



Wog - Well Optimization by Geosteering: a Pilot Software for Cooperative Modeling on Internet

J. F. Rainaud, O. Vincké, M. Elkouhen, M. Perrin, S. Brandel

► To cite this version:

J. F. Rainaud, O. Vincké, M. Elkouhen, M. Perrin, S. Brandel. Wog - Well Optimization by Geosteering: a Pilot Software for Cooperative Modeling on Internet. Oil & Gas Science and Technology - Revue d'IFP Energies nouvelles, 2004, 59 (4), pp.427-445. 10.2516/ogst:2004029 . hal-02017313

HAL Id: hal-02017313

<https://ifp.hal.science/hal-02017313>

Submitted on 13 Feb 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

WOG - Well Optimization by Geosteering: A Pilot Software for Cooperative Modeling on Internet

J.F. Rainaud¹, O. Vincké¹, M. Elkouhen¹, M. Perrin² and S. Brandel³

¹ Institut français du pétrole, 1 et 4, avenue de Bois-Préau, 92852 Reuil-Malmaison Cedex - France

² École des mines de Paris - Centre de Géologie de l'ingénieur, 60, boulevard Saint-Michel, 75272 Paris Cedex 06 - France

³ Université Louis-Pasteur, Strasbourg - LSIT, UMR 7005 CNRS-ULP, Parc d'Innovation, boulevard Sébastien-Brant,
BP 10413, 67412 Illkirch Cedex - France

e-mail: jean-francois.rainaud@ifp.fr - olivier.vincke@ifp.fr - mehdi.elkouhen@ifp.fr - michel.perrin@ensmp.fr - brandel@dpt-info.u-strasbg.fr

Résumé — WOG - Optimisation des forages par Geosteering : plate-forme logicielle de modélisation coopérative par Internet — L'IFP a commencé le développement d'un nouveau système de management de données appelé WOG - *Well Optimization by Geosteering*. Dans le cadre de ce projet, l'information est distribuée et partagée à travers le réseau Internet. Le plus grand avantage de ce système est d'offrir à chaque utilisateur (géologue, géophysicien, foreur, etc.) une vision propre à sa spécialité et qui lui est familière.

WOG utilise trois développements récents : Géopilote 3D, EpiSEM-Action et G3Server.

Géopilote 3D est un logiciel qui permet de calculer des relations paramétriques entre les différentes couches stratigraphiques et les failles afin d'établir une mise à jour compatible avec l'avancement du forage. En utilisant Géopilote 3D, le géologue et le géophysicien peuvent, à partir de l'information disponible sur une station de travail classique, construire un modèle géologique montrant différents types de valeurs reliées à des champs disciplinaires variés mais toutes rattachées à un même *conceptual Earth model*.

Le projet EpiSEM-ActionTM propose une nouvelle approche permettant la collaboration entre modèles géologiques partagés sur Internet ou intranet et la capitalisation des connaissances métier qui viennent s'y ajouter. Les services Web fournissent les outils nécessaires pour manager ces connaissances spécifiques avec une sécurité optimale entre serveurs et applications. Ceci permet de produire et de manager des modèles basés sur des hypothèses et des informations différentes.

G3Server est un serveur Corba dont la finalité est de distribuer de façon standardisée les objets IFP entre l'application Java ou C++. Son modèle interne (le modèle de communication IFP) a été défini pour être facilement utilisable avec les outils Open Spirit et Rescue. Son rôle dans le projet WOG est de distribuer les données entre les différents modules applicatifs. Il peut être utilisé par les différentes activités et accéder aux serveurs Open Spirit, aux fichiers Rescue et aux objets IFP. Les domaines couverts sont le puits, le réservoir et la sismique.

Abstract — WOG - Well Optimization by Geosteering: A Pilot Software for Cooperative Modeling on Internet — IFP has recently started the development of a new data management system named WOG for "*Well Optimization by Geosteering*". In this project, our aim is to distribute the information through the network so as to share knowledge. The key advantage of this software is to allow visualization for each of

the fields involved. Hence, the driller will have the view that he uses currently, the geophysicist will have the type of view that he sees every day, and so one for each specialties.

WOG uses three recent developments: 3D Geopilot, EpiSEM-Action and G3Server.

3D Geopilot is software that computes parametric links between the different layers and faults so as to allow future updating during the drilling process. Using the 3D Geopilot, engineers specializing in geology and geophysics could start from classical workstation interpretation information to set up a first version of a multi-discipline Earth model based on the same conceptual Earth model.

The EpiSEM-ActionTM project proposes a new approach to ensure the collaboration between Shared Earth Model (SEM) applications on Internet or intranet and the capitalisation of value-added business knowledge. Web services are provided to manage this business knowledge on the Web with optimized security between servers and applications. This enables models for coping with various hypotheses and interpretations to be produced and managed.

The G3Server product is a powerful Corba server whose objective is to distribute standardized IFP business objects between Java or C++ applications. The internal model (the IFP communication model) was defined so as to be easily used with Open Spirit and Rescue business data objects. Its role in the WOG project is to distribute data between application components. It can be called by specialized activities and can access Open Spirit servers, Rescue files and IFP proprietary business objects. The domain covered is the well, the reservoir, and seismic information.

INTRODUCTION

3D Earth modeling started to be experimented in the 1980's. Since then, outstanding progress has been made so that 3D geological models have become a major tool for hydrocarbon exploration, with companies dedicating an important fraction of their E&P budget to their development.

The main point of progress has been the variety of models produced. Reservoir models give an increasingly accurate picture of geological structures and of petrophysical property patterns at the reservoir scale, enabling hydrocarbon reserve estimation. Large-scale kinematical basin models represent the global basin geological history and related hydrocarbon formation and migration. Detailed physical or phenomenological models are built to describe particular geological processes or assemblages: sedimentary deposits and fold or fault arrangements of various types [1]. In addition, analogical modeling produces small-scale physical analogues of folded and/or faulted structures which can be reasonably realistic [2].

Significant progress has also been achieved in the model building technology. Increasingly appropriate surface representations are available, which enable complicated structures and discontinuities to be coped with. Sophisticated geo-statistical tools enable the visualization of spatial properties such as petrophysical repartitions, or microfracturation patterns but also the simulation of casual time dependant events such as turbidite deposits, catastrophic flooding, etc. Moreover, present day modeling tools are increasingly user-friendly; they often provide possibilities of exceptional interactivity in addition to sophisticated visualization tools [3].

For the oil companies, who are the main users, examples of present-day challenges concern operational work based on 3D

models which are updated while drilling (for day to day drilling management), as well as the storage of models of different types and their management (*i.e.* mutual comparison, reuse, updating and modification, etc.) by various end-users, in cooperative environments over the Internet.

These requirements imply the distribution of all the information over, and through, the network (*Fig. 1*) in order for each of the users to share the available knowledge. It is also desirable to provide specific visualization tools for engineers and experts (drillers, geologists, geophysicists, reservoir engineers) involved in different tasks, so that each of them, while cooperating with the others, can use the visualization he is familiar with.

IFP intends to meet this challenge in the case of drilling management through the development of a new combined knowledge and data management system to enable day to day cooperative work between the various professionals involved on rig-site and in office. This system is developed under the name of WOG: "Well Optimization by Geosteering".

1 OVERVIEW OF WOG 3D AND ITS USAGE

The WOG 3D prototype is an Internet enable platform allowing specific and "on the shelf" software packages to interoperate onshore and offshore so as to optimize the drilling process. With the WOG 3D prototype interoperability of all data and processes are piloted by the meta-data.

1.1 WOG 3D Architecture

The global WOG platform is shown in Figure 2.

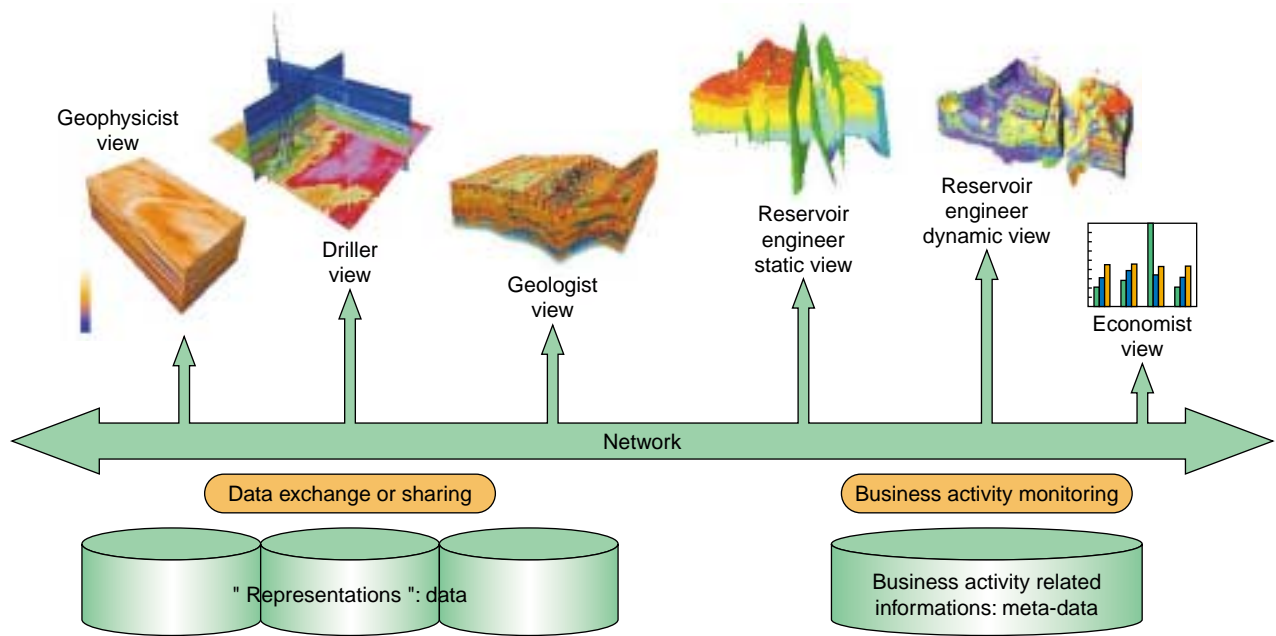


Figure 1

Data distribution through the network for knowledge sharing.

