A Grid Architecture for Manufacturing Database System

Laurentiu CIOVICĂ, Constantin Daniel AVRAM
Economic Informatics Department, Academy of Economic Studies Bucharest, ROMANIA
laurentiu.ciovica@gmail.com, costin.avram@gmail.com

Before the Enterprise Resource Planning concepts business functions within enterprises were supported by small and isolated applications, most of them developed internally. Yet today ERP platforms are not by themselves the answer to all organizations needs especially in times of differentiated and diversified demands among end customers. ERP platforms were integrated with specialized systems for the management of clients, Customer Relationship Management and vendors, Supplier Relationship Management. They were integrated with Manufacturing Execution Systems for better planning and control of production lines. In order to offer real time, efficient answers to the management level, ERP systems were integrated with Business Intelligence systems. This paper analyses the advantages of grid computing at this level of integration, communication and interoperability between complex specialized informatics systems with a focus on the system architecture and data base systems.
Keywords: *enterprise resource planning, architecture, grid computing, data base systems.*

1 Introduction

The first attempts to computerize the business functions within an enterprise have been materialized in the 60s, in small and isolated applications, developed internally. These applications were specifically designed to provide support for financial-accounting operations and for stock management in order to streamline the production process.

Due to rapid changes inside companies and to new requirements, existing software products continuously needed adaptive and progressive maintenance. Their advantages allowed expansion in other business areas, where they interacted with other applications even though these where not designed to interact and work together. The evolution process continued and the software products required more maintenance, expansion of functionalities and integration. Application packages have emerged being added to existing software collections. On this background, in the late '80s the ERP - Enterprise Resource Planning - systems were born.

The offered solutions were integrated into a single package. The request of such software products was huge and soon the ERP's were everywhere. In [1] and [2] are described the most important advantages of ERP systems. With a single and centralized database system the organization has access to **high quality information** obtained from consistent, accurate and elementary data. Databases' normalization within ERP systems and the usage of a single and central database, **minimize data redundancy**.

The ERP systems provide **small response times**. They can be easily reconfigurable and they allow adaptation to permanent changes in today's volatile economic environment. Process flows, applications, solutions and functional modules are integrated in the EPR system.

Being module oriented the systems allow to add new functionalities and their integration and configuration are possible with acceptable efforts. This gives to ERP systems the advantage of **scalability.** The complexity of ERP applications but also the need of a continuous improvement led to the integration of the maintenance service, offered by suppliers, inside the development process. The result is an **improved maintenance and support system**.

The evolution of ERP products to “ERP-extended" concept, gives them a **collaborative dimension** by integrating applications such as Advanced Planning and Scheduling - APS, Customer Relationship Management - CRM, Supply Chain Management - SCM, Product Lifecycle Management - PLM, Supplier Relationship Management - SRM.

Inside manufacturing organizations the ERP systems are integrated with the production systems, known as Manufacturing Execution Systems. The objective is to better plan and control the production unit.

In today's ERP systems there is a MRP, Material Requirement Planning module that is managing the production cycle, but is not a production dedicated systems.

Manufacturing Resource Planning systems represents the foundation of today's ERP systems. As described in **Fig. 1**, the development of MRP and ERP systems started almost in the same time.

There have also been developed and integrated decision support applications at the top level management known as Business Intelligence, BI. Through the development and integration of web-oriented components, the ERP systems offer the advantage of openness to e-business. This level of integration and communication is possible because of Service Oriented Architectures, SOA, which are focused on the entire enterprise life cycle.

The concept is called Enterprise Service Oriented Architecture or simpler ESA, Enterprise Service Architecture. Enterprise Service Oriented Architectures allow manufacturing organizations to optimize the production lines and to develop new products personalized and adapted to their customers' needs.

