

Enterprise Resource Planning Systems for Small and Medium-sized Enterprises

Prof Rogerio Atem de Carvalho, D. Sc.

Fluminense Federal Institute (IFF), Brazil

Dr Björn Johansson, Ph. D.

Center for Applied ICT, Copenhagen Business School, Denmark

ABSTRACT

After the implementation peak of ERPs that occurred during the pre- and post-Y2K periods, the high-end ERP market started to saturate and major ERP vendors started to seek for new business opportunities, in special towards Small and Medium-sized Enterprises (SMEs). On the buyer side, demands for becoming more competitive in a globalized market, have been pushing SMEs to adopt ERP too. Additionally, influenced by the free/open source movement, new types of ERP licensing appeared by the beginning of the decade, creating a classification according to the basic licensing model: free/open source ERP (FOS-ERP) and proprietary ERP (P-ERP). Therefore, this paper aims at exploring the merge between SMEs, P-ERP, and FOS-ERP, by analyzing the differences between the two proposals and offering guidance for prospective adopters.

INTRODUCTION

For the last fifteen years the biggest software category in terms of investment has been Enterprise Resource Planning (ERP) systems and nowadays most of the bigger companies, including practically all global-players, have implemented ERP systems in the search for achieving competitive edge in their business areas (Church, 2008; Hendricks, Singhal and Stratman, 2007). After the implementation peak that occurred during the pre- and post-Y2K periods, the high-end ERP market started to saturate and, according to Kim and Boldyreff (2005), major ERP vendors started to seek for new business opportunities, in special towards small and medium-sized enterprises (SMEs).

On the buyer side, demands for becoming more competitive in a globalized market, have been pushing SMEs to adopt ERP too. However, the ERP strategic nature (Caulliriaux, Proença and Prado, 2000), and the fact that it cannot be used instantaneously (Dreiling, Klaus, Rosemann and Wyssusek, 2005), makes ERP a different kind of software which implementation requires high quantities of resources and entails high risks. These factors have been raising the interest on ERP versions for SMEs on both the demand and supply side of the market, making a recent market movement towards simplified versions of proprietary ERP (P-ERP) and bringing more attention to free/open source ERP (FOS-ERP). Therefore, it is clear that the merge of P-ERP, FOS-ERPs and SMEs is a topic with many facets and yet to be more explored in both theory and practice.

This chapter aims to explore this merge, firstly by briefly introducing Generalized Enterprise Reference Architecture and Methodology conceptual framework, which in the following section is used to guide the description of the adoption process by SMEs, followed by an detailed analysis of the differences of each proposal, and finally by a basic guidance for prospective adopters.

A FRAMEWORK TO GUIDE ERP ADOPTION

Selecting an ERP for adoption is a complex process, because, besides the size of the task, it is an important enterprise component that impacts the adopter organization in financial and self-knowledge terms. In that aspect, the Generalized Enterprise Reference Architecture and Methodology (GERAM) is a well-known standard that can be used to identify the main phases of an ERP adoption project, helping guiding the comparison of free and proprietary alternatives.

The GERAM framework provides a description of all elements recommended in enterprise engineering and a collection of tools and methods to perform enterprise design and change with success (IFIP – IFAC, 1999), providing a template lifecycle to analyze ERP selection, deployment, and evolution. GERAM defines seven lifecycle phases for any enterprise entity that are pertinent during its life. These phases, presented in Figure 1, can be summarized as follows:

Identification: identifies the particular enterprise entity in terms of its domain and environment.