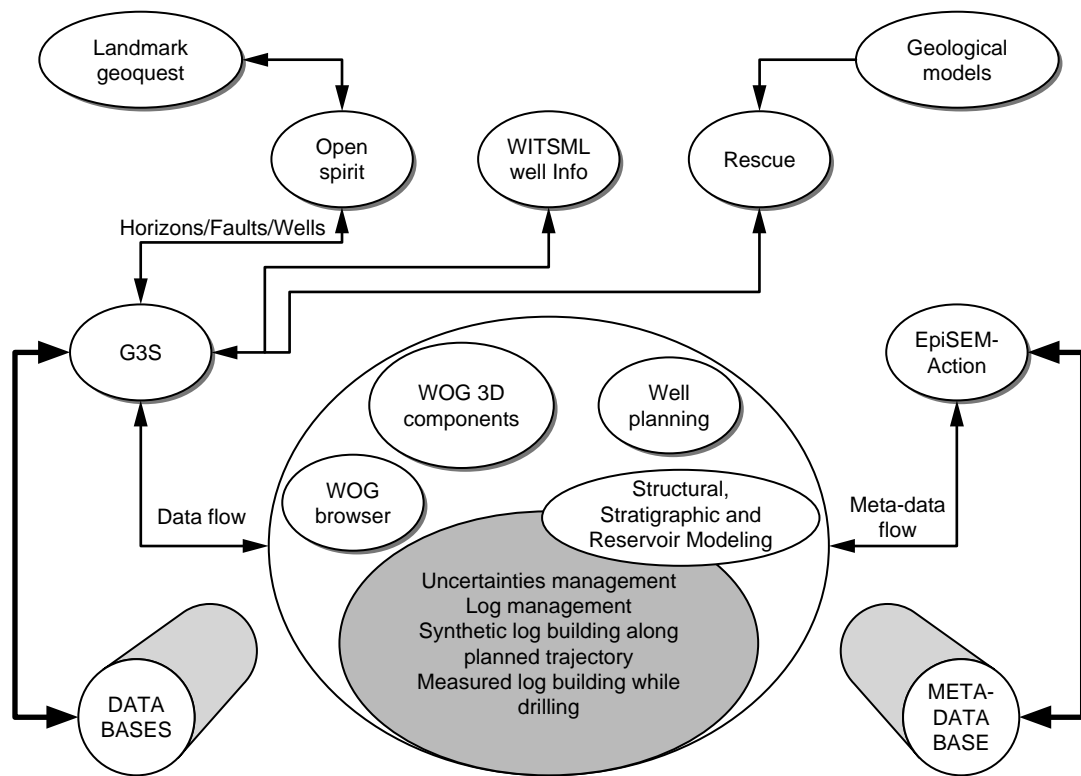


Figure 2

WOG 3D architecture.

WOG imports and exports data and models as standard Rescue files. This ensures compatibility with the existing software already developed at *IFP* and with the various geomodelling packages currently used in industry.

Using a methodology recently developed at *IFP*, under the name of 3D Geopilot, ensures automatic building of structural and stratigraphic models. It relies on an abstract description of the required geological knowledge (Geontology), which is presently being developed in the frame of the EpiSEM-Action project and on a codified representation of the geological interpretation which underlies the GES model (GES: Geological Evolution Scheme) to be built. The 3D Geopilot automatically computes a 3D model by deducing parametric links between the different layers and faults from the GES, then determining step by step the surface intersections that must be operated. The procedure is facilitated by using an efficient topology management tool (3D G-maps) developed at the *University of Strasbourg* and exploited by the *University of Poitiers*, France [4].

A major advantage of the Geopilot methodology consists in separating the geological level (GES) from the geometrical levels when building the earth model topology. In the course of a drilling process, new information could lead to revision at the geological level or at the geometrical level. The current 3D model can thus be easily updated by updating the GES or by replacing the previous estimation of the geometry, using a new estimation based on measurements. The consequences of the modifications at the topological and geometrical levels are then automatically deduced. This ability is also very useful to accommodate various geological interpretations and possible geometrical uncertainties, as well as providing a full set of “consistent earth models” starting from one set of non-structured geological surfaces (one “earth model” in the Rescue terminology).

The WOG 3D platform also provides important advantages for well applications. One important challenge is to take into account uncertainties, either related to the earth model or contained in the information generated while drilling. WOG 3D provides high-quality handling of the uncertainties, which helps the geologist and the drilling engineer to take the right decisions at the right time. Moreover WOG 3D enables targets to be chosen in a consistent earth model with the selection of various petrophysical attributes (multiple cutoff, connexity, volumetric and fluid circulation simulations). From this selection, the geologist and the reservoir engineer will be able to identify promising geobodies inside the geological model and to accurately define their location in terms of center, major axis and tolerances.

1.2 Usage of WOG 3D

WOG 3D is a well-planning package using conventional drilling constraints (torque and drag, casing design,

anti-collision) share data on the exploration targets in order to determine several reference well paths. Using the results provided by the reservoir engineer, the drilling engineer will then plan the minimum number of wells needed to reach all the targets.

When drilling, LWD and mud-logging information will be accessed on request from both locations: rig-site and office. The preprocessed data will be used to determine traces of geological feature boundaries and properties along the well bore paths. This interpreted information will be shared between the WOG 3D software modules located on rig-site and in the office (*Fig. 1*). Geologists and drilling engineers can each visualize the Earth model by using standard or 3D visualizations. Using this new interpreted information, the geologist will update the earth model and change, if necessary, the target locations or sizes in “real time” (*i.e.* depending on the daily progress in drilling).

In contrast to geologists and geophysicists, who are located in the office and are at all times connected to the network, the drilling engineer is a highly mobile user who is often operating on a PC which can be either connected and disconnected from the network, depending on the location. This situation requires an integration strategy that allows for the drilling engineer to work in either situation. The usage is a simple connect, disconnect, reconnect sequence. In this case, the management of the data flow is quite complex. WOG 3D will provide a process allowing the database used by the drilling engineer to be easily resynchronized with the database updated by the geologist. The drilling engineer will then be able to decide what use he wants to make of the updated target locations and can decide for instance to modify the well trajectory.

This type of process implies exceptional application interoperability. *IFP* has a solid background in this field as a result of its participation the Omega™ project and the present-day EpiSEM-Action project. In the context of these projects, *IFP* and its partners will focus on developing procedures which will help to generate interpreted information from measurements operated downhole (LWD) or at the surface (mud-logging).

2 PILOT SOFTWARE FOR SHARED EARTH MODELING (SEM)

WOG 3D intends to cover identified needs concerning updating of the 3D reservoir model in pace with the drilling progress (updating while drilling). It is thus an obvious candidate as a pilot tool for testing innovative ideas and modelling tools which may concern:

- abstract geological representations;
- automatic model building and updating;
- comparison with previous models and interpretative representation of numerical, spatial and topological uncertainties;

- cooperation between various experts (geoscientists, reservoir engineers, computer engineers, etc.) in distributed cooperative environments.

3 INTERNAL AND EXTERNAL COMMUNICATION

For a software package to be easy to use and easy to connect to other applications, the choice of adequate communication formats and modules is essential. The choice has been to base the development of WOG on two widely used tools:

- the Rescue format, which is a widely accepted standard for reservoir modelling;
- the *IFP* G3Server, which has been fully tested and is compatible with Rescue and EpiSEM.

3.1 The Rescue Format

Rescue is a standard for data exchange between geological models (grid, fault, well, etc.), that was developed in a joint industry project managed by the *Petrotechnical Open Software Corporation (POSC)*. The acronym “Rescue” stands for *REservoir Characterization Using Epicentre*.

The Rescue project started in 1995. Currently, six sponsors: *BP, Shell, Exxon/Mobil, Statoil, Total and Agip* are providing resources to organise work sessions, and to develop and maintain some utilities (validator, classifier) and the class libraries. *POSC* facilitates meetings, work sessions and project management. All vendors now provide within their packages facilities for writing or reading of Rescue files, at the very least for marketing reasons. The *IFP* group has been represented in the Rescue consortium from its foundation by *POSC* in 1996.