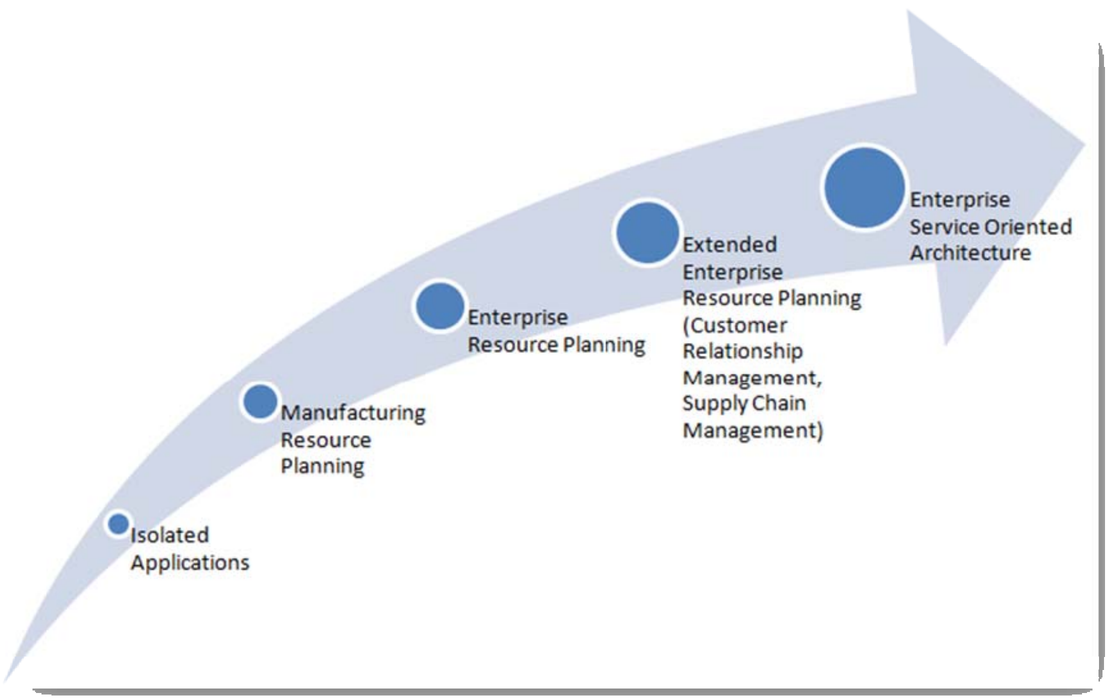

Fig. 1. Enterprise software evolution

According to **Fig. 1,** standard, isolated software applications that were using the power of mainframe computers were integrated in unique software systems for better communication and interoperability, giving life to the ERP systems.

The ERP systems became an imperative requirement especially for the manufacturing enterprises were specialized systems for production management were being developed - Manufacturing Resource Planning. This phase of software integration in single platforms was supported from the technical point of view by client-server architectures as described in **Fig. 2**. The evolution of software integration didn't stop here as ERP concept was evolved in extended ERP by integrating it with specialized systems for the management of customers and vendors in order to optimize even more the product life cycle management by eliminating delays caused by stock failure or raw material failure.

The final stage of software integration was born thanks to web service technologies and is represented by Service Oriented Architectures and Enterprise Service Oriented Architectures.

Grid computing and cloud computing technologies are already empowering these architectures in ways that will allow production industries to become capable of real-time equipments configuration and transformation in order to execute differentiated and personalized products.

2 Grid computing inside ERP architecture

The ERP systems are transactional applications, client-server and distributed. Servers are often centralized, but the customers are spread throughout the organization. Multinational organizations consolidate their servers into a single point, e.g. the headquarters of the organization, and can access this point from around the world. A general ERP architecture includes three functional entities. The database component is the central repository of transactional data both within and outside the organization. The second component consists in the applications client from where queries on database are initiated and where the processed results from these queries are sent. The third component is an intermediate zone between clients and database and is represented by the ERP internal applications. The ERP systems can be implemented as a two-tier architecture meaning that there is a single level which handles and manages the database and the applications, or a three-tier architecture can be implemented where, in addition to the client level, there are two more levels, one for database and a separate one for applications. Inside two-tier architecture, the database server and applications server are physically installed on the same equipment but remain logically different entities. **Fig. 2** describes three-tire architecture.

The techniques of grid computing are a higher growing concern in the informatics systems world and especially in the case of integrated ERP systems. In [3] we find an ERP architecture based on the model Open Grid Service Architecture, OGSA. The author states that this architecture solves the inefficiency problems of sharing the distributed resources through the benefits of the OGSA model, through the web services, by using the grid computing techniques and through the use of XML and SOAP protocols, in order to integrate core business services of the organization.

The grid computing techniques are found in the components of ERP systems and also at the level of ERP platform itself. The databases used by ERP systems are scalable and allow the storage and retrieval of any type of data in a grid environment. Data are saved in a single virtual group that is shared and made available depending of the competing demands of users, ensuring also the optimization of memory used and avoiding locking the requests between them. Databases are distributed in several physical servers and grouped at logical level into clusters, providing scalability without affecting applications that use the database system. The organization of database servers into clusters enables the possibility to extend the server capacities, by adding new nodes in high demand areas and eliminates the risk of a single point of failure. User applications have permanent access at the fair and efficient level of resources and computing power through management functions in terms of work load. Not only can the database of ERP systems be distributed on multiple physical servers, but also the application level.

