

Free/Open Source ERP and Translation Processes: Four Empirical Cases

Elena Erbizzi, Maurizio Teli, Gianmarco Campagnolo, Stefano De Paoli¹, and Vincenzo D'Andrea²

1 Department of Sociology and Social Research
Via Verdi 26, 38100 Trento, Italy

WWW home page:

http://www4.soc.unitn.it:8080/dsrs/content/index_ita.html

2 Department of Informatics and Telecommunications
Via Sommarive 14, I-38050 Povo (TN)

WWW home page: <http://dit.unitn.it>

Abstract. Can ERP be studied from the point of view of Science and Technology Studies? What kind of insights this approach can bring to the comprehension of the phenomenon? These are the questions we are trying to answer with this paper. In the Information Systems field, ERP have been studied according to three main approaches: ethnomethodology, phenomenology, and deconstructivism. We show how these accounts are not able to describe ERP as complex artifacts.

We suggest that an actor – network theory approach, without making a priori assumptions about the role that technology, society, politics and economics can enact, is a powerful theoretical instrument in the study of ERP. Drawing from a comparative analysis of four systems (Compiere, Erp5, GNUe and Open for Business), we show how the choice of a free/open source license was made according to different goals, and entered in the translation processes taking place in the ERP development; shaping communities, customers and developers.

In conclusion, we believe that our approach can be fruitfully applied to the study of other topics connected to ERP, such as accountability, thanks to its ability to show how the different aspects are relationally shaped.

1 Introduction

ERP systems are complex artifacts with technological, social, political, and economical aspects. Our aim with this paper is to suggest how a research based on

Please use the following format when citing this chapter:

Erbizzi, E., Teli, M., Campagnolo, G., De Paoli, S., D'Andrea, V., 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.695-704.

the Science and Technology Study (STS, [1]) approach can bring to new insights about Free/Open Source ERP. Our work is based on a comparative case study.

In the first part of this paper we will discuss the theoretical approaches generally adopted in the analysis of ERP. We argue that the ethnomethodological approach, the phenomenological one and the deconstructivist one, are not able to give descriptions of the phenomenon useful in order to understand the relationships between the technical, social, political and economical aspects. We suggest that the sociology of translation [2], lately known as actor – network theory [3, 4], is a powerful theoretical instrument, capable of tracing the relationships taking place when studying ERP. This theoretical approach is based on a symmetrical point of view, not assuming *a priori* distinction in the construction of technology between the human and non – human elements.

In the second part, we will approach the study of four cases: Compiere*, Erp5+, GnuEnterprise (GNUe)[†] and Open for Business (OfBiz)[‡]. In this analysis, we will focus on the websites of these projects, showing how the common element, the choice of a software license, is relationally connected with every aspect of an ERP. The analysis will be done in accordance to the translation process described by Callon [2] We will stress the relational interdefinition taking place when the aim of the developers of the ERPs studied brings to the adoption of Free/Open Source licenses. In addition we also discuss the capability of these artifacts to act in translating communities, potential customers, business partners and source code, in other words their *agency*. To this end, before the case descriptions, we discuss the views of the world inscribed in the licenses adopted.

In the last part of the paper, we will compare the results of the case analyses, drawing a first picture of the potential of STS descriptions in the study of Free/Open Source ERP.

2 The Limits of Previous Approaches

Among Information System scholars, we can identify three perspectives on studying ERP: ethnomethodology, phenomenology, and deconstructivism. Even though we are limited by space constraints in the discussion of these rich research traditions, we will try to briefly highlight what are, in our view, their limitations in capturing the complexity of the phenomena connected to Free/Open Source ERP.

