Development of Software for Geothermal Applications through Integration Technologies

Andrea Spinelli[1], *Franco Graziano*[1], *Paolo Salvaneschi*[1]
Fausto Batini[2], *Ivano Dini*[2]

Key words: software, geophysical applications, SIGMA, IPOGEO

We introduce two systems developed for geothermal applications, and illustrate the techniques used for their development. SIGMA is an integrated system for management, processing and visualisation of geomagnetic, gravimetric and electrical resistivity data, integrating public domain codes for filtering with commercial interpolation routines. IPOGEO is an integrated system for the management and processing of seismic data. Several algorithms have been integrated to produce synthetic seismograms and seismic tomographies.

We used a re-engineering process, which allows to take programs embodying significant domain knowledge and improve some of the characteristics of those programs, supporting geophysicists in their work, hiding unnecessary implementation details in the integration machine.

We present the technique and the software environment we used for re-engineering and integration of programs. Our methodology is centred around the idea of software model, *which is a description of a set of software components, their activation modalities and their interconnections. Requirements have been derived though the use of hypertext for rapid prototyping. We used a well-defined software architecture and a set of C++ libraries specifically designed to support integration.*

1. Introduction

In the field of scientific applications, several algorithms and application codes have become a standard for h e level of functionality h e y provide. In other words they solve extremely well some specific problems and it would be useless and risky to try to substitute hem. **On** the other hand. they do not conform **D** today's standards of usability (e.g h e y have character-oriented user interfaces): moreover, it is quite difficult **D** use several of hem together. for instance because their data formats are incompatible.

In a coordinated effort among geothermal researchers and software specialists, we developed two applications , *Sigma* and *Ipogeo*, that integrate existing software tools in a coherent. user-friendly. graphical environments. They are discussed in more detail in the next two sections.

Afterwards we briefly outline h e software techniques used for integration ofthe components. The last section describes some ongoing work

2. Sigma

Sigma is an integrated system for the management and processing of magnetometric, gravitational and geoelectrical data. The processing of such data types is a well-established geophysical technique used to infer positions and shape of bodies chracterised by anomalous properties, respectively magnetic susceptibility, mass density and electric conductivity Information obtained through these techniques are used to formulate hypoteses about geothermal reservoirs and surrounding geological structures.

The main objective of Sigma is to make easy the **manipulation of** heterogeneous dam sets, heir interpolation and **filtering** and **h e production** of maps of various fields.

2.1. Metaphor and User Interface

Sigma exploits the desktop metaphor: **h e user** sees on h e screen a stylised desktop, populated by icons representing dam sets and by folders, which may contain more data sets and/or folders.

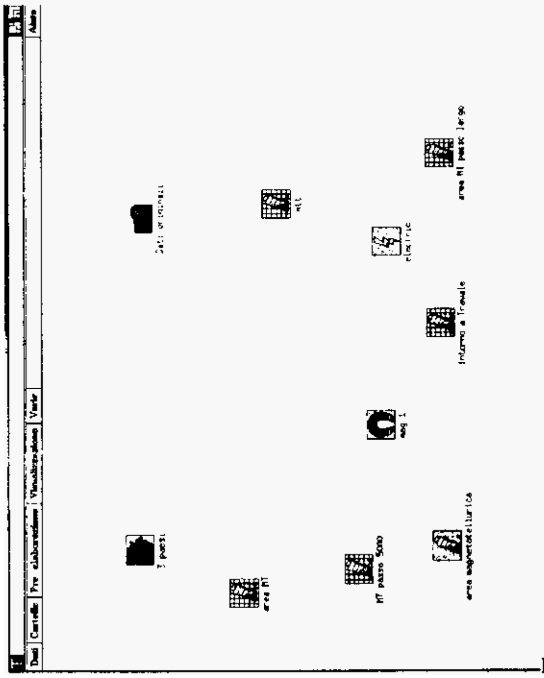


Figure **1:SIGMA** Desktop

The user may select one or more dam sets with **h e mouse.** may move them around the desktop and perform operations on h e data sets by choosing appropriate menu items.