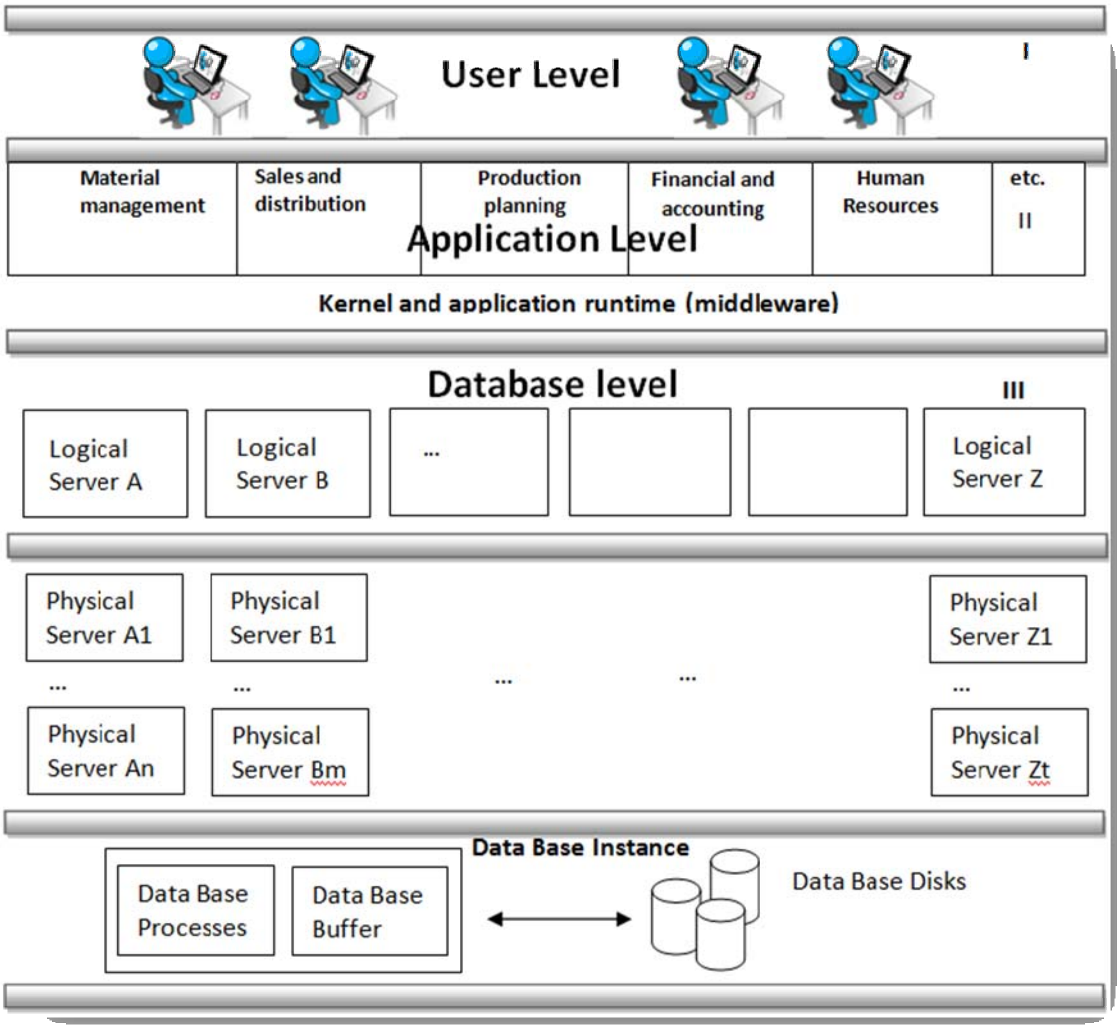

Fig. 2 ERP Architectures

The application server is accessed by the end user as a single point, but planning systems and optimizing of end-user access to ERP system resources, optimally redirects the requests on multiple applications servers. At the programming level, it can be transmitted on which application server will run a particular query. In retrieving data from ERP databases, the authors experienced interfaces that extracted in a work session tens of millions of records in order to transfer them to data warehouses. The interfaces were running during the night, but in many cases, the amount of data was so big that twelve hours were not sufficient to complete the execution, which was unacceptable because during the day the system should have been as free as possible for the key users. The problem was solved by applying grid computing techniques. Assessments and estimations of the volume of data were made and regarding their homogeneity, so that the final result to be divided into partial results.

The advantage of this technique allowed the execution of multiple partial jobs, each one using the resources of a different application server. In **Fig. 2** the existence of an intermediate level between the database

server and the applications server can be observed.
Due to the database distribution and of the applications on multiple servers, an intermediate level is necessary to ensure optimal communication between the two levels. The end user is not involved in this phase. His requests, launched from clients all over the world, are taken over and managed with maximum efficiency by the grid architecture in a non transparent way for the end user.

3 A grid software architecture for manufacturing organizations

The authors think that customers today are more and more oriented on personalized products and they want to participate in an active way during the phase of product design.
In particular, fashion industry shows an increased interest in customers' opinion and the trend is to design and develop personalized fashion products matching perfectly clients' demands.
Personalized demands can be limited to some characteristics such as measure, or can even include personalized product design.
Pomarfin Company, a Finland leading manufacturer for casual footwear is known for its "left foot" concept. The idea is to have 3D scanners available inside their stores in order to obtain the perfect measure for each customer. Clients will choose the model and thanks to the 3D scanners the shoe model they like most will come in perfect size, especially for them.
This paper is extending the "left foot" concept by assuming that very soon 3D scanners will be a technology available for everybody.
Customers will design, from their home, the perfect show model, they will obtain their exactly measure using personal 3D scanners and they will order via internet their products. Software architecture for the development and implementation of such business model is described in **Fig. 3.**
It's a grid architecture because different software components with dedicated data base systems are working together to serve a specific customer request. The grid is controlled by a central element, a Service Proxy containing:

