

R-04-55

Rock Visualization System

Technical description (RVS v.3.5)

P Curtis, M Elfström, I Markström, FB Engineering

March 2004

Svensk Kärnbränslehantering AB

Swedish Nuclear Fuel
and Waste Management Co
Box 5864

SE-102 40 Stockholm Sweden

Tel 08-459 84 00
+46 8 459 84 00

Fax 08-661 57 19
+46 8 661 57 19



ISSN 1402-3091

SKB Rapport R-04-55

Rock Visualization System

Technical description (RVS v.3.5)

P Curtis, M Elfström, I Markström, FB Engineering

March 2004

This report concerns a study which was conducted for SKB. The conclusions and viewpoints presented in the report are those of the authors and do not necessarily coincide with those of the client.

A pdf version of this document can be downloaded from www.skb.se

Summary

The Rock Visualization System (RVS) has been developed by SKB for use in visualizing geological and engineering data in 3D.

The purpose of this report is to provide a technical description of RVS aimed at potential program users and interested parties as well as fulfilling the function of a more general RVS reference that can be cited when writing other technical reports.

It is a description of RVS version 3.5. Updated versions of this report or addenda will be made available following further development of RVS and the release of subsequent versions of the program.

The report covers the following main items:

Technical description of the program with illustrations and examples

Limitations of the program and of functionality

For most RVS functions step-by-step tutorials are available describing how a particular function can be used to carryout a specific task. A complete set of updated tutorials is issued with each new version release of the RVS program. However, the tutorials do not cover all the possible uses of all the individual functions but rather give an overall view of their functionality. A detailed description of every RVS function and how it can be used is included in the RVS online Help system.

Contents

1	Introduction	7
2	Setup	9
3	Overview of the modelling process	11
4	Working with models	15
4.1	Creating a model	15
4.2	Model versions	16
4.3	Coordinate systems	16
4.4	Background models	17
5	Data input	19
5.1	Data input from SICADA	19
5.2	Data input from external sources	22
5.2.1	Primary data	22
5.2.2	Existing geometries	23
5.2.3	Attachments	23
6	Visualizing data in a model	25
6.1	Working with boreholes	25
6.1.1	Borehole data	25
6.1.2	Planning future borehole investigations	26
6.2	Constructing and visualising surfaces	26
6.2.1	Topographic surfaces	29
6.2.2	Seismic reflectors	30
6.3	Modelling lenses	31
7	Managing the model contents	33
7.1	Managing objects	33
7.2	Managing local data and parameter tables	34
8	Block modelling	35
9	Managing models between different RVS and Microstation® installations	37
10	Managing models within a project team and between different software systems	39
	References	41
	Appendix A	43

1 Introduction

The Rock Visualization System (RVS) has been developed by SKB for use in visualizing geological and engineering data. It aims to assist in the interpretation of the geological environment by the construction of 3D structural geological models. These models will supply a framework for the creation of integrated models covering all science areas, ultimately leading to the selection and design of the final repository system.

The purpose of this report is to provide a technical description of RVS aimed at potential program users and interested parties as well as fulfilling the function of a more general RVS reference that can be cited when writing other technical reports.

RVS has been under development since 1994. The current version of RVS, Version 3.5, is based on MicroStation V8[®] and the database MS/Access 97/2000[®]. The system is certified for Windows NT4 Workstation[®], Windows 2000 Professional[®] and Windows XP Professional[®].

The report presents a description of RVS Version 3.5, released in February 2004. Addenda or updated versions of this report will be made available following further development of RVS and the release of subsequent versions of the program.

RVS has been developed for interpreting data collected during the Swedish radioactive waste disposal programme, including data from the Äspö Hard Rock Laboratory and from the potential disposal sites. Its focus is on the construction of structural geological models based on borehole and various forms of surface mapping data in crystalline rocks. Much of the necessary input to a model is in the form of primary data and the system has therefore been integrated with SKB's geological database SICADA. The acronym SICADA stands for Site ChAracterization DAtabase, a relational database management system developed by SKB for the storage and maintenance of data collected during the site investigations. Further details about SICADA can be found in section 6.1 of this report.

Models created in RVS can be published or otherwise distributed and exchanged with other RVS users through SIMONE, an Internet based data repository service. This service is available for designated RVS users.

2 Setup

RVS Version 3.5 is based on MicroStation V8® and the database MS/Access 97/2000®. The system is certified for Windows NT4 Workstation®, Windows 2000 Professional® and Windows XP Professional®.

RVS has been designed to enable a close integration with SKB's investigation database SICADA. Since all raw data originating from the many investigation programs is carefully quality controlled before it enters SICADA, the close integration between the two systems ensures that this quality is maintained by automating data processing and transfer. This close integration ensures an acceptable level of data quality as well as ensuring a high level of traceability.



MicroStation® v8



The other major data input source to RVS is from SKB's GIS database supplied via ArcSDE® (Spatial Data Engine produced by ESRI® GIS software). This type of data differs from SICADA data since it has already been processed and interpreted within a commercial GIS system and is therefore in a proprietary file format, which requires conversion before it can be imported and used by RVS. The necessary manipulation, conversion and import process has yet to be standardized and streamlined. Until this occurs the operation cannot be said to follow the same strict quality routines as have been established for SICADA imports.

Normal CAD work in MicroStation® is based on design files and levels, while work in RVS is object based. Drawing elements in a normal MicroStation® design file have no intelligence in that they are simple graphical objects with specific graphical attributes that control their appearance. However, objects in an RVS-model often consist of a number of graphical elements, which together build a unit, linked to a local database with flexible tools for controlling their appearance. They are saved with a name in a logical structure depending on what they represent. All manipulative actions such as viewing selection and change of properties are done on the object level.

All work in RVS is done within models. A model can be thought of as a collection of certain data for a particular site, and for a special modelling purpose. Different models can interact so that you can be working in one model, while using other model(s), possibly from other users, as background models.

It should be noted that the term Model is used by Bentley Systems in MicroStation V8® with quite a different meaning, which should not be confused with the RVS Model concept. A model in MicroStation V8® is a collection of design elements that share a common coordinate space, similar to the familiar design plane or cube in earlier versions of MicroStation® and are comparable to the multiple worksheets in a Microsoft Excel® file /MicroStation® V8, 2003/. All references to Model in this report refer to the RVS meaning of the term.

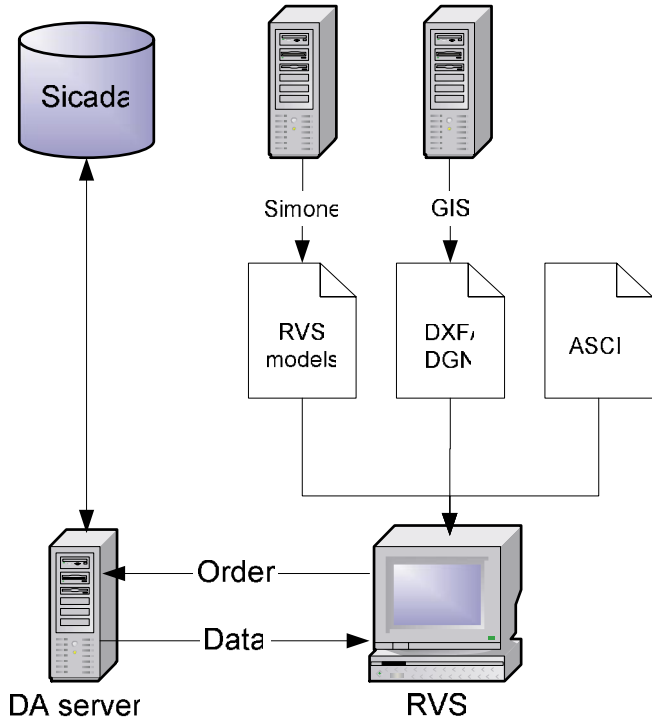


Figure 2-1. Transfer of data to RVS.

3 Overview of the modelling process

This section of the report outlines a general modelling procedure following an imaginary investigation of a rock volume. The underlying methodology used for constructing, visualising and presenting 3-dimensional models, based on data from the site investigations is described in /Munier et al, 2003; Munier and Hermanson, 2001/. The graphics shown have been based on archive data from SKB's underground hard rock laboratory at Äspö and are not related to any active SKB project. Further details of the individual modelling steps and the tools used are presented in subsequent sections of the report.

The RVS modelling process begins with the definition of the model block volume in 3D. The black cube in the figure below represents the RVS model boundary.

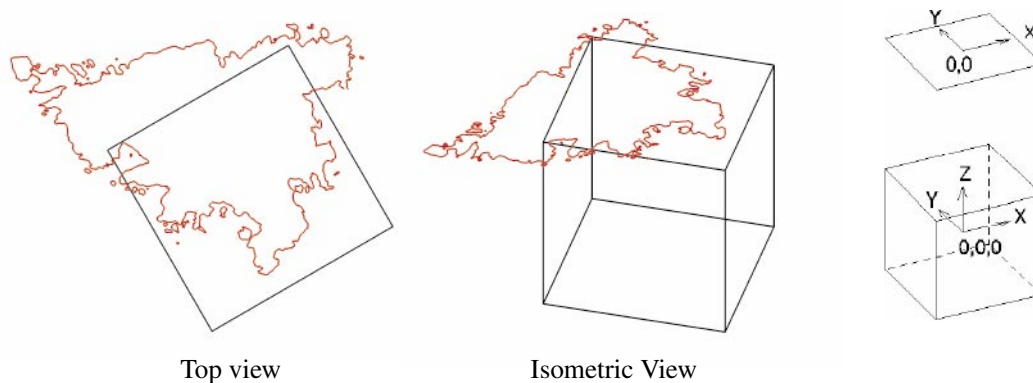


Figure 3-1. Defining the model volume.

A model may be built up initially from the results of a desk study and preliminary fieldwork /Andersson et al, 2002/. The desk study involves assembling all the relevant basic data available, such as topographic and geological maps and remote sensing data. The initial fieldwork may include basic geological mapping and ground control. Such key maps and data are input into a GIS system and then reprocessed and attached to the RVS model.

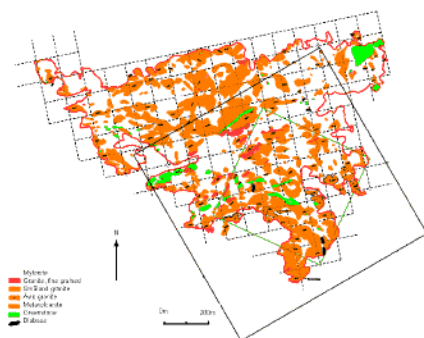


Figure 3-2. Attaching reference material. Here is shown the model volume with an attached rocktype map, based on field mapping. The results from lineament studies and geophysical mapping may be attached in the same way.

The interpretation of these data may lead to the proposal of a number of surface traces (2D) representing various deformation zones; shown by the red lines in the figures below. Where evidence is available, such surface traces may be given tentative orientations and visualized in 3D space. Such a zone can be visualized using the RVS *Model surface* tool, where a surface trace can be drawn or fixed by specifying a strike and then given a value for dip. In addition, a thickness can be specified so the zone appears as a 3D volume rather than a plane. Alternatively, the coordinates of a number of known points can be input to a *Point table* and RVS can be asked to interpolate a *best-fit* plane for the data.

