Requirements Engineering for Cross-organizational ERP Implementation: Undocumented Assumptions and Potential Mismatches

Maya Daneva, Roel Wieringa
Department of Computer Science, University of Twente, The Netherlands
m.daneva@utwente.nl, roelw@cs.utwente.nl

Abstract

A key issue in Requirements Engineering (RE) for Enterprise Resource Planning (ERP) in a cross-organizational context is how to find a match between the ERP application modules and requirements for business coordination. This paper proposes a conceptual framework for analyzing coordination requirements in inter-organizational ERP projects from a coordination theory perspective. It considers the undocumented assumptions for coordination that may have significant implications for ERP adopting organizations. In addition, we build a library of existing coordination mechanisms supported by modern ERP systems, and use it to make a proposal for how to improve the match between ERP implementations and supported business coordination processes. We discuss the implications of our framework for practicing requirements engineers. Our framework and library are based on a literature survey and the experience with ERP implementation of one of us (Daneva). In future empirical research we will further validate and refine our framework.

1. Introduction

From their origin in material requirements planning, ERP systems have evolved into software packages that support coordination of different actors in a company. Current ERP systems contain modules not only for material management, but also for accounting, human resource management and all other functions that support business operations. In the past years, the role of ERP systems as coordination support has been extended to cross-organizational coordination. By 'cross-organizational' we mean that the ERP system is used by different independent, or nearly independent, businesses. For example, businesses cooperate with their customers, suppliers, and other stakeholders to form value webs. Large companies have structured themselves as sets of nearly independent business units, each responsible for their own profit and loss. ERP implementation is considerably more difficult in such a networked context than in an intra-organizational context because in a networked context, we have different business actors who make decisions based on their own local criteria. Different businesses have different infrastructures, different enterprise systems, different business processes, different semantics of data, different authorization hierarchies, and different decision centers. If these businesses decide to cooperate for a particular purpose, all these differences still exist and none of the participating businesses will be prepared to change their infrastructure, business processes, and semantics, just for this particular cooperation, or to reveal the confidential business rules embedded in their processes and applications.

ERP implementation is the customization and introduction of an ERP system in a (possibly networked) business. One of the most crucial tasks in such a project is requirements engineering, in which the properties of the ERP system to be implemented are aligned to the requirements of the busines(es) that will use it.

ERP vendors and their consulting partners offer standard RE processes for ERP projects. Recent research in ERP RE has identified flaws with these standard processes and proposed creative solutions to reduce the cost of ERP RE by avoiding scope creep, involving the right stakeholders, allocating sufficient resources, and enlisting vendors' and consultants' support to RE problems [3, 4, 8, 9, 10, 11, 16, 28, 34]. Nevertheless, the central problem of ERP implementation still exists: to find a match between the flexibility often required by the business, and the rigidity usually imposed by the ERP system. This problem is aggravated in a cross-organizational context because, as we will see, the rigidity of the ERP system is imposed by built-in assumptions about business semantics, business processes, business communication channels and business goals. If these hidden assumptions do not match the business, the business will experience the ERP system as being rigid and unable to meet the business requirements. In a networked context, there is a mismatch between the ERP and *each* of the participating businesses.

The present paper proposes to tackle this problem from the point of view of coordination theory [2,6,29,32,33]. Since ERP systems are coordination support systems, we should be able to identify the coordination mechanisms supported by an ERP system. If we explicitly specify these mechanisms in a cross-organizational setting, then the requirements engineer should be able to find a match between the coordination support offered by the ERP system and the coordination mechanisms selected by the cooperating companies.

We will present an inventory of coordination mechanisms implicitly assumed by ERP systems, and analyze the role that selection of these mechanisms plays in balancing rigidity imposed by an ERP system against the flexibility required by the cooperating organizations. We will see that rigidity will allow the benefits of cross-organizational cooperation to be reaped, whereas flexibility will decrease the benefits and, at the same time, increase the cost of implementing and maintaining the ERP system.

The paper is organized as follows: Section 2 provides background and related work. Section 3 presents our inventory of cross-organizational coordination mechanisms supported by ERP packages. Section 4 analyzes the mismatch between business flexibility and ERP rigidity and explains the impact of the coordination mechanisms on this problem bundle. Section 5 discusses the implications of our framework for ERP RE and offers some hypotheses that we think are worth further research efforts. Section 6 concludes with plans for future research.

2. Background and Related Work

This paper rests on our previous work in ERP RE [9,10,11] and the experiences published by [3,4,8,28,34]. Our research also builds on the works of Davenport [12,13,14], Hong and Kim [22], Gattiket and Goodhue [18], Markus [5,31], Scott & Kaindl [37], and Soh et all [39,40] in analyzing the alignment perspective in ERP projects. Although the ERP literature highlighs the issue of misalignments in ERP projects, the mere assertion that they arise from unmet organizational requirements masks the variety of sources of misalignments. Prior studies investigating the nature of the ERP mismatches are scarce. One exception is the misalignment typology suggested by Soh et all [40] who adopt a dialectic perspective and extend their initial mismatch categorization [39] by considering four pairs of opposing forces: (1) push towards integration versus differentiation, (2) process orientation versus functional orientation, (3) flexibility versus restrictiveness, and (4) package domain specifics versus organization domain specifics. However, their typology was empirically derived from cases in an inter-departmental integration project in one organization in the public sector. We explore these opposing forces in the area of cross-organizational ERP implementation.

