

## Grid-VirtuE: A Layered Architecture for Grid Virtual Enterprises

Alfredo Cuzzocrea<sup>1,2</sup>, Alessandro D'Atri<sup>3</sup>, Andrea Gualtieri<sup>4</sup>,  
Amihai Motro<sup>5</sup>, and Domenico Saccà<sup>1,2</sup>

cuzzocrea@{icar.cnr.it, si.deis.unical.it}, datri@luiss.it, gualtieri@exeura.it,  
ami@gmu.edu, sacca@{icar.cnr.it, si.deis.unical.it}

<sup>1</sup> ICAR Institute, Italian National Research Council, Cosenza, Italy

<sup>2</sup> DEIS Department, University of Calabria, Cosenza, Italy

<sup>3</sup> CeRSI, Luiss "Guido Carli" University, Rome, Italy

<sup>4</sup> Exeura, Cosenza, Italy

<sup>5</sup> CS Department, George Mason University, Fairfax, VA, USA

**Abstract.** A *grid virtual enterprise* is a community of independent enterprises concerned with a particular sector of the economy. Its members (nodes) are small or medium size enterprises (SME) engaged in bilateral transactions. An important principle of a grid virtual enterprise is the lack of any global "guiding force", with each member of the community making its own independent decisions. In this paper we describe Grid-VirtuE, a three-layer architecture for grid virtual enterprises. The top layer of the architecture, representing its ultimate purpose, is an environment in which grid virtual enterprises can be modeled and implemented. This layer is supported by middleware infrastructure for grids, providing a host of grid services, such as node-to-node communication, bilateral transactions, and data collection. The bottom layer is essentially a *distributed data warehouse* for storing, sharing and analyzing the large amounts of data generated by the grid. Among other functionalities, the warehouse handles the dissemination of data among the members of the grid; it confronts issues of data magnitude with an aging mechanism that aggregates old data at a lower level of detail; and it incorporates privacy-preserving features that retain the confidentiality of individual members. Warehouse information is also used for data and process mining, aimed at analyzing the behavior of the enterprise, and subsequently inducing evolutionary changes that will improve its performance.

**Keywords:** *Enterprise Engineering, Enterprise Information Architecture, Enterprise Information Integration (EII), Enterprise Information Systems (EIS), Enterprise Integration, Enterprise Modeling and Integration*

### 1. INTRODUCTION

A *grid virtual enterprise* is a community of independent enterprises concerned with a particular sector of the economy, such as automotive, fashion, or entertainment. Its members (nodes) are small or medium size enterprises (SME) engaged in bilateral transactions. An important principle of a grid virtual enterprise is the lack of any global "guiding force": Each member of the community makes its own independ-

**A. Cuzzocrea, A. D'Atri, A. Gualtieri, A. Motro, and D. Saccà**

ent decisions, such as the products it offers, the prices it charges, and the members it wishes to do business with.

In this paper we describe Grid-VirtuE, a three-layer architecture for grid virtual enterprises.

1. The top layer of the architecture, representing its ultimate purpose, is an *environment for modeling grid virtual enterprises*. This environment incorporates and expands on features and properties that have been defined in earlier work (particularly, the VirtuE model).
2. The top layer is supported by *middleware infrastructure for grids*, providing a host of grid services, such as node-to-node communication, bilateral transactions, and data collection.
3. The bottom layer is essentially a *distributed data warehouse* for storing, sharing and analyzing the large amounts of data generated by the grid. Among other functionalities, the warehouse handles the dissemination of data among the members of the grid; it confronts issues of data magnitude with an aging mechanism that aggregates old data at a lower level of detail; and it incorporates privacy-preserving features that retain the confidentiality of individual members. Warehouse information is also used for data and process mining, aimed at analyzing the behavior of the enterprise, and subsequently inducing evolutionary changes that will improve its performance.

The focus of this paper is mostly on the top and bottom layers, as these layers incorporate most of the novel aspects of Grid-VirtuE. Section 2 describes the modeling layer, and Section 3 is dedicated to the data warehouse layer. Section 4 places this work in the context of other research, and Section 5 concludes with a brief summary and discussion of future work.