- a message service used for sending messages between the production grid components
- a web service mediation used to convert different messages from different components into a general language; it can be associated with a common protocol or a communication standard
- an enterprise service bus used to receive data from third parties components and systems using different types of protocols

The grid architecture contains a web portal which helps end customers to design their personalized products and place the order inside the manufacturing system.
It contains ERP systems with all its components and applications. The data base system used for the ERP system is using a grid model as described in **Fig. 2**.
The ERP system is integrated with Customer Relationship Management Systems and Supplier Relationship Management Systems.
Different data from different data base systems are grouped in data warehouses.
The ERP system in connected to the production systems used at shop floor lever.
Between the two systems there is a middleware component, the manufacturing execution system.
As described in [4] the manufacturing execution system is responsible with the management of production. It represents a communication layer between the ERP as transactional system and the systems dedicated to production lines.
The Business Process Management inside our grid architecture enables the organization to define executable processes using visual flow steps.
It allows process adaptation to new market requirements and ensures a competitive

advantage for the enterprise.
This component is essential for our grid architecture as the main purpose of this platform is to offer real time transformation and configuration of the business model due to the products' differentiation.
The Enterprise Decision Management component represents a tool for managing business rules which management of the business model inside the software architecture.

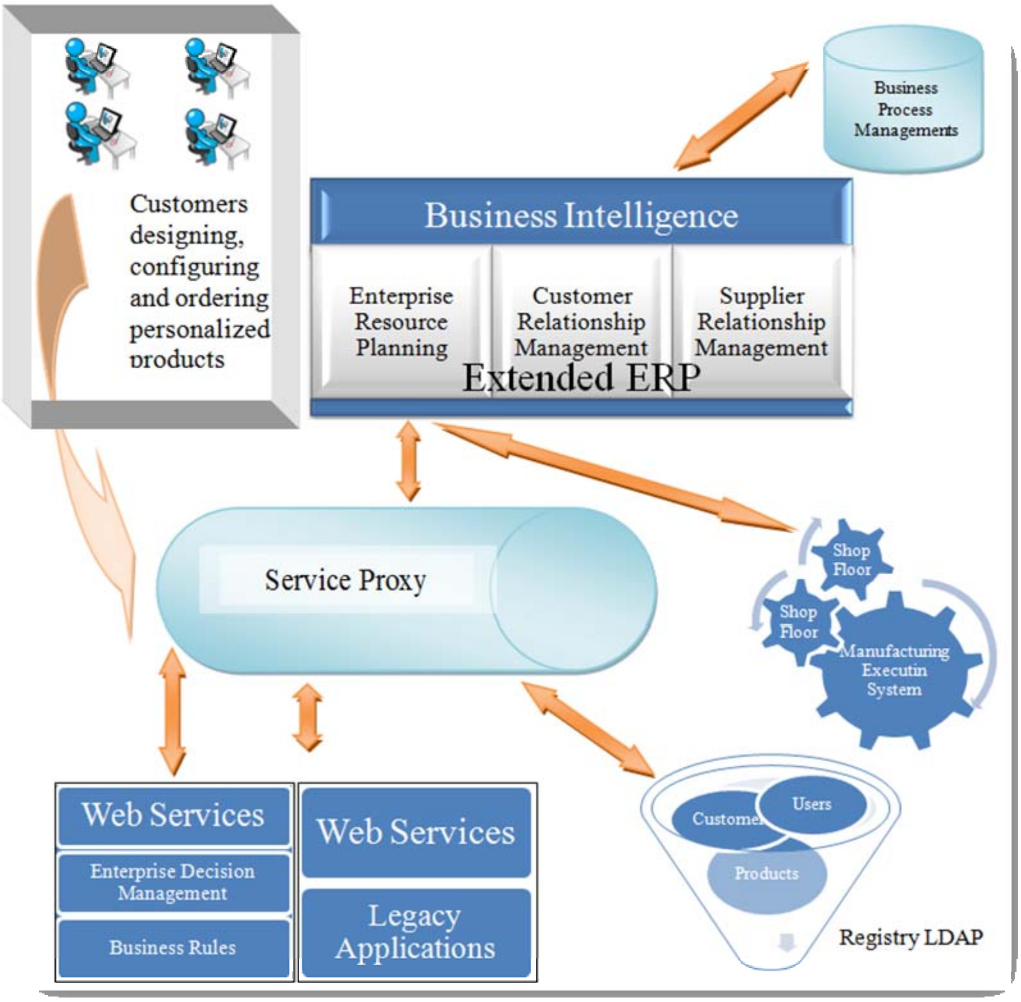

Fig. 3 A grid software architecture for manufacturing organizations

A detailed analysis of the database systems must be undertaken considering the complexity and the number of specialized software applications integrated in a single software architecture, in order to allow more interoperability.
Each component described in **Fig. 4** works with a specific data base model and a database system must be designed for the entire architecture in order to ensure its stability and efficiency.