- a) **Concept:** conceptualizes an entity's mission, vision, values, strategies, and objectives. During this phase, high-level objectives are established, such as the acquisition strategy, preliminary time and cost baselines, and the expected impact of ERP adoption.
- b) **Requirements:** comprise a set of activities needed to develop descriptions of operational requirements of the enterprise entity, its relevant processes and the collection of all their functional, behavioral, informational, and capability needs. The requirements phase deals with ERP's functional and non-functional requirements. The adopter may model some main business processes – part of the Preliminary Design – as a way to check how the alternatives fit to them.
- c) **Design:** activities that support the specification of the entity with all of its components that satisfy the entity requirements. The scope of design activities includes the design of all human tasks, and all machine tasks concerned with the entity's customer services and products and the related management and control functions. The design of the operational processes includes the identification of the necessary information and resources. This phase is divided into Preliminary Design, which deals with overall enterprise specifications - sufficient to obtain approximate costs and management approval of the ongoing project, and Detailed Design, which is the work necessary for guiding the creation of the final physical system. The detailed design phase focus on refining models, and is associated to business process modeling and parameter identification and value definition.
- d) **Implementation:** transforms the design into real components. Comprises implementation in the broadest sense, covering a) commissioning, purchasing, (re)configuring or developing all resources needed; b) hiring and training personnel, and developing or changing the human organization; c) component testing and validation, system integration, validation and testing, and releasing into operation.
- e) **Operation:** is the actual use of the system, and includes user feedback that can drive to a new entity lifecycle. The resources of the entity are managed and controlled so as to carry out the processes necessary for the entity to fulfill its mission. During the operation phase the resources of the entity are managed and controlled so as to carry out the processes necessary for the entity to fulfill its mission. Deviations from goals and objectives or feedbacks from the environment may lead to requests for change; therefore during this phase system maintenance and evolution occur.
- f) **Decommission:** represents the disposal of parts of the whole entity, after its successful use.

Taking as a principle that most software development (and customization) today is done through interactive and incremental lifecycles, it can be considered that there is no clear borderline between the Requirements and Preliminary Design phases and between the detailed design and implementation phases, thus they are considered together in this approach.

Except for *identification* and *decommission*, which are not influenced by licensing models, the GERAM phases can be used to better understand the differences between FOS-ERP and P-ERP, providing key aspects for evaluating alternatives and successively refining objectives, requirements and models, as the next section addresses.

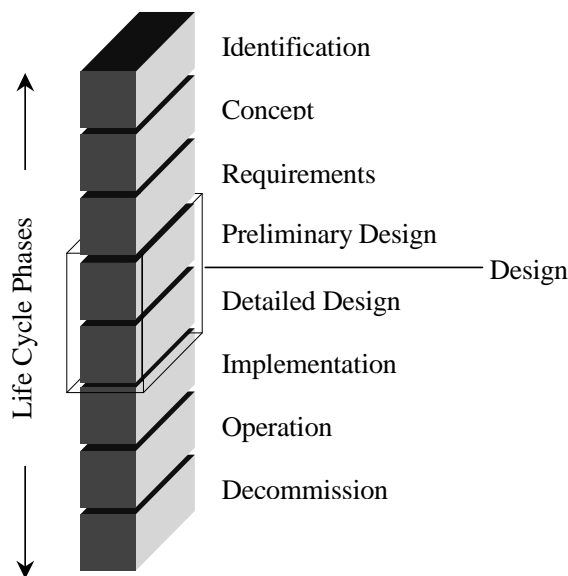


Figure 1. GERAM Lifecycle Phases.

ANALYZING FOS-ERP AND P-ERP ADOPTION

Concept

In the case of FOS-ERP, the level of involvement of the adopter in development can be established from the beginning of the adoption process. In other words, at this point the adopter can already start considering the possibility of actively contributing to an open source project, or at least making by itself the necessary customization. Of course, the final decision will only be possible during the more advanced phases, when the adopter better knows the solution requisites and the decision alternatives.

The level of involvement in development of P-ERP is something that an adopter could also consider. However, involvement comes in another shape since P-ERPs often are delivered under a partnership model. This means that adopters are not directly involved in the development, but they are involved indirectly by cooperation with the specific partner hired to deploy the system. This also means that adopters in a way choose the level of involvement by selecting between different P-ERPs, since the way adopters engage in the development differ. Examples of different engagement could be understood if comparing the “old” partnership model (such as SAP/R3) and the “new” way of delivering P-ERPs as software as a service (SaaS, such as –Workday). In the SaaS model the adopter engagement in development of the P-ERP is lower than in the partnership model, since the partnership

model to a higher extent builds on engagement through cooperation between adopters and partners when adjusting the specific ERP.