The Rescue class libraries remain under active development, the latest versions being restricted to use by project members. However, the Rescue group remains committed to an open standard, and a tested version of the class libraries is available to any nonmember company that wishes to develop Rescue compliant applications. Rescue members endeavour to maintain backward compatibility to the latest public version of the libraries.

Included within Rescue files are collections of horizons, faults, wells, grids, properties which correspond to a coherent interpretation step of the shared Earth model. This model is often topologically consistent and contains implicit geological knowledge.

Thanks to the Rescue libraries, it is possible to access within WOG representations of the various geological objects and their properties, and to understand the attached geological knowledge in order to further reuse it. For this, WOG uses services developed by the Open Geo project, which enable one to read/write the objects in a Rescue file and to publish them in the *IFP* data model using the G3Server Corba mechanisms described below (Fig. 3).

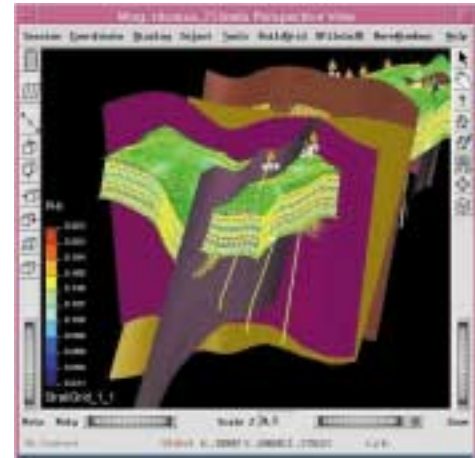


Figure 3

View of 3D business objects assembled into a Rescue file.

3.2 The G3Server

During the 1990's, *IFP* practice was to use a simple “*IFP* standard” file communication system between applications (the *IFP* Neutral Files). A first *IFP* communication model was implemented using the LibNF Libraries (C language), that were provided to all developers within the *IFP* group and to a few external partners (*i.e. Géovariances, Earth Decision*) to support the communication with *IFP* packages.

At the end of the decade, the emergence of new methods of communication and standards required the use of a more efficient technology. It appeared mandatory to ensure easy communication between the various *IFP* applications and between *IFP* applications and other vendor applications. This implied the design of a communication model based on defined standards, which would support a group of related distributed services.

The G3Server has been designed to facilitate data access, interoperability and exchanges between software and to allow examination of different views, designed to meet the interests of various users having different skills and possibly using different languages (C++, Java). It is based on a powerful Corba server, whose role is to distribute standardized *IFP* business objects between Java or C++ applications. The internal model (the *IFP* communication model) was designed in order to be easily used with Open Spirit and Rescue business data objects. Within WOG, a G3Server is used for distributing data between various application components. It can be called by business activities and can access Open Spirit servers, Rescue files and *IFP* proprietary business objects (*i.e. Reservoir Modeling Line entities (RML) BeicipTM* [5]). The domain covered is the well, reservoirs and seismic data.

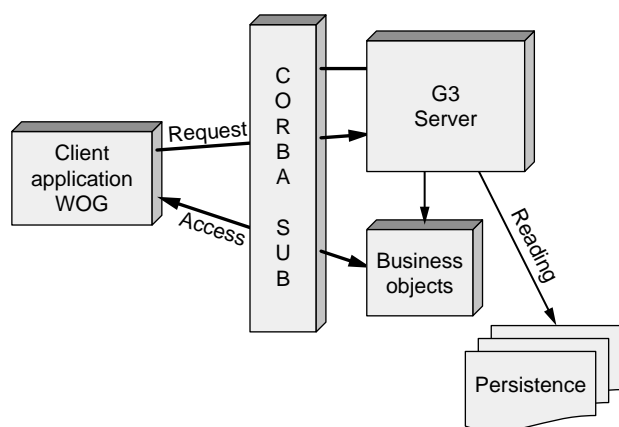


Figure 4
Server architecture.

The G3S package (*Fig. 4*) comprises a distributed (Corba) data model, server access to Rescue files, a graphic client, documentation describing the accessible services, the UML description of the data model and a set of best practices to explain how to connect a C++ or Java client. Some clients are also provided with: a Rescue file analyser, a Java client to browse in the model content, an event selection notification and a 3D viewer (Python Mayavi) which is a public domain tool.

The G3Server respects the Rescue, Open Spirit and EpiSEM-Action, standards. It provides a facilitated access to Rescue through an access interface (C++ or Java) and run-time collaborative services for sharing business objects between applications. It allows adding one's own server with its storage and provides standard access to many storage solutions.

The communication services provided by the G3Server are used by all earth modeling application components involved in WOG activities. G3S is not considered as an application by itself but is at the heart of all applications. It is notably very useful to interoperate between Java and C++ applications.

4 TOWARDS A TRUE SEM

4.1 Data-Driven vs. Knowledge-Driven Modeling

In the geosteering domain, there is a strong need to share information between various applications of different vendors coming from multiple disciplines. As a consequence, it is necessary to find an extensive and economic way to associate partners having various technological skills and various marketing strategies. Moreover, the information must be shared, in this case, over a much shorter time period than that of a complete reservoir life cycle. The challenge is thus to produce SEMs that can be quickly updated, extended, modified or rebuilt by any potential user.

3D geological models are particular geological representations and, as such, they are highly dependent on the interpretations made by the geologist using his expert knowledge. Geological interpretation operates at various stages in the workflow leading to decisions on which surfaces should be modeled, how they should be assembled and which relations they should have with the internal stratification within each block of the model [6].

The interpretation knowledge which underlies the geological model, or which is acquired during the model building or the drilling phase through interactive procedures, is implicitly shared inside "Earth modeling proprietary workflows". However, a current difficulty is that this knowledge is generally not stored as such. Hence, it cannot be easily reused by other applications for future model exploitation.

To present, the starting point for developing SEM has been the assumption that the original seismic or well data, as well as the resulting 3D models, are all graphic items, which can be described by identifying the elementary objects from which they are composed and by specifying their geometry and their topological relationships. To take a naïve example, in such a data-driven approach, a stratigraphical unconformity will simply be interpreted as a specific geometrical/topological assemblage consisting in a lower surface A interrupted by an upper surface B (*Figs. 5a and 5b*). Efforts

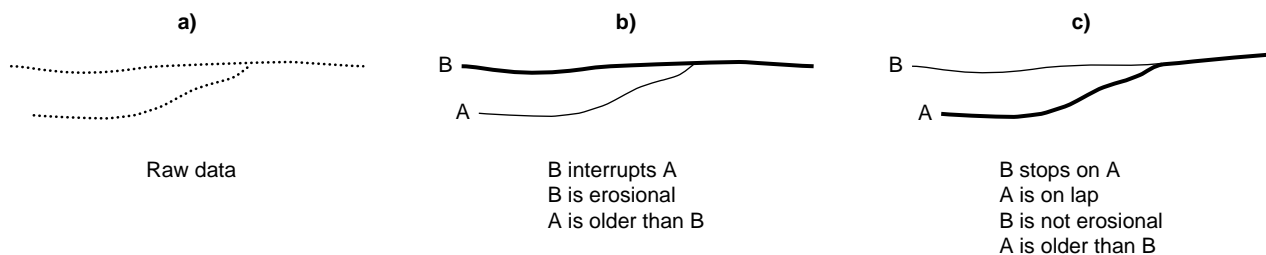


Figure 5
Relationships between geology and topology: changes in the geological hypotheses induce changes in the topology and surface identification.

to produce SEMs will then be dedicated to making sure than this particular geometrical/topological assemblage is correctly taken into account throughout the model building. However, this approach takes no account of the specific geological dimension of the models. For instance, there is no possibility of producing models in which surface B would no longer be erosional but onlap and where it would consequently be desirable to have B stopping on A rather than the contrary.

This difficulty can easily be overcome if the geometrical/topological relationships between surfaces A and B are no longer considered as an intrinsic feature of the model but as a consequence of the geological interpretation, *i.e.* of the added value brought by the geologist to the raw data. Surfaces A and B can then be considered as two geological objects linked by specific geological relationships. By assuming that B is an erosional surface younger than A, it is possible to deduce a topology in which B interrupts A. If we assume, on the contrary, that B is still younger than A (*Fig. 5*) but no longer erosional and that A is an on-lap surface, we will then deduce a topology in which A interrupts B. Such an approach can be considered as knowledge-driven since the model topology is the consequence of the geological interpretation *i.e.* of the specific features that the geologist has attributed to the various elementary objects in the model in view of his or her expert knowledge.

The production of knowledge-driven SEMs naturally supposes the sharing throughout the workflow not only of the raw data and of the various representations of the geometrical objects included in the model, but also of the geological interpretation itself. This supposes that such interpretation is represented by an adequate descriptor, specifically designed for capturing shared geological knowledge.

4.2 The EpiSEM Collaborative Knowledge Platform

For true interoperability between technical applications, one needs a clear understanding of the implicit knowledge captured in each application. The understanding, then the exchange of this knowledge is a topical problem in artificial intelligence. It can be solved by using ontology-based representations, which enable generic structured information attached to particular technical fields to be manipulated, so as to be reused by different kinds of applications. Formally, an ontology can be considered as a graph of classes (like an empty structured file or a template), which can be filled by ontology instances, which catch the real world information created by the user. Such formalisms can be used inside the WOG software by applying the results of the EpiSEM-Action project.