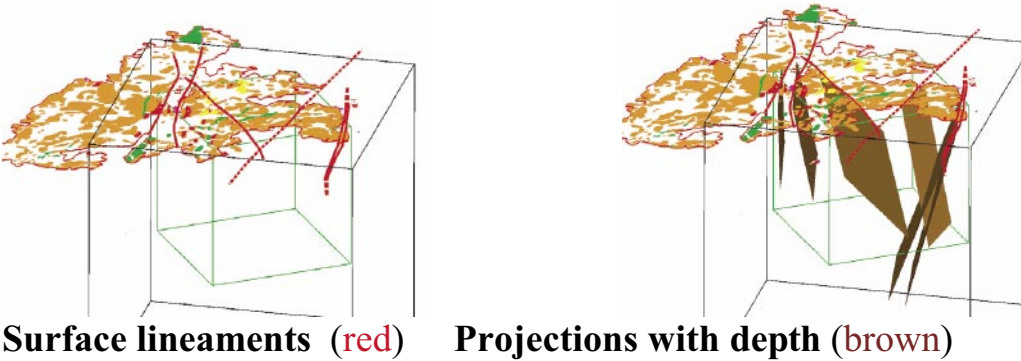
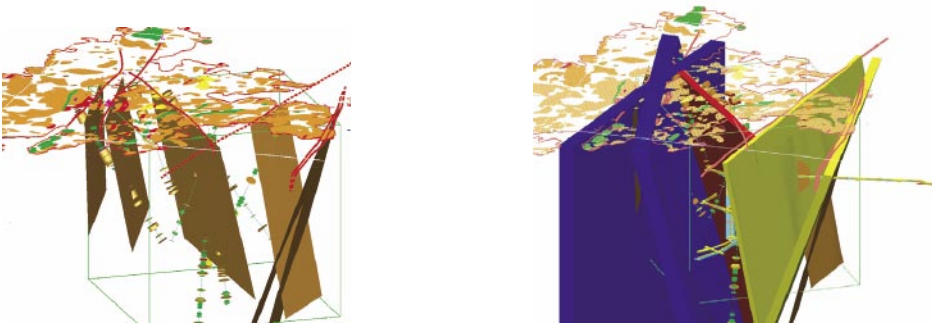


Figure 3-3. Projecting surface interpretations with depth.

Direct physical evidence of the prevailing rock conditions at depth will be initially obtained by drilling. The geometry of the boreholes, along with results from drill-core mapping and in situ testing, are assembled in the SICADA database and retrieved for modelling by RVS. RVS has various tools to aid working with borehole data.

Data is ordered from the SICADA database and imported into the local RVS database for modelling. As well as modelling the information already available, RVS can help with the planning of further drilling operations with various tools, such as *analyse borehole intersections* and *borehole planning*. As the names suggest, these tools allow the operator to input alternative drilling geometries and run a check on their suitability.

As more information becomes available the geometry of the various interpreted structures may be modified with adjustments to their persistence, alignment and width to better fit with the perceived reality.



Additional borehole data

Modified zone geometries

Figure 3-4. Modifying zone geometries after new information.

The model volume may be adjusted at some stage to focus on a smaller rock volume for more intensive study as represented by the smaller green cube in the figure below.

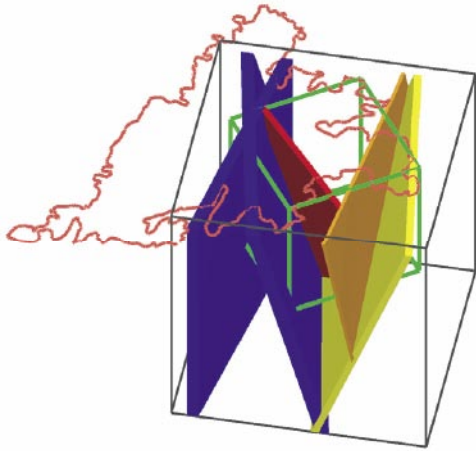


Figure 3-5. *Modifying the model volume and focus of studies.*

The network of intersecting interpreted deformation zones defines a series of discrete blocks. Other interpreted features that, along with the deformation zones, take part in the demarcation of blocks, which are not shown here, are the interpreted rock type boundaries from the modelling of the lithology in 3D. Each block is given its own identity within the model and assigned to a particular block group, depending on the type of the bounding surfaces e.g. deformation zone or rock unit boundary. All such groups and contained blocks are listed in the RVS *Object manager*. From all the obtained investigation data and interpretations, each block group is characterised by the assignment of characteristic values for the various parameters; rock-type, strength, permeability etc. In this way a description of the entire rock volume under study is generated.

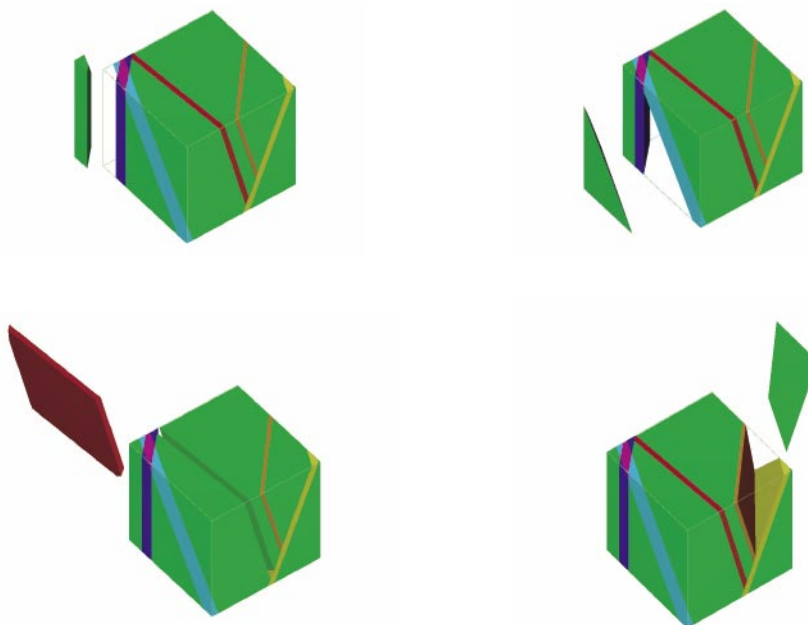


Figure 3-6. *Block modelling – defining block geometries.*

Even during the initial investigation stages of a project, any available preliminary design layout of a planned excavation can be attached to the model. In this way the design process can be dynamically interactive with the ongoing investigation and modelling work of the rock volume. By examining the block geometries, in situ stress and rock conditions, the overall siting and layout of any planned facility can be optimised and supply feedback for the planning of the ongoing investigations.

The example below shows the tunnel geometry of SKB’s underground research facility beneath the island of Äspö.

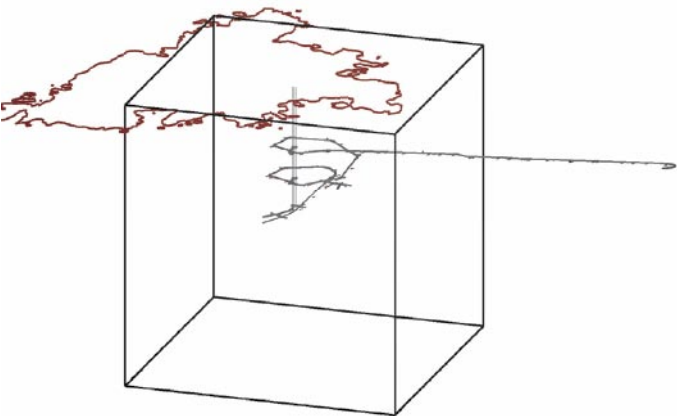


Figure 3-7. Dynamic design – attaching a design layout to the model.

The visualisation and modelling work in RVS continues throughout the detailed design and construction phases. RVS is used in combination with other tools to visualise the detailed mapping of the excavations as well as for the planning and visualisation of results from continued investigations.

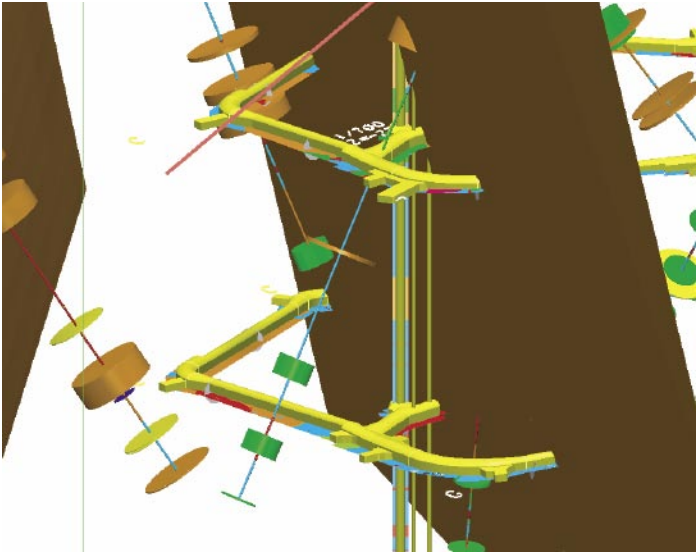


Figure 3-8. Visualization work during the detailed design and excavation phases. The figure shows a tunnel geometry, planar deformation zone surfaces, tunnel mapping and borehole data.

4 Working with models

4.1 Creating a model

All work in RVS is done within models. A model is based on a selection of available data and with the analyses and modelling that have been done using that data. Different models can interact so that you can be working in one model, while using other model(s), possibly from other users, as background models. It should be noted that the term Model is used by Bentley Systems® in MicroStation V8® with quite a different meaning, which should not be confused with the RVS Model concept. A model in MicroStation V8® is a collection of design elements that share a common coordinate space, similar to the familiar design plane or cube in earlier versions of MicroStation® and are comparable to the multiple worksheets in a Microsoft Excel® file /Bentley System Inc, 2003/.

Before any work can be carried out in RVS, a model must be created or loaded. The Load Model dialog opens automatically after a successful User login.

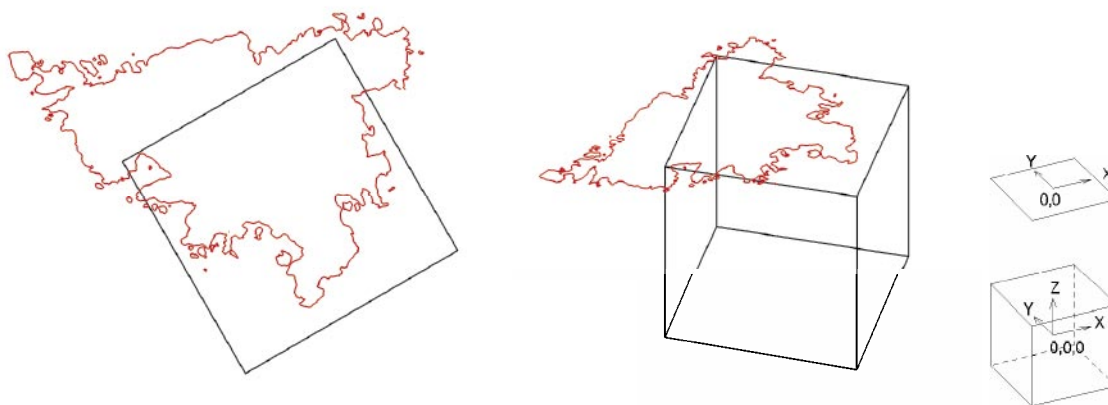


Figure 4-1. Defining the model volume.

Any user can create a new model. A new model is defined with a model name and a link to a site registered in SICADA. The model name needs to be unique; RVS will not allow a model name that is already logged in the local model database with the same User ID.

An automatically updated list of recognised sites is received from SICADA whenever a data-order is processed by RVS. The link to a SICADA site is a way of making the data handling process more efficient. It ensures that when the DA Server (Database Administration Server, see section 6 of this report) is searching for data in SICADA it is only looking for information that is relevant to the named site. However, it should be noted that this process is not restrictive in that orders can be linked to more than one site if desired.

When the User opens a named model for the first time an *Edit Model* toolbox automatically appears. It is in this dialog box that the user defines the extent of the model volume. This is done by defining an origin point for the model, followed by the extension from this point in the x,y and z directions. It is also possible to define a model box that is rotated in relation to a coordinate system.

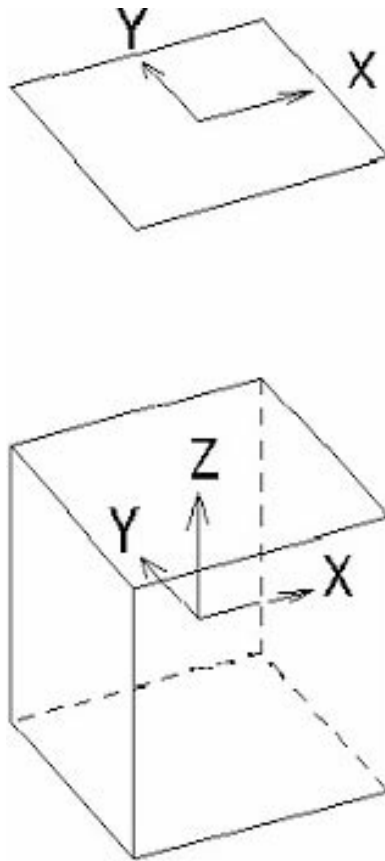


Figure 4-2. Defining the model volume – cuboid dimensions.