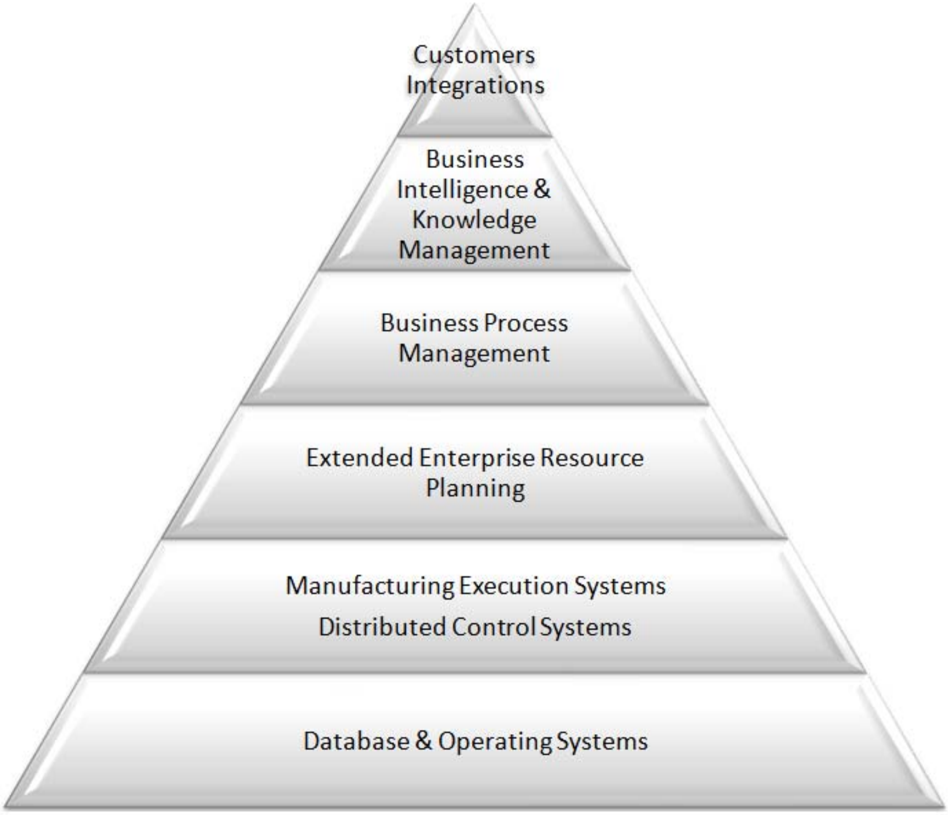

Fig. 4 Components of grid software architecture for manufacturing organizations

4 Data Base Systems Architecture – An overall view

As mentioned earlier in article, the customers will order their desired product through a web application capable in interacting with 3D scanners.

Beside the scanning functionality the application will expose functionalities like: getting all available products along with their models directly from manufacturers, getting all applicable colors, shapes, sizes, etc – basically everything that can be customized and is supported by the manufacturer, the possibility to see the order status in every moment, the possibility to assign notifications channels through which the system will notify the user about any problems that may occur with their order, etc.

The user will interact with a highly usable interface through which in matter of minutes he can choose and customize his desired product and place the order.

From architectural point of view the application itself interacts with third-party applications, services, systems in a transparent manner for the user.

Basically the portal – the web application with which the user interacts – is just a client for the main application server which coordinates the entire order flow and interacts with all necessary third-parties systems.

From database system architectural point of view the system communicates with multiple database servers, as shown in **Fig. 5** – one server per a major functionality domain: e.g customer's database server, product information system database server, orders database server, etc.

In order to handle the huge number of customers the database servers are distributed and a load balancing router is implemented.

In order to avoid data looses and corruption the distributed systems are in sync by using native DBMS replication and specialized data sync applications.

As shown in **Fig. 5** the application server uses the CDB – Customers DB, ODB –

Orders DB and PDB – Product Information System DB as primary data sources. These sources are shared within the E-ERP itself.