Requirements and Preliminary Design

During this phase evaluation takes place. At this point FOS-ERP starts to differ more from P-ERP. Evaluating P-ERP involves comparing alternatives under the light of functionality, Total Cost of Ownership (TCO), and technological criteria. For FOS-ERP these criteria and others related specifically to FOSS, such as the maturity level of the project, its survivability, and its partner network strength (Carvalho, 2006), must be also taken into account – remembering that even if the implementation represents a smaller financial impact, in terms of a company's self-knowledge it can assume a much bigger importance, since it holds not only a inventory of records and procedures, but also how those records and procedures are realized in technological form – through source code.

At this point the core matter of the so-called “best practices” becomes more visible. A best practice can be defined as the most efficient and effective way of accomplishing a given business process, based on repeatable procedures that have proven themselves over time for large numbers of organizations. However, since organizations differ from each other, the adopters first need to find the solution that fits their business, and after doing that they have to either reconfigure existing business processes or reconfigure the software so that the software and the business processes fits each other. In the case of P-ERP, best practices are built on top of many years of experience of the global players, making these practices well tested and experimented, allowing a safer implementation on the adopter side. On the other hand, if FOS-ERP, being younger software projects, offer less experimented practices, an adopter can use the ones based on standardized procedures - such as accounting - *as is* and for free; and then expend its resources on adapting solutions related to more strategic procedures. This adaptation can be done by the adopter – demanding more knowledge on the solution, or by someone from the FOS-ERP partner network, like in P-ERP.

From this standpoint, the strategic positioning of an adopter in relation to a FOS-ERP seems to be of greatest importance, given the possibility of deriving competitive advantage from the source code. Therefore, the adopter must decide to behave as a simple *consumer*, only getting the solution from the vendor, or become a *prosumer*, by mixing passively purchasing commodity parts of the system with actively developing strategic ones by itself. Thus it is clear that when an adopter considers FOS-ERP as an alternative, it should also consider developing parts of it to fit its requirements – taking into account that, as said before, this kind of positioning involves allocating managerial and technical resources for development tasks in a FOSS environment.

In other words, a FOS-ERP can have a smaller financial impact but a much bigger knowledge and innovation impact. Although P-ERP are also highly parameterized, and adaptable through APIs and/or dedicated programming languages, the access to the source code in FOS-ERP can drive much better exploration of the ERP's capabilities, thus allowing a better implementation of differentiated solutions. But, as stated above, this demands a high level of knowledge about the ERP's platform, and if the adopters do not have this they have to trust on other sources for having that knowledge, and since in some countries there is a reduced number of FOS-ERP experts, costs for hiring them can be higher than it is for P-ERP consultants in these cases.

Detailed Design and Implementation

This phase is the one where the two kinds of ERP differ more. In the case of FOS-ERP, if the adopter decided to participate actively in the project, deeper design decisions are involved, such as creating

entire new modules or extending the basic system's framework. A consequence of assuming a more active role is to invest more human and financial resources for learning the FOS-ERP platform and framework, developing and maintaining parts of it, and managing the relationship with the project community. In that case, customization and maintenance contracts must define responsibilities of each part on the deployment process. For instance, what should the vendor do if the adopter finds a bug in the original code, written by the first, which is being adapted by the second? What is the priority that the vendor must follow for correcting this bug? Actually, is the vendor responsible for correcting this bug, since for this part the adopter decided to take advantage of the solution's free license, therefore exempting the vendor of responsibility for the bug?

Still on FOS-ERP, the adopter has the option of assuming different grades of involvement for each part of the system. For ordinary modules, such as payroll, the adopter can let the vendor do the work. However, for strategic modules, where the adopter believes that it holds competitive advantage in the related business processes, it can take an active role, to be sure that the business knowledge, or at least the more precious details that keep the competitive advantage, will be kept in the adopter environment. In that situation the vendor is limited to act as a kind of advisor to the adopter. One can think that it is possible to keep secrecy on parts of the system by properly contracting a P-ERP vendor, which is true, but the adopter will become dependent of the vendor in a strategic part of the system. Becoming dependent means to wait for other vendor's priorities or pay a high price to become the priority when changes are needed.

In P-ERP new features development takes a different way, usually in the form of add-ons. A solution partner often develops these add-ons for a specific vendor's product, which can be customer specific, but most often can be used by several adopters. If the adopter decides to develop add-ons, it has to consider that it could be problematic in the future when new versions of the core product are released. In some cases software vendors do not take responsibility for that the new version is interoperable with the customers developed and implemented add-ons.