The theoretical perspective that we found most helpful in examining the issue of misalignments and their sources was from the field of coordination [29]. Our point of departure is the observation that modern ERP systems are widely used as an administrative framework for planning, conducting and monitoring a large array of functionally segmented operations in ways that both (i) accommodate, in real time, the intrinsic cross-organizational interdependencies underlying these operations, and (ii) enable their control. Consequently, from coordination theory perspective, these systems can be viewed as coordination technology.

We follow Malone and Crowston [29] in defining coordination as the management of shared actions by different business actors. Classic coordination mechanisms distinguished in economic sociology include market-based coordination, in which goods and services are exchanges based on price, and relational coordination, in which actors work towards a common goal based on shared and implicit norms of behavior [2, 32, 33]. IT mechanisms to support coordination include, among others, shared ERP systems and Enterprise Application Integration middleware [30].

The question investigated in this paper is which ERP mechanisms are available to support different coordination mechanisms. The relevance for cross-organizational RE is that an analysis of the desired coordination mechanisms of a set of organizations will lead to requirements for ERP package implementation. Preferences about coordination mechanisms are usually hidden in the ERP packages and therefore lead to unpleasant surprises when the chosen ERP package turns out not to match at all the implicitly desired but undocumented coordination mechanisms.

To the best of our knowledge, so far there has been no systematic analysis of the role of coordination requirements in ERP projects. There is to date no unified framework for describing the various kinds of coordination mechanisms, nor a systematic set of rules for dealing with the coordination needs of organizations. Requirements engineers and business representatives still have to rely on their intuition and experience, and the problem of coordination is still being confronted in a largely ad-hoc fashion. Moreover, the few ERP publications that include coordination aspects in the assessments of ERP systems [17,41], describe these aspects only in general terms, without characterizing in detail differences between (i) how agreements on joint actions are achieved, and (ii) how the default coordination mechanisms in ERP address those needs. This vagueness makes it difficult to determine what alternative coordination mechanisms might be useful in a given organizational context or to directly translate these alternative coordination process designs into specifications of individual activities or uses of ERP to support a process (e.g., as part of a business process redesign effort [13,14,20]).

3. An Inventory of Coordination Mechanisms

We classify coordination mechanisms based on the scheme shown in Figure 1.

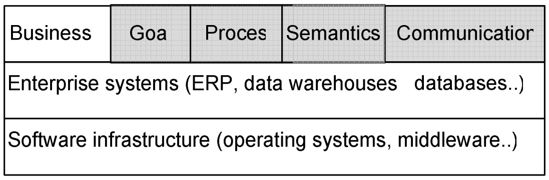
Business
Goa
Proces
Semantics
Communicatior
Enterprise systems (ERP, data warehouses databases..)
Software infrastructure (operating systems, middleware..)

Figure 1. The framework for business-IT alignment.

The horizontal layers classify entities in a service provisioning hierarchy: physical entities provide services to a software infrastructure, which provides services to enterprise systems, which provide services to businesses. We take four views on businesses: Businesses have goals, they perform processes, they communicate with one another, and while doing that, they exchange data with semantics. This framework is taken from our previous research on business-IT architecture alignment. For motivations we refer the reader to those papers [15,43].

Our interest is in the upper two layers of the framework because this is where the process and systems alignment in networked organizations takes place. A review of the literature on ERP implementation [5,7,12,13,14,23,25, 31,36,38,41] and of our own experience in implementing ERP solutions based on the SAP package [9,10,11] reveals that there is a small number of coordination technologies in use at the enterprise systems level:

- Shared database,
- Data warehouse,
- ERP functional application modules, and
- Workflow management systems.

There are additional infrastructure level technologies such as Enterprise Application Integration middleware, mobile technologies for information sharing, and, still experimentally, web service technologies. We surmise that there is little connection between coordination processes at the business level and integration technologies at the infrastructure levels, and we will not pursue this here. Integration technologies at the enterprise system level, on the other hand, have built-in assumptions about coordination processes at the business level, and this is the topic of this paper. Note incidentally that at the enterprise system level, ERP is but one of different possible integration technologies such as shared databases, data warehouses, ERP modules and workflow management systems [30]. Although these technologies can be provided together by one ERP vendor, a networked business may decide to use any multi-vendor combination of them and we consider them as distinct technologies. However, we hypothesize that at least some of our findings can be generalized to the other integration technologies at the enterprise system level.

Our review has yielded the following coordination mechanisms at the business level:

- *Goal-oriented mechanisms* referring to the partners' agreements on the goals of coordination.
- *Process-oriented mechanisms* concerned with establishing end-to-end inter-organizational processes, for example, client order fulfillment processes or product provisioning processes.
- *Semantics-oriented mechanisms* referring to the partners' agreements on the definition and the use of common meanings of key information entities.
- *Communication-oriented mechanisms* including the transmission and interpretation of information in the networked organization.

Table 1 describes which coordination mechanisms our study covers and how these are supported in state-of-the-art ERP systems. The table structures ERP support for coordination in a networked context and provides examples. It is meant to illustrate the different coordination mechanisms and is in no way exhaustive. Indeed, it can't be exhaustive, as new approaches to cross-organizational coordination are getting implemented on ongoing basis.

