Modeling Architecture and Reference Models for the ERP5 Project

Renato de Campos[1], Rogério Atem de Carvalho[2], and Ailton Ferreira[3]
[1]São Paulo State University-UNESP, Production Engineering Departament,
Av. Eng. Luiz Edmundo C. Coube nº 14-01,
Bauru – SP, Brazil, rcampos@feb.unesp.br
[2]Federal Center for Technological Education of Campos, R. Dr. Siqueira,
273 – Pq. Dom Bosco – Campos dos Goytacazes, RJ - Brazil,
ratem@cefetcampos.br
[3] State University of North Fluminense, Production Engineering
Laboratory, Av. Alberto Lamego, 2000, Pq. Califórnia, Campos dos
Goytacazes, RJ, Brazil, ailton@uenf.br

Abstract. The adoption of ERPs systems by small and middle-sized companies may not be possible due to their cost. At the same time, when adapting ERP to the company's particular needs, the user keeps depending on the system's sellers due to the lack of access and knowledge of the respective code. Free and open-source software may promote advantages to the enterprises, however, for its adoption it is necessary the development of techniques and tools in order to facilitate its deployment and code maintenance. This article emphasizes the importance of defining modeling architectures and reference models for the development and maintenance of open-source ERPs, in special the ERP5 project.

1 Introduction

Nowadays, the market is more and more competitive, thus the companies must react in order to follow customers demands. For the companies to organize their resources and obtain better planning to their production processes, one of the options is the deployment of ERP (Enterprise Resource Planning) systems [1, 2]. However, the price of these systems may be an obstacle for small and medium-sized enterprises (SME) willing to obtain such systems.

The adaptation of ERP modules according to the characteristics of each company may also become important to its competitiveness, but the closed systems make the

Please use the following format when citing this chapter:

de Campos, R., Atem de Carvalho, R., Ferreira, A., 2006, in International Federation for Information Processing, Volume 205, Research and Practical Issues of Enterprise Information Systems, eds. Tjoa, A.M., Xu, L., Chaudhry, S., (Boston:Springer), pp.677-682.

companies depend on the payment of such services to the system owner-developers. Open source and free software might be an alternative to the SME in order to cut down expenses. Another advantage is the possibility of the software adaptation, allowing for the users to adjust system processes or modules to the reality of their company by means of altering the source code, without being dependent on the owner-developers of proprietary codes. However, it is somewhat difficult to adapt these ERPs, regarding the modification of these codes and the deployment of the systems in an enterprise.

The analysis and documentation of the business and software requirements by means of models are essential for the open source ERPs development, making necessary the use of proper techniques and tools. In this sense, a modeling architecture that properly contemplates the modeling of all aspects of the business processes, including the other aspects related to the software development, can facilitate the reuse, better functionality, better performance, and the better system understanding, avoiding waste of efforts and resources [3, 4]. In the case of free/open source ERP systems the advantage of free code modification can be jeopardized by the lack of references from where specializations of this code can be derived.

This matter is being dealt on the ERP5 project [1]. One of the proposals is the utilization of modeling architecture and reference models, since the documentation and good understanding of business processes and information flow, are essential to facilitate the definition of the enterprise's particular requirements and for the maintenance of the associated codes [5-7].

This paper aims to highlight the need for the definition of a modeling architecture and reference models in order to facilitate the code generation and maintenance of open-source ERPs. Thus, after this introduction, the ERP5 project is briefly presented, and then some comments are made on modeling architectures and reference models

2 ERP5

The ERP5 project [4] is a free-code ERP project which aims at offering a high technology solution at low cost for SMEs. This system uses the open source Zope platform and it is object, workflow and Web based.

The ERP5 architecture incorporates advanced concepts since its inception, such an object database, a content management system, and data synchronization among different sites. It also contains a clear method for process modeling, and consequently, for source code generation. The ERP5 defines an abstract model to business management based on five categories [4]:

- Resource: describes an abstract resource in a given business process (such as individual skills, products, machines etc). Material lists, as well as prototypes are defined by the relationship between the nodes.

- Node: receive and send resources. They can be related to physical entities (such as industrial facilities) or abstract (such as a bank account). Metanodes are nodes containing other nodes, such as companies.

- Movement: describes a movement of resources among nodes, in a given moment and for a given period of time. For example, such movement can be the shipping of raw material from the warehouse to the factory.

- Path: describes how a node accesses needful resources. Paths are abstract and used for planning. Movements are realizations of paths.

- Item: a physical instance of a resource.

ERP5 is based on a model that can connect anything to a category. Some examples include a resource category (such as services, raw material, skills or money) or a company category (such as a group of companies, a group of people or retail chain stores).

3 Modeling Architecture and Reference Models

Vernadat [6] defines the word "model" as an abstraction of the reality expressed by some formalism defined by a modeling method based on the user's objective. Enterprise modeling is related to the following matters: what (it refers to the operations and objects processed by the company), how (it refers to the manner how things are made), when (it provides a notion of time and is connected to the events representing changes in the state of the enterprise), how much (for example, the economic aspects), who (it refers to the human or machine resources or agents) and where (logistic aspects, for example).

Enterprise modeling [8,9] allows not only a better understanding of the company's requirements which will interfere in the systems, but also the identification of alternatives for the several enterprise processes, reducing efforts during information systems development and allowing a better integration between the system development processes [10,11]. For Scheer [5, 6], reference models can be developed in real or theoretical situations, and they document the know-how of a process that can be used by other people. For Keller and Teufell [2] the reference models can be applied to accumulate experience in some type of business or for business process solutions implemented and performed by a management information system, for example.

