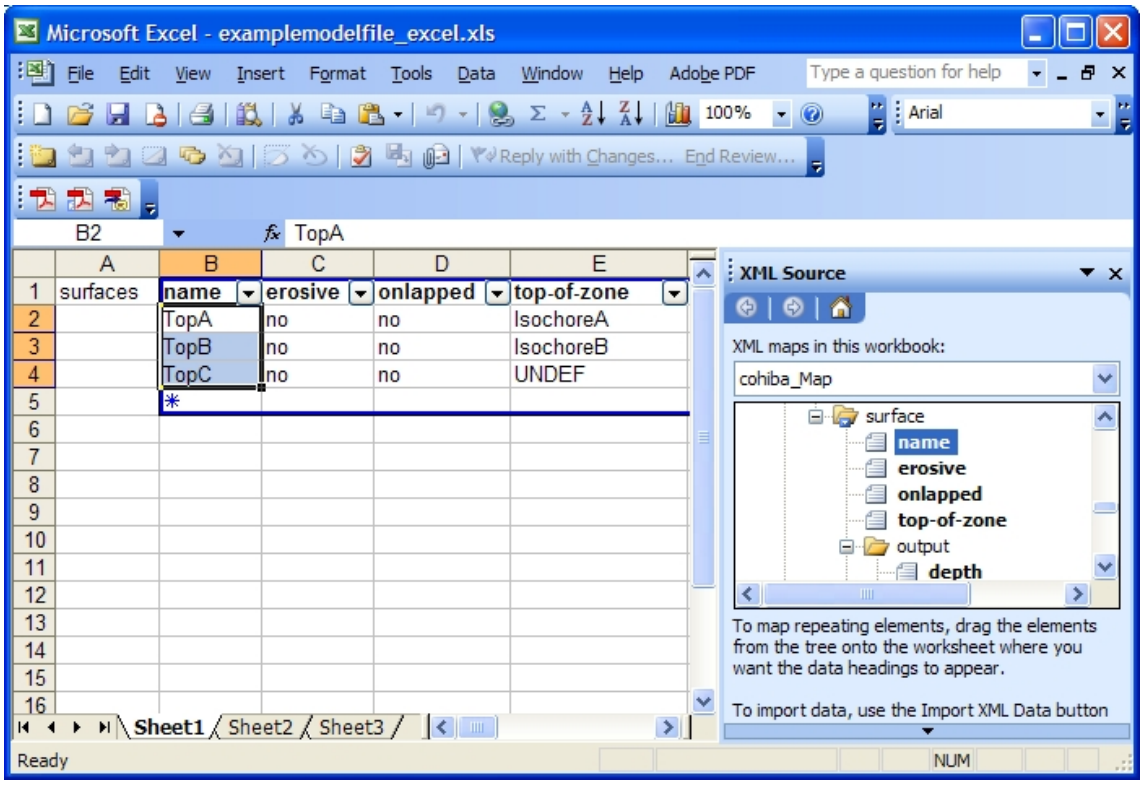
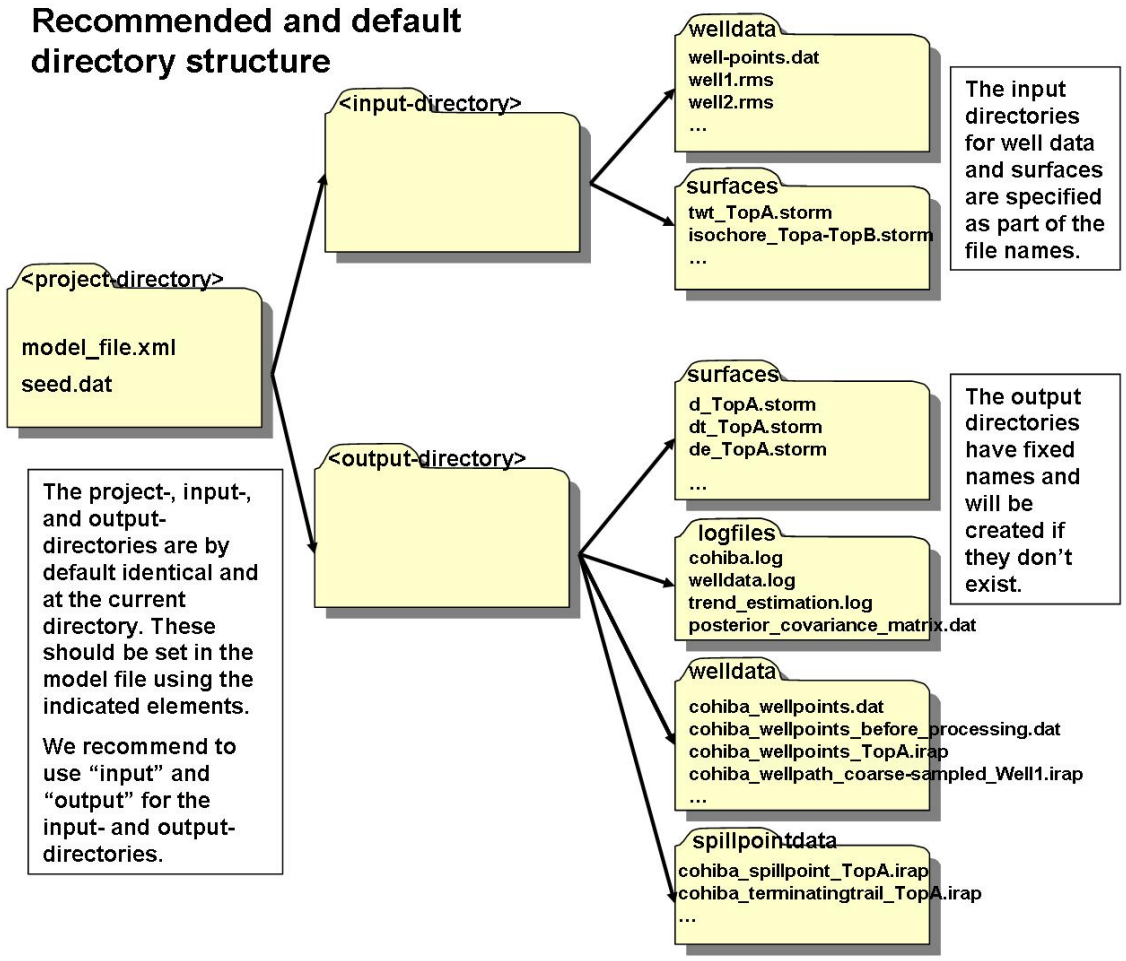


Figure 3. A mapping into Microsoft Excel of the repeated element `<surface>` in the COHIBA model file used in Section B

6 File directory structure

COHIBA reads a lot of information from files and writes results to numerous files. It is partly up to the user to organize these files in a convenient way. Here is an overview of a recommended directory structure and some files used by COHIBA:



Note that the file names are just examples and other names will be apropriate for a specific project.

The directories and file names are specified in the model file. Some output files and directories have fixed names and can not be modified by the user.

7 Surface file format

The following three surface file formats are supported:

1. `constant` Allows to easily define a constant surface with just a single real value.
2. `storm` The STORM BINARY format. See [6] for information on this format.
3. `irapclassic` The Irap Classic format. See [6] for information on this format.

8 Well data

There are two types of well data used by COHIBA: *Well points* and *well paths* with *zone logs*, and optionally, fault indicator logs. Well points, which are sometimes called well picks or well markers, are the intersections between a well path and a surface, and these can be imported directly from file, or alternatively, extracted from a well path (well trajectory) using a zone log. COHIBA detects zone transitions and records a well point if the transition is associated with one of the surfaces in the model. If the well is deviated or horizontal, COHIBA can also use the well path and zone log to ensure that the surfaces are located correctly according to the zone log.

In summary, COHIBA may use the following type of well data:

1. Well paths with zone logs and fault indicator logs given in RMS WELL file format.
2. COHIBA well point format.
3. A combination of 1 and 2.

8.1 Using zone logs

The RMS WELL format is used for importing well paths with zone logs. See [6] for information on this format. These files are specified in the `<well-log>` element.

There is no generic relationship between the surfaces in the COHIBA model and zones in the zone log, and the names of the zones for which a particular surface represents the top, must therefore be supplied using the `<top-of-zone>` element. Non-unique zone names are discussed in Section 8.5.5.

When zone logs are used as input, it is assumed that zones are logged as numbers, and that these numbers are linked to zone names in the log-specification part of the file header. This specification has the structure:

```
GM2005a DISC 1 ABOVE 2 DRAUPNE 3 6BC 4 6AC 5 6AM 6 5CC 7 5CM 8 5BC
```

where `GM2005a` is the log name (specified by `<logname>` element), `DISC` specifies that the log is discrete, and the rest is the list of zone numbers and their associated zone names. The zone numbers must constitude a sequence of positive integers that increase by one.

8.2 Using fault indicator logs

COHIBA can not distinguish between a zone transition made over a surface and a zone transition made over a fault. In both cases, therfore, the transition will be interpreted as a zone top. This is not geologically correct.

To avoid this situation, we can add *fault indicator logs* to the existing well logs. These logs are discrete and consist of 0's and 1's. A 1 means that we are close to a fault and that COHIBA should ignore zone transitions.

The fault indicator log is properly specified in the header of the RMS WELL file as

```
FaultIndicator DISC 0 no 1 yes
```

where the structure is fixed except the label `FaultIndicator`.

8.3 COHIBA well points format

Well points can be imported using a simple ASCII text file format with one line for each observation. The file(s) are specified using the `<well-points>` element. The format is illustrated in Table 1. Each well point must contain the surface name, the well name, the x-, y-, and z-coordinate of the observation, and the vertical uncertainty (standard deviation). The name of the surface must be identical to the string specified in the `<name>` element for the surface. The name of wells are used by COHIBA for consistency checks and reporting.

Table 1. The COHIBA well point format. One line for each observation. The two first columns identify the observations by surface name and well name. The next three columns specify the well point in space, and the last column allows a measurement uncertainty (standard deviation) to be added. The ordering of the observations is arbitrary and has now effect on the generated surfaces. Lines beginning with a # are treated as comments.

```
#-------------------------------------------------------------
# Surface   Well    X-coord.    Y-coord.   Depth     Depth
# name      name                                    uncer.
#-------------------------------------------------------------
TopA        w1     535045.09  6742547.10  1255.12     0.00
TopB        w1     535045.09  6742511.36  1383.87     0.00
TopC        w1     535045.09  6742515.42  1400.23     0.00
#-------------------------------------------------------------
TopA        w2     532273.48  6737634.37  1194.15     1.00
TopB        w2     532223.72  6737667.37  1284.10     1.00
TopC        w2     532203.58  6737694.37  1301.15     2.00
#-------------------------------------------------------------
TopA        w3     536393.00  6736356.62  1155.23     0.50
TopB        w3     536194.01  6736389.87  1255.60     0.65
TopC        w3     536168.78  6736397.00  1300.75     2.98
#-------------------------------------------------------------
```

8.4 Close well points

Two well points are considered laterally close if their separation in the x- and y-directions are smaller than the corresponding grid cell sizes. Since this closeness measure is linked to the grid resolution, fewer points will be considered close as the resolution increases. For two well points to be considered *close* they must also be close vertically. The vertical closeness, however, depends on whether the well points belong to the *same* or *different* surfaces.

If two or more well points are close they are likely to either carry *redundant* information or *conflicting* information. In both cases COHIBA has to modify the well points to avoid numerical instabilities. Depending on the situation, COHIBA will either delete well points, merge well points and/or add uncertainties to well points.

8.4.1 Identical well points

If two or more *identical* well points are observed only one is retained. By identical is meant, same well, same surface, same lateral coordinates (within one centimetre), and same z-coordinate (within one centimetre).

Identical well points occur if different well data files contains identical information or if a single

file contains repeated information.

8.4.2 Close well points belonging to different surfaces — pinch outs

If two or more well points belong to different surfaces, they are considered close if they are laterally close and their separation in the z-direction is smaller than 1 centimetre.

This situation creates a *pinch out*. If three or more points are involved in a pinch out all but the uppermost and lowermost well points become redundant and are deleted.

Well points that are part of a pinch out are flagged by a `P` in the `PinchOut` column of the `welldata.log` file. If the point has been deleted the `P` is given in the `Deleted` column instead.

8.4.3 Close well points belonging to the same surface

If two or more well points belong to the same surface, they are considered close if they are laterally close. They are considered *similar* if the slope between them are less than 10%, that is, dz/dxy < 0.1. They are considered *potentially conflicting* if the slope is larger than 10%. When two well points are involved in a potential conflict, these points, their lateral and vertical distance as well as their intermediate slope are logged to the main log file, again, provided the log level is sufficiently high. A point involved in a potential conflict is also indicated by a `C` in the `Conflict` column of the `welldata.log` file.

If the close well points come from the *same well*, the well points are *merged*. The merged observation gets the arithmetic average coordinates ($\bar{x}, \bar{y}$, and $\bar{z}$), and a new vertical uncertainty is estimated as $\sigma_z = \frac{1}{N-1}\sqrt{\sum_i^N (z_i - \bar{z})^2}$, where N is the number of merged well points. This uncertainty is used for the merged well point. If this uncertainty is smaller than the individual well point uncertainties, then the smallest of the individual well point uncertainties is used instead. A well point that is a result of a merge is indicated by an `M` in the `Merged` column of the `welldata.log` file. The coordinates of the removed points are only written to the main log file provided the log level is sufficiently high.

If the close well points come from *different wells*, the well points cannot be merged since it is impossible to dedicate the merged observation to one particular well. All well points are therefore kept, but an uncertainty equal to that estimated for merged well points is calculated for each pair. This uncertainty is used for one or both well points if it is larger than the uncertainties already assigned to each well point, else no action is taken.

When two well points are close it is important that both well points have a sufficiently high uncertainty; not only to reflect the actual uncertainty in the two well points, but also to ensure numerical stability.

A well point that has been given an increased uncertainty during the well point processing is flagged by a `U` in the `AddUncert` column of the `welldata.log` file.

8.4.4 Well points giving model error

If one or more residual uncertainties are set too low, one or more diagonal elements of the kriging matrix may become too small, possibly leading to numerical instabilities. If too small diagonal elements are found, COHIBA deletes the corresponding well points. This is shown by a `E` in the `Deleted` column of the `welldata.log` file.