The crucial observation to make of these mechanisms is that each one starts with the word "shared". Now, inside one company it may often be true that these mechanisms are shared without everyone ever talking about them explicitly. After all, within one company we can assume that there is one culture and one shared way of doing things. But across different companies, what members of one business silently assume to be the normal way of working can be quite different from what people in another company silently assume about the normal way of working. And even within one company there may be severe mismatches, for example if the company consists of different nearly independent business units. Table 1 therefore represents a list of hidden assumptions that must be brought out in the open in cross-organizational ERP implementation. The assumptions about shared coordination mechanisms may be quite different across different business partners.

Types of mechanisms	Implicitly assumed coordination mechanisms	What it means to ERP adopters?	Examples
Goal-oriented mechanisms	Shared vision of the networked organization [13,14]	Presenting one face to clients and sharing corporate identity	Unified brand management by using common order management, sales force, service & marketing analytics applications.
	Shared view of services offered by network to clients	Motivating dependencies between services of different businesses	Updating supply chain partner's databases depends on a single event, e.g. when inventory is depleted to a critical amount.
Process-oriented mechanisms	Common agreement about business process environment [12,13,14]	Standardized operational procedures, access permissions, and control patterns	Common payment processing procedures
	Agreement about process-orientation [12,13,14]	Reducing organizational operations to a large series of procedural steps tied together to sequences, sub-functional categories, modules and cross-modular operations	Creating vendor master files or charts of accounts
	Common agreement on mngnt policies [13]	Sharing enforceable business rules that are explicit and consistent	Rules for tracking employee attendance and absence
	Solution maps [23]	Descriptions of the most important business processes within an industry sector, the technologies (ERP elements, add-ons) & services needed to support the processes	Branch-specific solution maps, like mySAP Aerospace & Defense solution map [23]
	Shared transaction processing engines [12]	Shared understanding of the position of ERP in the cross-organizational architecture	Trading partner portals, auction and exchange mechanisms, catalogs, as provided by Oracle Exchange [12] and SAP [23]
Semantics-oriented mechanisms	Shared data dictionary [7,9,13,25]	Common definitions of information entities	Maintaining a centralized view of company's customers and business partners data
	Shared reporting formats and semantics [13]	Standard presentation formats and information content of output	Integrating global data on site capacity, production and transportation costs, tariffs, and demand to schedule across multiple sites [13]
	Delegation about data access permission [38]	Distributed access to data and distributed application logic	Use of self-services to balance top-down control with bottom-up empowerment through open information sharing culture
	Common principles of cross-org. data management [14,23]	Data consistency [24] and alignment with businesses	Data ownership, data modularity, trust: no need of alternative sources to verify data accuracy [24]
	Reference models [7,23,25,38]	Representing practices embedded in the package in the form of reusable process and data models	The R/3 Reference Model [7,36]
	Shared product models [25]	Industry-specific solution aspects of the ERP package	Configurable master lists to allow product specification with variances, a requirement specific to the metal, paper and textile industries
Communication-oriented mechanisms	Agreements about communication channels and standards (at business level)	Shared understanding about information transmission and interpretation	ERP-package compliant XML schemas, by which e-network transactions are structured, e.g. RosettaNet for electronic industry and Acord for insurance [12]
	Sharing of knowledge	Bringing people to the required level of understanding to get their job done	Bringing information together based on user's role within the company (like SAP role-specific portals), or calling up customers' purchase history with the firm, external reports and discussion items from other sales and service staff members who have dealt with the customers [12]

Table 1 An inventory of cross-organizational coordination mechanisms.

Furthermore, if we compare the coordination support provided by today's ERP systems as indicated in Table 1 to the general coordination mechanisms as studied in coordination science [29], it becomes clear that the latter are too general to be helpful in making choices in ERP implementations. However, we agree with these authors that a coordination mechanism can be characterized by the extent to which it is suited to different organizational tasks, corporate cultures and environments. Thus, in case of cross-organizational ERP projects, coordination mechanisms vary in the degree to which coordination is prescribed at the time of RE, the cost in terms of time and effort associated with setting up the mechanism in question, and the degree of change this mechanism brings to the organization at the post-implementation stage.

Further research of the coordination perspective in the field of ERP may indicate the need for more or finer distinctions. However, for the time being, we do believe that this inventory is adequate for understanding the choices for coordination mechanisms that any ERP adopter is confronted with.

4. Causal Analysis

In this section, we analyze the role that the coordination mechanisms from Table 1 play in the clash between flexibility and rigidity typical for ERP implementation. We explored this link as documented in the literature [1, 5, 9, 10, 11, 12, 13, 14, 17, 18, 19, 20, 21, 22, 24, 25, 26, 27, 31, 35, 36, 37, 38, 39, 40, 41, 42] in order to understand which problems and misalignments could have been spotted and understood early in the ERP project, if the undocumented assumptions would have been brought up as part of ERP RE and coordination requirements would have been specified as part of the business requirements.

The review of our above cited sources let us derive the detailed problem dependency map (in both intra- and inter-organizational context) shown in Figure 3. As an introduction to it, we first present a high-level summary of the map in Figure 2. (In preparing this paper, we derived Figure 2 as an abstraction of Figure 3.) Figure 2 says that integration benefits increase through more sharing, e.g. sharing of standardized and harmonized processes and common data. Also, more sharing decreases the total costs of ownership. On the other hand, the more the organizational processes get integrated via the shared process and data environment, the more these get adapted to the default ERP structures, and so the more the change imposed on the organization and the more the organizational resistance to it. We believe Figure 2 maps the basic problem of rigidity versus flexibility in ERP implementation.

