

# Information GRID in the Corporate World

N.Bogonikolos  
Zeus Consulting S.A - LogicDIS Group. 93 Riga Feraiou str.,  
26221 Patras, Greece,

M. Chrysostalis,  
Zeus Consulting S.A - LogicDIS Group. 93 Riga Feraiou str.,  
26221 Patras, Greece

K. Giotopoulos  
Zeus Consulting S.A - LogicDIS Group. 93 Riga Feraiou str.,  
26221 Patras, Greece and  
Department of Computer Engineering and Informatics  
26110 Patras, Greece

S. Likothanassis  
Department of Computer Engineering and Informatics  
26110 Patras, Greece and  
Computer Technology Institute 3 Kolokotroni Str,  
26221 Patras, Greece

K.Votis  
Zeus Consulting S.A - LogicDIS Group. 93 Riga Feraiou str., 26221  
Patras, Greece. and  
Department of Computer Engineering and Informatics 26110  
Patras, Greece  
*kvotis@zeusnet.gr*

**This paper introduces the COG (Corporate Ontology Grid) project, an EU project funded under the Information Society Technologies programme. In particular the Integration of the Ontological Modelling Suite (Unicorn system) with Industrial Modelling and EAI Tools is discussed. The COG (Corporate Ontology Grid) project addresses the problem of accessing and communicating data held in heterogeneous data formats scattered across disparate systems. COG demonstrates the tangible value of applying grid technologies to industry and it serves as an enterprise level test-bed for a scientifically and technologically innovative approach. COG is developing both a series of ontologies and an entire Information Grid ontological system capable of interacting with disparate data sources from numerous environments. The expected result of COG is the automated generation of transformation scripts from the ontology and from the mappings by turning the ontologies into an “active thesaurus” capable of making data dynamic.**

*Keywords: grids, information grids, corporation, ontologies, EAI tools, modelling*

## 1. INTRODUCTION

Modern Corporate Information Systems utilize numerous heterogeneous databases, storing different types of data, using different applications to manage data, running on different hardware. Corporate, manufacturing and government concerns throughout the world today face an ever-increasing problem. They are encumbered by the difficulties and inefficiencies inherent in rationalizing, utilizing, viewing, comprehending and querying data that is held in heterogeneous data formats and scattered across disparate systems – a problem whose solution demands the creation of corporate Grids.

Grid technologies are at a cusp in their development. They are beginning to emerge from ‘number crunching’ computational networks serving scientific research projects and to move into the industrial workplace. The success of the assimilation of Grid technologies into industry depends on the extension of their use beyond the purely computational dimension. They must be developed to perform in the complex and difficult world of the corporation and organization.

As many organizations have discovered, even if they use exactly the same hardware and software components during data interchanges, differences in the way database or XML schemas are defined at the syntactic and semantic level, no matter how small, can hamper data sharing. Indeed this is the main reason that even the widespread adoption of schematic standards, whether these be Entity- Relationship Diagrams, UML or XML, have not delivered on the promises of interoperability that originally accompanied their adoption. Only a corporate Grid to access the data, combined with a central data dictionary, or ontology, to make semantic sense of the data can provide the transparent knowledge interchange and interoperability that is so acutely needed.

Today, the ability of companies to compete depends more than ever on their ability to optimise their information assets. They need on-demand access to data, regardless of the format in which it is stored, along with smooth transfer of information from one data source to another, without days of labour. COG project synthesizes Grid technologies with ontologies and web services that will interact with disparate data sources, unify them by comprehending the semantic meaning underlying the data, and produce automatic data translation on demand. The successful realization of this goal can reduce costs and substantially boost efficiency in corporate and manufacturing environments. This type of commercial Information Grid can then allow application of the known techniques of computation Grids to the integrated information.

Section 2 briefly describes the three layers of Grids and then in section 3 we are presenting the information Grid within current corporations. In section 4 we are describing the COG architecture and in section 5 we provide the possible integration ways of COG Unicorn system by using an EAI solution.