The EpiSEM-Action project is an EU project (IST-2000-30078) managed by *IFP*. It proposes a new approach to ensure collaboration between SEM applications via Internet or intranet and the capitalisation of value-added business

knowledge via a “knowledge centric approach”. In addition to *IFP*, the other participants in the project are *POSC*, *EADS*, *Shell*, *TNO* and the *Technical University of Berlin (TUB)*. The acronym “EpiSEM Action” stands for EPIcentre Shared Earth Model Activity Collaboration Through Meta data Interoperability Over the Net.

The EpiSEM platform is a powerful generic tool, which enables capture, visualization, exchange, sharing, management, integration, protection and reuse of knowledge acquired during an interpretation phase. It also facilitates the evolution of this knowledge during an exploitation phase.

The back-office EpiSEM-Action framework provides developer functions which enable the EpiSEM-Action platform to be adapted to possible domain ontology evolutions. It also provides a development Application Programming Interface (API) for programmers to allow them to write their own clients. All these operations are also accessible for platform administrators through a graphical user interface (EpiSEM-Action Swing UI client).

5 THE 3D GEOPLOT

The 3D Geopilot is an innovative knowledge-driven approach for building a 3D model.

5.1 Geological Syntax Applied to Modeling

Geological 3D models generally comprise large volumes of data and have a large size. For this reason, their building often requires significant computing delays and their revision, in case of data or interpretation changes, is generally a delicate and lengthy operation. This does not induce major difficulties in the case of studies concerning sites which are not the object of active prospecting at the time of the study, but it is no longer the case when active exploration drillings are performed. Optimal drilling trajectory guidance can then be decided only in view of 3D models updated step by step according to the data collected during the drilling progression itself.

The approach that we have taken considers that significant improvements to 3D geological modeling can be achieved by taking advantage of the specific structure, the “*Geological Syntax*” [6], which underlies consistent geological assemblages. Any geological assemblage represents a given “geology” which records underground evolution on geological timescales [7]. Specific processes took place during various spans of time, generally millions of years, inducing creation, destruction or transformation of matter. Each surface defined in the model is the record of one remarkable geological event, which can be considered as having been instantaneous with respect to the geological time scale. Consequently, before building a model, the geologist interprets the “geology” to be represented, by establishing a

total or partial order relationship between the various surfaces based on chronology. Specific links also exist between geological blocks. Each block is made of matter created during a given time span and thus belongs to a well-defined geological formation. The geological nature of the model influences its geometry. For instance, since geological surfaces limit volumes occupied by solid matter, two geological surfaces cannot cross each other. For this reason, a geological model will only be consistent if surface crossings always consist in one surface interrupting the other with no possible X-crossings.

Within WOG 3D, automatic model building is achieved by the “geological pilot”. This 3D Geopilot is also compatible with all currently used modelers. It includes two functions which do not exist in classical 3D geological modelers: the recording of all the geological relationships that the end user wishes to put in the model in a Geological Evolution Scheme (GES) and automatic building of the model, surface by surface, according to instructions read on the GES (Fig. 6).

The overall methodology involves the following four main operations:

- data input and GES building;
- determination of the intersections to be performed;
- fault preprocessing;
- model building (involving at each stage surface intersection and removal of all undesired parts).

Initial surfaces are assumed to be continuous, unstructured and geometrically independent from one another, possible intersections between them being ignored.

The GES must be built by the user before the building of the model itself. For this, the user has to define the relative age order of the various geological surfaces as well as the properties of each of them. He must also specify the links

between the various geological surfaces and the various files describing their geometries. This operation can be seen as the writing of the “legend” related to the 3D model to be built, which is necessary to understand the model structure in the same way as the legend of a geological map is a necessary complement to the map itself. The whole process is illustrated on Figure 7.

The building of the GES may appear to geologists as a difficult task and possibly as an unnecessary one as long as the model interpretation need not be shared with other users. One condition for the GES to be easily accepted by the modeling community, and to thus fully play its role, is that it can be easily built. For this, a user friendly “geological knowledge editor” has been included in WOG. This editor takes as inputs the geological characteristic attributed by the geologist to each of the surfaces in the model: nature of the surface (sedimentary interface, fault, etc.), relative age, chronospatial relationships with the other surfaces (for instance “erosional” or “onlap” in the case of a sedimentary interface) and automatically builds the related GES. We are also presently studying a methodology for building a GES in a semi-automatic manner from interpreted cross-sections (“GES builder”). For each of the various surfaces present in the model, the geologist will simply be asked to validate or modify proposals for geological attributes that will be automatically generated by the software. The GES’s related to the various cross-sections will then be built and, after detection and solving of possible inconsistencies, these 2D GES’s will be merged in order to generate automatically the GES related to the full 3D model.

5.2 Automatic Model Building

The partial order relationship established between the various geological surfaces by means of the GES is used inside the

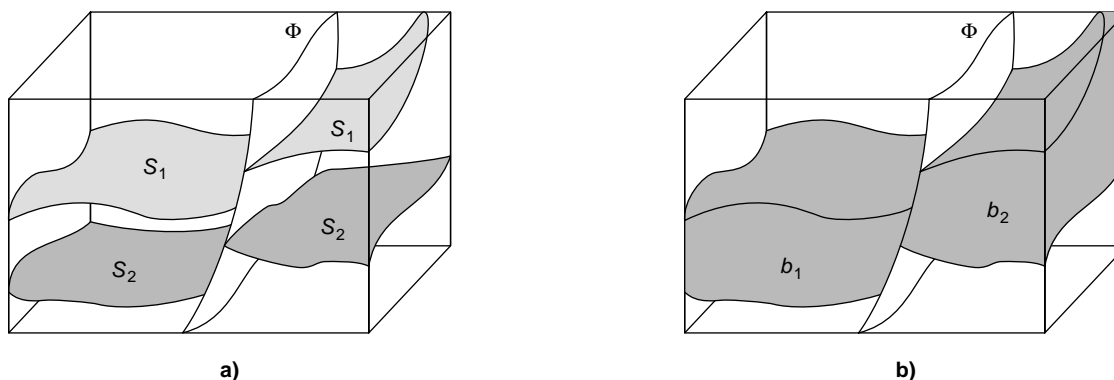


Figure 6

- a) S_1 and S_2 are two geological surfaces (S_1 younger than S_2) intersected by a later fault Φ ;
 b) b_1 and b_2 are two geological blocks between S_1 and S_2 which belong to one formation.

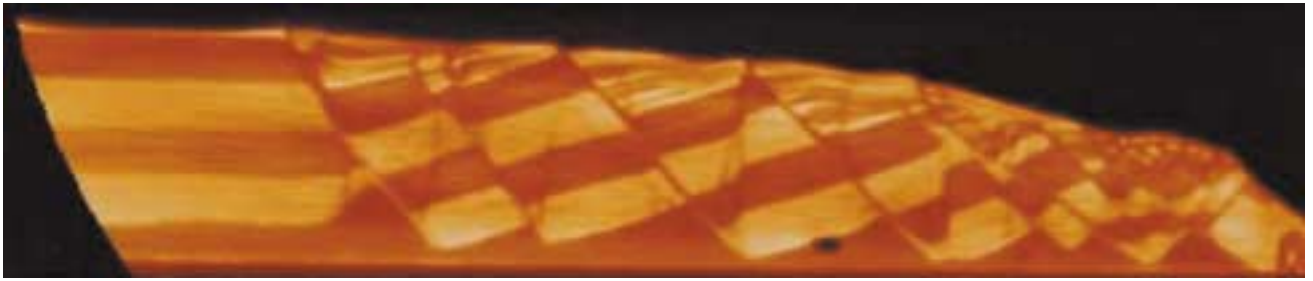


Figure 7a
Tomographic data.

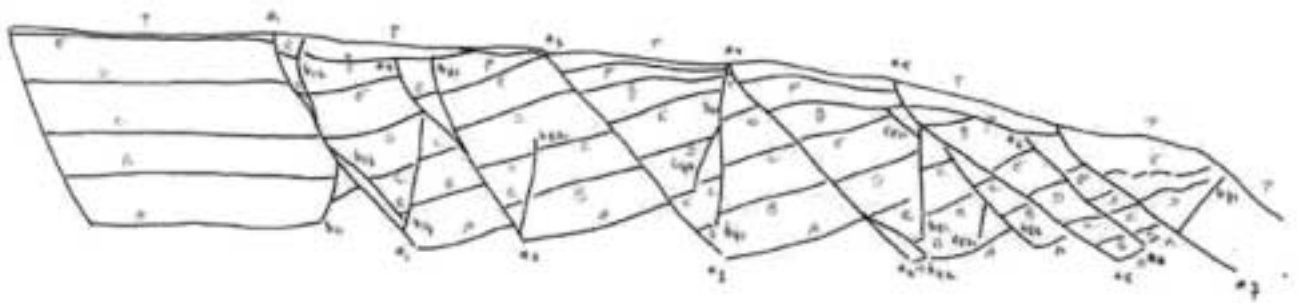


Figure 7b

Interpreted cross-section.