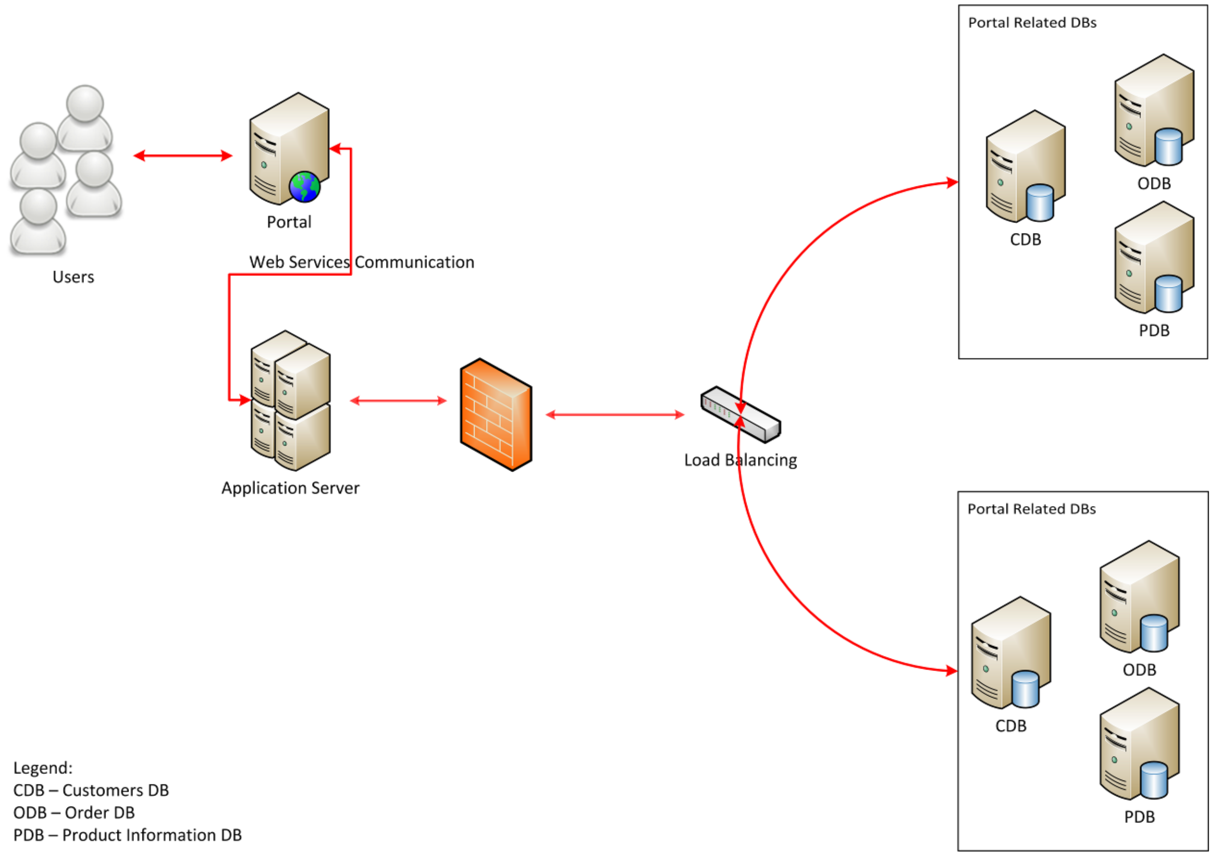

Fig. 5 DB System architecture - overall view

As shown in **Fig. 6** where a database system design is described – a system is composed from one master database along with its mirror database and slave databases.

The idea behind this design is that all the update requests are redirected by a router to the master database and all those read requests to the slave's databases. In order to keep the data in sync inside the system, native replication processes are used. For backup purposes the mirror database is a clone of the master database. The cloning process can be done once or multiple times per day.

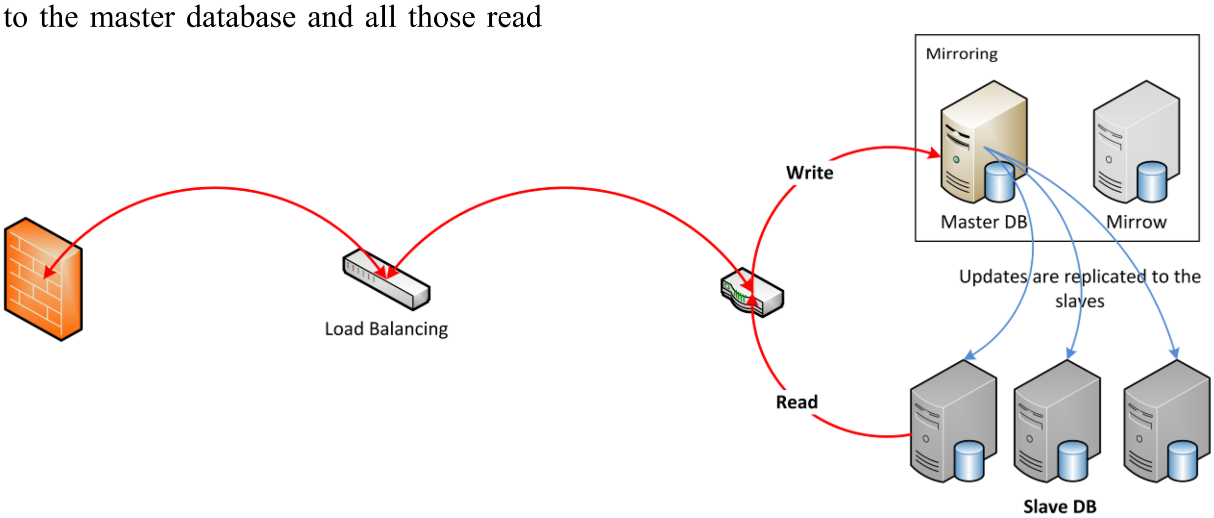

Fig. 6 Database system architecture

For our system beside the customer's databases the most important and vital data source is the one from the Product Information System. This data source contains all those information related to products, their models, sizes and other ways of customization.