## **2. THREE LAYER GRID ABSTRACTION**

Conceptually, the Grid can be thought of in terms of three layers. Underlying everything is the computational and data grid: the computer hardware and data networks upon which the work will be conducted. Above this is the 'information grid': the databases of information to be accessed by the hardware, and systems for data manipulation. On top is the 'knowledge grid', where high-level applications will mine the data for the knowledge that can form the basis of semantic understanding and intelligent decision making.

A data/computational grid forms the fabric of the Grid to provide raw computing power, high-speed bandwidth and associated data storage in a secure and auditable way. Diverse resources are represented as a single met computer so the way that computational resources are allocated, scheduled and executed, and the way that data is shipped between processing resources, is handled here.

An information Grid provides homogeneous access to heterogeneous distributed information by dealing with the way that all forms of information are represented, stored, accessed, shared and maintained. This layer orchestrates data and applications to satisfy the request, including toolkits for composing work flows, accessing metadata, visualisation, data management, and instrumentation management. The web and other well-known and current middleware technologies are incorporated into one framework.

A knowledge Grid using knowledge based methodologies and technologies for responding to high-level questions and finding the appropriate processes to deliver answers in the required form. This last layer includes data mining, machine learning, simulations, ontologies, intelligent portals, workflow reasoning and Problem Solving Environments for supporting the way knowledge is acquired, used, retrieved, published and maintained. A knowledge Grid should provide intelligent guidance for decision makers and hypothesis generation.

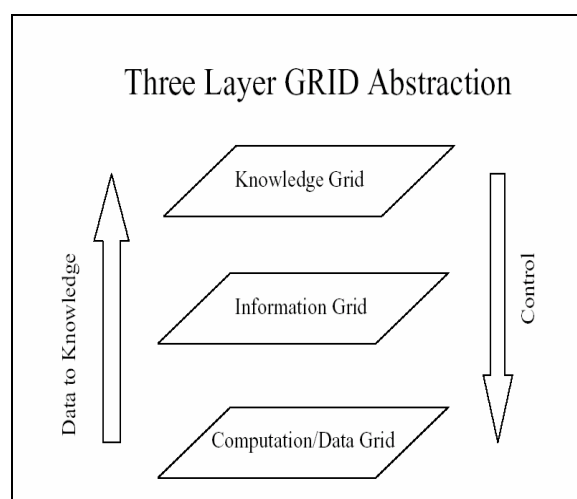


FIGURE 1: Three layer Grid

### 3. INFORMATION GRID IN CURRENT CORPORATIONS

The difficulties and inefficiencies inherent in rationally utilizing, viewing, comprehending and querying data that is scattered across heterogeneous and disparate data sources are understood and keenly felt in all European corporations. This statement holds true regardless of the industry in question. Europe's larger companies typically have upwards of two thousand incompatible databases. This problem is well known in all global markets and particularly in the European Union, where member nations have a variety of languages, histories, conventions, currencies, measures, units, and technological traditions.

Data management is crucial in all aspects of running a modern business – for inventory control, human resources, sales, customer service, product management, manufacturing floor processes, to name only a few crucial areas. However, what is rapidly emerging is an understanding that it is not sufficient for a company simply to store raw information in large quantities; such information is nearly useless unless data that is kept in separate locations is readily available on-demand by interacting employees and departments. Thus the new field of *data-interoperability* is being developed using an Information Grid approach. This is also a prerequisite to the adoption of computational Grids in industry. It complements Application Integration Middleware (AIM) (or Enterprise Application Integration, EAI), the field which deals with the physical movement of data between and among applications.

The full extent of data-interoperability is likely to be realisable only within the context of an Information Grid, and cannot attain full expression and efficiency without an underlying Information Grid. This is because an Information Grid enables the construction of a hub-and-spoke model of data and knowledge interconnectivity that is much more efficient than a point-to-point model that would otherwise be constructed. Given an Information Grid, an organisation wishing to work with  $n$  separate data sources as one 'Virtual Knowledge Repository' need only map the schemas of each data source once to the central ontology (the hub-and-spoke approach) for a total of  $n$  mappings. The organisation can then rely on the operating system inherent in the Grid to automatically generate the semantic connections, and determine in a flexible manner which data instances from which sources need to be accessed at any given time. Without a Grid, each data source must be mapped separately to each other source, for a total of  $o(n^2)$  mappings, and the efficient selection of instance sources for transformations or query returns, from amongst all the possibilities, cannot truly be executed.

