62063

MINISTERIO
DE MEDIO AMBIENTE

Instituto Tecnológico
GeoMinero de España

INFORME	Identificación:
	Fecha:

TÍTULO

GEOMIST

PROYECTO

Geological and Mining Information System on the Iberian Pyrite Belt

RESUMEN

La información geológico minera sobre la Faja Píritica Ibérica disponible en el ITGE, IGM (Portugal) y las compañías mineras operando en la zona, se ha integrado en un servicio de información en Internet.

El servicio de información está basado en servidores instalados en Madrid y Lisboa, a través de los cuales se puede acceder a los servidores de bases de datos en el ITGE e IGM.

Los servicios disponibles en el servidor son: consultas a la metabase, consultas a las bases de datos y servidor de mapas interactivo. Además se ha desarrollado un demostrador en 3D modelizando sondeos de exploración minera

* continuar al dorso en caso necesario

Revisión

Nombre

Luis Delgado L. Delgado

Unidad

Planificación y Gestión

Fecha: 28.4.2000

Autores: F. Herrera, L. Delgado, J. R. Hernández Manchado, G. Ortíz

Responsable: L. Delgado

CORREO ELECTRÓNICO
l.delgado@itge.mma.es

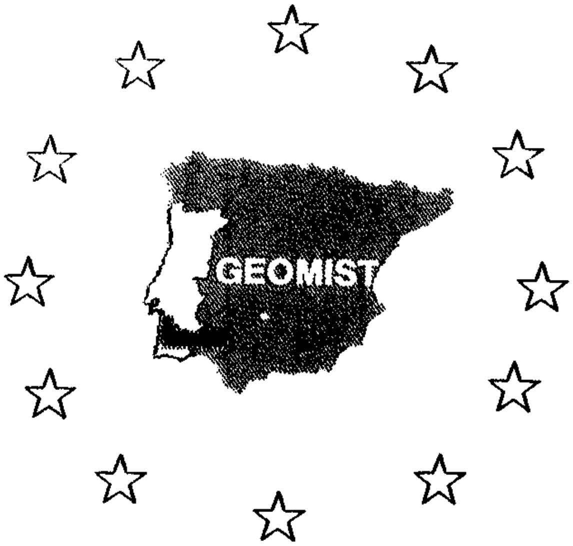
Rios Rosas, 23
28003 MADRID
Teléfono 91 349 59 68
Fax 91 349 59 29

62063

Geological and Mining Information System on the Iberian Pyrite Belt

GEOMIST

FINAL PROGRESS REPORT



L. Delgado, F. Herrera, R. Hernández, G. Ortiz, M.P. Moreno – ITGE
L. Torres, J. Lopes – IGM
B. Lesecf – MTDV

September 1999

General Management

Task 01. General and technical management of the project plan to manage and coordinate the team work, supervise the allocation of resources and take appropiate actions for the timely availability of deliverables and reports. Organise links and liaisons to related projects GEIXS and OMEGA.

Key points

The third GEOMIST Review Meeting was held in Brussels on 28 Jannuary 1999 at the premises of EuroGeoSurveys.

On 4 February 1999, a technical meeting between ITGE and Junta de Andalucía in Spain, was held in Sevilla to try to encourage user feedback to the developments achieved within GEOMIST. A workplan was agreed to involve more deeply to Junta de Andalucía and Spanish companies in future developments, including the establishment of communication links between SIGMA (Information System of Junta de Andalucía) and GEOMIST, agreements on the management and administration of the distributed system and general procedures to regulate user access to the system.

On 7th, 8th and 9th April 1999 technical meetings betwen ITGE, IGM, BEICIP FRANLAB (OMEGA developper) and MTDV were held in Paris to proceed with the geological validation of the 3D model and to agree on the final specifications to be implemented in the 3D demostrator.

The fifth GEOMIST Technical Meeting was held in Lisboa on 28-30 June 1999 with assistance of the technical teams of ITGE and IGM. A common set up of the retrieval applications for data, metadata and maps of the GEOMIST web server at ITGE and IGM was achieved.

A decision was taken on the selection of a new data set for the development of the 3D model in an area whit higher density of data allowing for a better extrapolation of the lithologies.

After this meeting the 3D model was validated and agreements were reached on the work program and scenario to build up the 3D demostrator, that has now been completed.

Good progress has been made within the more critical task to achieve a valuable and visible results. i.e. the GEOMIST Website. A reliable and demostrable web server is now in operation.

Actions were also taken to react to the recommendations made by the Commission and experts evaluators during the third project review on:

- Definition of realistic plans to complete the project

- Definition of clear and specific plans for future activities beyond the project

- Involvement of users and government agencies

A description of the specific tasks and actions taken within each work package is summarised below.

Summary of work carried out

Work Package 1. Data Inventory and Modeling

Complete fulfillment of this workpackage has been reported before showing the completion of the scheduled tasks T1.1 Data Inventory. T1.2 Data Model and Data Dictionary and T1.3 Harmonization tools.

The establishment of an unique data model for the selected data bases and of harmonisation tools to harmonize heterogeneous and dispersed data, was he first step to build-up and homogeneous information service.

The plan to complete and update data beyond the project completion into the GEOMIST web server in given in Annex 1.

Work Package 2. Redesign of Data Bases and GIS

The completion of task T2.3 Data loading and validation, has been the last step for the fulfillment of workpackage 2.

Therefore the same harmonized data sets have been loaded and validated at each data server at ITGE and IGM.

Cartographical data are stored under ARC/INFO while for the factual data ORACLE DBMS has been used.

Work Package 5. 3D representation and retrieval applications on geodata

A realistic fulfillment of workpackage 5 has been finally achieved, through the implementation of the 3-D applications, on the new selected data set presented in Annex 2: 3-D Data Set.

The final developments were made by using Java 1.2., CAS-CADE 1-5 B and OMEGA products on a SUN SOLARIS platform, in order to avoid delays on the availability of some tools on NT platforms.

All the pending tasks to complete the development of the 3D applications, have been terminated, including:

- Building of interpreted cross sections for model building understanding

- Correlation between geological surfaces and polygon files to extract topological faces

- Transcription of cross sections into the 3D model

- Volumetric modeling

- Completion of horizon surfaces

- Development of the agreed demostrator functionalities

- Validation of the 3D geological model

The 3d demonstrator is able to generate VRML files from gr2d, well and polygon files in ASCII OMEGA format, so that a 3D visualization of the geological model can be displayed through the Internet.

A complete description of the work done, including the description and codes of the OMEGA file formats and software tools used, as well as on the 3D data model, 3D demonstrator architecture and 3D demonstrator user guide is given in Annex 3: 3D Application Programs.

Work Package 6. Build up and Validation of the Web Server

The GEOMIST Web Server has been completed. The Service in based on web servers installed at ITGE (Madrid) and IGM (Lisboa).

The respective URL are:

http://itge.mma.es/geomist
http://geomist.igm.pt

General information on the GEOMIST partners as well on the project, the geographical area, i.e. the Iberian Pyrite Belt is given at the GEOMIST home page.

The GEOMIST interactive services include:

- Metadata Query

- Data Base Query

- Interactive Map Server

The description of these interactive services as well as on the System Security is given below.

METADATA SERVICES

According to the conceptual schema for metadata, defined in the European Standard prEN 287009, prepared by the Technical Committee CEN/TC 287 "Geographic Information", metadata is defined as "data about datasets" including data about the content, representation, extent (both geometric and temporal), spatial reference, quality and administration of a geographic dataset.

Usually, such geographic data are organised into datasets, containing one or more geographic objects described by their properties and associations. A dataset may consist of a combination of other datasets.

There is a set of main elements of metadata used to describe geographic datasets:

1. **Dataset identification**
   Sufficient data provided to clearly identify the dataset, such the dataset title or alternative title.

2. **Dataset overview**
   Data to give an overall description of the dataset, including a brief textual description of the dataset, usage, spatial sub-schemas, spatial reference system type, document references or related datasets.

3. **Dataset quality elements**
   Quality data allows to define how well a dataset meets its nominal ground, such overall positional, thematic and temporal accuracy.

4. **Spatial reference system**
   Describes how geographic objects are positioned in space. The spatial reference system may be either direct, based upon a co-ordinate system, or indirect. A dataset may contain objects referenced by multiple direct and indirect spatial reference systems.

5. **Extent**
   Although geographic datasets vary in their planar, vertical and temporal extent, this model only describes the planar extent of each dataset through a bounding area. There is coverage, named *FPI_Limits*, limiting the area which is covered by all datasets.

6. **Data definition**
   Geographic datasets may be defined in different ways in different datasets. To allow easier comparison between them, each one is provided describing those characteristics of geographic objects used for:
   - Distinguishing among classes of geographic objects, called object type
   - Distinguishing among geographic objects belonging to the same class, called attributes
   - Describing relationships between classes of objects, called associations

7. **Classification**
   A general description of the content of a dataset.

8. **Administrative metadata**
   Data describing how to physically acquire a dataset, how a dataset is held and how it may be transferred, if required. Also explain about the organisation, the organisation role and point of contact.

9. **Distribution**
   This data is about the distribution of the dataset, such restrictions on use, copyright owners, price information, unit of distribution, data media, formats or support services.

10. **Metadata reference**
    It is necessary to include data about the metadata themselves. This include the entry date, last check date, last update data or language.

The GEIXS metadata is based on the previous schema and expands it to take into consideration geological data. To make it easier to implement it a stand alone Windows Access application was developed by the GEIXS team, for its own use. Geomist adopted the GEIXS metadata schema but not its metadata entry application. The solution adopted is described bellow, and the reason behind the choice is the need for a tighter connection between the database and metabase administration, as well as a better overall integration. The implementation is based on Oracle.

DATA MODEL FOR METABASE

As defined above the metadata describing geographical datasets includes various types of data. This may also be classified in 'static' metadata and 'dynamic' or structural metadata. The first type characterises the global dataset and this data provides fixed knowledge about the dataset, whichever it is. On the other hand the structural metadata checks the structural definition of a dataset and reports a different view for each dataset, reflecting the changes it suffered.

According to the set of types defined above, the structural definition corresponds to the Data definition and the static metadata corresponds to the remaining types.

Follows the data model for the metadata base. The objects involved are subdivided in four types: System objects - corresponding to the system tablespace created to archive data; Main objects - storing data about the main entities characterised; Dictionary objects - with auxiliary data used to classify the main objects; and Ancillary objects - created from the relation with the other objects.

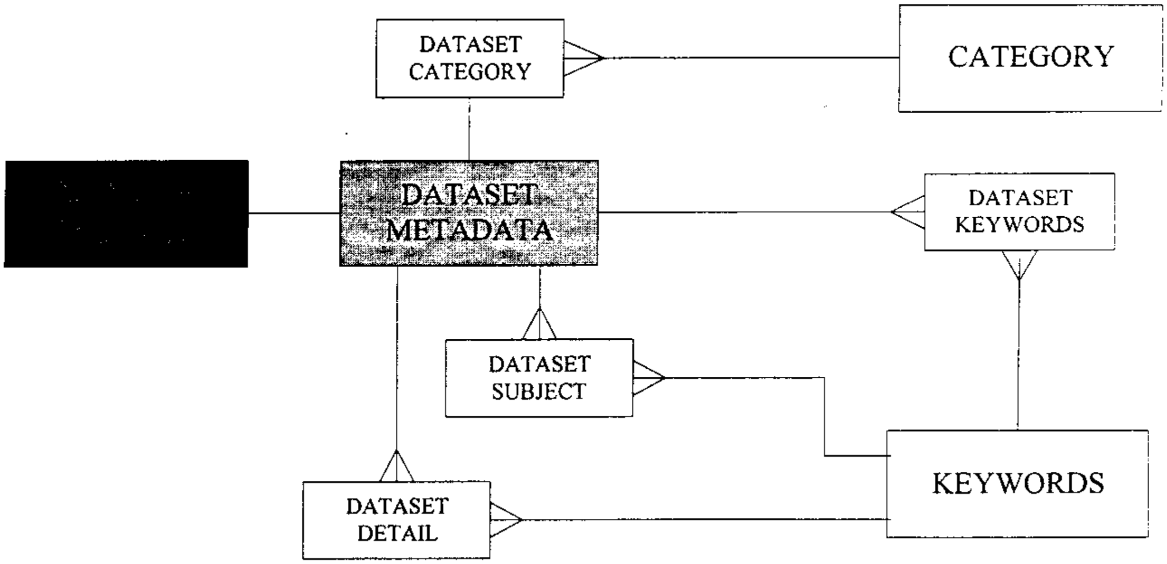


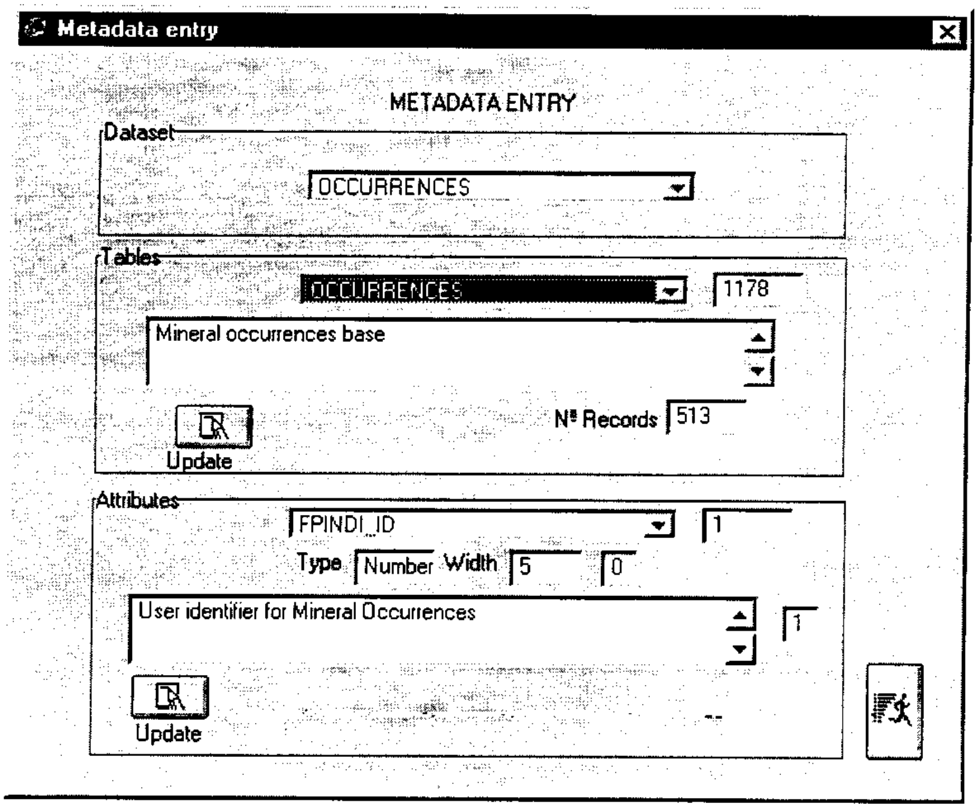
Fig 1 - Data model for metadata base

METADATA ACCESS

The entry of metadata is performed through an interface to the database management system which validates, stores and provides all data. To insert and update data there is an interface, designed to use under the Windows platform. To inquiry the system about metadata one uses a service available as a Geomist service in the Web environment.

1. METADATA ENTRY

There are two distinct levels to metadata entry, both through an especially designed interface .



Metadata entry

METADATA ENTRY

Dataset: OCCURRENCES

Tables: OCCURRENCES 1178

Mineral occurrences base

Nº Records 513

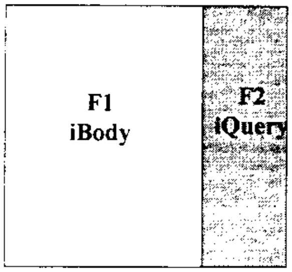
Attributes: FPINDI_ID 1

Type Number Width 5 0

User identifier for Mineral Occurrences 1

2. METADATA QUERY

On start up the process launchs a new window divided in two distinct frames: one is to be used to select the dataset from a list. The dataset metadata is displayed in the other part of window.



On database server one dataset corresponds to a tablespace with specific datafiles and data control files. Each tablespace has tables and tables associations in ways to implement the relational model defined in Entity Relations Diagram. When the query sub-process is launched the system look for datasets alive at Oracle instance database. This information is stored in system table named sys.tab$. To the advantage of this system query the tablespace identifier is also returned. That is a simple Structured Query Language (SQL) expression:

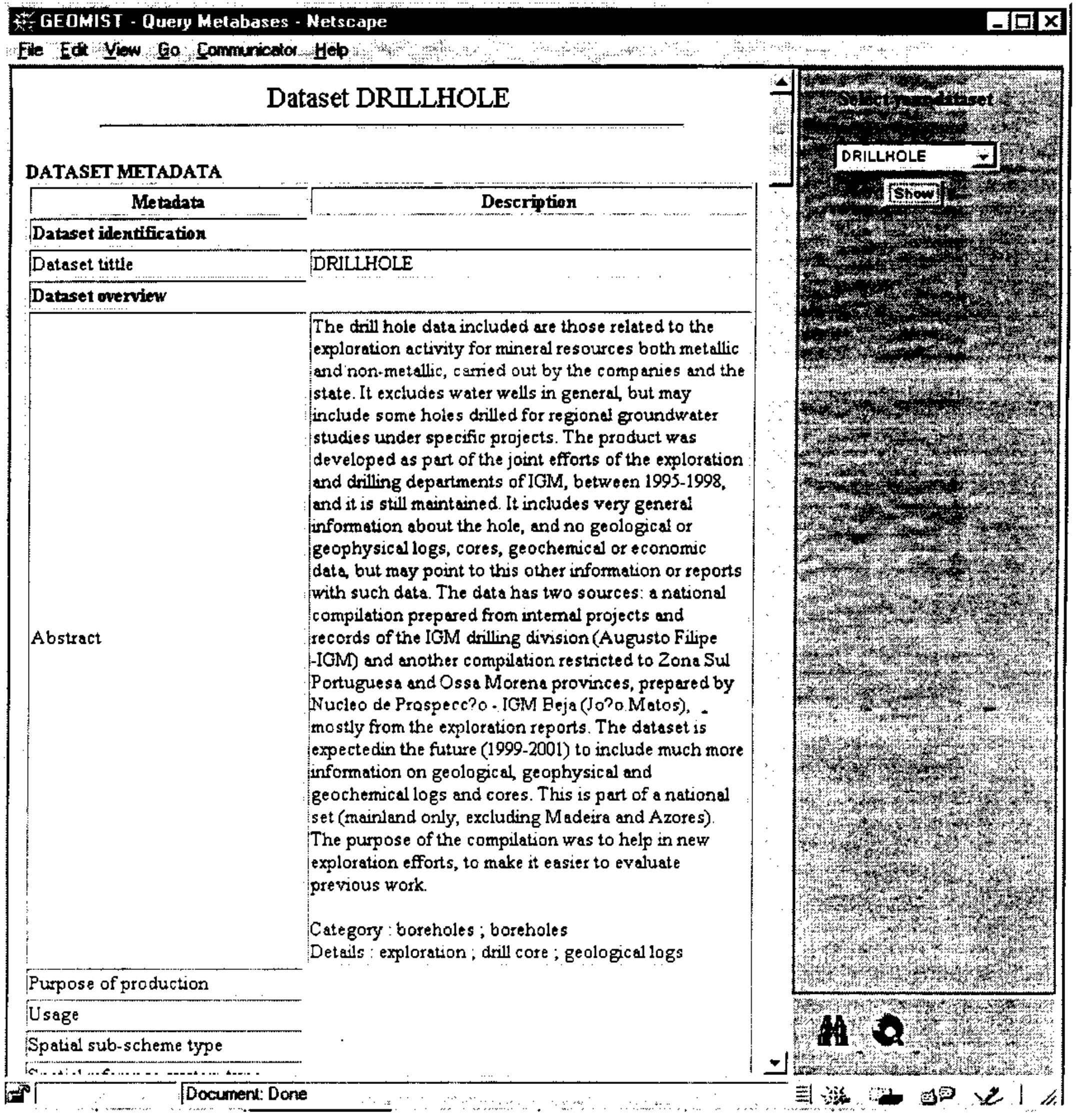
Select t.ts#, t.name from sys.ts$ t where t.ts#>3

The condition *t.ts#>0* is used to avoid to retrieve system tablespaces (0–*system*; 1-*user_data*; 2-*rollback_data*; 3-*temporary_data*).
After the user chooses one dataset the request is sent to the system to display the metadata. As we described above, there is a specific table with meta-information about datasets. The other metadata is obtained dynamically from system tables. There is a simple scheme about these tables.

Table	Description
dataset_metadata	User table with metadata about user dataset
sys.ts$	System table with information about tablespaces. (ts#, name, owner#, online#, ...)
sys.obj$	System table with information about general objects. Includes tablespaces, tables, associations, and attributes (obj#, name, owner#, ...)
sys.tab$	System table with data about tables structures
sys.col$	System table storing data about table attributes (obj#, col#, name, type#, fixedstorage, precision, scale, ...)
sys.ccol$	System table with data about which attributes are present in constrains (com#, obj#, col#, pos#, spare1)
sys.cdef$	System table with data about table associations (con#, obj#, cols, type, robj#, rcon#, rrules, ...)
sys.con$	System table with data about all constraints (owner#, name, com#, spare1)
sys.com$	System table with comments (description) about objects(obj#, col#, comments$)

To get the information we need to fill the metadata form, the system starts to search for the dataset (tablespace) identifier. With this code the process queries *dataset_metadata* to obtain the static metadata associated to this dataset (e.g., ts#, name, spat_ref_type, language, ...). Additionally the system get the categories, subject, subject detail and keywords, which characterise the dataset. The next step is to get data about Object Metadata (tables). To the advantage of this query the expression is relate multiple tables and retrieves exactly the columns needed.

For each table in this dataset, the process queries the database server for their columns, including data types, width storage, precision, and constraints imposed by the data model.


GEOMIST - Query Metabases - Netscape
File Edit View Go Communicator Help
DRILLHOLE
Show
Document: Done

Dataset DRILLHOLE

DATASET METADATA

Metadata	**Description**
Dataset identification	
Dataset tittle	DRILLHOLE
Dataset overview	
Abstract	The drill hole data included are those related to the exploration activity for mineral resources both metallic and non-metallic, carried out by the companies and the state. It excludes water wells in general, but may include some holes drilled for regional groundwater studies under specific projects. The product was developed as part of the joint efforts of the exploration and drilling departments of IGM, between 1995-1998, and it is still maintained. It includes very general information about the hole, and no geological or geophysical logs, cores, geochemical or economic data, but may point to this other information or reports with such data. The data has two sources: a national compilation prepared from internal projects and records of the IGM drilling division (Augusto Filipe -IGM) and another compilation restricted to Zona Sul Portuguesa and Ossa Morena provinces, prepared by Nucleo de Prospecc?o - IGM Beja (Jo?o Matos), mostly from the exploration reports. The dataset is expectedin the future (1999-2001) to include much more information on geological, geophysical and geochemical logs and cores. This is part of a national set (mainland only, excluding Madeira and Azores). The purpose of the compilation was to help in new exploration efforts, to make it easier to evaluate previous work. Category : boreholes ; boreholes Details : exploration ; drill core ; geological logs
Purpose of production	
Usage	
Spatial sub-scheme type	

The source codes developped to interact with the Metabase are given in Annex 4.

DATABASE QUERY

The problematic of the distribution of geographic data through the Web, raises three distinct, although interrelated, questions, requiring further analysis: (i) firstly, there is a process to formulate the access request, in which the user selects a specific dataset. The responsibility of this process is assigned to the security system which checks for the identification of the user and creates a new session, classified according to the user access level; the server generates then a specific interface access, according to the dataset required and session level; (ii) in the following step the server receives the request message composed in the client side by an attendant process, interprets the contents and creates a standard expression to access the DBMS; (iii) finally the results are packed into a message and returned back to the client, followed by processes to unpack and display results in standard form (e.g. Hipertext Markup Language – HTML). There are unambiguous syntactic schemes to support inter process messages.

DATABASE QUERY IMPLEMENTATION

This service is implemented at two distinct levels of interrelated processes. When the user issues a request to access data bases, a set of applications are invoked to provide proper interfacing and, later on, to accept messages requested with user defined criteria. These applications includes the *alphaAgent* - which verifies the validity of the user session - and the process to interpret the message and to create a new request message, including access restrictions associated with the session level. Afterwards such new message is sent to second level where another application connects to the Database Management System (DBMS) via Oracle's Application Server.

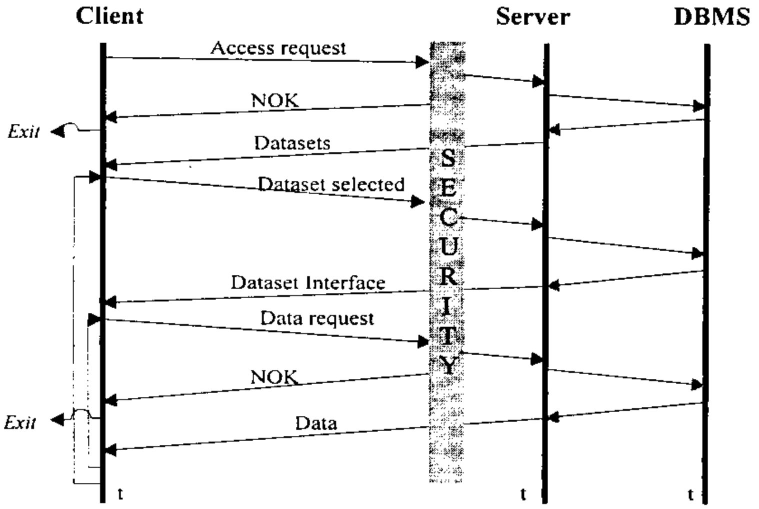


Figure 1 – Data flow in database access

The dataset interface, provided to the user, consists on a list of components built according to the privileges of the user session and the metadata stored in *datasetInterface* table whose structure is presented below.

datasetInterface = { datasetID, objectName, objectOrder, objectType, defaultValue,
viewName, attributeName, accessLevel, action }

To provide client side functionality the page, containing the dataset interface, leads to a number of JavaScript programs. For example, on querying database action, there is a function to pack the component's contents and send the resulting expression to the server. When the query results are returned back to the client, some attributes are shown in a ordered list at the main frame of the page. Each list item contains an hyperlink to an independent process, also written in JavaScript, which displays in detail (inside another frame) the selected instance. To prevent access overload to the database, the entire stack of query results is kept in cache.

All dataset objects include metadata about the access level for each instance and attribute. Every database instance also includes the owner of data, used to avoid undesired accesses to concurrent entities. The user belonging to the ALL group has superuser privileges, i.e., can access every occurrence classified at the same level, as well as all lower levels. This way each database query is restricted by these constraints. The algorithm below aims at performing the database query according to a previously built *queryExpression* and the present user session.

```
(IN: datasetID, viewName, accessLevel, userGroup, queryExpression ; OUT: result){
result =
DBMS.Select
            (Select name from sys.col# where access <= n and sys.obj# = datasetID)
      from viewName
      where
            queryExpression
            and    ( access < accessLevel
                   or (access = accessLevel and group = userGroup)
                   or (access = accessLevel and group = ALL))
}
```

The programming language chosen to write these applications is Perl.

The source codes programmed for data base query are given in Annex 5.

INTERACTIVE MAPPING

This service implements a set of functionalities such as the broadcasting of maps drawn on the fly according to the certain parameters, such as scale, map extent, themes to be include and legend. The server draws the map as an image, and the browser displays that image. Whenever the user wishes to change the view (by panning, zooming in or out, or turning themes visible or not), a new request is sent to the server with slightly different parameters than the last view. The server immediately generates a new map with the new parameters, which is returned to the user, restarting the cycle.

The service is made of a set of distinct processes interacting at two distinct depth levels, as in the other services. Beside these programs, implemented to grant connectivity, performance and security, there are additional commercial software: the GIS engines. The ARC/INFO software is used to collect, structure, edit, analyse and export GIS datasets. On the other hand, ArcView is used to perform GIS data display, and mainly, to enable the communication with the web server through the Internet Map Server extension. The web server hosts the ESRIMap extension to enable appropriate connection with ArcView. These have been implemented using processes in Java 1.2, Avenue, AML and C.

At start up, the client browser imports an applet Java which contacts with the remote server to get the initial data, that is, the base map and the list of themes characterised by name, data type, pre-defined legends and attributes to use in queries. As well as the other maps or data requests, this one is filtered by the security system in accordance with the user session. This is done invoking a system application, called from an Avenue script, which connects to the *sessionAgent.*

Each map request message contains information about the map extent, themes to include and, for each theme, the chosen legend. There is an ArcView script which receives this data, redraws the new map dynamically and converts the map to an image which is returned to the user. On other operations, such as *identifyFeature*, the applet verifies the theme selected and the current location of the mouse. With these parameters at hand, the applet invokes the Avenue script, in the ArcView environment, to get intersected features and, with this list, querying the DBMS to return their description attributes.

Moreover the system provides ways to find features with particular attributes and to perform simple analysis operations, such as find features based on their proximity to other features or associated polygon features.

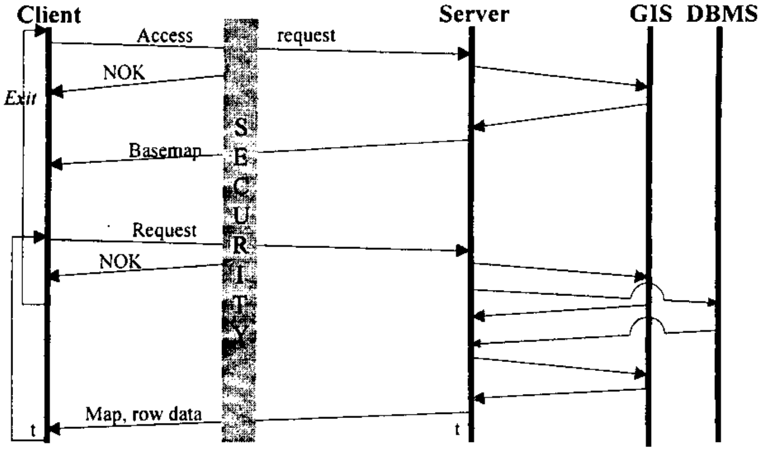


Fig. 1 – Data flow at interactive map service

The basic architecture is similar to the Client/Server model where the client is typically a Web browser and, at the server side there is a multi-tired server.

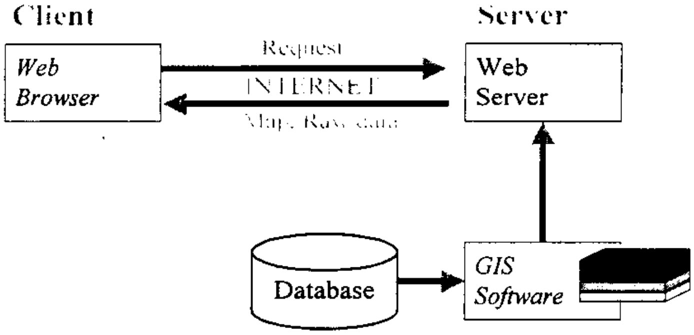


Fig. 2 – Global interactive mapping architecture

INTERACTIVE MAPPING IMPLEMENTATION

The implementation of this service is based in the implementation of various co-operating services, providing a global functionality. In one side there is security service, checking and recording each request and, on the other side, the interaction between the Web server, GIS server and the database server, integrating distributed processes.

The overall architecture of these services is shown in next figure which also explains about the heterogeneity of the global system.

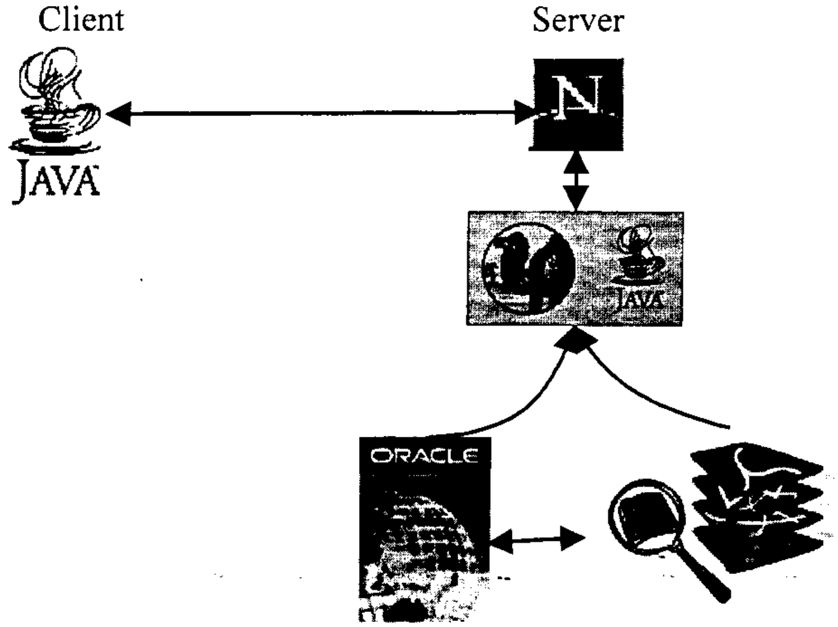


Fig. 3 – System service architecture

The source codes for Interactive Mapping Services are given in Annex 6.

SYSTEM SECURITY

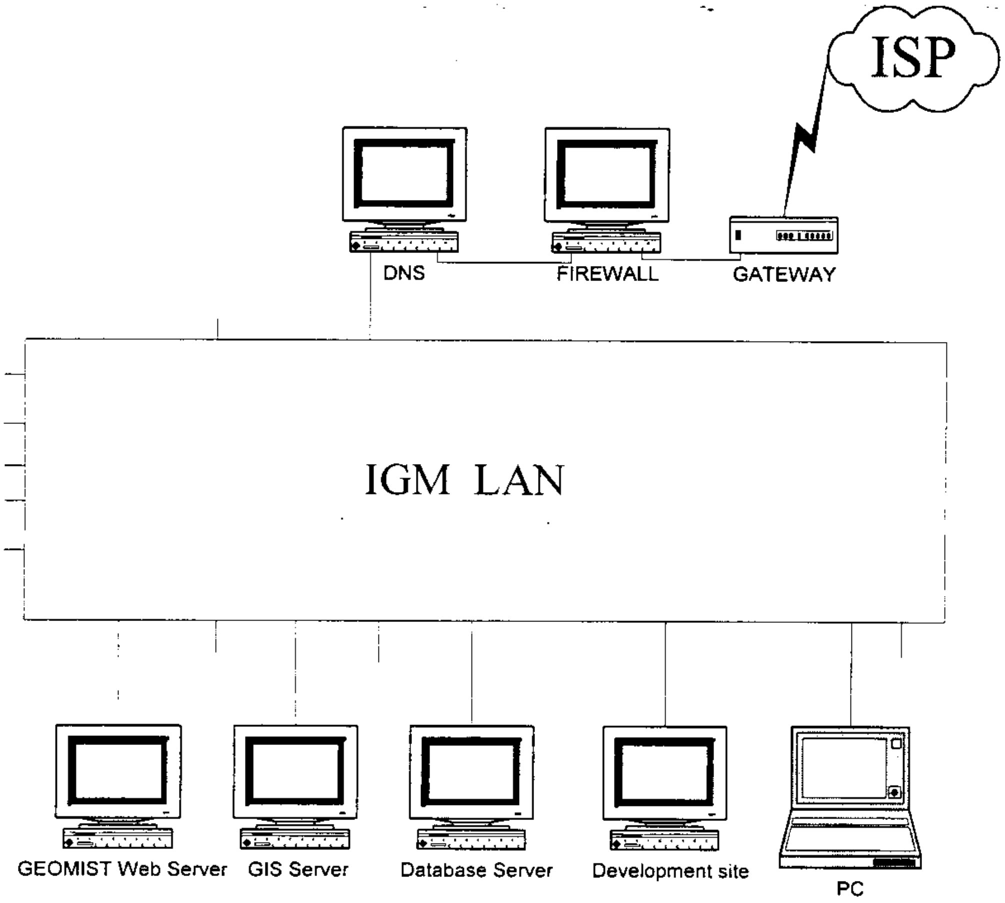
The security of the Geomist system must be evaluated in two distinct ways. Firstly, the security system protecting the access to the organisation, hindering the access other than through the Hypertext Protocol. On the other hand the access to the applications, data and general resources available at Geomist level.

We will approach both levels but in detail the security barriers stabilised to avoid undesired accesses to the Geomist applications and datasets. System is explained from the point of view of the Portuguese server but, similar considerations apply to the Spanish server. However as noted in the appropriate section describing the Spanish server, this is based on Windows NT

1. GENERAL SECURITY ARCHITECTURE

The general security architecture is similar to the general topology of Internet connection security from the organisation to the Internet Service Provider. The scheme is shown below.

Figure 1 – General security architecture



2. GEOMIST SECURITY

The system grants access to the Geomist reserved services, including Metabase Search, Databases query and Interactive mapping, through a username/password pair. Upon authentication, a new session is created which is characterised by a username, IP address of remote host, access level, user group and a timestamp. Each data request is checked and registered by the security system verifying the user session and updating the timestamp. If the time elapsed between two requests becomes higher than a predefined timeout, an exception is raised and the session is closed. The interface presented and the returned results are also in accordance with the user profile.

On his/her first connection, the user is requested to fill in a form with his/her identification, passwords and additional data used to perform some statistical computations and to contact the user. Afterwards the connection is established with the username/password pair.

If the user identification matches the server knowledge about this user, there is an agent to create a new session to him. From there, each request is associated to this session to check and update the session log.