The model volume can be modified at anytime using the same toolbox. However, if the cuboid dimensions are subsequently increased in any direction then the visualised geometries of some objects may be affected and will need to be reviewed.

4.2 Model versions

The present state of a model can be frozen and saved by defining a version of the model. All such saved model versions remain available for review at anytime. They may be reviewed in isolation or they can be attached as background models to the active versions to facilitate comparison. Each new version will be sequentially numbered for traceability and may be saved as a major or minor version with a short description. There is no rigorous definition of what constitutes a major or minor version; it is left to the user's own subjective judgement.

4.3 Coordinate systems

RVS can handle data in different coordinate systems. Once the geometrical relationships between the different systems are defined, the transformations that are needed are made on the fly, quite transparent to the User without interruption to the workflow.

In RVS versions prior to 3.3 the user was required to save a model in the selected site's *Master Coordinate System*. This is no longer the case; in the current version of RVS the user can save a model in any defined coordinate system.

Data from SICADA is always received in the original coordinate system that was used to define the data, along with an identification of this system. Similarly, RVS stores the data using the same coordinate system and identification in which it was received.

Any data delivery received from SICADA is automatically accompanied by transformation details between all of the recognised SICADA coordinate systems.

Any model can be opened in any of the defined coordinate systems that are recognised as appropriate for a particular site. Each RVS site may employ one or more coordinate systems for the input and output of data. The relationship between defined systems for a site are stored in SICADA and delivered to RVS with each order. In addition, the User can define his own local sites and local coordinate systems for the in- and output of local data if desired, by defining transformations to the recognised sites and systems. However, it is unlikely that a typical SKB RVS user will need to define a new site.

The RVS naming and handling of the systems can be illustrated by the following example:

Data in SICADA for the Simpevarp site is stored in the original coordinate system in which the data was measured, e.g. Äspö96, RT90, RT38 etc.

Suppose a user now creates a new model in RVS and defines the model's *site* as Simpevarp and its *Native coordinate system* as RT90.

The user then places an order to SICADA and receives the requested data. RVS stores the data in its local database in the original coordinate systems, just like SICADA: Äspö96, RT90 and/or RT38.

The user then visualises some boreholes and related data, and models some objects based on these data. All the Boreholes and objects will be drawn and their geometries saved in the RT90 *Native coordinate system* even though the background data itself may be stored in the local database in another system.

Suppose the user then closes and reopens the model. The *Native coordinate system* has already been defined and is now "locked". However, the user can choose to view and work with the model in another related coordinate system- the *active coordinate system* e.g. RT38, and RVS will, on-the-fly, transform the coordinates of the boreholes and objects, and show them in the selected *active coordinate system*, though the geometries, including any newly visualised or modelled objects will be saved and stored by RVS in the *Native coordinate system*, RT90. At the same time the underlying data will remain stored, unchanged, in its original system e.g. Äspö96, RT90 and RT38.

4.4 Background models

Other models may be attached as background models to the active model. It may be of interest to attach earlier versions of the same model as background models. This assists in keeping a visual track on the model development, enabling the viewing of different versions of the same object at the same time. In addition, this method allows the comparison between models that may have been produced at different times for different purposes, but are located in a similar geographical region. All objects in a background model can be viewed in the same way as the objects in the active model but they can never be deleted or edited, though they may be copied and imported to the active model. The name of the attached *background model* will be visible as a separate branch in the *Object manager*.

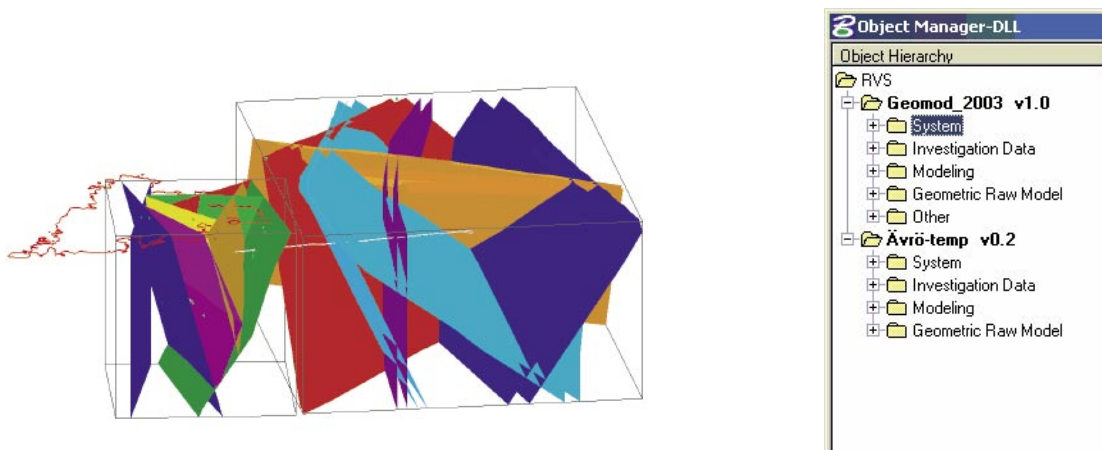


Figure 4-3. Attaching background models.

Above is shown the *Active Model* “Geomod” on the left with an attached *Background Model* “Ävrö” on the right /Markström et al, 2001/. The RVS Object Manager is shown on the far right.

5 Data input

RVS has been designed to enable a close integration with SKB's investigation database SICADA, which supplies the vast majority of primary input data to a model. The other major data input source to RVS is from SKB's GIS database.

5.1 Data input from SICADA

The data collected during site investigation work from drilling and borehole testing is organised and stored in SICADA as various parameters such as rock type, rock quality, permeability etc. The predominant source for RVS primary data is the SICADA database.

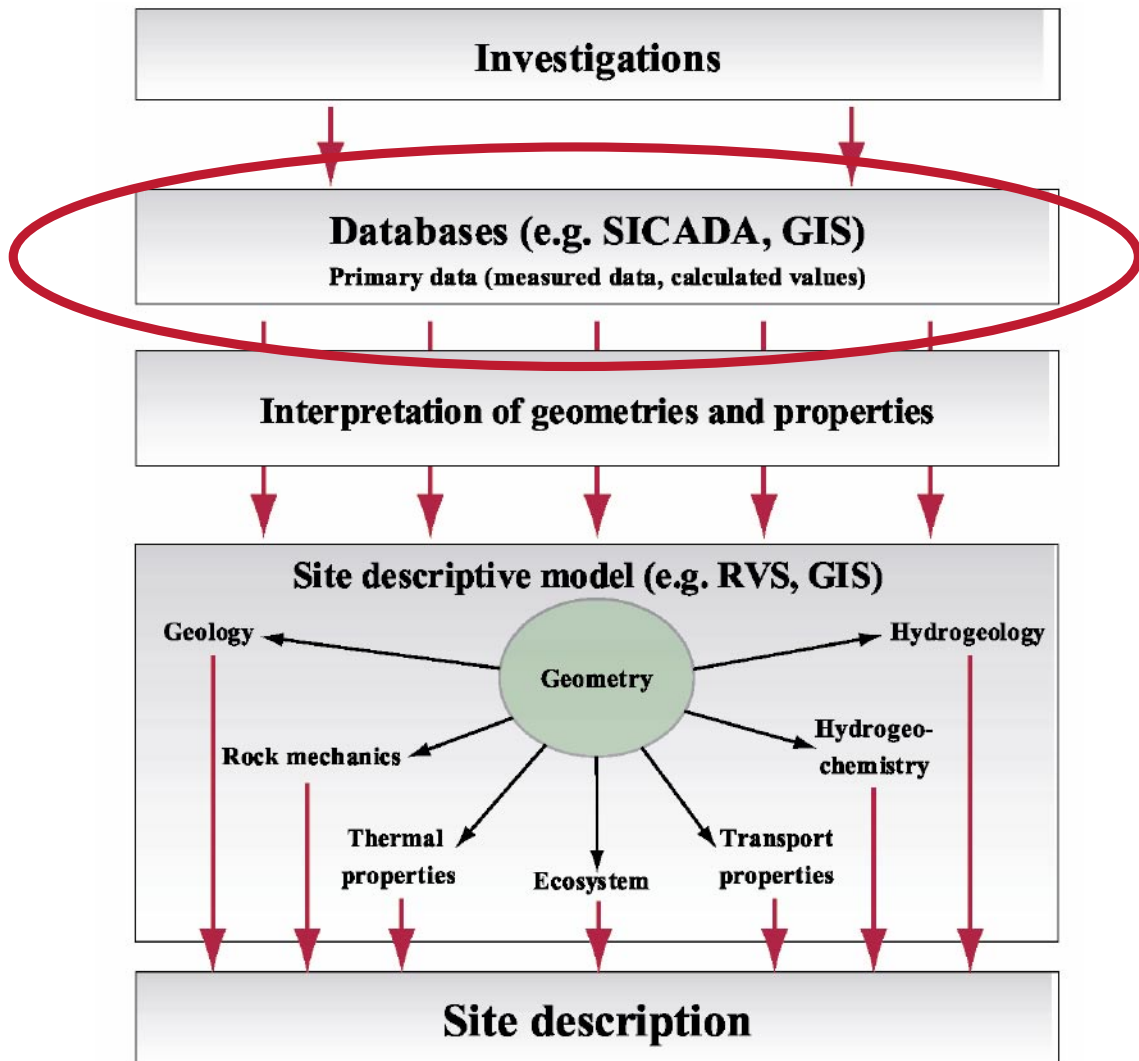
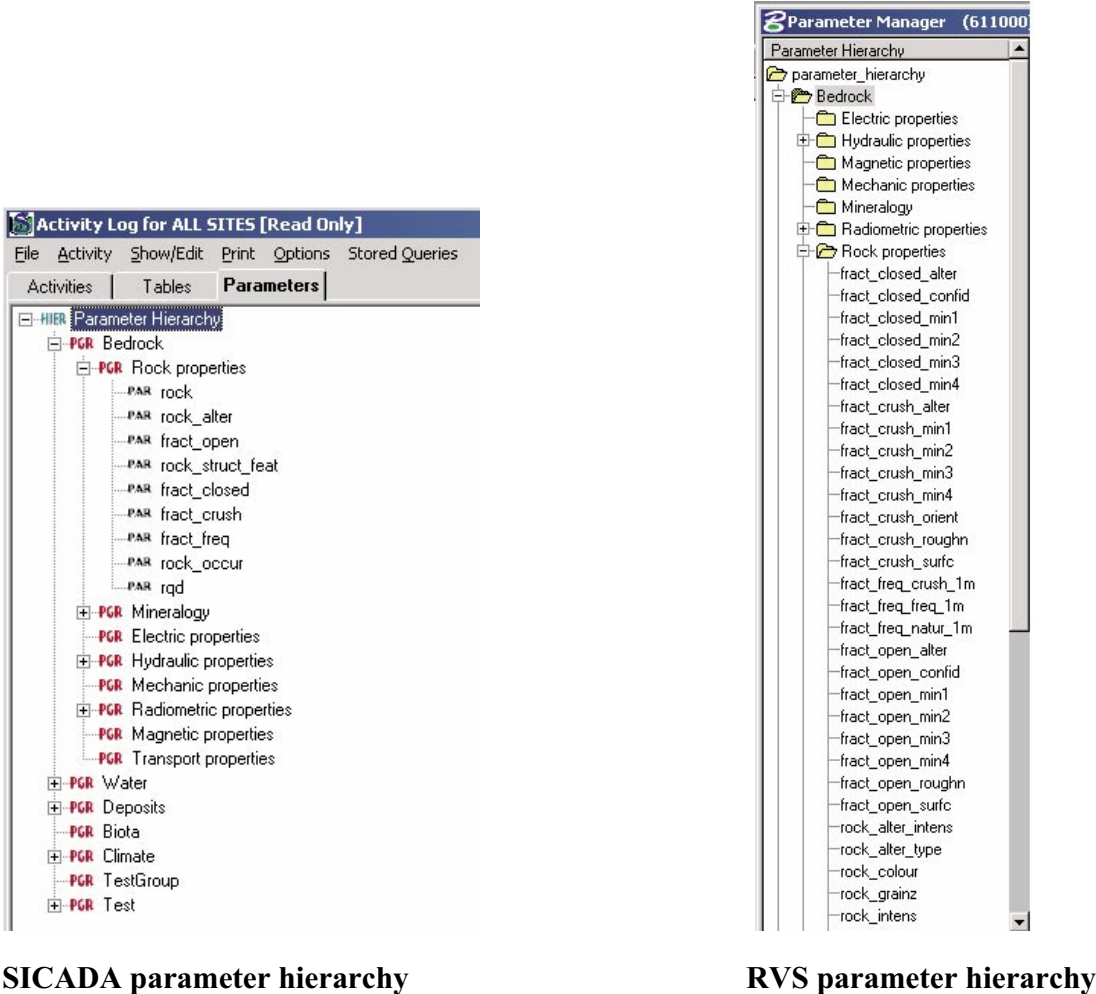


Figure 5-1. Data input from SICADA.

