December 2004

Alignment in Enterprise Systems Implementations: The Role of Ontological Distance

Michael Rosemann
Queensland University of Technology

Iris Vessey
Indiana University

Ron Weber
Monash University

Follow this and additional works at: http://aisel.aisnet.org/icis2004

Recommended Citation
http://aisel.aisnet.org/icis2004/35

This material is brought to you by the International Conference on Information Systems (ICIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in ICIS 2004 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.
ALIGNMENT IN ENTERPRISE SYSTEMS IMPLEMENTATIONS:
THE ROLE OF ONTOLOGICAL DISTANCE

Michael Rosemann
Centre for IT Innovation
Queensland University of Technology
Brisbane, Queensland, Australia
m.rosemann@qut.edu.au

Iris Vessey
Kelley School of Business
Indiana University
Bloomington, IN U.S.A.
ivessey@indiana.edu

Ron A. G. Weber
Faculty of Information Technology
Monash University
Caulfield East, Victoria, Australia
ron.weber@infotech.monash.edu.au

Abstract

The development, implementation, operation, support, maintenance, and upgrade of enterprise systems (ES) have given rise to a multibillion dollar industry. Nonetheless, this industry has been perceived as having difficulties in achieving an adequate return on investment. We believe that a major reason why organizations encounter problems is that they fail to understand properly how well an ES package aligns with or fits their needs. Misfits are external manifestations of the differences between two worlds: that of the organization’s needs on the one hand and the system’s capabilities on the other.

To address issues of fit in ES package implementation, we need a way to describe and evaluate fit. To achieve this end, we use the concept of ontological distance. Specifically, we provide an approach to evaluate the significance (size) of a gap between the world of organizational requirements and the world of system capabilities. The greater the distance between the organizational world and system world, the more likely we believe that organizations will encounter difficulties in their engagement with an ES package. Distances arising from the joint evolution of organizational requirements and system capabilities occur at a number of stages during the process of implementing an ES.

From a theoretical perspective, our conceptual development shows how measures of ontological distance can be used to predict the likely deficiencies an organization will have on implementing an enterprise system. From a practical perspective, our research helps organizations to avoid problems when implementing an enterprise system and to select the best package for the organization’s needs.

Keywords: Enterprise systems, ontology, alignment

Introduction

Over the past decade, ready-to-install software in the form of enterprise systems (ES) packages has resulted in a fundamental shift in how information systems are developed in user organizations (Sawyer 2001). These systems seek to provide their users with comprehensive, integrated support for their information system needs (Shang and Seddon 2002). They include enterprise resource planning (ERP) software (Klaus et al. 2000), customer relationship management (CRM) software, and supply-chain management
(SCM) software. Embedded within ES packages are business models that their designers believe represent best practice in certain contexts. ES packages require parameterized input to allow their generic capabilities to be configured to better support their users’ particular business needs.

The development, implementation, operation, support, maintenance, and upgrade of enterprise systems have given rise to a multibillion dollar industry. This industry is replete, however, with stories of high-cost problems. Many ES implementation projects run late, and some even fail (Scott and Vessey 2002). Moreover, as organizations’ use of ES matures, the high costs of operating and maintaining them have become apparent (Nolan Norton Institute 2000). Many organizations now report that achieving an adequate return on investment in ES has become a major priority (Accenture 2002; Stein and Hawking 2002).

Research that seeks to understand the reasons for these problems is relatively recent (e.g., Markus et al. 2000). Based on our prior research and experience with ES packages, we believe that a major reason why organizations encounter problems when they purchase and use them is that they fail to understand properly how well an ES package aligns with or fits their needs. Furthermore, in a large-scale survey of mid-size European organizations, van Everdingen et al. (2000) found that alignment or fit of an ES package with an organization’s needs was by far the most important criterion used in selecting a package.

The alignment issue has been studied by examining misfits identified following ES implementations (Sia and Soh 2000; Soh et al. 2003; Soh et al. 2000). Although certain misfits can be regarded as exogenous alignment issues, such as those specific to a particular country, sector, or industry, others are organization-specific. To a large extent, these latter types of alignment issues are under the control of organizations via the choices they make about tailoring the ES during implementation. In this regard, Brehm et al. (2001) identify nine different ways in which an organization can improve the fit between organizational needs and package capabilities. These choices range from configuration (setting parameters in tables to specify how the package should function in certain circumstances), to customization (e.g., employing user exits to extend the functionality of the package via published software interfaces), to modification (changing source code). Each way of addressing misfits potentially leads to its own set of problems.