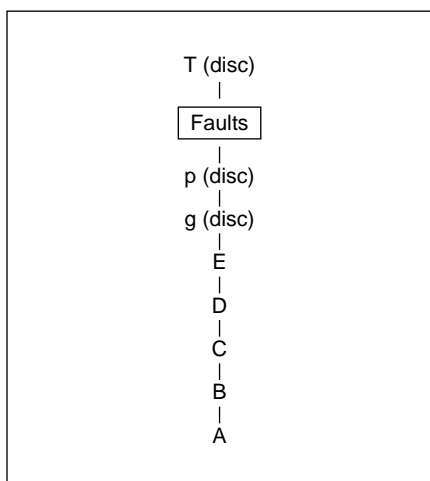


Figure 7c
Global GES.

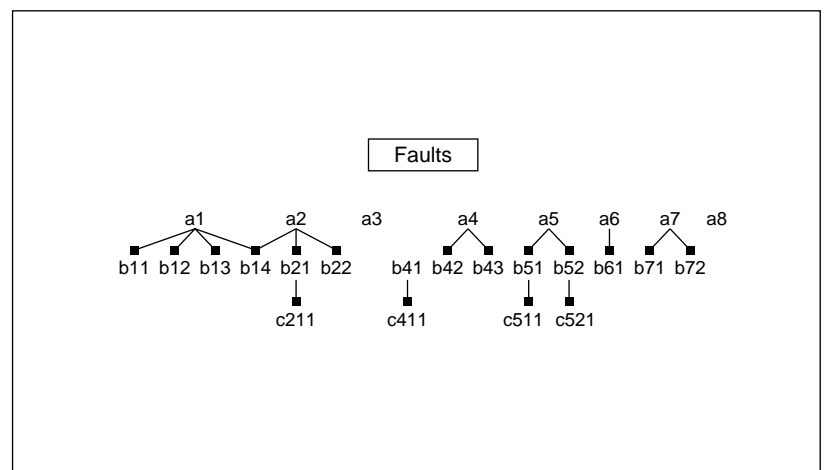


Figure 7d
GES of the fault network.

geological pilot itself to build step by step a consistent model. The rule is that an older geological event cannot modify a younger one. Thus, the various geological surfaces are introduced one after the other in reverse chronological order, beginning with the youngest. In order to obtain a consistent model at each stage of the process, each new geological surface introduced is intersected with all the younger ones already present in the model, starting from the oldest and ending with the youngest.

The model building is performed by corefinement of triangulated surfaces. This method uses triangulated surfaces that can easily be obtained with the WOG package. When the model to be built comprises only faults, a preprocessing stage is performed in order to remove geometrical uncertainties around faults. An offset solid is built around each fault in order to surround it by an uncertainty “pillow” volume (Fig. 8 and Fig. 9).

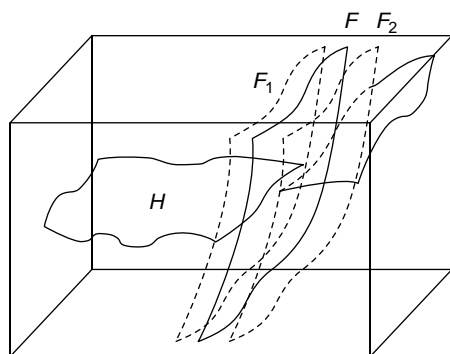


Figure 8
The preprocessing stage.

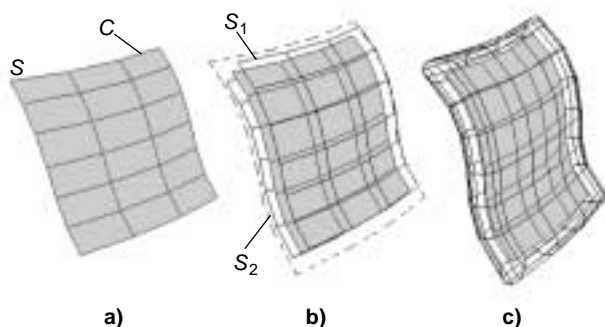


Figure 9
The offset solid is constructed by a) building two surfaces S_1 and S_2 parallel to S at a given distance d ; b) extending S to a contour C' at a distance d from the contour C of S ; c) closing the offset solid along C' .

For model building, the various surfaces are simply introduced into the model one after the other in an order corresponding to the downward interpretation course read on the GES. Each new surface n_i introduced is corefined with all the n_j surfaces already present in the model. This corefinement consists in computing the intersections between the various triangulated surfaces and in updating the microtopological model. When the corefinement procedure is completed, the portions of the surfaces which no longer belong to the model must be removed. This is done by considering the n_j surfaces one after the other, in an order corresponding to the upward intersection course read on the GES, and by removing in each case one portion of the surfaces n_i or n_j which is interrupted by the other.

When the model to be built comprises not only individual faults, but one or several fault networks, the model building procedure must be slightly modified. It then goes through the following steps:

- building of an offset solid around each fault of the network;
- corefinement of all the offset solids corresponding to a given network;
- removal of the parts of the offset solids which do not belong to the model; the parts to be removed are determined by examining, on the GES, which faults overlap on another.

The result of the above operation is a volumic fault network consisting in an assemblage of offset solids inside which individual planar faults are kept just as in the case of individual offset solids (Fig. 10). The volumic fault networks are then introduced into the model and treated in accordance with the procedure described above in the same way as individual faults.

The initial result of the modelling is a microtopological model (Fig. 11). However, the model building algorithm enables a **macrotopological** structure to be extracted (Fig. 12). This is the model the user will consider in practice to identify actual geological structures.

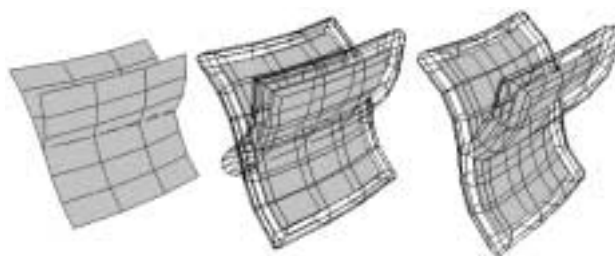


Figure 10
The offset solid built around a fault network.

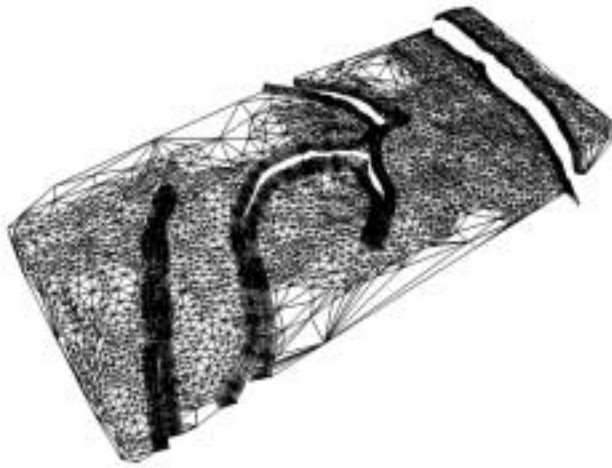


Figure 11
Microtopological model.

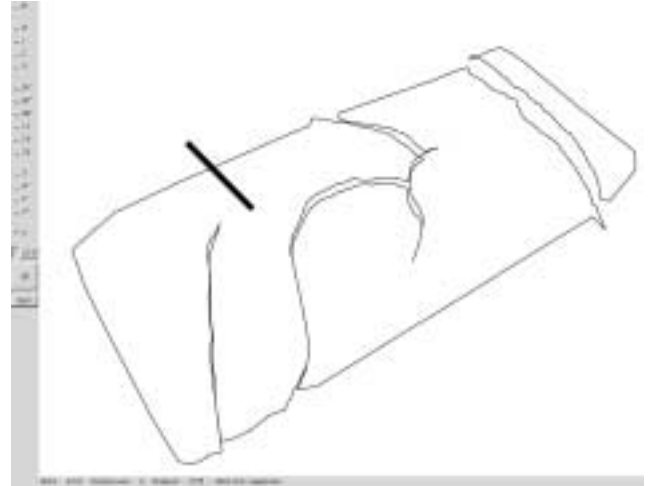


Figure 12
Macrotopological model.