## **2. THE MODELING LAYER**

The modeling environment of Grid-VirtuE is largely based on previous work on the VirtuE model [6,7,8]. In this section we review the features of the VirtuE model that have been adapted for Grid-VirtuE.

### **2.1 Members and the Catalyst**

A *virtual enterprise-breeding environment* (VBE) [1] is a community of business entities that are potential participants in business coalitions. The launching of a new virtual enterprise begins when a VBE player assumes the role of catalyst [5]. The catalyst has a business plan, knowledge of the additional expertise and resources needed to accomplish the business plan, and information on VBE players that can satisfy these needs. The catalyst then chooses the enterprise members from the VBE, and becomes a privileged member of the virtual enterprise, with several exclusive roles, such as changing the membership of the enterprise or modifying its goals (i.e., end

## Grid-VirtuE: A Layered Architecture for Grid Virtual Enterprises

products). In Grid-VirtuE, the catalyst role consolidates several distinct organizational roles, such as broker, planner and coordinator [1].

The members of the new enterprise are independent but have shared interests. They are independent in the sense that they remain autonomous and maintain their own assets. These assets include human, equipment or financial resources, as well as business expertise, such as knowledge about their production and delivery processes. Their shared interests are reflected in that they agree to cooperate with each other to produce joint products that are provided to common clients. After the community had been established, it could evolve because a new member joins or an exiting member departs. This form of evolution provides the virtual enterprise with flexibility and allows it to adapt to new market situations.

### 2.2 Products and the Product Dictionary

In practice, virtual enterprises may produce many different kinds of products. In Grid-VirtuE, we consider only *information* products, of the type that can be delivered over computer networks. Information products are provided by members of the enterprise to their clients. This provision is the ultimate purpose of an enterprise. Information products are also exchanged among the members of the enterprise in the production phase that precedes the provision of a product to a client. We distinguish between two kinds of information products: *content* and *process*. Content is an information item; for example, a specific data table, a specific document, or a specific image. Process is an operation that modifies given content to provide new content; for example, summarization of a data table, encryption of a document, or compression of an image.

All products, contents as well as processes, are classified into product *types*. A product type describes the common attributes of all products of that type. For example, all images could be instances of the content type *Image*, and all compression processes could be instances of the process type *Compression*. We assume that all intensional information (i.e., types and their attributes) is maintained in an enterprise-wide resource called the *dictionary*. This global knowledge resource is available to every member of the enterprise. Every product in the virtual enterprise is an instance of a type described in the dictionary. The purpose of the dictionary is to assure semantic consistency across the enterprise.

### 2.3 Local Inventories and the Global Catalog

The products either used or created by each enterprise member are described in a local resource called *inventory*. Items in the inventory are *instances* of product types described in the dictionary. Among other information, the inventory specifies the *source* and *target* of each product. The source is either *native* or *import*: A native product is produced locally, whereas an import product is procured from another member of the enterprise. The target is either *internal* or *export*: An internal product is an intermediate product used by this member in the manufacturing of other products, whereas an export product may be delivered to other enterprise members. The

**A. Cuzzocrea, A. D'Atri, A. Gualtieri, A. Motro, and D. Saccà**

product *catalog* is an enterprise-wide resource that lists the products that are available for procurement from enterprise members; i.e., it is the union of the products marked “export” in all inventories. Each member *publishes* his list of export products in this catalog and is responsible for keeping it updated. Note that products in the catalog are *instances* of the types in the dictionary. Hence, the dictionary *regulates* the catalog.

## **2.4 Production Plans**

Another designation of inventory products is whether they are *basic* or *complex*. A product is complex if it is derived from other products; otherwise, it is basic. For native complex product, production plans must be provided. A production plan specifies how other contents and processes are combined to derive the new product. In particular, it specifies the dependence of a product on products that must be procured from other members. Production plans are provided for both complex contents and complex processes, and a product may have multiple (alternative) production plans.