The SICADA system; the SKB Site Characterization Database system, is made up of several parts. The most central is the SICADA database itself. The database contains a large data set from many different investigations, probe methods and locations. The system has been continually developed and expanded over the last 20 years and contains data based on the experience obtained over a working period of more than 70 years. It covers data based on ca 60 000 *activities* covering 300 *activity types* that are logged in 800 *datatables* with an extensive *file archive* of raw data.

Several applications can be used when working with the data set. Important applications for the general user are the *SICADA Diary* and *SICADA Project* applications. Complete descriptions of both of these applications are available from the SICADA online help system. The SICADA Diary application is for activity and data registration and also for searching activities and data. The applications are built on top of an Ingres® RDBMS (Relational Database Management System)

RVS Version 3.5 and SICADA have essentially the same structure to their parameter hierarchies, see Figure 6-2 below. It should be noted that two other hierarchies are available in SICADA to serve other data access functions: a so-called Standard Hierarchy “Science/ Subject/Method/Activity type/Table” and a slightly modified version of the Standard Hierarchy, developed for specifically for PLU (SKB’s ongoing site investigation activities) /SKB, 2000/.



SICADA parameter hierarchy

RVS parameter hierarchy

Figure 5-2. Parameter hierarchies.

Within RVS an order can be placed to SICADA requesting the desired data. Commonly the request takes the form of a combined borehole list and selection of parameters. These form an integral part of the basis for most modelling projects.

The request for data is made within RVS and then sent to SICADA via the Database Administration Server (*DA Server*). It is the DA Server, which accepts the order from RVS, interprets the request, searches the SICADA database, extracts and packages the required data and then sends the data parcel to the RVS user.

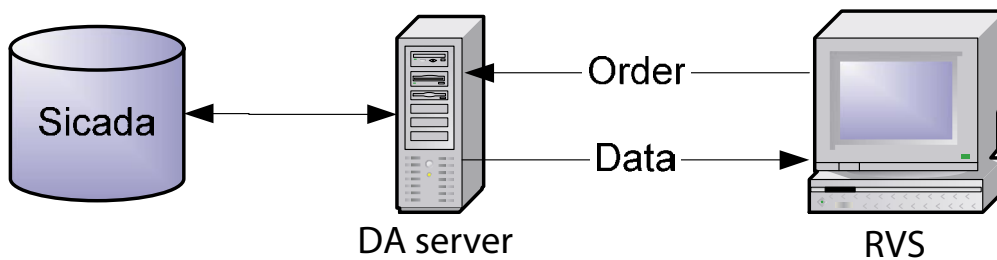


Figure 5-3. Data transfer between SICADA and RVS.

RVS is primarily designed for users who are connected to the SKB network and thus have an on-line connection with the SICADA database server. Data orders will be automatically monitored by the data access server, processed, and returned to the RVS without user intervention.

In order to enable data transfer to users on other networks, or connected to the Internet by modem, a semi-automatic, e-mail based delivery system has been established. User assistance is needed to e-mail and receive the SICADA orders before they can be loaded into RVS. The general workflow for such an offline order is as follows:

Create a data order to SICADA within RVS using the custom *Order data* tool and then browse to your designated RVS mailbox folder. By default, this folder is located in the RVS tree at the root level (below RVS). Your order file will have the name N.ord, where N is the order number.

Attach this file to an e-mail message and send it to the current e-mail address for data requests (available from SKB's RVS Development Group, see Appendix A). An automated SICADA function will reply with a confirmation that the order has been received and will be processed. The order is then sent to the DA server. With a short time-delay, depending on the size of your order, SICADA will deliver the result by e-mail with an attached *.rvs file. This file should be placed in the same mailbox folder where you found the N.ord file. Select the RVS *Transfer data* option. RVS will find the compressed *.rvs file, unpack it, and if its contents correspond to a pending order, change the state of that order number to READY.

The reply may be *analyzed* and ultimately *transferred* to the RVS local database by selecting the appropriate option. *Analyze* – allows the user to browse the contents of an answer file before transferring it to the local RVS database.

As well as the requested data, in every delivery SICADA includes an up-to-date set of *Check Tables*. *Check tables* contain descriptions of the naming routines and permitted values for the various parameters. These aid with the understanding of the data and need to be referred to when defining *Look-Up Tables* that control the graphical appearance of any visualized data.

Once the data has been received by RVS and stored in the RVS local database, a parameter can be selected for visualization.

5.2 Data input from external sources

5.2.1 Primary data

The *Import ASCII* and *Import local data* functions allow the user to import data from sources other than SICADA. However, these tools are not to be seen as alternative way of importing or processing data that otherwise lies in SICADA. SICADA data should always be imported directly via the semi automated RVS *order data* function for quality reasons. Review and sub-selection of data within a SICADA order is possible prior to transfer of data into the local database with the *Analyse data* function.

Many kinds of local data can be imported to the local database tables from ordinary ASCII or text files, including topographic surface data, borehole coordinate data and data for a parameter in a borehole. Such data can be connected to SICADA *sites*.

When importing data in delimited ASCII files, the ASCII file columns must be manually matched to the corresponding local database table columns. The formatted table must then be matched with the appropriate data type; surface, borehole coordinates, borehole parameter or animation data. It must also be linked to a *site* and a specified *coordinate system*.

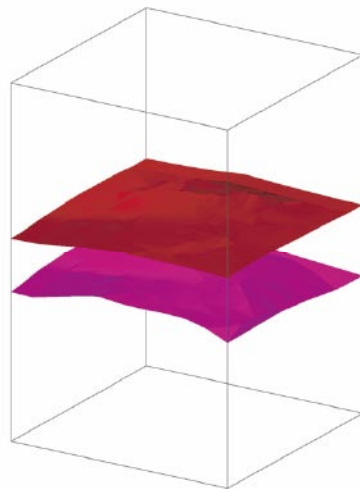


Figure 5-4. Modelling surfaces with RVS. Two surfaces generated in RVS via the *import ASCII* function. The upper surface represents topography and the lower an underlying bedrock surface.

5.2.2 Existing geometries

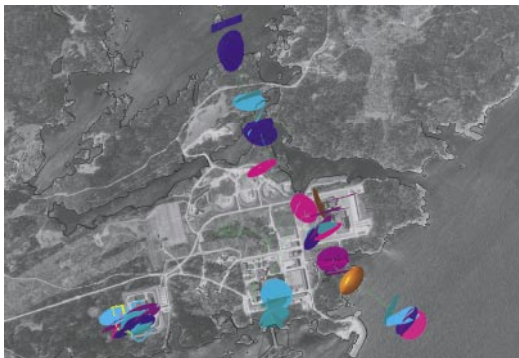
As well as data tables, existing geometries can also be imported into the active RVS model. External design file objects can be imported and transformed into RVS objects. However, it should be noted that with this method the imported geometry's link with its underlying data is broken. Certain kinds of such geometries can be assigned *type* and *parameters* allowing inclusion in the *block modelling* process. In addition, it is possible to import objects from existing RVS background models.

5.2.3 Attachments

Most external Microstation® design files can be attached to an RVS model. There are coordinate conversion facilities built-in so that design files in any defined local coordinate system can be used. Such files remain as attachments and are handled with the standard Microstation® tools. They are not imported and do not become RVS objects. However, it is possible to make a selection from such an attached reference file and truly import an element or shape, which will then become an RVS object and appear in the RVS *Object Manager* with a user defined name.

This method can also be used to work with AutoCad® and GIS files, which can be converted to Microstation® design file format and then attached in the same way.

A third type of attached data that can give a useful input into the modelling process is a raster image. These two dimensional images can be scaled, rotated, warped and attached using Microstation's® own Raster Manager. Examples of these types of images are orthophotos of study areas and the results of filed surveys e.g. aerial magnetic coverage.



a)



b)

Figure 5-5. Attaching georeferenced raster images to a model. a) Orthophoto of the Simpevarp peninsula attached to an RVS model. b) Results of an aerial magnetic survey attached to an RVS model /Curtis et al, 2003/.

6 Visualizing data in a model

6.1 Working with boreholes

6.1.1 Borehole data

Direct physical evidence of rock conditions at depth is obtained by rotary core drilling and testing. Data describing the borehole geometry together with the results of drill core mapping and in situ testing are stored in the SICADA database and retrieved by RVS for modelling.

RVS has a number of different tools for working with borehole data. Parameter data can be visualized as 3D bodies along the length of a borehole. In this process, a simple linear object representing the borehole geometry and a label with the borehole name is automatically created (see Figure 6-1 below). The borehole geometry can also be created without reference to parameter data.

RVS offers many kinds of parameter visualizations in boreholes. The appearance of the visualization is controlled through a *LookUp Table* (LUT), which defines the attributes of the visualization as a function of the parameter values. For example, different ranges of RQD (Rock Quality Designation) found in a borehole, can be represented by cylinders of differing colour and diameter placed along the borehole's centreline. In order to achieve a certain amount of standardization in the modelling process an RVS colour table has been predefined, along with a certain number of standardized LUTs representing key parameters. However, flexibility remains in the system allowing the user to define alternative LUTs. (see Microstation[®] reference documentation for further information about colour tables and LUTs). One of the main reasons for the creation of such standard Look Up Tables is that it facilitates the comparison of the rock mass between different models.

Parameters can be of three kinds, *point data*, *interval data* and *orientation data*. *Point data* are stored as a depth in the borehole and a measured value. *Interval data* are stored as "from"-depth, "to"-depth in the borehole and a value. Such data can be visualized as *lines*, *cylinders* or *prisms*. For *lines*, *cylinders* and *prisms* the attributes such as line style, colour, weight, diameter (for Cylinders and Prisms) and material, can be varied as a function of the parameter value.

Orientation data are stored as a depth in the borehole and the orientation of such structures as fractures and foliations can be visualized as oriented discs along the borehole. The colour and/or diameter are controlled by an LUT value and can be varied with orientation if desired. If a mapped parameter is selected, the LUT defines the graphical attributes of the discs as a function of the parameter value. Graphical attributes can be chosen so that only selected fractures will be visualized.

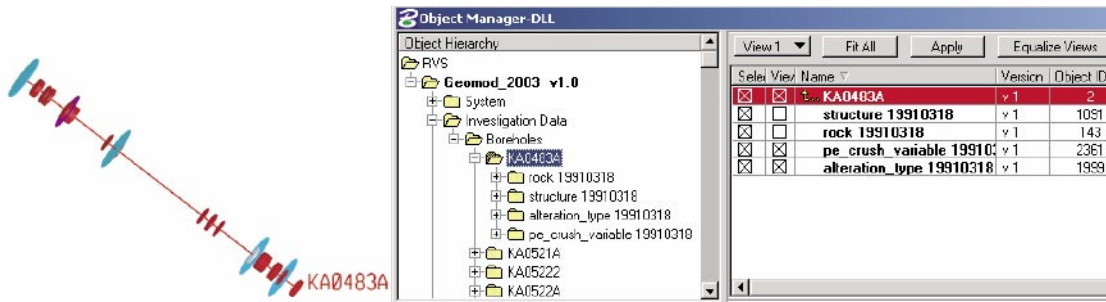


Figure 6-1. Borehole visualizations and representation in the Object Manager.

All of the visualisations and modelled objects, including the modelled structures, are defined as objects in RVS and appear in the *Object Manager* following an organised hierarchical tree structure. These object names are linked to their graphical representations in the design file by RVS as they are created.

SICADA is an active database and some of the contained parameter data maybe subject to ongoing modifications and updates. In order to assist the user in keeping track of such changes and to ensure that the most up to date data is being used, RVS automatically notifies the user when an existing parameter visualization is based on out of date values. Such visualizations are marked in red in the Object Manger. If the user wishes to update such visualizations using the existing *LUT* and *method* there exists an *Update Parameter* option in the *Object Manager* menu.