Detailed analysis of the reported experiences brought us to the multifaceted representation of the ERP problem space in Figure 3, in which the boxes represent typical issues that adopters encounter or try to avoid in ERP implementations, and the directed arrows show causality. The references that suggest the 33 links presented here are provided as appendix to this paper. Figure 3 shows generic domain knowledge about the impact of choices of an ERP system. It reflects the opposing forces of process standardization and process diversity [20] and, also, shows that tradeoffs need to be made between rigidity and flexibility in ERP solution design. To check if our library of coordination mechanisms can help making those tradeoffs, we investigated how the four groups of mechanisms fit into the multifaceted problem description in Figure 3.

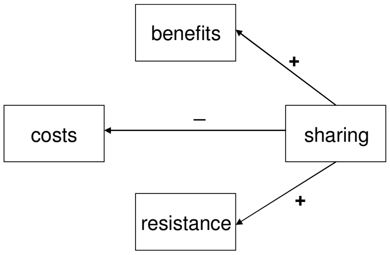

Figure 2. High-level representation of dependencies in ERP implementations.

We made the following observations: First, Figure 3 not only expands the four boxes of Figure 2 but it adds new insights into how benefits are shifting in dependence on the coordination decisions that may favor either process standardization (and rigidity of the ERP solution) or process diversity (and flexibility of the solution). For example, the more diverse the ERP adopters decide to keep their processes (and the more flexible the solution they want to design), the more the options for fostering creativity and maintaining the spirit of innovation in the organization [24]. The latter two benefits, however, will be of less consideration, if the ERP adopters decide on a higher level of process standardization. This decision, in turn, will favor the realization of the benefits due to sharing, namely, reduced transaction costs, organizational transparency, data visibility, data accuracy.

Second, Figure 3 reflects the fact that benefits from bringing ERP in never come cheaply. It makes it clear what price ERP adopters should expect to pay in order to realize the benefits due to sharing or the ones due to flexibility. Getting a more flexible solution means customizing the system to fit the business processes, which also means that cost, like customization (29), maintenance (30) and testing costs (33), and risks, like customization (27), system performance (31) and release lag risks (26), will increase. On the other side, opting for more sharing means incrementally or radically changing (cross-) organizational business processes to fit the system. It means less customization and maintenance costs as well. The price for implementing these changes, though, comes in the form of costs for managing and coordinating large-scale business process changes and coping with politics, resistance and corporate inertia [5,39,40].