Misfits manifest differences between “two worlds.” One world reflects the organization’s needs. The other reflects the package’s capabilities. To improve ES implementations, this gap needs to be minimized. Alternatively, the overlap between the two worlds needs to be maximized.

In this paper, we elucidate the idea of alignment between an ES and the needs of the user organization using the concept of ontological distance. We use the term ontological distance as a label for the extent of the difference between the capabilities embedded within an ES package and the capabilities that an organization needs to be able to operate effectively and efficiently. Ontology is the branch of philosophy that seeks to articulate models of the real world in general (Bunge 1977). Each ES package is developed to support certain types of “worlds.” If organizations are to survive and prosper, however, they need to operate in particular types of worlds that are appropriate to their mission and goals. The greater the distance between the former worlds and the latter worlds, the more likely we believe that organizations will encounter difficulties in their engagement with an ES package.

From a theoretical perspective, our conceptual development shows how measures of ontological distance can be used to predict the likely deficiencies an organization will have on implementing an enterprise system. From a practical perspective, our research helps organizations to avoid problems when implementing an enterprise system and to select the best package for their needs. Over the lifetime of an ES implementation, our notions can also be used to aid in making the decision to upgrade or further customize the system.

The next section defines what we mean by ontological distance and presents the concepts required for its measurement. We then present a model that shows the way in which misfits arise during the ES implementation process. Finally, we discuss the findings of our study and present the implications for future research.

**Concept of Ontological Distance**

In this section, we seek to characterize ontological distance. Specifically, we provide an approach for evaluating the significance (size) of a gap between the world of organizational requirements and the world of system capabilities. We characterize the gap from the viewpoint of a selected ontology. We first discuss the nature of ontological distances, in general. Next, we examine different situations that can occur in identifying those distances. Finally, we show how an ontology can be used to measure these distances.
Classification of Distances

Figure 1 presents the constructs involved in our examination of gaps between the user requirements and package capabilities. Circle A on the left includes all organizational requirements that are deemed relevant at some stage in the system selection and implementation process. Conceptually, it includes any requirement that some stakeholder might deem necessary at some time during the selection of an ES package and the implementation of the chosen package.

Circle B includes those requirements perceived as relevant at some point in time during the system selection or implementation process (i.e., requirements 1 through 7). It represents the first pass or initial specification of organizational requirements at a particular stage of the system selection and implementation process. For example, at an early stage of the system selection and implementation process, it might include only those key requirements that organizational stakeholders use to evaluate an ES package to determine whether it is worthwhile investigating the package in more detail.

Circle C includes the requirements that are actually relevant at the same point in time that circle B is identified. Such a selection could result from a dialogue between experts who specialize in the implementation of ES packages and representatives of the organization who are familiar with the companies’ strategies, objectives, and expectations. The difference between these two circles, B and C, highlights the potential impact of selecting nonrelevant requirements (see requirements 1 and 2 in Figure 1). It also emphasizes the fact that certain relevant requirements could be excluded from the analysis (see requirements 8 and 9 in Figure 1).

These requirements are then mapped to the capabilities of the ES. The capabilities of the system can be viewed from three perspectives. They could be the actual capabilities (circle D), the perceived capabilities (circle E), or the appropriated capabilities (circle F) (Orlikowski and Robey 1991). Appropriated capabilities are the capabilities of the system as viewed by its users; they reflect the way in which users actually use the system.

The distances identified can now be further classified using this figure.

- First, a decision has to be made regarding the relevant organizational requirements. Thus, distances may be relevant or non-relevant. Nonrelevant distances result from the analysis of nonrelevant requirements (B\(\cap\)C). The analysis of nonrelevant requirements is not only a time-consuming and non-value adding activity, it also impacts in a misleading way the calculation of the ontological distance.

- Second, distances can be differentiated in terms of actual and perceived distances. Once the relevant requirements have been identified, a decision has to be made about the extent to which the actual system capabilities should be explored. Perceived distances map the actual organizational requirements against the perceived system capabilities (i.e., they are not based on a detailed analysis of the system, but rather on assumptions about the system capabilities). It is important to understand that insufficient analysis of the system capabilities can provide a wrong impression of the fit of the system with the organizational requirements and can negatively impact the calculation of the ontological distance.
• Third, a decision has to be made regarding the scope of the system capabilities. The focus could be on the “vanilla” system capabilities only (actual: circle D), or it could include appropriated system use (circle F). The differentiation between actual and appropriated capabilities is important for a clear definition of the unit of analysis.