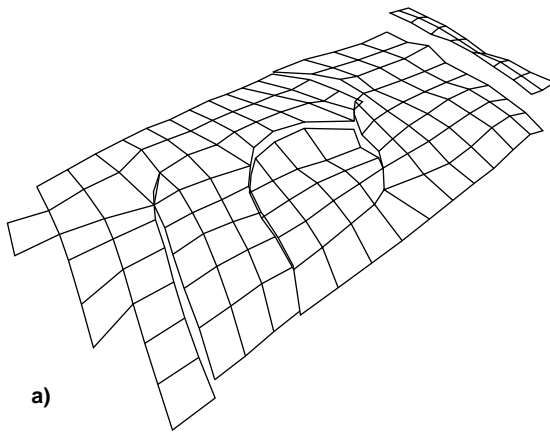
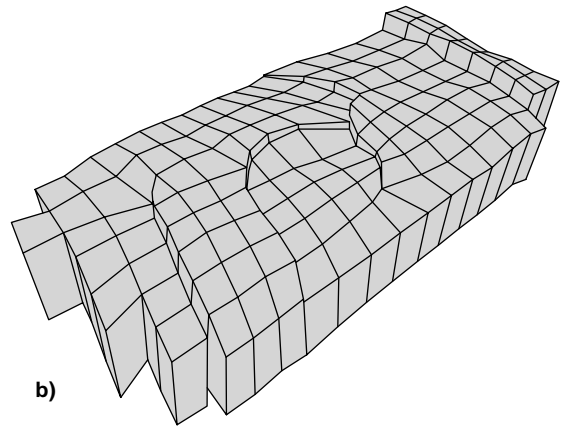


Figure 13
a) gridding of and b) meshing from the referential horizon.



The consequence is the production of models of better quality, which are fully consistent both topologically and geologically and which not only can be built but can also be updated or modified in a fully automatic way.

When a structural model has been built, it remains to fill the various volumes limited by geological surfaces with meshes and petrophysical properties. This can be done by applying the RML Modsim module. In addition, a new methodology is currently being developed to compute stratigraphical meshes directly from the macrotopological data structure attached to the 3D structural model computed by the geological pilot. This methodology involves:

- gridding of a referential horizon; the grid to be obtained must be tied to the discontinuities corresponding to the various fault traces (Fig. 13a);

- meshing from the referential horizon inside the topological volume defined by the macrotopological data structure (Fig. 13b).

This technique enables easy construction of rough meshes inside geological volumes, taking into account fault networks and discontinuities on horizons.

5.3 Examples of Results

As an example, we show here models built from one set of initial data.

This data set concerns tomographic data resulting from experiments operated on the analogical simulator of *Institut français du pétrole*. The data consists of two parallel surfaces corresponding to limits of sedimentary formations and five

surfaces corresponding to faults stopping on one another, which cut the two parallel surfaces into different patches shifted with respect to each other.

An unconformable surface posterior to the faults has been added to the original data in order to produce a model showing an erosion configuration. In this example, all the faults either stop on another fault or cross the entire model. The DSI data have been interpreted using the two methodologies above described.

The GES corresponding to this geology is shown on Figure 14. Figure 15 shows the results obtained by operating with the macrotopological approach and Figure 16 and Figure 17 show a result obtained with the microtopological approach.



Figure 16

Microtopology, DSI initial geology. Final model.

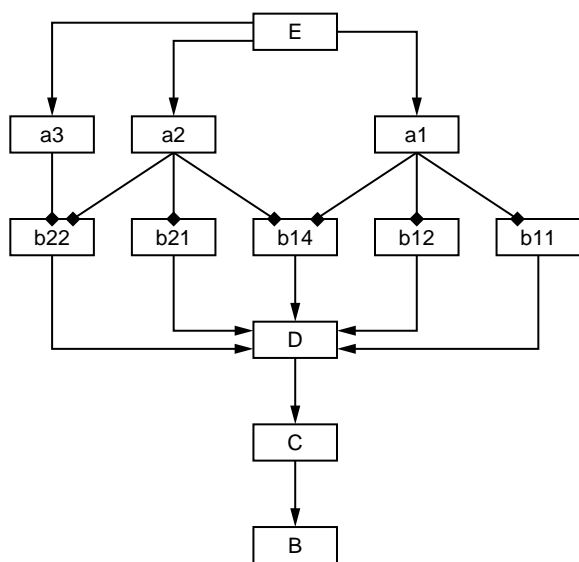


Figure 14

Geological evolution scheme.

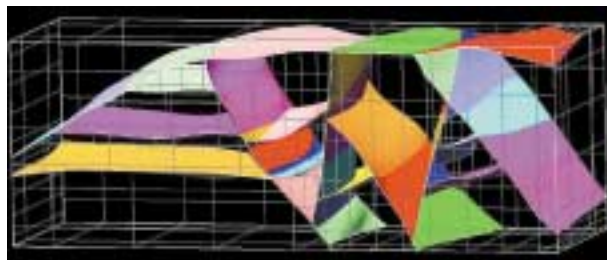


Figure 15

Macrotopology, DSI modified model. Final model.



Figure 17

Macrotopology, DSI initial geology. View of a geological block.

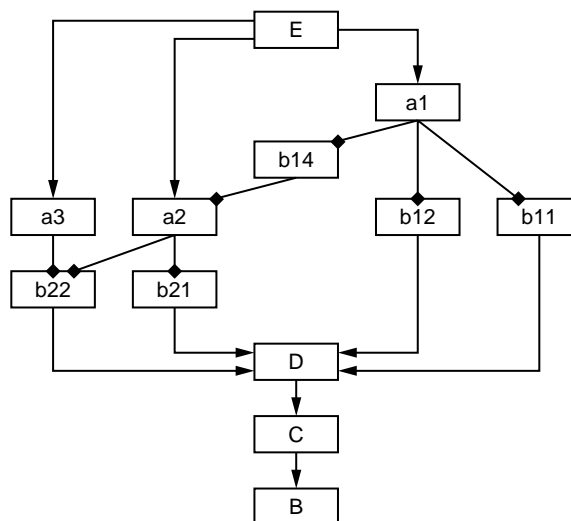


Figure 18

Modified geological evolution.

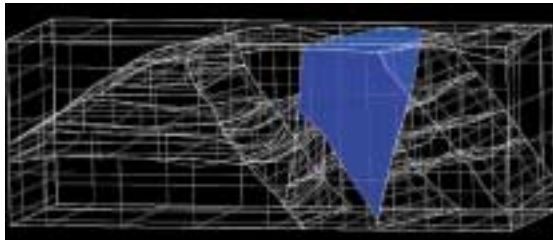


Figure 19

Macrotopology, DSI modified geology. View of a geological block.

The model can also be revised in the case of changes to the geological interpretation. Figure 19 shows result obtained by modifying the GES as shown on Figure 18. The new GES specifies that faults a1 and a2, previously interrupted by b14, are now interrupted by b14.

Figure 19 shows that the change in the GES has induced severe modifications in the topology. In particular, the shape of the block on Figure 19 differs near the fault on the middle of the scene. This is an outstanding result, the method that we propose being the only which allows such 3D topological modifications to be automatically taken into account.

6 WOG 3D WELL SERVICES

In addition to being an innovative tool enabling automatic model building and updating, WOG 3D also provides innovative services for representation of well incertitude and for target identification.

6.1 Well Trajectory Uncertainty Visualisation

For optimal well trajectory guidance, the WOG user must be able to visualize in 3D any planned well trajectory imported from a well planner such as DSP-One, together with the related uncertainties and also the estimated properties of the neighbouring lithostratigraphic sequences. Moreover, the users would like to deliver a record of the estimated properties along a planned well to other packages in order to calibrate drilling tool usage.

The simultaneous visualization of a well trajectory and of the estimated properties of neighbouring rock inside a WOG model supposes that the well trajectory can be positioned in the 3D mesh (Cartesian or CPG) in which the lithofacies or petrophysical properties are displayed, a problem that is in no way trivial. To reach a solution, the WOG 3D prototype proposes an extension of the well object of the RML Geosim/Modsim module. In this module, the user can import

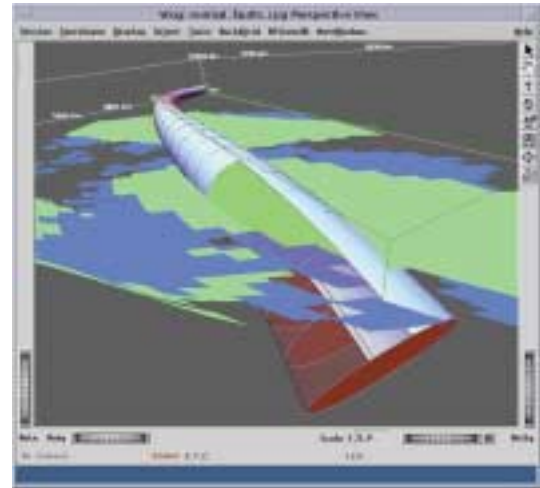


Figure 20

Well trajectory and its location uncertainty.

each WITSML well file generated by a well planning software (*TechDrill*) and visualize or hide for each imported well:

- objects related to the geomodel such as horizons, faults, lithofacies or petrophysical property grids;
- the well bore deviated survey and the original target collection attached;
- the ellipse of uncertainty (with or without transparency);
- an uncertainty cone joining ellipse boundaries (*Fig. 20*);
- a colour representation of lithofacies and/or petrophysical properties along the uncertainty cone or ellipses.

The well trajectory uncertainty visualization services will be applied after the planned well deviated survey computation to a geomodel grid populated with lithofacies and/or attributes handled by WOG 3D. This will be done both during the initialization phase and during further updates. In both cases, the applied workflow will be the following:

- well planning trajectory import;
- comparison with target locations;
- comparison with property grids;
- discretization along well trajectory;
- export of discretized properties.