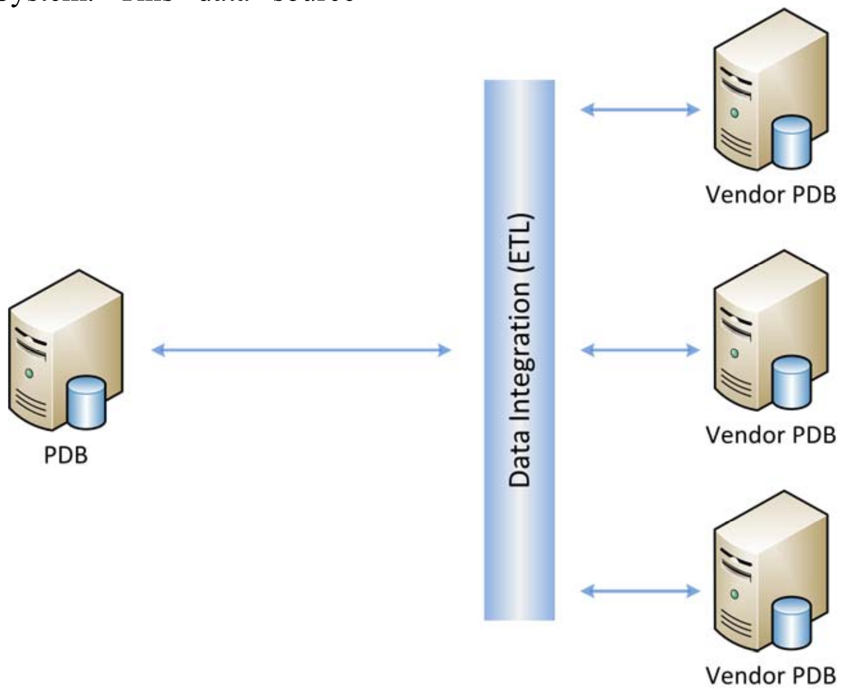

Fig. 7 Product Information System Database Server

As show in **Fig. 7** the data is imported from third-party vendor's database servers. The data importing and synchronization process is done by a specialized application capable in extracting, transforming and storing the necessary and required data.
This application translates the data acquired from vendors in a format understood by our application server and its modules.

5 E-ERP Database System Architecture

As presented earlier in article the proposed ERP - Extended has in its structure specialized modules like: Customer Relationship Management - CRM, Supply Chain Management - SCM, Business Intelligence and Knowledge Management – BI & KM, Manufacturing execution system – MES, and others.
Each module has its own customized database system. The required data is acquired from other data sources inside the entire application system.
We are presenting in **Fig. 8** the main modules and their interactions. Thereby the MES DB contains data acquired by extracting and transforming data from database systems such as: PDB and CDB.
The CRM DB contains data from systems such as ODB and CDB.
The BI & KM module uses data from almost all the existing systems – as show in **Fig. 8** – from systems such as ODB, CDB, PDB, MES DB, CRM DB and so on.
Each module uses specialized application capable in extracting and transforming all the necessary and required data from other internal / external database systems.
The proposed architectural design allows a big database server distribution factor. Thus each database server can be hosted on a single machine, everywhere. Besides this feature, the proposed architectural design eases the use of grid computing design in order to increase performance and speed.