To avoid this error, ensure that the relevant residual uncertainties are larger than zero at these well points.

8.4.5 Well points in undefined areas

Well points become redundant if at least one of the input surfaces (usually a time surface) has a undefined value in one of the four grid nodes around the observation. In such cases the well point has to be removed. To avoid such a removal, the surface with undefined values has to be interpolated or extrapolated so that the area near the well point becomes defined.

Well points removed because of undefined values are flagged by `S` in the `Deleted` column of the `welldata.log` file.

The region outside the output grid is also treated as undefined. In this case the well point is flagged by a `B`.

8.5 Extracting well points from zone logs

Consider the 5 zones: `above`, `A`, `B`, `C`, `D`, and refer to the surfaces (transition between zones) as `TA`, `TB`, `TC`, and `TD`. See Figure 4.

8.5.1 Simple zonation

If all zones have thicknesses larger than zero, it is straight forward to extract surfaces from zone transitions. The well passing through Figure 4, makes the zone transitions `above`→`A`, `A`→`B`, `B`→`C`,

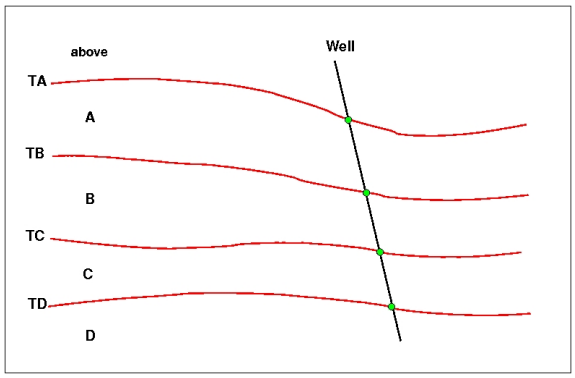


Figure 4. Transitions between a simple set of zones. The well points are given as green bullets.

and `C`→`D`, and the corresponding well points belongs to surfaces `TA`, `TB`, `TC`, and `TD`. The direction of the transition is irrelevant for the assignment of surfaces. The transition `A`→`B` and `B`→`A` are both observations of surface `TB`.

8.5.2 Zones with pinch outs

If a zone has zero thickness in some area, it is said to have a *pinch out*. When a zone transition is made in a pinch out, two surfaces are observed.

In Figure 5, we have depicted a well crossing a set of zones where zone `B` has zero thickness in an area. The three zone-transitions reported in the zone log are `above`→`A`, `A`→`C`, and `C`→`D`. While the former and latter of these transitions refer to surfaces `TA` and `TD` respectively, the middle transition is really the joint transition `A`→`B`→`C`, and is consequently an observation of the two surfaces `TB` and `TC`.

A total of 12 different zone transitions may be observed between 5 zones.

In Table 2, we have summarised these transitions.

Note that the table is symmetric in the “From” and “To” zones.

A well passing through a pinch out is a challenge as two well points get exactly the same coordi-

Table 2. Surfaces observed for different zone-transitions.

From\To	A	B	C	D
A	—	TB	TB, TC	TB, TC, TD
B	TB	—	TC	TC, TD
C	TB, TC	TC	—	TD
D	TB, TC, TD	TC, TD	TD	—

Table 3. Surfaces observed for different zone-transitions if surface B is erosive.

From\To	A	B	C	D
A	—	TB	TB	TB
B	TB	—	TC	TC, TD
C	TB	TC	—	TD
D	TB	TC, TD	TD	—

nates. This may lead to numerical instabilities, requiring one of the observations to be removed. COHIBA is implemented to allow observations of different surfaces to coincide.

8.5.3 Erosive and onlapping surfaces

If we encounter a zone transition of type A→C, it is not always true that this transition may be interpreted as the transition A→B→C. For this relation to hold, surface TB cannot be erosive. If TB *is* erosive, the transition A→C will be an observation of TB alone and *not* TB and TC.

A similar situation occur when there is a surface with onlapping zones/surfaces. This surface is called *onlapped*. The situation is almost identical to the situation with erosion except that erosive surfaces truncate surfaces below whereas onlapped surfaces truncate surfaces above. Note that a surface can be both erosive and onlapped.

In Figure 6, we have illustrated a set of zones where the surface TB is erosive. The zone-transitions observed in the well, are above→A, A→B, B→C, and C→A. As argued above and illustrated in the figure, the latter of these transitions is an observation of surface TB only. Note how the erosiveness of surface TB, implies that one or more zones may be completely undefined (as opposed to being defined but with zero thickness).

Figure 7 illustrates the concept of an onlapped surface. Surface TD is onlapped by zones A, B, and C. As the onlapped surface truncate from below, the zone transition B→D becomes an observation

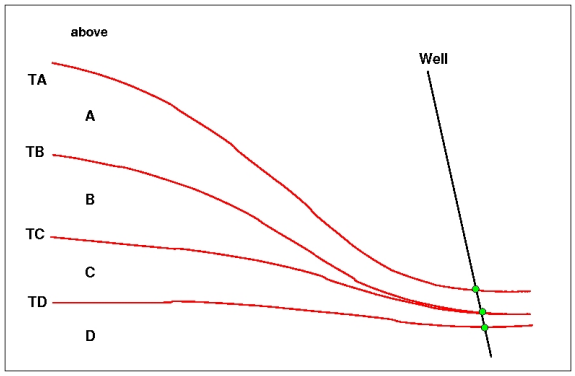


Figure 5. Zones with pinch out. The well points are given as green bullets.

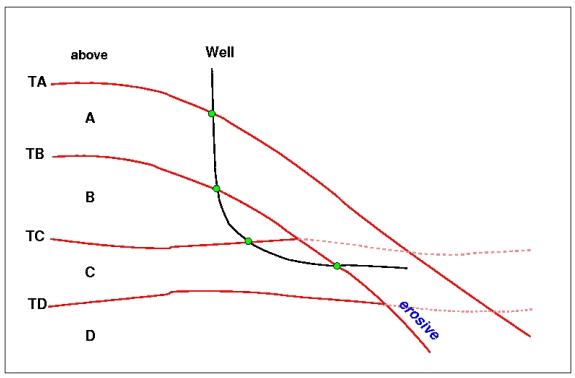


Figure 6. Zones where surface TB is erosive. The well points are given as green bullets.

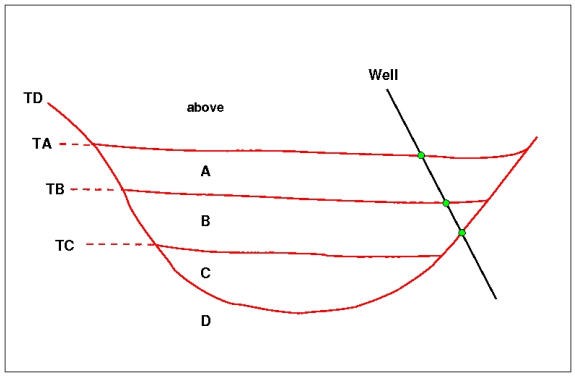


Figure 7. Zones where surface TD is onlapped. The well points are given as green bullets.

of surface TD only.

In Table 3, we have summarised the different zone-transitions that may be encountered with zones A, B, C, and D, where surface TB is erosive, and listed what surface-observation each transition corresponds to. The difference between this table and Table 2 is that, in all zone-transitions involving zone A, only surface TB is observed. This is a result of the erosion. The symmetry in the "From" and "To" zones is maintained, however.

Whether a surface is erosive and/or onlapped or neither of these, is specified by the elements <erosive> and <onlapped>. When two or more surfaces are erosive, the upper surfaces erode the lower. In COHIBA, an erosive surface will erode an onlapped surface.

8.5.4 Faults

A transition between zones can be caused by a fault rather than a surface. This situation is illustrated in Figure 8.

Interpreting a zone transition caused by a fault as a well point will give erroneous results. The well shown in the figure enters from zone A into zone B via the fault. This zone transition is not an observation of TB.

It is impossible to distinguish a transition caused by a fault from an ordinary transition using a zone log alone. Therefore, some extra information must be supplied. This can be done by supplying *ignore points* using the element <well-points-to-ignore>. These are well points that should be excluded if found in the zone logs.

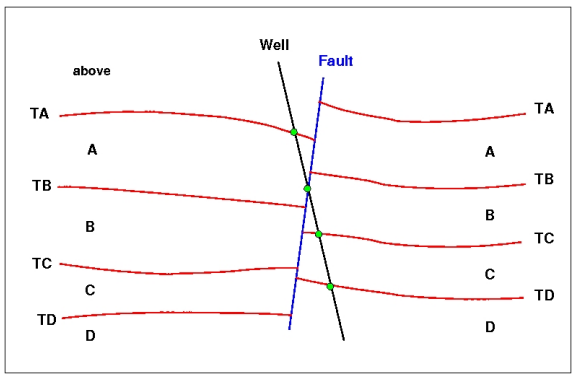


Figure 8. A set of faulted zones. The well points are given as green bullets.

8.5.5 Non-unique zone names

A zone log must be expected to contain more zones than those included in the COHIBA model. In Figure 9, we have depicted this situation. The figure shows nine zones and surfaces, but only

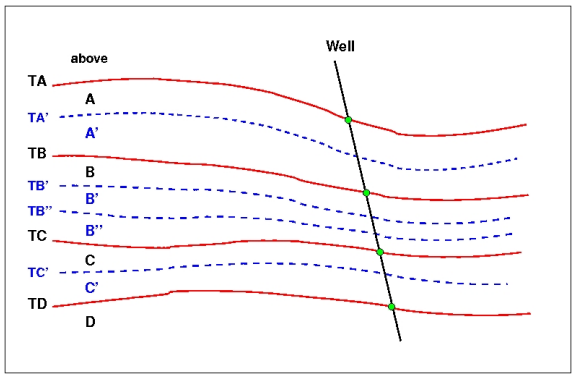


Figure 9. A simple set of zones with a zonation finer than the surface model. The well points are given as green bullets.

four of these surfaces are of interest; the red surfaces labelled `TA`, `TB`, `TC`, and `TD`.

To be able to extract the well points for these surfaces COHIBA needs to know the sequence of the surfaces and which zones that are directly below the surface. The necessary specification is:

```
TA: A, A'
TB: B, B', B''
TC: C, C'
TD: D
```

The corresponding COHIBA model file specification is:

```
<surface>
    <name>                TA          </name>
    <top-of-zone>         A A'        </top-of-zone>
    ...
</surface>
<surface>
    <name>                TB          </name>
```

```
    <top-of-zone>          B B' B''     </top-of-zone>
    ...
</surface>
<surface>
    <name>                 TC           </name>
    <top-of-zone>          C C'         </top-of-zone>
    ...
</surface>
<surface>
    <name>                 TD           </name>
    <top-of-zone>          D            </top-of-zone>
    ...
</surface>
```

It is sufficient to specify only the first zone *if* zone numbers increase with increasing depth. E.g.:

```
...
<surface>
    <name>                 TB           </name>
    <top-of-zone>          B            </top-of-zone>
    ...
</surface>
...
```

This alternative may be used to minimise the probability of introducing errors like typing errors or zonation errors. It requires that zone names are listed in the well-header in the correct depth order. By correct depth order we mean a consecutive increasing sequence of non-negative integers.

Example: Valid zone log:

```
LOGNAME DISC 1 ABOVE 2 A  3 B  4 C  5 D
```

Example: Valid zone log:

```
LOGNAME DISC 1 ABOVE 2 A  3 A'  4 B  5 B'  6 B''  7 C  8 C'  9 D
```

Example: Valid zone log:

```
LOGNAME DISC 10 ABOVE 11 A  12 A'  13 B  14 B'  15 B"  16 C  17 C'  18 D
```

Example: Invalid zone log:

```
LOGNAME DISC 9 ABOVE 11 A 12 A' 13 B 14 B' 15 B" 16 C 17 C' 18 D
```

8.6 Horizontal wells

In parts of a well where the well runs parallel to a surface, the well points alone are not enough to guarantee surfaces consistent with the zone logs. This is illustrated in the left part of Figure 10, where the well is coloured magenta in the region with incorrect zonation.

To avoid that surfaces cross the well path at incorrect locations, COHIBA select a series of constraints along the well path. In the right part of Figure 10, five additional constraints have been introduced. These force the surface TC to lie above the well path in accordance with the zone log. They do not, however, state explicitly where the surface TC is located.

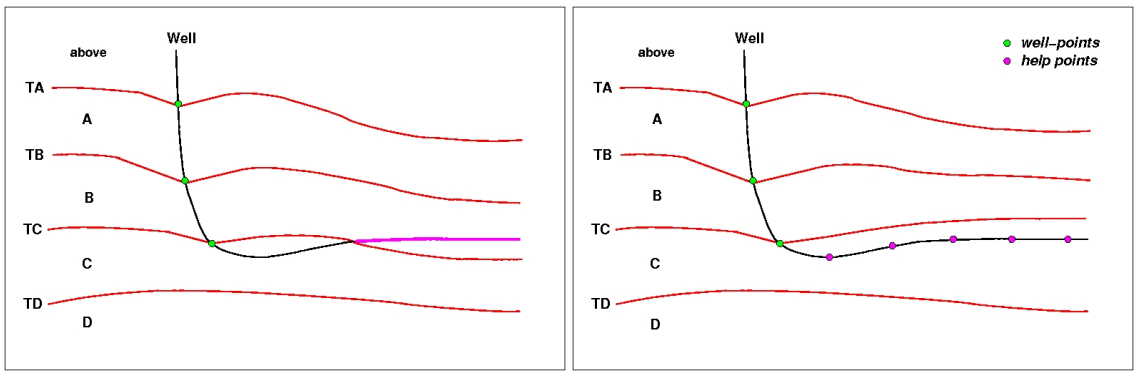


Figure 10. A horizontal well following a zone. In the left figure a part of the well (coloured magenta) have ended up in the wrong zone after the surface-adjustment. In the right figure this problem has been corrected by the introduction of help points (magenta).

COHIBA handles constraints from zone logs according to the method developed in [3]. Without going into the technical details, the approach boils down to finding some additional synthetic well points used in the kriging equations. The additional synthetic well points are chosen so that surfaces are consistent with the model assumptions for trends and residuals and all the well data, well points and zone log constraints.