6.1.2 Planning future borehole investigations

As well as visualizing and analysing existing borehole geometries and data, RVS can assist with the planning of new boreholes and optimising of the investigations. The *borehole planning* function enables the user to place a (straight) borehole in 3D space, either by keying in the coordinates for the starting and ending points, or by entering a starting point along with the intended bearing, inclination and length. The user can also explore predicted intersections between the planned borehole and modelled objects or compute distances between the planned borehole and modelled objects that are not intersected. Alternatively, tentative boreholes can be imported as ascii files. This enables the user to create curved boreholes but requires more manual handling.

6.2 Constructing and visualising surfaces

Point information from the visualised boreholes, for example an interpreted crush zone, along with other point and trace geometries from, for example, surface rock type mapping or geophysics, can be used with the RVS modelling tools to generate surfaces, representing interpreted geological structures such as deformation zones. The created surfaces can be either planar or undulating.

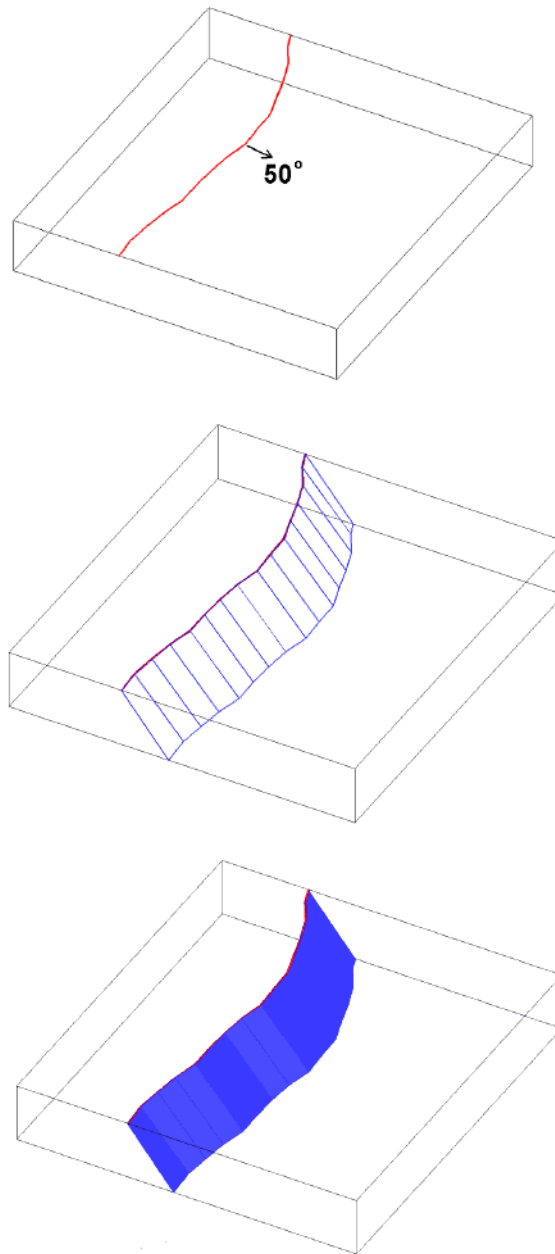


Figure 6-2. Modelling surfaces based on projections of surface lineaments. The example illustrates the generation of a surface in RVS, representing a deformation zone, from a topographic lineament trace combined with tentative dip information based on, for example, outcrop mapping or geophysical surveying. The user may vary the number of points, taken along the initial trace-profile line as considered appropriate.

It is possible to calculate the best fit of a planar surface to a given point set. Restrictions on Strike, Dip and or location can also be given. It is also possible to create a triangulated undulating surface using a given set of points. The points can be irregularly spaced and sorted in any order.

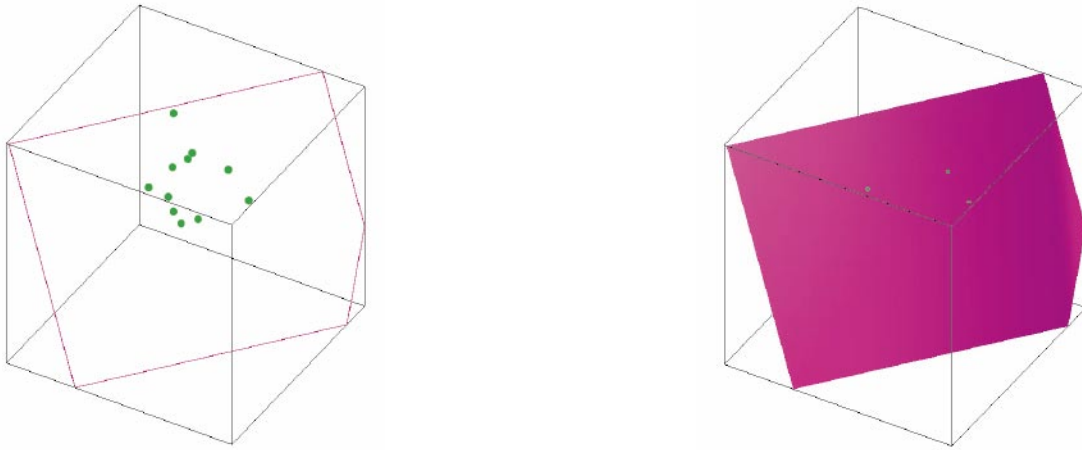


Figure 6-3. Generating a “best fit” planar surface. The example illustrates the generation of a planar surface in RVS, representing a deformation zone, based on a series of point locations interpreted from mapping of drill core. For example, such point locations may represent sections of drill core with low *rqd* values and characteristically similar patterns of alteration, which led a geologist to interpret them as belonging to the same tectonic structure. RVS has generated a “best-fit” mean planar surface to the point cloud.

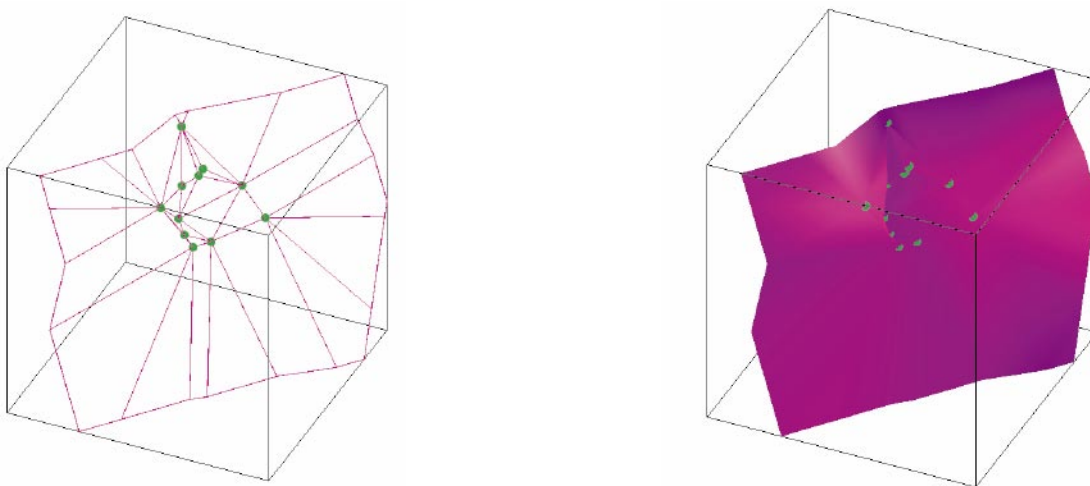


Figure 6-4. Generating a “best fit” undulating surface. The example illustrates the generation of an undulating surface in RVS based on the same point data as used above. RVS has triangulated the points (see method description below) and generated an undulating surface, which passes through all of the individual points.

RVS generates an undulating surface in the following manner:

- The point cloud is analyzed in order to determine the best projection onto a 2D plane.
- All the 3D points are then projected onto this 2D plane. The user can specify a DTMAXY tolerance- a filter that filters out points that appear to be very close in a 2D projection of the point set, in order to avoid bad triangulation. The measure is the minimum distance allowed between two points, expressed in meters.
- Three new arbitrary points are added, forming an all-encompassing triangle, fencing the 2D point cloud.
- All the points are triangulated in 2D.
- Edges that are connected to the three fencing points are removed.
- True 3D vertices are restored.
- A triangular mesh surface is created and clipped to the Model Boundary. *Note:* the resulting mesh contains a collection of planar polygons, not necessarily limited to triangles.
- Adjacent polygons are further analyzed and whenever coplanar polygons are found, the connecting edges are removed.

The xyz coordinates of the individual points, which form the underlying basis for the interpretation of any surface are saved as a *point table* in RVS and are always available for review and editing; allowing further remodelling. The points may also be saved as a sub-object to the surface, appearing in the *Object Manager* tree.

Additional RVS tools exist to modify and further develop the modelled surfaces as more information from investigations becomes available. For example, an already created surface can be remodelled if point locations have been added or changed; surfaces that intersect can be trimmed against each other. A thickness can be applied to a modelled surface to generate a narrow block that may better represent, for example, a deformation zone. The thickness maybe applied asymmetrically to the defining surface plane. Both the initial defining plane and the applied outer thickness bounding surfaces are available via the *Object Manager* and are included in the XML export file.

6.2.1 Topographic surfaces

The most efficient method to visualize a topographic surface in RVS is by importing a ready-made DTM file (Digital Terrain Model) using the custom made RVS tool *Import DTM Surface*. This makes it possible to import a point set from an already gridded surface and save it as a modelling object. The points must be sorted in rows and columns and can be based on a quadrangular or triangular grid. The source of the point table needs to be a comma delimited ASCII file with one 3D coordinate per line, in xyz order. The coordinate list must be ordered either by grid row or grid column. Inconsistencies will halt the process and produce an error message. This method short cuts the otherwise potentially lengthy processing time required to run geometry checks on a large unordered point set.

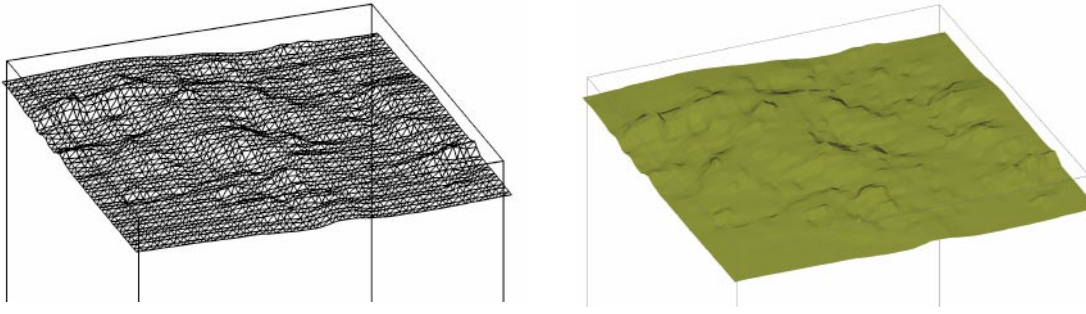


Figure 6-5. Visualizing a topographic surface by importing a ready-made DTM file.

6.2.2 Seismic reflectors

Data from Seismic Reflection investigations can be imported into RVS and the reflectors can be evaluated in 3D space, thus giving an opportunity to compare the possible interpretations of each reflector to other data and existing modelled objects. The investigation data is pre-processed before importing into RVS. The reflectors are interpreted by external programs and only the 3D analysis is done within RVS.

The interpretation of data from seismic reflection and radar investigations gives as a result a number of reflectors, each derived from a structure where any two points, p_1 and p_2 , have known distances from the recording points, r_1 and r_2 , on the profile.

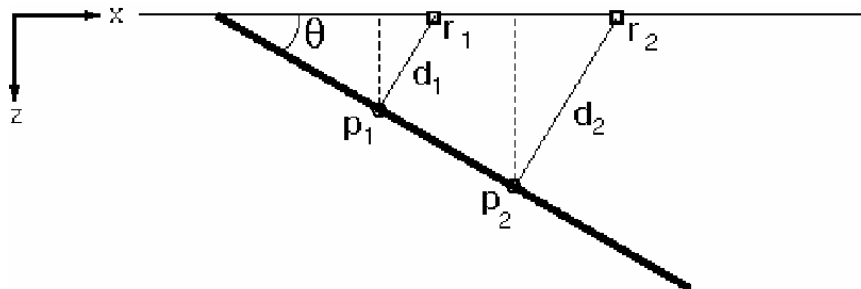


Figure 6-6. Seismic interpretation theory.