The ethnomethodological approach has its foundation in the work of Garfinkel [5] and was brought to the field of Computer Science and Information Systems by Suchman [6]. The central node in this approach is the link between representations of action contained in a culture - and interpreted by the traditional sociology as a guide

* <http://www.compiere.org>

† <http://www.erp5.org>

‡ <http://www.gnuenterprise.org>

- <http://www.ofbiz.org>

for human action - and the methods by which members construct locally the reality. The cultural tendency to isolate human membership from non-humans membership (e.g. the tools in use in a situation), brings to the consequence that ethnomethodology cannot recognize the ordering capabilities of tools.

To understand how the phenomenological approach has been brought to the Information System field, we have to refer to Schutz writings [7]. In particular, when he concentrates on the relationship between daily life and the world of scientific contemplation. To clarify this relation he coins the term “finite provinces of meaning”. Each province of meaning is characterized by a specific cognitive style. All the experiences inside a province of meaning are per se coherent with the cognitive style and compatible to each other. But what it is more important is the concept of ‘finitude’ of any sub-world that clarifies its relation with the others. Each province is finite because there isn’t the possibility to refer one province to another via transformation formulae but only with a leap, due to a trauma. Within this approach it is also hard to grasp the interaction between social, technical, economical, and political aspects of ERP.

For deconstructivist perspective we mean a research approach addressed to unveil concepts, assumptions, and implications inscribed in artifacts and representations by their designers or engineers. Such perspective was initiated by the philosopher Langdon Winner [8] that in his essay entitled “Do artifacts have politics?” underlines how technologies aren’t neutral, but might irreversibly embody forms of oppression. Winner's argument is that tangible artifacts (including architectural space and other material and spatial devices) embody social relations (that is forms of power). Deconstructionism has been drawn into the Information Systems field by Star [9]. In this perspective the situated mediations of inscriptions are underestimated, it seems that technology is bad in itself. If we maintain that a technology is value-laden in itself, it is hard to study how the social factors interact with it, not to speak of the economical and political issues.

On the other end, a description in terms of translation processes seems able to overcome the limits of the above discussed approaches, not taking a priori assumptions about the mutual shaping of human and non – human elements [10]. The process of translation is both the movement of an entity in space and time, as well as its displacement from one context to another, with the necessary transformations of meanings that this always implies. Translations don’t take place in one point in time, but are recursively generated and reproduced thanks to maneuvers and strategies of human and non – human actors. Michel Callon [2] identifies a recurrent pattern, consisting of typical phases, problematization, interessement, enrolment and mobilization. This pattern shapes our methodology for empirical research. Our theoretical discussion is valuable for both proprietary ERPs and Free/Open Source ones. A distinctive characteristic of the Free/Open Source software is the different conception of copyright, inscribed in the legal artifact connected to that: the license. For that reason, we will now present a brief analysis of the GNU GPL license, to which the licenses associated with our cases will be connected.

3 A (short) Actor – Network Theory of the GNU GPL 2.0[#]

In an actor-network perspective licenses should be considered as actor/actant inside the process of software's construction. Furthermore, according to Lanzara and Morner [11] licenses results from the inscription and translation in legal form of some programmers habitus [12], such as the practice of share information and software programs. We need then to briefly outline what is inscribed in the ERP licenses and what is translated by them.

Along this line of reasoning in different licenses we can find inscribed some different “weltanschauung”/view of the world, coming from different practices of sharing computer programs. Both in Free Software and in Open Source licenses seems to be inscribed the programmers' habitus of sharing programs, but differences among licenses describe and reflect different ways in which this habitus give shape to what software and users are. In this paragraph we will shortly describe the GNU GPL 2.0 license.

In our view when applied to software, this license would act as a separator between what are considered “good” and “dangerous” actors from the Free Software Foundation (FSF) point of view. The GPL preamble considers dangerous actors: (a) most part of software licenses; (b) software patents and (c) people which deny the Free Software rights. For the FSF, all the dangerous entities/actors would define the software as proprietary (not Free) and the users as not free to change and share the software. Also most of the Open Source Licenses should be considered as “dangerous” in this sense: they allow for example to release modified software under proprietary licenses. The GPL will cut then the link between software and users (good actors) from one side and the various dangerous entities from the other.