If, in seeking to reduce complexity, we focus only on the relevant requirements (circle C in Figure 1), the following situations can be differentiated when mapping those requirements to the (actual) system capabilities (Figure 2).

First, system completeness refers to the situation in which there is a one-to-one mapping between an organizational requirement and a system capability. Second, system excess reflects system capabilities that do not correspond to any organizational requirements. This situation can be further differentiated: (1) these capabilities might not be relevant at the time that system capabilities are analyzed, but might become relevant later in the system lifecycle; (2) the capabilities may never be relevant to the organization. Third, system deficit includes those organizational requirements that cannot be mapped to any system capability. Again, there are two possibilities: (1) the requirements may not be meaningful, justified requirements; (2) they may be justified requirements, but the system may be unable to support them.

Note that we address issues of one-to-many and many-to-one mapping as aspects of system completeness. For example, one organizational requirement may be supported by more than one system capability, and many organizational requirements may be supported by the same system capability. The ontological distance of one enterprise system consists therefore of an evaluation of the extent of system completeness and an evaluation of the combined system excess and system deficit, which could lead to potential misfits.

The Nature of Ontological Distance

The previous section described what should be measured when it comes to the distance between the world of organizational requirements and system capabilities. We now address how these distances can be measured by using the concept of ontological distance. That is, we can gain a greater understanding of these distances by evaluating them in terms of an ontological model. For the purpose of our research, we selected the Bunge-Wand-Weber (BWW) ontological model (Wand and Weber 1989, 1990a, 1990b, 1993, 1995; Weber 1997). The core of this ontology is formed by the representation model, which defines constructs such as thing (e.g., employee), property (e.g., the attribute age of an employee), and system (e.g., an organizational unit), as well as their relationships.

Characterizing gaps in ES requirements using ontological distance has three major benefits over characterizing them as misfits. First, it is possible to cluster the identified mappings into ontological constructs such as thing, property, and system. Different distances can then be discussed in light of the underlying constructs. Second, relationships between the distances can be identified...
with reference to the relationships between the ontological constructs. An example would be a gap (distance) related to a property, which can be linked to the referenced thing(s). Further, this approach potentially provides the opportunity to identify cause-effect relationships among the misfits of an ES. Third, these ontological classifications can be used to derive weights for the groups based on the significance of the ontological construct. Such a weighting can help to identify the more critical gaps.

The following criteria provide some insights into the possible design of such weights.

- A distance that refers to a thing should receive a higher weighting than a distance related to a property. Things are perceived as more important (of higher weight) than properties because the real-world is made up of things, which are further characterized by properties.

- A mutual property (e.g., “is supervisor of”) describes a property that derives its meaning only in terms of other things, while an intrinsic property (e.g., “age of an employee”) is a property of a thing that does not require any other thing. A mutual property is perceived as more important than an intrinsic property because it is related to two or more things and therefore has impact beyond a single thing.

- Within mutual properties, we perceive binding mutual properties, which make a difference to the involved things (e.g., “is project manager of”), as being of higher weight than non-binding mutual properties (e.g., “is older than”).

- In an ontology, it is typically possible to identify super-type–subtype relationships. An example in the BWW ontological model is the super-type event, which can be further differentiated into the subtypes internal/external event, or poorly defined/well-defined event. A distance that relates to a super-type should receive a higher weight than a distance that refers to a subtype because a missing subtype is less significant.

- Laws that traverse two or more things should be assigned a higher weight than laws that apply only to one thing. The former constrain more state spaces and more event spaces than the latter.

- In addition to weights that can be assigned to different ontological constructs based on their nature, weights will also be influenced by the number of occurrences of a construct. In the evaluation of the MRP II solution within an enterprise system, for example, it will be possible to identify a number of things such as material, asset, tool, and employee. The more instances of an ontological construct, the higher should be its weight.

Such an ontologically based weighting of organizational requirements or system capabilities can be used to derive default values for system selection criteria. These values can be over-written based on individual organizational needs. However, they provide, at least, a theoretically based starting point to determine such weights.

**Reference Frame for Assessing Ontological Distance**

Having provided a framework for measuring the alignment between the ES and the user organization, we now need to understand the context in which ontological distances arise. To do so, we introduce the notion of a reference frame (Bunge 1977).