Data interoperability is also important in both *inter-corporation interactions* (i.e. *business-to-business*) and *intra-corporation settings*. The inter-corporation realm requires data interoperability for a different set of problems. First, due to a number of factors including globalisation, industry consolidation, the impact of the Internet and the recent European monetary integration in many member states, many organisations are increasingly dispersed geographically and are organised around virtual teams, with their data dispersed widely. Second, the immense spending on IT products in recent years, along with the dizzying pace with which new standards and technologies are being developed, has left organisations with a bewildering assortment of hardware and software components that cannot "talk to each other" without significant expenditures and effort.

The same problems appear when different corporations attempt to share data in order to smoothly conduct transactions or to gain efficiencies in supply-chain management. Conflicting hardware or software standards can

quickly defeat such cooperative efforts. Even if such organisations use exactly the same hardware and software components, they may discover that even small differences in the way database or XML schemas are defined can hamper data sharing; indeed, this is the main reason that even the widespread adoption of schematic standards, whether these be Entity-Relationship Diagrams (ERD), Unified Modelling Language (UML) or eXtensible Markup Language (XML), have not fully delivered on their promises of comprehensive interoperability. Organisations continue to face the choice of giving up on joint data-sharing Projects or of relying on expensive and slow manual data translation. In the 1990s, there was some attempt to overcome these issues by introducing single all encompassing Enterprise Resource Planning (ERP) applications such as SAP from Germany. However, these were found to be too restrictive and all corporations now adopt something of a best-of-breed mix of applications, leading to interoperability issues. Overcoming these difficulties requires *ontologies* and *semantic networks* in order to patch together different data standards and formats in a comprehensible manner as anticipated in the Key Action Item

#### 4. COG'S OVERVIEW ARCHITECTURE

COG's architecture provides a generalized architecture that is flexible, reusable and meets the specific requirements of the specific test cases for COG project, while integrating the mentioned technologies.

The architecture specifies the following:

- Implementation tools
- Recommended platforms
- Integration methods
- A specific software structure
- Initial operational structure

Overall, the architecture is an integration architecture, bringing together multiple systems that require complex interactions. The current configuration is not a coherent architecture, but instead are simply a number of standalone systems. The Architecture that is used for the COG can be seen to the following diagram.

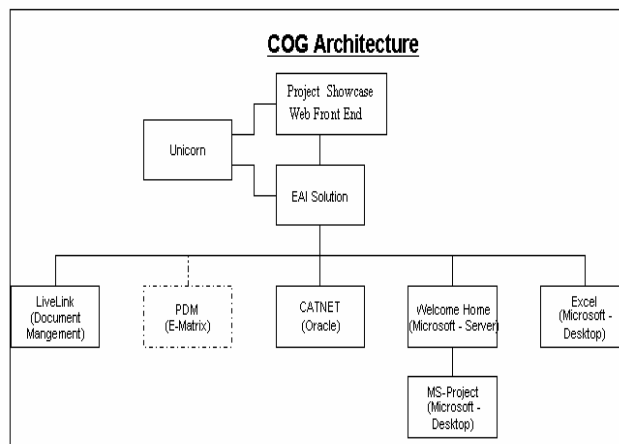


FIGURE 2: COG Architecture

As we mentioned before and as we can see in the above figure COG integrates different modules such as Unicorn system, a web front end Portal, an Enterprise Application Integration solution and the particular systems of FIAT's industry.

- **Unicorn system** is a standalone software system that provides a complex management of enterprise data and support of Central Information Model. It combines rich abilities in information modeling (Protégé) with power tools of metadata analyzing and generation (ERWin, SilverRun, MSVisio). It enriches these with semantic relations between concepts of information model and entities of metadata structure allowing the new approach to enterprise data management - Semantic Information Management. It also supports integrity checking, data discovering and transformation, publishing, sharing and synchronization. It has also its own portal with EJB components.
- **Web front End Portal** is a java based customized portal to view based on needs, roles and permissions. It enables the virtual view of the data for knowledge navigation.