## **2.5 Transactions**

Since component products are often obtained from other enterprise members, a procurement mechanism is necessary. Procurement is executed in bilateral *transactions*. A transaction begins when a request for a catalog product (content or process) is sent from one participant to another, and terminates when the request is satisfied. There are two types of transactions in a virtual enterprise. An *external* transaction is a request for a product that is submitted from a client to one of the members of the virtual enterprise. The member processes the request and provides a solution. A member of the virtual enterprise who processes an external transaction acts in the role of a *provider*. To satisfy an external transaction, a provider may decide to purchase products from other members. Such transactions are *internal*. A member of the virtual enterprise who processes an internal transaction acts in a role of a *subcontractor*. The execution of external transactions is the ultimate purpose of the virtual enterprise. Each member of a virtual enterprise may act as a provider on some transactions and as a subcontractor on other transactions.

## **2.6 Times and Logging**

Grid-VirtuE incorporates a concept of time. Time is simply a system-wide clock that is stamped on transactions and other enterprise events. Enterprise events, such as execution of transactions, updates of the catalog, or changes in membership, are recorded in a log. Log records provide essential information on each such event and the time of its occurrence. There are two types of log: local and global. Every enterprise member maintains his own local log, and, in addition, there is an enterprise-wide log. Local logs may be used by members to improve their operations. The global log can be used for studying enterprise-wide performance, as well as for communication among members.

## Grid-VirtuE: A Layered Architecture for Grid Virtual Enterprises

### 2.7 Performance Indicators and Constitutional Rules

Grid-VirtuE allows the definition of *performance indicators*, which are formulas that capture various quantitative characteristics of the virtual enterprise; for example, the enterprise assets, interdependence levels, the number of transactions performed over a period of time, or the average turnaround time.

Another feature of Grid-VirtuE are *constitutional rules*, which are constraints that express behavior standards that are expected. Such rules enable virtual enterprises with different style or flavor; for example, an organization without any competition (similar products are not available from different members), or an organization that resembles a free market. Rules may be established that require members to maintain activity at some minimal frequency, or limit the number of transactions in a given period. Compliance with constitutional rules is monitored and disseminated, but not enforced.

### 2.8 Information Dissemination and Behavior Mining

Altogether, Grid-VirtuE incorporates four global information resources: the product dictionary, which describes products and types; the product catalog, which lists the products available from members; the log, which chronicles the activities of the enterprise; and the constitution, which establishes the conventions of the enterprise. These resources are managed by the catalyst. It is important to note that the information in these global resources is maintained and processed in a manner identical to any other content. Thus, the catalyst can supply other members with products that calculate performance indicators, or assess compliance with constitutional rules. This information is exchanged using the ordinary transaction mechanism of Grid-VirtuE.

The information derived from the enterprise information resources need not be limited to the calculation of performance measures or the validation of constitutional rules. The global log, and to some degree the other information resources, can be subjected to a wide array of data mining techniques. The overall purpose is to study the performance and behavior of the enterprise, and to associate the outcome with various evolutionary decisions. Log mining is further discussed in Section 3.5.

## 3. THE DATA WAREHOUSE LAYER

The choice of a data warehouse [13] for the latent layer of Grid-VirtuE is well-justified, as data warehouses can sustain computational grids [10] efficiently. A data warehouse will thus provide all the necessary support for the functionalities of Grid-VirtuE. In this section we describe the main aspects of the data warehouse layer of Grid-VirtuE.

A. Cuzzocrea, A. D'Atri, A. Gualtieri, A. Motro, and D. Saccà

### 3.1 Data Fragmentation and Dissemination

A *virtual data warehouse* maps its overall data domain onto a set of geographically distributed grid nodes. It then provides a wide variety of functionalities (e.g., data integration, data querying, and metadata services) over the entire distributed environment, while maintaining complete transparency to its users (i.e., users need not be aware of the placement of data). Each Grid-VirtuE enterprise encapsulates such a virtual data warehouse.