Data sets have a type, in the sense that certain operations are allowed only on certain classes of data: for instance, h e bandpass filler can work only on gridded data and not on scattered dam. The interface is aware of interconnections among dam and operations. enabling only the menu items corresponding **D** operations applicable to the currently selected data sets. Operations hat are not compatible with h e current selection are disabled and cannot be activated.

---

[1]ISMES SpA, Via Pastrengo, Seriate BG, Italy, phone +39-35-307111, fax +39-35-307710, e-mail aspinelli@ismes.it

[2]ENEL-DPT-VDAG, Via Andrea Pisano, Pisa PI, Italy.

2.2. Data Types

Sigma currently handles three families of data sets:

- *magnetic*, whose meaning is the magnetic anomaly in a given point of the surface of the Earth
- *gravimetric*, referring to the gravimetric anomaly
- *geoelectric*, referring to resistivity measures centred in a given point; the Schlumberger arrangement is used, providing for each point considered a curve relating the resistivity with the semi-distance of the electrodes.

Each basic family has three subfamilies:

- *scatter set*: a set of measures actually taken in a set of points (stations). Usually these measures result from geophysical surveys
- *grid*: a set of interpolated values, aligned on a rectangular array of points. Sigma handles grids aligned with the kilometric UTM (or Gauss-Boaga) reticulate, allowing for different grid steps in the NS and EW directions. Grids are the result of the interpolation of scatter sets or of transformation of other grids.
- *profile*: a set of interpolated values along a straight line segment.

For the purpose of producing useful maps of the fields, it was necessary to add the Auxiliary data set class, which includes topographic features such as cities, polygonals (for rivers, coastlines and the like), wells and labels.

2.3. Operations

We classified the available operations in several groups, sketched below.

2.3.1. Data management

It is possible to import data files in a few common formats; each data file becomes a scatter data set, which is stored in Sigma's internal database. Scatter sets can also be manually input and modified. Scatter values and topographical data may also be input through a graphic tablet.

In the set-theoretical sense, it is possible to take the union of two data sets: a new data set will contain the values of both sets, making it possible to combine the results of different surveys. Moreover, given a data set, it is possible to build another set by selecting only the measurements belonging to a smaller area, thereby increasing the speed of further processing.

2.3.2. Preprocessing

Scatter data may be interpolated over two-dimensional grinds or along straight lines. We used a commercial interpolation package (Uniras) to provide both local-fit and triangulation-based interpolation. Grids may further be preprocessed calling the USGS computing code fftfil_new [FFT], which features the following transformations:

- *passband filtering*, to eliminate from the observed anomaly field components of undesired wavelength; for instance it is useful to cut short-wavelength anomalies, related to very small bodies, retaining only components relating to bodies of dimensions relevant for the current investigation.
- *vertical derivatives* (1^{st} and 2^{nd}), which can delineate boundaries of anomalous bodies, through processing of magnetic and gravimetric data.
- *reduction to the Pole*, used for interpretation of magnetic data. It is well known that at middle latitudes an anomalous magnetic body produces two anomalies, one positive and one negative. The reduction to the Pole computes the anomaly that could affect the magnetic field at Pole, where the magnetic field vector is verticl, so that the anomalous body would produce a unique anomaly (maximum or minimum).
- *pseudo-gravimetric and pseudo-magnetic* transformation; assuming that a body has anomalous magnetic or gravimetric properties, it is possible to compute the magnetic anomaly from the gravimetric field, and vice-versa. For instance, comparing a magnetic anomaly with the pseudo-magnetic the interpreter can state whether a body has both magnetic and gravimetric anomalous properties
- *directional filtering*, which has purposes similar to those of the passband filtering. However, in this case the desired (or undesired) components are those characterised by their orientation on the horisontal plane. This filter is used to extract or to eliminate components whose minima/maxima have a certain orientation.