Operation

When it comes to operation, giving the licensing differences, it can be suggested that there are also major differences between FOS-ERP and P-ERP. In the first case, during operation the adopter can decide at any moment, unless specific contractual clauses hinders, to shift to another vendor or to assume the system's maintenance by itself. Minor changes can also be conducted by the own adopter or even by community individuals that may help on specific matters.

In the P-ERP case the adopter are to some extent dependent on the software vendor, but, it can be claimed that they also have a high grade of support from the vendor. This support depends on if the adopters chose to sign a service contract or not. The service agreement is nothing that are forced on the adopter but if the adopter chose that they have access to a service organization that helps them out if they have problems with the software. In the case of FOS-ERP, prospective adopters must be aware of the type of FOSS project the ERP is, since in sponsoring terms there are two kinds of free/open source projects, which are the community and the commercially sponsored projects (West and O'Mahony, 2005). In the first case, the adopter will become more or less dependent on voluntarism or in others words, uncertain schedules and priorities. Only the commercially sponsored FOS-ERP normally offers a partner network in a similar mode of P-ERP.

FOS-ERP AND P-ERP DIFFERENCES

ERP for SMEs is a relative new market; however a series of conclusions, from different categories, can be drawn from the comparison of the proprietary and the free/open source solutions.

On the economic side, the most obvious, and a motivator for this work, is that lower costs in both FOS-ERP and P-ERP open new opportunities for SME to become ERP adopters. Moreover, some P-ERP and FOS-ERP are offering basic modules for free, such as Microsoft's small ERP system Office Accounting (in US and UK), and the Accounting module of ERP5 on its SaaS version. However, lower costs can also mean that adopters have to deal with lower service levels, then stressing the necessity of carefully evaluating ERP options regarding service levels and pricing. In the case of FOS-ERP, consulting certification is yet on the early stages, thus quality of service must be carefully addressed during contract negotiation. On the other hand, although SMEs can change their business processes faster than bigger companies, cheaper but restrictive P-ERP support contracts can make the software change slower than the real business.

Also on the economic side, following the common reasoning about FOSS pricing, FOS-ERP vendors can take advantage from open source software because, according to Riehle (2007), open source systems "increase profits through direct costs savings and the ability to reach more customers through improved pricing flexibility", allowing partners and free-lance vendors switch from more expensive proprietary software to less expensive open source software, since, as a general rule, FOS-ERP relies on other open technologies. For instance, while most P-ERP systems export and import data to and from MS-Office, FOS-ERP in general interacts with the also free Open Office. The same can be affirmed for databases and operational systems – thus reducing costs on ERP supportive software too.

Another economic factor is that P-ERP vendors generally impose high costs and a rigid set of rules for companies that desire to enter their partner network, raising the difficulties for smaller firms to become players in this market. In contrast, smaller consulting firms can enter the FOS-ERP market in an incremental way, increasing their commitment to a project as new business opportunities appear and bring more financial income. In other words, firms can start contributing with small improvements to the project as a way of gaining knowledge on the system platform and framework, and, as customers to the solution appears, more money can be invested on a growing commitment to the project. With more partners entering the market, consulting costs can be reduced, help shifting the market perspective from vendor to adopter. On the other hand, the P-ERP partner networks rely on more mature consulting processes and experiences of global ERP players. In that sense, FOS-ERP must learn on how to create certification and consulting processes that are at the same time high quality products, like the P-ERP ones, and cheaper than those, to survive on the market.

Regarding the access to the source code, the experience has shown that most of the times the adopter will not get involved on customization or even maintenance tasks. Still, FOS-ERP can be a good choice, since it reduces vendor dependency. Moreover, its code openness also makes adapting it to specific needs easier, thus reducing costs in customization and further evolution of the software.

Yet on the development and customization arena, another interesting remark is the fact that FOS-ERP and P-ERP development becomes more and more similar. The similarities are that P-ERP developers work more and more with open source in their products and they also to a higher degree starts to make the source code available. It can also be claimed that more and more open source development is carried by developers that are paid by software vendors – the so-called Sponsored Open Source (West et al., 2005).