Our analyses suggest that the implementation process is evolutionary in nature, resulting from a continual adaptation of perceived user requirements in the context of what is offered by the software system(s) under consideration. The model shown in Figure 3, therefore, covers the entire process of selecting a new ES solution from the first specification of the organizational requirements to the final and revised organizational requirements based on the system as actually implemented. The left-hand side of the figure captures the development of the organizational requirements for such large systems, and thus forms the demand side. The right-hand side of the figure essentially describes the system lifecycle. As such, it captures the supply side (i.e., the models, theories, and systems that are available when implementing an ES). Various interrelationships exist between the elements of the organizational requirements side and those of the systems side. Each relationship shown represents a distance between an organizational requirement at a point in time and a system capability.

**Evolution of Organizational Requirements in the ES Context**

Next, we characterize the different stages on both sides of the model, as well as the identified distances between related elements.
Problem Domain

ES are applications that provide integrated solutions for the most common functional areas and business processes of an organization. As such, they typically rely on a plethora ($n$) of existing models and theories ($S_{1n}$). For example, such models could be based on certain accounting standards, approaches for cost management (e.g., activity-based costing), purchasing models (e.g., dynamic lot-size models), or specific approaches for operations management (e.g., optimized production technology, OPT). These models describe a selected problem domain that is independent of any system implementation. They form the core body of knowledge of the business community and academia.

The development of ES depends to a large extent on the existence and quality of such models and theories. ES may be differentiated based on the selected business models and theories used to conceptualize their domains. During the process of developing an integrated software solution, these models and theories are often modified, extended ($S_{2m}$), or eliminated.

From an organizational viewpoint, the system selection process starts by deriving an initial list of organizational requirements ($O_1$). To a large extent, these requirements are based on current practice (i.e., the relevant part of the real world that impacts the organization). Furthermore, existing business models and theories influence these requirements. A deep understanding and critical evaluation of those models and theories, which goes beyond the current practice of an organization, can significantly increase the quality of the organizational requirements. However, lack of appreciation of these models and theories means that, most often, these aspects are investigated only cursorily.

Perceived distances between $O_1$ and $S_{1n}$ exist for various reasons. First, the organizational requirements may not have been derived based on all relevant models. Second, inappropriate models may have been included in the organizational requirements. Third, selected models may have been applied incorrectly. Actual distances between $O_1$ and $S_{1n}$ exist if the current status of models and theories does not provide the required solutions for the organization.
System Selection Domain

Organizational requirements (O1) for ES selection typically are summarized in criteria catalogues. These initial criteria are often clustered and weighted. They are also influenced by available models and theories (S1n), which leads to more complete organizational requirements (O2). In this form, they represent the starting point for evaluating a selected number of available systems (S2m) in the system selection domain. The arrow between O2 and S2m describes this relationship.

Typically, a detailed study of the current functionality of available systems (S2m) would impact the organizational requirements (O2). In addition to available models and theories, the functionality of available systems will be another valuable source for the further improvement of organizational requirements. Complete organizational requirements (O2) consider innovative and relevant solutions in available systems.

Again, multiple reasons may lead to actual and perceived distances between O2 and S2m. Actual distances exist when initial organizational requirements cannot be supported by a selected system. The sum of all (weighted) actual distances between O2 and S2 forms one important indicator to be taken into account during the overall evaluation of the appropriateness of ESi. Perceived distances may be more critical in this stage, however. It will often be difficult to analyze the support for all organizational requirements because the functionality of the selected systems might not only be overwhelming, but also difficult to comprehend without detailed systems’ know-how.

An actual distance between O2 and S2m may exist in two situations. First, it arises when at least one organizational requirement is not supported by any of the selected systems (system deficit). Second, it arises when each and every organizational requirement is supported by at least one system, but no one system supports all organizational requirements. Perceived distances can occur at this stage if existing system solutions are not translated correctly into organizational requirements.

The core of the system selection domain is the evaluation of each selected (short-listed) system in light of the complete organizational requirements. Ideally, the system with the most comprehensive support for the complete organizational requirements will be the system selected (S3). In other words, the system with the minimal distance measured as a weighted sum over all distances between the analyzed items will be selected. The actual distance between O2 and S3 is the first indication of potential misfit between the organizational requirements and the best possible ES solution. The perceived distance between O2 and S3 is an indicator of the risk of the decision being suboptimal.