The location of the additional synthetic well points are found in a few steps:

1. Sample all the well paths at grid resolution.
2. Calculate predicted surfaces (kriging) and corresponding prediction errors at the location of the sampled well paths using only the original well points.
3. Identify conflicts and potential conflicts along the sampled well paths.

Conflicts are locations where a zone log is on the wrong side of a surface. Potential conflicts are locations where the well path is near a surface. The prediction error is used for determining what is near. Potential conflicts are important since they can turn into real conflicts when simulating (Monte Carlo) the surfaces.

The surface depth at the identified locations are re-calculated using a Monte Carlo simulation approach called the Data Augmentation Algorithm. The result is a (non-Gaussian) probability distribution for the depths at the identified locations. This probability distribution is an exact representation of the truncated multivariate Gaussian distribution that has taken all the original well points and the zone log constraints into account. Also note that the trend coefficients are handled so that they will be influenced by the zone log constraints.

From this point, prediction and simulation use two slightly different approaches:

- For simulation, COHIBA draws a single sample from the truncated multivariate Gaussian distribution for all the identified conflicts. This sample is potentially a huge set of *additional* well points along the well paths that are in agreement with the constraints from the zone log. The additional well points are then used together with the original well points as condition data in the simulation.
- For prediction, COHIBA finds the expectation and covariance of the truncated multivariate Gaussian distribution at all the identified conflicts. Then a set of synthetic well points along the well paths, are introduced. These synthetic well points are chosen such that they will

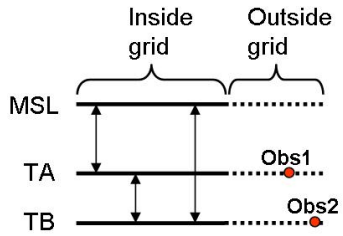
reproduce the expectation and covariance of the truncated multivariate Gaussian distribution at all the identified conflicts. An important property of these synthetic well points, is that they have measurement uncertainty. This has the effect that they gently pull and push the surfaces to their correct depth while maintaining the correct non-zero prediction error.

The simulation approach is accurate in the sense that all model assumptions and well data are treated without any approximations. The prediction approach however, relies on the assumption that the probability distribution at any surface location is Gaussian. In principle, this is not true since the truncations introduced by the zone log constraints are strict and asymmetric. This approximation introduces an error that is small compared to other uncertainties in most realistic cases.

8.7 Using well points outside lateral grid

Well points outside the grid definition can be added using the element `<values-outside-grid>` under `<well-data>`. At least one value will trigger the estimation of trend coefficients with this new point included in the estimation. Well points outside the lateral grid are not included in kriging. For each of these *outside grid well points* additional information must be provided. The number of `<value>` elements defines the minimum input required for other data types such as interval trend outside grid values, travel time outside grid values and standard deviation outside grid values. If no (X,Y) coordinates are provided, COHIBA will automatically generate large coordinates. If a well-name is omitted, COHIBA generates the name "outside-well".

Example: Given three surfaces: `MSL`, `TA`, and `TB` (both `TA` and `TB` are reflectors) and three intervals: `MSL->TA`, `TA->TB` and `MSL->TB`:



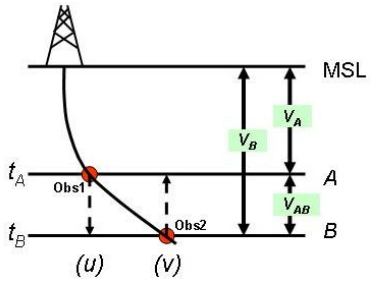
An outside grid well point (`Obs1`) at `TA` and an outside grid well point (`Obs2`) at `TB` require the following input:

Two values for travel time to `TA`
Two values for travel time residual standard deviation for `TA`
Two values for travel time to `TB`
Two values for travel time residual standard deviation for `TB`
Two values for interval velocity trend `MSL->TA`
Two values for interval velocity residual `MSL->TA`
Two values for interval velocity/isochore trend `TA->TB`
Two values for interval velocity/isochore residual `TA->TB`
Two values for interval velocity trend `MSL->TB`
Two values for interval velocity residual `MSL->TB`

This amounts to 20 values in total. If any of the values above are not entered, COHIBA will assume 0 as input, giving meaningless results. The required trend values must be taken at the same lateral location as the well point. This applies whether these extra well points originate from a vertical well or not. A consequence of this is that for a vertical well with one well point at both of two

successive surfaces, the same values must be entered twice. Once for the first well point, and once for the second well point. COHIBA cannot assume that the well is vertical, and has no check for this.

A detailed look at the region outside the reservoir is illustrated below.



Following the system given in the list above, when Obs1 is located at u and Obs2 is located at v COHIBA therefore require:

$t_A(u)$ and $t_A(v)$
$\sigma_{t_A}(u)$ and $\sigma_{t_A}(v)$
$t_B(u)$ and $t_B(v)$
$\sigma_{t_B}(u)$ and $\sigma_{t_B}(v)$
$V_A(u)$ and $V_A(v)$
$\sigma_{V_A}(u)$ and $\sigma_{V_A}(v)$
$V_{AB}(u)$ and $V_{AB}(v)$
$\sigma_{V_{AB}}(u)$ and $\sigma_{V_{AB}}(v)$
$V_B(u)$ and $V_B(v)$
$\sigma_{V_B}(u)$ and $\sigma_{V_B}(v)$

Omitting the model dependency of the V_B velocity model would ideally reduce the required input to:

$t_A(u)$ and $t_A(v)$
$\sigma_{t_A}(u)$ and $\sigma_{t_A}(v)$
$t_B(v)$
$\sigma_{t_B}(v)$
$V_A(u)$ and $V_A(v)$
$\sigma_{V_A}(u)$ and $\sigma_{V_A}(v)$
$V_{AB}(v)$
$\sigma_{V_{AB}}(v)$

However, COHIBA has not yet included the functionality of checking for *required data* only. At the time COHIBA reads the model file part containing the data for one surface, the information on whether or not it is a coupled model with two or just one possible interval paths to each surface, is not available. Therefore, COHIBA demands that the data is given under the assumption of coupled models. Dummy values for $t_B(u)$ and $\sigma_{t_B}(u)$ must therefore be provided when specifying data for surface B.

The necessary information is provided hierarchically as follows:

```
<cohiba>
  <project-settings>
     ......
```

```
</project-settings>
<modelling-settings>
   ......
</modelling-settings>
<welldata>
   ......

  <values-outside-grid>
    <value well-name="my-outside-well" surface-name="TA"
               x="1e10" y="1e10" stddev="0.1">
       1010
    </value>
    <value surface-name = "TB">
       1040
    </value>
  </values-outside-grid>

   ......
</welldata>
<surfaces>

   <!--Surfaces MUST be listed in stratigraphic order -->

   <reference>
     ......
   </reference>

   <surface>
     <name>   TA    </name>
     ......
      <travel-time>
       ......

       <values-outside-grid>
         <value>  1.01  </value>
         <value>  1.03  </value>
       </values-outside-grid>

       <variogram>
          ......

          <values-outside-grid>
            <value>  0.005  </value>
            <value>  0.005  </value>
          </values-outside-grid>

        </variogram>
     </travel-time>
         ......
```

```
   </surface>

    <surface>
     <name>   TB    </name>
     ......
      <travel-time>
       ......

       <values-outside-grid>
         <value>  1.11  </value>
         <value>  1.13  </value>
       </values-outside-grid>

       <variogram>
          ......

          <values-outside-grid>
            <value>  0.005  </value>
            <value>  0.005  </value>
          </values-outside-grid>

        </variogram>
         ......
     </travel-time>

         ......
   </surface>
</surfaces>


<interval-models>
   <interval>
      <name>    MSL-to-TA        </name>

       <trend>
          ....

           <values-outside-grid>
             <value>  2505  </value>
             <value>  2510  </value>
           </values-outside-grid>
          ....
        </trend>

        <variogram>
            ......

           <values-outside-grid>
             <value>  10.0  </value>
```

```
          <value>  15.0  </value>
        </values-outside-grid>
        .....
      </variogram>

   .....
</interval>

<interval>
   <name>    TA-to-TB        </name>

     <trend>
       ....

        <values-outside-grid>
          <value>  32  </value>
          <value>  33  </value>
        </values-outside-grid>

       ....
     </trend>

     <variogram>
       .......

        <values-outside-grid>
          <value>  4.0  </value>
          <value>  5.0  </value>
        </values-outside-grid>
       .......
     </variogram>

   .....
</interval>


<interval>
   <name>    MSL-to-TB        </name>

     <trend>
       ....


        <values-outside-grid>
          <value>  13  </value>
          <value>  14  </value>
        </values-outside-grid>

       ....
```

```
            </trend>

            <variogram>
                 ......

               <values-outside-grid>
                 <value>  12.0  </value>
                 <value>  13.0  </value>
               </values-outside-grid>

               ......
            </variogram>



         .....
      </interval>


   </interval-models>
</cohiba>
```

9 The COHIBA output

COHIBA stores results in the directories `surfaces/`, `logfiles/`, `welldata/` and `spillpointdata/` under the user specified `output-directory`. These directories are automatically created if they are not already present. The main result of COHIBA are the surfaces, but the log files also contain important information on the quality of the result. The `welldata/` directory contains the data used by COHIBA for generating surfaces in formats that are convenient for visualization. The outcome of the spill point detection algorithm is a set of trap surfaces under `surfaces/`, and spill points in `spillpointdata/`.

9.1 Output grids

All the output grids are written to the `surfaces/` directory. The surfaces are classified according to the tag specified by the user in `<filename-tags>` element. COHIBA can produce the following output surfaces:

Depth: The main product — depth to the surface. In prediction mode the most likely surface is generated. In simulation mode, one out of an infinite number of possible surfaces is generated.

Depth trend: Similar to **Depth** but the residual is omitted. For prediction this means that the local adaption to well observations is omitted. For simulation, the simulated residual is also omitted.

Depth uncertainty: Standard deviation at each point on the surface.[4]

Depth trend uncertainty: Standard deviation caused by the uncertainty in trend coefficients.

Depth residual: Difference between **Depth** and **Depth trend**.

Trap: The trap determined by the spill point. The trap surface is 1 if the surface point is inside the trap and 0 otherwise.

Interval thickness: Interval thickness computed as the difference between the two depth surfaces defining the interval (specified in `<top>` and `<base>` elements).

Interval thickness trend: Interval thickness trend computed as the difference between the two depth trend surfaces defining the interval.

Interval velocity: Interval velocity computed as the difference between the two depth surfaces defining the interval divided by the corresponding travel time difference.

9.2 The log files

The four log files are written to the `logfiles/` directory under the user specified output directory. The log files can be used for quality control and inspection of COHIBA results.

9.2.1 The COHIBA log file

The COHIBA log file is specified in the `<logfile>` element described in Section A.1.8.1. The default name is `cohiba.log`. There are 9 sections in the log file with their individual user specified level of detail:

1. **Model settings:** Reports the settings in the COHIBA model file. Detail level set by `<model-settings>`.
2. **Loading input data:** Reports what data has been read from file and possible errors. Detail level set by `<data-loading>`.
3. **Processing well points:** Reports redundant data and possible merging of duplicate well

4. Should be zero at well points but gridding errors will usually result in a non-zero value.

points. Reports potential conflicts and lists well points that have been removed or given additional uncertainty. Reports on pinch outs. Detail level set by `<well-points>`.

4. **Sampling well paths/zone logs:** Reports on processing of well paths with zone logs. Detail level set by `<well-paths>`.
5. **Trend coefficients:** Reports the estimated values for the trend coefficients in the trends. Detail level set by `<trend-coefficients>`.
6. **Well points compared to posterior trend surfaces:** Reports poor matches between well points and trend model. These are classified as outliers or leverage points. Detail level set by `<outliers>`.
7. **Condition to well points:** Reports errors. Usually caused by a trend model incompatible with the well points. Detail level set by `<well-point-conditioning>`.
8. **Condition to well paths:** Reports the additional data points sampled from the well paths and possible mismatch between zone logs and final surfaces. Detail level set by `<well-path-conditioning>`.
9. **Post process surfaces:** Reports compensation for gridding errors. Detail level set by `<post-processing>`.

The log file is intended for catching problems during execution and for examining the main results. The output to screen is identical to the log file output provided the same detail levels have been specified. More detailed analysis are provided in the files `trend_estimation.log` and the `welldata.log`.

9.2.2 The `trend_estimation.log` file

This file contains an analysis of the trend coefficients in the trends. There is one line for each trend coefficient. The file is organised in columns described in Table 4. Missing or undefined entries are marked by an '–'.

The file is formatted for easy import into an Excel spreadsheet. See Section 9.2.5.

Use the `trend_estimation.log` file to inspect the quality of the trend coefficient estimates. Trend coefficients with low `UncertRed` might introduce too much uncertainty and could be removed if the t-value (`tEstim` or `tPost`) is small.

9.2.3 The `posterior_covariance_matrix.dat`

Posterior covariance matrix between trend coefficients. The diagonal is the square of the posterior uncertainty `PostUncert` given in file `trend_estimation.log`. The sequence follows the sequence in `trend_estimation.log`.

9.2.4 The `welldata.log` file

This file contains an analysis of the well points and how they fit to the surfaces. It also contains a summary of actions taken on well points. There is one line for each well point. See Table 5 for a detailed description of the columns.

The file is formatted for easy import into an Excel spreadsheet. See Section 9.2.5.

This file is important to study for a proper quality control of results.

9.2.4.1 Flags in `welldata.log`

The user should inspect for possible flags. This is an important quality control step that may detect serious errors in model and data.

Close well points in the same well and surface will be merged by COHIBA. Merged data are flagged in the `Merged` column. This will usually not require any actions from the user but it is recommended to check why redundant well points are supplied.

When two close well points are identified, COHIBA checks if there is a potential conflict, that is, the slope between the well points is more than 10%. A potential conflict is flagged in the `Conflict` column. This typically happens if side tracks have been inconsistently adjusted. A potential conflict might be a serious error and should be checked by the user. An erroneous well point can be removed by adding it to the well point file specified in the `<well-points-to-ignore>` element.