SYSTEM SECURITY IMPLEMENTATION

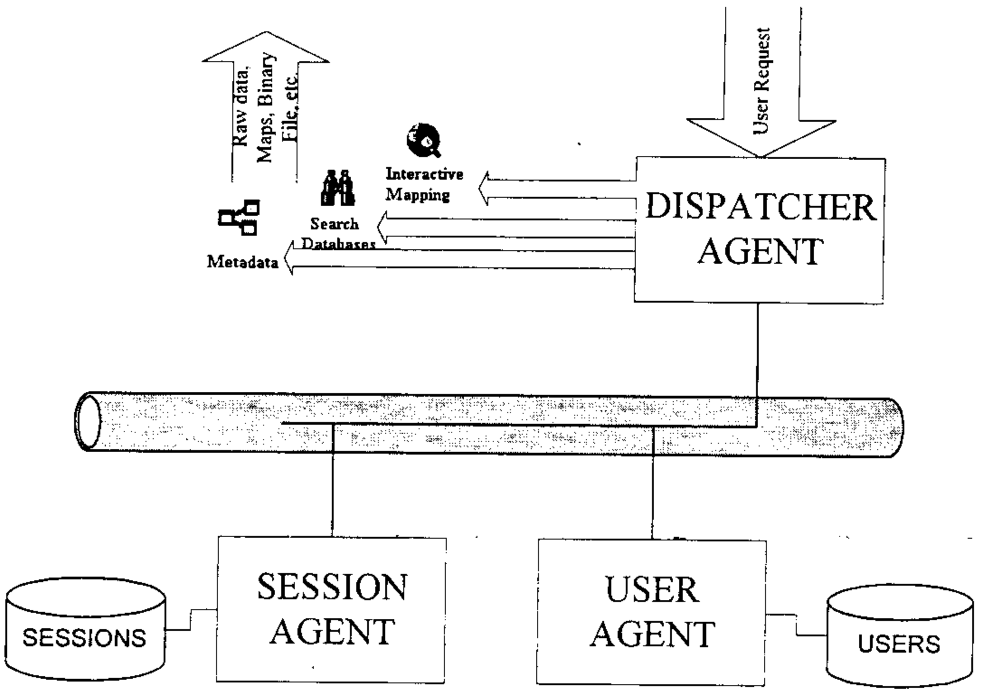
The security system consists of a set of conceptual and software agents whose interoperability provides secure access to the data and applications. The programming interface languages chosen to implement the security agents are C, Java 1.2 (Sun Microsystems, 1998), Practical Extraction and Report Language - Perl 5 (Larry Wall, 1996).

The session agent (*sessionAgent*) is an independent deamon programmed in Java and aims at storing the data about each user session and to accept connections from other agents to register the user session and therefore to check the validity of the user requests. The *sessionAgent* remains listening for connections through a socket port.

The identification agent (*authAgent*) is a conceptual agent programmed in Perl and provides a simple interface to the user registering, authentication and identification updating. When the user proceeds with authentication the *authAgent* connects to the *sessionAgent* to register a new session.

Afterwards the connection is established by the specific service, the security is translated to the access agent dedicated to each service. The data access agent (*alphaAgent*) also written in Perl, aims at checking the validity of the user session. reporting the *authAgent* upon error. On the other hand, the interactive maps agent (*mapsAgent*), written in C, has the same specifications.

Figure 2 – Geomist security architecture



The source codes to implement the GEOMIST Security System are given in Annex 7.

WP 7. DISSEMINATION ACTIONS

GEOMIST dissemination has evolved in Spain and Portugal through presentations of the project to different interested sectors, mainly environment and land planning as well as continuous animation of the user groups and involvement of the Regional Bodies in charge of mining rights regulation.

In conjunction with the final review meeting to be held in Lisboa on 22 september, a workshop on "Sistemas de Informaçao Geográfica e Geológica de Base Regional" will be organized on 23 september in Beja, in the hearth of the Portuguese part of the Iberia Pyrite Belt, where the GEOMIST services will be demonstrated to potentially interested users of differente sectors: Land Planning, Natural Hazards, Geological & Hydrogeological Resources, Tourism, ...

In Spain, an agreement to manage the shared data bases between Junta de Andalucía and ITGE has been reached and the principles for the general management and administration of the system, information access, data pricing policy, user categories, are now stablished.

The continuation of GEOMIST beyond the end of the Commission funding will be pursued through a cofunded project between Junta de Andalucía and ITGE.

Concerning the Exploitation Plan, MTDV have decided to be no longer involved as an active GEOMIST partner beyond the project completion, therefore the future exploitation of GEOMIST as a transborder information system, besides national developments in Spain and Portugal, has to be worked out. Some preliminary ideas to face this situation are presented here.

Future Plans

We may define Geomist in a number of ways, but perhaps the shortest definition is *a web based geological and mining information system for the IPB*. That presupposes it must have an administration body, hardware and software, a budget, a maintenance and development team and appropriate channels and agreed rules to receive and deliver information. Those are easier and more logical to implemented at national level. But some kind of trans-border structure to co-ordinate actions, to normalise procedures as well as to support its expansion is crucial, otherwise it will expand within Spain and Portugal into divergent and incompatible ways and will not reach an European dimension.

GEOMIST is not at present a "structure" in the sense that it has no staff, no budget and no approved working plan. A number of steps are proposed to maintain and expand operations beyond the end of September. The most important steps are:

- Commitment of both ITGE and IGM management's as host institutions in order to ensure maintenance of the hardware and software licenses and a minimum of manpower
- Establishing an adequate relationship with the administration agents (Junta de Andaluzia, Dep. of Geological Resources-IGM and other), who are the legal interface between the state and the mining companies

- Creation of a joint administration, with managerial support at IGM and ITGE, working on the basis of an approved exploitation plan.

An exploitation plan should be based on a few principles, which we discuss in turn:

1. A legal status for Geomist
2. Existence of an approved working plan
3. Existence of an "administration board"
4. Existence of a budget, identifying the sources of money
5. Clear data and price policy
6. Medium term data loading policy
7. Strong regional links
8. A working team (developers, maintenance, technology providers, etc.)
9. A research plan
10. Co-operation agreements with third parties
11. Involvement in geo-data normalisation issues at European level
12. A marketing plan

A legal status for Geomist

This would demand the assistance of lawyers. But some basic information must be collected: on IPR, legal principles on data ownership, a description of the components, on the preference for a formula, like consortia for instance.

Its components are:

a) hardware and commercial software
b) internally developed software
c) administrators
d) users
e) associated partners
f) developers
g) data (from various sources and independently of ownership status)
h) metadata
i) services

The legal framework must explicitly consider all those components. The roles of ITGE and IGM as hosts and administrators seems a natural one (with enlargement to other 5 Geological Surveys (GS) in case of approval of an European programme to be submitted in December). The case of ownership of data, software, etc. is a delicate one, which turns even more complex in each country because there are other players (regional governments, institutes, etc.). We might only consider ownership of the physical means (hardware, software, media), but even that may be part of the installed infrastructure and just hired to Geomist, and obligations on maintaining the general infrastructure by the hosts (LANs, staff, Internet connections, etc.). The relationship with information centres within each GS is not considered in the legal status and is an internal matter each GS (Geomist maybe seen as a separate unit within those centres, as the centre itself, as a distinct unit from those centres, etc.). However the problems of global data policy, which might involve existent agreements with other parties and the

PSI problems addressed by the Commission's Green Paper on information is obviously underlying that and must be re-examined by each GS. Explicitly the problem of E-Commerce and association with other parties, for marketing, etc. must be considered. (This could be the laboratory for GIVES)

We consider associated partners, the data providers. The users are those that acquire data and services. Mining companies are not data providers from a legal point of view, because they do not provide data directly to Geomist but to the administrations; they are users in the sense that they acquire data and services from Geomist. The data providers are mostly the administration and public institutes. Including them as associated partners would help in the funding, the capability to generate important links, to create a forum to overcome ownership problems.

Some associated partners are simultaneously users (like the GS and part of the administration). Associated partners and permanent users would be funding agents of Geomist. In return they would benefit of discounts in the acquisition of services and data.

Geomist would have site administrators and advisers on the different subjects (part of the coverages, normalisation, e-commerce). The administration board would be appointed by the GS and Associated partners. The work of the experts would be contracted. Permanent staff would be kept to a minimum at each site (two people: one for interfacing with the users, data loading and site maintenance, another as part of the development team). This is opposed to creating a central server, but instead to consider a European distributed system (moving from 2 to 5 and then 15 countries), which would be deeply interconnected with the outside world from the on-set.

Working Plan

Draft proposal based on this chapter presented to Directors of ITGE and IGM (after all contributions, including reviewer's criticisms). Informal contacts at the Beja workshop, followed by draft proposal written with expert legal advice sent to potential permanent users and associated partners. A tentative future research and workplan would be circulated at the same time. That would take about 4 months, and the Geomist service for the IPB would then come to existence. Meanwhile the server would remain alive and all conditions for the supply data and services should be created. New projects would then be submitted to the appropriate programmes in accordance with the tentative workplan. Some deadlines should be taken into account like December for the V Framework.

Administration board

Would be responsible for implementing the workplan. Each member responds individually to its GS or organisation. Annual report of activities to management and a public one. See also notes on the General Principles.

Budget

The budget will rely on contributions from the host institutions and associated partners, permanent user fees and revenues from services and data sales. The data will not be "owned" by Geomist so a 10%-20% would collected from data providers on the data supplied to users. On the other hand permanent users would have a 20% discount on services and data. Costs are variable from site to site but a cost of 175000 Euro should be typical. Some 25% would be supported by the host institutions, another 20% by the associated partners, some 10% by the users. The remaining 45% would come from services, data sales and co-sharing of approved projects submitted at regional, national and community level. In fact this last mechanism would be the basis for the development of new services and research.

It is not expected that data sales by themselves would reach significant values, because most of it will be public information or information for the development. We should then weight the information available against the cost of it not being available and give credit to the sources and the added value provided by Geomist. Large regional projects or programmes should then contribute to this service.

Clear data and price policy

This is a hot topic, of which the Green paper on Public Sector Information is an echo and a reference. Several debates organised by the commission followed on as Green papers on the member states reflecting the national problems. We will not be able to adopt the right policy, but we cannot wait and see, because we will be unable to provide to users data and services whose price (or whether they are free) we cannot agree upon. This a topic to be addressed at the Beja workshop. It was requested to IGeoE, CNIG, IPCC and IGM to present views on that. In an appendix examples of data prices from ITGE and IGM catalogues are given. In what follows some other ideas are presented.

In Portugal there is the Green Book for the Information Society in Portugal. There are defined :

a) information for the citizen (it should be universally available for free)
b) information for the development (it should be available at nominal costs to the social and economic agents)
c) added value information (it should be available through the market, like any other commodity)

we add,

d) information provided by the economic agents to the State for statistical and technical purposes and which are confidential (it is used by the Commission as statistical aggregates for planning or by the relevant State departments to be converted to a), b) or c) as appropriate)

ITGE and IGM must clarify in concrete what that means regarding Geomist holdings. We think metadata and some cleared identified files should be considered of the a) type. However for the remainder price tags should be given. Although a draft document refers the possible types of services in a previous progress report no further attempt was made to define what Geomist will be capable of providing as services. This is dependent on the installed capabilities at ITGE and IGM (and/or other partners).

Medium term data loading policy

If Geomist is frozen at its present level of data it will soon die. However geological compilation and data gathering programmes are expensive and demand specialised human resources which are scarce and clearly this should not be the role of Geomist. But mechanisms to improve the quality and quantity of coverages must be enforced. The role would then be to identify the needs, publicise and get associated with the initiatives of other parties, providing some expertise, standards, know-how and the host services.

This would be included in the workplan. Of course the level of association with the various initiatives will vary according to priorities. In some cases no direct involvement would exist, just a posting.

Strong Regional links

Through some of the associated partners the regional links will be established. The existent regional programmes (and central government investments with regional impact), their lack of data and services, possible funding available will be identified and concrete actions included in the workplan.

Working team

Staff would be based on 2 people per site. The roles would be site administration and maintenance (security, including co-ordination of data loading and interfaces, metadata, security), routing of user requests, research and development, training. The system is quite complex involving GIS (Arc/Info, ArcView, Avenue), DBMS (Oracle), Access, Web (Netscape, Java, TCP/IP, IMS), Operating systems (Unix, NT), SQL, Cas.Cade. It is difficult to have someone proficient in so many aspects at each site. Additionally the support of those products is most of the time insufficient to handle the problems. The best would be to have the pool of expertise spread through the different Geomist sites with a somewhat unified procedure and implementation. The local technical administrator would specialise on an area or two and be able to help other sites either remotely or on site. This would involve internal costs to be beard by the 'client sites'.

There is a tendency and legitimacy for each Department in the GS to develop their databases and the same happens at the future associated partners. This is good and bad news. Good because the experts know better what to do, but bad because they most often avoid compatibility with other disciplines making it costly the future integration, harmonisation of the data and even the production of metadata. Metadata is often prepared later in an abstract way by non-experts giving rise to lower quality products and data loss. The mere preparation of catalogues is not enough. So someone in Geomist must be concerned with the data loading. By data loading we mean remotely populating Geomist databases, physically transferring datasets to Geomist servers and making it available, establishing first-time connections to other servers which are associated with Geomist and which will provide new datasets. We will have three situations: data residing in Geomist servers, data residing in other servers which are associated with a Geomist site and data residing in non-digital archives. The adoption of common data models in the key datasets and common sets of tools is very important (thesaurus, data dictionaries, common numbering of permits across organisations, etc.)

Research plan

The technician in charge of development and research would be in charge of implementing the research plan and participating or acting as a link to EC research projects with impact on Geomist. Both the maintenance and research would involve contracting services and forging partnerships on specific topics (like POSC, 3D, text and data mining, E-commerce).

Co-operation agreements

Some strategic alliances must be forged. By inclusion into Geomist itself of the associated partners some ground is covered. But technology providers are essential. We consider ESRI (Arc/Info, ArcView, IMS, 3D analyst), Matra (Cas.Cade), Oracle (DBMS, OAS) as the most relevant ones. Publicity on the Geomist site ("Powered by ..") and getting certified site categories are possibilities. IGM made a bilateral agreement in the framework of GeoFrance3D on the use of the Cas.Cade based Geological Editor. Other technology providers could be Ark Geophysics (UK) which made an agreement with BGS to extend POSC to gravity and magnetics and announced it already has a product. New University of Lisbon, which has links with INRIA (France) is working on text mining with IGM to be incorporated on IGM Geomist's site.

Involvement in geo-data normalisation issues at European level

Normalisation and harmonisation issues are very important topics. This is a daunting task, so the tendency will be to restrict the normalisation and harmonisation to metadata. This tendency in the medium term will be an handicap for the construction of Europe. So there must be resources and efforts to counteract this. Relationships with GIS metadata normalisation bodies should be sought. In the future this will be useful for other European non-EC countries (as naturally the EC will be growing to include more countries) and in other regional blocks on other continents. A stronger involvement with ISO bodies is essential and the following contacts are suggested:

- Ark Geophysics for POSC
- European Geological Society for Geological standards (stratigraphy, geological time table, geological formations and units, geological symbols and codes)
- European Geophysical Societ and Seociety of Exploration Geophysicists for Geophysical standards (Grid and image formats, excahnge data formats)

in addition to OpenGis, W3C, SGML.

Disseminate and use the existent standards, encourage users and data providers to adopt them, create compatible products, identify areas which should benefit from their existence.

Marketing plan

Prior to any marketing plan all conditions to supply data must exist (legal aspects, valid datasets, e-commerce implemented, price list and data policy, contracts, forms, etc.).

This would demand at least 4 months. This task should be contracted out. Representatives of the potential user community will be in Beja and to pave the road an inquiry on the services which they value most, possible prices they find acceptable will be circulated, as well reactions to what is available and immediate plans. The follow up of this activity (data gathering, contacts to ensure minimum number of replies, email to potential customers, etc.) should be part of the plan preparation by the contractor.

This would be one of the first tasks of the administration board.

ANNEX 1

PLAN FOR COMPLETION AND UP DATING

GEOMIST DATA SETS

1. Inventory and data modelling

In the first stages of the project it was carried out an inventory of the most relevant datasets, a survey of the plans on both institutions towards acquiring and homogenising such datasets, an identification of common datasets and of missing relevant ones from an anticipated user perspective.

That made clear the dependencies on other projects, namely in the case of Portugal GEOAMB, which was expected to be concluded Dec 1999, but will be continued for another 8 months into 2000. We should also bear in mind that SIGMA is alive in Junta de Andaluzia and is expected to proceed.

Now, at the closing report of GEOMIST it is important to review the situation. Important advances were made towards improving the quantity and quality of data in each coverage as well acquiring new data. That means the situation is better in some respects, but also that some of the projections made in 1997, were too optimistic, because of the delays in the start of most of the actions in GEOAMB due to administrative and financial constraints. However despite of that a large bulk of information is already available.

The simple enumeration of the coverages reveals a very ambitious goal for Geomist as a service. This does not mean it should be reached in the time frame of the pilot project under consideration. We are convinced that a diversified and comprehensive number of coverages is very important for the sustainability of Geomist. Otherwise it will die once that project is closed in September. In the exploitation plan this point is re-examined.

Before describing the coverages we should mention that the "layers of geo-information" of Geomist are organised in such a way that only the Arc aspect of Arc/Info is used, with the alphanumeric attributes of the data included in Oracle tables.

We now consider all the coverages, independently of being available on both Geological Surveys. In fact the tendency is that sooner or later they will be available on both, so the inventory should be as complete as possible and that will lay the foundation for future work. We should consider that Geomist is to be pursued as a service, the exploration is an on-going activity and several geoscience related projects are taking place or are being proposed for the region.

Detailed descriptions of the data models were presented in previous progress reports and are not reproduced here. For the new coverages introduced here no data models are provided, and this a topic for future work.

1.1 List of Coverages

Those are in fact the subject categories, using the GEIXS terminology, which are considered relevant and feasible in the case of Geomist. Because our goal is to reach the data level (remember the data triangle concept) we were forced to tackle harmonisation issues and develop data models in a number of cases. But the subject is a vast and specialised one. It must be continued and deepened. There are different levels of harmonisation in the datasets and some of them are not available in both countries, but this does not diminish the interest of its inclusion. In fact using the server as a focal point some of those handicaps may be quicker overcome.

The idea is to publicise the list below within the server itself and encourage the users to complete it and co-operate or generate projects to improve the coverages.

Geology
Geological maps
Mineral Occurrences
Non-metallic occurrences and quarries
Exploration drill-holes
Gravimetric data
Aeromagnetic and radiometric data
Petrophysical data
Geochemical data
Remote sensing data
Topographic Base
Digital Elevation Model
Bibliography
Geophysical maps
Geophysical surveys
Gravimetric Base network
Spatial mapping indexes
Lineament maps
Exploration permits
Hydrogeological data
Geological Formations
Heat flow data
Thermal springs
Paleomagnetic data
Deep seismic refraction profiles
Deep magneto-telluric profiles
3D models

Geology

An assembly at 1:100.000 of the Spanish side of the IPB and an assembly at 1:200.000 of the Portuguese side was merged. A common very simplified geological legend was adopted, but the reader may revert to the original paper legends (in digital form) of both maps which are preserved in the coverages.

This digital product (Arc/Info coverage) did not exist when Geomist was launched. This very recent product may be used as a basis for a Geological Map of the IPB, considering the advances in the programmes at 1:50.000 in both countries.

The product does not fully implement the agreed data model, but it is close to it. A full implementation should be part of an IPB mapping initiative, triggered by Geomist (to be described later). The data model is part of a document submitted to the geology sector, which once revised and approved should provide guidelines for future work by mining companies, concerning geological cartography. There is now complete coverage of the Geomist polygon. Despite the different ways digital cartography is dealt with by the Geological Surveys, we think the experience may be generalised, to other trans-border European regions.

The involvement of the cartographic departments of ITGE and IGM is suggested as a way to get the message into the corresponding EuroGeosurveys topic network, so as to disseminate Geomist results in this regard and identify (from the EuroGI transborder European regions list) where that might be of immediate interest.

Visualisation issues

Data is available in both raster and vector forms. It is useful to visualise the data as coloured polygons where point data is overlaid (such as drill-holes, mineral occurrences), or as transparent polygons overlaid on other coloured raster data such as an aeromagnetic map, where mapped geology may be identified and correlated with mapped magnetic units.

Geological Maps

Geomist also offers links to scanned geological maps at larger scales (1:25.000 and 1:50.000), because those products already exist. This is illustrated in the Geomist server in a few cases only, but will be implemented in a more general way. This opens the possibility of combining non-harmonised digital data (which may offer more detail) with the other datasets in a similar (but less versatile) way as for the prepared datasets.

An important aspect for land management is the use of geology and special purpose non-metallic evaluation resource maps at even larger scales (1:5.000, 1:10.000). Those are not included for the moment, but actions will be taken to get information in this regard and include appropriate links in Geomist.

Visualisation issues

These products will be explored as geo-referenced rasters or vector CAD products.

Mineral occurrences

In Spain mining data from 353 mineral occurrences and deposits are classified by size, morphology and ore type. In the case of Portugal some 144 mineral occurrences were identified in the IPB, but they are only partly classified according to the above criteria. A sounder and more comprehensive data model was created as part of the Geomist efforts. In Portugal GEOAMB took in charge the complete digitisation from 1:50.000 maps of occurrences on a countrywide basis (about 4000 mineral and 3000 non-metallic ones) and synthetic associated paper files. Using this as a starting point and applying the Geomist data model, some 600 metallic occurrences have been classified in Portugal by the IGM Dep Mineral Assessment, but unfortunately they fall outside the IPB.

Work is still in progress within GEOAMB to conclude the work on the Portuguese side of the IPB. But it should be noted that this complex issue demands a specific project, with the participation of experts, in order to be adequately addressed. However Geomist is aware that important initiatives exist at all levels (national and European) and that the sharing of information and co-ordination of efforts should be pursued. In fact there is an important project called GEODE which is funded by the European Science Foundation and includes the study of ore deposits in a number of geological provinces including the IPB. As part of its activities GEODE has asked the MIRO to create an European database of world class mineral deposits. As part of the dissemination efforts Geomist was presented at the Geode workshop in Lisbon. Fernando Tornos (ITGE) and Fernando Barriga (UL) are members of the IPB group. IGM has joined GeoFrance3D and the counterparts are Patrick Ledru (also in charge of variscan deposits in Geode) and Luis Torres (Geomist team) . Application of the geological editor will be on the Aljustrel mining area and the Portuguese IPB in general. This initiative was triggered by Geomist. There is confidence to believe that within the next year much more information will be fed into Geomist as a result of those parallel efforts.

Visualisation issues

Data is point type data and is best explored as such. Some occurrences are in fact associated with a mining camp, of which they are mere details (such as orebodies in a mine). Association tools (not available on-line) and a coverage that has the centroids of such "clusters" is also provided, and it is sometimes a more suitable way of exploring the data.

Non-Metallic occurrences

As part of the workshop in preparation a number of initiatives involving GIS based non-metallic resources compilations were identified in the IPB in Portugal. The point here is a definition of a framework of co-operation with those other initiatives on behalf of Geomist. Links to those other works will be added after the Beja workshop.

This is, by its nature the sector with larger impact on land planning. A number of projects on both sides of the border do exist.

Junta de Andaluzia also has information of this type which might be incorporated in the future and other sectors of IGM also play a role on this. This is closely linked to the cadastre and statistics of the extractive industry.

This a topic of economic geology that demands some knowledge of the aggregates, ornamental and dimension stone industry. It is not the role of Geomist to collect and compile such information, but it may help by providing adequate data models and normalisation guidelines. This might also be an action triggered by Geomist in the Beja workshop.

The philosophy here is that Geomist should be considered open and that it may act as a host, leading the action for data contributors to use the server as a vehicle for dissemination of their own information. Of course the integration and harmonisation of such new data might be a challenge. The Geomist exploitation plan provides further details on this topic.

Visualisation issues

This involves two types of coverages: point data (with the location of occurrences and quarries) and polygons (with the limits of the exploration areas), Both are readily handled by the interactive mapping module. However some data is CAD (details of exploration areas) which is easy to handle through the interactive mapping module (envolves a common geographical reference system for all files-UTM). Other data is raster (pictures of explorations). Not supported, but easy to implement (event file).

Exploration drill-holes

Exploration drill-holes number 300 in Spain and 1000 in Portugal. However SIGMA , has a much larger collection of bore-holes, which may be incorporated in the future. Also a large number of fields is empty in both databases. However as part of the information already compiled within Geomist, and by the use of the Geomist data model all compilation efforts may easily be incorporated. This is a very valuable database. Its full potential may be unlocked when associated information which is at present not directly available in this coverage can be accessed, like documents from the technical archives describing them and *lithotheques* where the cores are stored. However the mining permits to which they are associated is part of the database and this allows the retrieval of exploration documents which are available for some bore-holes and will be available in a general way for the Portuguese IPB within a few months. This is a coverage whose data model, because of its implications to 3D deserves more attention in the future. A working data model (based on the Geosoft model, appendix) was used for the preparation of the 3D exercises, both using the Matra viewing tools and the BRGM geological editor. A number of packages exist to deal with drill-holes (Hole-in-one from Geosoft, Vulcan from Maptek, GDM from BRGM which is accepted directly by the Geological Editor, RMF from BEICIP, adopted by OMEGA). Geosoft, or GDM ASCII files are easily constructed using an editor and digital data.

In short, bore-holes cover simple descriptions, which we might consider 'extended metadata' and detailed descriptions of the geometry, lithologies and physical properties. Our model covers the former and we suggest for the latter the use of a suite of Oracle tables allowing a mapping of both the GDM and Geosoft formats, with some extensions to cover geophysical logs (this was not implemented, but a scheme is provided to do it)

Visualisation issues

Data may be visualised as point data and explored as such within the SIG. A coverage of drilling campaigns provides the centroids of "clusters" of drill-holes made in mining areas, to unclutter the visualisation window.

The link between bore-holes and technical reports is a very powerful way of exploring the data (it allows to view scanned logs and textual descriptions, scanned cross-sections with interpretations of the drill-holes, etc.).

Data could also be explored as thick lines in 3D space and an illustrative VRML demo was made earlier in the project, but visualisation of geological and geophysical logs are not directly included in Geomist (only in the 3D demo).

Gravimetric data

As mentioned before this is an important dataset from the point of view of the mining companies and the mineral exploration community. A homogeneous dataset exists for Spain but is not yet available in Portugal. An agreed and satisfactory data model exists. About 30000 data points exist in Spain providing a fairly uniform regional coverage. Some 100000 gravity points (of the estimated 220000 stations) are in digital form in Portugal in an uneven and non-comprehensive coverage; of those only about 40000 are homogenised. More effort will be spent within GEOAMB, to complete this work, within the next few months. However no funding exists for the acquisition of more gravity data in the blank areas, for which only a very coarse gravity survey (0.04 p /Km2) is available from IPCC (coverage should be at least 1p/km2). As part of the mapping initiative to be proposed to ITGE and IGM management every effort to include the gravity map should be made (see bellow)

Visualisation

This type of data exists as a coverage of data points, as grid files and as geo-referenced colour coded raster files. It can be explored as point files (posting) which is of limited value or raster within the 2D GIS, which is much more useful. Because it is useful to visualise the gravity grids for different densities, a procedure to recalculate the grids for requested densities (available separately) and directly visualise the grids will be later included in Geomist. This last module is already available but it is too slow (use of Java procedures from GeoFrance3D).

Aeromagnetic and radiometric data

As explained before ITGE owns a new high resolution airborne dataset of the IPB. IGM merged several magnetic datasets in Portugal, and hired the plane that surveyed Spain to fill in an important gap in the IPB and to fly the border area so as to link Spanish and

Portuguese surveys. A homogeneous aeromagnetic dataset now exists across the IPB. This is not directly loaded in Geomist as they represent very large datasets. A number of products (line and point data, grid files, etc.) were prepared by the contractor that flew Spain and are available in CD-ROM, from ITGE and Junta de Andaluzia for the Spanish side. Within GEOAMB IGM will be preparing similar products and documentation about the mosaic of 5 airborne surveys used to create its dataset.

In the case of the spectrometric data the Spanish data relies on the above mentioned high resolution flight (which also acquired radiometrics), while the Portuguese IPB is covered by 4 airborne surveys, complemented with a coarse carborne spectrometric survey suitable for use at 1:200.000 scale. Again a number of products (line and point data, grid files, etc.) were prepared by the contractor that flew Spain and are available in CD-ROM, from ITGE and Junta de Andaluzia for the Spanish side. Within GEOAMB IGM will be preparing similar products and documentation about the mosaic of the airborne and carborne surveys used to create its dataset and has prepared the 1:200.000 exposure rate map. Work is also in progress in another project called the natural radioactivity map of Portugal nearing final national coverage in 1999.

A number of different maps are usually produced from the above mentioned information for the benefit of the users: Total Field magnetic map, reduced to the pole magnetic map, shaded relief reduced to the pole magnetic map, to mention the most relevant ones. In the case of radiometrics: Potassium, Uranium, Thorium, ternary maps and exposure rate maps are usually produced.

Only two of the 8 mentioned maps are presently available in Geomist. However it is neither difficult nor expensive to produce the remaining ones, which could be done in the framework of the suggested ITGE/IGM mapping initiative.

Data model is suitable and allows selection per survey, flight line, etc. But its Oracle implementation is inefficient and will be abandoned. Compressed ASCII versions of flat files or binary grid files (close to the form to be used) seems to be a better approach. This is under study in GEOAMB and would provide a much more efficient solution. This also the type of data format that will be requested to data providers. Once available the solution will be made available to Geomist partners.

Visualisation

This type of data exists as a coverage of data points, as grid files and as geo-referenced colour coded raster files (limited range for the moment, more will be generated). It might be explored as point files (posting) which is of limited value and not feasible considering the size of the datasets or raster within the 2D GIS, which is much more useful. Because it is difficult to anticipate all the combinations of grids the user might wish, a procedure to calculate the grids on-demand (available separately) and to directly visualise the grids will be later included in Geomist. This last module is already available but it is too slow (use of Java procedures from GeoFrance3D). Some solutions like incorporation of Surview features in the Geomist server to visualise flight lines and to view stacked profiles was looked at and seems promising, but it is too costly from the point of view of development. Surview is a product of the GSC, and maybe bought from their site for US $50. We suggest that the users may view the data off-line using this utility. If agreement is reached with the GSC we should allow the download of Surview in the Geomist web site.

Petrophysical data

In Spain a large Petrophysical database (with 1201 surface samples) exists for the IPB. It is not included in Geomist, but it exists as Arc/Info coverages and so it is very simple to include it, and in fact it would increase the value of the site. No similar data set exists for the Portuguese IPB, but a large number of measurements in drill-holes do exist (mostly densities and magnetic susceptibilities) and a work programme is under preparation, but it will not produce useful results before the next 2 years. So in this case no common data model is necessary now. Some work to ensure the use of consistent geological formations and units in the geological cartography and in this database on a transborder framework might be needed in the future.

Visualisation

This is point data, so it is readily handled.

Geochemical data

Lithogeochemical data from a regional survey carried out in Spain for the main geological units and mineral deposits is available. It was not included in Geomist but it was recommended by the evaluators (FS) its inclusion as ITGE is the reference institution for soils in Spain and is working in CLARINET. Because it exists as Arc/Info coverages it is very simple to include it, and in that would increase the value of the site.

In the meantime as result of the efforts of the metallic exploration department (IGM) a geochemical database was created (and it covers partly the IPB - in Access). It refers to soil data on heavy metals (Cu, Pb, Zn, etc.). It will be demonstrated at the Beja workshop. It could in the near future be included on Geomist (it would involve some discussion with the relevant department). Technically it would be easily displayed within Arc/View (a few hours work) and so made available in GEOMIST. A better integration (Oracle, review of the data model) would involve some more work. Taking in mind its importance for the topic of abandoned mines it would be worthwhile the effort.

Visualisation

Data should be viewed as point files (useful for the case of the Lithogeochemical data, but only for posting for the soil data, which is collected in a much denser grid), point symbols, contour maps and grid files. A number of maps could be prepared by the geochemists and displayed as geo-referenced rasters within Geomist (1:50.000).

Remote sensing

Landsat TM images are available in Geomist for the Spanish side. A new IRS-1C coverage was recently acquired within GEOAMB (panchromatic at 5m resolution and 23m resolution for the LISS-III multispectral data), and will be made available in Geomist. Low resolution geo-referenced files will be generated to be used as backdrops.

As part of the suggested ITGE/IGM IPB mapping initiative, these datasets could be used to generate useful products and combined images. Some manpower will be available in GEOAMB for this purpose.

Visualisation

Those are raster files in special formats (BIL, etc.). At present no support for image file formats exists in Geomist. Interesting solutions based on public domain software GMT (University of Hawaii) and derived products exist and could be integrated in the future. Meanwhile geo-referenced rasters (geoTIFF) at lower resolution would be derived and can be handled by Geomist.

Topographic Base

A common base map was produced from digital versions available at ITGE and IGM. It is a light 1:200.000 version suitable to be used through the Internet and it is consistent with the geological basis. It is an Arc/Info coverage with the road network, railways, rivers, dams, towns, administrative units, first order trig points, etc. However following reviewers suggestions contacts with the PETIT project will be established. The fact that the product is at 1:250.000 would not be a relevant factor, and an overview of the PETIT prototype specifications indicates that it seems suitable for Geomist. An import facility could be created in Geomist to replace the present base map by the PETIT one at a later stage. This would be quite helpful for a generalisation of Geomist to the Iberian peninsula, France, UK and Germany as suggested in future plans. PETIT maps also include elevation data, and the coarse DEM (see above) might also be replaced in the future.

Visualisation

This covers point and line data and readily handled.

Digital Elevation Model

This was not mentioned as a coverage in previous listings but a DEM , where geology could be draped, geological cross-sections cut across, and 3D models build upon is an important dataset. A sufficient detailed DEM may be bought, but its costs are not contemplated in Geomist. As an alternative we built a simplified DEM from the gravity coverages, creating a model which has a resolution of roughly 1.5 Kmx1.5 Km. This is what is available at present.

Under GEOAMB a DEM for most of the Portuguese IPB (at around 50mx50m), will be bought, but this will only happen later in 2000. A more comprehensive and satisfactory solution might be reached in case there is sufficient demand form the Geomist user group and funding or shared buying is agreed upon.

Visualisation

Data is available as point data, TIN, vector (coloured coded unlabeled contour lines every 25 m) and grid file. It may be visualised as a vector coverage together with the other coverages as part of the 2D interface, which also allows the exploration of the point data which may be visualised as a sort of labelled spot elevations jointly with other coverages (raster or vector). The TIN and grid files are intended for use later within the 3D viewers or for download.

Bibliography (technical reports)

A survey of the available bibliography for the IPB, existent at ITGE under the GEOMINER database, showed that it is too small for the known literature body (books, journals and thesis) in the case of the IPB. As mentioned before no similar database exists at IGM.

However with the help of the users group, experts working in the IPB including those working at ITGE and IGM, and by accessing GEOREF we hope to build a catalogue with the most important references.

Different is the case of the technical reports. Some 1000 titles exist at IGM and are organised in an Oracle database. Also under GEOAMB a special programme for scanning, OCRing reports (in Portuguese, English, French and Spanish) and Marking Up using a Document Type Definition based on HTML 4.0 and the Dublin Core is underway. Because the interface is a browser the service is easily integrated in Geomist. It will be in fact incorporated in Geomist and will be demonstrated at the Beja workshop. A total of about 90% of the company reports for the Portuguese IPB have already been contracted out and will be available until the end of September. The text search engine is under development, but a prototype exists already and will be demonstrated at the Beja workshop. Keywords are based on the ITGE Geoscience thesaurus expanded to Portuguese in Geomist. Similar strategies could be adopted by SIGMA.

The adopted Data model is suitable for both books, journals and technical reports. The documents themselves are stored as text files (according to pre-defined rules) and not in Oracle.

Visualisation

The text and figures are visualised as web pages with a browser navigation system based on the index of the document. Because documents are related to mining permits a geographical navigation system is also possible, whereby the user gets the relevant documents for the permit(s) and from then on can navigate through the document and apply text searches.

Geophysical maps

This category includes the aeromagnetic, gravity and spectrometric coverages described above. But it also includes scanned maps for gravity, resistivity and induced polarisation, old TURAM surveys, Airborne INPUT, reflection section maps, at 1:5.000 and 1:25.000 scales. Those are provided as raster geo-referenced images. for the Portuguese IPB. It was contracted out under GEOAMB. A limited amount (10% of the total) is available and will be demonstrated at the Beja workshop. The remainder will be available within three months.

Visualisation

The idea is the use of such raster maps as backdrops for visual correlation with other coverages, to download for printing, off-line analysis like for example pin-pointing geophysical anomalies. They integrate directly into the server interactive mapping window. Both a high resolution 300 dpi image (too heavy for interactive use in most cases) and a thumbnail version exists for building mosaics and overviews.

Geophysical surveys

This dataset was not mentioned in the previous inventory, but it would be an important add-on. There are useful paper and digital files at both geophysics sectors in ITGE and IGM and an agreed data model could easily be built. Its importance is mostly in correctly providing metadata for all the geophysical datasets. In fact despite all the harmonisation strategies there is a point when a user wants to get down to the "raw" data.

That might involve work from both parties. But the work coming out from GEOAMB (which is addressing this topic) and what is already available in Spain could be combined and made available.

Visualisation

This data is of polygon and line type and is readily handled.

Gravimetric base network of the IPB

This was not considered in previous inventories, however it would be very helpful to mining companies conducting gravity surveys, because it would mean they could tie their surveys immediately to the IPB database. Work in Spain was concluded by ITGE long ago and in Portugal IGM finished it recently. Even using the published data it would be very easy to incorporate this database. IGM will take that in charge the task of incorporating this in the web server.

Visualisation

This data is of point type and includes raster images (pictures of bases). The first are handled and the second are easily accomodated (event files).

Spatial mapping indexes

This is provided as an ancillary polygon coverage for overlay, query, searches and spatial indexing. It has information on the cartographic series. The data model is a simple one. It might eventually evolve in articulation with the mapping agencies, so as to adhere to their nomenclature.

Visualisation

Data is explored as polygons either transparent or coloured. Some links to update information from the mapping agencies (for example last edition of a 1:25.000 map), might be useful.

Lineament maps

This is a derived set from the analysis of remote sensing data and exists only for Spain. It exists as Arc/Info coverages. It is not part of Geomist, but it can be easily included as such. In the future once a similar dataset exists for the Portuguese IPB, the data model may eventually be reviewed. But because of its simple structure this is not foreseen.

Visualisation

Data is explored as lines, so this is handled directly by the interactive mapping server of Geomist.

Exploration permits

An historical overview of the exploration permits exists in Portugal and Spain as a coverage of polygon type. This serves several purposes: to know about free areas to apply for exploration; to know about areas that were explored before and when; to use as a spatial index for searches at all levels of geological and exploration data.

The data model is a suitable one and covers both the situation in Portugal and Spain.

Visualisation

Data is explored as polygons either transparent or coloured. Some links with the licensing agents to keep data up to date is recommended. Small changes in the data model are anticipated to accommodate this.