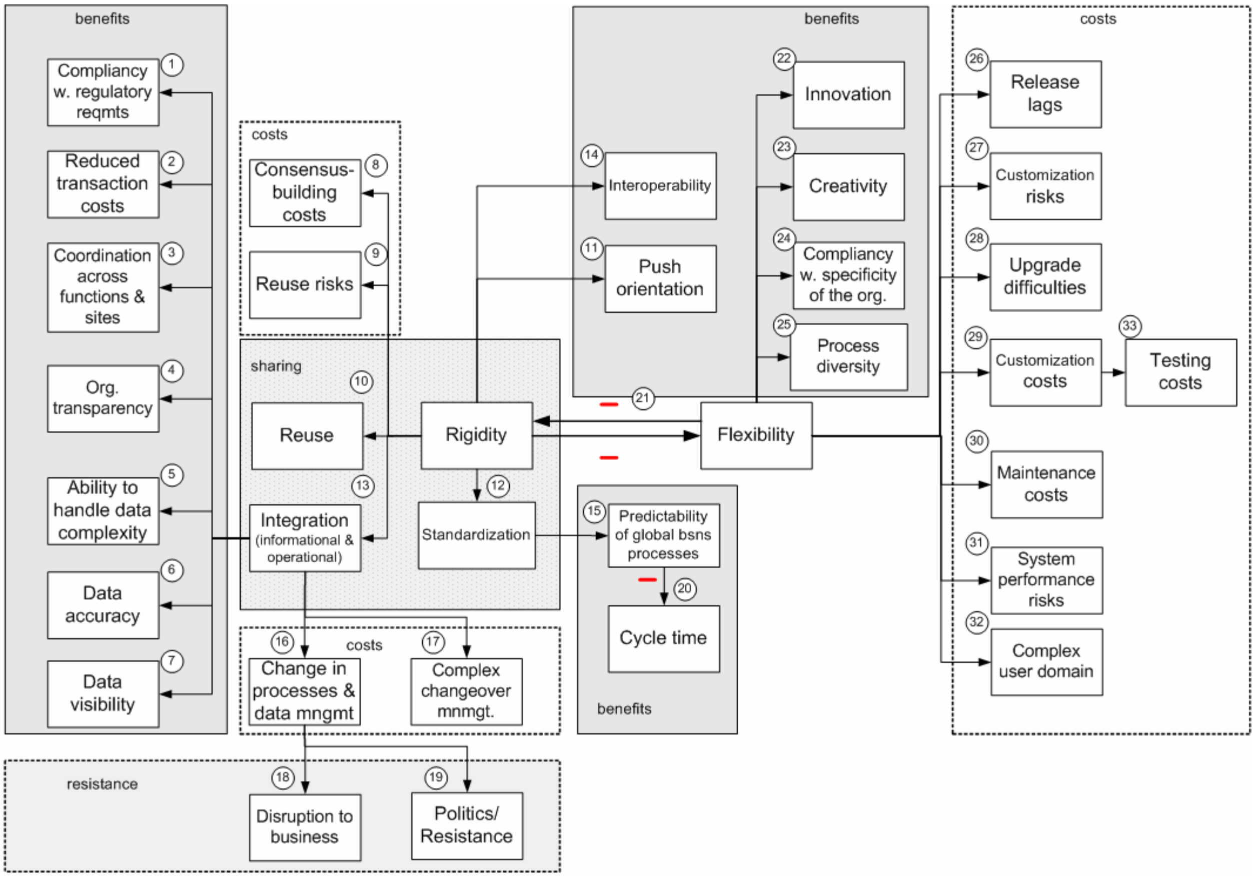

Figure 3. The problem dependency map in ERP Implementation practice.

Third, as the coordination mechanisms from Table 1 are available for ERP clients to achieve sharing, they clearly tend to support rigidity, reuse (10), standardization (12), and integration (13). These observations and the fact that the coordination mechanisms from our library are all about sharing encouraged us to consider substituting the boxes in the sharing-labeled gray area in Figure 3 with the groups of coordination mechanisms from Table 1. Indeed, if we replace boxes 10, 12, and 13 in Figure 3 with the groups of coordination mechanisms from our library, one can clearly see how our library fits in and what type of problems it could potentially help to explain (Figure 4):

Goal-oriented mechanisms. An orientation towards shared goals pushes the partners in the networked organization towards (i) developing a clear sense of who and what they are, and (ii) certainty as to how the network delivers value to customers and how it differentiates itself. In order for networked organizations to be economically advantageous, partner companies have to issue and receive transactions directly to and from their ERP systems without any human intervention [12]. This requires changes in how things get done internally in each of the partner's organizations (16), as handling business transactions is no longer limited by organizational boundaries. This would most likely be in opposition to other forces arising from each partner's organization having their own systems.

Process-oriented mechanisms. A process orientation in an inter-organizational ERP context means predisposing each company to manage itself along business processes. Also, it means shared process ownership. This requires redistribution of management responsibility in each of the partner companies as shared process ownership can't be imposed on a fragmented organization [20]. Differences between new agreements on shared processes and previously used business practices typically lead to disruption to business (18), and, ultimately, increased resistance to change (19).