COHIBA automatically adds a measurement uncertainty to all close well points to ensure numerical stability.[5] This is flagged in the `AddUncert` column. The amount of uncertainty added can be found by comparing the columns `TotMeasUncert` and `MeasUncert`. The added uncertainty depends on the mismatch between the potentially conflicting data and the difference in the merged data.

The `LevPoint` flag signals that a particular data point is a leverage point. A leverage point has a very strong influence on the trend coefficients in the trends so these data should be inspected to ensure that they are correct. Leverage points are less likely when there is a lot of data.

Mismatches and conflicts between the trend model and the well points are flagged in the `Outlier` column. If the data had a perfect Gaussian distribution, 5% of the data would be classified as outliers. Outliers are classified in three levels: *Could* be an error, *likely* to be an error, and *surely* an error. COHIBA automatically adds a measurement uncertainty to the likely errors, and automatically removes those well points that are classified as surely an error. COHIBA handles all these situations automatically, *but* it is strongly recommended to look into the reasons for the errors.

The final flag in the `Deleted` column shows the well points that have been deleted prior to the trend estimation. There are three situations when well points are removed: Well points outside the grid boundary are not used, well points are a pinch out between other surfaces, and if the specified residual uncertainty (variogram) is zero at the well point location.

9.2.4.2 Trend values in `welldata.log`

The trend values — columns `PriorTrend`, `EstimTrend` `PostTrend`, and `SimTrend` — are obtained by evaluating the trends at the well point locations. Some of the trend values might be undefined depending on the kriging method and the mode (prediction/simulation).

These values should be compared to the observed depth value in column `Depth`.

9.2.4.3 Trend uncertainty in `welldata.log`

These are given in columns `PriorTrendUncert`, `EstimTrendUncert` and `PostTrendUncert`. These values are obtained by evaluating the impact of the uncertainty in the trend coefficients at the well point locations. Some of these columns may be undefined depending on the kriging method and the mode (prediction/simulation). All values are standard deviations.

For the Bayesian kriging mode the column `TrendUncertRed` is evaluated. It shows

$$(\texttt{PriorTrendUncert} - \texttt{PostTrendUncert})/\texttt{PriorTrendUncert}$$

in percent. The value will be between 0% and 100%. A small value indicates that well points has minor impact on the estimated trend in this location. Consider increasing the prior standard deviation of the trend coefficents if well data has minor impact on the trend at many locations.

9.2.4.4 Measurement uncertainty in `welldata.log`

Measurement uncertainty (error) is specified by using the last column (`Depth uncertainty`) in the well points files. These uncertainties are reported in column `MeasUncert`. COHIBA adds additional measurement uncertainty when

5. A measurement uncertainty is only added if the well point has no initial measurement uncertainty.

well points are merged or if potential conflicts are detected. The resulting total uncertainty is reported in `TotMeasUncert`. These are reported as standard deviations.

To obtain the measurement uncertainty automatically added by COHIBA use the formula:

$$\sqrt{\texttt{TotMeasUncert}^2 - \texttt{MeasUncert}^2}.$$

(Variances add linearly.)

9.2.4.5 Residuals and uncertainties in `welldata.log`

The column `Residual` contains the difference between the observed depth (in `Depth` column) and the trend. The trend depends on the kriging method and the mode (prediction/simulation) specified in the model file.

The `ResidualUncert` column shows the model uncertainty obtained by aggregating the uncertainty in all intervals that are needed to obtain this surface. This include uncertainties in travel time, interval velocity, and interval thickness. All these uncertainties are specified in the `<variogram>` elements.

The `TotUncert` column contains the total uncertainties and is a combination of `ResidualUncert` and the uncertainty in the trend (i.e. `PostTrendUncert` with Bayesian kriging mode). Comparing the `TotUncert` column to the `ResidualUncert` column shows if it is the local residual uncertainty or the global trend uncertainty that dominates.

The surfaces are stored on grids and well observations will rarely coincide with the grid nodes. A gridding error is therefore inevitable and the `GriddingError` column shows the error at each observation. Note that the gridding error is proportional to the `Residual` values. Severe outliers can have large gridding errors.

9.2.4.6 Statistical measures for outliers in `welldata.log`

Leverage points are well points that influence the estimated trend values strongly. A corrupted leverage point might not be identified as an outlier because the estimates adapts to it. The column `h` contains the diagonal elements of what is called the hat-matrix. This is a by-product of linear regression and can be used to identify leverage points. It is not available if the linear regression fails.

The `t` column contains the t-value for each well point:

$$|\texttt{Residual}|/\texttt{ResidualUncert}.$$

The t-value is used for classifying outliers. Large values indicates that there is a severe mismatch between the well point and trend. Values above 1.96 are flagged as outliers, above 2.57 as uncertain, and values above 3.10 is classified as error. If the well points had a perfect Gaussian distribution this would occur with a frequency of 5%, 1%, and 0.2% respectively.

The `tStud` column is similar to `t` but includes a correction for leverage points. This makes it possible to identify leverage points as outliers even though the trend coefficients has adapted strongly to them. See [5] for more details.

9.2.5 Importing log files into Excel

Both `trend_estimation.log` and `welldata.log` are normal (ASCII) text files, but formatted for easy import into Excel. This is recommended since Excel offers many ways of sorting and colour coding that simplifies the investigation. Also, the log files have many columns so using the `Hide/Unhide` column possibilities that Excel offers is convenient.

To import the files into Excel follow these steps:

1. Start Excel.

2. Choose the appropriate file from the **File→Open...** menu. (Choose **Files of type: All Files (*.*)**).

3. A **Text Import Wizard** appears. Toggle on **Delimited** and choose **Next**.

4. Toggle on **Space** in the **Delimiters** section. Toggle on **Treat consecutive delimiters as one**. Then choose **Next**.

5. Choose **Advanced...**. In the new panel change the **Decimal separator** to '.' and press **OK**.

6. Press **Finish**.

9.3 Well point files

For convenience COHIBA generates a set of ASCII IRAP CLASSIC well point files.

9.3.1 The `cohiba_wellpoints_before_processing.dat` file

This file contains all the well points initially considered by COHIBA. The well points can be imported from multiple COHIBA well point files and from well paths containing zone logs. The well points specified in the files in the `<well-points-to-ignore>` element is removed from this list.

This file is in the COHIBA well point format illustrated in Table 1.

9.3.2 The `cohiba_wellpoints.dat` file

COHIBA merges duplicate data and removes unnecessary or corrupt data. The remaining data after this pre-process is written to the `cohiba_wellpoints.dat` file.

This file is in the COHIBA well point format illustrated in Table 1.

This file can be used as input to COHIBA. Note however that it does not contain all information from well paths with zone logs; only the zone transitions are kept.

9.3.3 The `cohiba_wellpoints_"surface name".irap` file

Contains the well points in `cohiba_wellpoints.dat` that belongs to surface `"surface name"`.

9.3.4 The `cohiba_wellpath_coarse-sampled_"well name".irap` file

The points detected in `"well name"` are sampled every 200m. This constitutes a coarse sample.

Table 4. The `trend_estimation.log` file. Description of columns.

Tag	Description
`Interval`	Interval name.
`TopSurface`	Name of top surface.
`NumObs`	Number of observations in top surface.
`BaseSurface`	Name of base surface.
`NumObs`	Number of observations in base surface.
`Trend`	The interval often contains more than one trend coefficient. If there are 3 in this interval they will be labelled `1`, `2`, and `3`. The numbering will follow the sequence of the `<trend>` elements.
`PriorMean`	Prior mean value for the trend coefficient. The value is specified by the `<coefficient-mean>` element.
`EstimMean`	Estimated mean value for the trend coefficient. This value is estimated using generalized least squares (GLS) which amounts to linear regression where the spatial correlation in the well data is accounted for. This value may fail to exist if there are too few data in some of the intervals. Not calculated if simple kriging is specified.
`PostMean`	Posterior mean value for the trend coefficient. This value is estimated using a Bayesian approach which amounts to a weighting between the prior specification and the GLS estimate. Note that this estimate exist even if the GLS estimate fails to exist. Not calculated unless Bayesian kriging is specified.
`SimMean`	Simulated (Monte Carlo) value for the trend coefficient. The simulated values are drawn from the estimated distribution. For simple kriging it will be identical to the value specified in the `<coefficient-mean>` element unless the `<simulate-trend-uncertainty>` element is given the value `yes`. Only calculated in simulation mode.
`PriorUncert`	Prior standard deviation for the trend coefficient. The value is specified by the `<coefficient-stddev>` element.
`EstimUncert`	Estimated standard deviation for the trend coefficient. This value is estimated using generalized least squares (GLS). Not calculated if simple kriging is specified.
`PostUncert`	Posterior standard deviation for the trend coefficient. This value is estimated using a Bayesian approach. This value will always be less than `PriorUncert` and `EstimUncert`. Not calculated unless Bayesian kriging is specified.
`UncertRed`	For universal kriging: (`PriorUncert` – `EstimUncert`)/`PriorUncert`. For Bayesian kriging: (`PriorUncert` – `PostUncert`)/`PriorUncert`. Uncertainty reduction in percent from the prior specification. Value is always in the range 0 - 100% for Bayesian kriging. For universal kriging negative values occur when the prior specification is more certain than the estimate. A small (or negative) value signals that well data has minor impact. Consider increasing the prior standard deviation of the trend coefficients if well data has minor impact. Consider using Bayesian kriging if a negative value occurs.
`tPrior`	`PriorMean/PriorUncert`.
`tEstim`	`EstimMean/EstimUncert`. A large t-value (>2) shows that this is an important trend for explaining depth or interval velocity variations according to the well points. A small value (<1) shows that the well points does not support the existence of such a trend. Consider removing it.
`tPost`	`PostMean/PostUncert`. See the comments on `tEstim`. Note that the conclusions using this t-value are less significant since these estimates can be heavily influenced by the a priori specifications.

Table 5. The `welldata.log` file. Description of columns. File contains one line for each well point.

Col.	Tag	Value	Description
Data identification:			
A	`Surface`	Text	Name of surface.
B	`Wellname`	Text	Name of well.
C	`X-coordinate`	Real	UTM coordinate.
D	`Y-coordinate`	Real	UTM coordinate.
E	`Depth`	Real	Observed depth. From well points file or transitions in zone logs.
Flags:			
F	`Merged`	`M`	Well point merged with other well point(s) from same well and surface.
G	`Conflict`	`C`	In potential conflict with other close well points in the same surface. (Slope > 10%)
H	`AddUncert`	`U`	Uncertainty added to well point because it is close to other well points. (Within grid cell distance)
I	`Pinchout`	`P`	Pinch out. Identical to well point from different surface. Deleted if squeezed between two surfaces.
J	`LevPoint`	`L`	Leverage point. Large impact on the trend estimation. Check input values!
K	`Outlier`	`O/U/S`	Data does not fit the trends properly. Check input values! `O` means that the data *could* be erroneous. `U` means that it is *likely* that data is erroneous so uncertainty has been added. `X` means that this data *surely* is erroneous and it is removed from estimation and interpolation.
L	`Deleted`	`B/P/E/X/F/S`	The data has been excluded from *estimation* and *kriging*. There are 6 reasons for deletion: `B` if well point is outside grid boundary. `P` if well point is a pinch out between other surfaces. `E` if the specified residual uncertainty (variogram) is zero. `X` if well point is a severe outlier. `F` if well point was close to a fault according to a fault indicator log. `S` if some surface or interval has a missing value at the well point.
Trend values:			
M	`PriorTrend`	Real	Depth trend using prior trend coefficients. (‘`-`’ if universal kriging mode.)
N	`EstimTrend`	Real	Depth trend using estimated (least squares) trend coefficients. (‘`-`’ if too few data or simple kriging.)
O	`PostTrend`	Real	Depth trend using posterior trend coefficients. (‘`-`’ unless Bayesian kriging mode.)
P	`SimTrend`	Real	Depth trend using simulated trend coefficients. (‘`-`’ unless Simulation mode.)
Trend uncertainty (standard deviation):			
Q	`PriorTrendUncert`	Real	Prior depth trend uncertainty. (‘`-`’ if universal kriging mode.)
R	`EstimTrendUncert`	Real	Estimated depth trend uncertainty. (‘`-`’ if too few data or simple kriging.)
S	`PostTrendUncert`	Real	Posterior depth trend uncertainty. (‘`-`’ unless Bayesian kriging mode.)
T	`TrendUncertRed`	Real	Reduction (in %) between prior and posterior uncertainty. (‘`-`’ unless Bayesian kriging mode.)
Measurement uncertainty (standard deviation):			
U	`MeasUncert`	Real	Measurement uncertainty specified by the user.
V	`TotMeasUncert`	Real	Total measurement uncertainty including user specified and automatically added.
Residuals and uncertainties:			
X	`Residual`	Real	Difference between data and used trend.
Y	`ResidualUncert`	Real	Residual depth uncertainty from specified variograms.
Z	`TotUncert`	Real	Sum of trend uncertainty and residual uncertainty.
AA	`GriddingError`	Real	Numerical resolution error due to gridding.
Statistical measures for outliers:			
AB	`h`	Real	Number between 0 and 1. Value above 2 $\times$ (`# of trend coefficients`)/(`# of data`) is classified as leverage point. (‘`-`’ if too few data.)
AC	`t`	Real	$t = \lvert$`Residual`$\rvert$/`ResidualUncert`. Value above 1.96 (5%) is flagged as ‘`O`’ in `Outlier`. Value above 2.57 (1%) is flagged as error ‘`U`’ and `Residual`/2 is added as measurement uncertainty. Value above 3.10 (0.2%) is classified as error ‘`X`’ and data is deleted before kriging interpolation.
AD	`tStud`	Real	Same as `t` but takes into account `h`. Is used for outlier classification whenever `h` is available.

9.4 Spill point files

In addition to the trap output grids written to `surfaces/`, COHIBA generates a set of ASCII IRAP CLASSIC point files containing the coordinates of the spill point for each surface and the trails to termination.

9.4.1 The `cohiba_spillpoint_"surface name".irap` file

Contains the spill point coordinates (x-, y-, and z-coordinate) calculated for the corresponding surface. These files are written to `spillpointdata/`.

9.4.2 The `cohiba_terminatingtrail_"surface name".irap` file

This is the set of last points visited by the spill point detection algorithm before it terminates. The trail starts at the spill point and finish at some boundary point. These files are written to `spillpointdata/`.