However, the direction to the reflecting element cannot be determined. The reflecting element therefore can be located anywhere on a cone whose radii from the recording points on the profile equals the distances, d_1 and d_2 , between the profile and the reflecting element. Theoretically, the cone could be built from full circles but as the profile for seismic reflection investigations is on the ground surface (or seabed), in practice only a half-cone is analysed.

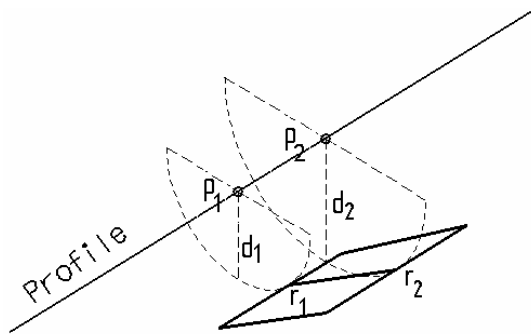


Figure 6-7. Examining alternative orientations for interpreted planes.

RVS makes it possible to analyse every possible position for each reflector along the half-cone described above.

The distances d_1 and d_2 , from the profile to the reflecting object are in practice recorded as travel times. These are transformed to distances by estimating the effective velocities of the host rock. If the travel times in the rock are relatively constant, this transformation is linear but if the travel times changes, i.e. if the effective velocity increases with depth, the cone will be flattened, see the Figure 6-7 where $d_{nH} < d_{nV}$. In RVS, different vertical and horizontal velocities can be entered and the user can immediately view the results.

For each reflector, a data sheet is stored. This sheet is created on import, initially with data from the input file. After a modelled reflector has been saved, the data sheet contains the interpreted strike, dip and reflector type, along with the input parameters.

Full details of the working method are included in the RVS on line help system; however, in summary the workflow is as follows:

Locate and import an investigation file. During the process, select or create a suitable LUT for your data. Also make sure you select the proper coordinate system.

Save the resulting investigation objects. The imported data will become an *Investigation Data* object with each reflector cone, line and eventually plane as sub-objects.

Modify the reflectors by rotating them to align with other planes or modelling objects. Save each modified reflector.

6.3 Modelling lenses

RVS has a simple tool for constructing lenticular bodies, which may be used to represent rock bodies such as intrusions or as an aid in representing anisotropy in the fabric of a rock mass. The shape and size of an ellipsoid are defined by the parameters *Length*, *Width* and *Depth*.

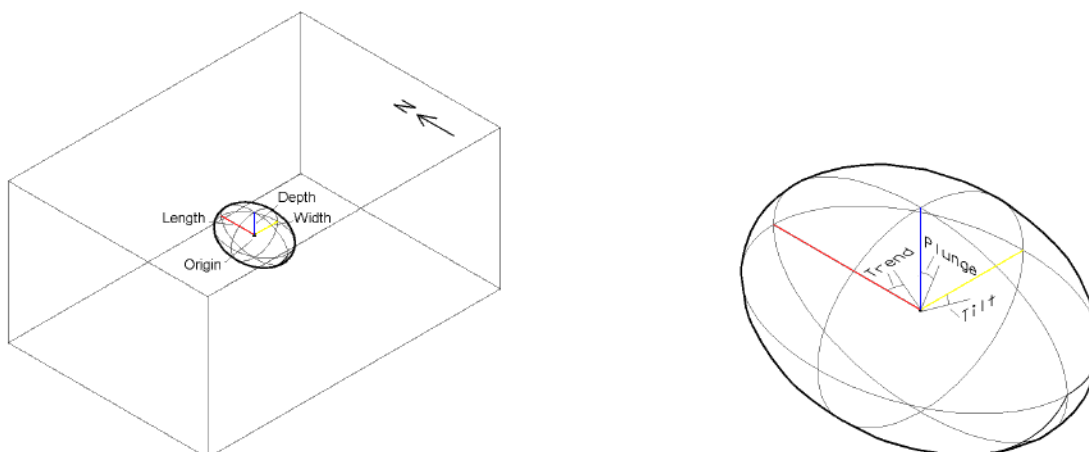


Figure 6-8. Input variable definitions for visualizing a lens.

The lens' position is determined by the x, y and z coordinates of the origin of the object body. The orientation is defined by the three angles; *Trend* (the rotation angle around the *Depth*-axis), *Plunge* (the rotation angle around the *Width*-axis) and *Tilt* (the rotation angle around the *Length*-axis). The user may also assign a predetermined *type* to a created lens so that it may take part in any subsequent block modelling. Like all RVS objects the user gives a name and depending on the *assigned type*, the object will appear in the relevant branch of the *Object Manager* tree.

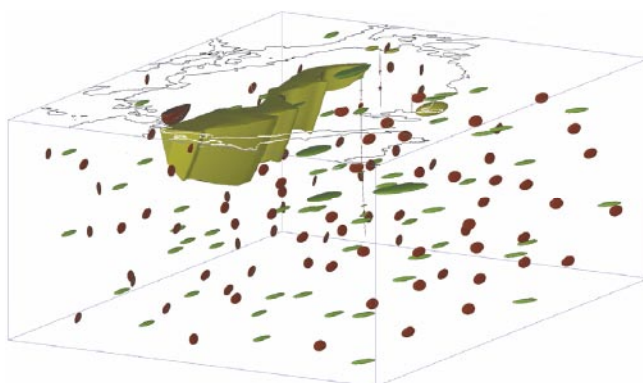
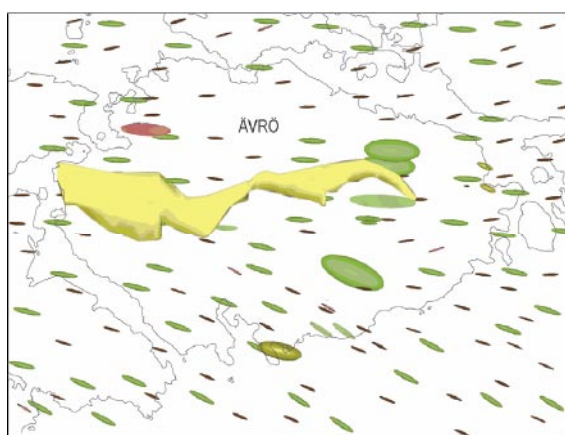
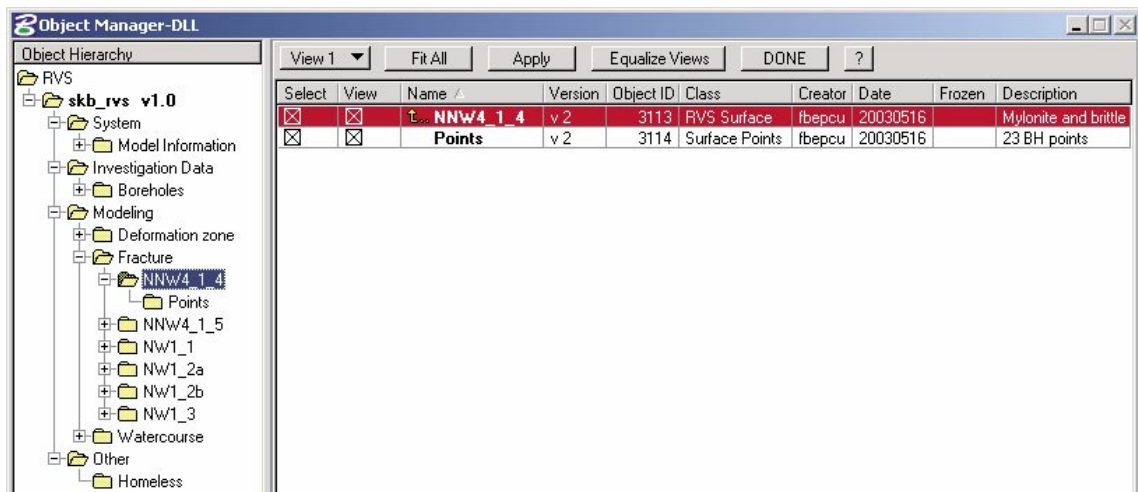


Figure 6-9. Schematic modelled rock bodies in the area of Ävrö. Red = fine grained granite to quartz-syenite, yellow = metavolcanite, green = diorite and gabbro, /Markström et al, 2001/.

7 Managing the model contents

7.1 Managing objects

The *Object Manager* is the main control panel for the viewing and graphical evaluation of RVS objects. Some attributes can also be edited through the interface. All objects in RVS are listed in the Object Manager. The Object Manager consists of two halves; on the left the objects are listed hierarchically in a tree structure and on the right any highlighted object and its first sublevel are listed with additional information.



Column descriptions

Select	A checkbox to select objects in the model. Only selected objects can be made visible, or take part in a block model.
View	A checkbox to turn visibility on or off for the selected objects.
Name	The name of the object or sub-level. The names of certain objects can be edited in the table. These are user created boreholes, modelled objects, blocks and block groups.
Version	The version number of the object.
Object ID	An internal sequential ID number.
Class	Displays the MicroStation or RVS class designation for the object.
Creator	The RVS User ID for the user who created the object or visualization.
Date	The creation date for visualization, modelling or block objects. The date for data capture and coordinate calculation for parameters or borehole locations. The date form is YYYYMMDD
Frozen	A black dot here signals a frozen branch or a frozen object. For example, if you have opened a read-only version of a model.
Description	A user added description of the subject in the row. This description is usually entered when the object is created but may in certain cases be added or edited in the Object Manager.

Figure 7-1. The Object Manager.

7.2 Managing local data and parameter tables

All parameters in the RVS local database, both those delivered from SICADA as well as those created locally, are listed in the *Parameter Manager*. The Parameter Manager consists of two halves. On the left the parameters are listed hierarchically in a tree structure. When a parameter is highlighted the boreholes that hold data for that parameter are listed on the right.

To the right is shown the list of boreholes that carry data for the selected parameter. Boreholes with parameter visualizations are presented in bold, and all visualizations are listed in the bottom part of the box. Here the parameter visualization for one ore more boreholes may be switched on or off. The actual parameter data itself can also be viewed.

Parameter data, borehole coordinates and look-up-tables that have been created locally or imported through the Import ASCII function can be exported to other RVS users. Parameters and borehole coordinates that have been delivered from SICADA cannot be exported for quality reasons. Instead, it is necessary to for other RVS installations to order such data from SICADA separately.

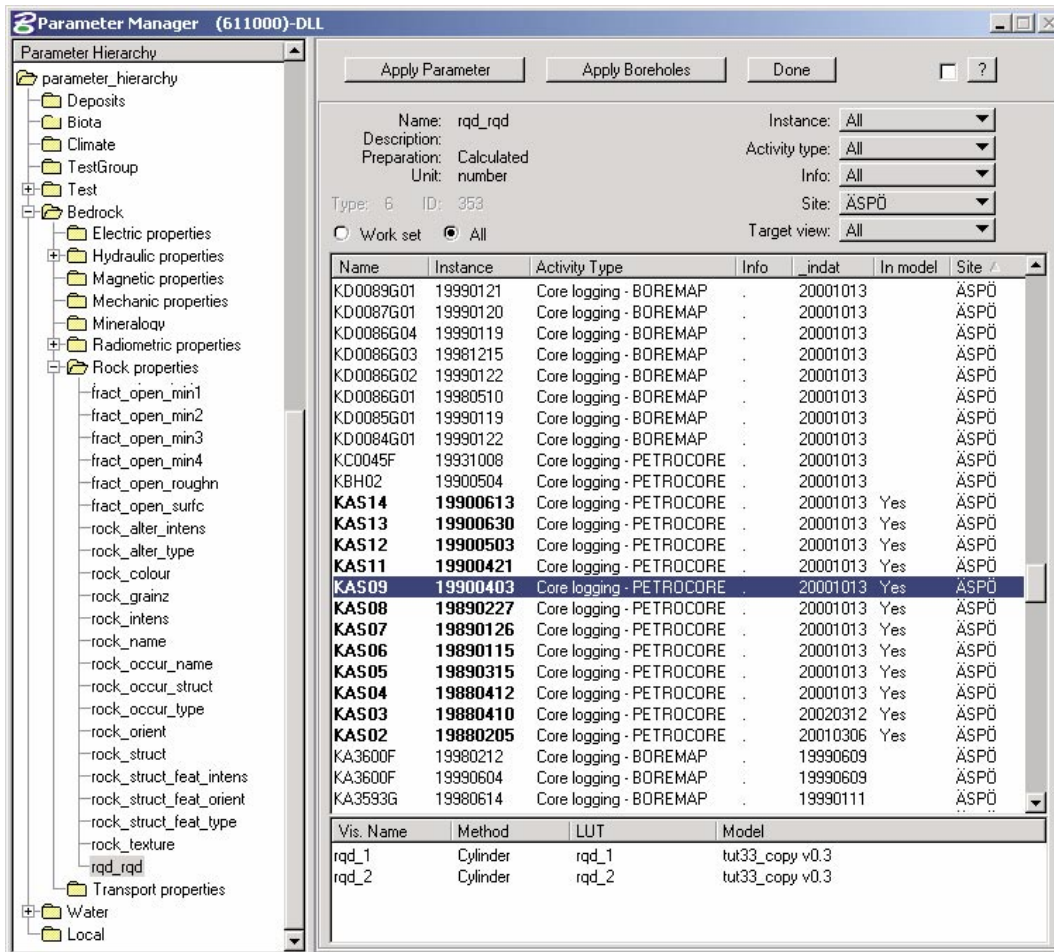


Figure 7-2. The Parameter Manager. Note: the rock properties parameter list shown above has been artificially shortened for clarity.