We also added a polynomial filter, which fits a polynomial of arbitrary degree to the given grid, in the least squares sense.

Another possibility is the linear combination of grids. It is porriblc, for instance to subtract a regional anomaly polynomial from the anomaly grid, obtaining a sharpeer local-anomaly map. Or, it is possible to add a gravity anomaly grid with a pseudo-gravity grid derived from a magnetic grid from the same area, obtaining a *stacking* effect with some noise concellation.

2.3.3. Visualisation

The visualisation tools include a profile tool and a map tool. The profile tool is quite simple, and displays a x-y representation of the variation of a quantity along a straight line.

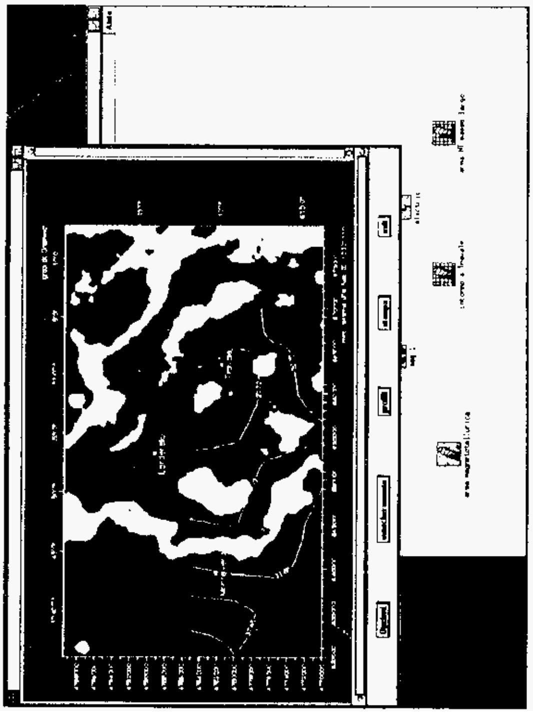


Figure2: SIGMA profile tool

The map tool is richer and allows to display simultaneously two grids, a scatter set and a topographic ret. The first grid is contoured in colour, while the second grid is represented as isolines.

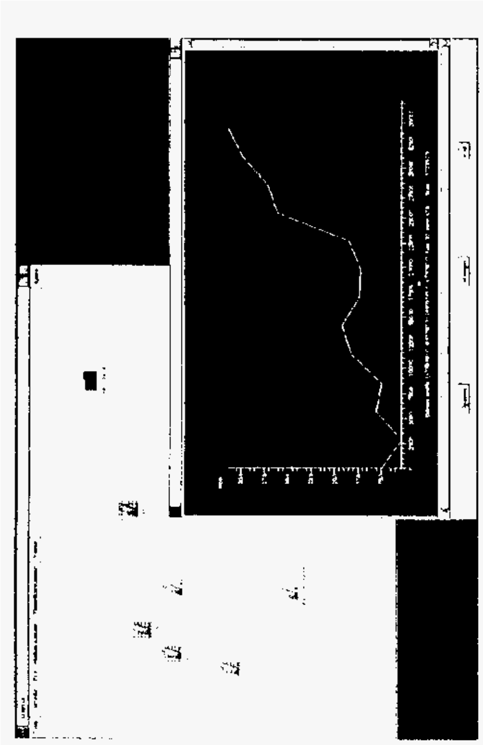


Figure 3: SIGMA map tool

Both kilometric and geographic coordinates are shown. Maps (and profiles) can be printed on several devices, including A0 electrostatic plotters. The user may choose a standard topographic scale (e.g. 1:25000) allowing for comparisons with standard maps.

2.3.4. *Other functionalities*

Several service functionalities are provided, among which: choosing the preferred coordinate system, changing the identifier of a data set, deleting data sets, moving data sets across folders, creating folders, etc.