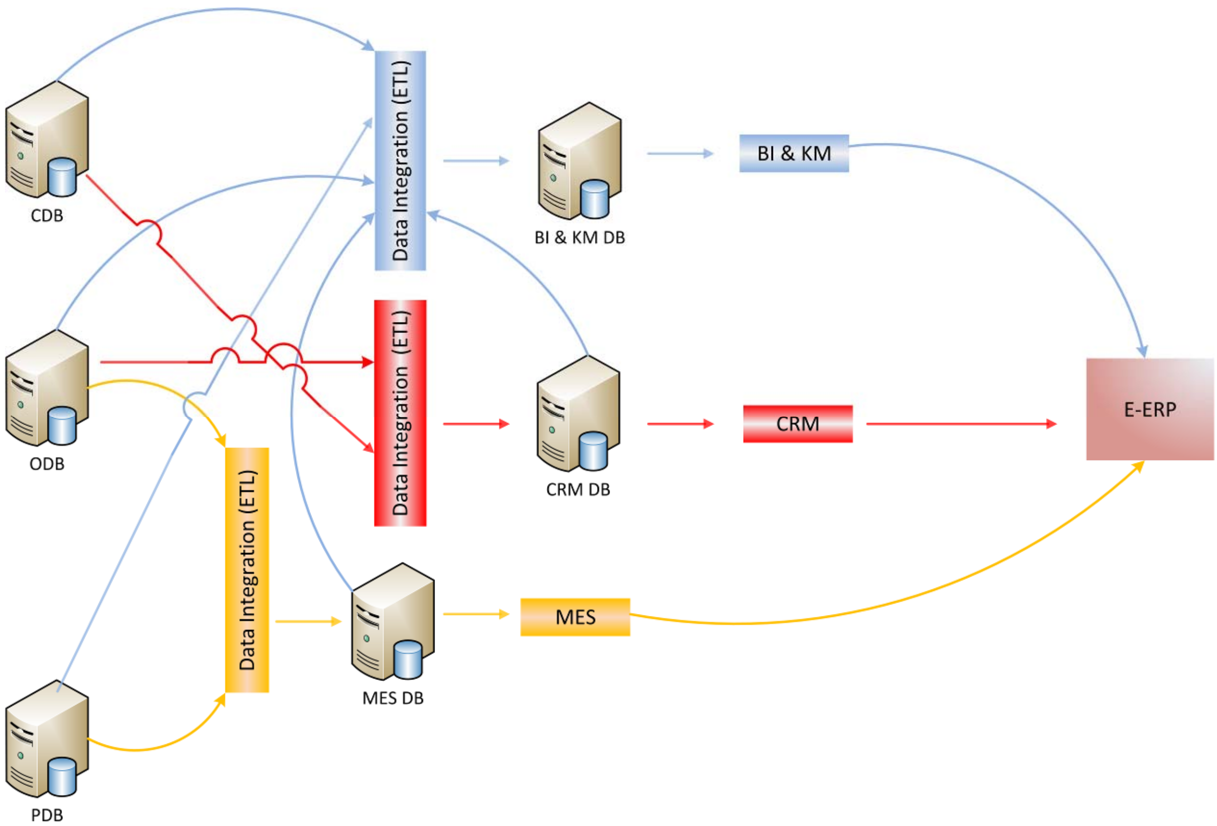

Fig. 8 E-ERP Database System Architecture

6 Conclusions

Grid technologies allow a full integration and information partitioning by providing access to several applications to the IT infrastructure of the organization instead of requiring dedicated servers to be installed for a sole application with a well defined purpose.

By using grid computing the software architecture proposed for the manufacturing organizations, allows dynamic grouping in computer sets, application servers and databases in order to ensure an efficient support to volatile needs in the business environment. ERP platforms become simultaneously more flexible, scalable, with an increased performance level and high availability.

Grid technologies allow resource allocation and efficient use of processing power by optimizing the IT infrastructure.

An organization can overcome the critical moments in an IT infrastructure zone using other resources available at that time. Registering an additional cost is avoided in this way by improving IT performances in an area only to overcome a crisis by splitting processing power to different sources.

This kind of split permits replacing components of the IT infrastructure without affecting functionalities of the system. A better organizational management is obtained by the access to a better image of the IT infrastructure and through and a unique resource focused management process is achieved. ERP systems offer today complete grid computing solutions by allowing access to the system using the internet and sharing application servers and databases.

7 Acknowledgments

This work was co-financed from the European Social Fund through Sectoral Operational Programme Human Resources Development 2007-2013; project number POSDRU/107/1.5/S/77213„ Ph.D. for a career in interdisciplinary economic research at the European standards”

References

[1] I. Lungu, A.R. Bologa, V. Diaconita, A. Bara and I. Botha, Integration of Informatics Systems. Buchares, Romania: ASE Publishing House, 2007, pp. 1-297.

[2] L. Hurbean and D. Fotache, Integrated software solutions for business management-ERP. Bucharest, Romania: Economics Publishing House, 2007, pp. 1-264.

[3] *Development of a Grid ERP Architecture: Integration of Grid Computing and Enterprise Resource Planning Application,* accessed on 29/05/2011, http://ieeexplore.ieee.org/search/freesrchabstract.jsp?tp=&arnumber=4681186&queryText%3Derp+grid+computing%26openedRefinements%3D*%26searchField%3DSearch+All

[4] A. Dashchenko, *Reconfigurable Manufacturing Systems and Transformable Factories*. New York, SUA: Springer Publishing House, 2010, pp. 1-759

Laurentiu CIOVICĂ has graduated the Faculty of Science, in 2008 gaining a Bachelor of Science degree in Information Technology with a thesis on Translators and Interpreters for Code Generation and Software Optimization. In 2010 he gained a Master of Management degree in the field of Cybernetics, Statistics and Economic Informatics with a thesis on Intelligent Agents. He is currently a PhD student at Academy of Economic Studies in Bucharest. He is the author and co-author of more than 12 scientific articles in the field of software quality and optimization, code generation techniques, collaborative systems, data bases, programming environments and techniques, mobile platforms and economic informatics systems. Besides the scientific activity he is also an active software developer, being the author of few applications. His area of interests includes among others: software quality, optimization techniques and algorithms, code generation techniques, economic informatics systems, intelligent and collaborative systems, mobile platforms.

Constantin-Daniel AVRAM has graduated the Faculty of Economic Cybernetics, Statistics and Informatics in 2005. He holds a master degree in Computerized Project Management. He is the author and co-author of several scientific articles in the field of ERP systems, manufacturing-oriented informatics systems, software recognition, risk management, software and projects quality assurance, project management. He is currently attending a PhD Program in Economic Informatics. He has more than six years of experience in ERP implementations. He was involved in large and very large international ERP projects.