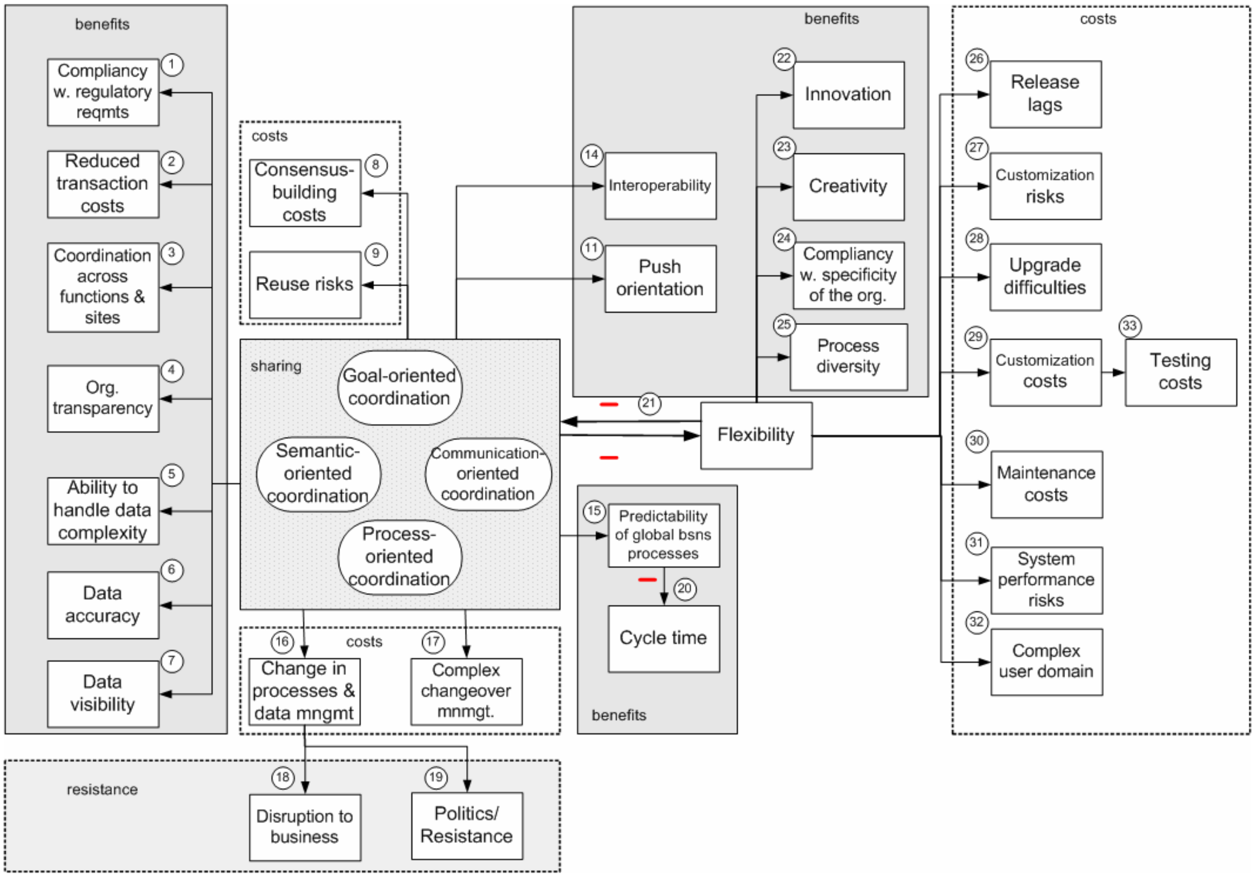

Figure 4. The problem dependency map: a coordination perspective.

Semantics-oriented mechanisms. Next, a semantics orientation implied through the coordination mechanisms pushes the partner companies towards adopting a common terminology for those areas of business activity in which the partner want to do things together. This gives rise to cross-organizational issues in terms of common data structures, data ownership and mastering, data flow transparency, responsibility for data entry and updates, and related changes in workflow [40].

Communication-oriented mechanisms. Finally, communication-oriented coordination mechanisms tend to support interoperability standards that directly contribute to building "collaborative communities" [12]. The underlying assumption is that partner companies have to realign horizontally and the variety of shared tasks that are performed requires less flexibility. In networked context, however, this assumption contradicts with the need of each of the partner organizations to dynamically build connections that handle specific portion of the shared process, that change as business opportunities arise, are taken advantage of, then abandoned as their value diminishes.

The above observations give an indication that the library of coordination mechanisms has its corresponding components in the problem dependency map (in Figures 3 and 4). Thus, our preliminary analysis allows us to draw two early conclusions: First, exploring coordination requirements by addressing the 15 coordination mechanisms in our library (Table 1) can be an important step towards unpacking the dimensions on which misalignments can arise. Second, the library of mechanisms can serve as a preliminary inventory on which critical inter-organizational integration issues, costs and risks can be surfaced early in the ERP project.

5. Conclusions and Further Research

In this paper, we attempted to show what the role of the undocumented built-in ERP assumptions is in inter-organizational ERP RE. We took an inventory of existing coordination mechanisms and mapped them onto typically encountered problems identified in empirical studies. We presented a perspective that, we believe, helps the requirements engineers to develop an understanding of the opportunities and issues associated with the ERP coordination mechanisms as undocumented assumptions: First, our problem dependency map is a problem domain theory; it allows the requirements engineers to reason about the impact of choices. Second, the undocumented assumptions make the coordination choices more explicit. Our library not only can facilitate interdisciplinary transfer of knowledge about ERP-supported coordination, it provides a guide for analyzing organization-specific coordination needs and generating alternative ways to fulfilling them. The variety of coordination mechanisms that we analyzed and included in our library is not found in previous research. Also, we provided a start in the direction of how to organize these coordination mechanisms. In addition, we used real life examples to motivate our analyses.

Each directed arrow in our problem dependency map represents a hypothesis that can become a subject of future empirical validation studies. Thus, for IS scientists, we formulated 33 hypotheses with a very preliminary analysis that indicates that it will be useful to do this research.

We believe that our approach provides a meaningful starting point in classifying ERP misalignments. However, for the framework to be useful at application and project level in the long-term and to progress from its current state, more analytic capabilities need to be built-in. Therefore, our immediate plans are to use it as a vehicle to explain typical misalignment phenomena in cross-organizational implementations and to refine it based on experiences we will collect in case studies.

As our proposal rests on cases from the ERP implementation practice, we are interested in knowing if our ideas can be extended to projects implementing other technologies for inter-organizational integration, like data warehouses, workflow management systems, or Enterprise Application Integration middleware [30]. This will be subject to validation in field research too.

Given ERP coordination mechanisms support a variety of intra- and inter-organizational interactions [17], to design a new RE process for cross-organizational ERP implementations, it will be useful to consider alternative coordination mechanisms that could be used to manage data and process sharing. One question that comes out of this paper and, we think, seems worth exploring, is: In what ways an ERP system can be arranged differently while achieving the same goals? Understanding the coordination problems addressed by a networked business suggests alternative coordination mechanisms that could be used, thus creating a space of possible business process designs.

6. Acknowledgement

The authors would like to thank Dr. Pascal van Eck and the anonymous reviewers for their comments and suggestions that led to the improved version of our paper.

7. References

[1] Al-Mashari, M., Process Orientation through Enterprise Resource Planning: A Review of Critical Issues, Knowledge and Process Management, 8(3), 2001, pp. 175-185.

[2] Alstyne, M., The State of Network Organizations: A Survey in Three Frameworks, Journal of Organizational Computing, 7(3), 1997, pp.83-151.

[3] Alvares, R., J. Urla, Tell Me a Good Story: Using Narrative Analysis to Examine Information Requirements Interviews during an ERP Implementation, The DATA BASE for Advances in IS, 33(1), pp. 38-52

[4] Arinze, B., M. Anandarajan, A Framework for Using OO Mapping Methods to Rapidly Configure ERP Systems, Comm. of ACM, 46(2) pp. 61-65.