Hydrogeological data

This was not explicitly considered before. However it is logical to consider it in view of the fact that both ITGE and IGM may directly contribute. Reference was made in previous reports to the existence of the EC funded "Evaluation of the Groundwater Resources of Alentejo". Results of this project will be presented at the Beja workshop, and will make evident the need for co-ordinated action with Geomist. In the future provision to include its results (it being a SIG based project) will be discussed with the participating agencies. This is clearly geological information of interest to land management (the co-ordinating agency). On the other hand water resources are clearly trans-border issues, which have been the object of negotiation at high level between the Portuguese and Spanish governments. A modern view of the problem considers that both ground water and surface waters should be jointly modelled.

This type of information is in fact an aggregate of coverages spanning: wells, aquifers, piezometers, meteorological data (precipitation, evapo-transpiration) and chemical analytical data. Within Geomist we would consider only the first two. Data models and harmonisation issues would have to be handled (namely consistent use of geological formation names at surface and in depth, expansion of drill-hole data models to cover water wells).

The analysis and display of groundwater reservoirs is also an area of interest for 3D modelling. At the Beja workshop this will be raised as a topic for discussion, and the inclusion of the agencies involved into the user group (to be later enlarged to the corresponding agencies in Spain) is suggested.

This would involve a certain amount of work and the regional agencies will be made aware of the benefits that would arise from taking advantage of the Geomist server and concept. This would be a separate action for which funding would have to be sought and partnerships forged.

Visualisation

Well data will be explored as point data in the same way as drill-holes (see above, for which the 3D considerations also apply). Polygons either transparent or coloured is the type of data of the aquifers.

Geological Formations

This was not considered before. However while implementing the geological data model it became clear the importance of making it publicly available. This would provide a sound basis for the mining companies involved in the production of cartography, hydrogeologists, etc. The Australian Geological Survey Organisation has successfully followed this approach.

As a result of the geological model developed in Geomist preliminary results are already available. However a participation of the cartography departments at ITGE and IGM will be necessary in order to streamline the process. This is not a costly process and it should be associated with the proposed mapping initiative.

Visualisation

Data is of text type and is viewed as HTML.

Heat Flow data

It was not considered before. It is useful for the scientific community and some compilation work was done in Spain and Portugal. A recent work has been published. As part of the Geophysical Atlas of Portugal this information and some additional one is available in ArcView. We will include the whole set in Geomist (point data), clipped to the polygon limits, because the database is quite small. New maps will be generated and clipped to the Geomist polygon. This is being done in the framework of GEOAMB.

Visualisation

Data may be visualised as point data (drill-holes and wells surveyed), geo-referenced colour raster with the heat flow or vector colour contours with the heat flow data. This is the type of information easily dealt by the 2D interactive map service of Geomist. Temperature logs, also available, fall in the category of geophysical logs and the same considerations apply as discussed under drill-holes.

Thermal Springs

Maps of thermal springs are available as ArcView coverages at both ITGE and IGM. A subset for the Geomist polygon may easily be created. No harmonisation will be attempted at this stage (just the inclusion of UTM-29 co-ordinates). According to the location of the point picked is in the Portuguese or Spanish IPB, the corresponding attribute table will be pulled and displayed.

Visualisation

This is point data and is easily handled as such.

Paleomagnetic data

An Access database sponsored by the American Geophysical Union is publicly available. Some samples of this world-wide coverage belong to the IPB. This database maybe converted into Oracle and ArcView coverages (adopting the corresponding data model).

Visualisation

This is point data, but to be viewed as such it will demand some work. Before this is done a georeferenced raster with the posted points and a link to the AGU web site were data is available is a good alternative.

Deep Seismic Refraction Profiles

It was briefly mentioned in the initial presentations of Geomist that some projects like ILHIA and Anisotropy experiments in the Peninsula under EuroProbe have covered the area. There are references with co-ordinates of the profiles and other relevant data.

Visualisation

Data consists of profiles, records, record sections, inverted cross-sections. For the moment only line data (profiles) is displayed and a limited range of models of cross-sections are presented.

Magnetotelluric profiles

It was briefly mentioned in the initial presentations of Geomist, that some work done by academia both in Portugal and Spain provided a few results. For the moment only location of the profiles, and text with figures of inverted models with references to the publications will be given, with author's permissions.

Visualisation

Data consists of profiles, records, resistivity curves and inverted cross-sections. For the moment only line data (profiles) is displayed and scans of models are presented.

3D models

For the moment there are only two: the 3D demo model of Geomist in Spain and the Furadouro model in Portugal. Under agreement between BRGM and IGM in the framework of GeoFrance3D more models will be generated for the Portuguese IPB (Aljustrel and global model for the South Portuguese zone). More are expected in the future, as a part of the preparation of a proposed geotraverse in the region. For this the existence of a geological editor, improved data models for drill-holes and compatible packages for geophysical inversion and visualisation are very important. Work proceeds in those topics and other actions will be submitted for funding.

Visualisation

The visualisation is done through the 3D viewer developed by Matra, and based in Cas.Cade.

ANNEX 2

3D DATA SET

3D DATA SET PREPARATION

The objective of this work is the preparation of a data set from which to generate a 3D model using the Beicip-Franlab software, GEOSURF. The following tasks are needed:

Gathering the existent data in the Iberian Pyrite Belt.

Study the OMEGA data input objects and identify the existent data in the Iberian Pyrite Belt with some type of OMEGA Objects.

Select an area of the Iberian Pyrite Belt where data density is enough for modelling purpose.

The dataset used to create the 3D model is given below the following. Each record is a well with lithology, deep information and a well identifier.

Deep and lithology information

```
ID,ID_SOND,DEEP,LITOLOGIA

878,130,76.00,lava intermedia
879,130,104.00,toba de grano fino
880,130,130.00,mineral
881,130,146.00,toba de grano fino
882,131,55.00,lava intermedia
883,131,92.00,toba de grano fino
884,131,105.00,mineral
885,131,112.00,toba de grano fino
886,132,63.00,lava intermedia
887,132,85.00,toba de grano fino
888,132,97.00,mineral
889,132,115.00,toba de grano fino
890,132,120.00,lava acida
891,133,108.00,lava intermedia
892,133,152.00,toba de grano fino
893,133,170.00,mineral
894,133,192.00,toba de grano fino
895,134,115.00,lava intermedia
896,134,157.00,toba de grano fino
897,134,194.00,mineral
898,134,201.00,toba de grano fino
899,135,110.00,lava intermedia
900,135,150.00,toba de grano fino
901,136,104.00,lava intermedia
902,136,136.00,toba de grano fino
903,136,142.00,mineral
```

```
904,136,182.00,toba de grano fino
905,136,192.00,lava acida
906,137,168.00,lava intermedia
907,137,221.00,toba de grano fino
908,137,225.00,mineral
909,137,338.00,toba de grano fino
910,138,150.00,lava intermedia
911,138,188.00,toba de grano fino
912,138,198.00,mineral
913,138,216.00,toba de grano fino
914,138,230.00,mineral
915,138,257.00,toba de grano fino
916,139,150.00,lava intermedia
917,139,216.00,toba de grano fino
918,139,259.00,mineral
919,139,272.00,toba de grano fino
920,140,110.00,lava intermedia
921,140,135.00,toba de grano fino
922,140,138.00,mineral
923,140,147.00,toba de grano fino
924,140,154.00,lava acida
925,141,108.00,lava intermedia
926,141,140.00,toba de grano fino
927,141,146.00,mineral
928,141,165.00,toba de grano fino
929,142,71.00,lava intermedia
930,142,165.00,toba de grano fino
931,142,185.00,mineral
932,142,205.00,toba de grano fino
933,143,72.00,lava intermedia
934,143,126.00,toba de grano fino
935,143,142.00,mineral
936,143,160.00,toba de grano fino
937,143,185.00,mineral
938,143,198.00,toba de grano fino
939,144,155.00,lava intermedia
940,144,230.00,toba de grano fino
941,144,243.00,mineral
942,144,260.00,toba de grano fino
943,145,110.00,lava intermedia
944,145,152.00,toba de grano fino
945,145,166.00,mineral
946,145,190.00,toba de grano fino
947,145,194.00,mineral
948,145,200.00,toba
949,145,204.00,mineral
950,145,223.00,toba de grano fino
```

```
951,145,240.00,lava acida
952,146,46.00,lava intermedia
953,146,111.00,toba de grano fino
954,146,127.00,mineral
955,146,151.00,toba de grano fino
956,146,167.00,mineral
957,146,180.00,toba de grano fino
958,147,90.00,lava intermedia
959,147,160.00,toba de grano fino
960,147,200.00,lava acida
961,148,95.00,lava intermedia
962,148,174.00,toba de grano fino
963,148,198.00,lava acida
964,149,50.00,lava intermedia
965,149,131.00,toba de grano fino
966,149,140.00,mineral
967,149,153.00,toba de grano fino
968,149,157.00,mineral
969,149,189.00,toba de grano fino
970,150,61.00,lava intermedia
971,150,97.00,toba de grano fino
972,150,110.00,mineral
973,150,140.00,toba de grano fino
974,150,158.00,mineral
975,150,178.00,toba de grano fino
976,151,98.00,lava intermedia
977,151,105.00,toba de grano fino
978,151,107.00,mineral
979,151,111.00,toba de grano fino
980,151,113.00,mineral
981,151,137.00,toba de grano fino
982,151,158.00,mineral
983,151,176.00,toba de grano fino
```

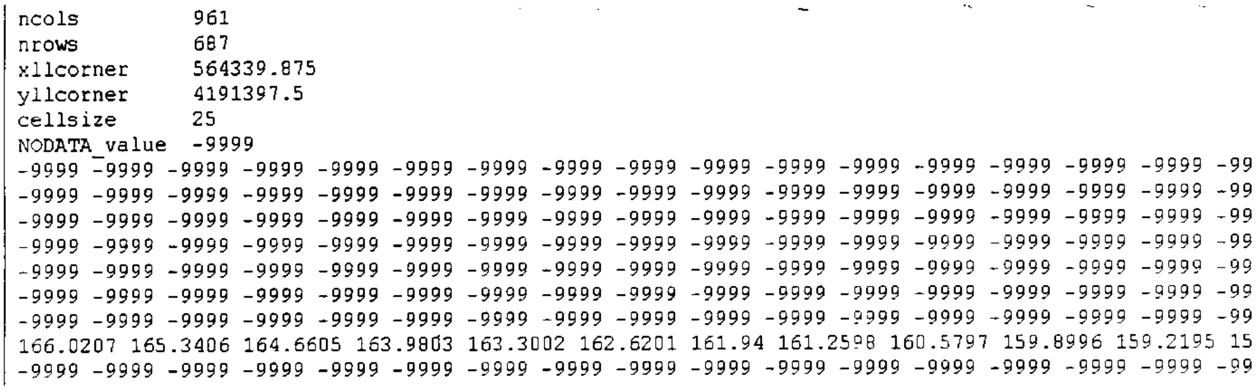
Well location and well name information

```
"ID,"Xutm29","Yutm29","Cota","Well name","Inclinac"
130,646931,4172718,146.00,B-1 PAYMOGO,0
131,646830,4172743,112.00,C-1 PAYMOGO,0
132,647028,4172774,120.00,A-1 PAYMOGO,0
133,646825,4172823,192.00,C-2 PAYMOGO,0
134,646925,4172828,201.00,B-2 PAYMOGO,0
135,646723,4172857,150.00,D-2 PAYMOGO,0
136,647023,4172859,192.00,A-2 PAYMOGO,0
137,646422,4172869,338.00,G-3 PAYMOGO,0
138,646821,4172893,257.00,C-3 PAYMOGO,0
139,646920,4172905,272.00,B-3 PAYMOGO,0
```

140,646516,4172943,154.00,F-2 PAYMOGO,0
141,646616,4172971,165.00,E-2 PAYMOGO,0
142,646914,4173015,205.00,B-4 PAYMOGO,0
143,646813,4173022,198.00,C-4 PAYMOGO,0
144,646611,4173050,260.00,E-3 PAYMOGO,0
145,646710,4173076,240.00,D-5 PAYMOGO,0
146,646807,4173132,180.00,C-5 PAYMOGO,0
147,646603,4173180,200.00,E-4 PAYMOGO,0
148,646703,4173186,198.00,D-6 PAYMOGO,0
149,646903,4173195,189.00,B-5 PAYMOGO,0
150,646800,4173242,178.00,C-6 PAYMOGO,0
151,646796,4173322,176.00,C-7 PAYMOGO,0

Other necessary information is Digital Terrain Model of the area and surface geology.

ASCII data grid with information about the Z coordinate of the different cells that conforms the grid, the data structure is as follow:

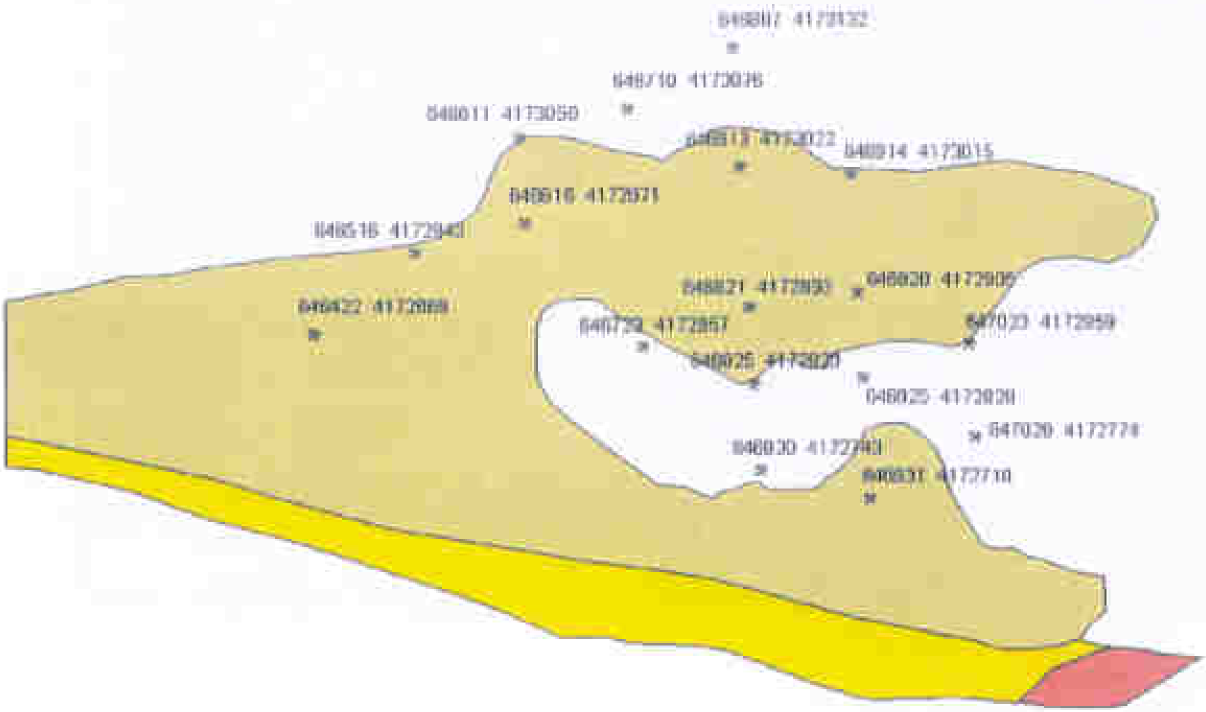


```
ncols         961
nrows         687
xllcorner     564339.875
yllcorner     4191397.5
cellsize      25
NODATA_value  -9999
-9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -99
-9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -99
-9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -99
-9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -99
-9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -99
-9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -99
-9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -99
166.0207 165.3406 164.6605 163.9803 163.3002 162.6201 161.94 161.2598 160.5797 159.8996 159.2195 15
-9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 -99
```

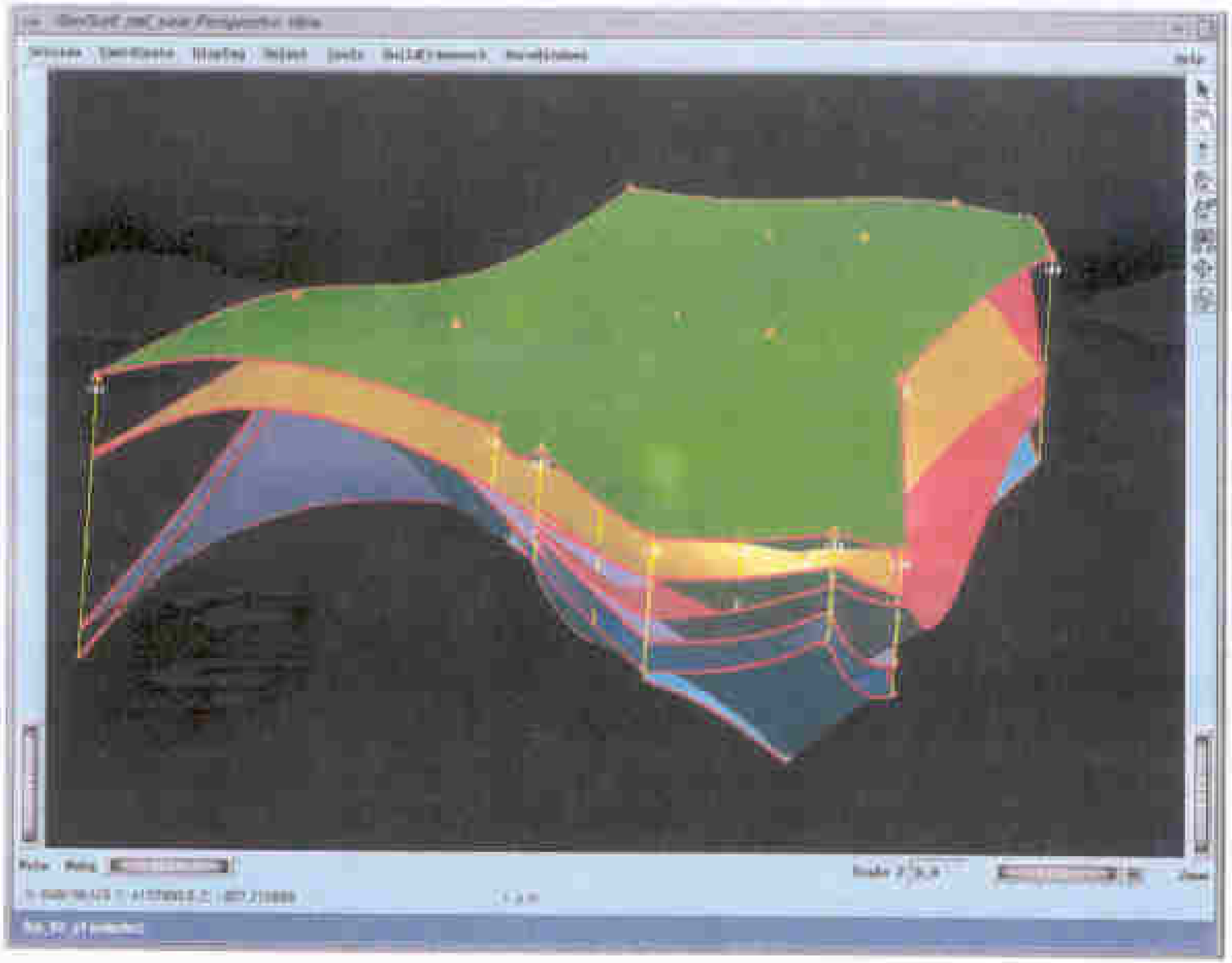
The ASCII information file include the number of lines, number of columns, origin coordinate and the cell size in meters, besides the value of the Z for each cell.

The geographical environment reaches the whole area of the Iberian Pyrite Belt previously selected for the existent well data.

2D View with lithology information:



3D View



3D exchange format Program

```
Sub convert_well ()
Dim db As database
Dim tb As table
Dim HOLE_ID, HOLE_IDN As String
Dim X_UTM, Y_UTM, Z_UTM As Single
Dim FORM_UNIT As String
Dim azimuth, INCLINATIO As Integer
Dim MIN, MAX As Single
Dim i As Integer
Dim X1, Y1, Z1 As Single
Dim X2, Y2, Z2 As Single
Dim PI As Single
PI = 3.141593
Me.MousePointer = 11
```

```
Set db = OpenDatabase("d:\geomist\brgm\zona3d.mdb")
Set tb = db.OpenTable("bore3d29_2")
tb.Index = "HOLE_ID"
HOLE_ID = ""
i = 1
While Not tb.EOF
  HOLE_IDN = tb("HOLE_ID")
  X_UTM = tb("X_UTM")
  Y_UTM = tb("Y_UTM")
  Z_UTM = tb("COTA")
  azimuth = tb("AZIMUTH") * PI / 180
  INCLINATIO = tb("INCLINATIO") * PI / 180
  MIN = tb("MinDeFROM")
  MAX = tb("MaxDeTO")
  FORM_UNIT = tb("FORM_UNIT")
  If IsNull(azimuth) Then azimuth = 0
  X1 = MIN * Cos(-INCLINATIO) * Cos(azimuth)
  X2 = MAX * Cos(-INCLINATIO) * Cos(azimuth)
  Y1 = MIN * Cos(-INCLINATIO) * Sin(azimuth)
  Y2 = MAX * Cos(-INCLINATIO) * Sin(azimuth)
  Z1 = MIN * Sin(-INCLINATIO)
  Z2 = MAX * Sin(-INCLINATIO)
  If HOLE_ID <> HOLE_IDN Then
      HOLE_ID = HOLE_IDN
      Close #1
      Open "d:\geomist\conver\datos\" & HOLE_IDN & ".wel" For Output As #1
      Print #1, "APPLICATION          # HERESIM3D # Version3.3"
      Print #1, ""
      Print #1, "SURVEY_NAME        # FAJAPIRITICA"
      Print #1, ""
      Print #1, "WELL_NAME         # " & HOLE_ID
      Print #1, ""
      Print #1, "UNIQUE_WELL_ID       #" & i
      Print #1, ""
      Print #1, "PLOT_SYMBOL          # 1"
      Print #1, ""
      Print #1, "WELL_DEVIATED        #  YES         #"
      Print #1, ""
      Print #1, "UP_DIRECTION         #  DOWN          #"
      Print #1, ""
      Print #1, "REPRESENTATION_CODE   # ASCII"
      Print #1, ""
      Print #1, "#######################"
      Print #1, ""
      Print #1, "DATUM_ELEVATION       # 0"
      Print #1, ""
      Print #1, "X_LOCATION      # " & X_UTM
```

```
        Print #1, ""
        Print #1, "Y_LOCATION      # " & Y_UTM
        Print #1, ""
        Print #1, "Z_LOCATION      # " & Z_UTM
        Print #1, ""
        Print #1, "TRUE_VERTICAL_DEPTH   # CHANNEL_1"
        Print #1, ""
        Print #1, "X_DEVIATION         # CHANNEL_2"
        Print #1, ""
        Print #1, "Y_DEVIATION         # CHANNEL_3"
        Print #1, ""
        Print #1, "WELL_MARKER_ID       # CHANNEL_4"
        Print #1, ""
        Print #1, "WELL_LITHOFACIES_BOT  # CHANNEL_5"
        'Print #1, ""
        'Print #1, "WELL_POROSITY        # CHANNEL_6"
        Print #1, ""
        Print #1, "########################"
        Print #1, Format$(Z1, "0.######"), "#", Format$(X1, "0.######"), "#", Format$(Y1,
"0.######"), "# # "
        Print #1, ""
        i = i + 1
  End If
  Print #1, Format$(Z2, "0.######"), "#", Format$(X2, "0.######"), "#", Format$(Y2,
"0.######"), "# #" & FORM_UNIT
  Print #1, ""
  tb.MoveNext
Wend
tb.Close
db.Close
Me.MousePointer = 0
End Sub


Sub convert_grid ()
Dim a As Single
Open "d:\geomist\conver\datos\topo.gr2" For Output As #1
Open "d:\geomist\brgm\elev_gri.txt" For Input As #2
Print #1, "FILE_DESCRIPTION       # GR2D       # 4.1"
Print #1, ""
Print #1, "APPLICATION           # no comment # Heresim"
Print #1, ""
Print #1, "SURVEY_NAME            # GEOMIST"
Print #1, ""
Print #1, "SURFACE_NAME           # surface"
Print #1, ""
Print #1, "INTERFACE_ID          # 16"
Print #1, ""
Print #1, "SURFACE_TYPE            # REGULAR_GRID # UNKNOWN"
Print #1, ""
Print #1, "UP_DIRECTION            # UP"
Print #1, ""
Print #1, "INT_NULL_VALUE          # -9999"
Print #1, ""
Print #1, "FLOAT_NULL_VALUE        # -9999.00000"
Print #1, ""
Print #1, "VALUE_LOCATION          # CORNER"
Print #1, ""
Print #1, "ROW_COLUMN_ORIENTATION  # ROW"
Print #1, ""
Print #1, "REPRESENTATION_CODE     # ASCII"
Print #1, ""
Print #1, "##########################"
Print #1, ""
Print #1, "ANGLE           # 0.000000 # DEG"
Print #1, ""
Print #1, "ROW_COUNT               # 961"
Print #1, ""
Print #1, "COLUMN_COUNT            # 687"
Print #1, ""
Print #1, "X_ORIGIN        # 564339.875 # m"
Print #1, ""
Print #1, "Y_ORIGIN        # 4191397.5 # m"
Print #1, ""
Print #1, "ROW_DISTANCE            # 25.00000 # m"
Print #1, ""
Print #1, "COLUMN_DISTANCE         # 25.000000 # m"
Print #1, ""
Print #1, "Z_VALUES                # CHANNEL1 #    # m"
Print #1, ""
'Print #1, "CONNECT_ID              # CHANNEL2 #    # DISCRET"
'Print #1, ""
Print #1, "##########################"
Print #1, ""

Input #2, a
Input #2, a
Input #2, a
Input #2, a
Input #2, a
Input #2, a
Input #2, a
While Not EOF(2)
   Input #2, a
```

```
  Print #1, Format$(a, "0.00000")
  DoEvents
Wend
Close #1
Close #2
End Sub

Sub convert_poly ()
Dim FORM_UNIT As String
Dim i As Integer
Dim a As String
Dim x, y As String
Dim ab As String
i = 1
Open "d:\geomist\conver\datos\ff.pol" For Input As #2
Input #2, y
      Open "d:\geomist\conver\datos\faja" & Trim(i) & ".pol" For Output As #1
      FORM_UNIT = y
      Input #2, x
      Print #1, "FILE_DESCRIPTION     # POLYG     # 4.2"
      Print #1, ""
      Print #1, "APPLICATION          # translate # RML"
      Print #1, ""
      Print #1, "SURVEY_NAME          # GEOMIST"
      Print #1, ""
      Print #1, "SURFACE_NAME         # " & FORM_UNIT
      Print #1, ""
      Print #1, "INTERFACE_ID         # " & i
      Print #1, ""
      Print #1, "UP_DIRECTION         # UP"
      Print #1, ""
      Print #1, "FLOAT_NULL_VALUE     # -99999."
      Print #1, ""
      Print #1, "INT_NULL_VALUE       # -9999"
      Print #1, ""
      Print #1, "REPRESENTATION_CODE  # ASCII"
      Print #1, ""
      Print #1, "########################"
      Print #1, ""
      Print #1, "POLYGON_ID           # CHANNEL1 # polygon_id # DISCRET"
      Print #1, ""
      Print #1, "X_VALUES             # CHANNEL2 # x_val # m"
      Print #1, ""
      Print #1, "Y_VALUES             # CHANNEL3 # y_val # m"
      Print #1, ""
      Print #1, "GEOMETRY_CODE        # CHANNEL4 # geometry_code # CODE"
      Print #1, ""
      Print #1, "########################"
      Print #1, ""
      Print #1, ""
      i = i + 1
While Not EOF(2)
  Input #2, x, y
  If x = "END" Then
      Close #1
      Open "d:\geomist\conver\datos\faja" & Trim(i) & ".pol" For Output As #1
      'Input #2, FORM_UNIT
      'Input #2, ab
      FORM_UNIT = y
      Input #2, y
      'FORM_UNIT = y
      Print #1, "FILE_DESCRIPTION     # POLYG     # 4.2"
      Print #1, ""
      Print #1, "APPLICATION          # translate # RML"
      Print #1, ""
      Print #1, "SURVEY_NAME          # GEOMIST"
      Print #1, ""
      Print #1, "SURFACE_NAME         # " & FORM_UNIT
      Print #1, ""
      Print #1, "INTERFACE_ID         # " & i
      Print #1, ""
      Print #1, "UP_DIRECTION         # UP"
      Print #1, ""
      Print #1, "FLOAT_NULL_VALUE     # -99999."
      Print #1, ""
      Print #1, "INT_NULL_VALUE       # -9999"
      Print #1, ""
      Print #1, "REPRESENTATION_CODE  # ASCII"
      Print #1, ""
      Print #1, "########################"
      Print #1, ""
      Print #1, "POLYGON_ID           # CHANNEL1 # polygon_id # DISCRET"
      Print #1, ""
      Print #1, "X_VALUES             # CHANNEL2 # x_val # m"
      Print #1, ""
      Print #1, "Y_VALUES             # CHANNEL3 # y_val # m"
      Print #1, ""
      Print #1, "GEOMETRY_CODE        # CHANNEL4 # geometry_code # CODE"
      Print #1, ""
      Print #1, "########################"
      Print #1, ""
      Print #1, ""
      i = i + 1
  End If
```

```
    If x <> "END" Then
            Print #1, FORM_UNIT & "#" & x & "#" & y & "#"; "C"
            Print #1, ""
    Else
            Input #2, x, y
            Print #1, FORM_UNIT & "#" & x & "#" & y & "#"; "B"
            Print #1, ""
    End If
    a = x
Wend
End Sub
```

ANNEX 3

3D APPLICATION PROGRAMS

CONTENTS :

1. INTRODUCTION

This 3D demonstrator is to make a link between the two ESPRIT projects OMEGA and GEOMIST in order to develop tools to visualize OMEGA object and to export OMEGA entities such has grid, polygon and well as VRML files to be able to visualize them in any WEB browser.

The development of this 3D DEMONSTRATOR will be available on SUN SOLARIS platform by using Java 1.2, CAS.CADE 1.5B and OMEGA products.

By using the 3d demonstrator you are able to visualize 3 kind of object :

- Boreholes
- Horizon
- Areas on horizon defined by a convex limit

The partner involved in this task are :

- ITGE for GEOMIST
- Matra Datavision for GEOMIST and OMEGA
- Beicip Franlab for OMEGA.

This final report is to detail the different steps of the development of the 3D DEMONSTRATOR. Those steps had been validated by both ITGE and Matra.

2. OMEGA ASCII FILE FORMAT DESCRIPTION

Extract from « Format of Reservoir Modelling Files » Beicip Franlab

Structural representations (regular grid) files: < >.gr2d

This paragraph covers the header for files of 2D grids of surfaces <SURFACE_NAME>.gr2d.

Two cases are considered :

- Regular grids (REGULAR_GRID), and
- Irregular grids (PARAMETRIC).

Format of < >.gr2d files:

The following example is the format for the representation of a surface by a regular grid.

FILE_DESCRIPTION	#	GR2D	#	4.1	
APPLICATION	#	String	#	String	
SURVEY_NAME	#	String			
SURFACE_NAME	#	String			
INTERFACE_ID	#	Integer			
SURFACE_TYPE	#	String (REGULAR_GRID /	#	String (HORIZON / FAULT /	

		PARAMETRIC)		LITHOSTRAT / UNCONFORM / CHEMICAL / PTY_ONLY / UNKNOWN	
UP_DIRECTION	#	String (UP / DOWN)			
ROW_COLUMN_ORIENTATION	#	String (ROW / COLUMN)			
INT_NULL_VALUE	#	Integer			
FLOAT_NULL_VALUE	#	Real			
VALUE_LOCATION	#	String (CENTER/CORNER)			
REPRESENTATION_CODE	#	String (ASCII/BINARY)			
######################					
ANGLE	#	Real	#	Unit (DEG or RAD)	
ROW_COUNT	#	Integer			
COLUMN_COUNT	#	Integer			
X_ORIGIN	#	Real	#	Unit (String)	
Y_ORIGIN	#	Real	#	Unit (String)	
Z_ORIGIN	#	Real	#	Unit (String)	
ROW_DISTANCE	#	Real	#	Unit (String)	
COLUMN_DISTANCE	#	Real	#	Unit (String)	
X_VALUES	#	BULK 1	#	String	#Unit (String)
Y_VALUES	#	BULK 2	#	String	#Unit (String)
Z_VALUES	#	BULK 3	#	String	#Unit (String)
CONNECT_ID	#	BULK 4	#	String	#Unit (String)
DZ_VALUES	#	BULK 5	#	String	#Unit (String)
SUM	#	BULK 6	#	String	#Unit (String)
RATIO	#	BULK 7	#	String	#Unit (String)
SW	#	BULK 8	#	String	#Unit (String)
NTG	#	BULK 9	#	String	#Unit (String)
MISCELLANEOUS	#	BULK 10	#	String	#Unit (String)

.<surface_name>.x_values #.<surface_name>.y_values #.<surface_name>.z_values #

```
<surface_name>.connect_id # .<surface_name>.dz_values #.<surface_name>.sum1
#.<surface_name>.ratio #.<surface_name>.SW #.<surface_name>.NTG
#.<surface_name>.miscellaneous#
```

Glossary for < >.gr2d file format:

APPLICATION
Name of the application which generated the file, or for which the file was generated.

SURVEY_NAME
Study name.

SURFACE_NAME
Name of the surface.

INTERFACE_ID
Code number characteristic of the object being treated.

SURFACE_TYPE
cf .xyz (see above). In this case, a surface can be in terms of two types of geometry: a regular grid (REGULAR_GRID) or irregular (PARAMETRIC). The geological significance (LITHOSTRAT, UNCONFORM, CHEMICAL, UNKNOWN, HORIZON, FAULT, PTY_ONLY) are the same as for the .xyz format.
Similarly, it is obligatory to specify the type of geometry; the geological significance is optional (UNKNOWN by default).
If the geological significance is PTY_ONLY, the presence of Z_VALUES is not obligatory.

UP_DIRECTION
If the Z are negative, these are usually elevations (value = UP). If positive, they are probably depths (value = DOWN). [In general elevations are negative and depths are positive relative to the reference level (sea-level)].

VALUE_LOCATION
The location of data in each block of the grid . Values are CENTER or CORNER.

ROW_COLUMN_ORIENTATION
Type of grid: Row-oriented (=ROW) or Column-oriented (= COLUMN).

REPRESENTATION_CODE
Mode of.stocking the data in the container file (ASCII or BINARY).

ANGLE
Azimuthal angle between NS and the Y axis(see above figure) supposing a clockwise rotation about X_ORIGIN, Y_ORIGIN, Z_ORIGIN.

ROW_COUNT
Number of rows.

COLUMN_COUNT Number of columns.

X_ORIGIN, Y_ORIGIN Coordonnates of the origin of the 3D grid.

Z_ORIGIN Obligatory if SURFACE_TYPE = PARAMETRIC
ROW_DISTANCE Distance between two rows.

COLUMN_DISTANCE Distance between two columns.

X_VALUES } X
Y_VALUES } Y Dimensions of the blocks.
Z_VALUES } Z

CONNECT_ID Sector identifier, allows the organization of data in areas which have different characteristics (for example from one side of a fault to the other).

DZ_VALUES Thickness value.

SUM, RATIO, SW, NTG Name of an entity, or property, which has been assigned to a given point (e.g. sum, ratio, net or gross thickness, porosity, permeability).

MISCELLANEOUS Name of a supplementary property .... (e.g. porosity, permeability, coefficient, etc.)

The format .gr2d allows the description of a surface by two types of grid:

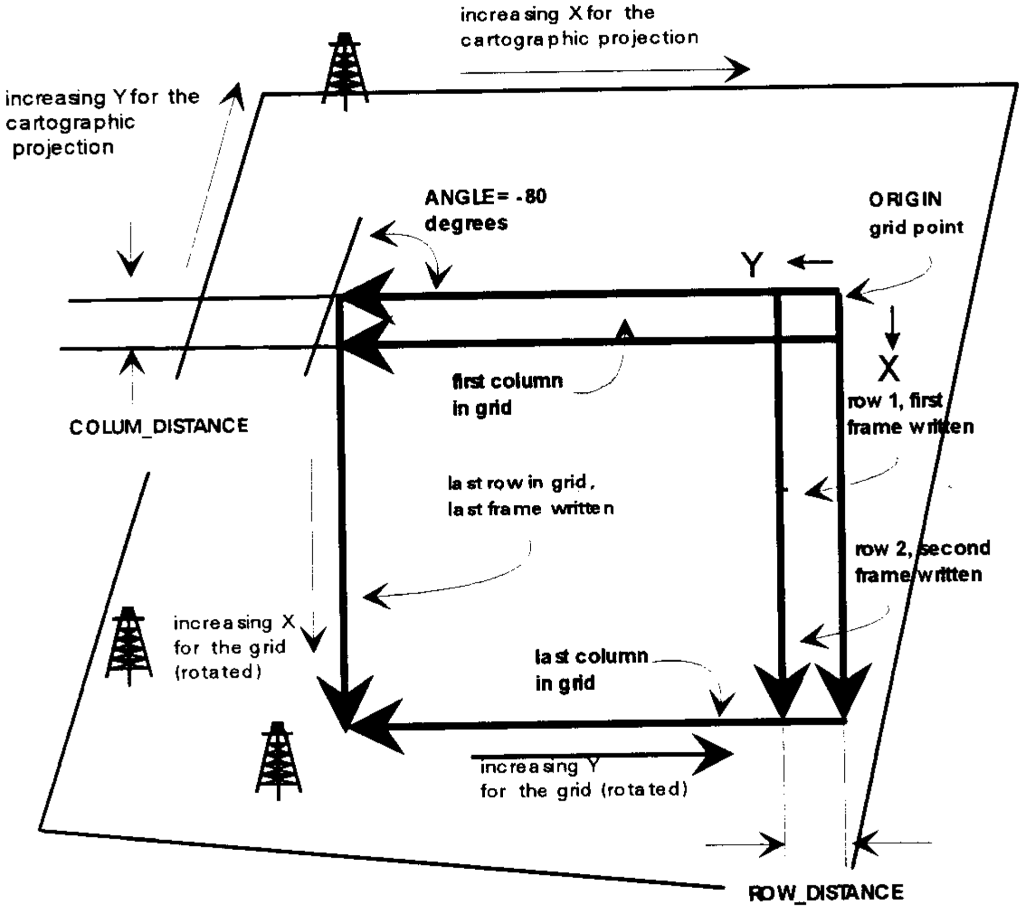
- regular (REGULAR), or
- irregular (PARAMETRIC).