- **An EAI (Enterprise Application Integration) tool** based on web services in order to integrate Unicorn system and other components within COG project.
- The particular systems and databases from FIAT's industry such as PSI database, CATNet database, WelcomHome (Project Management System), LiveLink (Document Management System)

For that reason we have presented a specific e-learning platform, which is based on grid technologies. Grid technologies are getting so much noise last year, because it's a fascinating idea, which the goal is to provide unlimited power and access to vast amounts of information from any computer. The main aim of our proposed grid platform is to provide the basis for user-friendly access to information and communication services. So it will enable both business and private users, whether they are mobile or stationary, to benefit from services and applications provided through the internet.

## 5. INTEGRATION METHODS AND EAI TOOLS CONSIDERATION

Enterprise data resources can take many different forms: Relational Database (RDB) tables, XML documents, Electronic Data Interchange (EDI) messages, COBOL records and others. Independent Software Vendor (ISV) applications — such as Enterprise Resource Planning (ERP), Customer Relationship Management (CRM) software, or home-brew software — define their own input and output schemas.

Often, the different schemas hold similar information, though the structure of the information may differ. Enterprise Application Integration (EAI) systems must overcome the problem of heterogeneous schemas. To do this, EAI implementers must fully understand the semantics of each schema, since the schemas often represent the decisions of a single application developer working from specifications of a single step in a process. Ultimately, new schemas multiply as requirements change, and the investment in analyzing requirements, as captured in older schemas, is lost.

Today, overcoming this heterogeneity requires significant time, with little return. Developers must create transformation code manually, whether by coding in textual languages, such as eXtensible Stylesheet Language Transformation (XSLT), or by using graphical transformation design tools. While this is tenable with a few schemas, a larger number results in a skyrocketing number of such transformations. The number of transformations goes up as the square of the number of schemas.

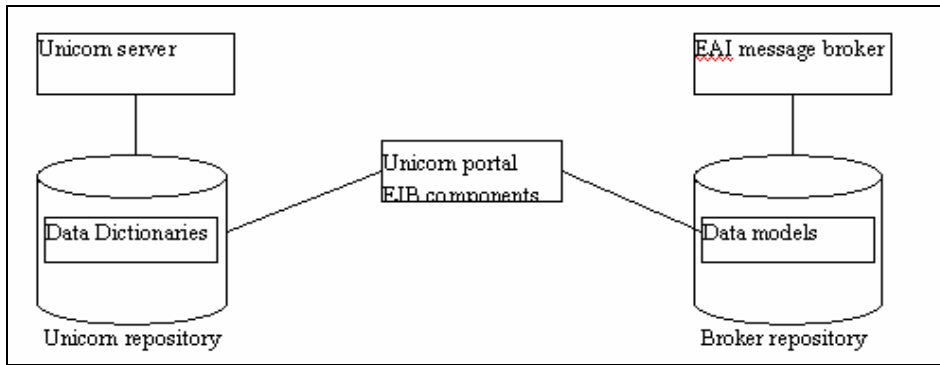
Even if applications are manually integrated at great expense, the solution isn't maintainable. When schemas change, as they often do in a quickly changing environment, existing code for transforming data from one schema to several others becomes unusable.

EAI tools use built-in mechanisms of data transformation and brokers for the transformation process control. They all enable to use XML format for incoming and outgoing information. Internal data formats, which are linked to by means of built-in tools of links determination, are used for the transformation processes control. Usage of adapters enables EAI to set the link with data sources, eliminating intermediate layer of applications.

Unicorn can be used for the creation of the intermediate layer between integrators and applications. Unicorn server is underlied of the specially designed Unicorn Portal with EJB components that provides an interface for interaction with EAI tool.