In this paradigm, the fragmentation and dissemination of data among the member nodes is an important process, as its efficiency has critical impact on the performance of all ensuing data management and query processing activities.

Fragmentation techniques have been studied extensively in the context of distributed databases [15], and can be classified in two main variants. In data-oriented techniques, the fragmentation process is driven by dimensional parameters of data; for example, the size and number of attributes in relational data, or the depth and fan-out degree in XML data. In *semantics-based techniques*, the fragmentation process is driven by the semantics of data; for example, relational data could be fragmented so that each fragment satisfies a given constraint set expressed as functional dependencies; and XML data could be fragmented so that each fragment stores semantically-related elements (e.g., a “book\_author” element will be stored with its “book\_title”, “year”, and “price” subelements). Most likely, the second variant is more appropriate for the Grid-VirtuE architecture.

### 3.2 Data Alimentation, Acquisition and Registration

In Grid-VirtuE, the data stored in individual nodes, termed *local data repositories*, could be alimanted by stream data sources located outside the grid environment. To this end, the grid provides data acquisition and registration services. These services can be managed in a distributed manner based on the *Universal Description Discovery and Integration* (UDDI) paradigm [2]. This distributed data acquisition and registration mechanism, which can be reasonably intended as a “natural” application of the well-known Web services paradigm to grids, was proposed in our earlier work on the *SensorGrid* system [4], a high-performance system based on a data compression and approximation paradigm, for efficiently representing and querying sensor network data on grids.

### 3.3 Data Aging and Aggregate Information Processing

To deal with the massive amounts of data expected in virtual grid enterprises, Grid-VirtuE adopts a *data aging mechanism*: As time passes, data are progressively aggregated in information blocks of lower level of detail; these blocks, described by patterns, provide succinct description of the original information content. This data aging mechanism provides tangible benefits in the data management and query processing layer of different applications for virtual grid enterprises, including business intelligence, supply chain management, and market analysis.

## Grid-VirtuE: A Layered Architecture for Grid Virtual Enterprises

In an earlier work [3] we used a similar grid-based approach for handling massive amounts of multidimensional data repositories generated by sensor networks, where streams of data are delivered by sensors that monitor environmental parameters of a given geographical area, and we demonstrated the advantage of data aging and aggregate information processing. Here, we argue the applicability of these methods in “business” contexts as well, for example for supply chain management.<sup>1</sup>

### 3.4 Preserving Privacy

Preserving data privacy has become an important issue in all environments where data is collected, maintained and accessed (e.g., repositories of census, medical, or financial data). Grid architectures, such as Grid-VirtuE, are naturally prone to privacy breaches, as malicious members could join the virtual enterprise only to extract sensitive information. Hence, an important issue in Grid-VirtuE is to preserve the privacy of the local data repositories (the components of the virtual data warehouse).

To support privacy preservation, members retrieve information from local data repositories under an *accuracy-privacy contract*. Such contracts are based on the generally accepted trade-off between data accuracy and data privacy: Data privacy increases as its accuracy decreases, and vice versa. Thus, when data mining tools generate accurate data (i.e., data that is specific and of high quality), they are also likely to breach the privacy of the data (i.e., disclose information that should be protected). An accuracy-privacy contract balances these concerns, by specifying the maximal level of accuracy of delivered data that meets the privacy concerns of the owner of the data.

To meet these privacy concerns, schemes must be devised that represent data in various degrees of accuracy (i.e., at different levels of specificity), to meet different levels of required privacy. Obviously, different approximation (“compression”) techniques must be devised for different types of data (e.g., relational, multidimensional, XML).

The collaborative nature of grid virtual enterprises poses further challenges to privacy preservation, as members can extend their individual capabilities for inferring protected information, by forming coalitions with other grid members. Thus, members with knowledge on different, non-overlapping portions of data could combine their efforts, thereby acquiring knowledge to which they are not entitled. The reason for this deficiency is that privacy-preserving schemes that work well against transgressions of a single member might fail under an attack coordinated by multiple members.