A COHIBA model file reference manual

<cohiba>

Description: Wrapper for all other elements of the COHIBA model file. This is called the root element in XML jargon.
Argument: The five main elements in the model file.
Default: No default.

A.1 <project-settings>

Description: General settings such as file name conventions, measurement units, and definition of project coordinate system.
Argument: Wrapper for elements.
Default: Empty string.

A.1.1 <project-title>

Description: A COHIBA project title.
Argument: String.
Default: Empty string.

A.1.2 <project-description> (optional)

Description: Detailed description of the project.
Argument: String.
Default: Empty string.

A.1.3 <seed format = "" > (optional)

Description: Sets the random seed for the random number generator used in simulations. Seed files are imported from the directory specified in <project-directory> using the attribute value file. When running a simulation, the last seed is always written to file in the directory specified in <project-directory>. Note that the seed file is rewritten. It is also possible to set the random seed employing the option -s on the command line. E.g.: cohiba.exe -s 123 modelfile.xml
Attribute: **format** Seed format.
Attribute value: number / file
Attribute default: file
Argument: File name or a number depending on attribute value.
Default: 0 if number format is given or seed.dat if file format is given.
Example: Seed number:

```
<seed format = "number">
  123
</seed>
```

Example: Seed file:

```
<seed>
  myseed.dat
</seed>
```

A.1.4 <project-directory> (optional)

Description: A relative or absolute path to a COHIBA project. If a relative path is used, COHIBA must be executed from the directory the relative path refers to.

Argument: A valid path to a directory.
Default: Current directory (. in Linux).
Example: Relative path:

```
<project-directory>
   my-cohiba-project/
</project-directory>
```

Example: Absolute path:

```
<project-directory>
   /path/to/projects/my-cohiba-project/
</project-directory>
```

A.1.5 `<input-directory>` (optional)

Description: Relative or absolute path to all COHIBA input. The relative path is relative to the project directory.
Argument: A valid path to a directory
Default: The project directory.
Example: Relative path:

```
<input-directory>
   my-input/
</input-directory>
```

Example: Absolute path:

```
<input-directory>
   /path/to/projects/my-cohiba-project/my-input/
</input-directory>
```

Example: Absolute path can be to any location:

```
<input-directory>
   /path/to/my-input/
</input-directory>
```

A.1.6 `<output-directory>` (optional)

Description: Relative or absolute path to the COHIBA output directories. The output directory has 3 subdirectories: `logfiles/`, `surfaces/`, and `welldata/`. All these directories are created if they do not exist prior to execution of COHIBA.
Argument: A valid path to a directory.
Default: The project directory.
Example: Relative path:

```
<output-directory>
   my-output/
</output-directory>
```

Example: Absolute path:

```
<output-directory>
   /path/to/projects/my-cohiba-project/my-output/
</output-directory>
```

Example: Absolute path can be to any location:

```
<output-directory>
   /path/to/my-output/
</output-directory>
```

A.1.7 `<filename-tags>` (optional)

Description: Specification of prefix (or postfix) on output grids. For example, a depth surface called TopA with a prefix `d` will be assigned the filename `d_TopA.irap`. Note that the underscore "_" is always added before (after) the name.
Argument: Elements for each grid type.
Default: Empty — default prefix for all grid types will be used.

A.1.7.1 `<type>` (optional)

Description: Type of tag.
Argument: `prefix` / `postfix`
Default: `prefix`

A.1.7.2 `<depth>` (optional)

Description: A tag describing the output depth.
Argument: String.
Default: `d`

A.1.7.3 `<depth-trend>` (optional)

Description: A tag describing the output depth trend.
Argument: String.
Default: `dt`

A.1.7.4 `<depth-error>` (optional)

Description: A tag describing the output depth error.
Argument: String.
Default: `de`

A.1.7.5 `<depth-trend-error>` (optional)

Description: A tag describing the output depth trend error.
Argument: String.
Default: `dte`

A.1.7.6 `<depth-residual>` (optional)

Description: A tag describing the output depth residual.
Argument: String.
Default: `dr`

A.1.7.7 `<trap>` (optional)

Description: A tag describing the output trap connected to the spill point.
Argument: String.
Default: `trap`

A.1.7.8 `<thickness>` (optional)

Description: A tag describing the interval thickness.

Argument: String.
Default: `t`

A.1.7.9 `<thickness-trend>` (optional)

Description: A tag describing the interval thickness trend.
Argument: String.
Default: `tt`

A.1.7.10 `<velocity>` (optional)

Description: A tag describing the interval velocity.
Argument: String.
Default: `v`

A.1.8 `<messages>` (optional)

Description: Wrapper for elements specifying logging output levels to log file and screen (monitor). Available logging levels are:

Level	Description
0	No messages reported.
1	Errors reported.
2	Warnings reported.
3	Information messages reported. (Default)
4	Detailed information reported.
5	Very detailed information reported (mainly for debugging purposes).

The output information accumulate so that level 5 includes everything reported on level 4 and so on.
Argument: Elements specifying output to log file and screen.
Default: Void.

A.1.8.1 `<logfile>` (optional)

Description: Wrapper for elements describing the output to the COHIBA log file. The log file is written to the `logfiles/` directory, under the output-directory.
Argument: Elements specifying output to the COHIBA log file.
Default: Void.

A.1.8.1.1 `<name>` (optional)

Description: A string describing the name of the COHIBA log file.
Argument: String.
Default: `cohiba.log`

A.1.8.1.2 `<detail-level>` (optional)

Description: Wrapper for elements specifying the level of detail in the COHIBA log file. Each element can be set individually.
Argument: The elements corresponding to the different sections of the log file.
Default: Void.

`<overall>` (optional)

Description: Level of detail in output to log file. This level can be overruled within each output section mentioned below. See Section 9.2.1 for a brief explanation of each section.
Argument: 0 – 5
Default: 3

`<model-settings>` (optional)

Description: Level of detail in model settings output.
Argument: 0 – 5
Default: 3

`<data-loading>` (optional)

Description: Level of detail in reporting data loading.
Argument: 0 – 5
Default: 3

`<well-points>` (optional)

Description: Level of detail in reporting the processing of well points.
Argument: 0 – 5
Default: 3

`<well-paths>` (optional)

Description: Level of detail in reporting the processing of zone logs along well paths.
Argument: 0 – 5
Default: 3

`<trend-coefficients>` (optional)

Description: Level of detail in reporting trend coefficients.
Argument: 0 – 5
Default: 3

`<outliers>` (optional)

Description: Level of detail in reporting match between well points and trend models.
Argument: 0 – 5
Default: 3

`<well-point-conditioning>` (optional)

Description: Level of detail in reporting of errors in conditioning (kriging).
Argument: 0 – 5
Default: 3

`<well-path-conditioning>` (optional)

Description: Level of detail in reporting of errors from well points sampled along well path.
Argument: 0 – 5
Default: 3

`<post-processing>` (optional)

Description: Level of detail in report post processing. The post processing handles erosion, pinch outs, and gridding errors.
Argument: 0 – 5
Default: 3

A.1.8.2 `<screen>` (optional)

Description: Wrapper for elements describing the output to screen (monitor).
Argument: Elements specifying output to screen.
Default: Void.

A.1.8.2.1 `<detail-level>` (optional)

Description: Wrapper for elements specifying the level of detail in the output logging to screen. Each element can be set individually.
Argument: The elements corresponding to the different sections of logging.
Default: Void.

`<overall>` (optional)

Description: Level of detail in output to screen. This level can be overruled within each output section described in the logfile Section A.1.8.1.
Argument: 0 – 5
Default: 3

A.1.9 `<measurement-units>` (optional)

Description: Wrapper for specifying measurement units. Available units:

Unit	Description
m	Meter
s	Second
ms	Millisecond
m/s	Meters per Second

Depth and length units are assumed to be in meters and interval velocities are assumed to be in meters per second.
Argument: Elements for each unit type.
Default: Void.

A.1.9.1 `<time-unit>` (optional)

Description: Unit of time.
Argument: `s` / `ms`
Default: `s`

A.1.9.2 `<angle-unit>` (optional)

Description: Unit of angle. Degrees ($360°$) or radians (2π) .
Argument: `deg` / `rad`
Default: `deg`

A.1.9.3 `<two-way-time>` (optional)

Description: A flag controlling whether the input time grids are given in two-way times (TWT) or not.
Argument: `yes` / `no`
Default: `no`

A.1.10 `<output-grid format = "" >`

Description: Grid dimensions for the lateral output grid. The input grids must share these dimensions. The following grid consistency equations must be satisfied:
$x_{end} = x_{start} + x_{inc} \times (n_x - 1)$ and $y_{end} = y_{start} + y_{inc} \times (n_y - 1)$.
Attribute: **format** File format.
Attribute value: `storm` / `irapclassic`
Argument: Elements specifying the grid dimensions.
Default: no default.

A.1.10.1 `<xstart>`

Description: Minimum X value.
Argument: Real.
Default: No default.

A.1.10.2 `<xend>`

Description: Maximum X value.
Argument: Real satisfying the consistency equations.
Default: No default.

A.1.10.3 `<ystart>`

Description: Minimum Y value.
Argument: Real.
Default: No default.

A.1.10.4 `<yend>`

Description: Maximum Y value.
Argument: A real satisfying the consistency equations.
Default: No default.

A.1.10.5 `<xinc>` (optional)

Description: X increment; not obligatory if nx is given.
Argument: Positive real.
Default: No default.

A.1.10.6 `<yinc>` (optional)

Description: Y increment; not obligatory if ny is given.
Argument: Positive real.
Default: No default.

A.1.10.7 `<nx>` (optional)

Description: Number of grid nodes in X; not obligatory if xinc is given.
Argument: Positive integer.
Default: No default.

A.1.10.8 `<ny>` (optional)

Description: Number of grid nodes in Y; not obligatory if yinc is given.
Argument: Positive integer.
Default: No default.

A.2 `<modelling-settings>` (optional)

Description: Wrapper for elements determining the methods and algorithms used.
Argument: Elements specifying overall modelling settings.
Default: No default.

A.2.1 `<mode>` (optional)

Description: The default mode is to generate a set of most likely surfaces using *prediction*. Alternatively a Monte Carlo sample can be drawn using *simulation*. Prediction is typically used for well prognosis, and simulation is typically used for uncertainty studies on volumes. The *estimation* mode gives the possibility to run through the initial adaption of the trend coefficients to well points without entering the more time consuming kriging steps in the prediction or conditional simulation steps.
Argument: `prediction` / `simulation` / `estimation`
Default: `prediction`

A.2.2 `<kriging-method>` (optional)

Description: The three available kriging method to be used, see [5]. *Simple kriging* assumes no uncertainty on the trend coefficients. *Universal Kriging* assumes the trend coefficients are unknown and fits them to well data. This method fails if there are too few well data. *Bayesian Kriging* assumes the trend coefficients are partly known (expectation and standard deviation specified) and adapts the trend coefficients to well data. This method is robust and works for any number of well data. If there is a lot of well data, Bayesian kriging will be very similar to universal kriging.
Argument: `simple` / `bayesian` / `universal`
Default: `bayesian`

A.2.3 `<simulate-trend-uncertainty>` (optional)

Description: This flag has only effect in simulation mode and if simple kriging has been specified. If `yes` is specified, the trend coefficients will be drawn from the prior specification. This option can be used to specify exactly the standard deviation on the trend coefficients used during simulation.
Argument: `yes` / `no`
Default: `no`

A.2.4 `<advanced-settings>` (optional)

Description: Wrapper for elements controlling algorithms. Don't touch it if you don't mean it.
Argument: Elements controlling algorithms.
Default: Void.

A.2.4.1 `<threshold-for-conditioning-in-neighbourhood>` (optional)

Description: Controls the use of neighbourhoods in the kriging algorithm. If the number of well points is larger than this number, kriging is performed in overlapping subsets. This algorithm runs fast and gives a good approximation. If there appears rectangular patches in the output

grids, consider increasing this number.
Argument: Positive integer.
Default: `100`

A.2.4.2 `<max-obs-direct-estim-trend-coef>` (optional)

Description: When the number of well points is smaller than this threshold, the trend coefficients and their covariance matrix are estimated using the formulas

$$\widehat{\boldsymbol{\beta}}_{\text{Bayes}} = \boldsymbol{\beta}_0 + \boldsymbol{\Sigma}_0 \mathbf{F}' \left(\mathbf{K} + \mathbf{F}\, \boldsymbol{\Sigma}_0\, \mathbf{F}'\right)^{-1} (\mathbf{Z} - \mathbf{F}\, \boldsymbol{\beta}_0)$$
$$\widehat{\boldsymbol{\Sigma}}_{\text{Bayes}} = \boldsymbol{\Sigma}_0 - \boldsymbol{\Sigma}_0 \mathbf{F}' \left(\mathbf{K} + \mathbf{F}\, \boldsymbol{\Sigma}_0\, \mathbf{F}'\right)^{-1} \mathbf{F}\, \boldsymbol{\Sigma}_0$$

where $\mathbf{F}$ is the design matrix, $\mathbf{K}$ is the kriging matrix, and $\mathbf{Z}$ is the vector of well points.
These formulas are very robust for few well points, including zero well points.
Argument: Positive integer.
Default: `200`

A.2.4.3 `<max-obs-GLS-approx-trend-coef>` (optional)

Description: When the number of well points is more than `<max-obs-direct-estim-trend-coef>` and less than `<max-obs-GLS-approx-trend-coef>`, the trend coefficients and their covariance matrix are estimated using the formulas:

$$\widehat{\boldsymbol{\beta}}_{\text{Bayes}} = \widehat{\boldsymbol{\Sigma}}_{\text{Bayes}} \left(\boldsymbol{\Sigma}_0^{-1} \boldsymbol{\beta}_0 + \mathbf{F}' \mathbf{K}^{-1} \mathbf{Z}\right)$$
$$\widehat{\boldsymbol{\Sigma}}_{\text{Bayes}} = \left(\boldsymbol{\Sigma}_0^{-1} + \mathbf{F}' \mathbf{K}^{-1} \mathbf{F}\right)^{-1}$$