3. Ipogeo

Ipogeo is an integrated system for seismic data management and processing. Its main goal is to increase the usability of existing codes for the direct and inverse modelling of physical properties of rocks.

3.1. Metaphor and User Interface

Ipogeo adopts the 3-D metaphor: the user sees a local portion of the Earth simultaneously from three differents point of view: from above, from a side and as an axonometry. The system visualises both physical objects, as wells or seismic sources and receivers, and non-physical objects, such as 2-D and 3-D velocity models.

In the following figure, the map is the upper left portion of the window, the side view is on the lower left, the axonometry is on the right.

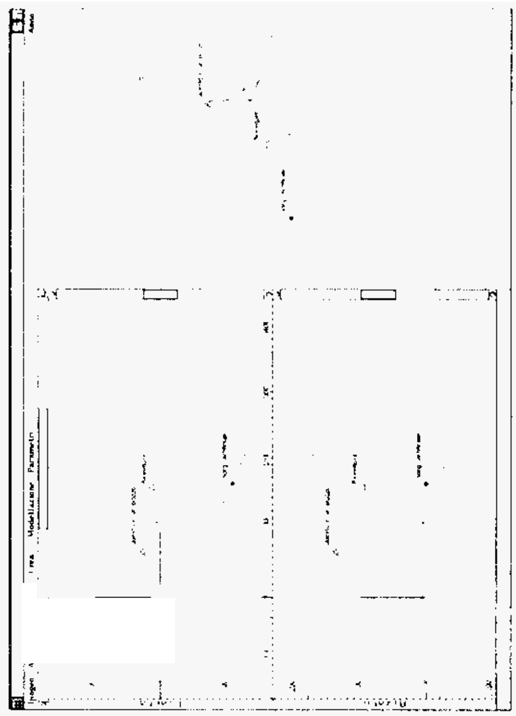


Figure 4: IPOGEO world view

The user may select objects or parts of them with the mouse, move them, modify their parameters through a set of forms and performs modelling actions selecting appropriate menu items.

Modelling actions are performed on the currently selected objects; wrong selections are sensed and signaled to the user.

3.2. Data types

Ipogeo currenty handles the following data types:

- seismic sources, both artificial (shootings) and natural (earthquakes);
- seismic receivers;
- seismic events, seismograms and first-arrival times;
- 2-D interface and velocity models;
- 3-D parallel plane velocity models;
- 3-D regular grid velocity models;
- 3-D irregular grid velocity models.

3.3. Operations

The operations may be classified in several groups, explained in the following paragraphs.

3.3.1. *Preliminaries*

The user can choose a *project*, which is the representation of a specific area of the subsoil, over which all operations will be performed. Project data are persistent across working sessions.

3.3.2. Data Managemenr

It is possible to import data in several formats, rave the current project and, of course, exit the program.

3.3.3. Creation

It is possible to create all he objects in the domain by menu item selection. Objects have a default configuration. which may be modified by selecting them.

3.3.4. Edit

Objects may be destroyed, moved and modified via form: each data type has an associated set of operation. which are evidenced through a popup menu. For instance, parallel plane models have h e associated operations *add a* plana, *delete a plane*, *modify a stratum*.

The same operations may be performed by using an appropriate combination of mouse keys on the image of the desired object Moving is performed through mouse dragging. while a click with the middle mouse button pops up the specialised popup menu.

Objects may be quite complex: for instance, a parallel plane model is composed by *n* strata and *n*+1 interfaces. A special protocol has been devised for selecting a whole object and for selecting specific parts of a complex object.

3.3.5. Modelling

The rysm is capable of performing direct and inverse modelling. Direct modelling allows to generate synthetic seismograms from an experimental configuration and a crustal model. On the other hand. inverse modelling takes a set of experimental data, such as a set of earthquakes and their first-arrival times at given locations, an initial model of some rock properties, and. using the experimental data. adjusts the model to better fit the observed values.