### 3.5 Process Mining on Event Logs

As already explained in Section 2.8, Grid-VirtuE enterprises continuously log their business activities. These event logs can be used for computing various performance indicators and for extracting useful knowledge by means of process mining techniques.

---

<sup>1</sup> Like sensor networks, a supply chain management based on Radio Frequency Identification (RFID) produces continuous data streams.

**A. Cuzzocrea, A. D'Atri, A. Gualtieri, A. Motro, and D. Saccà**

Knowledge thus extracted can then be used in ex post analysis for re-engineering the activities of the virtual enterprise, or for constructing advanced deployment scenarios. Grid-VirtuE members can perform process-mining activities in a *collaborative* manner, and can thus benefit from the possibility of correlating different process mining results. As grid executions are easily modeled with workflows (e.g. [9]), the workflow mining techniques we proposed in [12] can be adapted for Grid-VirtuE.

## 4. BACKGROUND

Virtual enterprises and grid architectures have been researched quite extensively, so we only review here briefly efforts that combine both technologies.

The convergence between grid technologies and virtual enterprises has been introduced by Foster *et al.* [10], which asserted that logical components of a virtual enterprise can be mapped onto technological components that are grid nodes or software components (e.g., resources, protocols, or middleware). This work has influenced heavily all subsequent initiatives in grid computing research, with particular emphasis on computational issues; in contradistinction, our aim is to realize this philosophy in the rather different domain of virtual enterprises.

Lican *et al.* [14] extend these concepts towards the definition of a *Virtual and Dynamic Hierarchical Architecture* (VDHA) for *e*-science grids. VDHA incorporates certain P2P characteristics, and efficiently supports scalable and highly dynamic virtual architectures for *e*-science computing. Compared with our approach, VDHA is specifically tailored to high performance computing environments, whereas we are mostly concerned with business intelligence environments. However, VDHA and Grid-VirtuE share many points in common, as the idea of supporting *dynamism* within virtual enterprises.

In [11], Foster *et al.* propose models and architectures for efficiently implementing *virtual data grids*, a novel computational paradigm aimed at sharing available computational and data resources on grids, to improve the performance of time consuming activities on very large repositories of data. These architectures are useful in *e*-science settings found in areas such as high-energy physics and astronomy. Although designed for different environments, virtual data grids bear similarities to the data warehouse layer of Grid-VirtuE.

## 5. CONCLUSION

We proposed a layered architecture for grid virtual enterprises, an architecture that combines the features of a virtual enterprise model such as VirtuE, with the operational advantages of a grid organization. Many of the innovations of Grid-VirtuE arise from adopting a distributed data warehouse organization for its underlying level.

Much work remains to be done, and we mention here three interesting research directions. First and foremost is the need for significant case studies, to test the applicability of Grid-VirtuE in real-life settings (as already mentioned, supply chain management may provide a particularly attractive example). Second, in Section 3.3 we



## Grid-VirtuE: A Layered Architecture for Grid Virtual Enterprises

suggested using patterns to reduce the *volume* of data, and in Section 3.4 we suggested using approximation schemes to reduce the *specificity* of data to meet privacy requirements. These are two related forms of data abstraction, and we plan to investigate and exploit their relationship (for example, concise patterns are often achieved at the cost of precision). One of the most promising directions for virtual enterprises is the possibility of evolving them rationally on the basis of their recent history and performance. The third research direction is to develop suitable process mining techniques, and to embed them in a system that will intelligently associate their conclusions with suitable corresponding evolutionary changes.

**Acknowledgement.** This work has been performed within *Interop: Interoperability Research for Networked Enterprises Applications and Software*, European Network of Excellence IST-508011 (<http://interop-noe.org>), and has been partially supported by the SFIDA-PMI project.