6.2 Target Identification/Publication

WOG Target identification and publication are developed by *IFP* in the WOG 3D prototype to provide services for characterizing potential drilling targets and for exporting the results into a standard WITSML file (to be communicated to the well planner software: DSP-One a well planner software (produced and sold by *TechDrill*, a software company).

The target identification problem is as follows. Considering a given geomodel, consisting of various litho-stratigraphic formations or elementary sequences, whose lithofacies and petrophysical properties are estimated in 3D mesh cells (Cartesian or CPG), geologists and reservoir engineers need a flexible and easy to use tool to characterize 3D potential drilling areas.

In WOG, target identification can be achieved using the “attribute display” function, which is a filter operating with a simple “cutoff” function and by computing a “remaining connected volume” from the result. The filtered parameters are either original lithofacies or petrophysical properties estimated by using the RML Geosim/Modsim RML module or new attributes also computed by Geosim/Modsim. In addition, many other interactive functions help the user to select, orient and export a collection of targets in a WITSML standard format, which characterizes each target by its centroid location, dip azimuth and bounding box (*Fig. 21*).

The target identification/publication services are used in the initialization phase or during further updates after geomodel building or updating and prior to the planned well deviated survey computation.

The following workflow applies:

- cell selection by cut off operated on litho facies or properties;
- connexe components computation;
- geometrical attributes computation;
- filtering;
- target selection/identification;
- target adjustment;
- recording/export WITSML targets.

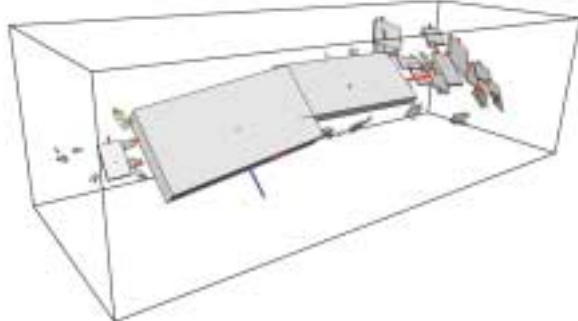


Figure 21

Target boundary extensions in 3D (representing the centroid of the connected volumes and their extension in 3D).

7 MODEL MANAGEMENT AND STORAGE

7.1 The WOG 3D Entity/Activity Browser

At different stages of a given project, and notably in its initialization phase, the end-user must be able to navigate in the EpiSEM catalogues and to define a workflow for launching various activities. This supposes the use of a navigator able to manage both business entities and business activities. This function is provided by the WOG browser, an autoconfigurable Java user interface which communicates only with the EpiSEM server.

The WOG 3D browser has been designed to be able easy integration of new entities (business objects) and new activities (applications/components) without specific development. The user's view is configured in accordance with nested (combined) business domain/generic workflow ontologies.

7.2 The EpiSEM Server

From the WOG viewpoint, we can see the geosteering process as a concurrent engineering process allowing separated software elements to cooperate on the Net. Since these elements can come from various sources and be related to various technologies, they have to share synthetic universally understood information *i.e.* XML (eXtended Markup Language) messages.

To provide the above services the EpiSEM-Action server must be:

- An XML messenger node between activities (an activity being defined as a functional application or a set of interconnected components). If a given activity has to talk with another one, the message must first be sent to the server, then distributed on the Net.
- A knowledge repository/publisher (a meta-data catalog) able to exchange structured information between activities (using a structured XML file).

Inside a given activity, data can be shared between applications and/or components through the *IFP* G3Server, which encapsulates the access to the data external world: Rescue files, Open Spirit Business Objects (*Landmark*, *Schlumberger*, *IFP* legacy storage). This supposes a permanent connection between the various activities and components and the EpiSEM server. This connection cannot be physical since activities have to be launched on request but must be logical.

7.3 Interoperable Management

WOG allows interoperable management of workflow, activities, data and meta-data.

The geosteering process is driven by users: a user request interactively commands the execution of an activity corresponding to an element of the activity catalogue. An example of request is described below.

A first solution to make applications/components interacting is the coding of complex activities within applications: a global complex activity will then execute one or more other business activities. This solution has at least one drawback since it is hard to get a view of the dependencies existing between the various activities. Moreover the applications code is likely to be hard to understand and to maintain. For this reason, we prefer building complex activities by capturing them with the help of workflow construction tools and by executing workflows with the help of a workflow engine. In this way, each activity need only know how to execute elementary applications and/or apply elementary components.

Let us now give an illustration of an actual workflow management.

Each activity of the workflow is defined as a Java method call that has the following skeleton:

- request the execution of an activity;
- wait for its termination.

The workflow engine itself can be considered as an activity. An example of execution of a workflow could then be the following:

- the WOG browser requires the workflow engine to execute a workflow;
- the workflow engine receives the request;
- the workflow engine downloads the description of the workflow;
- the workflow engine executes the workflow.

All along this process, the knowledge (on domain and workflow) is stored “somewhere in the net” in meta-data repositories (meta-data catalogues geographically distributed). While running the EpiSEM-Action server is able to deliver this information on internet to any type of WOG clients (Java or C++), running anywhere (on the rig or on offices). This facilitates the building of multiple workflows, which can invoke, in the future, not only the already developed components, but all components which could be useful to add.

CONCLUSIONS

With WOG, we can produce driller and geologist views before drilling, during the planning phase and while drilling, in order to optimize the trajectory before landing in the reservoir. This helps us to reach interesting targets with more accuracy, by taken into account the uncertainties on the drill bit trajectory and stratigraphical events location.

This result is obtained by monitored updates of the Earth model stratigraphy when the measured marker location information, obtained while drilling, doesn't match with

previous estimations. Our (PC based) platform can operate in, and distribute information to, geographically distributed locations (RIG and/or offices).

Every component has been developed with a generic approach, using standard communication tools (EJB, XML) and data standards (Rescue, WITSML), and is incorporated in an adaptive knowledge platform (EpiSEM-Action). This platform is able to record and replay applications or structured workflows. To date we have experimented, on this platform, several business oriented components such as entity and activity browsers, knowledge editors and automatic Earth model creation applications (such as the Geopilot 3D), able to share and use the geological knowledge to build and rebuild the model. To avoid creation of terabytes of data, this can be carried out only in case of drastic changes to hypotheses and/or geological object representations, induced by new measurement interpretations.

As we can see, today we are experimenting methods to rebuild the overall Earth model. Thanks to the generic nature of our platform, we can, tomorrow, add new functions: for instance, more accurate methods to update only locally, near well bore, the geometry and the property model on a fine scale. Our methodology also allows the system to be extended in order to produce additional types of expert view such as those for geophysicists, reservoir engineers, economists. All this views will contribute to enlarge the expertise, build new diagnosis tools and improve the decision processes in a domain in which every mistake or delay is paid for at a high price.

REFERENCES

- 1 Eschard, R. and Hegre, J.A. (2004) Workshop Process and Stratigraphical Modeling for Basin & Reservoir Evaluation”. *EAGE 66th Conference & Exhibition*, Paris, 7-10 June.
- 2 Bertotti, G., Buiter, S., Ruffo, P. and Scheurs, G. (2004) *GeoMod 2004 International Conference*, 9-11 June, Emmenten, Switzerland, Boll. Geofisica Teorica ed Applicat, **45**, no. and suppl.
- 3 Mallet, J.L. (1992) GOCAD: A Computer-Aided Design Program for Geological Applications, Three-Dimensional Modeling with Geoscientific Information Systems. *NATO ASI*, 354, 123-142, A.K. Turner, Kluwer Academic Publishers, Dordrecht, 1992.
- 4 Bertrand, Y., Dufourd, J.F., Françon, J. and Lienhardt, P. (1993) Algebraic Specification and Development in Geometric Modeling. *Proceedings of TAPSOFT'93*, BDF*93, April.
- 5 <http://www.beicip.com>, BEI, the *BEICIP Web site*.
- 6 Perrin, M. (1998) Geological Consistency: an Opportunity for Safe Surface Assembly and Quick Model Exploration. *3D Modeling of Natural Objects, A Challenge for the 2000's*, **3**, 4-5, Nancy, France, June 1998.
- 7 Hobbs, B.E, Means, W.D. and Williams, P.F. (1976) *An Outline of Structural Geology*, Wiley and sons.
- 8 Brandel, S., Haeefe, M. and Bechmann, D. (2003) A Geological Application in Immersive Virtual Environments.