Ipogeo hides to the user all the low-level derails linked to the usage of complex coder. For instance, after having placed seismic sources and receivers in the 3D world and after having created a 2D interface model. the user can select with the mouse h e model. some sources and some receivers and select h e menu item *2D direcr modelling*. The system collects data from its internal database, builds appropriate input filer. launches h e PRX87 program (described below) on the generated files. reads h e output files translates their format and launches a visualisation program which graphically depicts h e generated seismograms.

The geophysical knowledge required for modelling is embedded in some state-of-the an codes developed at h e Massachussetts Institute of Technology. which have been left intact. A brief description of the codes is given in the following paragraphs.

PRX87 (see [Bay]) computes approximate asymptotic Green's functions (travel time. amplitude and phase) of a seismic wave field, using the paraxial ray method. The model is 2D, the medium is laterally inhomogeneous, with curved interfaces. Results are convoluted with a wavelet to obtain synthetic seismograms.

TIWVN (see [Tok]) computes the complete frequency domain response for transversely isotropic media. As in PRX87, synthetic seismograms are obtained.

XRAY (see [Man]) performs inverse tomography using travel time data such as hose obtained fmm Vertical Seismic Profiling (VSP); the result is a velocity model.

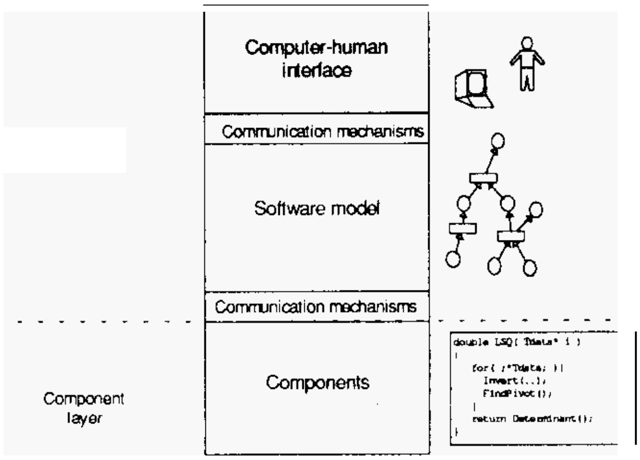
HVI (see [Blo]) inverts the arrival time data of local earthquakes to obtain simultaneously hypocentral coordinates and a velocity model of the crust beneath a local seismic network

4. Integration Environment

For Ipogeo and Sigma. the MI integration environment, developed by Ismes SpA, was adopted. Its characteristics are sketched in his paragraph. For a more complete discussion. h e reader is referred to [Spi] and [Sal].

The general aim of MI is to make it simple for project leaders and programmers to develop a new application is h e scientific and engineering area, reusing existing components.

Every MI application shares a common structure, shown in h e following figure.



There are two main layers: the component layer. which contains h e software tools to be integrated, and h e integration layer, which maps the original components to a model of interaction that may be very different from the original one.

4.1. Software component layer

The software components to be integrated are usually at h e granularity level of a program. Often they are available only at the executable level; even when source code is available, they are untouched, as far as possible. when necessary, a capsule is hilt around several components, making hem look like a single complex component; sometimes the same program is urcd by several different capsules.

4.2. Integration layer

The focus point of the architecture is h e software model. which is an abstraction of the existing components and provides for presenting than to h e user. The model uses some communication mechanisms for interacting with the user and with the software components. In other words. the model translates h e components in a visual language understandable by h e user. and translates in turn h e user commands into input and activation commands for h e components. [Tak]

Software models are not constrained to a specific formalism: support for various typologies of models is instead pmvided. Models an built with C++ linguistic constructs and are, in thin sense. object oriented. For instance. one possible model is h e hierarchical model: the system is a tree, whose nodes are functions; h e top level functions are decomposed into sub-functions, which in turn are decomposed into sub-sub-functions, going down to h e leaves. which are connected with individual programs. This model has been used to impose some degree of classification over a set of programs coming from different sources, by collecting them in families of tools. From h e user viewpoint. his model visualises itself as a cascade of expanding windows. A different visualisation of the same model may be a hierarchy of menus.