These formulas are equivalent to the formulas above if the kriging matrix, $\mathbf{K}$, is positive definite. These formulas are numerically better when there are many well points.
If the number of well points is larger than `<max-obs-GLS-approx-trend-coef>`, COHIBA uses the weighted least squares (WLS) approximation

$$\widehat{\boldsymbol{\beta}}_{\text{Bayes}} = \widehat{\boldsymbol{\Sigma}}_{\text{Bayes}} \left(\boldsymbol{\Sigma}_0^{-1} \boldsymbol{\beta}_0 + \mathbf{F}' \mathbf{D}^{-1} \mathbf{Z}\right)$$
$$\widehat{\boldsymbol{\Sigma}}_{\text{Bayes}} = \left(\boldsymbol{\Sigma}_0^{-1} + \mathbf{F}' \mathbf{D}^{-1} \mathbf{F}\right)^{-1}$$

where $\mathbf{D}$ is the diagonal of the kriging matrix $\mathbf{K}$.
The WLS approximation can handle a lot of well points.
Argument: Positive integer.
Default: `2000`

A.2.4.4 `<max-obs-update-trend-coef-using-well-paths>` (optional)

Description: The trend coefficients are not adjusted to fit horizontal well paths *if* the number of well points are more than this value. This saves a lot of CPU time. The idea is that the trend coefficients are estimated with sufficient accuracy when the number of well points exceed this limit.
Argument: Positive integer
Default: `200`

A.2.4.5 `<threshold-for-trace-clustering>` (optional)

Description: The constraints from each well path are grouped in correlated clusters. This element specifies the threshold for the correlations. Specifying a low correlation means that all constraints are probably grouped into one cluster. Specifying a high value means that constraints must be strongly correlated to be grouped. Ideally all constraints should be handled simultaneously, but small clusters are important to obtain fast performance.
Argument: Real between 0 and 1.
Default: `0.3`

A.2.4.6 `<threshold-for-cluster-merging>` (optional)

Description: The correlated clusters are merged if any pair of constraints have a correlation larger than this threshold. A small value will cause all clusters to be merged. A large value will avoid any merging of clusters. Some merging is needed if well paths are near.
Argument: Real between 0 and 1.
Default: `0.5`

A.2.4.7 `<post-process-erosive-and-onlapping-surfaces>` (optional)

Description: Overrides all input flags related to erosive and onlapped surfaces. Default is to truncate all surfaces by the erosive and onlapped surfaces.
Argument: `yes` / `no`
Default: `yes`

A.2.4.8 `<post-process-crossing-surfaces>` (optional)

Description: Post-process surfaces to produce a sequence of stratigraphically correct surfaces. Negative thickness will be removed and replaced by a zero thickness interval — a pinch out. This algorithm does not take into account well observations. The default is to remove all negative thicknesses.
Argument: `yes` / `no`
Default: `yes`

A.2.4.9 `<condition-to-well-paths>` (optional)

Description: Starts an algorithm that attempts to move surfaces so that they are in correspondence with the zone logs. The algorithm will sample the zone log and generate a lot of additional well points used in prediction and simulation. If speed is critical set the value to `no`.
Argument: `yes` / `no`
Default: `yes`

A.2.4.10 `<post-process-make-surfaces-interpolate-well-points>` (optional)

Description: Removes gridding errors at well points. Modify the 4 grid values closest to a well point to ensure perfect bilinear interpolation. This is only done for well points with zero measurement uncertainty. Kriging is a perfect interpolator but the gridding of the surfaces introduce a smoothing of the bulls eye at the well point.
Argument: `yes` / `no`
Default: `yes`

A.2.4.11 `<remove-extreme-values-in-velocity-trends>` (optional)

Description: Preprocess every velocity trend. Looks for extreme values outside the limit trend, which is the mean $\pm$ 2.54 standard deviations. Such extreme values are truncated to the limit value. This removes unrealistic velocity trend contributions occationally found in input maps.
Argument: `yes` / `no`
Default: `yes`

A.3 `<well-data>` (optional)

Description: Wrapper for `<well-log>`, `<well-points>`, `<values-outside-grid>` and `<well-points-to-ignore>` elements.
Argument: Any number of `<well-log>`, `<well-points>`, and `<well-points-to-ignore>` elements
Default: No default.

A.3.1 `<well-log>` (optional)

Description: A wrapper for elements describing zone logs or fault indicator logs.
Argument: Elements specifying the zone log or fault indicator log along well paths.
Default: No default.

A.3.1.1 `<zone-log-name>` (optional)

Description: Zone log identifier. A file may contain several zone logs and this element is used to select the correct one. Currently, only one zone log can be specified.
Argument: A string.
Default: Empty string.
Example: `<logname> GM2005a </logname>`
Example: `<logname> ZonesSnorre </logname>`

A.3.1.2 `<fault-log-name>` (optional)

Description: Specifies the name of the fault indicator log.
Argument: A string.
Default: Empty string.
Example: `<fault-log-name> FaultIndicator </fault-log-name>`

A.3.1.3 `<files format = "" >` (optional)

Description: The well path files to be imported. This element may be repeated in order to import different files in different locations (directories).
Attribute: **format** File format. See [6] for details on the format.
Attribute value: `rms`.
Argument: Any valid path relative to input directory. Wildcards (*) are supported.
Default: Empty string.
Example: One single file: `<files format = "rms"> mywells/31-2_K-11H.w </files>`
Example: All files in directory: `<files format = "rms"> mywells/* </files>`

A.3.1.4 `<tops-as-mean-values>` (optional)

Description: COHIBA supports two methods for extracting the well points from the zone log:

1. The mean value of the position of the two points in the zone transition. This is the default and is selected with argument `yes`. A measurement uncertainty (standard deviation)

equal to $|z1 - z2|/4$ is added to the well point. The typical logging distance is one foot which is approximately 30cm. The standard deviation will then be approximately 7.5cm.

2. The last (along the well path) point in the zone transition. This is select with argument `no`.

The method chosen affects all files imported in `<files>`.

Argument: `yes / no`
Default: `yes`

A.3.2 `<well-points>` (optional)

Description: Wrapper for `<files>` elements which specify the files containing well points. The `<well-points>` element can be ignored or be used to add additional well points to the data extracted from zone logs.
Argument: `<files>` elements.
Default: Empty string.

A.3.2.1 `<files format = "" >` (optional)

Description: The well point files to be imported. This element can be repeated in order to import different files in different locations.
Attribute: **format** File format. See Table 1 on page 36.
Attribute value: `cohiba`
Argument: Any valid path relative to input directory. Wildcards (*) are supported.
Default: Empty string.
Example: `<files format = "cohiba"> mywellpoints/* </files>`

A.3.3 `<well-points-to-ignore>` (optional)

Description: Wrapper for `<files>` elements that specify well points that should be discarded.
Argument: `<files>` elements.
Default: Empty string.

A.3.3.1 `<files format = "">` (optional)

Description: The well points that should be excluded if found in the zone logs. This element can be repeated in order to import different files in different locations.
Attribute: **format** File format. See Table 1 on page 36.
Attribute value: `cohiba`
Any valid path relative to input directory. Wildcards are supported.
Default: Empty string.
Example: `<files format = "cohiba"> ignorepoints/* </files>`

A.3.4 `<values-outside-grid>` (optional)

Description: Wrapper for `<value>` elements which specify the depth coordinate of the well point outside the grid. Each `<value>` element defines one well point.
Argument: `<value>` elements.
Default: Empty.

A.3.4.1 `<value well-name = "" surface-name = "" x = "" y = "" stddev = "">` (optional)

Description: A depth value associated to the `<well-data>` element.
Attribute: **well-name** The name of the well associated to this observation.
Attribute value: String

Attribute default: "outside-well"
Attribute: **surface-name** The name of the surface associated to this observation.
Attribute value: String.
Attribute default: Empty string.
Attribute: **x** The X coordinate
Attribute value: Real
Attribute default: Largest machine real
Attribute: **y** The Y coordinate
Attribute value: Real
Attribute default: Largest machine real
Attribute: **stddev** The standard deviation
Attribute value: Real
Attribute default: 0.0
Argument: Real.
Default: No default.
Example: `<value surface-name = top> 1105.01 </value>`

A.3.5 Example of `<well-data>` specification

```
<well-data>

  <well-log>
    <zone-log-name>       GM2005a                    </zone-log-name>
    <files format = "rms" > ../welldatas/*.rmswell </files>
  </well-log>

  <well-points>
    <files format = "cohiba" > ../welldata/synthetic_markers.dat </files>
  </well-points>

  <values-outside-grid>
    <value well-name="my-outside-well" surface-name="TopA" x="1e10" y="1e10" stddev="0.1">
      1010
    </value>
    <value surface-name = "TopB">  1040  </value>
  </values-outside-grid>

</well-data>
```

A.4 `<surfaces>`

Description: Wrapper for `<reference>` and `<surface>` elements in stratigraphical order. A valid model must contain at least one surface different from the reference surface. The surfaces must be defined in a common rectangular grid $[x_{start}, x_{end}] \times [y_{start}, y_{end}]$. The input grid must be the same as the output grid specified in the `<output-grid>` element.
Argument: A `<reference>` element and `<Surface>` elements in correct depth order.
Default: No default.

A.4.1 `<reference>` (optional)

Description: Wrapper for elements specifying the reference surface. This is typically MSL (mean sea level). There can only be one `<reference>` element.
Argument: Elements specifying the reference surface.
Default: No default.

A.4.1.1 `<name>` (optional)

Description: A reference surface name.
Argument: String.
Default: `MSL`

A.4.1.2 `<depth format = "">` (optional)

Description: A reference surface.
Attribute: **format** Input format.
Attribute value: `constant / storm / irapclassic`
Attribute default: `constant`
Argument: A real number in case of constant format or a grid with reals in the given format.
Default: `0.0`

A.4.2 `<surface>`

Description: Wrapper for elements specifying a single surface. This element is repeated for every surface (except the reference) in the model. Important: The surface elements must be in stratigraphic order.
Argument: Elements specifying the surface.
Default: No default.

A.4.2.1 `<name>` (optional)

Description: A *unique* name for the surface. The name is used for identifying data in well points files and for identifying top and base of intervals. All output surface files concerning this surface will include this name.
Argument: String.
Default: No default.

A.4.2.2 `<erosive>` (optional)

Description: A flag controlling whether the surface is erosive or not. Erosiveness for all surfaces may be turned off using the element `<post-process-erosive-and-onlapping-surfaces>`.
Argument: `yes / no`
Default: `no`

A.4.2.3 `<onlapped>` (optional)

Description: A flag controlling whether this surface is going to be onlapped by surfaces above. An onlapped surface can be eroded by a surface above. To override this flag for all surfaces use the element `<post-process-erosive-and-onlapping-surfaces>`.
Argument: `yes / no`
Default: `no`

A.4.2.4 `<spill-point>` (optional)

Description: This element triggers the detection of the spill point of the surface and the optional calculation of the corresponding trap. Wrapper for elements specifying how to handle missing codes and the starting point for searching for the spill point. The starting point can be anywhere inside the trap. If the starting point is outside the trap the algorithm will fail. It is recommended to choose a location that is known to belong to the trap such as a producing well.

Argument: Elements specifiying coordinates to the starting point and how to interpret missing codes.

Default: Void

A.4.2.4.1 `<missing-as-wall>` (optional)

Description: Determines how the spill point algorithm shall handle missing codes.
They can either be considered as a wall or as a sink. If missing codes act as a wall (`yes`), the trap can not spill into areas with missing codes and the spill point search stops when the map boundary is found. If the missing codes are considered as a sink (`no`), the spill point search algorithm stops when it encounters cells with missing codes that are in contact with the map boundary.

Argument: `yes / no`

Default: `yes`

A.4.2.4.2 `<xstart>` (optional)

Description: Start x-coordinate for the spill point searching algorithm.

Argument: Real

Default: The x-coordinate where the surface has a minimum (highest point in surface).

A.4.2.4.3 `<ystart>` (optional)

Description: Start y-coordinate for the spill point searching algorithm.

Argument: Real

Default: The y-coordinate where the surface has a minimum (highest point in surface).

A.4.2.5 `<top-of-zone>` (optional)

Description: This defines a remapping of the zone log in case the zone log contains more zones than the zones defined by the surfaces. If zone numbers increase with increasing depth, it is sufficient to take the first zone after *this surface*. For details see Section 8.5.5.

Argument: A sequence of strings.

Default: Empty if no zone log is specified, otherwise it is required.

Example: `<top-of-zone> A' B' C' </top-of-zone>`

A.4.2.6 `<travel-time>` (optional)

Description: Specification of the grid containing the travel times for the surface. If this element is used, this surface automatically becomes a seismic reflector.

Argument: Elements specifying the travel time.

Default: No default.

A.4.2.6.1 `<value format = "" >` (optional)

Description: The travel time value or grid.
Attribute: **format** Input format.
Attribute value: `constant / storm / irapclassic`
Attribute default: `constant`
Argument: File name or constant value depending on attribute value..
Default: `0.0`.

A.4.2.6.2 `<values-outside-grid >` (optional)

Description: Wrapper for `<value>` elements which specify the travel time values. Each `<value>` element is in one-to-one correspondence with one well point outside the grid in `<values-outside-grid>` of `<well-data>`.
Argument: `<value>` elements.
Default: One `<value>` with value 0.0 for each `<value>` given in `<values-outside-grid>` from `<well-data>`.

`<value>` (optional)
Description: A travel time value for a well point outside the grid.
Argument: Real.
Default: No default.
Example: `<value> 0.1 </value>`

A.4.2.6.3 `<variogram>` (optional)

Description: Wrapper for elements specifying the travel time residual. The residual models the interpretation uncertainty in the travel time. It is specified by the standard deviation and spatial correlation function. Typical values for the standard deviation is 4–20msec. two-way time.
The available correlation functions (1.0 - variogram) are found in Table A.1.
Argument: Elements specifying the travel time uncertainty.
Default: Default values for child elements.

Table A.1. Supported variogram types in COHIBA.