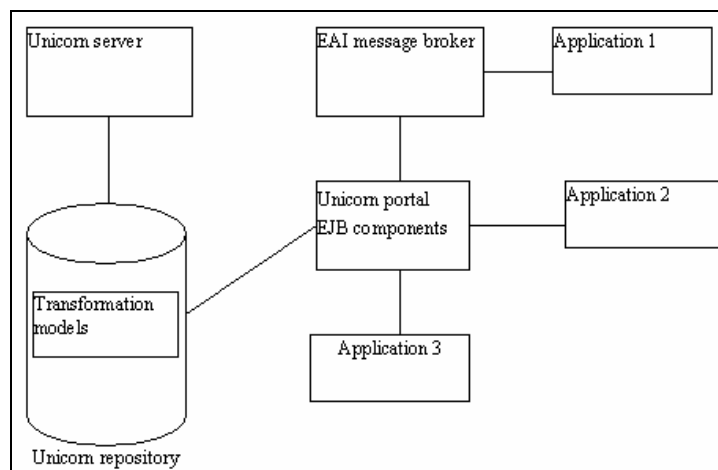
There are three possible ways for the integration with EAI tool.

5.1 Data dictionaries, created in the frame of the semantic model, can be exported to the models, which EAI message broker uses. In this case Unicorn is used more as central storage of metadata in an enterprise. It enables to check the accuracy of data models, to connect them to the enterprise central model. EAI tools use this model for their message brokers, which use transaction model developed in the frame of EAI through the synchronized information model.



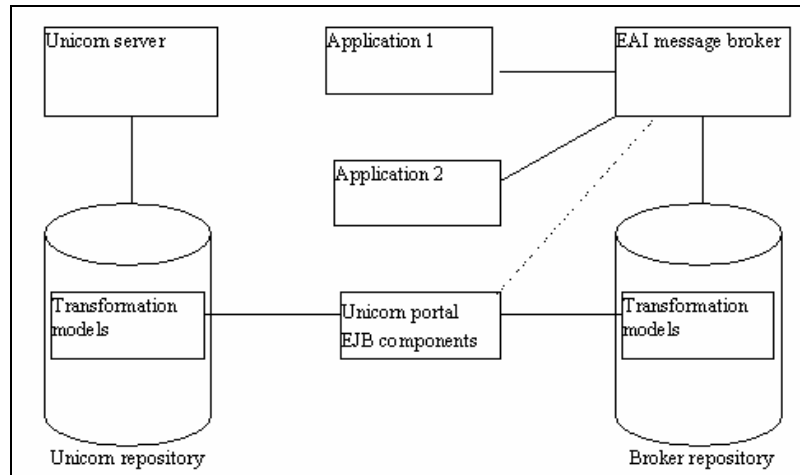
**FIGURE 3:** EJB components of Unicorn Portal translate data dictionaries from Unicorn Repository to Broker repository of EAI tool. All transformation scripts are prepared inside EAI.

5.2 SQL and XSLT transformation scripts can be transferred to EJB components of Unicorn portal in the form of parameters. In this case, EAI sets the link for the data transformation with Unicorn portal components, which get initial data and return transformed data. This way of integration is more resistant to changes – all changes, entered into the enterprise information model, can be compiled immediately for EJB components. However, in this case, there appears one more intermediate layer in the data swap between applications and EAI message broker. This may lead to the essential decline of productivity, as there will be a necessity to swap huge information volumes, earlier processed at the application server.



**FIGURE 4.** EAI message broker receives a signal from Application 1 for data from Application 2 and Application 3. It translates this request to EJB component. The last one searches the corresponding transformational models in Unicorn repository, makes requests to Application 2 and 3, transform the received data and return the result to Message broker. It converts the received data to a comprehensible form for Application 1 and sends it.

5.3 Transformation logic prepared in Unicorn is translated into the format available for message broker. The third possible way is more attractive from the point of operation speed and stability of the system. Logic model of data transformation as XSLT and SQL scripts should be transformed into the form clear to EAI message broker. Specially developed EJB component of Unicorn Portal can control the access to the transformation model and its transformation. In this case, when the message is received broker addresses to Unicorn portal for the new transformation model. This gives possibility to save big volumes of transferred data and automate the system of EAI tools transformation models update.