A **regular grid** has an identical distance between each adjacent line, and also an identical distance between adjacent columns. The X and Y values, therefore do not have to be defined for each Z value. It suffices to define the two step distances.

In the following example, the grids can be stored in an identical fashion, line by line or column by column. The lines and columns are generally regularly spaced.
In the case of an orthogonal grid which has no azimuthal rotation, a line is a set of values with a common Y coordinate, and a column is a set of values with a common X coordinate.

A **parametric grid** imposes no constraint on the position of the points relative to each other. Each point has its own X, Y and Z coordinate. This type of grid can be used to define a surface which, for example, cuts the vertical several times (such as an overfolded surface).

For both types of grid, the origin is defined by X_ORIGIN, Y_ORIGIN and Z_ORIGIN. Though one might think that the origin always corresponds to the minimum X and Y values, **this is not obligatory**. By using negative values for ROW_DISTANCE or COLUMN_DISTANCE, one can invert the grid and have the origin at a point of Xand Y maximum - see the example below.


increasing X for the cartographic projection
increasing Y for the cartographic projection
ANGLE= -80 degrees
ORIGIN grid point
Y
X
first column in grid
row 1, first frame written
COLUM_DISTANCE
last row in grid, last frame written
row 2, second frame written
increasing X for the grid (rotated)
last column in grid
increasing Y for the grid (rotated)
ROW_DISTANCE

Example of defining a grid.
From 'RP66 Encoding for ***GEOSHARE*** *data exchange version 2.1'*

In **GEOSHARE**, the user can only exchange grids which are considered to be a collection of lines of values (ROW_ORIENTED), as in the example below. This means that the first values are given with Y = 1 and X varying from 1 to COLUMN_COUNT. However, **SGM** accepts either ROW_ORIENTED or COLUMN_ORIENTED grids and it is not uncommon that the data are organized differently.

So, we have implemented a specific keyword to take account of this situation: ROW-COLUMN_ORIENTATION which, when the value is ROW implies that the grid is ROW_ORIENTED, and, if the value is COLUMN, the grid is COLUMN_ORIENTATED.

Similarly, we have added another keyword in order to define more precisely the location of values within the grid. The values can be located either at the left-hand corner (VALUE_LOCATION = CORNER) or at the center of each block (VALUE_LOCATION = CENTER).

All the data, except the X_VALUES, Y_VALUES and Z_VALUES, can be affected by the correction related to the value of VALUE_LOCATION.

Depending on the values given to SURFACE_TYPE and VALUE_LOCATION, the file will contain different elements.
In the case of REGULAR_GRID/CENTER, the file will contain ROW_COUNT and COLUMN_COUNT, in all other cases, it will contain (ROW_COUNT+1) and (COLUMN_COUNT+1).

Note that, in the limit case of PARAMETRIC/CENTER, the file contains a line and a range of values outside the grid defined in X, Y, and Z. Normally these should be initialised as non-values.

Format extensions:

The SURFACE_TYPE value 'PTY_ONLY' allows for both regular and parametric grids, i.e. in cases where the Z value is not specified. From the user's point of view, this allows properties which are uniquely defined in 2D to be specified by X and Y values.

Optional lines [default values]:

FILE_DESCRIPTION	[type=UNKNOWN, version=-9999.]
UP_DIRECTION	[UP]
ROW_COLUMN_ORIENTATION	[ROW]
INT_NULL_VALUE	[-999]
FLOAT_NULL_VALUE	[-9999.]
ANGLE	[0.]
ROW_DISTANCE	} If SURFACE_TYPE = PARAMETRIC
COLUMN_DISTANCE	}
X_VALUES	} If SURFACE_TYPE = REGULAR_GRID
Y_VALUES	}
Z_VALUES	If SURFACE_TYPE (geological) = PTY_ONLY.
CONNECT_ID	

DZ_VALUES

MISCELLANEOUS1-5

Well files: < >.well

There are 5 different types of information contained in < >.well files, as follows:

- Trajectory,
- Marker,
- Facies zones (n.b. of zones along the well-bore containing only one lithofacies),
- Petrophysical properties,
- Dip.

The **trajectory** is indicated by TVD (True Vertical Depth) values, the absolute depth (Z) of each point, as well as the step-out in X and Y (by default in meters).

Markers are represented by a Code number, which is located at a given position on the trajectory. A Code Number is used as the 'Channels must contain data of the same type (in this case, just numerical).

Facies zones are located by a number which corresponds to a reference description in the .init.**Heresim3D** file description. This number is located at the lower limit of the zone.

The dip data are given as follows:

1. The dip azimuth (i.e. the clockwise angle between North [0°] and the dip direction).
2. The magnitude of dip (DEG).
3. The radius of infuence of the dip (diameter of the circle in which the dip is considered to be constant).

NOTE: The File <volume>.wells used by ACTRIS (for transfer to fluid-flow simulators) gives the trajectory in absolute X, Y, and Z.

The data are stored in ASCII format.

Format of < >.well files:

FILE_DESCRIPTION	#	WELL	#	4.1
APPLICATION	#	String	#	Comment
SURVEY_NAME	#	String		
WELL_NAME	#	String		
UNIQUE_WELL_ID	#	Integer		
PLOT_SYMBOL	#	Integer		
WELL_DEVIATED	#	String (YES / NO)		
INT_NULL_VALUE	#	Integer		

FLOAT_NULL_VALUE	#	Real				
UP_DIRECTION	#	String (UP / DOWN)				
REPRESENTATION_CODE	#	ASCII				
########################						
INDEX_MIN	#	Integer				
INDEX_MAX	#	Integer				
DATUM_ELEVATION	#	Real	#	Unit (String)		
X_LOCATION	#	Real	#	Unit (String)		
Y_LOCATION	#	Real	#	Unit (String)		
Z_LOCATION	#	Real	#	Unit (String)		
TRUE_VERTICAL_DEPTH	#	CHANNEL1	#	String	#	Unit (String)
X_DEVIATION	#	CHANNEL2	#	String	#	Unit (String)
Y_DEVIATION	#	CHANNEL3	#	String	#	Unit (String)
WELL_MARKER_NAME / WELL_MARKER_ID	#	CHANNEL4	#	String / Integer	#	Unit (String)
WELL_LITHOFACIES_BOT / WELL_LITHOFACIES_TOP	#	CHANNEL 5	#	String	#	Unit (String)
WELL_POROSITY	#	CHANNEL 6	#	String	#	Unit (String)
WELL_PERMEABILITY	#	CHANNEL 7	#	String	#	Unit (String)
DIP_AZIMUTH	#	CHANNEL 8	#	String	#	Unit (String)
DIP_ANGLE	#	CHANNEL 9	#	String	#	Unit (String)
DIP_EXTENSION	#	CHANNEL 10	#	String	#	Unit (String)
WELL_SW	#	CHANNEL 11	#	String	#	Unit (String)
MISCELLANEOUS1	#	CHANNEL 12	#	String	#	Unit (String)
MISCELLANEOUS2	#	CHANNEL 13	#	String	#	Unit (String)
MISCELLANEOUS3	#	CHANNEL 14	#	String	#	Unit (String)
###############################						
Real# Real# Real#Integer#Integer# Real# Real# Real# Real# Real# Real# Real# Real# Real						

Example of table for the ASCII file descriptor of a well (only the first seven CHANNELS are described).

1 (Real)		2 (Real)		3 (Real)		4 (Int)		5 (Int)		6 (Real)		7 (Real)
VALZ0	#											
VALZ1	#	VALXdev1	#	VALYdev1	#	1	#		#	VALporo1	#	VALperm1
VALZ2	#		#		#		#	33	#		#	
VALZi	#	VALxdevi	#	VALYdevi	#	3	#		#		#	
VALZn	#		#		#		#	40	#			
VALZn+ 1	#		#		#		#		#	VALporoj	#	VALperm2

Glossary for format of < >.well files:

APPLICATION — Name of the application which generated the file, or for which the file was generated.

SURVEY_NAME — Name of the study.

WELL_NAME — Name of the well-head.

UNIQUE_WELL_ID — Unique well identifier.

PLOT_SYMBOL — N° for the well-status symbol (refer to Manual of the application)

WELL_DEVIATED — Indicates if the well is deviated: (YES ou NO).

UP_DIRECTION — The Z values are either elevations (UP) or depths (DOWN). In general altitudes are negative and depths are positive below the reference level (usually sea-level).

REPRESENTATION _CODE — Mode of storing the data (= ASCII).

DATUM_ELEVATION — Position of local zero, relative to reference level (sea-level or the geoid).

X_LOCATION,Y_LOCATION — Position of the well head in X/Y (usually UTM coordinates), in meters.

Z_LOCATION — Elevation of the well-head relative to the reference level, in meters.

TRUE_VERTICAL DEPTH — Absolute depth (TVD - true vertical depth) relative to Z_LOCATION.

X_DEVIATION — Step-out X from X_LOCATION (obligatory if WELL_DEVIATED is YES).

Y_DEVIATION — Step-out Y from Y_LOCATION (obligatory if WELL_DEVIATED is YES).

WELL_MARKER_NAME — Name of marker intersecting the well.

WELL_MARKER_ID — Number of the interface intersecting the well (INTERFACE _ID). This is the marker number for the current study. It enables a link to be made between the surfaces, or horizons, or faults in the sections.

WELL_LITHOFACIES_BOT — Lower limit of a zone of a single lithofacies represented by a whole number code for the lithofacies established in the current study.

WELL_POROSITY — Instantaneous value of porosity.

WELL_PERMEABILITY — Instantaneous value of permeability.

DIP_AZIMUTH — Azimuthal direction of dip.

DIP_ANGLE — Magnitude (in DEG) of the dip.

DIP_EXTENSION — Radius of the circle within which the given dip is constant (meters)

WELL_LITHOFACIES_TOP — If the lithofacies zones are defined by WELL_LITHOFACIES_TOP, they are automatically converted to WELL_LITHOFACIES_BOT,

When creating the file, always follows the, WELL_LITHOFACIES_BOT convention.

Using both WELL_LITHOFACIES_BOT and WELL_LITHOFACIES_TOP in the same file, is forbidden.

WELL_SW — Water saturation.

MISCELLANEOUS1-3 — (cf .xyz, .gr2d).

Optional lines [default values]:

FILE_DESCRIPTION	[type = UNKNOWN, version = -9999.]
PLOT_SYMBOL	[0.]
WELL_DEVIATED	[NO]
UP_DIRECTION	[DOWN]
INT_NULL_VALUE	[-999]
FLOAT_ULL_VALUE	[-9999.]
INDEX_MIN	[1]

INDEX_MAX [calculated]

DATUM_ELEVATION [0.]

X_DEVIATION,

Y_DEVIATION,

WELL_MARKER_NAME,

WELL_MARKER_ID,

WELL_LITHOFACIES_BOT,

WELL_LITHOFACIES_TOP,

WELL_POROSITY,

WELL_PERMEABILITY,

DIP_AZIMUTH, DIP_ANGLE,

DIP_EXTENSION, WELL_SW,

MISCELLANEOUS1-3.

Polygon files:< >.POLYG

A polygon file of type <SURFACE_NAME>.polyg defines areas with convex limits.

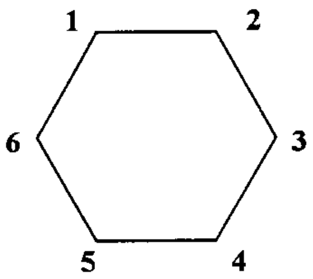
Such files contain information for points for which the X and Y cordinates are known but often the Z coordinates are unknown. Examples are:

- Licence areas,
- Geological zones with stationary properties,
- Fault traces on a surface, and
- Isocontours organized in polylines which are either open or closed (in such cases, the Z values are repeated for each point)

These data exist as segments of a given interface or surface. The points of each segment can have a meaning, which is either:

- **Geometric** (beginning of the segment or polygon, intermediate points, and the end of the segment or polygon), or
- **Topological** (being part of a convex, open or closed, contour).

In this case, the segments must be organised in a logical order. The segments are listed in the order of the corners.



This allows the polygon to be closed.

The data are obligatorily stored in ASCII format.

Format of <surface_name>.polyg file:

FILE_DESCRIPTION	#	POLYG	#	4.1		
APPLICATION	#	String	#	String		
SURVEY_NAME	#	String				
SURFACE_NAME	#	String				
INTERFACE_ID	#	Integer				
UP_DIRECTION	#	String (UP / DOWN)				
INT_NULL_VALUE	#	Integer				
FLOAT_NULL_VALUE	#	Real				
REPRESENTATION_CODE	#	ASCII				
##########################						
INDEX_MAX	#	Integer				
POLYGON_ID	#	CHANNEL1	#	String	#	Unit (String)
X_VALUES	#	CHANNEL2	#	String	#	Unit (String)
Y_VALUES	#	CHANNEL3	#	String	#	Unit (String)
GEOMETRY_CODE	#	CHANNEL4	#	String	#	Unit (String)
Z_VALUES	#	CHANNEL5 (optional)	#	String	#	Unit (String)
##########################						
Integer # float # float # string # float						

If there is no topological organisation, the value of POLYGON_ID is aways 0.

The values for the GEOMETRY_CODE is the same as for the sections (see above):

B (Beginning):	the first point of the polygon,
C (Current):	intermediate points (the default value),
E (End):	the last point of the polygon (before joining to B).

The polygons are always transferred as ASCII files; the CONTAINER mode cannot be used for this type of file.

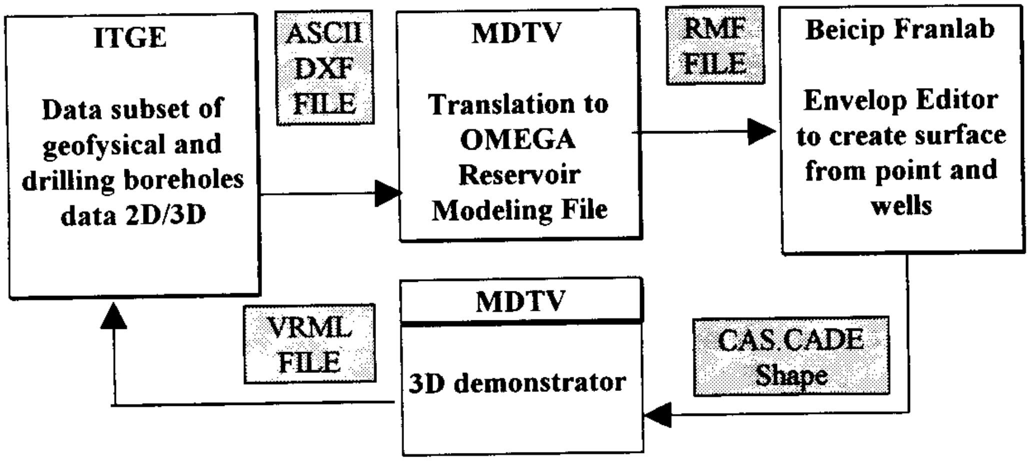
Optional line [default values]:

FILE_DESCRIPTION	[type = UNKNOWN, version = -9999.]
UP_DIRECTION	[UP]
FLOAT_NULL_VALUE	[-9999.]
INDEX_MAX	[calculated]
POLYGON_ID	[0.]
Z_VALUES	

3. 3D data model construction

The 3D Demonstrator will be built by using the OMEGA tools. Consequently, all 3D data should have to be treated with those tools. This requires different changes concerning the 3D data. This change was done in MDTV and Beicip Franlab and the 3D data model is know available in OMEGA tools format .

The different steps of work done by Beicip Franlab and Matra Datavision are presented in the following table.


ITGE
Data subset of geofysical and drilling boreholes data 2D/3D
ASCII DXF FILE
MDTV
Translation to OMEGA Reservoir Modeling File
RMF FILE
Beicip Franlab
Envelop Editor to create surface from point and wells
CAS.CADE Shape
MDTV
3D demonstrator
VRML FILE

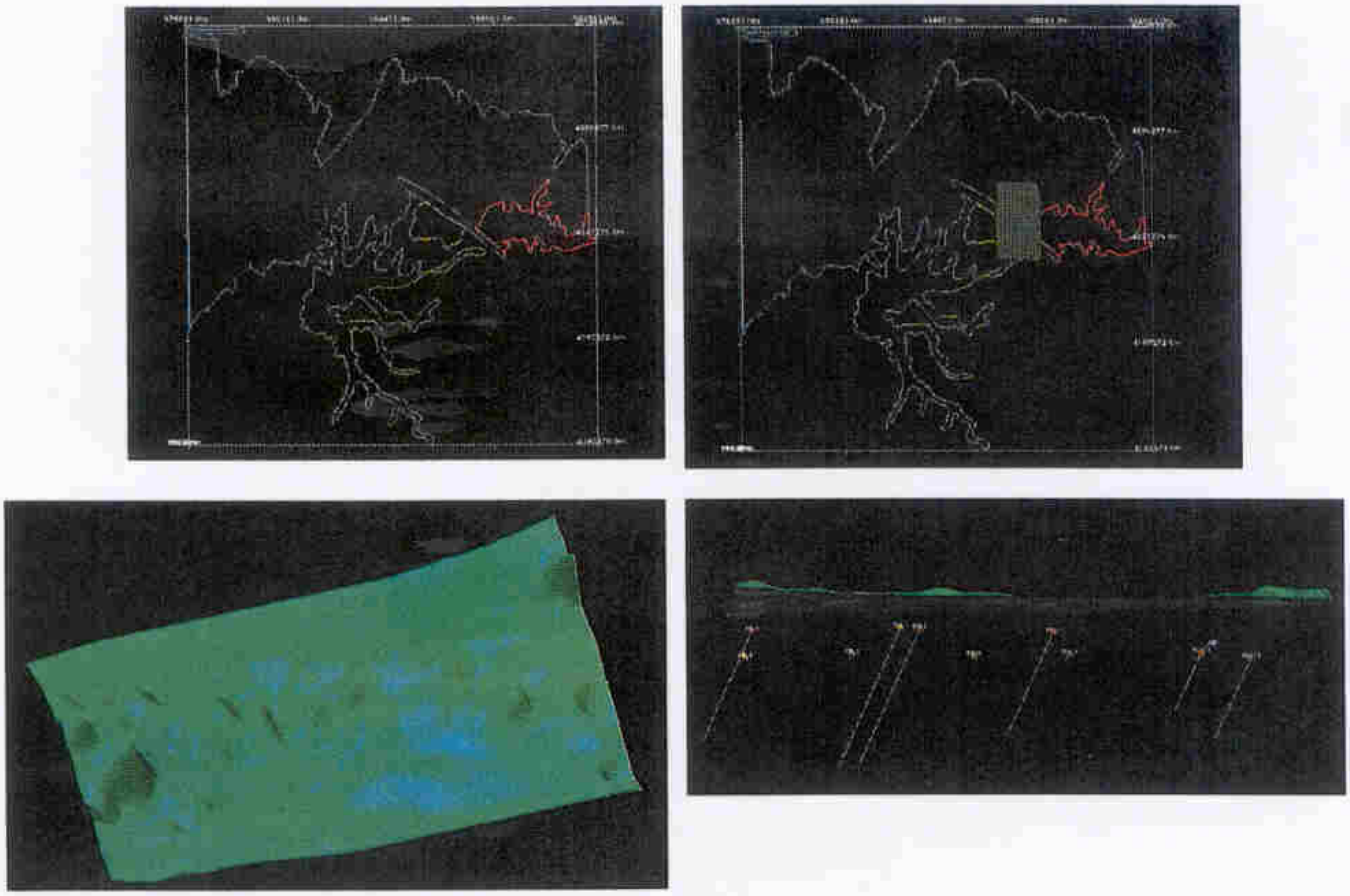
The selected OMEGA FORMAT are :

- ASCII file in which are described boreholes properties. (.well)

- ASCII and Brep file for horizon definition. (.gr2d, .brep)
- ASCII file for Subsurface limits. (.polyg)

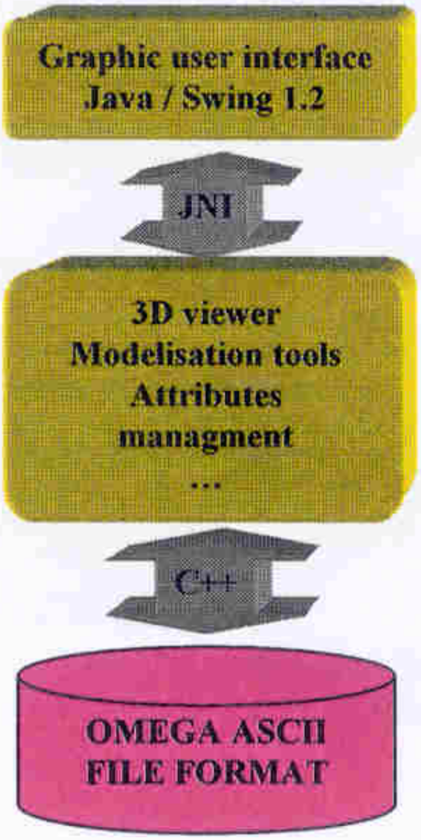
All this files are defined in OMEGA ASCII format and was validated by people from ITGE and BEICIP FRANLAB before using it in the 3d demonstrator by MATRADATAVISION.

The result of this action is a boundary representation of the horizon, well and subsurface wich can be loaded in the envelop editor OMEGA application as shown in the folowing picture.



4. 3d demonstrator architecture

The 3d demonstrator is based on Java1.2, Cas.Cade 1.5B and OMEGA tools. Java will be used for the graphic user interface and Cas.Cade and OMEGA to create a 3d data model of the geological object and to visualize it. A VRML output of this geological object are available by using Cas.Cade functionality. Java was choose in order to insure portability between UNIX and PC platform for future development on PC. Cas.Cade and OMEGA development are pure C++ and then can be reused directly on PC.



5. JAVA / CAS.CADE interface

In order to use easaly the C++ components from Java, we have to be able to manipulate C++ class exaclty in Java as in C++. We must insure the same syntax to be able to use the C++ documentation of Cas.Cade and OMEGA C++ object from Java.

Prerequisite :

- We must reproduce the same hierarchy of C++ object on the Java side to have the same type checking as in C++
- The Java finalize must delete the C++ object
- All the Java classes mirror of C++ class must implement a native Create method to instantiated the C++ object
- The Java Package name for a Cas.Cade package Pack is JPack
- The Java class name for a Cas.Cade class Pack_Class is JPack.Class

In the next chapter we will first focus on the argument of C++ method and next on the returned value.

There is 2 kind of argument and return value in Cas.Cade from a C++ point of view, basic type such as Integer, Real, Enumeration, String, ... and CLASS. All the basic types are manipulated by value and the rest by reference.

Basic Type Argument

Passing mode

Argument can be "**in**" or "**in out**" in CAS.CADE.

Then if some C++ methods use "**in out**" basic type arguments we must create some new Java Class such as Standard.Integer, Standard.Real, Standard.Boolean ... to encapsulate the Java int, double, boolean and to be able to have "**in out**" basic typed arguments. Then in C++ we will have to access to the value filed of this object and change its value.

String

String object will be used as a basic type and mapped to the Java String class.

When you are using a Java String as **in** argument, you must first instantiate a Java a String object and use it as an argument of a native method. Then in C++ you must extract the CString from the Java one by using the **GetStringUTFChars()** method from the **JNIEnv** object and release it before returning to Java by using **ReleaseStringUTFChars() JNIEnv** method.

Example :

```
JNIEXPORT void JNICALL Java_JV3d_Viewer_Create
  (JNIEnv *env, jobject aJV3d_Viewer, jstring atitle)
{
        const char *aName= env->GetStringUTFChars(atitle,0);
        ...
        env->ReleaseStringUTFChars(atitle,aName);
}
```

String as out argument, you must first create a empty Java **StringBuffer** object and use it as an argiment. Then in C++ you must use the **Append()** method of the **StringBuffer** java object to copy the C++ CString in the Java String object.

Example / C++ :

```
JNIEXPORT void JNICALL Java_ProjetSession_LastIO
  (JNIEnv *env, jclass,       jobject aDriverStringBuffer,jobject aNameStringBuffer) {
        jclass StringBufferCLS =
                JavaUtil::FindClass(env, "java.lang.StringBuffer" );
        jmethodID       mid = env->GetMethodID(
                StringBufferCLS, "append","(Ljava/lang/String;)Ljava/lang/StringBuffer;");

        // driver and name are OUT
        Standard_CString driver, name;
        ProjetSession::LastIO(driver, name);

        env->CallObjectMethod(aDriverStringBuffer, mid, env->NewStringUTF(driver));
        env->CallObjectMethod(aNameStringBuffer, mid,   env->NewStringUTF(name));
}
```

Java :

```
        StringBuffer driver = new StringBuffer(); // out
        StringBuffer name   = new StringBuffer(); // out

        prgm.ProjetSession.LastIO(driver,name);
```

Enumeration

Enumeration object don't exist in JAVA. Then we create a new JAVA Class to manipulate this C++ object in JAVA and to insure the same type checking as in C++. All the JAVA enumeration class will inherit JCASCADEEnum object. The value of the enumeration will be manipulated in a short integer in JAVA as a field of JCASCADEEnum object and as a jshort in C++. This java class will provide as public static final member all the predefined C++ enumeration value.

The JAVA list of short can be build from the C++ header file by using some awk command like:
`"nawk -F\, '/Quantity_TOC/{print $1 " " i++}' Quantity_NameOfColor.hxx"`

```
public class TypeOfColor extends JCASCADE.Enum {
  // Quantity_TypeOfColor
  public static final TypeOfColor Quantity_TOC_RGB = new TypeOfColor((short)0);
  public static final TypeOfColor Quantity_TOC_HLS = new TypeOfColor((short)1);
}
```

Basic type

The first argument type is the basic type such as Integer, Real, Boolean. For this kind of argument passed by value we can easily map Java type to C++ type.

Java	JNI	C++
double	jdouble	Double
int	jint	Long
boolean	jboolean	Unsigned char
JCASCADEEnum	jshort	Enumeration

Cas.Cade	C++
Standard_Real	double
Standard_Integer	int
Standard_Boolean	unsigned int

Then we can use this mapping :

Cas.Cade	Java
Standard_Real	double
Standard_Integer	int
Standard_Boolean	boolean
Enumeration	short

CString is specific because Java manipulate string as a specific object. Java use UNICODE for **String** object and must be converted to ASCII before using it in C++ :

```
char * = JNIEnv->GetStringUTFChars(jstring) ;
```

All this basic type can be stored easily in any Java object.

Basic Type Returned value

Basic Type

For the basic type we have to cast the C++ type to a **jint**, **jboolean**, **and jdouble** or **jshort** for an enumeration and return the converted value (C++ -> Java).

String

To return a string from C++ to Java you must create a Java **String** object.

```
jobject ret = JNIEnv->AllocObject(JPack_ClassCLS);
```

CLASS

Prerequisite :

- We must be able to manipulate C++ class from Java as easily as directly from C++.

- We don't have to create additional documentation, then the Java class name is the same as in C++ prefixed by "**J**"
- We must reproduce the same hierarchy of C++ object on the Java side to have the same type checking as in C++
- The Java garbage collector (finalize) must delete the C++ object
- We need to be able to create automatically a Java class from a CDL class definition by reusing the same mechanism as in CCL (interface development unit)
- All the Java classes mirror of C++ class must implement a Create method

Encapsulation :

To be able to manipulate a Cas.Cade C++ class in Java we must find a way to store it as an attribute of a Java object. A way to do it is to store the address of an object in a Java **long**. Then in C++ from a long integer we can cast it to a valid class pointer and then we are able to access to the underlying C++ class.

The Java class name is the name of the C++ class with "J" prefix in order
to use the CAS.CADE C++ API documentation directly from Java.

All the Cas.Cade C++ classes manipulated from JAVA must extends **JCASCADEObject**. This object is in charge to manage a HID which is the address of a Cas.Cade Handle or an address of a C++ Cas.Cade object .

JavaUtil

A set of C++ method had been developed in order to manage this encapsulation. This method are static member of the **JavaUtil** C++ class.

static jclass FindClass(JNIEnv *env, char *className)

From a Java class name this method will return the Java jclass object in order to be able to access some Java class field or to call some Java method.

Example :

```
jclass DoubleCLS = JavaUtil::FindClass( env, "java/lang/Double");
jfieldID JDoubevalueFID = env->GetFieldID(DoubleCLS, "value", "D");

env->SetDoubleField(X, JDoubevalueFID, p.X());
```

static jobject Createjobject (JNIEnv *env,
char * aJavaClass,
Handle(MMgt_TShared) anObject)

From a C++ Cas.Cade Handle and a Java class name this method will create a Java object containing a pointer to the C++ class.

Example :

```
JNIEXPORT jobject JNICALL Java_JCPPApp_CreateView
 (JNIEnv *env, jclass, jobject aJAIS_InteractiveContext)
```

```
{ Handle(AIS_InteractiveContext) myAISContext =
 Handle(AIS_InteractiveContext)::DownCast(
             JavaUtil::getTransient(env, aJAIS_InteractiveContext));

  Handle(V3d_View) aView =
             myAISContext->CurrentViewer()->CreateView();

   return JavaUtil::Createjobject(env, "JV3d/View", aView);
}
```

static jobject Createjobject (JNIEnv *env, char * aJavaClass, Standard_Address anObject)

From an address of a C++ Cas.Cade class and a Java class name this method will create a Java object containing a pointer to the C++ class.

Example :

```
  Quantity_Color *pcol = new Quantity_Color(...);

  return JavaUtil::CreateObject(env, "JQuantity/Color", pcol);
```

static Handle(Standard_Transient) getTransient(JNIEnv *env, jobject aCASCASDEObject)

From a JCASCADE Java object containing a Handle to a Cas.Cade C++ class this method will return the Handle to this C++ class

Example :

```
JNIEXPORT void JNICALL Java_JV3d_View_StartRotation
  (JNIEnv *env, jobject aV3d_View,jint x, jint y)
{ Handle(V3d_View) myView =
       Handle(V3d_View)::DownCast(JavaUtil::getTransient(env, aV3d_View));
 ...
}
```

static Standard_Address getAddress(JNIEnv *env, jobject aCASCASDEObject);

From a JCASCADE Java object containing a Handle to a Cas.Cade C++ class this method will return the Handle to this C++ class

Example :

```
JNIEXPORT jshort JNICALL Java_JQuantity_Color_Name
  (JNIEnv *env, jobject aJQuantity_Color)
{ Quantity_Color *pcol =
       (Quantity_Color *) JavaUtil::getAddress(env, aJQuantity_Color);

  ...
}
```

Class Creation

All the Java object mirror of Cas.Cade C++ classes must implement a Java native methods called **Create**. The Create C++ method must returns the HID of the created Cas.Cade Engine_Handle object. Then the Java object constructor is in charge to call this native Create method and update the private HID fields from the returned value.

Java

```
package JQuantity;

public class Color extends JCASCADE.Object {
     public Color( JQuantity.NameOfColor nc )   {
           Create( nc.getValue() );
}
```

C++

```
JNIEXPORT void JNICALL Java_JQuantity_Color_Create__S
  (JNIEnv *env, jobject ajcolor, jshort aNaneOfColor)
 {
      Quantity_NameOfColor name = (Quantity_NameOfColor)aNaneOfColor;

      Quantity_Color *pcol = new Quantity_Color(name);

      JavaUtil::setJavaHID(env, ajcolor, pcol);
}
```

Class destruction

The Java finalize of any **JCASCADEObject** must calls the Finalize native method to delete the the C++ object or to delete the Handle to the C++ object.

Then by inheritance the memory management of the underlay Cas.Cade C++ object is done.

The JCASCADEObject from the Java side :

```
package JTopoDS;

public class Shape extends JCASCADE.Object {

      native synchronized void Finalize();

      protected void finalize() throws Throwable {
            Finalize();
      }
}
```

The JCASCADEObject from the C++ side :

```
JNIEXPORT void JNICALL Java_JTopoDS_Shape_Finalize
(JNIEnv *env, jobject ajobj )
{
  jlong HID = JavaUtil::getHID(env, ajobj);

  TopoDS_Shape *s = ( TopoDS_Shape * ) HID;

  delete s;

  return;
}
```

Class Argument

Passing mode

C++ Class argument can be "**in**" or "**in out**" in CAS.CADE and are always manipulated by reference. In Java all the classes are manipulated by reference and as "**in out**".

Then the Cas.Cade C++ class management can be done by using only the Java language specification.

To get the C++ class from the Java argument, you must use the JavaUtil static method to get the C++ pointer to the C++ object from the Java argument.

Handled class Example:

```
JNIEXPORT void JNICALL Java_JAIS_InteractiveContext_Redisplay
  (JNIEnv *env,
  jobject aJAIS_InteractiveContext,
  jobject aJAIS_InteractiveObject)
{

  Handle(AIS_InteractiveContext) myAISContext =
      Handle(AIS_ InteractiveContext)::DownCast(
            JavaUtil::getTransient(env, aJAIS_ InteractiveContext));

  Handle(AIS_InteractiveObject) aIobj =
      Handle(AIS_InteractiveObject)::DownCast(
            JavaUtil::getTransient(env, aJAIS_InteractiveObject));

  myAISContext->Redisplay(aIobj);
}
```

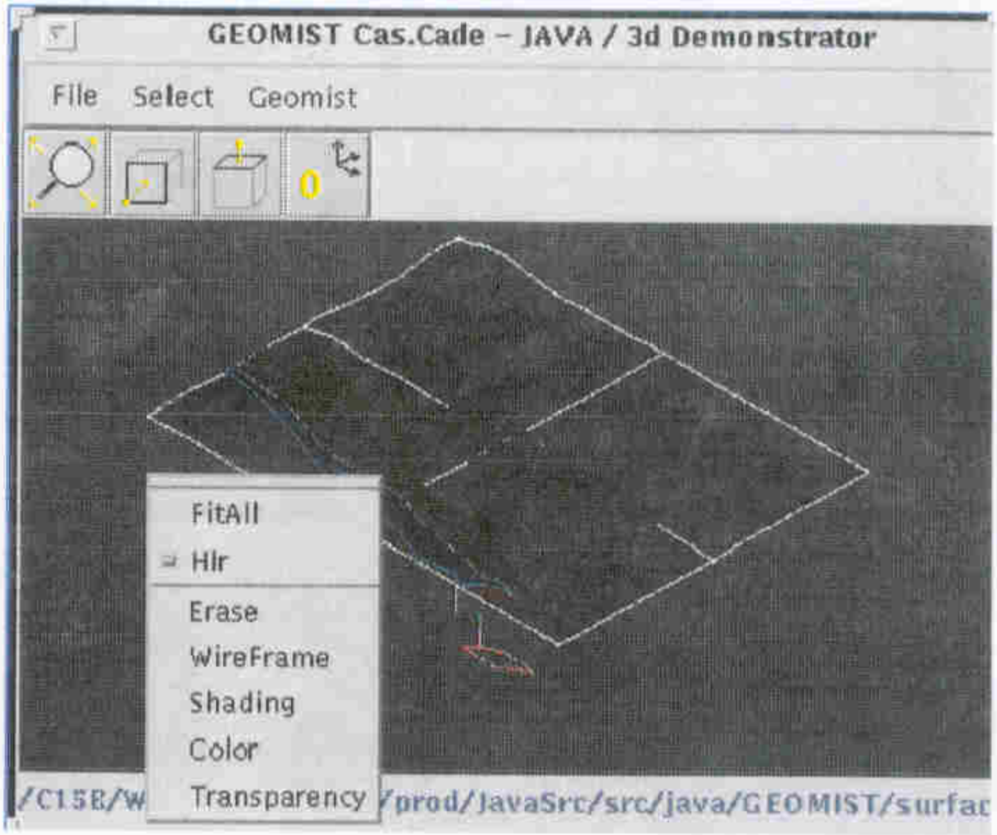
Class manipulated by value example:

```
JNIEXPORT jfloat JNICALL Java_JQuantity_Color_Red
  (JNIEnv *env , jobject aJQuantity_Color)
{
  jfloat ret = 0.;

  Quantity_Color *pcol = (
      Quantity_Color *) JavaUtil::getAddress(env, aJQuantity_Color);

  ret = (jfloat) pcol->Red();

  return ret;
}
```

6. 3D demonstrator User guide

The 3d viewer

Rotation, zoom and panning is done by using the keyboard control key and the mouse button. Rotation is done by moving the mouse over the view with the right mousse button and the shift key down. Panning is done by using the middle mouse button, Zoom by using the left mouse button.


GEOMIST Cas.Cade – JAVA / 3d Demonstrator
File
Select
Geomist
0
FitAll
Hlr
Erase
WireFrame
Shading
Color
Transparency
/C15B/W
/prod/JavaSrc/src/java/GEOMIST/surfac

To select an object you must pick on it with the left mousse button. When you move the mouse over an object, this object is prehighlighted in blue. It will become white after selection with the left mouse button. You can select multiple object by picking the objects with the Shift key and the left mouse button.

3d viewer popup menu

This popup menu is made from 7 item :

- FitAll is used to compute the best zoom in order to see all the displayed object
- Hlr is used to compute hidden line removal for solid object
- Erase is used to erase the selected object
- Wireframe is used to display the selected oject in Wireframe mode
- Shading is used to display the selected oject in Shading mode

- Color will change the current color of the selected objects by using a standard color chooser
- Transparency can change the transparency of the current object, to see the transparency effect the object must be displayed in shading mode. The transparency range is from 0, opaque to 100, fully tranparent.

The Erase, Wireframe, Shading, Color, Transparency item are only activated if an object is selected.