In actor – network theory terms this act of separation is the *interessement* and the artifacts used for achieve the separation are called *devices of interessement*. Thorough the *interessement* a stronger inter-definition of the good entities (users and software) is achieved. The well known copyleft term (2b) well explain the *interessement*: “You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License” [13]. By means of term (2b), the GPL ensures that licenses which clashes with the Free Software rights (the so called “Most part of Software licenses”) are detached from Free Software. The following scheme summarizes how the GPL acts in general as a *device of interessement*.

[#] For a complete analysis, see De Paoli [14].

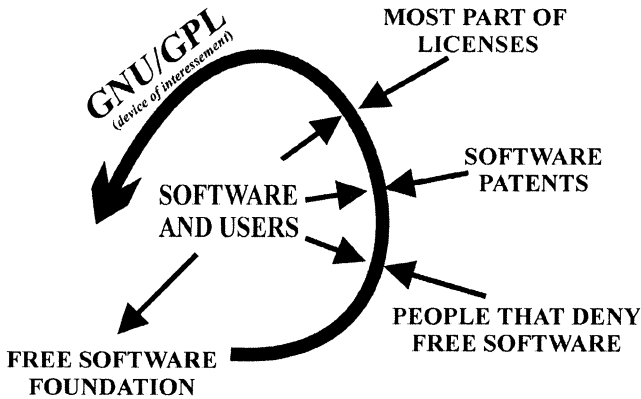


Fig. 1. The GNU/GPL applied to a generic software act as a device of intersement [14].

4 Methodology

As anticipated, our study draws from a preliminary analysis of four Free/Open Source ERPs: Compiere, Erp5, GnuEnterprise and OfBiz. They are the result of a reasoned sampling, with the aim to describe successful projects, with different goals and organizations behind them.

In order to be faithful to the ethnographic principle to “listen carefully to the natives” [15] we based our descriptions on what is publicly expressed in the projects websites, trying to give a “thick description” [16] of the translation process taking place, from the point of view of the organizations behind the projects, in a “natural documents” analysis [17]. This methodological choice is justified by the fact that our study is a preliminary one.

5 Four Empirical Cases

5.1 Compiere: choosing Open Source to gain the market leadership

The first case we present is that of Compiere, a mature project, started in 2000, with a declared goal: “we don't want to be just another ERP/CRM system for some niche industry. We want to become a primary horizontal ERP and CRM solution for small – medium enterprises”. In a translation description, the declared aim of the protagonist is relevant in order to give an interpretation of what happens. In this case the protagonist is Compiere, Inc., a profit – oriented company founded by Jorg Janke, the beginner of the project (with the help of Goodyear Germany). The choice

of an open source license, the Compiere Public License, based on the Mozilla Public License 1.1*, is motivated by the necessity to create a space in a saturated market, with the combination of two factors: reduced total cost of ownership and the development guidance by customers and partners.

If we try to discuss the Compiere case with the concept of translation, we can say some words about the different phases. In the problematization, defined as the phase of the interdefinition of the actors and the definition of an obligatory passage point (OPP), we can see how the involvement of Goodyear Germany act in the definition of Jorg Janke as a skilled programmer, how the use of the term open source recall the open source movement (and not the free software one, [18]), how the customers are searching money savings and quality product and how the partners are considered technical skilled entrepreneurs.