8 Block modelling

In RVS it is possible to create a *block model* where the entire model volume is composed of blocks divided by the modelled surfaces. The individual blocks are assigned characteristic values for physical properties and thereby provide a description for the entire rock volume. Only objects that have been assigned a *type* and have been *selected* in the *Object manager* take part in the block modelling process. The available types are defined in the database and include *rock unit boundary*, *deformation zone*, *topography* etc.

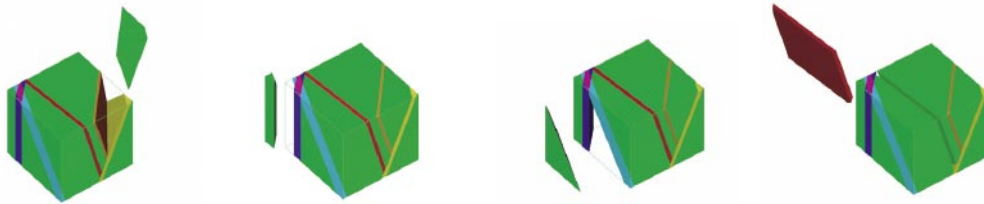


Figure 8-1. Individual block geometries.

An option exists to include *Smart Solid definitions* (ACIS information) in the block model. This is a time consuming operation and may be omitted. However, it is essential to include this option before generating an *XML export* file (see section 11 of this report). It is also possible to obtain the volumes of the individual blocks via a dedicated *compute block volumes* option. The results are available via the *set parameters* function and are included in the *XML export* file.

The block model appears under its own branch, *Geometric Raw Model*, in the Object Manager tree.

Limitation: On rare occasions, some 3D geometries can cause the MicroStation® core spatial engines to fail, thus producing erroneous results. Problems may appear when using large models with many complicated surfaces.

The individual blocks are automatically assigned to one in a series of potential *block groups* depending on the *type* of their modelled bounding surfaces. This surface type was assigned by the user during a previous modelling step. For example, a block bounded by a surface with the type *rock unit boundary* or *rock surface* will automatically be assigned to the *Rock unit block group*. Other potential groups are *Deformation zone*, *Soil unit* and *Water unit*.

After further modelling work the block model will need to be updated. The update is performed using the same tool that was used to generate the initial block model. Each time the block model is updated the resulting blocks are scanned and compared with those of the previous model. This scanning process involves RVS comparing the identities of the bounding surfaces of the individual block geometries in order to determine if a block in the updated model has the same bounding surfaces as its predecessor, to allow the direct inheritance of the previously assigned attributes and parameters. Since the inheritance process is dependent on the identities of the bounding surfaces it enables RVS to identify and transfer attributes and parameters to the new blocks even if there have been significant changes to individual block geometries.

However, if new modelling objects have been added prior to the block model update the user will need to define the new group assignments via the *RVS Group Inheritance Manager*. The *Group Inheritance Manager* lists all new groups that cannot automatically inherit features from a group in the old version of the block model. Here the user can manually, with visual help from a highlight function, transfer the data from groups in the old model version to groups in the new model version.

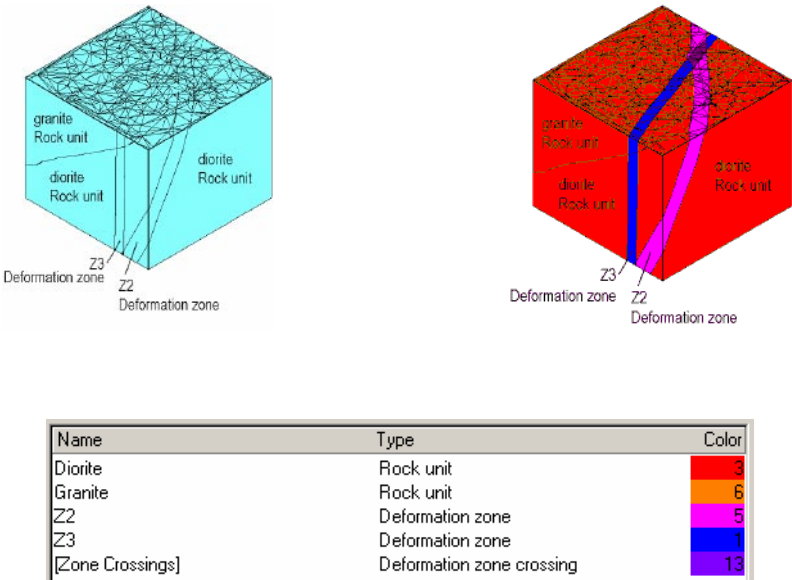


Figure 8-2. Block groups.

Different graphical attributes such as colour, line weight and style can be assigned to the different block groups using the *Block group symbology manager*. The colours are assigned such that the *Deformation zone group* colour overrides the *Rock unit group* colour.

Characteristic values for predefined parameters can be assigned to each block group. The list of potential parameters is extensive, including qualitative and quantitative parameters from all relevant disciplines. A basic example is shown below.

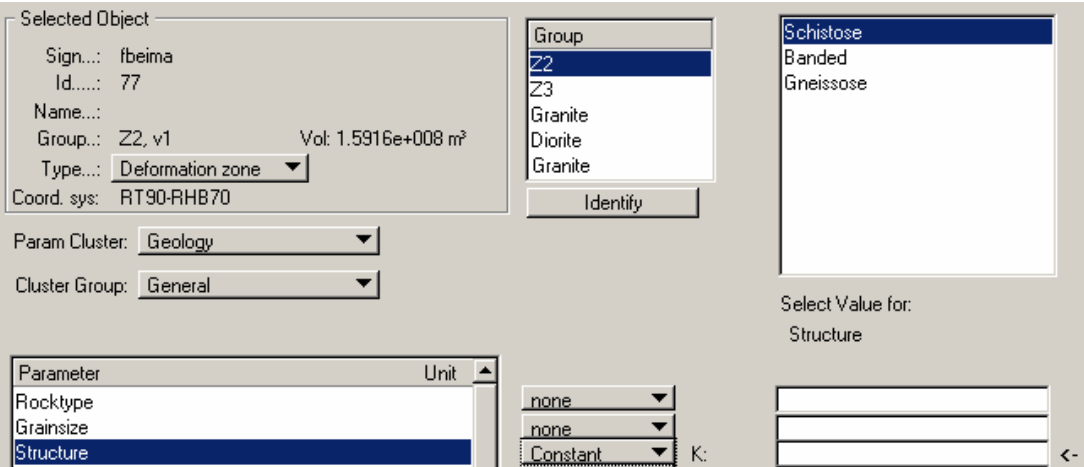


Figure 8-3. Parameter assignment.

9 Managing models between different RVS and Microstation[®] installations

RVS contains several options to assist with the management of models allowing such operations as the transfer of models between users, copying and backing up of RVS models. The internal RVS tools, which drive these actions, are briefly described in this section. Outside of RVS, the resulting exported models are distributed and exchanged with other RVS users through SIMONE, an Internet based data repository service. This service is available for designated RVS users.

It should be noted from the beginning that all of the following model export and copying functions only include the model itself and not the underlying local data, Look up tables (LUTs) or imported SICADA data, resident in the RVS local database. The local data and LUTs can be exported separately with RVS *local export* tools. Received SICADA data cannot be exported with a model since, for quality reasons, SICADA data can only be imported directly from SICADA to any individual RVS installation and not between installations. If a user has editing rights to an imported model he has received from another user, then he must first import the relevant data from SICADA before continuing the modelling work.

The *Export Model* function is used to export models from one RVS installation to another. There are four different types of export for models that a user “owns” or has *user rights* to: you can *Publish* your model to all other users, *Send* your model to one or more user(s), temporarily *Transfer* your model to another user for editing or *Relocate* your model permanently to another installation.

For models that are sent to you from another user, via the transfer option, the only export alternative available is to *Re-send* the model to the owner.

Publish – this option creates a copy of the model for all users to import and load for Read only. A new version is created on the home machine. This version is fully editable. Model parameters can be edited at other machines by those with the proper authorization. In addition, an XML formatted file is automatically generated.

Send – this option creates a transfer file to any user. All other users can import the model but only load it as Read only.

Transfer – this option creates a copy of the original model, which can be imported and loaded with Edit rights at another installation by the model owner. The remaining original version will be locked as *read only* until the return of the active copy, when it will be replaced.

Return – a published model or a model that has been temporary transferred to you can only be exported back to the original sender as a return copy. If the model was transferred to you, it will be removed from your machine. However, if it was a published model a copy will remain with you.

Relocate – this option creates an export file to permanently relocate a model to another installation. The model owner can import it and continue work on the receiving installation. However, the model will be completely removed from the original installation.

The *Import Model* function is for importing exported RVS models. Export files are stored with the extension *.rvs*. The *Import Model* function allows you to import models exported from other RVS installations. The models can be exported using one of the five different export types described above. The read only/edit status of the imported model depends on how the model was exported.

If the model was *Published* to all users or if the model was *Sent* to any installation for viewing, it will be available as read-only after import. If the model was temporarily *Transferred* to your installation for editing or *Relocated* permanently to your installation, users with user rights can edit the imported model. This also applies to any of your own models that have been previously transferred to another installation and then *Re-sent* to your installation.

The *CopyModel* function is used to make a local copy of any model. Note that the copy becomes a new and fully independent model with its own RVS identification. Thus, it can be used to make an editable copy of a read-only model. However, this function cannot be used to make a copy for use at another installation or by another user. A suitable Export Model type must be used for this purpose.

The Backup Model function allows any model to be backed up for safety purposes. The function will create an **.rvs* file that may be reactivated if necessary with the RVS *Restore Model* tool. The export local data function can be used to back up local, non-SICADA data, if required. SICADA data present in the local RVS database cannot be backed up or exchanged and must always be re-ordered from SICADA for quality reasons.

As described above it is possible to export an RVS model with various restrictions to other installations and users. There are similarly straightforward methods for the exchange of locally created data, parameters and look up tables. SICADA data cannot be exchanged between users and installations; each installation must order its data directly from SICADA.

As part of the modelling process it is common to attach Microstation® design files to the active model to provide background information originating from various sources. When a model file is exported as an **.rvs* model file or saved as a Microstation® design file and sent to another installation, the connection with the reference files is broken. Thus it is sometimes necessary to send a package of files and data to another installation if the model is to be reviewed.

RVS Limitation: the exchange of such file packages between different RVS installations presents no serious difficulty since each RVS user can reattach the relevant reference files if required. However, problems arise if the model is saved as a standard Microstation® design file with the intention of distributing the model to a wider audience, who do not have access to RVS. The same problem arises when this export process is used to generate a Microstation® design file, which can then be modified in Microstation® for the production of images for a report. The problem in both cases is centered on the fact that often reference files have been originally created in different coordinate systems. While RVS itself can import and transform a design file based on virtually any defined coordinate system, this transformation is not available when the file is subsequently viewed in Microstation® alone. This often prevents the straightforward reassembly and viewing of the model with its background data by non RVS users.