3d viewer icon

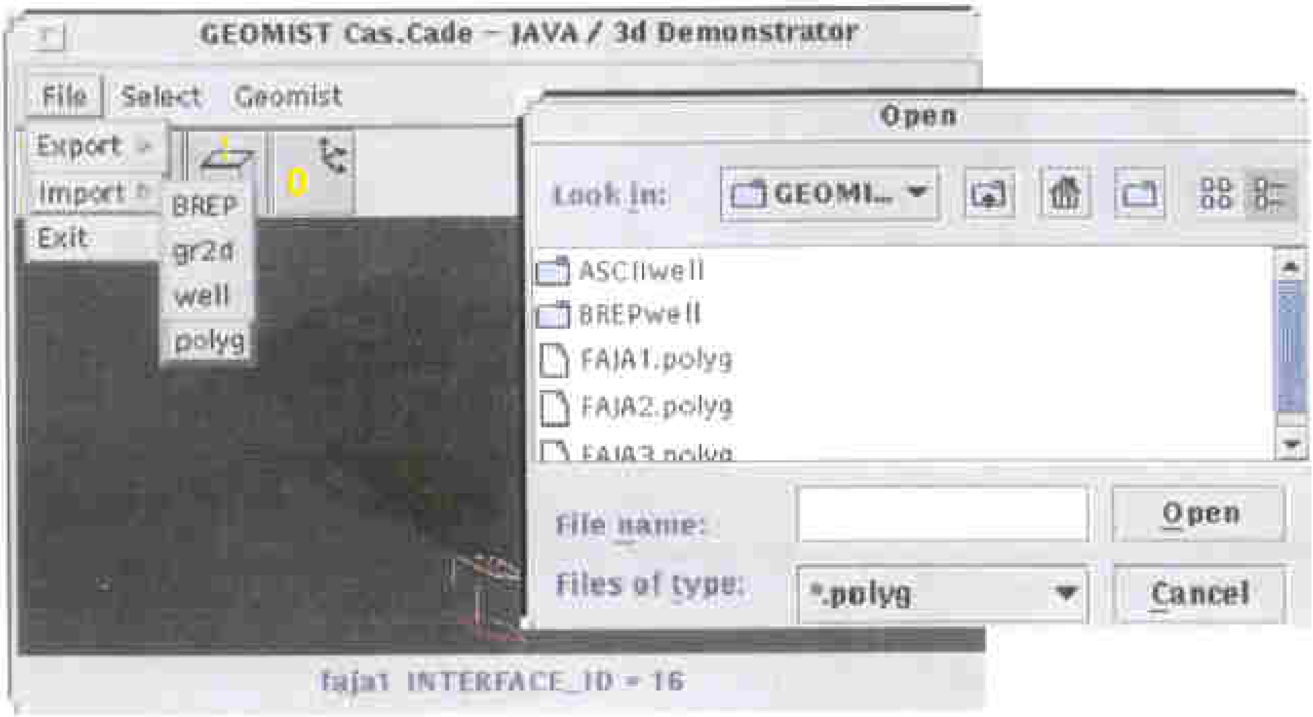
The icon is used to compute the best zoom in order to fit all the displayed objects.

The icon is used to switch to a Front view.

The icon is used to swith to a top view.

The icon is used to reset the view orientation to a (X,-Y,Z) orientation.

File / Import menu



ASCIIwell
BREPwell
FAJA1.polyg
FAJA2.polyg
FAJA3.polyg

File name:

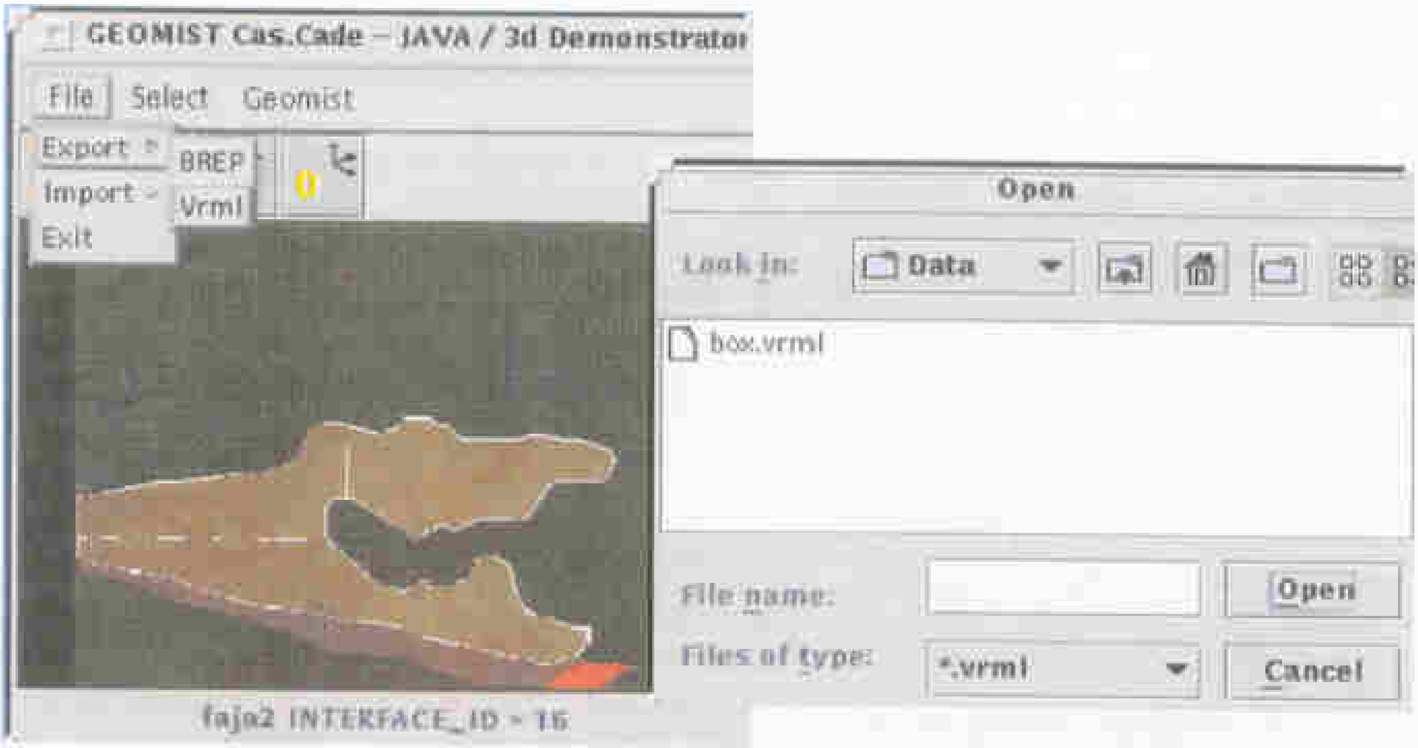
Files of type: *.polyg

fajal INTERFACE_ID = 16

Under File/Import menu you will find 3 submenu, "BREP", "gr2d", "well" and "polyg". This menu are used to load object in the 3d viewer. "BREP" is used to load any Cas Cade shape, the brep file is the native Cas Cade file format for geometry and topology. "gr2d", "well" and "polyg" menu are used to load Geological object stored in a OMEGA ASCCI file format. Then you can load and visualize OMEGA surface, polygons and well with their own geological attributes. All this submenu will display a standard file selection box in order to select the file with the specific extention.

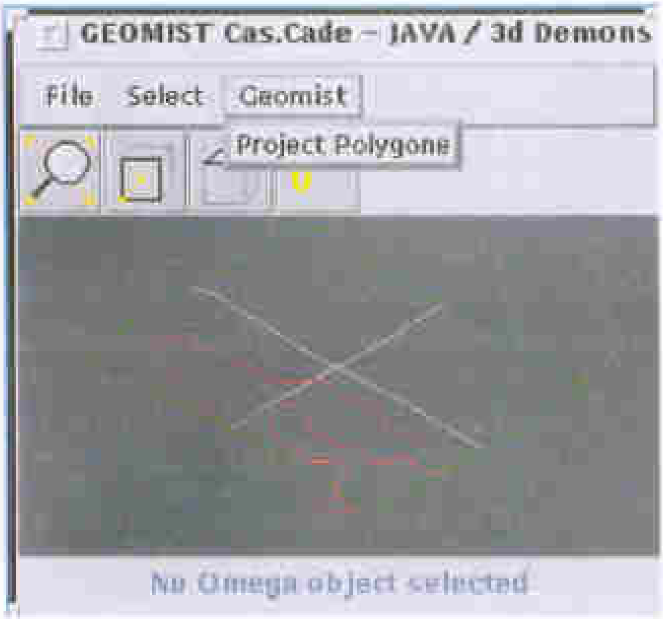
File / Export menu

Under File/Export menu you will find 2 submenu, "BREP" and "VRML". This menu are used to save the selected object in the 3d viewer. "BREP" is used to save an object in a brep file, this is the native Cas Cade file format for geometry and topology. "VRML" menu is used to save the selected object in the 3d Viewer in VRML format in order to export it in a VRML viewer and then to enable to visualize the geological object over the WEB. All this submenu will display a standard file selection box in order to select the file with the specific extension to save the selected object.

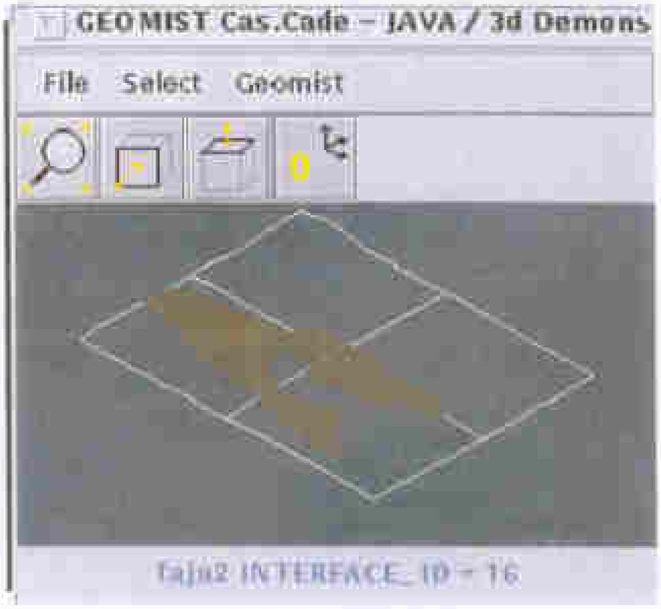


Geomist / Project Polygone menu

In order to use this menu, you must first load a gr2d object and then a polyg file. Then a dialog box will ask you to select the gr2d surface, then another one to ask to select the polyg object in the viewer. Some echo is done to show the computation and a third dialog box will confirm the success of the computation. The result is automaticaly displayed in the 3d viewer.



No Omega object selected



Taju2 INTERFACE_ID = 16

ANNEX 4

SOURCE CODE FOR METADATA APPLICATIONS

- METADATA CREATION
- METADATA INSERTION AND UPDATE
- METABASE QUERY

SOURCE CODE FOR APPLICATIONS

There are three distinct packages of software to interact to the metadata base, which range from the creation of each object to the access of the metadata inserted. The independence of these interaction levels allows the developer the choice of the best platform for implementation.

1. CREATING METABASE

Being a database itself, the metabase is designed and implemented over the Oracle DBMS. The definition of data model is written in language Data Definition Language (DDL), extending SQL*plus.

```
----------------------------------------------
-- Name:        ct_metabase.sql
--
-- Author:      Jorge Lopes <jlopes@igm.pt>
-- Creation:    02/12/98
-- Update:      04/06/99
--
-- Procedures to create geomist metabase
----------------------------------------------

create table category2(
        cat_c    varchar2(2) not null,
        lang_c   varchar2(2) not null,
        category varchar2(30) not null,
        date_entered        date not null,
        date_updated        date,
        primary key (cat_c))
        tablespace geixsdata;

create table keywords(
        keyword_c       varchar2(4) not null,
        lang_c          varchar2(2) not null,
        keyword         varchar2(30) not null,
        date_entered    date not null,
        date_updated    date,
        primary key (keyword_c))
        tablespace geixsdata;

create table dataset_meta(
        ts#             number not null,
        title           varchar2(30) not null,
        abstract varchar2(2000),
        purpose         varchar2(256),
        usage           varchar2(80),
        spat_sub_schema varchar2(30),
        spat_ref_type   varchar2(30),
        language        varchar2(80),
        reference       varchar2(80),
        related_dset    varchar2(60),
        ind_ref_system  varchar2(60),
        source          varchar2(60),
        type_ind_ref_s  varchar2(60),
        datum           varchar2(30),
        ellipsoide      varchar2(30),
        projection      varchar2(30),
        extent_date     varchar2(30),
        bounding_area   varchar2(80),
        geographic_area varchar2(80),
        name_areal_unit varchar2(80),
        vertical_extent varchar2(80),
        temporal_extent varchar2(80),
        from_date       varchar2(80),
        to_date         varchar2(80),
        organisation    varchar2(64),
        abbreviate      varchar2(10),
        address         varchar2(120),
        role            varchar2(80),
        function varchar2(80),
        contact         varchar2(80),
        contact_adress  varchar2(120),
        contact_role    varchar2(80),
        restrict_on_use varchar2(80),
        copywrite_owner varchar2(80),
        price           varchar2(40),
        data_media      varchar2(60),
        formats         varchar2(60),
        on_line_access  varchar2(60),
        a_order         varchar2(60),
        support_service varchar2(80),
        entry_date      date,
        last_check      date,
        last_update     date,
        met_language    varchar2(40),
        primary key (ts#))
        tablespace geixsdata;

create table dataset_category(
        ts#             number not null,
        cat_c           varchar2(2) not null,
        primary key (ts#,cat_c),
        Constraint FK_dataset_category_ts Foreign Key (ts#) references dataset_meta(ts#),
        Constraint FK_dataset_category_cat Foreign Key (cat_c) references category(cat_c))
        tablespace geixsdata;

create table dataset_keyword(
        ts#             number not null,
        keyword_c       varchar2(4) not null,
        primary key (ts#,keyword_c),
        Constraint FK_dataset_keyword_ts Foreign Key (ts#) references dataset_meta(ts#),
        Constraint     FK_dataset_keyword_key     Foreign     Key     (keyword_c)     references
keywords(keyword_c))
        tablespace geixsdata;

create table dataset_detail(
        ts#             number not null,
        keyword_c       varchar2(4) not null,
        primary key (ts#,keyword_c),
        Constraint FK_dataset_detail_ts Foreign Key (ts#) references dataset_meta(ts#),
        Constraint FK_dataset_detail_key Foreign Key (keyword_c) references keywords(keyword_c))
        tablespace geixsdata;

create table dataset_subject(
        ts#             number not null,
        keyword_c       varchar2(4) not null,
primary key (ts#,keyword_c),
Constraint FK_dataset_subject_ts Foreign Key (ts#) references dataset_meta(ts#),
Constraint FK_dataset_subject_key Foreign Key (keyword_c) references keywords(keyword_c))
tablespace geixsdata;
```

2. METADATA INSERTION AND UPDATE

To insert and update data into the metabase there are two distinct options. The static metadata is handled directly through the database server, while the structural metadata is done through an interface application implemented over ArcView GIS.

```
'---------------
'TabMeta.load
'---------------
av.GetProject.FindDialog("TabMeta").Open

'---------------
' tabMeta.open
'--------------------
mVT=av.GetProject.FindDoc("tableSpaces")
if (mVT<>Nil) then
  av.GetProject.RemoveDoc(mVT)
end

theSQL=SQLCon.Find("geomist")
theSQL.Login("SRVR=atlas;UID=jlopes;PWD=jlopes")
astring="Select ts#, name from sys.ts$ where ts#>3 and online$=1"
theVTab=VTab.MakeSQL(theSQL, astring)
myTable=Table.Make(theVtab)
av.GetProject.AddDoc(mytable)
mytable.setname("tableSpaces")

tsCombo=self.FindByName("tsCombo")
tsCombo.empty
mVT=av.GetProject.findDoc("tableSpaces").GetVTab
myFields = mVT.FindField("name")
tsCombo.DefineFromVTab(mVT, myFields, false)
_dicTS=Dictionary.make(20)
myFields2=mVT.FindField("ts#")
for each i in mVT
  _dicTS.Add(mVT.ReturnValue(myfields,i),mVT.ReturnValue(myfields2,i))
end

self.FindByName("tabCombo").empty
self.FindByName("attCombo").empty
self.FindByName("tabMetaCom").setText("")


'------------------
' tabMeta.att.select
'------------------
diag=av.getProject.FindDoc("tabMeta").GetDialog
mVT=av.GetProject.findDoc("attMeta.table").GetVTab
myFields = mVT.FindField("name")
_dicAtt=Dictionary.Make(100)
fl=mVT.getFields
for each i in mVT
  mList=List.Make
```

```
 for each k in 0..(fl.Count-1)
  if (k=5) then
    if (mVT.returnValue(fl.get(4),i)="Char") then
        mList.Add(mVT.returnValue(fl.get(5),i).truncate)
    else
         mList.Add(mVT.returnValue(fl.get(6),i).truncate)
    end
    k=k+1
  else
    mList.Add(mVT.returnValue(fl.get(k),i))
  end
 end
 _dicAtt.Add(mVT.returnValue(fl.get(2),i),mList)
end

mAtt=diag.FindByName("AttCombo").GetCurrentValue
mObj=diag.FindByName("tabCod").GetText
mCol=_dicAtt.Get(mAtt).Get(1)
'msgbox.info(mAtt,"Ola")
diag.FindByName("attType").setText(_dicAtt.Get(mAtt).Get(4))
diag.FindByName("attWidth").setText(_dicAtt.Get(mAtt).Get(5).AsString)
diag.FindByName("attDec").setText(_dicAtt.Get(mAtt).Get(7).AsString)
diag.FindByName("attCod").setText(_dicAtt.Get(mAtt).Get(1).truncate.AsString)

'Look for Attribute comments and display-it
mVTAtt=av.GetProject.findDoc("attCom.table").GetVTab
theBitMap=mVTAtt.GetSelection
myString="(([obj#]="+mObj+") and ([col#]="+mCol.AsString+"))"
'msgbox.info(myString,"Strinng")
mVTAtt.Query(myString,theBitMap,#VTAB_SELTYPE_NEW)
mVTAtt.UpdateSelection
if (mVTatt.GetSelection.Count=1) then
 for each i in mVTatt.GetSelection
   mCom=mVTatt.ReturnValue(mVTatt.FindField("comment$"),i)
  end
  diag.FindByName("attMetaCom").setText(mCom)
  diag.FindByName("attCom_flag").setText(1.AsString)
else
 diag.FindByName("attMetaCom").setText("")
 diag.FindByName("attCom_flag").setText(0.AsString)
end

'-------------------
' tabMeta.att.update
'-------------------
diag=av.getProject.FindDoc("tabMeta").GetDialog
curObj=diag.FindByName("tabCod").GetText.AsNumber
curCol=diag.FindByName("attCod").GetText.AsNumber

com=self.GetDialog.FindByName("attMetaCom").GetText
flag=self.GetDialog.FindByName("attCom_flag").GetText.AsNumber
theSQL=SQLCon.Find("geomist")
theSQL.Login("SRVR=atlas;UID=jlopes;PWD=jlopes")
if (com.trim<>"") then
  if (flag=0) then
    ' First Attribute comment
   astring="insert into sys.com$ (obj#,col#,comment$)
values("+curObj.asString+","+curCol.asString+",'"+com.trim+"')"
   theSQL.ExecuteSQL(astring)
   self.GetDialog.FindByName("attCom_flag").SetText("1")
 else
  'update attribute comment
  astring="update sys.com$ set comment$='"+com+"' where obj#="+curObj.AsString +" and
col#="+curCol.AsString
  theSQL.ExecuteSQL(astring)
 end
end

'------------------
' tabMeta.tab.select
'------------------
mVT=av.GetProject.FindDoc("attMeta.table")
if (mVT<>Nil) then
  av.GetProject.RemoveDoc(mVT)
end
mVT=av.GetProject.FindDoc("attCom.table")
if (mVT<>Nil) then
  av.GetProject.RemoveDoc(mVT)
end


diag=av.getProject.FindDoc("tabMeta").GetDialog
codCom=diag.FindByName("tabCombo").GetCurrentValue
tabComment=_dicTab.Get(codCom).Get(1)

tabMetaCom=diag.FindByName("tabMetaCom")
tabMetaCom.setText(tabComment)
diag.FindByName("tabCod").SetText(_dicTab.Get(codCom).Get(0).truncate.AsString)

theSQL=SQLCon.Find("geomist")
theSQL.Login("SRVR=atlas;UID=jlopes;PWD=jlopes")
astring="select o.obj#, o.col#, o.name, o.type#,"
astring=astring+"decode (o.type#, 1 ,'Char', 2, 'Number',12,'Date'),"
astring=astring+"o.length, o.precision, o.scale"
astring=astring+" from sys.col$ o"
astring=astring+" where o.obj#="+_dicTab.Get(codCom).Get(0).AsString
'astring=astring+" and c.col#(+)=o.col#"
astring=astring+" order by o.col#"

'msgbox.report(astring,"ola")
theVTab=VTab.MakeSQL(theSQL, astring)
myTable=Table.Make(theVtab)
av.GetProject.AddDoc(mytable)
mytable.setname("attMeta.table")

mVT=av.GetProject.findDoc("attMeta.table").GetVTab
myFields = mVT.FindField("name")
attCombo=av.GetProject.FindDoc("tabMeta").GetDialog.FindByName("attCombo")
attCombo.DefineFromVTab(mVT, myFields, false)
_dicAtt=Dictionary.Make(100)
fl=mVT.getFields
for each i in mVT
 mList=List.Make
 for each k in 0..(fl.Count-1)
  if (k=5) then
   if (mVT.returnValue(fl.get(4),i)="Char") then
      mList.Add(mVT.returnValue(fl.get(5),i).truncate)
   else
      mList.Add(mVT.returnValue(fl.get(6),i).truncate)
   end
```

```
    k=k+1
   else
    mList.Add(mVT.returnValue(fl.get(k),i))
   end
  end
  _dicAtt.Add(mVT.returnValue(fl.get(2),i),mList)
end

'msgbox.info(_dicAtt.Get(mVT.returnValue(fl.get(2),0)).Get(4),"")
mAtt=_dicAtt.Get(mVT.returnValue(fl.get(2),0)).Get(0)
'msgbox.info(mAtt.AsString,"")
diag.FindByName("attType").setText(_dicAtt.Get(mVT.returnValue(fl.get(2),0)).Get(4))
diag.FindByName("attWidth").setText(_dicAtt.Get(mVT.returnValue(fl.get(2),0)).Get(5).AsString)
diag.FindByName("attDec").setText(_dicAtt.Get(mVT.returnValue(fl.get(2),0)).Get(7).AsString)
diag.FindByName("attCod").setText(_dicAtt.Get(mVT.returnValue(fl.get(2),0)).Get(1).truncate.AsString)

'Now we select all comments about this table object (tables and attributes)
' mAtt is the current Object (table)
astring="select * from sys.com$ where obj#="+mAtt.asString++"order by col#"
'msgbox.info(astring,"")
theVTab=VTab.MakeSQL(theSQL, astring)
myTable=Table.Make(theVtab)
av.GetProject.AddDoc(mytable)
mytable.setname("attCom.table")
_dicAttCom=Dictionary.Make(100)
fl=theVTab.GetFields
for each i in theVTab
  cObj=theVTab.ReturnValue(fl.Get(0),i)
  cCol=theVTab.ReturnValue(fl.Get(1),i)
  cCom=theVTab.ReturnValue(fl.Get(2),i)
  if (cCol<>Nil) then
    _dicAttCom.Add({cObj,cCol},cCom)
  end
end


'Now lets
mObj={mAtt,1}
if (_dicAttCom.Get(mObj)<>Nil) then
  diag.FindByName("attMetaCom").setText(_dicAtt.Get(mObj))
  diag.FindByName("attCom_flag").setText(1.AsString)
else
  diag.FindByName("attCom_flag").setText(0.asString)
end


'Fill attribute comments
mObj=diag.FindByName("tabCod").GetText.AsNumber
mCol=diag.FindByName("attCod").GetText.AsNumber

'Look for Attribute comments and display it
mVTAtt=av.GetProject.findDoc("attCom.table").GetVTab
theBitMap=mVTAtt.GetSelection
myString="(([obj#]="+mObj.AsString+") and ([col#]="+mCol.AsString+"))"
mVTAtt.Query(myString,theBitMap,#VTAB_SELTYPE_NEW)
mVTAtt.UpdateSelection
if (mVTatt.GetSelection.Count=1) then
 for each i in mVTatt.GetSelection
   mCom=mVTatt.ReturnValue(mVTatt.FindField("comment$"),i)
 end
```

```
  diag.FindByName("attMetaCom").setText(mCom)
  diag.FindByName("attCom_flag").setText(1.AsString)
 else
  diag.FindByName("attMetaCom").setText("")
  diag.FindByName("attCom_flag").setText(0.AsString)
 end

'Look for the Number of Records
astring="select count(*) from "+codCom
'msgbox.info(astring,"")
theVTab=VTab.MakeSQL(theSQL, astring)
'myTable=Table.Make(theVtab)
'av.GetProject.AddDoc(mytable)
n=theVTab.ReturnValue(theVTab.FindField("count(*)"),0)
diag.FindByName("numRecords").setText(n.truncate.AsString)


'------------------
' tabMeta.tab.update
'------------------
tabComment=_dicTab.Get(self.GetDialog.FindByName("tabCombo").GetCurrentValue).Get(1)
curCod=_dicTab.Get(self.GetDialog.FindByName("tabCombo").GetCurrentValue).Get(0)

com=self.GetDialog.FindByName("tabMetaCom")
theSQL=SQLCon.Find("geomist")
theSQL.Login("SRVR=atlas;UID=jlopes;PWD=jlopes")
if (tabComment.trim="") then
  ' There is no metadata assigned this table
  astring="insert into sys.com$ (obj#,comment$) values("+curCod.asString+",'"+com.getText+"')"
  theSQL.ExecuteSQL(astring)
else
  astring="update sys.com$ set comment$='"+com.getText+"' where obj#="+curCod.AsString
  'msgbox.info(astring,"OBJ="++curCod.truncate.AsString)
  theSQL.ExecuteSQL(astring)
  'msgbox.info(astring,"OBJ="++curCod.truncate.AsString)
end

'if (com.getText.trim="") then
'  msgbox.info("free","OBJ="++curCod.truncate.AsString)
'else
'  msgbox.info(com.getText,"OBJ="++curCod.truncate.AsString)
'end


'------------------
' tabMeta.ts.select
'------------------
mVT=av.GetProject.FindDoc("tabMeta.table")
if (mVT<>Nil) then
  av.GetProject.RemoveDoc(mVT)
end

'msgbox.info(self.GetCurrentValue,_dicTS.Get(self.GetCurrentValue).AsString)

tsCode=_dicTS.Get(self.GetCurrentValue)

theSQL=SQLCon.Find("geomist")
theSQL.Login("SRVR=atlas;UID=jlopes;PWD=jlopes")
astring="select o.name, o.obj#, c.comment$ from sys.obj$ o, sys.com$ c, sys.tab$ t "
astring=astring+"where o.type=2 and o.owner#=10 and o.obj#=c.obj#(+) and c.col#(+) is null "
```

```
astring=astring+" and t.obj#=o.obj# and t.ts#="+tsCode.AsString
'msgbox.report(astring,"ola")
theVTab=VTab.MakeSQL(theSQL, astring)
myTable=Table.Make(theVtab)
av.GetProject.AddDoc(mytable)
mytable.setname("tabMeta.table")

mVT=av.GetProject.findDoc("tabMeta.table").GetVTab
myFields = mVT.FindField("name")
tabCombo=self.GetDialog.FindByName("tabCombo")
tabCombo.DefineFromVTab(mVT, myFields, false)
_dicTab=Dictionary.make(20)
myFields2=mVT.FindField("comment$")
mf3=mVT.FindField("obj#")
for each i in mVT
  _dicTab.Add(mVT.ReturnValue(myfields,i),{mVT.ReturnValue(mf3,i),mVT.ReturnValue(myfields2,i)})
end

com=self.GetDialog.FindByName("tabMetaCom")
'com.setText("")

self.GetDialog.FindByName("tabCod").SetText(_dicTab.Get(tabCombo.GetCurrentValue).Get(0).truncat
e.AsString)

tabComment=_dicTab.Get(tabCombo.GetCurrentValue).Get(1)

tabMetaCom=self.GetDialog.FindByName("tabMetaCom")
tabMetaCom.setText(tabComment)

av.run("tabMeta.tab.select",{})


'--------------
' tabMeta.done
'--------------
self.GetDialog.Close
```

3. METABASE QUERY

The metabase query tool is a Geomist service, providing inquiry capabilities to the metabase through the World Wide Web. The interaction available is restricted to the selection of a dataset.

The results are formatted into a pre-defined model, including static and structural data about the selected dataset.

To reach these goals there is a package of code, distributed at distinct platforms, but co-operating ,such as the Web server and the Database server.

```
' Name:    java.QueryMetabase
'
' Author:  Jorge Lopes <jlopes@igm.pt>
' Creation: 15/12/98
' Update:  21/07/99
'
'
theSQL=SQLCon.Find("Oracle")
```

```
theSQL.Login("jlopes/jlopes@geomist")
astring="Select ts#, name from sys.ts$ where ts#>3 and online$=1"
mVT=VTab.MakeSQL(theSQL, astring)

wl = Weblink.the
wl.WriteResponseHeader("Content-type: text/html"+CR+NL+CR+NL)
wl.WriteString("<HTML>"++CR+NL)
wl.WriteString("<BODY bgcolor=""#00ffff"">"++CR+NL)
wl.WriteString("<CENTER>"++CR+NL)
wl.WriteString("<TABLE BORDER=0 align=center>"++CR+NL)
wl.WriteString("<TR><TD align=center><b>Select your dataset</b>"++CR+NL)
wl.WriteString("<FORM>"++CR+NL)
wl.WriteString("<TR><TD ALIGN=CENTER> <Select>"++CR+NL)
myField = mVT.FindField("name")
for each i in mVT
 mVal=mVT.ReturnValue(myfield,i)
 wl.WriteString("<option value="""+mVal+""">"+mVal++CR+NL)
end
wl.WriteString("</select><p>"++CR+NL)
wl.WriteString("<TR><TD ALIGN=CENTER> <input type=""button"" value=""Show"""++CR+NL)
wl.WriteString("  onClick=""{mObj=top.frames[1].document.forms[0].elements[0];"++CR+NL)
wl.WriteString("          mVal=mObj.options[mObj.selectedIndex].value;"++CR+NL)
wl.WriteString("
top.frames[0].location='/.esrimap?nameX=GeomistMap&cmd=java.ShowMetabase&theMetabase='+m
Val}"">"++CR+NL)

wl.WriteString("</form>"++CR+NL)
wl.WriteString("</table>"++CR+NL)
wl.WriteString("</html>"++CR+NL)



wl.WriteString("<pre>")
wl.WriteString(" "++CR+NL)
wl.WriteString(" "++CR+NL)
wl.WriteString(" "++CR+NL)
wl.WriteString(" "++CR+NL)
wl.WriteString(" "++CR+NL)
'wl.WriteString("<a href=""mailto:jlopes@igm.pt""> Suggestions</a>")
wl.WriteString("</pre>")



'Name:    java.ShowMetabase
'
'Author:  Jorge Lopes  <jlopes@igm.pt>
'Creation: 15/12/98
'Update:   21/07/99
'
'
'IN: (dset : String)
'

'Geixs flag (1-true, 0-false)
geixs=1

theSQL=SQLCon.Find("Oracle")
```

```
theSQL.Login("jlopes/jlopes@geomist")
astring="Select ts#, name from sys.ts$ where ts#>3 and online$=1"
mVT=VTab.MakeSQL(theSQL, astring)

myBase=self.Get(0).Get(1)
theBitmap=mVT.GetSelection
myString="([name].trim="""+myBase.trim+""")"
mVT.Query(myString,theBitmap,#VTAB_SELTYPE_NEW)
mVT.UpdateSelection
for each i in mVT.GetSelection
  tsCod=mVT.ReturnValue(mVT.FindField("ts#"),i).truncate
end


wl = Weblink.the
wl.WriteResponseHeader("Content-type: text/html"+CR+NL+CR+NL)
wl.WriteString("<HTML>"++CR+NL)
wl.WriteString("<!--This page was generated dynamically>"++CR+NL)
wl.WriteString("<!--JLopes(Jan99)>"++CR+NL)

wl.WriteString("<BODY bgcolor=""#ffffff"">"++CR+NL)
wl.WriteString("<CENTER>"++CR+NL)

'Writes Database metadata
  wl.WriteString("<font size=5> Dataset "++myBase++"</font>"++CR+NL)
  wl.WriteString("<hr width=80%><p>"++CR+NL)
  wl.WriteString("</CENTER>"++CR+NL)
  wl.WriteString("<b>DATASET METADATA</b>"++CR+NL)
  wl.WriteString("<table border=1>"++CR+NL)
  wl.writeString("<tr><td                                align=center><b>Metadata</b><td
align=center><b>Description"++CR+NL)
  astring="select * from dataset_metadata where ts#="+tsCod.AsString
  rVTab=VTab.MakeSQL(theSQL, astring)
  wl.writeString("<tr><td><b>Dataset identification</b>"++CR+NL)
  wl.writeString("<tr><td>Dataset
tittle<td>"+rVTab.ReturnValue(rVTab.findField("name"),0)++CR+NL)
  wl.writeString("<tr><td><b>Dataset overview</b>"++CR+NL)

'wl.writeString("<tr><td>Abstract<td>"+rVTab.ReturnValue(rVTab.findField("abstract"),0)++CR+NL
)
  'Get keywords associated

  str=""
  if(tsCod>5) then
    'Get dataset category
    astring="select c.category from category c, dataset_category dsc where c.cat_c=dsc.cat_c and
dsc.ts#="+tsCod.AsString
    kVT=VTab.MakeSQL(theSQL, astring)
    kc= "Category : "+kVT.ReturnValue(kVT.GetFields.get(0),0).lcase

    'Get dataset Subject
    astring="select k.keyword from keywords k, dataset_subject dsk where  k.keyword_c=dsk.keyword_c
 and dsk.ts#="+tsCod.AsString
    kVT=VTab.MakeSQL(theSQL, astring)
    kc= kc+" ; "+kVT.ReturnValue(kVT.GetFields.get(0),0).lcase

    'Get dataset Subject Detail
    astring="select k.keyword from keywords k, dataset_detail dsk where  k.keyword_c=dsk.keyword_c
 and dsk.ts#="+tsCod.AsString
    kVT=VTab.MakeSQL(theSQL, astring)
  kw= "Details : "+kVT.ReturnValue(kVT.GetFields.get(0),0).lcase +" ; "

  astring="select k.keyword from keywords k, dataset_keyword dsk where  k.keyword_c=dsk.keyword_c
and dsk.ts#="+tsCod.AsString
  kVT=VTab.MakeSQL(theSQL, astring)
  for each i in kVT
    kw=kw+kVT.ReturnValue(kVT.GetFields.get(0),i).lcase+" ; "
  end
  kw=kw.left(kw.count-2)
  str="<p>"+kc+"<br>"+kw
 end
 mAbs=""
 i=0
 while (true)
  abstring="select substr(abstract,"+(i*250+1).asString+",250) from dataset_metadata where
ts#="+tsCod.AsString
  absVT=VTab.MakeSQL(theSQL, abstring)
  aux=(absVT.ReturnValue(absVT.getfields.get(0),0)).trim
  if((i>15) or (aux.Count=0)) then
    break
  end
  i=i+1
  mAbs=mAbs+aux
 end
 wl.writeString("<tr><td>Abstract<td>"+mAbs++CR+NL)
 wl.writeString(str++CR+NL)
 wl.writeString("<tr><td>Purpose of production<td>"++CR+NL)
 wl.writeString("<tr><td>Usage<td>"++CR+NL)
 wl.writeString("<tr><td>Spatial sub-scheme type<td>"++CR+NL)
 wl.writeString("<tr><td>Spatial reference system
type<td>"+rVTab.ReturnValue(rVTab.findField("spat_ref_type"),0)++CR+NL)
 wl.writeString("<tr><td>Dataset
language<td>"+rVTab.ReturnValue(rVTab.findField("language"),0)++CR+NL)
 wl.writeString("<tr><td>Dataset reference<td>"++CR+NL)
 wl.writeString("<tr><td>Related
datasets<td>"+rVTab.ReturnValue(rVTab.findField("related_dset"),0)++CR+NL)

 wl.writeString("<tr><td><b>Dataset quality elements</b>"++CR+NL)
 wl.writeString("<tr><td>Source<td>"++CR+NL)

 wl.writeString("<tr><td><b>Spatial reference system</b>"++CR+NL)
 wl.writeString("<tr><td><em>Indirect spatial reference system</em>"++CR+NL)
 wl.writeString("<tr><td>Type of indirect spatial reference
system<td>"+rVTab.ReturnValue(rVTab.findField("ind_ref_system"),0)++CR+NL)
 wl.writeString("<tr><td><em>Direct spatial reference system</em>"++CR+NL)
 wl.writeString("<tr><td>Datum<td>"+rVTab.ReturnValue(rVTab.findField("datum"),0)++CR+NL)

wl.writeString("<tr><td>Ellipsoide<td>"+rVTab.ReturnValue(rVTab.findField("ellipsoide"),0)++CR+
NL)
 wl.writeString("<tr><td>Datum<td>"+rVTab.ReturnValue(rVTab.findField("datum"),0)++CR+NL)
 wl.writeString("<tr><td>Map
projection<td>"+rVTab.ReturnValue(rVTab.findField("Projection"),0)++CR+NL)

 wl.writeString("<tr><td><b>Extent</b>"++CR+NL)
 wl.writeString("<tr><td><em>Currency of extent data and completeness of
dataset</em>"++CR+NL)
 wl.writeString("<tr><td>Extent date"++CR+NL)
 wl.writeString("<tr><td><em>Bounding area</em>"++CR+NL)
 wl.writeString("<tr><td><em>Geographic area</em>"++CR+NL)
 wl.writeString("<tr><td>Indirect spatial reference system"++CR+NL)
 wl.writeString("<tr><td>Name of areal unit"++CR+NL)
 wl.writeString("<tr><td>Id code of areal unit"++CR+NL)
 wl.writeString("<tr><td>Coverage"++CR+NL)
 wl.writeString("<tr><td><em>Vertical extent</em>"++CR+NL)
 wl.writeString("<tr><td><em>temporal extent</em>"++CR+NL)
 wl.writeString("<tr><td>From date"++CR+NL)
 wl.writeString("<tr><td>To date"++CR+NL)

 wl.writeString("<tr><td><b>Data definition</b>"++CR+NL)
 wl.writeString("<tr><td><em>Application schema description</em>"++CR+NL)
 astring="Select o.name, o.obj#, c.comment$ from sys.obj$ o, sys.com$ c, sys.tab$ t "
 astring=astring+"where o.type=2 and o.owner#=10 and o.obj#=c.obj#(+) and c.col#(+) is null "
 astring=astring+" and t.obj#=o.obj# and t.ts#="+tsCod.AsString
 mVT=VTab.MakeSQL(theSQL, astring)
 'if (mVT.GetNumRecords>0) then
 for each i in mVT
   wl.writeString("<tr><td><em>Object type</em>"++CR+NL)
   wl.writeString("<tr><td>Object                                                        type
name<td>"+mVT.ReturnValue(mVT.getFields.get(0),i)++CR+NL)
   wl.writeString("<tr><td>Object                                                        type
definition<td>"+mVT.ReturnValue(mVT.getFields.get(2),i)++CR+NL)
   theObj=mVT.ReturnValue(mVT.getFields.get(1),i)
   astring="select count(*) from "+mVT.ReturnValue(mVT.getFields.Get(0),i)
   nVTab=VTab.MakeSQL(theSQL, astring)
   n=nVTab.ReturnValue(nVTab.GetFields.Get(0),0)
   wl.WriteString("<tr><td>Occurrences<td>"+n.truncate.asString++CR+NL)

   wl.writeString("<tr><td><em>Attribute type</em>"++CR+NL)
   astring="select * from sys.col$ where obj#="+mVT.ReturnValue(mVT.getFields.get(1),i).AsString
   cVTab=VTab.MakeSQL(theSQL, astring)
   for each k in cVTab
     wl.WriteString("<tr><td>Attribute                                                   type
name<td>"+cVTab.ReturnValue(cVTab.FindField("name"),k)++CR+NL)
     wl.WriteString("<tr><td>Attribute type definition"++CR+NL)
     mObj=cVTab.ReturnValue(cVTab.GetFields.Get(0),k)
     mCol=cVTab.ReturnValue(cVTab.GetFields.Get(1),k)
     'msgbox.info("obj= "+mObj.AsString,"col= "+mCol.AsString)
     astring="select * from sys.com$ where obj#="+mObj.AsString+" and col#="+mCol.AsString
     aVTab=VTab.MakeSQL(theSQL, astring)
     if (aVTab.GetNumRecords>0) then
       wl.WriteString("<td>"+aVTab.ReturnValue(aVTab.GetFields.Get(2),0)++CR+NL)
     end
     codType=cVTab.ReturnValue(cVTab.FindField("type#"),k)
     dscrType=""
     dscrDomain=""
     if (codType=1) then
       dscrType="Char"
       dscrDomain="("+cVTab.ReturnValue(cVTab.FindField("length"),k).truncate.AsString+")"
     elseif (codType=2) then
       dscrType="Number"

dscrDomain="("+cVTab.ReturnValue(cVTab.FindField("precision"),k).truncate.AsString+","+cVTab.Re
turnValue(cVTab.FindField("scale"),k).truncate.AsString+")"
     elseif (codType=12) then
       dscrType="Date"
     end
     wl.WriteString("<tr><td>Attribute type domain<td>"+dscrType+dscrDomain++CR+NL)
   end
   wl.writeString("<tr><td><em>Association type</em>"++CR+NL)
```

```
  astring="select    c.name,    o.name,col.name,    cd.robj#,    decode    (cd.type,1,'Check',2,'Primary
Key',4,'Foreign Key',7,'Not Null')"
  astring=astring+"from sys.con$ c, sys.cdef$ cd, sys.obj$ o, sys.col$ col, sys.ccol$ cc"
  astring=astring+" where o.obj#=cd.obj# and "
  astring=astring+"    o.owner#=10 and"
  astring=astring+"    c.con#=cd.con# and"
  astring=astring+"    cd.con#=cc.con# and"
  astring=astring+"    cc.obj#=col.obj# and"
  astring=astring+"    cc.col#=col.col# and"
  astring=astring+"    cc.obj#=o.obj# and "
  astring=astring+"    cd.obj#="+theObj.AsString
  ccVTab=VTab.MakeSQL(theSQL, astring)
  astring="select obj#, name from sys.obj$ where owner#=10"
  lkVTab=VTab.MakeSQL(theSQL, astring)
  ccVTab.Join(ccVTab.GetFields.Get(3),lkVTab,lkVTab.GetFields.Get(0))

  for each k in ccVTab
    wl.WriteString("<tr><td>Association                                        type
name<td>"+ccVTab.ReturnValue(ccVTab.getFields.Get(0),k)++CR+NL)
    wl.WriteString("<tr><td>Association                                        type
definition<td>"+ccVTab.ReturnValue(ccVTab.getFields.Get(4),k)++CR+NL)
    wl.WriteString("<tr><td>From                                               object
type<td>"+ccVTab.ReturnValue(ccVTab.getFields.Get(1),k)++CR+NL)
    wl.WriteString("<tr><td>To                                                 object
type<td>"+ccVTab.ReturnValue(ccVTab.getFields.Get(6),k)++CR+NL)
    strDef=ccVTab.ReturnValue(ccVTab.getFields.Get(2),k)
    t=ccVTab.ReturnValue(ccVTab.getFields.Get(4),k)
    if (t="Check") then
      strDef=strDef+" checked in ('PT','ES')"
    elseif (t="Primary Key") then
      strDef=strDef+" is Primary Key "

    elseif (t="Foreign Key") then
      strDef=strDef+" references table " + ccVTab.ReturnValue(ccVTab.getFields.Get(6),k)
    else
      strDef=strDef+  " is Not Null "
    end
    wl.WriteString("<tr><td>Cardinality"++CR+NL)
    wl.WriteString("<tr><td>Constraints<td>"+strDef++CR+NL)
   end
  end
  wl.WriteString("<tr><td><b>Administrative Metadata</b>"++CR+NL)
  wl.WriteString("<tr><td><em>Organisation and Organisation role </em>"++CR+NL)
  wl.WriteString("<tr><td>Organisation
name<td>"+rVTab.ReturnValue(rVTab.findField("organisation"),0)++CR+NL)
  wl.WriteString("<tr><td>Abbreviated                                   organisation
name<td>"+rVTab.ReturnValue(rVTab.findField("abbreviate"),0)++CR+NL)
  wl.WriteString("<tr><td>Organisation
address<td>"+rVTab.ReturnValue(rVTab.findField("address"),0)++CR+NL)
  wl.WriteString("<tr><td>Organisation
role<td>"+rVTab.ReturnValue(rVTab.findField("role"),0)++CR+NL)
  wl.WriteString("<tr><td>Alternative rganisation name"++CR+NL)
  wl.WriteString("<tr><td>Function                  of                  organisation
<td>"+rVTab.ReturnValue(rVTab.findField("function"),0)++CR+NL)
  wl.WriteString("<tr><td><em>Point of contact and point of contact role</em>"++CR+NL)
  wl.WriteString("<tr><td>Point                     of                       contact
name<td>"+rVTab.ReturnValue(rVTab.findField("contact"),0)++CR+NL)
  wl.WriteString("<tr><td>Point                     of                       contact
address<td>"+rVTab.ReturnValue(rVTab.findField("cont_adress"),0)++CR+NL)
  wl.WriteString("<tr><td>Point of contact role"++CR+NL)
```

```
 wl.WriteString("<tr><td><em>Distribution</em>"++CR+NL)
 wl.WriteString("<tr><td>Restriction on use"++CR+NL)
 wl.WriteString("<tr><td>Copyright owners"++CR+NL)
 wl.WriteString("<tr><td>Price information"++CR+NL)
 wl.WriteString("<tr><td>Data media"++CR+NL)

wl.WriteString("<tr><td>Formats<td>"+rVTab.ReturnValue(rVTab.findField("formats"),0)++CR+NL
)
 wl.WriteString("<tr><td>On-line access"++CR+NL)
 wl.WriteString("<tr><td>Order"++CR+NL)
 wl.WriteString("<tr><td>Support services"++CR+NL)
 wl.WriteString("<tr><td><b>Metadata reference</b>"++CR+NL)
 wl.WriteString("<tr><td>Entry
date<td>"+rVTab.ReturnValue(rVTab.findField("entry_date"),0).left(10)++CR+NL)
 wl.WriteString("<tr><td>Last                                                        check
date<td>"+rVTab.ReturnValue(rVTab.findField("last_check"),0).left(10)++CR+NL)
 wl.WriteString("<tr><td>Last                                                       update
date<td>"+rVTab.ReturnValue(rVTab.findField("last_update"),0).left(10)++CR+NL)
 wl.WriteString("<tr><td>Spatial reference system of metadata"++CR+NL)
 wl.WriteString("<tr><td><b>Metadata language</b>"++CR+NL)
 wl.WriteString("<tr><td>Metadata
Language<td>"+rVTab.ReturnValue(rVTab.findField("met_language"),0)++CR+NL)

 wl.WriteString("</table><p>")
```