During the interdefinition of actors, Compiere Inc. tries also to define the interests of the potential customers, partners and developers, showing how it will be an obligatory passage point in the gain of the expected result, mainly through the use of the license as a device of intersement. In this case it is interesting to note how the incompatibility between the MPL and the GPL is overcome with a provision (section 13) that allows a program (or parts of it) to offer a choice of another license as well. On the Compiere licenses' web pages this point is specified in relation to the GPL: "if you use any product with a GPL license with Compiere [...] the Compiere (Mozilla) license terms do not apply [...] you need to comply with GPL terms". In this way it becomes possible to use Compiere with GPL software: the CPL does not apply at all and the users have to comply with the GPL. So, the license chosen act in the reshaping of the community directly involved in the project: the free software movement is not excluded, as well as proprietary developers (because of some restrictions on the appliance of the copyleft in the MPL). The open source choice is connected to: economical characteristics, as well as the absence of licenses costs and the fall down of the total cost of ownership; technical characteristics, the code is considered of high quality and with low cost of maintenance; political characteristics, as the opportunity to see the code and to trust Compiere ("no lies no surprises") and social characteristics, like the existence of a community surrounding the project.

The presence of a device of intersement doesn't warrant for the stabilization of Compiere like a recognized leader in the ERP industry. The enrolment, the definition and coordination of roles, is the successful completion of the intersement: the customers accept the role of Compiere, as well as the partner and the open source community. When the enrolment has taken place, the seduction of potential new actors in the actor – network of Compiere can take place (and the website is one of the instrument).

The success of a translation process is achieved when the mobilization takes place. During the mobilization, entities which were static become mobile, dispersed elements become enrolled ones. In this case, the website is an example: the listed partners, successful customers, the open source community, the websites and

* <http://www.mozilla.org/MPL>

magazines talking about Compiere, are mobilized by the company, speaking for them in front of the potential new actors in the network. If they will refuse their consensus to Compiere in the future, the entire project will be in peril.

As shown, the choice of an open source license, justified with the aim of entering successfully a saturated market, shapes the actors involved, their interests, and their role, in a relational way involving the open source high quality image, cost – saving aspects and political transparency.

5.2 Erp5: choosing Libre Software to create a common knowledge

In 2002 one of the French apparel leaders, Coramy, decided to abandon its in-house developed ERP solution to embrace the Free Software development model, with the aim “to reduce software maintenance costs and to allow Coramy to retain complete control over its custom developments, something that would have been impossible with standard proprietary ERP solutions.” In order to do this, it chose to start a partnership with Nexedi, a Zope based solutions provider: the result was the birth of the Erp5 project. The license chosen is the GNU GPL, and the protagonist we are following is mainly Nexedi. The declared goal of the project is “developing Free ERP Software and creating a common knowledge base on ERP technologies and methodologies.”

Reading the Erp5 case with an actor – network perspective leads us to reconsideration both of the project and of the Libre Software development characteristics. During the problematization, the actors defined are Coramy, the products, the others ERP solutions, potential customers and partners. The definition of Nexedi like a technical qualified actor, connected to the strategic reasons of companies interested in becoming customers, is similar to that of Compiere. The very interesting thing in this phase is that others Free/Open Source ERP solutions are considered, for example Compiere, and the contribution to a reshaping of the characteristics of a Free/Open Source ERP, not only as a technical object, but also as a documentation needed, in order to allow a clear distinction between Free/Open Source software and proprietary one: “Without any documentation on the Compiere model, there was no way for us to evaluate this risk. [...] Free ERP software requires extensive and free documentation on all the aspects of ERP: technical, juridical and theoretical.” In this case, the device of *interressement* is not only the license, but also the project website, with a great amount of documentation and underlining the role of Nexedi, the obligatory passage point for whom want to have not only an installation of Erp5, but also simplified access to knowledge related to the product.

In this process of translation, the role of Nexedi is that of the mediator between different needs and roles, that of Coramy, interested in being recognized as the main customer of the project, that of the others potential customers, interested both in a quality product and in access to a knowledge base, that of the others Free/Open Source ERP solutions, missing documentation. The role of Nexedi as spokesperson for that actor – network can be put in discussion by these actors, but now it seems successful.