`<type>`	**Correlation function**
`constant`	$f(x) = 1.0$
`white noise`	$f(x) = 1$ if $x = 0$, 0 otherwise
`exponential`	$f(x) = \exp(-3x)$
`gaussian`	$f(x) = \exp(-3x^2)$
`generalized exponential`	$f(x) = exp(-3x^p)$
`spherical1`	$f(x) = 1 - x$ if $x < 1$, 0 otherwise.
`spherical2`	$f(x) = 1 - 2(x\sqrt{1-x^2} + \arcsin(x))/\pi$ if $x < 1$, 0 otherwise
`spherical`	$f(x) = 1 - x(1.5 - 0.5x^2)$ if $x < 1$, 0 otherwise
`spherical5`	$f(x) = 1 - x(1.875 - 1.25x^2 + 0.375x^4)$ if $x < 1$, 0 otherwise
`rational quadratic`	$f(x) = 1/((1.0 + scale * x^2)^p)$

`<type>`
Description: The type of variogram.
Argument: The variograms in Table A.1.

Default: `spherical`

`<power>`

Description: The variogram power. Only used for `generalized exponential` and `rational quadratic`.

Argument: Non-negative real. Can not exceed 2.0 for `generalized exponential`.

Default: `1.0`

`<range>`

Description: The variogram range.

Argument: Non-negative real.

Default: `1000.0`

`<subrange>`

Description: The variogram subrange.

Argument: Non-negative real.

Default: range.

`<azimuth>`

Description: The variogram azimuth. Angle is measured clockwise from north. Units are specified in the `<angle-unit>` element.

Argument: Real.

Default: `0.0`

`<stddev format = "" >`

Description: The standard deviation (square root of sill) value or grid.

Attribute: **format** Input format.

Attribute value: `constant / storm / irapclassic`

Attribute default: `constant`

Argument: File name or constant value depending on attribute value.

Default: `1.0`

`<values-outside-grid>` (optional)

Description: Wrapper for `<value>` elements which specify the travel time residual values. Each `<value>` element is in one-to-one correspondence with one well point outside the grid in `<values-outside-grid>` of `<well-data>`.

Argument: `<value>` elements.

Default: One `<value>` with value 0.0 for each `<value>` given in `<values-outside-grid>` from `<well-data>`.

`<value>` (optional)

Description: A travel time residual value at the well point outside the grid.

Argument: Real.

Default: No default.

Example: `<value> 0.001 </value>`

A.4.2.7 `<output>` (optional)

Description: Wrapper for elements specifying if a particular type of output grid is written to file.

Argument: Elements for each grid type.

Default: Void

A.4.2.7.1 `<depth>` (optional)

Description: The output depth surface.

Argument: `yes / no`.

Default: `yes`

A.4.2.7.2 `<depth-error>` (optional)

Description: The output depth (prediction) error surface.
Argument: `yes` / `no`.
Default: `no`

A.4.2.7.3 `<depth-trend>` (optional)

Description: The output depth trend surface.
Argument: `yes` / `no`.
Default: `no`

A.4.2.7.4 `<depth-trend-error>` (optional)

Description: The output depth trend error surface.
Argument: `yes` / `no`.
Default: `no`

A.4.2.7.5 `<depth-residual>` (optional)

Description: The output depth residual surface.
Argument: `yes` / `no`.
Default: `no`

A.4.2.7.6 `<trap>` (optional)

Description: The output trap surface associated to a spill point. This element triggers the detection of the spill point of the surface.
Argument: `yes` / `no`.
Default: `no`

A.4.3 Example of `<surface>` specification

```
<surface>
    <name>                    TopA              </name>
    <erosive>                 no                </erosive>
    <onlapped>                no                </onlapped>
    <spill-point>
      <missing-as-wall>       yes               </missing-as-wall>
      <xstart>                342292            </xstart>
      <ystart>                6426345           </ystart>
    </spill-point>
    <top-of-zone>             IsochoreA         </top-of-zone>
    <output>
      <depth>                 yes               </depth>
      <depth-trend>           yes               </depth-trend>
      <depth-error>           yes               </depth-error>
      <depth-trend-error>     yes               </depth-trend-error>
      <depth-residual>        yes               </depth-residual>
      <trap>                  yes               </trap>
    </output>
     <travel-time>
```

```
        <value format = "storm">     T_TopA.storm       </value>
         <variogram>
           <type>                  spherical          </type>
           <range>                 1000               </range>
           <subrange>              2000               </subrange>
           <azimuth>               45                 </azimuth>
           <stddev format = "storm"> dT_TopA.storm      </stddev>
         </variogram>
      </travel-time>
 </surface>
```

A.5 <interval-models>

Description: Wrapper for <interval> elements. Each <interval> element links two arbitrary surfaces by describing the thickness or the interval velocity (depth conversion) between them. The intervals usually link two adjacent surfaces but it is possible to link surfaces having several surfaces (and intervals) between them. A single surface can be top and base in several intervals so that there are multiple ways of obtaining the depth to a surface.

Argument: <interval> elements.

Default: No default.

A.5.1 <interval> (optional)

Description: Wrapper for elements describing the interval. There are two types of intervals: Thickness intervals and velocity intervals. The latter is used to describe the interval velocity between two seismic reflectors. This element is repeated for each interval model between two surfaces.

Argument: Elements describing the interval.

Default: No default.

A.5.1.1 <name> (optional)

Description: An interval name. All output concerning this interval will use this name.

Argument: String.

Default: Empty string.

A.5.1.2 <top>

Description: The surface defining the top of the interval. Top and base surfaces must be different.

Argument: An existing surface name.

Default: No default.

A.5.1.3 <base>

Description: The surface defining the base of the interval. Top and base surfaces must be different.

Argument: An existing surface name.

Default: No default.

A.5.1.4 <interval-type>

Description: For thickness intervals, the trends describe the thickness, and for velocity intervals, the trends describe interval velocity. Velocity intervals require that the travel times are available for the top and base surfaces of the interval.

Argument: `thickness / velocity`.
Default: No default.

A.5.1.5 `<trend>` (optional)

Description: Wrapper for a single trend map and trend coefficient. An interval (thickness or velocity) may contain a sum of several trend maps so this element may be repeated any number of times.
Argument: Elements describing a trend map and its trend coefficient.
Default: Void.

A.5.1.5.1 `<coefficient-mean>`

Description: Prior mean value (expectation) of trend coefficient. The trend coefficient is multiplied by the value specified in `<value format = "" >` to obtain the trend value.
Argument: Real.
Default: `1.0`

A.5.1.5.2 `<coefficient-stddev>`

Description: Prior uncertainty (standard deviation) of trend coefficient.
Argument: Non-negative real.
Default: `0.0`

A.5.1.5.3 `<value format = "" >`

Description: A trend map. This map is multiplied by its corresponding trend coefficient.
Attribute: **format** Input format.
Attribute value: `constant / storm / irapclassic`
Attribute default: `constant`
Argument: File name or constant real value.
Default: `1.0`

A.5.1.5.4 `<values-outside-grid>` (optional)

Description: Wrapper for `<value>` elements which specify the trend values. Each `<value>` element is in one-to-one correspondence with one well point outside the grid in `<values-outside-grid>` of `<well-data>`.
Argument: `<value>` elements.
Default: One `<value>` with value 0.0 for each `<value>` given in `<values-outside-grid>` from `<well-data>`.

`<value>` (optional)

Description: A trend value associated to the well point outside the grid.
Argument: Real.
Default: No default.
Example: `<value> 1105.01 </value>`

A.5.1.6 `<correlations>` (optional)

Description: Correlations between trend coefficients.

Argument: A sequence of arguments <c*p*-*q*>, where p and q (p different from q) refers to the order of the trend coefficients within this <interval> element. Make sure that the values entered define a proper correlation matrix, that is, a symmetric positive definite (SPD) matrix.

Default: 0.0; no correlations.

Example: Correlation matrix for 3 trend coefficients:

```
<correlations>
  <c1-2> 0.01  </c1-2>
  <c1-3> 0.03  </c1-3>
  <c2-3> 0.002 </c2-3>
</correlations>
```

A.5.1.6.1 <c*p*-*q*> (optional)

Description: Prior correlations between trend coefficients p and q. The numbering is according to the order of appearance in the list of trends, starting at 1. Indices exceeding the maximum number of trends will be ignored.

Argument: Real between -1 and 1.

Default: 0.0

A.5.1.7 <variogram>

Description: Wrapper for elements specifying the interval thickness or interval velocity residual. The residual models the unknown difference between the true interval thickness/velocity and interval thickness/velocity trend. It is specified by the standard deviation and spatial correlation function.

Argument: Elements specifying the interval thickness or interval velocity residual. See the <variogram> element under the <surface> element for details.

Default: No default.

A.5.1.8 <output> (optional)

Description: Wrapper for elements specifying if a grid should be written to file.

Argument: Elements for each grid type.

Default: Void

A.5.1.8.1 <thickness> (optional)

Description: The thickness of the interval, i.e., the difference between top and base depth surfaces.

Argument: yes / no.

Default: no.

A.5.1.8.2 <thickness-trend> (optional)

Description: The thickness trend of the interval, i.e., the difference between the top trend and the base trend surfaces.

Argument: yes / no.

Default: no.

A.5.1.8.3 <velocity> (optional)

Description: The output interval velocity, i.e., the interval thickness divided by the time difference.
Argument: `yes` / `no`.
Default: `no`.

A.5.2 Example of `<interval>` specification

```
<interval>
    <name>                      MSL-to-TopA          </name>
    <top>                       MSL                  </top>
    <base>                      TopA                 </base>
    <interval-type>             velocity             </interval-type>
    <trend>
      <coefficient-mean>        2500                 </coefficient-mean>
      <coefficient-stddev>       200                 </coefficient-stddev>
      <value format = "constant"> 1                    </value>
    </trend>
    <trend>
      <coefficient-mean>        25.0                 </coefficient-mean>
      <coefficient-stddev>      12.5                 </coefficient-stddev>
      <value format = "storm"> Vcorr_Overburden.storm  </value>
    </trend>
    <correlations>
     <c1-2> 0.1 </c1-2>
    </correlations>
    <variogram>
      <type>                    spherical            </type>
      <range>                   1000                 </range>
      <subrange>                2000                 </subrange>
      <azimuth>                 45                   </azimuth>
      <stddev format = "storm">   dV_Overburden.storm  </stddev>
    </variogram>
    <output>
      <thickness>               yes                   </thickness>
      <thickness-trend>         yes                   </thickness-trend>
      <velocity>                yes                   </velocity>
    </output>
</interval>
```