ANNEX 5

DATA BASE QUERY

SOURCE CODE

SOURCE CODE

The implementation of this service integrates several levels of computation, ranging from the Web server platform to the Client browser, through the Security service, Database server and GIS server.
In this chapter we analyse each component separately, and at the end, the overall integration.

1. CGI WEB SERVER

When requesting this service the server checks the connection validity and the access level of the user session. According to the user privileges the service provides an appropriate interface to each dataset. There is an ancillary database with data about the components to be included in each dataset interface. The structure of this database is shown below.

Dataset Interface

TS#; NAME; DATASET; OBJECT_NAME; OBJECT_ORDER; OBJECT_T_ DEFAULT_VALUE; TABLE_NAME; ATTRIBUTE_NAME; ACTION

The process to check the session and turn the service available is written in Perl.

```
#!/usr/netscape/suitespot/install/perl

print "Content-type: text/html\n\n";

$qs=&getParameterIN("QUERY_STRING");
$sessionID=&jGetParameter("sessionID",$qs);

# Checks for session level validity
$mStr = `java jClient20 $sessionID`;
$result = &jGetParameter("result",$mStr);

if ($result eq "OK"){
        $level=&jGetParameter("sessionLevel",$mStr);
        $dsetIn=&jGetParameter("dataset",$qs);
        print "<TITLE> Geomist - Search Databases </TITLE>\n";
        print "<BODY BGCOLOR=\"#00ffff\">\n";
        print "<script language=\"javascript\">\n";
        print "function GetComponentData(d,st){\n";
        print "      str=\"\"\n";
        print "      compList=new Array();\n";
        print "    compList=st.split(\"___\")\n";
        print "      for (i=1; i<compList.length; i++){\n";
        print "              component=new Array();\n";
        print "              component=compList[i].split(\"£\");\n";
        print "              str=str+\"\$\"+component[3]+\"£\";\n";
        print "              if (component[1]==\"text\"){\n";
        print "                        aux=d.forms[0].elements[component[0]].value;\n";
        print "           if (aux==\"\") aux=\"9\"\n";
        print "                        str=str+aux+\"£\";\n";
        print "              }else if (component[1]==\"comboc\"){\n";
        print "                        elem=d.forms[0].elements[component[0]];\n";
        #print "                        str=str+elem.options[elem.selectedIndex].value+\"£\";\n";
        print "                        i=i+1;\n";
        print "                        component=new Array();\n";
        print "                         component=compList[i].split(\"£\");\n";
        print "                         aux=d.forms[0].elements[component[0]].value;\n";
        print "               if (aux==\"\") aux=\"9\"\n";
        print "                         str=str+aux+\"£\";\n";
        print "                 }else{\n";
        print "                         elem=d.forms[0].elements[component[0]];\n";
        print "                         str=str+elem.options[elem.selectedIndex].value+\"£\";\n";
        print "                 }\n";
        print "                 str=str+component[4]+\"£\";\n";
        print "                 str=str+component[2];\n";
        print "   }\n";
        print "   return (str)\n";
        print "}\n";
        print "</script><br>\n";

        #print "DSETIN= $dsetIn ; Session      = $sessionID";

        print "<TABLE BORDER=0 ALIGN=CENTER>\n";
        print "<FORM>\n";
        print "<TR><TD ALIGN=CENTER>\n";
        print "<STRONG>Select your dataset</STRONG>\n";
        print                           "<TR><TD                            ALIGN=CENTER><SELECT
onChange=\"{mObj=top.frames[2].document.forms[0].elements[0];top.frames[2].document.location='/cgi/queryBD
2.pl?cmd=xpto&dataset='+mObj.options[mObj.selectedIndex].value+'&sessionID=$sessionID';}\">\n";
        @mDatasets=&loadDatasets();
        print "<OPTION VALUE=\"$dsetIn\" [SELECTED]>$dsetIn\n";
        foreach $ds (@mDatasets){
                $d=$ds;
                if ($ds ne $dsetIn){
                        print "<OPTION VALUE=\"$d\">$d\n";
                }
        }
        print "</SELECT>\n";
        print "</P>\n";
        print "<HR WIDTH=80%>\n";


        $str=&loadComponents($dsetIn);
        # $pstr=&LoadPresent($dsetIn);
        # $fieldsStr=&loadFields($dsetIn);
        # $viewStr=&loadView($dsetIn);
        # print "$dsetIn=$str\n";
        if ($str ne ""){
                print "</P>\n";
                print "<HR WIDTH=80%>\n";
                print   "<TR><TD   ALIGN=CENTER><INPUT   TYPE=\"button\"   name=\"submitQuery\"
value=\"Submit          Query\"          onClick=\"{mDoc=top.frames[2].document;d=top.frames[2].document;
mQuery=GetComponentData(d,'$str');
top.frames[0].location='/.esrimap?nameX=GeomistMap&cmd=java.ShowDatabase&arg='+mQuery+'&dset=$dset
In&sessionID=$sessionID';}\">\n";
                print "<INPUT TYPE=\"reset\" value=\"Clear\">\n";
        }
        print "</FORM>\n";
        print "</TABLE>\n";
}else{
  print "<HTML>\n";
  print "<BODY onLoad=\"top.location='/geomist/timeout.html'\">\n";
  print "</HTML>\n";

}

#######################################################################################
sub loadDatasets{
        @ds=();
                if( ! open ( IN , "/disk2/httpd/data/datasets.txt")){
                print "Impossivel abrir ficheiro de datasets!\n";
        }else{
                while($line = <IN>){
                        chop($line);
                        push(@ds,$line);
                }
        }
        @ds;
}

sub LoadPresent{
        $ds=$_[0];
        $pstr="";
                if( ! open ( IN , "/disk2/httpd/htdocs/tmp/geomist/inter_database/present_fields2.txt")){
                print "Impossivel abrir ficheiro de datasets!\n";
        }else{
                while($line = <IN>){
                        @all =split (/ /,$line);
                        chop($all[1]);
                        if ($all[0] eq $ds){
                          $pstr=$pstr."£".$all[1];
                        }
                }
        }
        $pstr;
}
sub loadView{
        $ds=$_[0];
        local($xstr)="";
                if( ! open ( IN , "/disk2/httpd/htdocs/tmp/geomist/inter_database/dataset_views2.txt")){
                print "Impossivel abrir ficheiro de views!\n";
        }else{
                while($line = <IN>){
                        @all =split (/ /,$line);
                        chop($all[1]);
                        if ($all[0] eq $ds){
                          $xstr=$all[1];
                        }
                }
        }
        $xstr;
}


sub loadComponents{
        $ds=$_[0];
        #local($lev)=$_[1];
        $inCombo=0;
        local($n)=1;
        local($str)="";
        local($aux)="";

        if( ! open ( IN , "/disk2/httpd/data/dataset_interface2.txt")){
                print "Impossivel abrir ficheiro de datasets!\n";
        }else{
                $line=<IN>;
                $line=<IN>;
                while($line = <IN>){
                   @all =split (/;/,$line);
                   #print "$all[4] : $level<br>";
                   #$all[4]++;
                   if (($all[2] eq $ds) && ($all[4] <= $level)){
                        if (($inCombo==0) && (($all[5] eq "combo") || ($all[5] eq "comboc"))){
                                print "<TR><TD ALIGN=CENTER><SELECT>\n";
                                $aux=$n."£".$all[5]."£".$all[7]."£".$all[8]."£".$all[9];
                                chop($aux);
                                $str=$str."___".$aux;
                                $inCombo=1;
                                $n++;
```

```
            }elsif(($inCombo==1) && ($all[5] ne "combo") && ($all[5] ne "comboc")){
                print "</SELECT>\n";
                $inCombo=0;
            }
            &printComponent($all[3],$all[5],$all[6],$all[8]);
            if($all[5] eq "text"){
                $aux=$n."£".$all[5]."£".$all[7]."£".$all[8]."£".$all[9];
                chop($aux);
                $str=$str."___".$aux;
                $n++;
            }
         }
        }
        if($inCombo==1){
            print "</SELECT>\n";
        }
        $str;
    }
}

sub loadFields{
    $ds=$_[0];
    local($ftr)="";
    local($aux)="";

    if(!open(IN, "/disk2/httpd/htdocs/tmp/geomist/inter_database/access_fields.txt")){
        print "Impossivel abrir ficheiro de campos!\n";
    }else{
        while($line = <IN>){
          #print "L=$line<br>";
          @all =split (/:/,$line);
          if(($all[0] eq $ds) && ($all[2] <= $level)){
            $fstr=$fstr."£".$all[1];
          }
        }
        $fstr;
        #print "FL = $fstr<br>";
    }
}

sub printComponent{
    local($objectName)=$_[0];
    local($objectType)=$_[1];
    local($objectValue)=$_[2];
    local($objectField)=$_[3];
    if($objectType eq "label"){
        print "<BR><TR><TD ALIGN=CENTER>$objectValue\n";
    }elsif($objectType eq "text"){
        print    "<TR><TD    ALIGN=CENTER><INPUT    TYPE=\"text\"    name=\"$objectName\"
value=\"$objectValue\">\n";
    }elsif(($objectType eq "combo") || ($objectType eq "comboc")){
        print "<OPTION VALUE=\"$objectField\">$objectValue\n";
    }


}

sub getParameterIN{
    local($result);
    local($param)=$_[0];

    foreach $element (keys %ENV){
        if($element eq $param){
            $result=$ENV{$element};
        }
    }
    $result;
```

```
}

sub jGetParameter{
        my $ret;
        my $arg=$_[0];
        my $in=$_[1];
        @all=split(/&/,$in);
        foreach  $member(@all){
                @param=split(/=/,$member);
                if ($param[0] eq $arg){
                        $ret=$param[1];
                }
        }
        #print "QS=$in ARG=$arg RET=$ret\n";

        $ret;
}


sub getParameterStr{
        local ($str)=$_[0];
        local ($ds)="Nil";
        local ($aux)=$_[1];

        @all=split (/&/,$str);
        @all2=split(/=/,$all[1]);
        @all3=split(/=/,$all[2]);
        #print "ALL2[0]=$all2[0]<br>",
        if ($all2[0] eq $aux){
                $ds=$all2[1];
        }elsif ($all3[0] eq $aux){
                $ds=$all3[1];
        }
        $ds;
}

sub getFormParams{
 my $ret;

   if($ENV{'REQUEST_METHOD'} eq "GET"){
      $in = $ENV{'QUERY_STRING'};
   }
   elsif($ENV{'REQUEST_METHOD'} eq "POST"){
      read(STDIN,$in,$ENV{'CONTENT_LENGTH'});
   }
   return $in;
}

##
# IN: stringToParse
# RET: username, password
##
sub parseParams{
   my  ($username,$password, $ip);
   $newline = "&";
   @query = split(/$newline/, $_[0] ) ;

   foreach $member (@query) {
      ($head, $tail)= split(/=/, $member );
      if($tail){
            $tail= &unescapeStr($tail);
            $tail = &cleanDosStr($tail);
      }

       print "params:(head,tail): ($head, $tail)$nl" if $debug;
       SWITCH:
       {
            $head =~ /\busername\b/&& do {
                  $username= $tail;
                  last SWITCH;
            };
            $head =~ /\bpassword\b/&& do {
                  $password= $tail;
                  last SWITCH;
            };
       }

      }
      if(!$username){
        #one last try, trt get it from the environment
        #problem, passwd is inaccessible via this method
            $username = $ENV{'REMOTE_USER'};
            if($username){
               &logIt("WARN","REMOTE_USER filled, probably unable to get passwd!");
            }
      }
      $ip = $ENV{'REMOTE_HOST'};
      #print "IP= $ip";
      #if( &dangerousChars($username)){
      #       &error("username invalido");
      #       exit;
      # }
      #if( &dangerousChars($password)){
      #       &error("password invalida");
      #       exit;
      # }
      return ($username.$password, $ip);
}

##
# unescapeStr Converts escaped characters to proper char (ex. ~,$,...)
##
sub unescapeStr
{
   $buffer = $_[0];
   $buffer || return;

   $buffer =~ tr/+/ /;
   $buffer =~ s/%([a-fA-F0-9][a-fA-F0-9])/pack("C", hex($1))/eg;

   return $buffer;
}

##
# IN:   string
# RET:  string removed \r values
##
sub cleanDosStr{
 my $str = shift(@_);

   $str =~ s/\r\n/\n/g;
   return $str;
}


sub uncypher{
 return $_[0];
}

##
# IN: str to check
# don't let some dangerous chars pass
# Shouldn't realy be necessaru as no evals or
# shell invocations... Better be safe than sorry
```

```
##
sub dangerousChars{
 my $str = shift(@_);

     return $str;
}
```

2. CLIENT WEB BROWSER

On the client side the browser receives a new document with four distinct frames. First (F1) provide an appropriate interface according to the user profile and the dataset selected. When submitting a request the return the results to the frame F2, as a list of entities. Each entry in this list has an hyperlink to a specific function to display the full instance attributes at frame F3. From F4 the user can switch service.

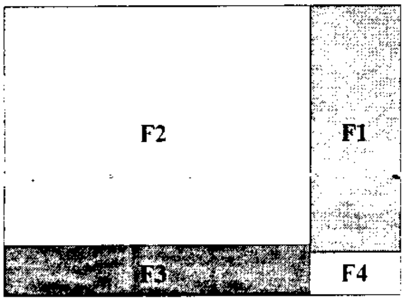


Figure 2 – Database query interface

Bellow is the example source code to be included in a document presented at F1 for an user with session id = *378923977817131934370661557438743998370661557438743998* and dataset *Technical Archive.*

```
<TITLE> Geomist - Search Databases </TITLE>
<BODY BGCOLOR="#00ffff">
<script language="javascript">

function GetComponentData(d,st){
        str=""
        compList=new Array();
   compList=st.split("___")
        for (i=1; i<compList.length; i++){
            component=new Array();
            component=compList[i].split("£");
            str=str+"$"+component[3]+"£",
            if (component[1]=="text"){
                  aux=d.forms[0].elements[component[0]].value;
             if (aux=="") aux="9"
                  str=str+aux+"£";
            }else if (component[1]=="comboc"){
                  elem=d.forms[0].elements[component[0]];
                  i=i+1;
                  component=new Array();
                  component=compList[i].split("£");
                  aux=d.forms[0].elements[component[0]].value;
             if (aux=="") aux="9"
                  str=str+aux+"£";
            }else{
```

```
            elem=d.forms[0].elements[component[0]];
            str=str+elem.options[elem.selectedIndex].value+"£";
        }
        str=str+component[4]+"£";
        str=str+component[2];
  }
  return (str)
}
</script><br>
<TABLE BORDER=0 ALIGN=CENTER>
<FORM>
<TR><TD ALIGN=CENTER>
<STRONG>Select your dataset</STRONG>
<TR><TD ALIGN=CENTER><SELECT
onChange="{mObj=top.frames[2].document.forms[0].elements[0];top.frames[2].document.location='/cgi/queryBD
2.pl?cmd=xpto&dataset='+mObj.options[mObj.selectedIndex].value+'&sessionID=378923977817131934370661557
438743998370661557438743998';}">
<OPTION VALUE="Exploration_Permits" [SELECTED]>Exploration_Permits
<OPTION VALUE="Drillholes">Drillholes
<OPTION VALUE="Geology">Geology
<OPTION VALUE="Gravimetry">Gravimetry
<OPTION VALUE="Magnetic">Magnetic
<OPTION VALUE="Mineral_Occurrences">Mineral_Occurrences
<OPTION VALUE="Radiometry">Radiometry
<OPTION VALUE="Technical_Archive">Technical_Archive
</SELECT>
</P>
<HR WIDTH=80%>
<BR><TR><TD ALIGN=CENTER>Company
<TR><TD ALIGN=CENTER><INPUT TYPE="text" name="company_txt" value="">
<BR><TR><TD ALIGN=CENTER>Substances_Group
<TR><TD ALIGN=CENTER><INPUT TYPE="text" name="group_txt" value="">
<BR><TR><TD ALIGN=CENTER>Substances
<TR><TD ALIGN=CENTER><INPUT TYPE="text" name="subs_txt" value="">
<BR><TR><TD ALIGN=CENTER>Abstract
<TR><TD ALIGN=CENTER><INPUT TYPE="text" name="abs_txt" value="">
<BR><TR><TD ALIGN=CENTER>Order by
<TR><TD ALIGN=CENTER><SELECT>
<OPTION VALUE="company">Company
<OPTION VALUE="groups">Subs_Group
</SELECT>
</P>
<HR WIDTH=80%>
<TR><TD ALIGN=CENTER><INPUT TYPE="button" name="submitQuery" value="Submit Query"
onClick="{mDoc=top.frames[2].document;d=top.frames[2].document;
mQuery=GetComponentData(d,'___1£text£vpermits£company£from___2£text£vpermits£groups£from___3£text£vpe
rmits£subsList£from___4£text£vpermits£remark£from___5£combo£vpermits£company£order');
top.frames[0].location='/.esrimap?nameX=GeomistMap&cmd=java.ShowDatabase&arg='+mQuery+'&dset=Explo
ration_Permits&sessionID=378923977817131934370661557438743998370661557438743998';}">
<INPUT TYPE="reset" value="Clear">
</FORM>
</TABLE>
```

3. DATABASE SERVER

The database server receives the request with all parameters, checks the database and formats the output according to the window structure presented above.

There is an ancillary database to relate parameters, database fields and names to present within the output document. The structure of this database is:

(DATASET ; ORACLE FIELD ; ALIAS NAME ; ORACLE TABLE/VIEW ; ACCESS LEVEL ; PRESENTATION FLAG ; MAPPING FLAG)

The *Presentation Flag* indicates if this field is to be present within the body frame of window (F1) and the *Mapping Flag* indicates if the dataset is directly geo-positioned, allowing the map display of results.

```
'Name:    java.ShowDatabase
'
'Author:  Jorge Lopes <jlopes@igm.pt>
'Creation: 05/01/99
'Update:   13/06/99
'
'IN:      (query : RegularList (fields:IrregularList({fieldName,value,action,table},"£"),$) ;
'         dataset  : String ;
'         sessionID : String ;
'         )
'
'OUT:     table : HTML Document
'
'
wl = Weblink.the
wl.WriteResponseHeader("Content-type: text/html"+CR+NL+CR+NL)
wl.WriteString("<HTML>"++CR+NL)
query=self.Get(0).Get(1)
query=query.right((query.count)-1)
pstr=""
dset=self.Get(1).Get(1)
fstr=""
vstr=""
mSession=self.Get(2).Get(1)

'msgbox.report("query="+query+nl+"dset="+dset+nl,mSession)

accessLevel=av.run("java.CheckSession",{mSession})
if (accessLevel < 1) then
 wl.WriteString("<BODY onLoad=""_top.location='/tmp/geomist/timeout.html'"">"++CR+NL)
 return "end"
end

mVT=av.getProject.FindDoc("oracle.tables").getVTab
themeField=mVT.FindField("Theme")
aliasName=mVT.FindField("aliasName")
oraField=mVT.FindField("oraField")
acLevel=mVT.FindField("acLevel")


strSQL = "Select "
bodyFields=List.Make
showFields=""
showFieldsList=List.Make
mapping=false
first=true

for each f in mVT
  t=mVT.returnValue(themeField,f)
  v=mVT.returnValue(acLevel,f)
  if ((t=dset) and (v <= accessLevel)) then
    if (first) then
      mapping=mVT.ReturnValue(mVT.GetFields.get(6),f)
    first=false
   end
   fld=mVT.ReturnValue(oraField,f)
   mView=mVT.ReturnValue(mVT.GetFields.get(3),f)
   p=mVT.ReturnValue(mVT.GetFields.get(5),f)
   if (p > 0) then
     bodyFields.add(fld)
   end
   showFields=showFields+mVT.ReturnValue(aliasName,f)+"££"
   showFieldsList.add(fld)
   strSQL = strSQL+fld+", "
  end
end
strSQL = strSQL.left(strSQL.count-2)
strSQL=strSQL + "from " + mView + " where "
'msgbox.report(strSQL,query)

compList=query.asTokens("$")
for each f in compList
  mList=f.asTokens("£")
  if (mList.get(2)="from") then
   if (mList.Get(1)<>"9") then
     strSql=strSQL+ "upper("+mList.get(0)+") like '%"+mList.Get(1).ucase+"%' and   "
   end
  else
   strSQL=strSQL.left(strSQL.count-6)+" order by "+mList.get(0)
  end
end

'msgbox.report(strSQL,"")

'mSQL=strSQL
theSQL=SQLCon.Find("Oracle")
theSQL.Login("jlopes/jlopes@geomist")
mVT=VTab.MakeSQL(theSQL, strSQL)

nRecs=mVT.GetNumRecords
'msgbox.report(strSQL,nRecs.AsString)

wl.WriteString("<script language=javascript>"++CR+NL)
wl.WriteString("function WriteRecord(d,st){"++CR+NL)
wl.WriteString("fl="""+showFields+""";"++CR+NL)
wl.WriteString("d.clear();"++CR+NL)
wl.WriteString("d.close();"++CR+NL)
wl.WriteString("d.writeln(""<body bgcolor='#ff7543'>"");"++CR+NL)
wl.WriteString("rows=new Array();"++CR+NL)
wl.WriteString("rows=fl.split(""££"");"++CR+NL)
wl.WriteString("d.writeln(""<table border=1>"");"++CR+NL)
wl.WriteString("for (i=0;i<rows.length;i++){"++CR+NL)
wl.WriteString("   cols=new Array();"++CR+NL)
wl.WriteString("   cols=st.split(""££"");"++CR+NL)
wl.WriteString("   d.writeln(""<tr><td><b>""+rows[i]+""</b>"");"++CR+NL)
wl.WriteString("   d.writeln(""    <td>    ""+cols[i]);"++CR+NL)
wl.WriteString("}"++CR+NL)
wl.WriteString("d.writeln(""</table>"");"++CR+NL)
wl.WriteString("}"++CR+NL)
wl.WriteString("</script>"++CR+NL)

wl.WriteString("<CENTER>")
wl.WriteString("<FONT SIZE=5> Dataset "+dset+"<BR>")
wl.WriteString("<FONT SIZE=3> N° = "+nRecs.AsString++CR+NL)
if (mapping) then

URL="'http://geomist.igm.pt/.esrimap?nameX=GeomistMap&cmd=java.showSelection&session="+mSe
ssion+nRecs.asString+"'"
 WINDOW_NAME=",'ShowResults',"

WINDOW_PARAMETERS="'width=300,height=200,directories=no,location=no,menubar=no,resizable
=yes,scrollbars=yes,status=no,toolbar=no'"
 wl.WriteString("<FORM>        <INPUT        TYPE=""button""        value=""Display        Map""
onClick=""window.open("+URL+WINDOW_NAME+WINDOW_PARAMETERS+")"">
</FORM>"++CR+NL)
end
wl.WriteString("<HR WIDTH=80%>")
wl.WriteString("</CENTER>")
wl.WriteString("<OL>")

wl.WriteString("<FONT SIZE=4>")
x=0
for each r in mVT
 wl.WriteString("<LI><A>")
 str=""
 for each k in showFieldsList
  str=str+"££"+mVT.ReturnValue(mVT.FindField(k),r).AsString
 end
 str=str.right(str.count-2)
 wl.WriteString("<A HREF=""javascript:WriteRecord(top.frames[1].document,'"+str+"')"">")
 for each p in bodyFields
  wl.WriteString("["+mVT.ReturnValue(mVT.FindField(p),r)+"]")
 end
 wl.WriteString("</a>"++CR++NL)
end
av.run("geomist.logTask",{"ShowDatabase"++strSQL})
if (mapping.not) then
 theSQL.logout
 return "NIL"
end
MAP_EXTENT="500000_4113000_252000_161000"
MAP_COVERS="FPI Limits__1-50000"
av.run("java.drawMap",{MAP_COVERS,MAP_EXTENT})
theView=av.GetProject.FindDoc("Basemap")
theDisplay=theView.getDisplay
theTheme=theView.FindTheme(dset)
theFTab=theTheme.GetFTab
fieldName=mVT.GetFields.Get(0).GetName
mF0=mVT.GetFields.Get(0)
mF1=theFTab.FindField(fieldName)

theFTab.join(mF1,mVT,mF0)
theBitmap=theFTab.GetSelection
f2N=theFTab.GetFields.Get(3).GetName
expr="(["+f2N+"].isNull.Not)"
theFTab.Query(expr,theBitmap,#VTAB_SELTYPE_NEW)
theFTab.UpdateSelection
theTheme.setVisible(true)
theHeight=350
theWidth=theHeight*1.565217
tmpFN = theView.ExportToJPEG(theDisplay.returnExtent,theWidth,theHeight,85,false)
system.execute("cp "+tmpFN.getName+" /tmp/"+mSession+nRecs.asString+".jpg")
file.delete(tmpFN)
```

```
theTheme.ClearSelection
file.delete(tmpFN)
theTheme.ClearSelection

for each t in theView.GetVisibleThemes
  t.SetVisible(false)
end
theFTab.UnjoinAll
theSQL.logout
```

ANNEX 6

INTERACTIVE MAPPING

SOURCE CODE

SOURCE CODE

As described above, there are distinct levels of software, corresponding to processes running at distinct platforms and scopes. Follow the code analysed and executed at Web server environment, GIS server and at client side, within a Web browser. There is a special chapter dedicated to the security level.