[5] Brehm, L, A. Heinzl, M.L. Markus, Tailoring the ERP Systems: a Spectrum of Choices and Their Implications, Proc. of the Hawaii Conf.on Systems Sciences, 2004.

[6] Clemons E.K., S. P. Reddi, M.C. Rows, The Impact of Information Technology on the Organization of Economic Activity: The ``Move to the middle'' Hypothesis, Journal of Management Information Systems, 10(2), 1993, pp. 9-35.

[7] Curran, T., A. Ladd, SAP R/3 Business Blueprint, 2nd ed., Prentice Hall, 2000.

[8] Dalal, N.P., M. Kamath, W.J. Kolarik, E. Sivaraman, Towards an Integrated Framework for Modelling Enterprise Processes, Comm. of ACM, 47(3) 2004, pp. 83-87.

[9] Daneva, M., ERP Requirements Engineering Practice: Lessons Learnt, IEEE Software, 21(2), 2004, pp. 26-33.

[10] Daneva, M., Patterns of Success and Failure in ERP Requirements Engineering Proc. 12th Int'l Workshop on Software Metrics, Shaker, Aachen, 2004, pp. 527-546.

[11] Daneva, Using Maturity Assessments to Understand the ERP Requirements Engineering Process, Proc. Joint Int'l Requirements Engineering Conf., IEEE CS Press, 2002, pp.255–262.

[12] Davenport, T., The Future of Enterprise System-Enabled Organizations, Information Systems Frontiers 2(2), 2000, pp. 163-180.

[13] Davenport, T., Mission Critical: Realizing the Promise of Enterprise Systems, HBS Press, 2000.

[14] Davenport, T., Putting the Enterprise into the Enterprise System, Harvard Business Review, 76(4), 1998, pp. 121-131.

[15] Eck, P.A.T. van, H. Blanken and R.J. Wieringa, Project GRAAL: Towards Operational Architecture Guidelines, IJ of Cooperative Information Systems, 13(3), 2004, pp. 235-255.

[16] Esteves, J., J.A. Pastor: Establishing the Importance of ERP Implementation - Critical Success Factors along ASAP

Methodology Processes, Proc. of Int'l. Conf. on Enterprise Information Systems, 2001, pp. 182-187.

[17] Fan, M., N. Stallaert, A.B. Whinston, The Adoption and Design Methodologies of Component-based Enterprise Systems, European Journal of IS, 9, 2000, pp. 25-25.

[18] Gattiker, T.F., D.L. Goodhue, Understanding the Local-level Costs and Benefits of ERP through Organizational Information Processing Theory, Information & Management, 41, 2004, pp. 431-443.

[19] Grossman, T., J. Walsh, Avoiding the Pitfalls of ERP System Implementation, Information Systems Management, Spring 2004, pp. 38-42.

[20] Hammer, M., S. Stanton, How Process Enterprises Really Work, Harvard Business Review, 77(6), 1999, pp. 108-118.

[21] Hirt, S.G., E.B. Swanson, Emergent Maintenance of ERP: New Roles and Relationships, Journal of Software Maintenance & Evolution: Research and Practice, 13, 2001, pp. 373-397.

[22] Hong, K.-K., Y.-G. Kim, The Critical Success Factors for ERP Implementation: an Organizational Fit Perspective, Information & Management 40, 2000, pp. 25-40.

[23] Kagermann, H., G. Keller, MySAP.com Industry Solutions, Addison Wesley, 2000.

[24] Kallinikos, J., Deconstructing Information Packages: Organizational and Behavioral Implications of ERP Systems, Information Technology & People, 17(1), 2004, pp. 8-30.

[25] Keller G. and T. Teufel, SAP R/3 Process Oriented Implementation, Addison-Wesley, 1998.

[26] Luo W., D.M. Strong, A Framework for Evaluating ERP Implementation Choices, IEEE Trans. On Engineering Management, 51(3) 2004, pp.322-333.

[27] Madapusi, A., D. D'Souza, Aligning ERP Systems with International Strategies, Information Systems Management, Winter 2005, pp. 7-17.

[28] Maiden N.A., C. Ncube, "Acquiring COTS Software Selection Requirements," IEEE Software, 15(3), 1998, pp. 46–56.

[29] Malone, T., K. Crowston, The Interdisciplinary Study of Coordination, ACM Computing Surveys, 26(1), 1994, pp. 87-119.

[30] Markus, M. L., Paradigm Shifts - Business and Business /Systems Integration, *CAIS* 4 (10) 2000.

[31] Markus, M.L., C. Tanis, P.C. v. Fenema, Multisite ERP Implementations, Comm. of ACM, 43(4), 2000, pp. 42-87.

[32] Miles, R.E. and C.C. Snow, Causes of Failure in Network Organizations, California Management Review, 34(4) 1992, pp. 53-72.

[33] Powell, W.W., Neither Market nor Hierarchy: Network Forms of Organization, Research in Organizational Behavior, 12, 1990, pp. 295-336.

[34] Rolland, C., N. Prakash, Bridging the Gap between Organizatopnal Needs and ERP Functionality, Requirements Engineering, 5, 2000, pp. 180-193.

[35] Ross, J. W., M. R. Vitale, The ERP Revolution: Surviving vs. Thriving, Information Systems Frontiers 2(2), 2000, pp. 233-241.