B COHIBA model file example

```
<cohiba>
<project-settings>
  <project-title>
   Synthetic
   </project-title>
  <project-description>
   This example used non-constant surfaces and surfaces without UNDEFs
   </project-description>
  <seed> seed.dat  </seed>
  <project-directory>             example         </project-directory>
  <input-directory>               input/surfaces  </input-directory>
  <output-directory>              output          </output-directory>

  <filename-tags type="prefix" >
    <depth>                       d       </depth>
    <depth-trend>                 dt      </depth-trend>
    <depth-error>                 de      </depth-error>
    <depth-trend-error>           dte     </depth-trend-error>
    <depth-residual>              dr      </depth-residual>
    <trap>                        trap    </trap>
    <thickness>                   t       </thickness>
    <thickness-trend>             tt      </thickness-trend>
    <thickness-residual>          tr      </thickness-residual>
  </filename-tags>
  <messages>
    <logfile>
      <name>                      cohiba.log  </name>
      <detail-level>
      <overall>                   4       </overall>
      <model-settings>            0       </model-settings>
      <data-loading>              0       </data-loading>
      <well-points>               0       </well-points>
      <well-paths>                0       </well-paths>
      <trend-coefficients>        5       </trend-coefficients>
      <outliers>                  0       </outliers>
      <well-point-conditioning>   5       </well-point-conditioning>
      <well-path-conditioning>    5       </well-path-conditioning>
      <post-processing>           5       </post-processing>
      </detail-level>
    </logfile>
    <screen>
      <detail-level>
      <overall>                   5       </overall>
      <model-settings>            0       </model-settings>
      <data-loading>              0       </data-loading>
      <well-points>               0       </well-points>
```

```
        <well-paths>                4       </well-paths>
        <trend-coefficients>        5       </trend-coefficients>
        <outliers>                  0       </outliers>
        <well-point-conditioning>   4       </well-point-conditioning>
        <well-path-conditioning>    4       </well-path-conditioning>
        <post-processing>           4       </post-processing>
        </detail-level>
      </screen>
    </messages>
    <measurement-units>
      <time-unit>                   s       </time-unit>
      <angle-unit>                  deg     </angle-unit>
      <two-way-time>                no      </two-way-time>
    </measurement-units>
    <output-grid format="irapclassic">
      <xstart>                       0.0    </xstart>
      <xend>                      4800.0    </xend>
      <ystart>                       0.0    </ystart>
      <yend>                      3800.0    </yend>
      <xinc>                          25    </xinc>
      <yinc>                          25    </yinc>
      <nx>                           193    </nx>
      <ny>                           153    </ny>
      <azimuth>                      0.0    </azimuth>
    </output-grid>
  </project-settings>

  <modelling-settings>
    <mode>                        prediction </mode>
    <kriging-method>              bayesian   </kriging-method>
    <simulate-trend-uncertainty>  no         </simulate-trend-uncertainty>
    <advanced-settings>
       <condition-to-well-paths>        no   </condition-to-well-paths>
       <post-process-crossing-surfaces> yes  </post-process-crossing-surfaces>
    </advanced-settings>
  </modelling-settings>

  <well-data>
    <well-points>
      <files format="cohiba" > ../welldata/synthetic_markers.dat </files>
    </well-points>
  </well-data>

  <surfaces>
<!--Surfaces must be listed in stratigraphic order -->
    <reference>
      <name>MSL</name>
      <depth format="constant" >  0                 </depth>
    </reference>
```

```
<surface>
  <name>                      TopA              </name>
  <erosive>                   no                </erosive>
  <onlapped>                  no                </onlapped>
  <top-of-zone>               IsochoreA         </top-of-zone>
  <output>
    <depth>                   yes               </depth>
    <depth-trend>             yes               </depth-trend>
    <depth-error>             yes               </depth-error>
    <depth-trend-error>       yes               </depth-trend-error>
    <depth-residual>          yes               </depth-residual>
    <trap>                    yes               </trap>
  </output>
   <spill-point>
     <missing-as-wall>         no      </missing-as-wall>
     <xstart>                  2000    </xstart>
     <ystart>                  2500    </ystart>
  </spill-point>
  <travel-time>
    <value format="storm">    T_TopA.storm      </value>
    <variogram>
      <type>                  spherical         </type>
      <range>                 1000              </range>
      <subrange>              2000              </subrange>
      <azimuth>               30                </azimuth>
      <stddev format="storm"> dT_TopA.storm     </stddev>
    </variogram>
  </travel-time>
</surface>

<surface>
  <name>                      TopB              </name>
  <erosive>                   no                </erosive>
  <onlapped>                  no                </onlapped>
  <top-of-zone>               IsochoreB         </top-of-zone>
  <output>
    <depth>                   yes               </depth>
    <depth-trend>             yes               </depth-trend>
    <depth-error>             yes               </depth-error>
    <depth-trend-error>       yes               </depth-trend-error>
    <depth-residual>          yes               </depth-residual>
  </output>
  <spill-point>
     <missing-as-wall>        yes               </missing-as-wall>
   </spill-point>

</surface>
```

```
  <surface>
    <name>                      TopC              </name>
    <erosive>                   no                </erosive>
    <onlapped>                  no                </onlapped>
    <top-of-zone>               UNDEF             </top-of-zone>
    <output>
      <depth>                   yes               </depth>
      <depth-trend>             yes               </depth-trend>
      <depth-error>             yes               </depth-error>
      <depth-trend-error>       yes               </depth-trend-error>
      <depth-residual>          yes               </depth-residual>
    </output>
    <travel-time>
      <value format="storm">    T_TopC.storm      </value>
      <variogram>
        <type>                  spherical         </type>
        <range>                 1000              </range>
        <subrange>              2000              </subrange>
        <azimuth>               45                </azimuth>
        <stddev format="constant"> 0.001          </stddev>
      </variogram>
    </travel-time>
  </surface>
</surfaces>

<interval-models>
  <interval>
    <name>                      MSL-to-TopA       </name>
    <top>                       MSL               </top>
    <base>                      TopA              </base>
    <interval-type>             velocity          </interval-type>
    <trend>
      <coefficient-mean>        2500              </coefficient-mean>
      <coefficient-stddev>       200              </coefficient-stddev>
      <value format="constant"> 1                 </value>
    </trend>
    <trend>
      <coefficient-mean>        25.0              </coefficient-mean>
      <coefficient-stddev>      12.5              </coefficient-stddev>
      <value format="storm"> Vcorr_Overburden.storm</value>
    </trend>
    <correlations>
     <c1-2> 0.1 </c1-2>
    </correlations>
    <variogram>
      <type>                    spherical         </type>
      <range>                   1000              </range>
      <subrange>                2000              </subrange>
      <azimuth>                 45               </azimuth>
```

```
    <stddev format="storm">   dV_Overburden.storm</stddev>
  </variogram>
  <output>
    <thickness>               yes                 </thickness>
    <thickness-trend>         yes                 </thickness-trend>
    <thickness-residual>      no                  </thickness-residual>
  </output>
</interval>

<interval>
  <name>                      TopA-to-TopB        </name>
  <top>                       TopA                </top>
  <base>                      TopB                </base>
  <interval-type>             thickness           </interval-type>
  <trend>
    <coefficient-mean>        1                   </coefficient-mean>
    <coefficient-stddev>      0.2                 </coefficient-stddev>
    <value format="storm">    S_IsochoreA.storm   </value>
  </trend>
  <variogram>
    <type>                    spherical           </type>
    <range>                   1000                </range>
    <subrange>                2000                </subrange>
    <azimuth>                 60                  </azimuth>
    <stddev format="storm">   dS_IsochoreA.storm  </stddev>
  </variogram>
  <output>
    <thickness>               no                  </thickness>
    <thickness-trend>         no                  </thickness-trend>
    <thickness-residual>      no                  </thickness-residual>
  </output>
</interval>

<interval>
  <name>                      TopB-to-TopC        </name>
  <top>                       TopB                </top>
  <base>                      TopC                </base>
  <interval-type>             thickness           </interval-type>
  <trend>
    <coefficient-mean>        1                   </coefficient-mean>
    <coefficient-stddev>      0.2                 </coefficient-stddev>
    <value format="storm">    S_IsochoreB.storm   </value>
  </trend>
  <variogram>
    <type>                    spherical           </type>
    <range>                   1000                </range>
    <subrange>                2000                </subrange>
    <azimuth>                 60                  </azimuth>
    <stddev format="storm">   dS_IsochoreB.storm  </stddev>
```

```
      </variogram>
      <output>
        <thickness>               no                </thickness>
        <thickness-trend>         no                </thickness-trend>
        <thickness-residual>      no                </thickness-residual>
      </output>
    </interval>

    <interval>
      <name>                      TopA-to-TopC      </name>
      <top>                       TopA              </top>
      <base>                      TopC              </base>
      <interval-type>             velocity          </interval-type>
      <trend>
        <coefficient-mean>        1                 </coefficient-mean>
        <coefficient-stddev>      0.2               </coefficient-stddev>
        <value format="constant"> 2500              </value>
      </trend>
      <variogram>
        <type>                    spherical         </type>
        <range>                   1000              </range>
        <subrange>                2000              </subrange>
        <azimuth>                 120               </azimuth>
        <stddev format="constant"> 75               </stddev>
      </variogram>
      <output>
        <thickness>               no                </thickness>
        <thickness-trend>         no                </thickness-trend>
        <thickness-residual>      no                </thickness-residual>
      </output>
    </interval>

  </interval-models>

</cohiba>
```

C Release notes

The small keys, e.g. COH-49, are links to the COHIBA project management system called JIRA. Note that the links might require access permission.

C.1 Changes from COHIBA Version 1.4.0 to Version 1.4.1

New features:

- The extreme velocity contrasts truncation is changed into an option, where the default does the truncation, but can be switched off. This option is modified in the element `<remove-extreme-values-in-velocity-trends>`. COH-9

Bug fixes:

- Error in the calculation of synthetic well points when conditioning to zone logs. COH-66

C.2 Changes from COHIBA Version 1.3.1 to Version 1.4.0

New features:

- Automatic truncation of extreme velocity contrasts. Models with large negative interval velocity contrasts, that is, a large interval velocity decrease when going from an interval into a deeper interval might give undesired results (crossing surfaces etc.). This typically happens in thin zones where the input interval velocities have been obtained by dividing interval thickness by interval travel time. To remove this problem, COHIBA now truncates velocity trends outside 2.54 std.dev of each velocity trend. COH-9

Changes:

- Significant improvement of speed when conditiong to horizontal well paths. There are several major modifications of the Data Augmentation Algorithm (DAA):
  1. The trend coefficients are no longer modified by the horizontal well paths *if* the number of well points are more than 200 (default). This can be modified by the element `<max-obs-update-trend-coef-using-well-paths>`. COH-14
  2. Reuse of temporary matrices in the DAA. COH-49
  3. Splitting the inequality constraints into smaller dependent clusters. Previous COHIBA versions treated each well indpendently. Now, each well is split into small clusters when possible. This makes it simpler and faster to draw candidates in the DAA. The inequality constraints are split into separate clusters if they are less correlated than a specified threshold. This threshold is specified by the element `<threshold-for-trace-clustering>`. The current default value is 0.30. Choosing 0.0 gives the old solution where all constraints in the well will belong to the same cluster. COH-16
  4. Use a local set of well points in the DAA algorithm rather than *all* well points. The current solution is to use all well points belonging to the wells associated with the cluster. COH-16

  The computational time has dropped from 5 hours to 15 minutes in one large case. The improvement in efficiency is larger for cases with lots of well points.
- More robust conditioning to horizontal wells. Previously each well was treated independently. This caused an error in the rare occations where two well paths were extremly close. Now, the clusters from different wells are merged if they are more correlated than

a threshold given by the element `<threshold-for-cluster-merging>`. The current default is 0.5. Choosing 0.0 is likely to cause that all clusters are merged. COH-17

Bug fixes:

- Error message: Kriging matrix with inequality points is not well defined. Reported by StatoilHydro. This problem is solved. COH-48
- Corrupt result for some random seeds. Caused by an error in the random number generator. Reported by StatoilHydro. This problem is solved. COH-47
- Input RMS well files with Windows linefeed will now run correctly on Linux. COH-60
- If COHIBA cannot calculate trend coefficients for universal kriging, COHIBA will use Bayesian estimation. However, COHIBA still used the very unstable universal kriging formulas to perform the surface prediction (kriging). This inconsistency is now removed so an Bayesian approach is used for trend coeficient estimation and the kriging. COH-59
- Depth trend errors are now correctly computed if outside points are close to (within the range) the output grid. COH-58
- The severe outliers that should be removed from all calculations are not reported in `cohiba_wellpoints.dat` anymore. COH-57
- Well points outside the grid are no longer incorrectly labeled as deleted in `welldata.log`. COH-56

C.3 Changes from COHIBA Version 1.3.0 to Version 1.3.1

Bug fixes:

- Model file parsing: Travel time variogram not optional if travel time is specified.
- Output: Velocity of intervals not exported although it is asked for.
- Spill-point detection: Trap prefix not ok in the filename-tags, it is not set to "trap".
- Spill-point detection: Error in storm-filename suffix on terminating trail. Should be .irap (irapclassic ascii)

C.4 Changes from COHIBA Version 1.2.0 to Version 1.3.0

New features:

- Robust estimation of trends. Severe outliers are removed from the trend estimation.

Changes:

- Improved sampling of well path conditioning. If two close points from different wells are detected, one is removed.

C.5 Changes from COHIBA Version 1.1.0 to Version 1.2.0

New features:

- Fault indicator log may now be used.
- New element `<max-obs-direct-estim-trend-coef>` introduced.
- New element `<max-obs-GLS-approx-trend-coef>` introduced.
- New element `<values-outside-grid>` and `<value>` introduced.

Changes:

- Element <zonelog> changed to <well-log>.
- Element <logname> changed to <zone-log-name>.
- Element <fault-logname> changed to <fault-log-name>.
- Element <surface-points> changed to <well-points>.
- Element <ignore-points> changed to <well-points-to-ignore>.
- Improvements have been made to the well path conditioning option. A pair of help points above and below a well path is now drawn simultaneously from a binormal distribution.

C.6 Changes from COHIBA Version 1.0.1 to Version 1.1.0

New features:

- Spill point detection. New elements under the <surface> element were added:
  1. <spill-point> and optional sub-elements <missing-as-wall>, <xstart>, and <ystart>.
  2. <trap> as sub-element of <output>.
  3. <trap> as sub-element of <filename-tags>.

  Spill point detection can also be applied directly from the command line to an imported surface.

Changes:

- Improvements have been made to the well path conditioning option.

C.7 Changes from COHIBA Version 1.0.0 to Version 1.0.1

New features:

- Correction for gridding error added. Can be switched on/off using new element <post-process-make-surfaces-interpolate-well-points>.
- The element <seed format = "" > now allows a filename as value. Seed is read from the file before simulation and the final seed is written to the file after simulation.

Changes:

- Element <welldata> changed to <well-data>.
- Element <wellpoint> changed to <well-point>.
- Element <wellpoint-conditioning> changed to <well-point-conditioning>.
- Element <wellpath> changed to <well-path>.
- Element <wellpath-conditioning> changed to <well-path-conditioning>.
- Element <condition-to-wellpaths> changed to <condition-to-well-paths>.
- Element <trend-parameters> changed to <trend-coefficients>.
- Element <kriging-in-segments-for-more-observations-than> changed to <threshold-for-conditioning-in-neighbourhood>.
- Default for <depth> changed from no to yes.
- Default for <depth-trend> changed from no to yes.

D Known issues

D.1 Missing codes at zone transitions

Problem: Some software (databases?) produce missing codes at zone transitions. COHIBA will not recognise these as zone transitions and potential well points will be missed.

Solution: Introduce a tolerance so that a zone transition is recognised even if there is a small section of the well path with missing codes. COH-11

D.2 Conditioning to well paths using universal kriging

Problem: Equations not implemented.

Solution: Straight forward implementation. COH-23

D.3 Faults

Problem: Faults are not handled explicitely.

Solution: Integration with e.g. Irap RMS will take care of this. COH-35, COH-36, COH-37, COH-38

D.4 Pinch out well points

Problem: Well points read from file are deleted if squeezed between other well points in the pinch out. This works in most cases but there will be situations where the intermediate surfaces might end up above or below the well point.

Solution: Only remove well points if they have correlation 1 to a well point that is kept. This will guarantee consistency with all observed well points. COH-10

D.5 Improve the calculation of synthetic well points

Problem: The Data Augmentation Algorithm might fail to include the proper correlations between the location of the constraints. This will in turn cause numerical instabilities in the equations calculating the synthetic well points used in the kriging.

Solution: Implement a better algorithm for drawing truncated multi-Gaussian variables. COH-45

D.6 Bugs

Wrong sample point for constraint along a well path chosen close to a pinch out. COH-13

Add tail-points to well sampling. This ensures that the whole trajectory is taken care of. COH-21

References

[1] Abrahamsen, P. (1993). Bayesian kriging for seismic depth conversion of a multi-layer reservoir. In Soares, A., editor, *Geostatistics Tróia '92*, pages 385–398, Dordrecht. proc. '4th Inter. Geostat. Congr.', Tróia Portugal, 1992, Kluwer Academic Publ. 16

[2] Abrahamsen, P. (2005). Combining methods for subsurface prediction. In Leuangthong, O. and Deutsch, C. T., editors, *Geostatistics Banff 2004*, volume 2, pages 601–610, Dordrecht. proc. '7th Inter. Geostat. Congr.', Banff, Canada 2004, Springer-Verlag Inc. 16

[3] Abrahamsen, P. and Benth, F. E. (2001). Kriging with inequality constraints. *Math. Geol.*, 33(6):719–744. 16, 43

[4] Abrahamsen, P., Hauge, R., Heggland, K., and Mostad, P. (2000). Estimation of gross rock volume of filled geological structures with uncertainty measures. *SPE Reservoir Eval. & Eng.*, 3(4):304–309. 16

[5] Abrahamsen, P., Vazquez, A. A., Dahle, P., and Skorstad, A. (2008). Cohiba — technical documentation. NR-note SAND/03/2008, Norwegian Computing Center, Oslo. 53, 65

[6] ROXAR ASA, Stavanger, Norway (2007). User Guide Irap RMS. http://www.roxar.com/. 34, 35, 68

Index