## REFERENCES

1. H. Afsarmanesh, and L.M. Camarinha-Matos, A Framework for Management of Virtual Organization Breeding Environments, *Proceedings PRO-VE 05: Collaborative Networks and their breeding environments, IFIP 6th Working Conference on Virtual Enterprises*, pp. 35–48, (2005).
2. F. Curbera, M. Duftler, R. Khalaf, W. Nagy, N. Mukhi, and S. Weerawarana, Unraveling the Web Services Web: An Introduction to SOAP, WSDL, and UDDI, *IEEE Internet Computing*, Volume 6, Number 2, pp.86–93, (2002).
3. A. Cuzzocrea, F. Furfaro, E. Masciari, D. Saccà, and C. Sirangelo, Approximate Query Answering on Sensor Network Data Streams, A. Stefanidis, S. Nittel (eds.), *Sensor-Based Distribution Geocomputing*, CRC Press, pp. 53–72, (2004).
4. A. Cuzzocrea, F. Furfaro, G.M. Mazzeo, and D. Saccà, A Grid Framework for Approximate Aggregate Query Answering on Summarized Sensor Network Readings, *Proceedings of the 1<sup>st</sup> International Workshop on Grid Computing and its Application to Data Analysis*, LNCS Volume 3292, pp. 144–153, (2004).
5. A. D’Atri, Organizing and Managing Virtual Enterprises: the ECB Framework, *Proceedings of 4<sup>th</sup> IFIP Working Conference on Virtual Enterprises*, pp. 171–178, (2004).
6. A. D’Atri, and A. Motro, VirtuE: Virtual Enterprises for Information Markets, *Proceedings of 10<sup>th</sup> European Conference on Information Systems*, pp. 768–777, 2002.
7. A. D’Atri, and A. Motro, VirtuE: A Formal Model of Virtual Enterprises for Information Markets, *Journal of Intelligent Information Systems*, Springer Science, to appear (online at DOI 10.1007/s10844-006-0013-0), (2007).
8. A. D’Atri, and A. Motro, Evolving VirtuE, *Proceedings of 8th IFIP Working Conference on Virtual Enterprises*, to appear, (2007).

**A. Cuzzocrea, A. D'Atri, A. Gualtieri, A. Motro, and D. Saccà**

9. E. Deelman, J. Blythe, Y. Gil, C. Kesselman, G. Mehta, K. Vahi, K. Blackburn, A. Lazzarini, A. Arbre, R. Cavanaugh, and S. Koranda, Mapping Abstract Complex Workflows onto Grid Environments. *Journal of Grid Computing*, Springer Science, Volume 1, Number 1, pp. 25–39, (2003).
10. I. Foster, C. Kesselman, and S. Tuecke, The Anatomy of the Grid: Enabling Scalable Virtual Organizations, *Journal of High Performance Computing Applications*, SAGE Publications, Volume 15, Number 3, pp. 200–222, (2001).
11. I. Foster, J.-S. Vöckler, M. Wilde, and Y. Zhao, The Virtual Data Grid: A New Model and Architecture for Data-Intensive Collaboration, *Proceedings of the First Biennial Conference on Innovative Data Systems Research*, online edition (available at [www-db.cs.wisc.edu/cidr/cidr2003/program/p18.pdf](http://www-db.cs.wisc.edu/cidr/cidr2003/program/p18.pdf)), (2003).
12. G. Greco, A. Guzzo, G. Manco, and D. Saccà, Mining and Reasoning on Workflows. *Transactions on Knowledge and Data Engineering*, IEEE, Volume 17, Number 4, pp. 519–534, (2005).
13. W.H. Inmon, *Building the Data Warehouse, 2<sup>nd</sup> Edition*, John Wiley & Sons, New York, NY, USA, (1996).
14. H. Lican, W. Zhaohui, and P. Yunhe, Virtual and Dynamic Hierarchical Architecture for E-Science Grid, *International Journal of High Performance Computing Applications*, SAGE Publications, Volume 17, Number 3, pp. 329–347, (2003).
15. M.T. Özsu, and P. Valduriez. *Principles of Distributed Database Systems*, Prentice-Hall, Englewood Cliffs, NJ, USA, (1999).