[36] Scheer, A.-W., F. Habermann, Making ERP a Success, Comm of ACM, 43(4), pp. 57-61.

[37] Scott, J., L. Kaindl, Enchancing Functionality in an Enterprise Software Package, Information & Management, 37, 2000, pp. 111-122.

[38] Selchert, M., Enhanced Project Success through SAP Best Practices - International Benchmarking Study, ISBN 1-59229-031-0, SAP Press, 2004.

[39] Soh C., S.S. Kien, J. Tay-Yap, Cultural Fits and Misfits: Is ERP a Universal Solution? Comm. of ACM 43(4), 2000, pp. 47-51.

[40] Soh, C., S.K. Sia, W.F.Boh, and M. Tang, Misalignments in ERP Implementations: a Dialectic Perspective, IJ HCI, 16(1)2003, pp. 81-100.

[41] Stefanou, C.J., A Framework for the Ex-ante Evaluation of ERP Software, European Journal of IS, Special Issue on IT Evaluation, 10(1) 2001, pp.204-215.

[42] Stevens, C., Enterprise Resource Planning: a Trio of Resources, Information Systems Management, Summer 2003, pp. 61-67.

[43] Wieringa R.J., H.M. Blanken, M.M. Fokkinga, and P.W.P.J. Grefen, Aligning Application Architecture to the Business Context, Proc. Of Conference on Advanced Information System Engineering (CAiSE 03), LNCS 2681, Springer, 2003, pp. 209-225.

8. Appendix

Link	Proposition and Relevant References
1	The tighter the integration of operational and informational procedures, the better the compliancy with regulatory requirements [31,39,40].
2	The tighter the integration of operational and informational procedures, the lower the transaction costs [12,13,14,20].
3	The tighter the integration of operational and informational procedures, the better the coordination across functions and sites [1,12,13,14,31,39].
4	The tighter the integration of operational and informational procedures, the more transparent the organization [12,13,14,39,40].
5	The tighter the integration of operational and informational procedures, the better the ability of the organization to handle data complexity [12,13,14,24,25].
6	The tighter the integration of operational and informational procedures, the higher the level of data accuracy that the organization can achieve [12,13,14].
7	The tighter the integration of operational and informational procedures, the better the level of data visibility that the organization can achieve [12,13,14,39,40].
8	The more rigid the solution, the higher the costs of building consensus among stakeholders [24].
9	The more rigid the solution, the higher the reuse risks [9,10,11].
10	The more rigid the solution, the higher the levels of reuse an organization can achieve [7,9,10,11,25].
11	The more rigid the solution, the stronger the push towards inter- and intra-organizational integration [5,12,13,14,20,31,35,39,40].
12	The more rigid the solution, the more standardized the business processes [5,9,12,13,14,17,20,24,25,26,27,36,39, 40,42].
13	The more rigid the solution, the tighter the integration of operational and informational procedures that the organization can achieve [1,5,9,10,11,12,13,14, 17, 18,19,20,21,22,24,25,26,27,31,35,36,37,38,39,40,41,42].
14	The more rigid the solution, the better the level of interoperability that can be achieved [1,24,25].
15	The more rigid the solution, the more predictable the global business processes [12,13,14,20,25].
16	The tighter the integration of operational and informational procedures, the greater the changes in processes and data management imposed on the organization [1,5,9,10,11,12,13,14,17,18,19,20,21,22,24,25,26,27,31,35,36,37,38,39, 40,41,42].
17	The tighter the integration in terms operational and informational procedures, the more complex the changeover management processes [5,13].
18	The greater the changes in processes and data management imposed on the organization, the greater the disruption to business [12,13,14,20,31,39,40].
19	The greater the changes in processes and data management imposed to the organization, the more organizational resistance to them and the more the potential sources for political issues [1,12,13,14,19,20,31,39,40].
20	The more predictable the global business processes, the less the cycle time and the better control over cycle times [12,13,14,20,25].
21	The more rigid the solution, the less flexibility it offers to business users [1,5,9,10,11,12,13,14,17,18,19,20, 21,22,24,25,26,27,31,35,36,37,38,39,40,41,42].
22	The more flexible the solution, the more the options for fostering innovative thinking [22,24].
23	The more flexible the solution, the more the options for inventing creative ways of working [22,24].
24	The more flexible the solution, the more compliant it is with the specifics of the organization [39,40].
25	The more flexible the solution, the more diverse the organizational business processes [1,5,9,10,11,12,13,14,17,18,19, 20,21,22,24, 25,26,27,31,35,36,37,38,39,40,41,42].
26	The more flexible the solution, the higher the risks of release lags [21,41].
27	The more flexible the solution, the higher the customization risks [5,12,13,14,26,27,31].
28	The more flexible the solution, the more the upgrade difficulties [5,12,13,14,26,31].
29	The more flexible the solution, the higher the customization costs [9,12,13,14,21].
30	The more flexible the solution, the higher the maintenance costs [9,12,13,14,21].
31	The more flexible the solution, the higher the system performance risks [39,40].
32	The more flexible the solution, the more complex the user domain (e.g. the more the data views that need to be consolidated, the more the interfaces that need to be maintained) [9,12,13,14,21].
33	The bigger the scope of customization, the more the testing efforts that are required [9,12,13,14,21].