**FIGURE 5.** EJB components translate SQL and XSLT transformation scripts prepared by Unicorn to the language that Message broker used for transformation and places them to Broker repository. Applications works only with Message broker. After receiving the signal it requests the fresh transformational model from the own repository or directly from EJB components

## 6. CONCLUSION

At present, computational Grids have wide application in industry. This has been limited, however, to the domain of engineering computations where single specific data sources are used. As yet, the required technology to fulfil these aims has not been developed or implemented in corporate settings. The realization of Information Grids using ontologies in an industrial setting, can solve interoperability issues, and enables the interaction between disparate data formats. The Information Grid will tie the disparate data together and comprehend the semantic meaning of the underlying data stored in the diverse data sources. So this type of Information Grid is a pre-requisite for the commercial use of computation Grids. An enterprise must first have uniform access to all its data before any computation can succeed. Indeed this is the key to allowing the progress in Grid technologies to bear fruit in the commercial world.

## REFERENCES

### Acknowledgements

This work is funded by the European Community under the "Information Society Technology" Programme (1998-2002). The authors wish to thank all the partners on this project (*Centro Ricerche Fiat Societa' Consortile per Azioni, Institut für Informatik, Leopold-Franzens Universität Innsbruck, Unicorn Solutions Ltd*)

- [1] COG (Corporate Ontology Grid) European IST project. Contract number: 2001-38491
- [2] Foster, I. and Kesselman, C. (eds.). *The Grid: Blueprint for a New Computing Infrastructure*. Morgan Kaufmann, 1999.
- [3] I. Foster, C. Kesselman, S. Tuecke *The Anatomy of the Grid: Enabling Scalable Virtual Organizations*, *International Journal of Supercomputer Applications*, 15(3), 2001.
- [4] W3C Semantic Web Activity Statement, <http://www.w3.org/2001/sw/Activity/>
- [5] J. Hendler, *Agents and the Semantic Web*, *IEEE Intelligent Systems Journal*, March/April 2001 (Vol. 16, No. 2), pp. 30-37.
- [6] B. McBride, "Four Steps Towards the Widespread Adoption of a Semantic Web", in *Proceedings of the First International Semantic Web Conference (ISWC 2002)*, Sardinia, Italy, June 9-12, 2002. LNCS 2342, pp 419-422.
- [7] I. Horrocks, *DAML+OIL: a reason-able web ontology language*, in *Proceedings of EDBT 2002*, March 2002.
- [8] *Resource Description Framework* <http://www.w3.org/RDF>
- [9] *OWL Web Ontology Language 1.0 Reference*

<http://www.w3.org/2001/sw/WebOnt/>

[10] K. Czajkowski, S. Fitzgerald, I. Foster, C. Kesselman. *Grid Information Services for Distributed Resource Sharing* Proceedings of the Tenth IEEE International Symposium on High-Performance Distributed Computing (HPDC-10), IEEE Press, August 2001.

[11] A. Chervenak, E. Deelman, I. Foster et al *Giggle: A Framework for Constructing Scalable Replica Location Services*, SC2002, November 11-16, 2002, Baltimore, Maryland.

[12] A. Rajasekar, M. Wan and R. Moore, *MySRB & SRB -Components of a Data Grid*, The 11th International Symposium on High Performance Distributed Computing (HPDC-11) Edinburgh, Scotland, July 24-26, 2002.

[13] P. Watson, *Databases and The Grid*, UK e-Science Programme Technical Report Number UKeS-2002-01.

[14] N. Paton, M. Atkinson, V. Dialani, D. Pearson, T. Storey, P. Watson, *Database Access and Integration Services on the Grid*. UK e-Science Programme Technical Report Number UKeS-2002-03.

[15] I. Foster, C. Kesselman, J. Nick and S. Tuecke, *The Physiology of the Grid: Open Grid Services Architecture for Distributed Systems Integration*, GGF4, Feb. 2002. See <http://www.globus.org/research/papers/ogsa.pdf>

[16] C. Wroe, R. Stevens, C. Goble, A. Roberts, M. Greenwood, *A suite of DAML+OIL Ontologies to Describe Bioinformatics Web Services and Data*, International Journal of Cooperative Information Systems *in press*.