However, when thinking on customizing the ERP by itself, a critical question for an organization adopting ERPs is if they have the resources needed for implementing ERPs. In both FOS-ERP and P-ERP cases it is very unlikely that a small company will have the resources and the knowledge for making the necessary modifications by itself. This situation is allowing a shift to the SaaS business

model, which indicates that in the future SaaS as delivery model could solve this problem for adoption of ERPs regardless their type (open or proprietary). As examples on the evolution of SaaS delivered ERPs there are ERP5 Express¹ on the FOS-ERP side and Fortnox on the P-ERP side. For the Medium Enterprise, which has more IT resources, it could be easier to adopt a FOS-ERP and take part on its customization, as occurred with some adopters.

The question is then what the new delivery models of P-ERP means when it comes to differences between support in the FOS-ERP case and the P-ERP case. One possible outcome of this is that it will only be the solutions that have a high level of relevant services connected to the specific ERP solution delivered that will survive. It also means that FOS-ERPs and P-ERPs will become closer to each other and this movement maybe, to some extent, makes it harder for the adopting organization to decide on which kind of ERP to select.

Following this reasoning, it can be stated that FOS-ERPs, when evaluated as products and not as services in a SaaS mode, can be harder to evaluate than P-ERPs. When evaluating FOS-ERPs the adopting organization is supposed to also carefully evaluate the maturity of the FOSS project, its survivability, and its partner network – having special attention on the risks associated with adopting a community founded ERP project.

Independent if it is FOS-ERPs or P-ERPs that are discussed, another interesting observation is the fact that empirical evidence (Hendricks et al., 2007) shows that it is not the software as such that provides an organization with competitive advantage, instead it can be claimed that it is the usage of the software, in the form of internal capabilities (Mata, Fuerst and Barney, 1995), that influences whether an organization is competitive or not. This then implies that even if two organizations working in the same industry adopts the same ERP they could have different performance. The question is then if this depends on that the inherited business process in the software are so generic that its supports only the very basic needs. It could be the fact that to be able to gain competitive advantage from adoption of ERPs, the system needs to be highly adjustable, but, also that the adopters could adjust the software by itself in a convenient way.

FINAL CONSIDERATIONS: GUIDANCE FOR ADOPTERS AND FUTURE RESEARCH

From the previous conclusions, some guidance for SME adopters can be provided, and it can be concluded that specific size of the company makes a difference. For small enterprises, the SaaS model seems to be the solution, and the adopter must evaluate, besides the necessary functionalities, service levels and pricing, regardless the licensing model.

In the case of medium enterprises, it can be suggested that if the company have enough development and supporting resources and already uses other open source applications in its organization it could consider an FOS-ERP solution, analyzing carefully the maturity of the project. A medium enterprise could also consider FOS-ERP if it needs to develop a solution that is very different from others, given innovative business processes. On the other hand, if the company does not have the resources or do not want to invest on customization by itself, and if it already uses – and is satisfied with - a partner helping with other software applications, it should definitely consider to stick to P-ERP, considering, again, whether a SaaS model should be adopted.

¹ In the specific case of ERP5 Express, an innovative business model allows the adopter to, at any moment, download all code and data and adopt it as a local version of the system, regardless the level of customization that was applied to the adopter's instance.

Future research subjects are to obtain experimental data on both kinds of ERP adoption by SMEs, investigate which changes FOS-ERP caused on the ERP market, and also investigate the differences on adoption of both kinds of ERP by SMEs in different regions of the world.

REFERENCES

- Carvalho, R. A. (2006) *Issues on evaluating Free/open source ERP systems*, In Research and Practical Issues of Enterprise Information Systems Springer Verlag New York Inc, New York, pp. 667-676.
- Caulliraux, H. M., Proença, A. and Prado, C. A. S. (2000) *ERP Systems from a Strategic Perspective*, Sixth International Conference on Industrial Engineering and Operations Management Niteroi, Brazil.
- Church, Z. (2008) *SAP ERP on-demand challenge lofty, but not impossible*.
- Dreiling, A., Klaus, H., Rosemann, M. and Wyssusek, B. (2005) *Open Source Enterprise Systems: Towards a Viable Alternative*, 38th Annual Hawaii International Conference on System Sciences Hawaii.
- Hendricks, K. B., Singhal, V. R. and Stratman, J. K. (2007) *The impact of enterprise systems on corporate performance: A study of ERP, SCM, and CRM system implementations*, Journal of Operations Management 25 (1), pp. 65-82.
- IFIP – IFAC (1999) *GERAM: Generalized Enterprise Reference Architecture and Methodology*, Task Force on Architectures for Enterprise Integration, 31p.
- Kim, H. and Boldyreff, C. (2005) *Open Source ERP for SMEs*, Third International Conference on Manufacturing Research Cranfield, U.K.
- Mata, F. J., Fuerst, W. L. and Barney, J. B. (1995) *Information technology and sustained competitive advantage: A resource-based analysis*, MIS Quarterly, 19 (4), pp. 487-505.
- Riehle, D. (2007) *The Economic Motivation of Open Source: Stakeholders Perspectives*, Computer IEE Computer Society, 25-32.
- West, J. and O'Mahony, S. (2005) *Contrasting Community Building in Sponsored and Community Founded Open Source Projects*, 38th Annual Hawaii International Conference on System Sciences, Hawaii.