In the CIMOSA (Computer Integrated Manufacturing Open System Architecture) modeling framework two parts are considered [3, 12]: (i) a particular architecture and (ii) a reference architecture. Particular architecture is a set of models documenting the business environment. Reference architecture is used to help the business users in the construction process of their own particular architecture as a set of models describing the several aspects of the enterprise in different levels of modeling (model instantiation principle). The reference architecture is separated in

two layers: a generic layer providing generic construction blocks (related to the modeling language) and a layer of partial models consisting of a library of classified and reusable partial models for some industrial sector, that is, models that can be adapted to company's specific needs.

In addition to the models Instantiation Principle, the CIMOSA modeling framework has the Derivation and Generation principles.

The Derivation principle models the enterprise according to three successive modeling levels (iterations among these levels are, of course, allowed):

a) requirements definition, in order to express the business needs as realized by the users;

b) specification design, in order to build a formal, conceptual and executable model of the enterprise system (time is considered);

c) description implementation, in order to document implantation details, installed resources, exception management mechanisms and to consider non-deterministic systems.

The Generation principle, which recommends the modeling of the enterprise according to the four basic and complementary viewpoints (other views can be defined):

a) the view of function, which represents the functionality and the enterprise behavior (that is, events, activities and processes) including time aspects and exception management;

b) the view of information, which represents enterprise objects and its information elements;

c) the view of resources, which represents the enterprise means, capacities and management;

d) the view of enterprise, which represents organizational levels, authorities and responsibilities.

As described, modeling architecture and reference models aim to facilitate the work of modeling and to provide a common understanding about the company's systems. A modeling language for the description of the models is needed.

Actually in the ERP5 project, reference models to a Production Planning and Control module are being currently developed using the Unified Modeling Language (UML) [24]. The UML (Unified Modeling Language) is a graphical language for specification, design, visualization and documentation of a software system [11, 13]. The UML 2.0 defines thirteen types of diagrams, divided into three categories, representing different aspects of interactions. The main objective of the UML is to define a visual and expressive modeling language, providing easy visualization, that is, the full understanding of the functions of a system from diagrams representing it. In the communication, the development team's communication is unified and facilitated through the diagrams. From the UML reference models it will be possible to generate code to create a generic system module with generic characteristics. These reference models will be utilized to the system documentation. Then, users will customize the reference models to define the particular models considering the requirements of enterprises. This facilitates the development of particular codes.

4 Final Considerations

An ERP system can help companies in the search for competitiveness, but it is difficult to be adopted by some companies because of the high purchasing cost and the dependence of the developers for possible system adaptations, due to the lack of knowledge and lack of access to its code documentation and models.

Free and open-source ERP systems can be a good alternative. To be adopted, it is necessary to develop and use techniques and tools which facilitate the development and maintenance of such software. Modeling architecture and reference models are essential to make possible the development, implementation and maintenance of the open-source ERPs. To define a modeling architecture for the ERP5 project, the use of the CIMOSA modeling framework concepts and the architecture proposed by Eriksson and Penker [11] is being studied.

The reference models for the ERP5 system modules shall be created to document the possible generic processes and related information, which may be used as base for adaptations (or particularizations) by means of new codes. In the ERP5 Project the UML language was adopted because are worldwide accepted and used, becoming a *de facto* standard language, facilitating the diffusion of models and codes.

References

1. J.P. Smets-Solanes and R.A. Carvalho, ERP5: A Next-Generation, Open-Source ERP Architecture, *IEEE IT Professional* **5**(4), 38-44 (2003).
2. G. Keller and T.S. Teufel, *SAP R/3 Process Oriented Implementation* (Harlow, Addison-Wesley, 1998).
3. F.B. Vernadat, *Enterprise Modeling and Integration: Principles and Application* (Chapman & Hall, London, 1996).
4. M. Odeh and R. Kamm, Bridging the Gap between Business Models and System Models, *Information and Software Technology* **45**, 1053-1060 (2003).
5. A.W. Scheer, *ARIS – Business Process Modeling* (Springer Verlag, Berlin, 2000).
6. A.W. Scheer, *ARIS - Business Process Frameworks* (Springer Verlag, Berlin, 1998).
7. H. Li and T.J. Williams, Management of Complexity in Enterprise Integration Projects by the PERA Methodology, **13**, 417-427 (2002).
8. F.B. Vernadat, Enterprise Modeling and Integration (EMI): Current Status and Research Perspectives, *Annual Reviews in Control* **26**, 15-25 (2002).
9. K. Kosanke and J.G. Nell, Standardization in ISO for Enterprise Engineering and Integration *Computers in Industry* **40**, 311-319, (1999).
10. H. Shen, B. Wall, M. Zaremba, Y. Chen, and J. Browne, Integration of Business Modelling Methods for Enterprise Information System Analysis and User Requirements Gathering, *Computers in Industry* **54**(3), 307-323 (2004).
11. H.E. Eriksson and M. Penker, *Business Modeling with UML* (USA, John Wiley & Sons, 2000).
12. K. Kosanke, F. Vernadat, and M. Zelm, CIMOSA: Enterprise Engineering and Integration, *Computers in Industry*, **40**(2), 83-97 (1999).

13. G. Booch, J. Rumbaugh, I. Jacobson, *The Unified Modeling Language User Guide* (Addison-Wesley, 1999).