1. CGI WEB SERVER

When starting the service there is a process at Web server environment to check the connection validity and launching a new Web page into the client browser.

```
(..........)
print TMPOUT "<HTML>\n";
    print TMPOUT "<TITLE>Geomist - Interactive Mapping </TITLE>\n";
    print TMPOUT "<BODY BGCOLOR='#FFFFFF'
onLoad=\"window.open('','showResults','width=300,height=200,directories=no,location=no,menubar=n
o,resiz
able=yes,scrollbars=yes,status=no,toolbar=no')\">\n";
    print TMPOUT "<CENTER>\n";
    print TMPOUT "<FONT SIZE=5> <STRONG>Interactive
Mapping</FONT><P></STRONG>\n";
    print TMPOUT "<TABLE BORDER=4><tr><td>\n";
    print TMPOUT "<APPLET CODEBASE=\"/java\" CODE=\"JMap45.class\" width=680
height=500>\n";
    print TMPOUT "<PARAM NAME=fname
VALUE=\"/.esrimap?name=GeomistMap&cmd=java.RedrawMap&tList=Topographic__FPI Limits__1-
50000&extent=50
0000_4113000_252000_161000&sessionID=$sessionID\">\n";
    print TMPOUT "<PARAM NAME=coversID VALUE=\"Drillholes&No Metallic&Metallic&\">\n";
    print TMPOUT "<PARAM NAME=coversList VALUE=\"Drillholes&Drillholes Campaign&No
Metallic&Metallic&Topographic&1-50000&FPI Limits&Exploration P
ermits&Geological Faults&Geological Synthesis&Gravimetric grid&Magnetic grid&Radiometric
grid&\">\n";
    print TMPOUT "<PARAM NAME=baseData VALUE=\"Topographic&FPI Limits&1-
50000&\">\n";
    print TMPOUT "<PARAM NAME=sessionID VALUE=\"$sessionID\">\n";
    print TMPOUT "</APPLET>\n";
    print TMPOUT "</TABLE>\n";
    print TMPOUT "<A
HREF='/cgi/start_service2.pl?sessionID=$sessionID&username=$username&service=metadata'
TARGET='_top'><IMG SRC='/tmp/geomist/
images/geomistMetadataMinW.jpg' ALT='Search Metadata' BORDER=0>\n";
    print TMPOUT "<A
HREF='/cgi/start_service2.pl?sessionID=$sessionID&username=$username&service=database'
TARGET='_top'><IMG SRC='/tmp/geomist/
images/geomistDatabasesMinW.jpg' ALT='Search Databases' BORDER=0>\n";
    print TMPOUT "</BODY>\n";
    print TMPOUT "</HTML>\n";
    print "<HTML>\n";
    print "<BODY onLoad=\"this.location='/temp/$sessionID.html'\">\n";
    print "</BODY>\n";
    print "</HTML>\n";
(..........)
```

2. GIS SERVER

From this point each user action is intersected at the Java Applet environment to be sent to the specific service designed and dedicated to this function.

```
'
'Name:    java.Connect
'
'Author:  Jorge Lopes <jlopes@igm.pt>
'Creation: 15/12/98
'Update:   15/12/98
'
'

WebServer  = "w3-geomist"
WebServerPort = 80
_WebServerURL = "/bin/scripts/.esrimap"
mapName    = "GeomistMap"
theTimeOut = "20"
theRetry   = "5"
theMaxPend = "10"
wl = Weblink.Make("Dispatch",0)

MyPort = Socket.The.GetPortNumber
username = ""
password = ""
web = Internet.Make(WebServer,WebServerPort,"","")

machine = String.MakeBuffer(256)
System.QueryLocalMachineName(machine,machine.Count)
theMachine = machine.Trim
MyPortString = MyPort.AsString
WebServerPortString = WebServerPort.AsString

web.SendRequest(_WebServerURL+"?Reg="+mapName
            +"&Machine="+theMachine
            +"&Port="+MyPortString
            +"&Timeout="+theTimeOut
            +"&Retry="+theRetry
            +"&MaxPending="+theMaxPend)


'
'Name:    java.CheckSession
'
'Author:  Jorge Lopes <jlopes@igm.pt>
'Creation: 20/07/99
'Update:   20/07/99
'
'IN:      (sessionID : String)
'
'OUT:     enum ( -1:Unknown error | 0:Session not valid | enum (1|2|3):sessionAccessLevel)
'
'

mSession=self.get(0)
result = -99
```

```
'------ Check for session --------
system.execute("cd /atlas/geomist ; java jClient21du "+mSession+" > jtest.log")

lf=LineFile.Make("/atlas/geomist/jtest.log".AsFileName,#FILE_PERM_READ)
str=lf.ReadElt
if (str.count=0) then
  result = -1
else
  resList = str.asTokens("& ")
  if (resList.get(0).asTokens("=").get(1)="Error:") then
    result = 0
  else
    result = resList.get(2).asTokens("=").get(1).asNumber
  end
end

return result



'
'java.DrawMap
'
'Author:       Jorge Lopes
'EMail:        jlopes@igm.pt
'Creation date: 20/03/1999
'Last update:   20/03/1999
'
'All rights reserved
'
'
'IN: coversList  (__ separator)
'    extentList   (_ separator)
'
'
coversList=self.Get(0)
extentList=self.Get(1)
theView=av.GetProject.FindDoc("Basemap")


bl=extentList.asTokens("_")
theView.GetDisplay.SetExtent(rect.Make((bl.get(0).AsNumber)@(bl.get(1).AsNumber),(bl.get(2).AsNumb
er)@(bl.get(3).AsNumber)))


for each t in theView.GetVisibleThemes
   t.SetVisible(false)
end
mList=coversList.AsTokens("__")
for each i in 0..(mList.Count-1)
  auxt=mList.get(i)
  if (theView.FindTheme(auxt)<>Nil) then
    theView.FindTheme(auxt).setVisible(true)
    if (theView.FindTheme(auxt+"_e")<>Nil) then
        theView.FindTheme(auxt+"_e").SetVisible(true)
    end
  end
end
theHeight=350
theWidth=theHeight*1.565217
```

```
theView.GetWin.Open
theView.GetWin.resize(theWidth,theHeight)
theView.GetDisplay.SetExtent(rect.Make((bl.get(0).AsNumber)@(bl.get(1).AsNumber),(bl.get(2).AsNumb
er)@(bl.get(3).AsNumber)))



'
'Name:    java.RedrawMap
'
'Author:  Jorge Lopes  <jlopes@igm.pt>
'Creation: 15/12/98
'Update:   21/07/99
'
'IN:      (sessionID : String ; Covers : List() ; MapExtent : List ("__"))
'
'OUT:     Map : Image (jpeg)
'
'


wl = Weblink.the
wl.WriteResponseHeader("Content-type: image/jpeg"+CR+NL+CR+NL)

theView=av.GetProject.FindDoc("Basemap")
'------ Check for 3 arguments -------
if (self.count<3) then
  wl.Writefile("/atlas/geomist/tmp/error.jpg".asFileName)
  return "end"
end

mSession=self.get(2).get(1)

accessLevel=av.run("java.CheckSession",{mSession})
if (accessLevel < 1) then
  wl.Writefile("/atlas/geomist/tmp/error.jpg".asFileName)
 return "end"
end


'------ The session is valid -------
mCovers=self.Get(0).Get(1)
mExtent=self.get(1).get(1)

'msgbox.report("mCovers="+mCovers+nl+"mExtent="+mExtent,"")

theFile="Nil"

'if (mExtent="500000_4113000_252000_161000") then
' str=mExtent+mCovers
' mVT=av.Getproject.FindDoc("Maps").GetVTab
' mName=mVT.FindField("map_name")
' for each i in mVT
'   theName=mVT.returnValue(mName,i)
'   if (theName=str) then
'     theFile=mVT.ReturnValue(mVT.FindField("map_file"),i)
'     wl.Writefile(theFile.asFileName)
'     return "end"
'   end
' end
'end
```

```
av.run("java.drawMap",{mCovers,mExtent})
theHeight=350
theWidth=theHeight*1.565217
theDisplay=theView.getDisplay
tmpFN = theView.ExportToJPEG(theDisplay.returnExtent,theWidth,theHeight,85,false)
wl.WriteResponseHeader("Content-type: image/jpeg"+CR+NL+CR+NL)
wl.Writefile(tmpFN)

'if (mExtent="500000_4113000_252000_161000") then
' mVT.SetEditable(true)
' nRec=mVT.AddRecord
' mVT.SetValue(mName,nRec,str)
' mVT.SetValue(mVT.FindField("map_file"),nRec,tmpFN.asString)
' mVT.SetEditable(false)
'else
 file.delete(tmpFN)
'end
for each t in theView.GetVisibleThemes
  t.SetVisible(false)
end
return true


' Name:    java.getIdentify
'
' Author:  Jorge Lopes  <jlopes@igm.pt>
' Creation: 15/12/98
' Update:   08/06/99
'
' IN:       (theme : String ; location : Point (x@y))
'
' OUT:      table : HTML Document
'
'


'----- Write page output header ---------
wl = Weblink.the
wl.WriteResponseHeader("Content-type: text/html"+CR+NL+CR+NL)
wl.WriteString("<HTML>"+CR+NL)
wl.WriteString("<TITLE> Geomist - Identify features</TITLE>"+CR+NL)

'------ Check for 3 arguments -------
if (self.count<3) then
  msgbox.info("poucos argumentos","")
  return "end"
end


'------ Unpack arguments ------------
mTheme=self.Get(0).Get(1)
mLocation=self.Get(1).Get(1).asTokens("@")
x=mLocation.get(0).asNumber
y=mLocation.get(1).asNumber
mSession=self.Get(2).Get(1)


accessLevel=av.run("java.CheckSession",{mSession})
if (accessLevel < 1) then
 wl.WriteString("<BODY onLoad=""this.close()"">"+CR+NL)
 return "end"
end

'----- Get Data -------------------------
theView=av.GetProject.FindDoc("Basemap")

if (mTheme="--Themes--") then
  wl.writeString("Select one dataset"+CR+NL)
  return "end"
end
wl.WriteString("<CENTER>"+CR+NL)
wl.WriteString("<FONT SIZE=5> Dataset:<B> "+mTheme+"</B></FONT><BR>"+CR+NL)


ml=av.run("theme.checkDataLocation",{mTheme})

if (ml = "nil") then
  wl.writeString("No data associated"+CR+NL)
  return "end"
else
  '-------- Check for intersected features ---------
  theTheme = theView.FindTheme(mTheme)
  recordList = theTheme.FindByPoint(x@y)
  if (recordList.Count=0) then
     '------ Try to find in Spanish covers --------
     theTheme=theView.FindTheme(mTheme+"_e")
     if (theTheme<>nil) then
       recordList = theTheme.FindByPoint(x@y)
        ml = "local"
     end
     if (recordList.Count=0) then
       wl.writeString("No features"+CR+NL)
       return "end"
     end
  end
  theFTab = theTheme.GetFTab
end

'----- Write number of occurrences --------
wl.WriteString("<FONT                    SIZE=2>                    "+recordList.Count.asString+"
occurrences</FONT><P>"+CR+NL)

'----- Now write the results ---------
wl.WriteString("<TABLE BORDER=2>"+CR+NL)
wl.WriteString("<TR>"+CR+NL)

if (ml = "local") then
 fieldList = theFTab.getFields
 for each f in fieldList
   if (f.isVisible) then
     wl.WriteString("<TD><B>"+f.getAlias+CR+NL)
   end
 end
 for each r in recordList
   wl.WriteString("<TR>"+CR+NL)
   for each f in fieldList
      if (f.isVisible) then
        wl.WriteString("<TD>"+theFTab.ReturnValueString(f,r)+CR+NL)
      end
   end
  end
 else
  mVT=av.getProject.FindDoc("oracle.tables").getVTab
  themeField=mVT.FindField("Theme")
  aliasName=mVT.FindField("aliasName")
  oraField=mVT.FindField("oraField")
  acLevel=mVT.FindField("acLevel")

  strSQL = "Select "
  for each f in mVT
    t=mVT.returnValue(themeField,f)
    v=mVT.returnValue(acLevel,f)
    if ((t=mTheme) and (v <= accessLevel)) then
      mView=mVT.ReturnValue(mVT.GetFields.get(3),f)
      wl.WriteString("<TD><B>"+mVT.ReturnValue(aliasName,f)+CR+NL)
      strSQL = strSQL+mVT.ReturnValue(oraField,f)+", "
    end
  end
  strSQL = strSQL.left(strSQL.count-2)
  r = theFTab.getFields.get(1).getName
  strSQL=strSQL + " from " + mView + " where "
  for each f in recordList
    strSQL = strSQL+r+" = '"+theFTab.ReturnValueString(theFTab.getFields.get(1),f)+"' or "
  end
  strSQL = strSQL.left(strSQL.count-3)

  'msgbox.report(strSQL,"OLA")
  theSQL=SQLCon.Find("Oracle")
  theSQL.Login("jlopes/jlopes@geomist")
  mVT=VTab.MakeSQL(theSQL, strSQL)
  'msgbox.report(strSQL,mVT.GetNumRecords.asString)

  for each r in mVT
    wl.writeString("<TR>"+CR+NL)
    for each f in mVT.GetFields
      wl.writeString("<TD>"+mVT.ReturnValueString(f,r)+CR+NL)
    end
  end
  theSQL.logout
 end

wl.WriteString("</TABLE>"+CR+NL)
wl.WriteString("</HTML>"+CR+NL)




' Name:    java.getSelectDataByArea
'
' Author:  Jorge Lopes <jlopes@igm.pt>
' Creation: 15/12/98
' Update:  21/07/99
'
' IN:      (theme : String ; polygon : List ({x@@y}, ";;") )
'
' OUT:     table : HTML Document
'
'
```

```
mSession=self.get(0).get(1)

wl = Weblink.the

wl.WriteResponseHeader("Content-type: text/html"+CR+NL+CR+NL)
wl.WriteString("<HTML>"+CR+NL)

accessLevel=av.run("java.CheckSession",{mSession})
if (accessLevel < 1) then
 wl.WriteString("<BODY onLoad=""this.close()"">"+CR+NL)
 return "end"
end

wl.WriteString("<TITLE> Geomist - Identify features</TITLE>"+CR+NL)
wl.WriteString("<BODY BGCOLOR=""#ffffff"">"+CR+NL)

theView=av.GetProject.FindDoc("Basemap")

mTheme=self.get(1).get(1)
strPol=self.get(2).get(1)

strPol=strPol.left(strPol.Count-2)

pList=List.Make
strList=strPol.AsTokens(";;")
for each i in 0..(strList.Count-1)
   mL2=strList.get(i).AsTokens("@@")
   mx=mL2.get(0).AsNumber
   my=mL2.get(1).AsNumber
   pList.add(mx@my)
end

mPol=polygon.make({pList})
themesList = {mtheme,mtheme+"_e"}

wl.WriteString("<CENTER>"+CR+NL)
wl.WriteString("<FONT SIZE=5> Dataset:<B> "+mTheme+"</B></FONT><BR>"+CR+NL)
k=0

fieldsList = List.Make

for each t in themesList
  mt=theView.FindTheme(t)
  if (mt<>Nil) then
    mFT=mt.GetFTab
    mFT.SelectByPolygon(mPol,#VTAB_SELTYPE_NEW)
    nRecs = mFT.getSelection.Count
    if (nRecs = 0) then
      continue
    end
    wl.WriteString("<FONT SIZE=3> N°"+nRecs.asString+"</FONT><BR>"+CR+NL)
    ml=av.run("theme.checkDataLocation",{t})
    if (ml = "nil") then
      wl.writeString("No data associated"+CR+NL)
      return "end"
    elseif (ml = "local") then
      if (k=0) then
        wl.WriteString("<TABLE border=2><TR>"+CR+NL)
        fieldList = mFT.getFields
        for each f in fieldList
        if (f.isVisible) then
          wl.WriteString("<TD><B>"+f.getAlias+CR+NL)
        end
      end
    else
      wl.WriteString("<TR>"+CR+NL)
    end
    for each r in mFT.GetSelection
      wl.WriteString("<TR>"+CR+NL)
      for each f in fieldList
        if (f.isVisible) then
          wl.WriteString("<TD>"+mFT.ReturnValueString(f,r)+CR+NL)
        end
      end
    end
  '------- for remote data -------
  else
    res=av.run("theme.getAttributes",{t,accessLevel})
    strSQL="select "+res.get(3)+" from "+res.get(2)+" where "+res.get(0).get(0)+" in ("
    str=""
    f=mFT.GetFields.Get(1)
    for each r in mFT.GetSelection
      str=str+"'"+mFT.ReturnValueString(f,r)+"',"
    end
    str=str.left(str.count-1)
    strSQL=strSQL+str+")"
    wl.WriteString("<TABLE border=2><TR>"+CR+NL)
    for each a in res.get(1)
      wl.WriteString("<TD><B>"+a+CR+NL)
    end
    'msgbox.report(strSQL,"OLA")
    theSQL=SQLCon.Find("Oracle")
    theSQL.Login("jlopes/jlopes@geomist")
    mVT=VTab.MakeSQL(theSQL, strSQL)

    for each r in mVT
      wl.writeString("<TR>"+CR+NL)
      for each f in mVT.GetFields
        wl.writeString("<TD>"+mVT.ReturnValueString(f,r)+CR+NL)
      end
    end
    theSQL.logout
  end
  mt.ClearSelection
 end
 k=k+1

end
wl.WriteString("</TABLE>"+CR+NL)

return true



'java.SelectByArea
wl = Weblink.the

theView=av.GetProject.FindDoc("Basemap")

mCovers=self.Get(1).Get(1)
```

```
mExtent=self.get(2).get(1)
mTheme=self.get(4).get(1)
strPol=self.get(5).get(1)

'wl.WriteString("("+x.asString+";"+y.asString+")<P>"+CR+NL)
strPol=strPol.left(strPol.Count-2)

'msgbox.report("0="+self.get(0).Get(1)+NL+self.get(1).Get(1)+NL+self.get(2).Get(1)+NL+self.get(3).G
et(1)+NL+self.get(4).Get(1)+NL,mTheme)
pList=List.Make
strList=strPol.AsTokens(";;")
for each i in 0..(strList.Count-1)
  mL2=strList.get(i).AsTokens("@@")
  mx=mL2.get(0).AsNumber
  my=mL2.get(1).AsNumber
  pList.add(mx@my)
end

mPol=polygon.make({pList})

av.run("java.drawMap",{mCovers,mExtent})

sTheme=theView.FindTheme(mTheme)
if (sTheme<>Nil) then
 sTheme.SelectByPolygon(mPol,#VTAB_SELTYPE_NEW)
 sTheme.SetVIsible(true)
 sTheme_e=theView.FindTheme(mTheme+"_e")
 if (sTheme_e <> Nil) then
  sTheme_e.SelectByPolygon(mPol,#VTAB_SELTYPE_NEW)
  sTheme_e.SetVIsible(true)
 end
else
 msgbox.info("Erro ","")
end
theHeight=350
theWidth=theHeight*1.565217
theDisplay=theView.getDisplay
tmpFN = theView.ExportToJPEG(theDisplay.returnExtent,theWidth,theHeight,85,false)
wl.WriteResponseHeader("Content-type: image/jpeg"+CR+NL+CR+NL)
wl.Writefile(tmpFN)
file.delete(tmpFN)
sTheme.clearSelection
sTheme.SetVIsible(false)
if (sTheme_e <> Nil) then
 sTheme_e.clearSelection
 sTheme_e.SetVIsible(false)
end
sTheme.ClearSelection

for each t in theView.GetVisibleThemes
 t.SetVisible(false)
end

return true



'Name:    java.SelectDataByAttribute
'
'Author:  Jorge Lopes <jlopes@igm.pt>
```

```
' Creation: 20/07/99
' Update:   20/07/99
'
' IN:      (dataset : String ;
'          query : String ;
'          sessionID : String
'          )
'
' OUT:      table : HTML Document
'
'


mdset=self.get(0).get(1)
mOpt=self.get(1).get(1)
mSession=self.get(2).get(1)
'msgbox.report("Dataset="+mdset+nl+"Opt="+mOpt+nl,mSession)

wl = Weblink.the
wl.WriteResponseHeader("Content-type: text/html"+CR+NL+CR+NL)
wl.WriteString("<TITLE> Geomist - Select by attribute</TITLE>"+CR+NL)
wl.WriteString("<HTML>"+CR+NL)
wl.WriteString("<BODY BGCOLOR=""#ffffff"">"+CR+NL)

accessLevel=av.run("java.CheckSession",{mSession})
if (accessLevel < 1) then
 wl.WriteString("Error: Time out"+CR+NL)
 return "end"
end

ml=av.run("theme.checkDataLocation",{mdset})

if (ml = "nil") then
  wl.writeString("No data associated"+CR+NL)
  return "end"
elseif (ml = "local") then
  theView = av.GetProject.FindDoc("Basemap")
  mTheme = theView.FindTheme(mdset)
  mFT = mTheme.GetFTab
  str=""
  for each f in mFT.getFields
     if (f.GetType = #FIELD_CHAR) then
     str = str + "["+f.GetName+"].contains("""+mOpt+""") or "
    end
  end
  str = str.left(str.count-3)
  'msgbox.report(str,mdset)
  theBitmap = mFT.GetSelection
  mFT.Query(str,theBitmap,#VTAB_SELTYPE_NEW)
  mFT.UpdateSelection
  wl.WriteString("<CENTER>")
  wl.WriteString("<FONT SIZE=4> Dataset "+mdset+"<BR>")
  if (mFT.GetNumSelRecords < 1) then
     wl.WriteString("No features selected"++CR+NL)
  else
    wl.WriteString("<FONT SIZE=3> N° = "+mFT.GetNumSelRecords.AsString++CR+NL)
    wl.WriteString("<TABLE BORDER=2><TR>")
    for each f in mFT.GetFields
      if (f.isVisible) then
        wl.WriteString("<TD><B>"+f.getAlias+CR+NL)
      end
  end
  for each r in mFT.GetSelection
   wl.WriteString("<TR> "+CR+NL)
   for each f in mFT.getFields
    if (f.isVisible) then
     wl.WriteString("<TD> "+mFT.ReturnValueString(f,r)+CR+NL)
    end
   end
  end
 end
 wl.WriteString("</TABLE> ")
 mTheme.ClearSelection
else
 mVT=av.getProject.FindDoc("oracle.tables").getVTab
 themeField=mVT.FindField("Theme")
 aliasName=mVT.FindField("aliasName")
 oraField=mVT.FindField("oraField")
 acLevel=mVT.FindField("acLevel")

 strSQL = "Select "
 first=true

 whereStr=""
 fieldList = List.Make
 for each f in mVT
  t=mVT.returnValue(themeField,f)
  v=mVT.returnValue(acLevel,f)
  if ((t=mdset) and (v <= accessLevel)) then
   fld=mVT.ReturnValue(oraField,f)
   mView=mVT.ReturnValue(mVT.GetFields.get(3),f)
   strSQL = strSQL+fld+", "
   whereStr=whereStr+" upper("+fld+") like '%"+mOpt.ucase+"%' or"
   fieldList.add(mVT.ReturnValue(mVT.GetFields.get(2),f))
  end
 end

 strSQL = strSQL.left(strSQL.count-2)
 whereStr = whereStr.left(whereStr.count-2)
 strSQL=strSQL + " from " + mView + " where "+whereStr
 'msgbox.report(strSQL,"OLA")
 theSQL=SQLCon.Find("Oracle")
 theSQL.Login("jlopes/jlopes@geomist")
 mVT=VTab.MakeSQL(theSQL, strSQL)
 nRecs=mVT.GetNumRecords
 wl.WriteString("<CENTER>")
 wl.WriteString("<FONT SIZE=5> Dataset "+mdset+"</FONT><BR>")
 if (nRecs < 1) then
  wl.WriteString("No features selected"++CR+NL)
  return 0
 end
 wl.WriteString("<FONT SIZE=3> Nº = "+nRecs.AsString++CR+NL)
 wl.WriteString("<TABLE BORDER=2><TR>")

 for each f in fieldList
  wl.WriteString("<TD><B> "+f+CR+NL)
 end
 for each r in mVT
  wl.WriteString("<TR> "+CR+NL)
  for each f in mVT.GetFields
   wl.WriteString("<TD> "+mVT.ReturnValueString(f,r)+CR+NL)
   end
  end
  wl.WriteString("</TABLE>")
  wl.WriteString("</HTML>")
  theSQL.logout
end

'Name:    java.SelectMapByAttribute
'
'Author:  Jorge Lopes <jlopes@igm.pt>
'Creation: 20/07/99
'Update:   20/07/99
'
'IN:      (dataset : String ;
'          query : String ;
'          mCovers : List ;
'          mExtent : List ;
'          sessionID : String
'          )
'
'OUT:     table : HTML Document
'
'


mdset=self.get(0).get(1)
mOpt=self.get(1).get(1)
mCovers=self.get(2).get(1)
mExtent=self.get(3).get(1)
mSession=self.get(4).get(1)

'Check entry data
'str=""
'for each x in self
'  str=str+x.get(0)+" = "+x.get(1)+nl
'end
'msgbox.report(str,"SELF")

wl = Weblink.the
wl.WriteResponseHeader("Content-type: image/jpeg"+CR+NL+CR+NL)

accessLevel=av.run("java.CheckSession",{mSession})
if (accessLevel < 1) then
 wl.Writefile("/atlas/geomist/tmp/error.jpg".asFileName)
 return "end"
end

ml=av.run("theme.checkDataLocation",{mdset})

if (ml = "nil") then
  wl.writeString("No data associated"+CR+NL)
  return "end"
elseif (ml = "local") then
  theView = av.GetProject.FindDoc("Basemap")
  theTheme = theView.FindTheme(mdset)
  mFT = theTheme.GetFTab
  str=""
  for each f in mFT.getFields
     if (f.GetType = #FIELD_CHAR) then
     str = str + "["+f.GetName+"].contains("""+mOpt+""") or "
    end
  end

  str = str.left(str.count-3)
  theBitmap = mFT.GetSelection
  mFT.Query(str,theBitmap,#VTAB_SELTYPE_NEW)
  mFT.UpdateSelection
  theFTab=mFT

elseif (ml="remote") then
  res=av.run("theme.getAttributes",{mdset,accessLevel})
  strSQL="select "+res.get(0).get(0)+" from "+res.get(2)+" where "
  str=""
  for each r in res.get(0)
    str=str+"upper("+r+")"+" like '%"+mOpt.ucase+"%' or "
  end
  str=str.left(str.count-3)
  strSQL=strSQL+str
  'msgbox.report (strSQL,"test")
  theSQL=SQLCon.Find("Oracle")
  theSQL.Login("jlopes/jlopes@geomist")
  mVT=VTab.MakeSQL(theSQL, strSQL)
  'msgbox.report (strSQL,mVT.GetNumRecords.asString)

  theView=av.GetProject.FindDoc("Basemap")
  theDisplay=theView.getDisplay
  theTheme=theView.FindTheme(mdset)
  theFTab=theTheme.GetFTab
  fieldName=mVT.GetFields.Get(0).GetName
  mF0=mVT.GetFields.Get(0)
  mF0.setAlias("aux")
  mF1=theFTab.GetFields.Get(1)
  theFTab.join(mF1,mVT,mF0)
  theBitmap=theFTab.GetSelection

  expr="([aux].isNull.Not)"
  theFTab.Query(expr,theBitmap,#VTAB_SELTYPE_NEW)
  theFTab.UpdateSelection
  theSQL.logout
end

av.run("java.drawMap",{mCovers,mExtent})
theTheme.SetVisible(true)
theDisplay=theView.getDisplay
theHeight=350
theWidth=theHeight*1.565217
tmpFN = theView.ExportToJPEG(theDisplay.returnExtent,theWidth,theHeight,85,false)
wl.Writefile(tmpFN)
file.delete(tmpFN)
for each t in theView.GetVisibleThemes
  t.SetVisible(false)
end
theTheme.ClearSelection
theFTab.UnjoinAll
return true

'Name:    theme.checkDataLocation
'
'Author:  Jorge Lopes <jlopes@igm.pt>
'Creation: 21/07/99
'Update:   21/07/99
```

```
'
'IN:      (theme : String)
'
'OUT:     table : HTML Document
'
'

mdset=self.get(0)

'------ Check for alphanumeric data location --------------
ml = "nil"
mVT=av.getProject.FindDoc("alphaData").getVTab
themeField=mVT.FindField("Theme")
locationField=mVT.FindField("Location")
for each t in mVT
  mt = mVT.ReturnValue(themeField,t)
  if (mt = mdset) then
    ml =  mVT.ReturnValue(locationField,t)
    break
  end
end

return ml

'Name:     theme.getAttributes
'
'Author:  Jorge Lopes <jlopes@igm.pt>
'Creation: 21/07/99
'Update:   21/07/99
'
'IN:      (theme : String)
'
'OUT:     (fields : List ;
'          alias :  List ;
'          view : String ;
'          fieldsString : String
'          )
'
'

mdset=self.get(0)
accessLevel=self.get(1)
'msgbox.info(mdset, "")

mVT=av.getProject.FindDoc("oracle.tables").getVTab
themeField=mVT.FindField("Theme")
aliasName=mVT.FindField("aliasName")
oraField=mVT.FindField("oraField")
acLevel=mVT.FindField("acLevel")

fieldsList=List.Make
aliasList=List.Make
strFields=""

for each f in mVT
  t=mVT.returnValue(themeField,f)
  v=mVT.returnValue(acLevel,f)
  if ((t=mdset) and (v <= accessLevel)) then
    mView=mVT.ReturnValue(mVT.GetFields.get(3),f)
    fieldsList.add(mVT.ReturnValue(oraField,f))
    strFields=strFields+mVT.ReturnValue(oraField,f)+","
    aliasList.add(mVT.ReturnValue(aliasName,f))
  end
end

strFields=strFields.left(strFields.count-1)
res=List.Make
res.add(fieldsList)
res.add(aliasList)
res.add(mView)
res.add(strFields)
return res
```

3. CLIENT JAVA APPLET

On the client side there is a Java Applet able to interact with the user through a spatial interface.

```
/*
 * name: JMapXX.java
 *
 * Author:                    Jorge Lopes
 * EMail:                     jlopes@igm.pt
 * Creation date: 21/02/1999
 * Last update:          19/07/1999
 *
 * All rights reserved
 */

import java.applet.*;
import java.applet.Applet;
import java.awt.*;
import java.awt.image.*;
import java.net.*;
import java.util.Vector;
import java.util.Date;
import java.lang.Math;
import java.io.*;

class ButtonPanel extends Panel{
        ButtonPanel (String id){
                setLayout(new BorderLayout());
                add("Center", new Button(id));
        }
}

/*
 * JMap Class
 * An interactive interface to manipulate maps (as images) loaded
 * dynamicaly from a GIS application
 *
 */
public class JMap45 extends Applet{
        // Map returned from GIS as an image
        Image img;
        Toolkit tk=Toolkit.getDefaultToolkit();
        String docDir="";
        //Data about user graphics drawn over map
        Vector m_vLocs;
Dimension m_dimCursorLoc;
        boolean inDraw=false;
        boolean inPanZoom=false;
        boolean idIdentify=false;

        //Interface components
        private Label lbl1=new Label(" ");
        private Label lbl2=new Label(" ");
        Label themes_lbl=new Label("Spatial Data");
        //Choice c=new Choice();

        Object o;
        private Label tf=new Label("00000000;00000000");
        private TextField auxtf=new TextField("aaaaaaaaaaaaaaaaaaaa");
        private TextField restf=new TextField(30);
        List resl=new List(3,false);

        CheckboxGroup jAction=new CheckboxGroup();
        Checkbox identifyCB=new Checkbox("Identify",jAction,true),
                        panCB=new Checkbox("Pan",jAction,false),
                        zoomInCB=new Checkbox("Zoom In",jAction,false),
                        zoomOutCB=new Checkbox("Zoom Out",jAction,false),
                        drawCB=new Checkbox("Querying Data",jAction,false);


        Choice zoomChoice=new Choice();
        Choice identifyChoice=new Choice();
        Choice drawChoice=new Choice();
        Choice queryChoice=new Choice();

        private Button drawBtn=new Button("Draw");
        private Button resetBtn=new Button("Reset");
        private Button executeBtn=new Button("Execute");

        Vector jCoversList=new Vector();

        //Auxiliar variables
        String str="";
        String fname="";
        String sessionID="";

        //String variables to get parameters data
        final String PARAM_fname="fname";
        final String PARAM_coversID="coversID";
        final String PARAM_coversList="coversList";
        final String PARAM_baseData="baseData";

        final String PARAM_sessionID="sessionID";

        //Data structures to store data about map
        Dimension shiftImg=new Dimension(130,65);                    //Image shift
        Dimension imageCorner=new Dimension(500000,4113000);
        //Dimension bottomRight=new Dimension(752000,4113000);
        Dimension mapExtent=new Dimension(252000,161000);
        Dimension wc;

        //Layout data
        Panel cards=new Panel();
```

```
    CardLayout cl=new CardLayout();
    float scale=460;

    /*
     * Constructor
     */
    public JMap45(){
            setBackground(Color.white);
m_vLocs = new Vector();
m_dimCursorLoc = new Dimension(0, 0);
            inDraw=false;
            inPanZoom=false;
            idIdentify=true;
    }


    public String[][] getParameterInfo(){
            String[][] info={{PARAM_fname,"String","File name"},
                                            {PARAM_coversID,"String","Covers to ID"},
                                            {PARAM_coversList,"String","Covers to view"},
                                            {PARAM_baseData,"String","Base map data"},
                                            {PARAM_sessionID,"String","Session ID"}};
            return info;
    }

    public void init(){
            setLayout (new BorderLayout(5,5));
            Panel pLeft=new Panel();
            pLeft.setLayout(new GridLayout(15,1));
            Panel pTop=new Panel();
            pTop.setLayout(new GridLayout(2,2,10,10));
            Panel pBottom=new Panel();
            pBottom.setLayout(new GridLayout(1,5));


            String aux=getParameter(PARAM_fname);
            if (aux != null){
                    fname=aux;
                    img=getImage(getDocumentBase(),fname);
            }

            sessionID=getParameter(PARAM_sessionID);

            identifyChoice.addItem("--Dataset--");
            drawChoice.addItem("--Dataset--");
            String aux2=getParameter(PARAM_coversList);
            pLeft.add(new Label("Datasets"));
            Checkbox cb;
            int k=0;
            while (true){
                    if (aux2.indexOf("& ")>0){
                            String aux3=aux2.substring(0,aux2.indexOf("& "));
                            aux2=aux2.substring(aux2.indexOf("& ")+1);
                            if (k>3 && k<7){
                                    cb=new Checkbox(aux3,null,true);
                            }
                            else
                                    cb=new Checkbox(aux3,null,false);
                            pLeft.add(cb);
                            jCoversList.addElement(cb);
```

```
                identifyChoice.addItem(aux3);
                drawChoice.addItem(aux3);
            }else break;
            k++;
        }

        queryChoice.addItem("--Operation--");
        queryChoice.addItem("Select by area");
        queryChoice.addItem("Select by attributes");

        zoomChoice.addItem("2  x");
        zoomChoice.addItem("4  x");
        zoomChoice.addItem("8  x");
        zoomChoice.addItem("16 x");
        zoomChoice.addItem("Full extent");


        zoomChoice.hide();
        drawChoice.hide();
        queryChoice.hide();
        resetBtn.hide();
        executeBtn.hide();

        pTop.add(identifyCB);
        pTop.add(panCB);
        pTop.add(zoomInCB);
        pTop.add(zoomOutCB);
        pTop.add(drawCB);
        pTop.add(lbl2);


        pTop.add(lbl1);
        pTop.add(identifyChoice);
        //pTop.add(lbl1);
        pTop.add(auxtf);
        auxtf.hide();
        pTop.add(zoomChoice);
        //pTop.add(lbl1);
        pTop.add(resetBtn);

        pTop.add(drawChoice);
        pTop.add(queryChoice);
        //pTop.add(c);
        pTop.add(executeBtn);

        pLeft.add(new Button("Redraw Map"));

        pBottom.add(tf);
        //tf.setEditable(false);
        //tf.setBackground(Color.lightGray);
        String[] fontList=tk.getFontList();
        tf.setFont(new Font(fontList[0],Font.PLAIN,5));
        pBottom.add(restf);

        add("North",pTop);
        add("West",pLeft);
        add("South",pBottom);
        //resl.hide();
        repaint();
    }
```

```
        public void destroy(){
        }

        public void repaint (long tm, int x, int y,int w,int h){
                super.repaint(tm,x,y,w,h);
        }

        public void identifyShow(){
                zoomChoice.hide();
                drawChoice.hide();
                queryChoice.hide();
                resetBtn.hide();
                executeBtn.hide();
                identifyChoice.show();
        }
        public void zoomShow(){
                zoomChoice.show();
                drawChoice.hide();
                queryChoice.hide();
                resetBtn.hide();
                executeBtn.hide();
                identifyChoice.hide();
        }
        public void panShow(){
                zoomChoice.hide();
                drawChoice.hide();
                queryChoice.hide();
                resetBtn.hide();
                executeBtn.hide();
                identifyChoice.hide();
        }
        public void drawShow(){
                zoomChoice.hide();
                drawChoice.show();
                queryChoice.show();
                resetBtn.show();
                executeBtn.show();
                identifyChoice.hide();
        }


        public void paint(Graphics g){
      // Coloca uma cruz no ponto de aplocacao

                if (!g.drawImage(img,shiftImg.width,shiftImg.height,this)){
                        g.drawString("Loading Image. Please hang on...",250,150);
                }else{
                        if (img.getWidth(this) < 400){
                                try{
                                        AppletContext ac=getAppletContext();
                                        String
mStr="http://geomist.igm.pt/tmp/geomist/timeout.html";
                                        ac.showDocument(new URL(mStr),"_top");
                                }catch (Exception e){
                                        System.err.println(e);
                                }
```

```
            }

            int nX = m_dimCursorLoc.width;
            int nY = m_dimCursorLoc.height;
            g.drawLine(nX-2,nY, nX + 2,nY);
            g.drawLine(nX, nY - 2,     nX, nY + 2);
            int imgHeight=img.getHeight(this);
            //scale=leftTop.img.getHeight(this)/

            //g.drawString(str,0,290);
            Dimension dimFrom;
            Dimension dimTo;
            int nSize = m_vLocs.size();
            for (int i = 0; i < nSize - 1; i++){
                dimFrom = (Dimension)m_vLocs.elementAt(i);
                dimTo = (Dimension)m_vLocs.elementAt(i+1);
                g.drawLine(dimFrom.width, dimFrom.height,
       dimTo.width,  dimTo.height);
                if (i == nSize - 2){
                    dimFrom = (Dimension)m_vLocs.elementAt(0);
                    g.drawLine(dimFrom.width, dimFrom.height,
       dimTo.width,  dimTo.height);
                }
            }
        }
    }

    public void start(){
    }

    public void stop(){
    }
    public void jIdentify(String t, int x, int y){
        URL u;
        String strId;
        //restf.setText("");
        //resl.delItems(0,resl.countItems()-1);
        wc= imgToWC(x,y);

        try{
            AppletContext ac=getAppletContext();
            String
mStr="http://geomist.igm.pt/.esrimap?name=Geomistmap&cmd=java.getIdentify&Theme="+t+"&Locat
ion="+wc.width+"@"+wc.height+"&sessionID="+sessionID;
            ac.showDocument(new URL(mStr),"showResults");
        }catch (Exception e){
            System.err.println(e);
        }
    }
    public Dimension imgToWC(int x, int y){
        Dimension mWC=new Dimension((int)(imageCorner.width+(x-shiftImg.width)*scale),

    (int)(imageCorner.height+(img.getHeight(this)-(y-shiftImg.height))*scale));
        return mWC;

    }
    public boolean onImg(int x, int y){
        return (x>shiftImg.width && y>shiftImg.height &&
                    x<(shiftImg.width+img.getWidth(this)) &&
                    y<(shiftImg.height+img.getHeight(this)));
        }
        public void jPanZoom(float nTimes,int x, int y){

                String tx = (int) imageCorner.width+";"+(int) imageCorner.height+">"+(int)scale+" -
>"+x+";"+y+">"+(int)nTimes+" -> ";
                float scaleNext=scale/nTimes;
                wc= imgToWC(x,y);

                if (nTimes==-1){
                        imageCorner.width=500000;
                        imageCorner.height=4113000;
                        mapExtent.width=252000;
                        mapExtent.height=161000;
                        scale = 460;
                }else{
                        imageCorner.width=(int)(wc.width-(img.getWidth(this)/2)*scaleNext);
                        imageCorner.height=(int)(wc.height-(img.getHeight(this)/2)*scaleNext);

                        mapExtent.width=(int) (img.getWidth(this)*scaleNext);
                        mapExtent.height=(int) (img.getHeight(this)*scaleNext);
                        scale=scaleNext;
                }

                String
str=".esrimap?name=GeomistMap&cmd=java.RedrawMap"+getCovers()+getExtent()+"&sessionID="
+sessionID;
                img=getImage(getDocumentBase(),str);

                repaint();
        }
        public boolean mouseDown(Event evt, int x, int y){
    // in the event of a double click...
                // cli
                int i=0;
    if (evt.clickCount == 1){
                        if (onImg(x,y)){
                                if(inDraw)
                                        m_vLocs.addElement(new Dimension(x, y));
                                else if(inPanZoom){
                                        float nTimes=2;
                                        if (panCB.getState())
                                                nTimes=1;
                                        else{
                                                i=zoomChoice.getSelectedIndex()+1;
                                                if (i==1)
                                                        nTimes=2;
                                                else if (i==2)
                                                        nTimes=4;
                                                else if (i==3)
                                                        nTimes=8;
                                                else if (i==4)
                                                        nTimes=16;
                                                else if (i==5) {
                                                        nTimes=-1;
                                                }
                                        }
                                        if(zoomOutCB.getState())
                                                nTimes=1/nTimes;
                                        jPanZoom(nTimes,x,y);
```

```
                                     }else if(idIdentify){
                                              String
mTheme=identifyChoice.getItem(identifyChoice.getSelectedIndex());
                                              jIdentify(mTheme,x,y);
                                     }
                            }
     }else if (evt.clickCount == -1){
        // ...otherwise, add an element
                            m_vLocs.removeAllElements();
                            inDraw=false;

     }
                  else if(inDraw) inDraw=false;

     repaint();
                  return true;
         }

         public boolean mouseUp(Event evt, int x, int y){
     // ignore the mouseUp event
     return true;
         }

         public boolean mouseDrag(Event evt, int x, int y)
         {
     // ignore the mouseDrag event
                  return true;
         }

         public boolean mouseMove(Event evt, int x, int y){
                  if (onImg(x,y)){
                            wc= imgToWC(x,y);
                            tf.setText(wc.width+";"+wc.height);
                  }
                  auxtf.setText(x+";"+y+">"+scale);
                  return true;
         }

         public boolean imageUpdate(Image img, int flags, int x, int y, int width, int height){
                  if ((flags & ImageObserver.ALLBITS) != 0)
                                     repaint();
                  return true;
         }

         public String getCovers(){
                  String str="&CoversList=";
                  Checkbox cb;
                  //identifyChoice=new Choice();
                  for (int i=0; i<jCoversList.size(); i++){
                            cb=(Checkbox) jCoversList.elementAt(i);
                            if (cb.getState()){
                                     str=str+cb.getLabel()+"__";
                                     //identifyChoice.addItem(cb.getLabel());
                            }
                  }
                  return str;
         }

         public String getExtent(){
                  String str="&newExtent="+imageCorner.width+"_"+imageCorner.height+"_"+
                                                                mapExtent.width+"_"+mapExtent.height;
            return str;
        }

        public boolean action(Event evt, Object obj){
            Object oTarget=evt.target;
            if(oTarget instanceof Button){
                Button btnTarget = (Button) oTarget;
                Button x=(Button) o;

                String strBtnTarget=btnTarget.getLabel();
                if (strBtnTarget.compareTo("Reset")==0){
                    m_vLocs.removeAllElements();
                    inDraw=false;
                    str="";
                    repaint();
                    //identifyCB.setState(true);
                }else if(strBtnTarget.compareTo("Execute")==0){
                    int nSize = m_vLocs.size();
                    if((drawChoice.getSelectedIndex())>0){
                        String
dataset=drawChoice.getItem(drawChoice.getSelectedIndex());
                        if ((queryChoice.getSelectedIndex())>0 ){
                            if ((queryChoice.getSelectedIndex())==1 ){ // Select
by area

        str=".esrimap?name=GeomistMap&cmd=java.SelectByArea&Area="+getCovers()+getExtent()
+"&sessionID="+sessionID;
                                str=str+"&Dataset="+dataset;

                                String mPol="&Polygon=";
                                for (int i = 0; i < nSize; i++){
                            Dimension loc = (Dimension)m_vLocs.elementAt(i);
                                    Dimension                    wc=
imgToWC(loc.width,loc.height);

        mPol=mPol+wc.width+"@@"+wc.height+";;";
                                }
                                str=str+mPol;
                                img=getImage(getDocumentBase(),str);
                                try{
                                    AppletContext
ac=getAppletContext();
                                    String
mStr="http://geomist.igm.pt/.esrimap?name=Geomistmap&cmd=java.getSelectDataByArea";

        mStr=mStr+"&sessionID="+sessionID+"&Dataset="+dataset+mPol;
                                    ac.showDocument(new
URL(mStr),"showResults");
                                }catch (Exception e){
                                    System.err.println(e);
                                }
                                repaint();
                            }else{

                                String opt=restf.getText().trim();
                                String
strMap="http://geomist.igm.pt/.esrimap?name=GeomistMap&cmd=java.SelectMapByAttribute&Dataset
="
```

```
        +dataset+"&opt="+opt+getCovers()+getExtent()
                                                            +"&sessionID="+sessionID;

        img=getImage(getDocumentBase(),strMap);
                                                      try{

        str="http://geomist.igm.pt/.esrimap?name=GeomistMap&cmd=java.SelectDataByAttribute&Da
taset="+dataset+"&opt="+opt+"&sessionID="+sessionID;
                                                            AppletContext
ac=getAppletContext();
                                                            ac.showDocument(new
URL(str),"showResults");
                                                      }catch (Exception e){
                                                            System.err.println(e);
                                                      }
                                                }
                                          }
                                    }
                                    repaint();
                              }else if(strBtnTarget.compareTo("Redraw Map")==0){

        str=".esrimap?name=GeomistMap&cmd=java.RedrawMap"+getCovers()+getExtent()+"&sessi
onID="+sessionID;;

                                    img=getImage(getDocumentBase(),str);
                                    drawBtn.setBackground(Color.lightGray);
                                    drawBtn.setForeground(Color.orange);
                                    repaint();
                              }
                        }else if(oTarget instanceof Checkbox){
                              Checkbox cb=(Checkbox) oTarget;
                              if (cb==identifyCB){
                                    inPanZoom=false;
                                    inDraw=false;
                                    idIdentify=true;
                                    m_vLocs.removeAllElements();
                                    resl.delItems(0,resl.countItems()-1);
                                    identifyShow();
                              }else if (cb==panCB){
                                    idIdentify=false;
                                    inPanZoom=true;
                                    inDraw=false;
                                    m_vLocs.removeAllElements();
                                    panShow();
                              }else if (cb==zoomInCB){
                                    idIdentify=false;
                                    inPanZoom=true;
                                    inDraw=false;
                                    m_vLocs.removeAllElements();
                                    zoomShow();
                              }else if (cb==zoomOutCB){
                                    idIdentify=false;
                                    inPanZoom=true;
                                    inDraw=false;
                                    m_vLocs.removeAllElements();
                                    zoomShow();
                              }else if (cb==drawCB){
                                    idIdentify=false;
                                    inPanZoom=false;
```

```
                inDraw=true;
                drawShow();
            }else{ .
                drawBtn.setBackground(Color.orange);
                drawBtn.setForeground(Color.orange);
                repaint();

            }
            //switch

        }//else if
        return true;
    } // action


}
```

ANNEX 7

SYSTEN SECURITY

SOURCE CODE

SOURCE CODE

The Geomist security system is embedded into the implementation of each software component used to grant access control to data and applications. This inter-operability between processes at remote locations raises some complexity and requires a global view of the system.