AUTHORS' PROFILES

Rogério Atem de Carvalho holds a Doctoral and a Master degree in Production Engineering and a Bachelor degree in Computer Science. He was awarded with the IFIP *Distinguished Academic Leadership Award*, for his studies on the field of FOS-ERP, in Vienna, Austria in 2006. He is a consultant for the ERP5 FOS-ERP (<http://www.erp5.com>), and a project leader for Enterprise Content Management solutions for governmental and private sectors since 2000. He is as a member and Chair of the Brazilian Chapter of the IFIP Working Group on Enterprise Information Systems and Founder Member of the IEEE SMC Society Technical Committee on Enterprise Information Systems. He is also an Associated Editor for Enterprise Information Systems journal, published by Taylor & Francis.

Björn Johansson holds a Doctoral and a Licentiate degree in Information Systems Development and a Bachelor degree in Business Informatics. Currently he works at the Center for Applied ICT at Copenhagen Business School, within the 3gERP project (<http://www.3gERP.org>). He is a member of the IFIP Working Group on Diffusion, Adoption and Implementation of Information and Communication Technologies and the research networks: VITS Work practice development, IT usage, Coordination and Cooperation and KiO Knowledge in Organizations.

KEY TERMS

ERP

Enterprise Resources Planning, a kind of software which main goal is to integrate all data and processes of an organization into a unified system.

Free Software

According to the Free Software Foundation, is a Software that gives to the user the freedom to run the program for any purpose, study how the program works and adapt it to his/her needs, redistribute copies, improve the program, and release his/her improvements to the public, so that the whole community benefits.

Open Source Software

According to the Open Source Initiative, licenses must meet ten conditions in order to be considered open source licenses:

1. The software can be freely given away or sold.
2. The source code must either be included or freely obtainable.
3. Redistribution of modifications must be allowed.
4. Licenses may require that modifications be redistributed only as patches.
5. No discrimination against persons or groups.
6. No discrimination against fields of endeavor.
7. The rights attached to the program must apply to all to whom the program is redistributed without the need for execution of an additional license by those parties.
8. The program cannot be licensed only as part of a larger distribution.
9. The license cannot insist that any other software it is distributed with must also be open source.
10. License must be technology-neutral

The official definition of Open Source Software is very close to the definition of Free Software; however, it allows in practice more restrictive licenses, creating a category of “semi-free” software.

Free/Open Source ERP

ERP systems that are released as Free Software or Open Source Software.

Proprietary Software

According to Wikipedia proprietary software is a term for computer software with restrictions on use or private modification, or with restrictions judged to be excessive on copying or publishing of modified or unmodified versions. The term is coined by the free software community and means that these restrictions are placed on it by one of its proprietors. Similarly, closed source is a term for software whose license does not meet the Open Source Definition.

Small and Medium-sized Enterprises (SMEs)

In the chapter the definition of SMEs follows the definition presented by the European Commission in 1996 which state that: a small enterprise is an enterprise with fewer than 50 employees and a medium-sized enterprise is an enterprise with more than 49 and fewer than 250 employees.

Software as a Service (SaaS)

According to the Software and Information Industry Association, in this model, the application, or service, is deployed from a centralized data center across a network providing access and use on a recurring fee basis. In other words, users "rent," "subscribe to," "are assigned", or "are granted access to" the applications from a central provider.