5.3 GNU Enterprise: bringing freedom to enterprises

In 1999, the maintainers of the Obelisk project and GNU G/L, a Free Software Foundation project, merged the projects, giving birth to GNU Enterprise (GNUe). In 2000 the merge with the Sanity project gave birth to the current incarnation of GNUe, GPLed software, based on volunteer contributions, with the declared goal “to develop enterprise-class data-aware applications as Free Software”. It is a set of tools, at a stable level, and packages, at a planning level.

In the interdefinition of actors, the very interesting thing is the absence of the terms customers and clients. The actors involved are: the GNU project, the code, enterprises and the community. The first one is recognized as the entire work of which GNUe is part, the second is characterized by its being Free and of high quality, enterprises want to be vendor independent, the community of developers is high skilled and it wants to contribute to the GNU project.

The license act as a device of intersement with its warranty of accomplishment of the interests of the actors. In this case we want to underline that the GNU GPL is the witness, as well as the website, of the involvement of GNUe in the GNU project, so the concepts expressed before about the license can be connected to GNUe. The same thing can be said looking at the project like one enrolled in the actor – network of the Free Software Foundation. The success of the project can not be evaluated, mainly because of the planning status of a large portion of the work.

5.4 Open for Business: choosing Open Source to develop high quality software

Open for Business (OfBiz) is a suite of business applications and a framework for developing such applications founded by Jones and Zeneski, born with the aim “to develop high quality open source enterprise software”, in a “community oriented” way. The license chosen is the MIT license. This is a simple non-copyleft license and is considered compatible with the GPL [13] and it allows the covered code to be used in proprietary software and do not require the distribution of Open Source version of the code.

The original developers are defined as skilled software engineers and architects, but for reaching the project goal, other competences are required, like accounting, sales and marketing. So the idea that other participants to the community life will bring these expertise, are they end user, services organizations or people who simply want to contribute. The community is defined in the characteristics of its members, the original developers are defined as leaders, the potential customers interests appears like a mix between in-house build or off-the-shelf buy, other companies can build their business on that, the code is based on adequate data models, the license is characterized as being “more open” than the GPL, as well as the development methodology started from a research of public standards disposable, the open source movement is changing irreversibly the way in which software is developed. Open for Business collects these interests and facilitate their satisfaction.

The use of the license act as a device of interessement, because it is considered able to allow the actors to reach their interests: customizations, low costs, transparency about the data model used, as well as the contribution of the project to the changes carried on by the open source community are elements protected by the license. The terms of the license, briefly discussed above, don't exclude GPLed software or proprietary one, making the involvement of other organizations more unstable. Another device of interessement is needed, for fixing the role of other companies: this is the creation of lists of Certified Affiliates, which lock customers, pulled to choose one of these companies for buying services, and the affiliates themselves, who recognize the leadership of the OfBiz project in strategic fields of their business. So the enrolment has taken place, and the mobilization is going on through the Open for Business website.

6 Conclusions: Different Translation Processes, Free/Open Source Licenses and Accountability

In this paper we focused on the translation processes enacted by four different Free/Open Source ERP projects, mainly to show how the choice of the license is connected to different goals, actors and strategies. The Free/Open Source ERP analysed appear to be part of networks of interconnected interests, like these of the developers, the customers, the partners, the Free Software community and the Open Source community. The choice of one particular license is a tool to interest and enroll actors, as well as it is a carrier of different views of the world, but its choice depends on the different interests of the individuals or organizations who launched the project. In our theoretical framework, the licenses appear able to enter these relational networks, contribute to the actor's definition and mobilizations, shaping the different meanings community, customers or partners have. The relevance of these networks of interconnected interests is shown, for example, by the research on accountability, studied in an historical period characterized by the transition from a "state manager" to a "regulatory state" [19].