10 Managing models within a project team and between different software systems

It is possible to save any model or a selection of objects as a standard Microstation® design file, which in turn can be converted to a standard AutoCad® drawing file if required. One RVS object is saved on each drawing level with the levels in the design file being given the same names as the objects in the RVS model. These files can be distributed along with a free viewing program (Bentley View®) that enables non Microstation® users in any project team to remotely follow and review the progress of the modelling work.

In addition to the proprietary *.rvs and *.dgn export formats, the geometry and parameter data of a model can be exported as an XML-file, which follows the rules laid down by W3C (The World-Wide Web Consortium). W3C's eXtensible Markup Language (XML) enables the creation of documents and databases whose contents are self-describing, i.e. the distinct items of data within such databases can be individually recognized and separately extracted from the medium in which they are typically stored and presented. Further information about W3C and XML can be found at www.W3.org.

A custom designed tool for exporting an RVS model in the XML format has been created in RVS. The resulting export file format allows subsequent import to other visualization tools such as TechPlot® and numerical modelling programs. This allows further science specific modelling and analysis work to be carried out on the geometrical framework representing the structural geological model created in RVS. The ultimate aim of this approach is to create an integrated geoscientific model covering the entire rock volume under study.

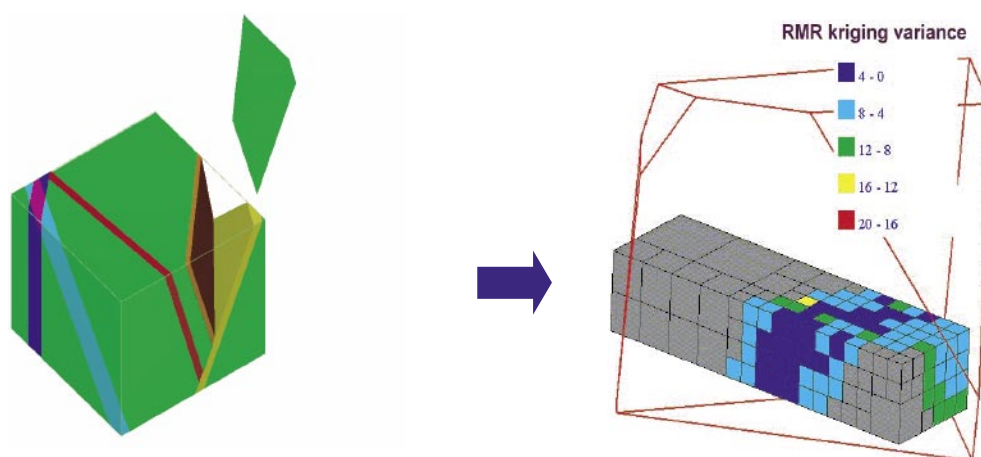


Figure 10-1. Export of block geometries and parameters for subsequent numerical modelling.

References

Andersson J, et al, 2002. Testing the methodology for site descriptive modelling. Application for the Laxemar area. SKB report TR-02-19. Svensk Kärnbränslehantering AB.

Bentley Systems Inc, 2003. Microstation V8.

Curtis P, Elfström M, Stanfors R, 2003. Visualization of structural geological data covering the Simpevarp peninsula, Ävrö and Hålö. SKB report P-03-86. Svensk Kärnbränslehantering AB.

Markström M, Stanfors R, Juhlin C, 2001. RVS-modellering. Ävrö Slutrapport. SKB report R-01-06. Svensk Kärnbränslehantering AB.

Munier R, et al, 2003. Geological Site Descriptive Model. A strategy for the model development during site investigations. SKB report R-03-07. Svensk Kärnbränslehantering AB.

Munier R, Hermanson J, 2001. Metodik för geometrisk modellering. Presentation och administration av platsbeskrivande modeller. SKB report R-01-15. Svensk Kärnbränslehantering AB.

SKB, 2000. Geovetenskapligt inriktat program för undersökning och utvärdering av platser för djupförvaret. SKB report R-00-30. Svensk Kärnbränslehantering AB.

The working environment

The RVS working environment will appear familiar to any Microstation® user. Below is illustrated a standard double screen installation with the RVS main tool menu shown, magnified for clarity, on the left.

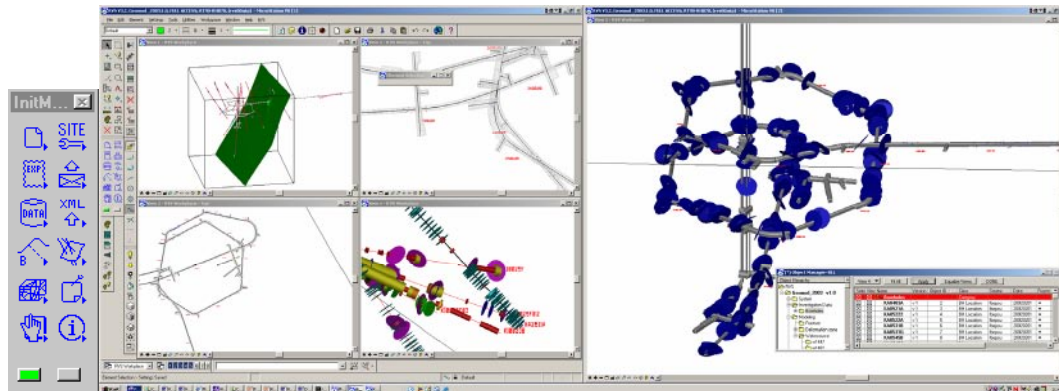


Figure A-1. RVS Working environment.

The GUI, the graphic user interface

RVS runs as a fully qualified MicroStation® application using MicroStation's® graphic user interface (GUI) and is configured to run automatically when MicroStation® is launched.

The User interacts with the software via dialog boxes where data can be entered and buttons or other controls determine the subsequent processing by the system. Most dialogs are non-modal, i.e. processing continues “behind” the box as data is being entered or as selections are made. Several such dialog boxes can be active at the same time. This is the usual mode for a MicroStation® dialog box. However, other boxes are modal, i.e. they will halt the execution of the program. Only one modal box can be open at a time, and processing continues only when the box is closed. This is the more common mode of the Microsoft Windows® GUI.

RVS Tool Boxes

Within RVS the various functions are subdivided into a number of toolboxes. All tool icons are found in the *Main menu*, which will open once a model has been loaded. Here can be found all of the available tools in RVS including functions for importing and exporting data, management of borehole information and geological modelling.

RVS Main Menu



**Model Manager, Site Manager
Model Transfer, Sicada Tables
Parameter Manager, External mport/Export
Data Data Visualization, Modelling
Block Modelling, Object Manager
System**



Model Manager: Create model, Delete model, Edit model, Create Model version, Attach/detach background project.



Site Manager: Site manager, Coordinate system manager.



Model Transfer: Export model, Import model, Copy model, Backup model.



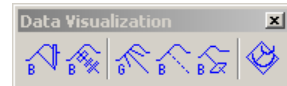
Sicada Tables: Order data, Transfer data, Check tables, Locate objects on activities, Locate activities on selected objects.



Parameter Manager: Parameter manager, View data tables, Export local data, Import local data, Mathematical analysis, Statistical Analysis.



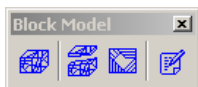
External Import/Export: Export model to XML, Export model to DGN, Attach external DGN, Import from ASCII, Import DTM, Import external geometry.



Data Visualization: Draw borehole geometry, Visualize parameter in borehole, Create borehole workset, Borehole planning, Analyze borehole intersections, Create/modify seismic reflector.



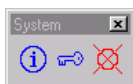
Modelling: Model surface, Model undulating surface, Model lens, Trim surface, Remodel surface, Modelling assign type, Change object attributes.



Block Modelling: Create block model, Groups inheritance manager, Block groups symbology manager, Set parameters.



Object Manager: Object manager.



System: InfoSnap RVS settings, change password, RVS Reset.

Figure A-2. RVS Tool boxes.

The current report describes the main features of the most important core tools in RVS and how they are used in the modelling process.

For most of RVS functions step-by-step tutorials are available describing how a particular tool can be used to carryout a specific task. A complete set of updated tutorials is issued with each new version release of the RVS program. However, the tutorials do not cover all the possible uses of all the individual tools but rather give an overall view of their functionality. A detailed description of every RVS tool and how it can be used is included in the RVS online Help system. A copy of the RVS help system in Microsoft Word® format accompanies each release of the program. The online help system is continually updated between RVS version releases and an up-to-date version of the online help accompanies the release of each RVS program “service pack”. Service packs are distributed intermittently between major version releases of RVS. As well as containing “fixes” to any identified program bugs, a service pack also contains new tools and general operating improvements. It is strongly recommended that anyone operating RVS makes sure that he/she is using the latest version of RVS and the most up-to-date service pack.

Any RVS user, potential user or interested party may send RVS related questions to the following RVS support service address:

rvs@skb.se

Alternatively, it is possible to take direct contact with one of the following members of SKB’s RVS Development Group.

Ingemar Markström
ingemar.markstrom@fbe.se

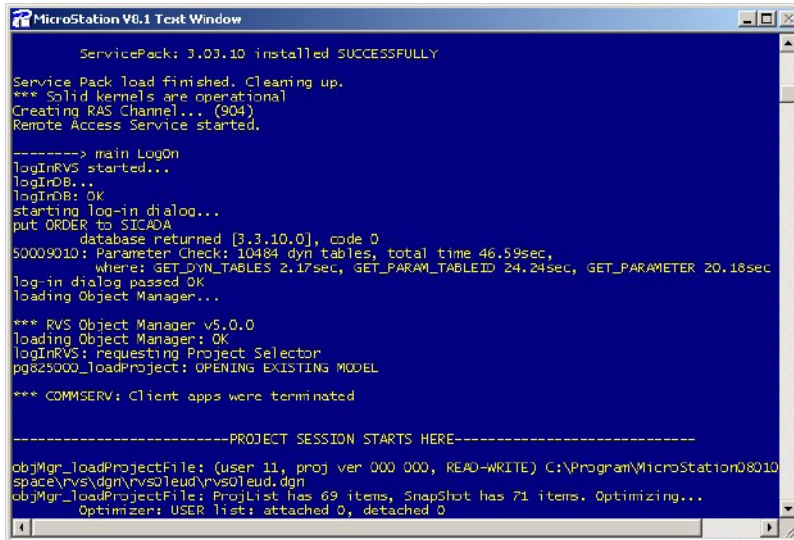
Mats Elfström
mats.elfstrom@fbe.se

Overall responsibility for RVS, both for program development and implementation, lies with Mats Ohlsson of SKB.

Mats Ohlsson
mats.ohlsson@skb.se

RVS text window

Once Microstation® and RVS have been launched, system information is displayed in a text window that runs underneath the RVS main window.



```
MicroStation V8.1 Text Window

ServicePack: 3.03.10 installed SUCCESSFULLY

Service Pack Load finished. Cleaning up.
*** Solid kernels are operational
Creating RAS Channel... (904)
Remote Access Service started.

-----> main LogOn
logInRVS started...
logInPB...
logInPB: OK
starting log-in dialog...
put ORDER to SICADA
database returned [3.3.10.0], code 0
50009010: Parameter Check: 10484 dyn tables, total time 46.59sec,
      where: GET_DYN_TABLES 2.17sec, GET_PARAM_TABLEID 24.24sec, GET_PARAMETER 20.18sec
log-in dialog passed OK
Loading Object Manager...

*** RVS Object Manager v5.0.0
loading Object Manager: OK
logInRVS: requesting Project Selector
pg825000_loadProject: OPENING EXISTING MODEL

*** COMMSERV: Client apps were terminated

-----PROJECT SESSION STARTS HERE-----

objMgr_loadProjectFile: (user 11, proj ver 000 000, READ-WRITE) C:\Program\MicroStation08010
space\rvs\dgn\rvs01eud\rvs01eud.dgn
objMgr_loadProjectFile: ProjList has 69 items, Snapshot has 71 items. Optimizing...
Optimizer: USER list: attached 0, detached 0
```

Figure A-3. The RVS Text window.

The text constitutes an ongoing operational log of actions carried out by Microstation® and in case of a database glitch or other system error, the information in this window will assist in troubleshooting.