Often, the model the user has of h e system is not coincident with the actual physical structure of the software. and it is useful to present as a single functionality a complex sequence of invocations of components:

conversely. it might be convenient to show the same component as a set of different functions. differing for some invocation parameter. Moreover. for some applications, data-driven models are better than functionality-driven ones; however, existing software components are usually hid-genemuon, functionality-oriented programs, and must be mapped to a data-driven model of operation.

Software models are clients of mechanisms for communication with the components. through an abstraction of the Unix and VMS concepts of *process.* and with the compuler-human interface, through some abstractions over the concepts of *interactor* and form.

Different models may feature different levels of integration; typically, data-driven models are more amenable for data integration, while functionality-driven models aim at control integration.

4.3. MI libraries

The MI libraries provide C++ classes for implementing:

- software models;
- communication between models and components;
- communication between models and the user interface;
- common types and data structures.

The computer-human interface layer uses a standard platform (X11), a public domain toolkit (InterViews 2.6, [Vli]), and a set of custom classes.

5. Discussion and Work in Progress

The use of Integration techniques has allowed to respect the applicative knowledge content of stale-of-he an codes, while building a modem user-friendly interface Moreover, the integration of heterogeneous funcionalities has brought, as a side-effect, new functionalities related to the possibility of visualising in a single environment different data from disparate sources. The only alternative would have been to develop all applications from scratch, with prohibitive time and cost requirements.

Currently. Sigma is being expanded so as to handle a new dam set family. namely the earthquake localisations. It will be therefore possible to superimpose earthquake locations with magneuc, gravitational and electric dam. Moreover. a new tool will allow to visualise localisations projected over a vertical plane.

Other variables which will be visualised are various seismic statistics, such as the frequency of earthquakes. as a 2-dimensional field.

6. References

[Bay] Baydoun, *W.*, B., *Asymptotic wave methods in heterogeneous media*, PhD Thesis, Massachussetts Institute of Technology. 1985.

[Blo] Block, L.. V., Joint hypocenter velocity inversion of local earthquake arrival time dam in two geothermal regions. PhD Thesis, Massachussetts Institute of Technology. 1991.

[FFT] Hildenbrand, T., G.. *FFTFIL.a filtering program based on two-dimensional Fourier analysis of geophysical data*, Open-File Report 83-237, USGS.

[Man] Mandal, H.,Toksoz, M., N., "Computation of complete waveforms in general anisotropic media". *Geophysical Journal*. IYYO.

[Sal] Salvaneschi, P., "The Use of Integration Tools in Software Maintenance", in: Proceedings "i the IEEE Conference on Software Maintenance. Sorrento, Italy. 1991.

[Spi] Spinelli, A.. Salvaneschi, P., Cadei, M., Rocca, M.. "MI - an object oriented environment for integration of scientific applications", to appear in: *OOPSLA '94, Conference on Object Oriented Programming System, Languages and Applications*, Portland, Oregon, 1994.

[Tak] Takahashi. S. Matsuoka S., Yonezawa, A., "A General Framework for Bi-Directional Translation between Abstract and Pictorial Dam". in: *UIST '91, Proceedings of the ACU Symposium on User Interface Software and Technology.* ACM Press. 1991

[Tok] Toksoz, M.. N., *Forward and Inverse Methods for VSP Tomography*, Annual technical report, Massachussetts Institute of Technology, 1990.

[Vli] Vlissides, J.M., Linton, M., "Applying Object-Oriented Design to Structured Graphics". in: *Proceedings of the USENIX C++ Conference,* Denver. Colorado, 1988.