Accountability is one of the main topic in ERP studies, and its conception is changing from a community related one, to a located one [20] involving responsibility in front of a public wider than the single community, that is in front of the society. The study of the relation between different accounts of ERP, as well as of different processes taking place contemporary to their development, shaping different actors, shows how the contribution by Free/Open Source ERP to the "accountability in a computerized society" [21] cannot be considered only in a technical way. This study can also bring to a reshaping of the concept of accountability, for example in the direction of understanding "how can this error be used to improve similar systems in order to improve the welfare of society?" [22].

We believe that more research following (and shaping) this theoretical framework can help to gain a deeper comprehension of the economical, political,

technological, and social aspects of Free/Open Source ERP, the success of their translation processes and their consequences.

References

1. S. Sismondo, *An Introduction to Science and Technology Studies* (Blackwell, Oxford, 2004).
2. M. Callon, *Some Elements of a Sociology of Translation: Domestication of the Scallops and the Fishermen of St Brieuc Bay*, First published in *J. Law, Power, Action and Belief: A New Sociology of Knowledge?* (Routledge, London, 1986), pp. 196-223.
3. J. Law, *Traduction/Trahisson: Notes on ANT*, published by the Centre for Science Studies, Lancaster University, Lancaster LA1 4YN (1997); <http://www.comp.lancs.ac.uk/sociology/papers/Law-Traduction-Trahisson.pdf> [31/01/2006].
4. B. Latour, *On Recalling ANT*, published by the Centre for Science Studies, Lancaster University, Lancaster LA1 4YN, (1998); <http://www.comp.lancs.ac.uk/sociology/papers/Latour-Recalling-ANT.pdf> [31/01/2006].
5. Garfinkel, *Studies in Ethnometodology* (Prentice-Hall, Englewood Cliffs, NJ, 1967).
6. L. Suchman, *Representing practice in cognitive science in Representation in scientific practice*, Michael Lynch (ed.), (MIT Press, Cambridge, 1990).
7. Schutz, *Scritti Sociologici* (UTET, Torino, 1979).
8. L. Winner, *Do Artifacts Have Politics?* *Daedalus* **109**, 121-136 (1980).
9. S. L. Star, *The Ethnography of Infrastructure*, *American Behavioural Scientist* **43**, 377-391 (1999).
10. B. Latour, *We Have Never Been Modern* (Harvard University Press, Cambridge, 1993).
11. Lanzara and M. Morner, *Artifacts Rule! How Organizing Happens in Open Software Projects*, in *Actor Network Theory and Organizing*. B. Czarniawska, T. Hernes (eds.), (Copenhagen Business School Press, Copenhagen, 2005).
12. P. Bourdieu, *Le sens pratique* (Edition de la maison des sciences de l'homme, Paris, 1980).
13. Free Software Foundation, *Various Licenses and Comments about Them* (2005); <http://www.gnu.org/philosophy/license-list.html> [31/01/2006].
14. S. De Paoli *An Actor-Network Theory for the GNU/GPL Software License*, Berlin. 21th EGOS Colloquium (2005).
15. D. Hakken, *The Knowledge Landscapes of Cyberspace* (Routledge, London, 2003).
16. C. Geertz, *Interpretation of Cultures* (Basic Books, New York, 1973).
17. P.T. Have, *Understanding Qualitative Research and Ethnology*. (Sage, London, 2004).
18. D.M. Berry, *The Contestation of Code: A Preliminary Investigation into the Discourse of the Free/Libre and Open Source Movement*, *Critical Discourse Studies* **1**(1), 65 - 89 (2004).
19. La Spina and G. Majone, *Lo Stato Regolatore* (Il Mulino, Bologna, 2000).
20. L. Suchman, *Located Accountabilities in Technology Production*. Sawyer Seminar on Heterarchies, Santa Fe Institute, October (2000).
21. Nissenbaum, *Accountability in a computerized society*, *Science and Engineering Ethics* **2**(2), 25-42 (1996).
22. S. David, *Opening the Sources of Accountability*, *First Monday* **9**(11), (2004).