- VRIC 2003 5th Virtual Reality International Conference, Laval, France, 13-18 May 2003.
- 9 Brandel, S., Perrin, M., Rainaud, J.F. and Schneider, S. (2001) Geological Interpretation Makes Earth Models Easier to Build. *EAGE 63rd Conference, Extended Abstracts*, 1: F-28, Amsterdam, The Netherlands, June.
 - 10 <http://www.opencascade.org>, the *CAS.CADE Web site*.
 - 11 Cazier, D. and Dufourd, J.F. (1996) Opérations booléennes robustes définies sous forme de systèmes de réécriture. *AFIG'96*.
 - 12 Cazier, D. and Dufourd, J.F. (1996) Dérivation de systèmes de réécriture pour construire efficacement des subdivisions planes. *Journées du GDR-PRC de programmation*, Orléans.
 - 13 <http://www.dgi.com>, the *EarthVision Web site*.
 - 14 Floater, M., Halbwachs, Y., Hjelle, O. and Reimers, M. (1998) OMEGA: A CAD-based approach to Geological Modelling. *Modelling 98 Conference*, Nancy, France, June.
 - 15 <http://www.gocad.com>, the *GOCAD Web site*.
 - 16 Halbwachs, Y. and Hjelle, O. (2000) Generalized Maps in Geological Modeling: Object-Oriented Design of Topological
 - Kernels. *Advances in Software Tools for Scientific Computing*, **10**, 1, 339-356.
 - 17 <http://www.lgc.com>, the *Landmark Web site*.
 - 18 Lienhardt, P. (1989) Subdivision of N-Dimensional Spaces and N-Dimensional Generalized Maps. In: *5th ACM Symposium on Computational Geometry*, 228-236, Saarbrücken, Germany.
 - 19 Lienhardt, P. (1994) N-Dimensional Generalized Combinatorial Maps and Cellular Quasi-Manifolds. *Journal of Computational Geometry and Applications*, **4**, 3, 275-324.
 - 20 Mallet, J.L. (1989) Discrete Smooth Interpolation in Geometric Modeling. *ACM-Transactions on Graphics*, **8**, 2, 121-144.
 - 21 Mallet, J.L. (2002) *Geomodeling*, Oxford University Press.
 - 22 <http://www.posc.org/rescue>, the *File Exchange Format RESCUE Web site*.
 - 23 Schneider, S. (2002) Pilotage automatique de la construction de modèles géologiques surfaciques. *Thèse*. Université Jean-Monnet et École des mines de Saint-Etienne.

Final manuscript received in July 2004

APPENDIX 1

An example of Cooperation within WOG: Request of a GES Building Using the Wog Browser and the Geological Knowledge Editor (GKE)

The following scenario describes the way one can use the WOG browser to require from the GKE (Geological Knowledge Editor) the building of a GES (Geological Evolution Scheme) and its adding to the browsed catalog. The example also shows that the WOG browser correctly updates the content of the catalog it is browsing.

1. User1 launches the WOG-browser.
2. The WOG browser “connection window” (W1) appears (*Fig. A1*).
3. User1 sets the address of the EpiSEM server, his connection name and password.
4. User1 closes W1 by clicking on the button “OK”. The connection parameters are stored in a configuration file.
5. A window (W2) containing the list of catalogs stored on EpiSEM appears (*Fig. A2*).
6. User1 selects a catalog.
7. User1 closes W2 by clicking on the button “OK”.
8. The Concept Container window (W3) appears (*Fig. A3*). It displays hierarchically the content of the catalog.
9. To request the construction of a GES, User1 right clicks on the root node: a menu appears. User1 clicks on the item “BuildGES”.
10. The “parameter setting” window (W3) appears. User1 sets the parameters of the request.
11. WOG browser sends a request of construction of a GES to GKE.
12. EpiSEM server receives the message. As there is no launched GKE, it adds the message to the lost message queue.
13. EpiSEM client launcher reads the message and launches the GKE. The GKE uses the same connection parameters (cf. item 4) as those used by the WOG browser.
14. GKE connects to EpiSEM server and reads the request sent by the WOG browser.
15. GKE imports the catalog.
16. GKE builds an empty GES.
17. User2 edits the GES (*Fig. A4*).
18. User2 requests from GKE to upload the GES.
19. GKE uploads the GES and sends a notification message to all the EpiSEM clients.
20. WOG browser receives the message and reloads the catalog.



Figure A1

Connection parameters.

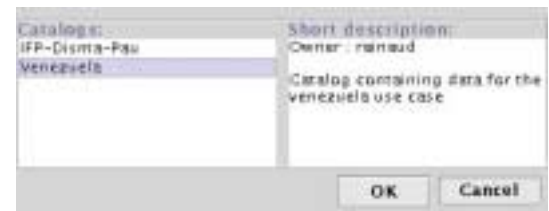


Figure A2

Catalog choice.



Figure A3

Hierarchic Display of the content of a catalog.

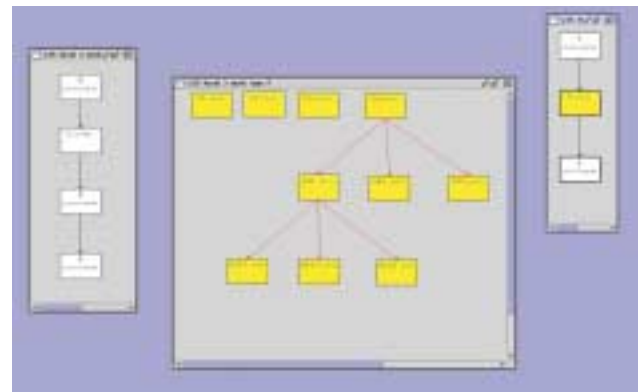


Figure A4

Geological evolution scheme displayed a graph in the GKE.

APPENDIX 2

Description of Some WOG Activities

The following table (*Table A1*) enumerates the activities used to construct an Earth model starting from a Rescue file. It describes the activities and specifies the application that knows how to execute them.

TABLE A1
Description of the activities

Activity	Application	Description
BuildCEM	WOG browser	Build a conceptual model that contains all the features contained in a Rescue file
BuildGES	GKE	Build a geological evolution schema that contains features stored in the conceptual model
BuildEM	WOG 3D	Build an Earth model by interpreting a GES. The built Earth model is defined by a set of formations.

As we explained previously, an application requests the execution of an activity by sending a message. The message contains the classification of the activity and the information needed to execute it. Of course, the application that receives the request must know how to fulfill it.

The fact that the message contains information necessary for the execution of an activity does not mean that the execution is fully automatic. For example, the user is requested to specify the Rescue file, during the execution of the activity **BuildCEM**. At the end of the execution of the activity, the application B constructs an XML file that contains information concerning the activity (start-end time,) and the evolution of knowledge (e.g. add FeatureOpinion to Feature etc.) during the execution of the activity. B uploads this file to the EpiSEM server and notifies A of the end of the activity.

The following table (*Table A2*) describes the information necessary to execute the three activities.

TABLE A2
Parameters needed for the execution of the activities

Activity	Parameters
BuildCEM	CatalogName: name of the catalog to build OntologyURI: URI of the ontology to associate to the catalog
BuildGES	CatalogURI: URI of the catalog GESName: name of the GES to build
BuildEM	CatalogURI: URI of the catalog GESName: name of the GES used during the interpretation EMName: name of the Earth model to Build

Links

- **LibcUrl**: tool for transferring files with URL syntax <http://curl.haxx.se>
- **Jawe**: Java workflow editor <http://jawe.objectweb.org>
- **Shark**: Enhydra shark workflow Engine project <http://shark.objectweb.org/>
- **GSoap**: generator tools for coding SOAP/XML Web Services in C and C++
<http://www.cs.fsu.edu/~engelen/soap.html>

APPENDIX 3

EpiSEM-Action Assets

The EpiSEM-Action project publishes the developed platform and its documentation on a portal, which can be accessed in the Internet. An overview about the used libraries from 3rd parties is found in a “Framework Design” design document. Additionally a set of scripts for compilation, packaging, and deployment is provided. The structure of the software project and the used tools are oriented to commonly known open source projects like those from the *Jakarta Group* of the *Apache Software* foundation. Java developers that have already experience with open source project should be able to get familiar with this software very quickly.

The source code in particular is divided into for main packages—three containing the implementation of the platform and one containing the software-based tests. In summary:

- **org.episemaction.client**: This package contains the implementation of a client for administration of the platform, custom connection proxy objects (for use with the Web service based communication with the server), and supplemental utilities for other domain dependent implementations.
- **org.episemaction.server**: This package contains the functional implementation of the EpiSEM-Action platform. This contains the management of the catalogue, the management of users, projects, and models. Also provided is the functionality for tracking the evolution of the catalogue or notification, when certain elements of the catalogue where changed.
- **Test**: A separate package contains different test implementations for testing the packages inside **org.episemaction**. This package offers unit tests for client an server implementation as well as tests, which ensure the functionality as described in the use cases.
- **org.episemaction.knowledge**: For the communication of the sophisticated data structures between client and server some specific implementation like a common object model is provided in this package, which do server and client use as well.

When deployed, The EpiSEM-Action platform comprises services to:

- edit/create ontology with the KAON graphic editor;
- configure automatically a data base schema using this Ontology;
- administrate this data base on the Web (definition of roles, authentication);
- use the data base to import/export instances on the Web;
- manage notification (by messaging) and evolution of instance.

Other services as dynamic connection facilities to end user activities are currently in development and will facilitate a lot

the hanging of new application to an EpiSEM-Action platform.

The EpiSEM-Action platform is used to orchestrate the geosteering process on the Net. The instances of pre-condition and postcondition of all activities are recorded, and in case of interesting update, all activities involved will be notified when a change occurs. This ensures that the data and knowledge acquired or modified during the whole drilling procedure, can easily be secured, integrated, and managed for further reuse at any time. during the whole. The reservoir features can thus be considered as generic evolutionary knowledge, that can be used and modified at any time.