1. CGI WEB SERVER

The first user interaction is required from the CGI Web server machine through the execution of internal procedures to provide an appropriate interface.

```
#!/usr/netscape/suitespot/install/perl

$i=0;
$access=0;
print "Content-type: text/html\n\n";
#Verificamos se o utilizador tem uma sessao valida
$mIP=&getParameterIN("REMOTE_HOST");
if($ENV{'REQUEST_METHOD'} eq "GET"){
        $in = $ENV{'QUERY_STRING'};
}
elsif($ENV{'REQUEST_METHOD'} eq "POST"){
  read(STDIN,$in,$ENV{'CONTENT_LENGTH'});
}
$username=&jGetParameter("username",$in);
$password=&jGetParameter("password",$in);
$service=&jGetParameter("service",$in);

$access_level=&authAgent($username,$password);


#Neste momento pode ou nao ter sessao
if ($access_level<1){
  #Ainda nao tem sessao
  #Volta a pedir identificacao
  print "<HTML>\n";
  print "<TITLE>Geomist - User identification</TITLE>\n";
  print "<BODY BGCOLOR=#FFFFFF></P>\n";
  print "<TABLE><TR><TD><IMG SRC=\"/geomist/images/g_logo3.gif\"><TD><B><FONT
SIZE=+3>Geological and Mining Information System</FONT></B><BR><FONT SIZE=+2>Iberian
Pyrite Belt</TABLE>\n";
  print "<CENTER>\n";
  print "<HR NOSHADE WIDTH=60%><BR>\n";
  print "<FONT SIZE=5>User Identifier<BR></font>\n";
  if ($username ne "anonymous") {
        print "(Error: Try again please)<br>";
  }
  print "<HR NOSHADE WIDTH=60%><BR>\n";
  print "<TABLE BORDER=1>\n";
  print "<FORM ACTION='/cgi/auth_agent.pl' METHOD='POST'><BR>\n";
  print "<TR><TD ALIGN=CENTER><FONT SIZE=4>Username:<INPUT name='username' value=''
size='9'>\n";
  print "<TR><TD ALIGN=CENTER><FONT SIZE=4>Password:<INPUT name='password'
type='password' value='' size='9'>\n";
  print "<INPUT TYPE='hidden' name='service' value='$service'>\n";
  print "<TR><TD ALIGN=CENTER><FONT SIZE=4><INPUT type='submit' value='OK'>\n";
  print "<INPUT type='reset' value='Clear'>\n";
  print "</FORM>\n";
  print "</TABLE>\n";
  print "<HR NOSHADE WIDTH=60%><BR>\n";
  print "<MAP NAME=\"subscribeMap\">\n";
  print "<AREA SHAPE=RECT COORDS=\"19,2,60,16\" HREF=\"/\">\n";
  print "<AREA SHAPE=RECT COORDS=\"91,2,152,16\" HREF=\"/geomist/subscribe.html\">\n";
  print "<AREA SHAPE=RECT COORDS=\"190,2,258,16\" HREF=\"mailto:torres\@igm.pt\">\n";
  print "</MAP>\n";
  print "<IMG SRC=\"/geomist/images/subscribeBar.jpg\" BORDER='0' USEMAP=\"\#subscribeMap\"
ISMAP>\n";
  print "</HTML>\n";
}else{
  $mStr = `java jClient20 543212345 $username`;
  $res = &jGetParameter("result",$mStr);
  if($res eq "OK"){
     $mSession = &jGetParameter("sessionID",$mStr);
         $mLevel = &jGetParameter("sessionLevel",$mStr);
         print "<HTML\n>";
         print "<TITLE>Geomist Services</TITLE>\n";
         print "<BODY BGCOLOR=#FFFFFF></P>\n";
         print "<TABLE><TR><TD><IMG SRC=\"/geomist/images/g_logo3.gif\"><TD><B><FONT
SIZE=+3>Geological and Mining Information System</FONT></B><BR><FONT SIZE=+2>Iberian
Pyrite Belt</TABLE>\n";
         print "<CENTER>\n";
         print "<P>\n";
         print "<HR NOSHADE WIDTH=60%><BR>\n";
     print "<FONT SIZE=5>Select your Service<BR>\n";
         print "<HR NOSHADE WIDTH=60%><BR>\n";

     print "<TABLE BORDER=0>\n";
         print "<TR><TD><A
HREF='/cgi/start_service2.pl?sessionID=$mSession&username=$username&service=metadata'><IMG
SRC='/geomist/images/geomistMetadata.jpg' ALT='Search Metadata' BORDER=0>\n";
         print "<TD><A
HREF='/cgi/start_service2.pl?sessionID=$mSession&username=$username&service=database'><IMG
SRC='/geomist/images/geomistDatabases.jpg' ALT='Search Metadata' BORDER=0>\n";
         print "<TD><A
HREF='/cgi/start_service2.pl?sessionID=$mSession&username=$username&service=mapping'><IMG
SRC='/geomist/images/geomistMapping.jpg' ALT='Search Metadata' BORDER=0>\n";
         print "</TABLE>\n";
         print "<HR NOSHADE WIDTH=60%><BR>\n";
         print "<MAP NAME=\"unsubscribeMap\">\n";
         print "<AREA SHAPE=RECT COORDS=\"19,2,58,18\" HREF=\"/\">\n";
         print "<AREA SHAPE=RECT COORDS=\"87,2,161,18\" HREF=\"/cgi/unsubscribe.pl\">\n";
         print "<AREA SHAPE=RECT COORDS=\"187,2,295,18\"
HREF=\"/cgi/changePassword.pl\">\n";
         print "<AREA SHAPE=RECT COORDS=\"318,2,370,16\"
HREF=\"mailto:torres\@igm.pt\">\n";
         print "</MAP>\n";
         print "<IMG SRC=\"/geomist/images/unsubscribeBar.jpg\" BORDER='0'
USEMAP=\"\#unsubscribeMap\" ISMAP>\n";

     print "</HTML>\n";
  }
}

###########################################################
# FUNCTION authAgent(2)
#
# CREATION DATE 23/02/1999
# LAST MODIFIED 23/03/1999
#
# This function receives the one username and the supposed password
# and returns the access level, if the username and password match,
# returns 0 if the  username and password mismatch.
# Otherwise returns -1
##------------------------------------------------------
sub authAgent{
        local($user)=$_[0];
        local($pass)=$_[1];
        local($ip)=&getParameterIN("REMOTE_HOST");

        local($i)=-1;

        if( ! open ( IN , "/disk2/httpd/access/passwd.txt")){
                print "Impossivel abrir ficheiro!\n";
        }else{
                #print "$user -> $pass <br>";
                while(($line = <IN>) && ($i<1)){
                        @all =split (/ /,$line);
                        #print "$all[0] -> $all[3] : $all[2]<br>";
                        if ($all[0] eq $user){
                                $i=0;
                                if ($all[3] eq $pass){
                                        $i=$all[2];
                                }
                        }
                }
        }
        $i;
}
########################################################

####################################################################
# FUNCTION getParameterIN(1)
#
# CREATION DATE 13/03/1999
# LAST MODIFIED 13/03/1999

# This function returns the variable from the current
# connection
##--
sub getParameterIN{
        local($result);
        local($param)=$_[0];

        foreach $element (keys %ENV){
                if ($element eq $param){
                        $result=$ENV{$element};
                }
        }
        $result;
}
###


sub jGetParameter{
        my $ret;
        my $arg=$_[0];
        my $in=$_[1];
        @all=split(/&/,$in);
        foreach $member(@all){
                @param=split(/=/,$member);
                if ($param[0] eq $arg){
                        $ret=$param[1];
                }
        }
        #print "QS=$in ARG=$arg RET=$ret\n";

        $ret;
}
```

2. DISPATCHER AGENT

After passing the identification process the server provides an interface with shortcuts for each service available.

```
#!/usr/netscape/suitespot/install/perl

$i=0;
$access=0;

print "Content-type: text/html\n\n";
#Verificamos se o utilizador tem uma sessao valida
$mIP=&getParameterIN("REMOTE_HOST");
if($ENV{'REQUEST_METHOD'} eq "GET"){
        $in = $ENV{'QUERY_STRING'};
}
elsif($ENV{'REQUEST_METHOD'} eq "POST"){
  read(STDIN,$in,$ENV{'CONTENT_LENGTH'});
}

$username=&jGetParameter("username",$in);
$sessionID=&jGetParameter("sessionID",$in);
$service=&jGetParameter("service",$in);

$mName ="/disk2/httpd/htdocs/temp/$sessionID.html";
#$mName=$path+"/"+$sessionID+".html";
open(TMPOUT,">$mName");

$mStr = `java jClient20 $sessionID $username`;
$res = &jGetParameter("result",$mStr);
if ($res eq "OK"){
  if ($service eq "database"){
        print "<HTML>";
        print "<HEAD>";
        print "<TITLE>GEOMIST - Query Databases</TITLE>";
        print "<frameset cols='*,205' border=2>";
        print "   <frameset rows='*,100' border=1>";
        print "                        <frame src='/geomist/inter_database/showtable.html'>";
        print "               <frame src='/geomist/inter_database/showdata.html'>";
        print "   </frameset>";
        print "       <frameset rows='*,60' border=0>\n";
        print      "                                                                <frame
src='/cgi/queryBD2.pl?cmd=xpto&dataset=Drillholes&sessionID=$sessionID'>";
```

```
        print "                                                        <frame
src='/cgi/return.pl?username=$username&sessionID=$sessionID&service=databases'>\n";
        print "      </frameset>\n";
        print "</frameset>";
        print "</BODY>";
        print "</HTML>";
  }elsif ($service eq "mapping"){
        print TMPOUT "<HTML>\n";
        print TMPOUT "<TITLE>Geomist - Interactive Mapping </TITLE>\n";
        print TMPOUT "<BODY BGCOLOR='#FFFFFF'
onLoad=\"window.open('','showResults','width=300,height=200,directories=no,location=no,menubar=n
o,resizable=yes,scrollbars=yes,status=no,toolbar=no')\">\n";
        print TMPOUT "<CENTER>\n";
        print TMPOUT "<FONT SIZE=5> <STRONG>Interactive
Mapping</FONT><P></STRONG>\n";
        print TMPOUT "<TABLE BORDER=4><tr><td>\n";
        print TMPOUT "<APPLET CODEBASE=\"/java\" CODE=\"JMap45.class\" width=680
height=500>\n";
        print TMPOUT "<PARAM NAME=fname
        VALUE=\"/.esrimap?name=GeomistMap&cmd=java.RedrawMap&tList=Topographic__FPI
Limits__1-50000&extent=500000_4113000_252000_161000&sessionID=$sessionID\">\n";
        print TMPOUT "<PARAM NAME=coversID VALUE=\"Drillholes&No
Metallic&Metallic&\">\n";
        print TMPOUT "<PARAM NAME=coversList VALUE=\"Drillholes&Drillholes Campaign&No
Metallic&Metallic&Topographic&1-50000&FPI Limits&Exploration Permits&Geological
Faults&Geological Synthesis&Gravimetric grid&Magnetic grid&Radiometric grid&\">\n";
        print TMPOUT "<PARAM NAME=baseData VALUE=\"Topographic&FPI Limits&1-
50000&\">\n";
        print TMPOUT "<PARAM NAME=sessionID VALUE=\"$sessionID\">\n";
        print TMPOUT "</APPLET>\n";
        print TMPOUT "</TABLE>\n";
        print TMPOUT "<A
HREF='/cgi/start_service2.pl?sessionID=$sessionID&username=$username&service=metadata'
TARGET='_top'><IMG SRC='/tmp/geomist/images/geomistMetadataMinW.jpg' ALT='Search Metadata'
BORDER=0>\n";
        print TMPOUT "<A
HREF='/cgi/start_service2.pl?sessionID=$sessionID&username=$username&service=database'
TARGET='_top'><IMG SRC='/tmp/geomist/images/geomistDatabasesMinW.jpg' ALT='Search
Databases' BORDER=0>\n";
        print TMPOUT "</BODY>\n";
        print TMPOUT "</HTML>\n";
        print "<HTML>\n";
        print "<BODY onLoad=\"this.location='/temp/$sessionID.html'\">\n";
        print "</BODY>\n";
        print "</HTML>\n";
  }elsif ($service eq "metadata"){
        print "<HTML>\n";
        print "<HEAD>\n";
        print "<TITLE>GEOMIST - Query Metabases</TITLE>\n";
        print "<frameset cols='*,180' border=2>\n";
        print "            <frame src='/geomist/inter_metabase/showtable.html'>\n";
        print "      <frameset rows='*,60' border=0>\n";
        print "            <frame src='/geomist/inter_metabase/query2.html'>\n";
        print "                                                        <frame
src='/cgi/return.pl?username=$username&sessionID=$sessionID&service=metadata'>\n";
        print "      </frameset>\n";
        print "</frameset>\n";
        print "</BODY>\n";
        print "</HTML>\n";
  }else{
     print "<HTML>\n";
          print "<HEAD>\n";
          print "<TITLE>GEOMIST - Download data</TITLE>\n";
          print "</HEAD>\n";
          print "<BODY BGCOLOR='#FFFFFF'>\n";
          print "<FONT SIZE=5>DOWNLOAD<FONT>\n";
          print "</BODY>\n";
          print "</HTML>\n";
  }
}else{
  print "<HTML>\n";
  print "<BODY BGCOLOR=#FFFFFF></P>\n";
  print "<TABLE><TR><TD><IMG SRC=\"/geomist/images/g_logo3.gif\"><TD><B><FONT
SIZE=+3>Geological and Mining Information System</FONT></B><BR><FONT SIZE=+2>Iberian
Pirite Belt</TABLE>\n";
  print "<CENTER>\n";
  print "<HR NOSHADE WIDTH=60%><BR>\n";
  print "<FONT SIZE=5>User identification</FONT><BR>\n";
  print "(Time out)\n";
  print "<HR NOSHADE WIDTH=60%><BR>\n";
  print "<TABLE BORDER=1>\n";
  print "<FORM ACTION='/cgi/auth_agent.pl' METHOD='POST'><BR>\n";
  print "<TR><TD ALIGN=CENTER><FONT SIZE=4>Username:<INPUT name='username'
value='$username' size='9'>\n";
  print "<TR><TD ALIGN=CENTER><FONT SIZE=4>Password:<INPUT name='password'
type='password' value='' size='9'>\n";
  print "<INPUT TYPE='hidden' name='service' value='$service'>\n";
  print "<TR><TD ALIGN=CENTER><FONT SIZE=4><INPUT type='submit' value='OK'>\n";
  print "<INPUT type='reset' value='Clear'>\n";
  print "</FORM>\n";
  print "</TABLE>\n";
  print "<HR NOSHADE WIDTH=60%><BR>\n";
  print "<MAP NAME=\"subscribeMap\">\n";
  print "<AREA SHAPE=RECT COORDS=\"19,2,60,16\" HREF=\"/\">\n";
  print "<AREA SHAPE=RECT COORDS=\"91,2,152,16\" HREF=\"/geomist/subscribe.html\">\n";
  print "<AREA SHAPE=RECT COORDS=\"190,2,258,16\" HREF=\"mailto:torres\@igm.pt\">\n";
  print "</MAP>\n";
  print "<IMG SRC=\"/geomist/images/subscribeBar.jpg\" BORDER='0' USEMAP=\"\#subscribeMap\"
ISMAP>\n";

  print "</HTML>\n";
}



############################################################
# FUNCTION authAgent(2)
#
# CREATION DATE 23/02/1999
# LAST MODIFIED 23/03/1999
#
# This function receives the one username and the supposed password
# and returns the access level, if the username and password match,
# returns 0 if the  username and password mismatch.
# Otherwise returns -1
##-----------------------------------------------------
sub authAgent{
          local($user)=$_[0];
```

```
    local($pass)=$_[1];
    local($ip)=&getParameterIN("REMOTE_HOST");

    local($i)=-1;

    if( ! open ( IN , "/disk2/httpd/htdocs/tmp/cgi/passwd.txt")){
        print "Impossivel abrir ficheiro!\n";
    }else{
        while($line = <IN>){
            #print "$line";
            @all =split (/ /,$line);
            if ($all[0] eq $user){
                $i=0;
                chop($all[3]);
                if ($all[3] eq $pass){
                    $i=$all[2];
                }
            }
        }
    }
    $i;
}
##########################################################

########################################################################
# FUNCTION getParameterIN(1)
#
# CREATION DATE 13/03/1999
# LAST MODIFIED 13/03/1999

# This function returns the variable from the current
# connection
##--
sub getParameterIN{
    local($result);
    local($param)=$_[0];

    foreach $element (keys %ENV){
        if ($element eq $param){
            $result=$ENV{$element};
        }
    }
    $result;
}
###

##########################################################

##########################################################
# FUNCTION sessionAgent(3)
#
# CREATION DATE 23/02/1999
# LAST MODIFIED 13/03/1999

# This function receives one username, access level and
# the IPAdress of the current connection. Then this
# function verify if there is one session dedicated. If is
# false then create a new session and returns the access level.
# If there is an open session up to time, then update the
# timestamp and returns the access level of this session.
# Otherwise return -1.
##----------------------------------------------------
sub sessionAgent{
        local($user)=$_[0];
        local($mip)=$_[1];
        local($level)=$_[2];

        local($i)=-1;
        #print "SA 1 $mip , $user , level=$level<br>";
        if ($level>0){
          if (!open(OUT,">>/disk2/httpd/htdocs/tmp/cgi/session.txt")){
                print "Impossivel abrir ficheiro de sessao!\n";
          }else{
                print OUT "$mip $user $level\n";
                $i=$level;
          }
        }else{
          if(!open (IN, "/disk2/httpd/htdocs/tmp/cgi/session.txt")){
                print "Impossivel abrir ficheiro de sessao para leitura!\n";
         }else{
                while($line = <IN>){
                        @all =split (/ /,$line);
                        if ($all[0] eq $user){
                                $i=0;
                                if ($all[1] eq $mip){
                                        chop($all[2]);
                                        $i=$all[2];
                                }
                        }
                }#while !eof
         } #if open
    } #if level
        $i;
}

sub jGetParameter{
        my $ret;
        my $arg=$_[0];
        my $in=$_[1];
        @all=split(/&/,$in);
        foreach $member(@all){
                @param=split(/=/,$member);
                if ($param[0] eq $arg){
                        $ret=$param[1];
                }
        }
        #print "QS=$in ARG=$arg RET=$ret\n";

        $ret;
}
```

3. USER AGENT

This conceptual agent is launched when there is a new user to be included in the users list. The process receives the user data and checks the validity of fields, according to the constraints defined previously.

Meanwhile the users database stores data about users within the structure:

(USERNAME ; USER GROUP ; ACCESS LEVEL ; PASSWORD ; NAME)

```
#!/usr/netscape/suitespot/install/perl

$i=0;
$access=0;
print "Content-type: text/html\n\n";

read(STDIN,$in,$ENV{'CONTENT_LENGTH'});

$username=&jGetParameter("username",$in);
$realname=&jGetParameter("realname",$in);
$companyname=&jGetParameter("companyName",$in);
$email=&jGetParameter("email",$in);
$password=&jGetParameter("password",$in);
$password2=&jGetParameter("confirmPassword",$in);
$realname=~ tr/+/ /;
$companyname=~ tr/+/ /;
$email=~ s/%40/\@/;
$pos=index($username,"+");
$res=& UserExists($username);
if ($username eq ""){
        & WriteInterface("Login name is required","",$realname,$companyname,$email);
}elsif ($pos != -1){
        & WriteInterface("Login name cannot contains spaces","",$realname,$companyname,$email);
}elsif ($res != 0){
        & WriteInterface("User <b>$username</b> alredy
exists","",$realname,$companyname,$email);
}elsif ($password eq ""){
        & WriteInterface("Password is required","$username",$realname,$companyname,$email);
}elsif ($password ne $password2){
        & WriteInterface("Passwords don't match","$username",$realname,$companyname,$email);
}else{
        open (APP,">>/disk2/httpd/access/passwd.txt");
        print APP "$username GR1 1 $password $realname\n";
        close (APP);
        print "<HTML>\n";
        print "<BODY BGCOLOR=#FFFFFF></P>\n";
        print "<TABLE><TR><TD><IMG SRC=\"/geomist/images/g_logo3.gif\"><TD><B><FONT
SIZE=+3>Geological and Mining Information System</FONT></B><BR><FONT SIZE=+2>Iberian
Pirite Belt</TABLE>\n";
        print "<CENTER>\n";
        print "<HR NOSHADE WIDTH=60%><BR>\n";
        print "<FONT SIZE=5>User Identifier<BR></font>\n";
        print "(User subscribed successfully)\n";
        print "<HR NOSHADE WIDTH=60%><BR>\n";
        print "<TABLE BORDER=1>\n";
        print "<FORM ACTION='/cgi/auth_agent.pl' METHOD='POST'><BR>\n";
        print "<TR><TD ALIGN=CENTER><FONT SIZE=4>Username:<INPUT name='username'
value='$username' size='9'>\n";
        print "<TR><TD ALIGN=CENTER><FONT SIZE=4>Password:<INPUT name='password'
type='password' value='' size='9'>\n";
        print "<INPUT TYPE='hidden' name='service' value='Nil'>\n";
        print "<TR><TD ALIGN=CENTER><FONT SIZE=4><INPUT type='submit'
value='OK'>\n";
        print "<INPUT type='reset' value='Clear'>\n";
        print "</FORM>\n";
        print "</TABLE>\n";
        print "<HR NOSHADE WIDTH=60%><BR>\n";
        print "<MAP NAME=\"subscribeMap\">\n";
        print "<AREA SHAPE=RECT COORDS=\"19,2,60,16\" HREF=\"/\">\n";
        print "<AREA SHAPE=RECT COORDS=\"91,2,152,16\"
HREF=\"/geomist/subscribe.html\">\n";
        print "<AREA SHAPE=RECT COORDS=\"190,2,258,16\"
HREF=\"mailto:torres\@igm.pt\">\n";
        print "</MAP>\n";
        print "<IMG SRC=\"/geomist/images/subscribeBar.jpg\" BORDER='0'
USEMAP=\"\#subscribeMap\" ISMAP>\n";
        print "</HTML>\n";


}


sub UserExists{
        $user=@_[0];
        $found=0;

        if(!open (IN,"/disk2/httpd/access/passwd.txt")){
                print "Error: Cannot open file\n";
        }else{
                while(($line = <IN>) && ($found==0)){
                        @all =split (/ /,$line);
                        if ($all[0] eq $user){
                                $found=1;
                        }
                }
        }
        close(IN);
        $found;
}

sub WriteInterface{
        $msg=@_[0];
        $username=@_[1];
        $realname=@_[2];
        $companyName=@_[3];
        $email=@_[4];

        print "<HTML>\n";
        print "<BODY BGCOLOR=#FFFFFF></P>\n";
        print "<TABLE><TR><TD><IMG SRC=\"/geomist/images/g_logo3.gif\"><TD><B><FONT
SIZE=+3>Geological and Mining Information System</FONT></B><BR><FONT SIZE=-2>Iberian
Pirite Belt</TABLE>";
        print "<CENTER>\n";
        print "<HR NOSHADE WIDTH=60%><BR>\n";
        print "<FONT SIZE=5>Sign in as User</FONT><BR>\n";
        print "(Error: $msg)";
        print "<HR NOSHADE WIDTH=60%><BR>\n";

        print "<TABLE BORDER=1>\n";
        print "<FORM ACTION='/cgi/subscribe.pl' METHOD='POST'><BR>\n";
        print "<TR><TD ALIGN=LEFT><FONT SIZE=4>Login name</FONT><FONT
COLOR=\"\#ff0000\">*</FONT><TD><INPUT name='username' value='$username' size='9'>\n";
        print "<TR><TD ALIGN=LEFT><FONT SIZE=4>User Name</FONT><FONT
COLOR=\"\#ff0000\">*</FONT><TD><INPUT name='realname' value='$realname' size='25'>\n";
```

```
	print "<TR><TD ALIGN=LEFT><FONT SIZE=4>Company<TD><INPUT
name='companyName' value='$companyname' size='25'>\n";
	print "<TR><TD ALIGN=LEFT><FONT SIZE=4>Email<TD><INPUT name='email'
value='$email' size='25'>\n";
	print "<TR><TD ALIGN=LEFT><FONT SIZE=4>Password</FONT><FONT
COLOR='#ff0000'>*</FONT><TD><INPUT name='password' type='password' value='' size='9'>\n";
	print "<TR><TD ALIGN=LEFT><FONT SIZE=4>Confirm Password</FONT><FONT
COLOR='#ff0000'>*</FONT><TD><INPUT name='confirmPassword' type='password' value=''
size='9'>\n";
	print "</TABLE><P>\n";
	print "<INPUT type='submit' value='OK'>\n";
	print "<INPUT type='reset' value='Clear'>\n";
	print "</FORM><P>\n";
	print "</FONT><FONT COLOR='#ff0000'>*</FONT> Indicates required field\n";
	print "</HTML>\n";
}

sub jGetParameter{
	my $ret;
	my $arg=$_[0];
	my $in=$_[1];
	@all=split(/&/,$in);
	foreach $member(@all){
		@param=split(/=/,$member);
		if ($param[0] eq $arg){
			$ret=$param[1];
		}
	}
	#print "QS=$in ARG=$arg RET=$ret\n";

	$ret;
}
```

4. SESSION SERVER AGENT

This effective agent, running as a daemon, handles session's database. Using inter-process communication techniques, this agent acts as a server, accepting connection from distinct applications.

```
import java.io.*;
import java.net.*;
import java.util.Date;
import java.util.Vector;
import java.util.Hashtable;
import java.lang.Integer;
import java.util.Random;
import java.lang.Math;

class jSession{
	String sessionID;
	String sessionName;
	Date startDate;
	Date lastDate;
	String sessionLevel;

	public void setSession(String id, String level){
        String s="";
        for (int k=0; k < 3; k++){
            Random r=new Random();
            long x=r.nextLong();
            if(x<0) x=x*(-1);
            s=s+x;
        }

        sessionID=s;
        sessionName=id;
        startDate=new Date();
        lastDate=new Date();
        sessionLevel=level;
    }
    public String getID(){return sessionID;}
    public String getName(){return sessionName;}
    public long getElapsedTime(){
        Date thisDate=new Date();
        long res=thisDate.getTime()-lastDate.getTime();
        return res;
    }
    public String getLevel() { return sessionLevel;}
    public void update(){ lastDate=new Date();}

}

class jUser{
    public String username;
    public String password;
    public String level;
    public String group;
}


public class jServer27{

    public static Hashtable userSet=new Hashtable();
    // { ( username ; user ) }
    public static Hashtable sessionSet=new Hashtable();
    // { ( sessionID : session ) }
    public static Hashtable userSessionSet=new Hashtable();
    // { ( username ; sessionID ) }
    public final static int port=8087;
    public static int nRequests=0;
    static int timeOut=10*60*1000;

    static Vector stringToList(String s, char c){
        String s2;
        int i;
        Vector xList = new Vector();
        while (true){
            i=s.indexOf(c);
            if(i>0){
                s2 = s.substring(0,i);
                s = s.substring(i+1);
                xList.addElement(s2);
            }else break;
        }
        xList.addElement(s);
        return xList;
    }

    public static void main (String[] args){
        ServerSocket theServer;
        Socket theConnection;
        PrintStream outputStream;
        String st = new String();

        /*
        try{
            DataInputStream in = new DataInputStream(
                        new BufferedInputStream(
                        new FileInputStream("/disk2/httpd/access/passwd.txt")));
            String s=new String();
            userSet.
            s=in.readLine(); // header line
            while (( s = in.readLine()) != null){
                Vector mList = stringToList(s,' ');
                jUser ju = new jUser();
                ju.username=(String) (mList.elementAt(0));
                ju.group=(String) (mList.elementAt(1));
                ju.level=(String) (mList.elementAt(2));
                ju.password=(String) (mList.elementAt(3));

                //System.out.println(ju.username+" ; "+ju.group);
                userSet.put(ju.username,ju);
            }
            in.close();
        }catch (IOException e){
            System.err.println(e);
        }
        */

        try{
            theServer=new ServerSocket(port);
            System.out.println("Server on-line...");
            try{
                while(true){
                  String mResult="";
                  theConnection=theServer.accept();
                  nRequests++;
                  try{
                    DataInputStream in = new DataInputStream(
                        new BufferedInputStream(
                        new
FileInputStream("/disk2/httpd/access/passwd.txt")));
                    String s=new String();
                userSet=new Hashtable();
                s=in.readLine(); // header line
                while (( s = in.readLine()) != null){
                    Vector mList = stringToList(s,' ');
                    jUser ju = new jUser();
                    ju.username=(String) (mList.elementAt(0));
                    ju.group=(String) (mList.elementAt(1));
                    ju.level=(String) (mList.elementAt(2));
                    ju.password=(String) (mList.elementAt(3));

                    //System.out.print(ju.username+" : ");
                    userSet.put(ju.username,ju);
                }
```

```
                        in.close();
                }catch (IOException e){
                        System.err.println(e);
                }

                         System.out.print(nRequests+" : ");
                         BufferedReader inputStream = new BufferedReader(
                                        new InputStreamReader(
                                        theConnection.getInputStream()));
                         outputStream=new PrintStream(theConnection.getOutputStream());
                         String mStr=inputStream.readLine();
                         Vector mList = stringToList(mStr,' ');
                         if ((mList.size())>1){
                                String mID = (String) mList.elementAt(0);
                                jSession ms=new jSession();
                                if (mID.equals("543212345")){
                                        //Create a new session
                                        if ((mList.size())>2){
                                                String mUsr = (String) mList.elementAt(1);
                                                System.out.print(mUsr+" : ");
                                                if (userSet.containsKey(mUsr)){
                                                        jUser      ju       =       (jUser)
userSet.get(mUsr);
                                                        if
(!userSessionSet.containsKey(mUsr)){
                                                                ms=new jSession();
                                                                ms.setSession(mUsr,
ju.level);

        sessionSet.put(ms.getID(),ms);

        mResult="result=OK&sessionID="+ms.getID()+"&sessionLevel="+ms.getLevel();

        userSessionSet.put(mUsr,ms.getID());
                                                                System.out.print("
"+ms.getID()+" "+ms.getLevel());
                                                        }else{
                                                                //Session already exists
                                                                ms       =       (jSession)
sessionSet.get((String) userSessionSet.get(mUsr));

        mResult="result=OK&sessionID="+ms.getID()+"&sessionLevel="+ms.getLevel();
                                                                System.out.print(          "
"+ms.getID()+" "+ms.getLevel());

                                                                ms.update();
                                                        }
                                                }else   mResult="result=Error:&Obs=User
not found";
                                        }else          mResult="result=Error:&Obs=Request
login";
                                }else{
                                        boolean found=false;
                                        if (sessionSet.containsKey(mID)){
                                                ms = (jSession) sessionSet.get(mID);
                                                long mTime=ms.getElapsedTime();
                                                if (mTime < timeOut){
                                                        ms            =            (jSession)
sessionSet.get(mID);

        mResult="result=OK&sessionID="+ms.getID()+"&sessionLevel="+ms.getLevel();
```

```
                                                            ms.update();
                                                            System.out.print("    "+    mTime+"
"+ms.getLevel());
                                                      }else{
                                                            sessionSet.remove(mID);

      userSessionSet.remove(ms.getName());
                                                            System.out.print(" "+mTime);

      mResult="result=Error:&Obs=Time out";
                                                      }
                                                }else    mResult="result=Error:&Obs=Session   not
found";
                                          }
                                          }else    mResult="result=Error:&Obs=jClient    <sessionID>
[<sessionUser>] [<passwordUser>]";
                                    outputStream.println(mResult);
                                    //outputStream.println(new Date());
                                    System.out.println(";");
                                    theConnection.close();
                              }
                        }
                        catch (IOException e){
                              theServer.close();
                              System.err.println(e);
                        }
                  }
                  catch (IOException e){
                        System.err.println(e);
                  }
      }
}
```

5. SESSION CLIENT

This client, also written in Java, is able to connect Session Server, and can be loaded from anywhere within the system.

```
import java.net.*;
import java.io.*;

public class jClient20{

      public final static int port=8087;
      public final static String hostname="localhost";

      public static void main(String [] args){

            Socket theSocket;
            DataInputStream theStream;
            PrintStream ps;
            try{
                  theSocket=new Socket(hostname, port, true);
                  ps=new PrintStream(theSocket.getOutputStream());
                  String s=new String();
                  for (int i=0; i<args.length;i++) s+=(args[i]+" ");
                        ps.println(s);
                        theStream=new DataInputStream(theSocket.getInputStream());
                        while ((s=theStream.readLine()) != null){
                                System.out.println(s);
                        }
                }
                catch (IOException e){
                        System.err.println(e);
                }
        }
}
```

6. CHECK SESSION

```
'
' Name:    java.CheckSession
'
' Author:  Jorge Lopes <jlopes@igm.pt>
' Creation: 20/07/99
' Update:  20/07/99
'
' IN:      (sessionID : String)
'
' OUT:     enum ( -1:Unknown error | 0:Session not valid | enum (1|2|3):sessionAccessLevel)
'
'

mSession=self.get(0)
result = -99

'------ Check for session --------
system.execute("cd /atlas/geomist ; java jClient21du "+mSession+" > jtest.log")

lf=LineFile.Make("/atlas/geomist/jtest.log".AsFileName,#FILE_PERM_READ)
str=lf.ReadElt
if (str.count=0) then
  result = -1
else
  resList = str.asTokens("& ")
  if (resList.get(0).asTokens("=").get(1)="Error:") then
    result = 0
  else
    result = resList.get(2).asTokens("=").get(1).asNumber
  end
end

return result
```