The capabilities of the system selected (S3) represent a constraint on the complete organizational requirements (O2) because the system typically will not be able to support all organizational requirements. This situation leads to a subset of the complete organizational requirements (O2) that we term specific organizational requirements (O3). These specific organizational requirements are derived based on the functionality of the selected system. They represent requirements that can be implemented. An actual distance between S3 and O3 will exist if O3 still includes requirements that can be realized only by system configurations or additional solutions external to S3. In this stage, a danger of unknown actual distances exists. The danger is that the selected system may appear to support core requirements when it does not.

Implementation Domain

The specific organizational requirements (O3) are a major input to the specification of the implemented system (S4). The level of detail in the analyses of distances will be higher in the implementation domain because more criteria will be analyzed. An actual distance between O3 and S4 can result from known or unknown actual distances between O3 and S3. In particular, many unknown actual distances will be converted into known actual distances by the complete, exact analyses demanded when developing implementation requirements.

The identified distances between O3 and S4 will again impact the specific organizational requirements. The final organizational requirements (O4) describe those requirements that we might consider to be the shortcomings of the implemented system. Distances between O4 and S4 can now be characterized as those misfits that remain after the go-live date. Differentiation between actual and perceived distances is important at this stage to make decisions about how these distances should be approached. Unknown actual distances should be minimized as they pose a critical risk. Such distances could, for example, be problems in processes that occur a few months after the go-live date (e.g., during the consolidation process in financial accounting).
Second Wave: Extending the System

The distance between O4 and S4 represents ongoing requirements that will be considered during the second wave (i.e., in extending the system capabilities). This distance could be addressed by continuous benefit realization activities, by system upgrades, or by modifying the organizational requirements. Any remaining distance between O4 and the extended system (S5), as well as organizational changes, will lead to a new set of final organizational requirements (O5: termed revised organizational requirements).

Implication of the Reference Frame for Measuring Ontological Distance

The model of information requirements evolution shows, therefore, that a number of steps or transformations contribute to the misfits that can be identified after an ES has been implemented. Note also that the number of analyzed requirements and system capabilities will increase during the implementation process (i.e., the scope of the analysis will increase). In other words, in Figure 2, circles B and C will become synonymous with circle A.

The value and the innovation of the proposed model are particularly important in a number of areas. Although the model differentiates between different stages of organizational requirements for these systems, in practice we observe that many organizations develop one comprehensive set of requirements, evaluate a number of short-listed systems based on these (weighted) requirements, and then select a system. Our proposed four-stage process of requirements engineering in the context of ES selection, therefore, acknowledges the fact that ES are based on theoretical, system-independent models that can be evaluated separately from the systems themselves.

Discussion and Implications

Our research addresses the alignment between an ES package and an organization’s information requirements as a way of mitigating problems associated with the acquisition, implementation, operation, maintenance, and upgrade of an ES package. Packages are aligned to organizational needs via a tailoring process that includes activities such as configuration, customization, and modification of source code. Alignment problems, which are often characterized as misfits in tailoring an ES to meet users’ needs, frequently become apparent to the user organization only after implementation.

We propose the notion of ontological distance as a way of evaluating differences between the world of organizational requirements and the world of system capabilities. Further, we provide an approach for evaluating the significance (size) of a gap between these two worlds by characterizing such distances from the viewpoint of a selected ontology. The greater the distance between the organizational and system worlds, the more likely organizations are to encounter difficulties in their engagement with an ES package. Further, when evaluating ontological distance, our analysis reveals that it is important to differentiate between actual, perceived, and appropriated distances.

Different types of ontological distance are apparent throughout the requirements determination process. Knowledge of both organizational requirements and systems capabilities evolve over time in a series of stages that largely result from the interaction between the two. Each of the many interactions between the organization and system sides of the model reflects changing distances. The existence of large distances helps stakeholders to pinpoint where more-careful evaluations of organizational requirements and ES capabilities are needed.

From a theoretical perspective, our conceptual development shows how measures of ontological distance can be used to predict the likely deficiencies an organization will have on implementing an ES. Our goals now are to refine our measures of ontological distance and to empirically test the predictions we make about the seriousness of misfits based on the distance measures.

Practically, our research helps organizations select the best package to meet their needs and to avoid problems when developing, implementing, operating, supporting, and maintaining an ES. Our notions can also be used as an aid in structuring the evaluation process and in making the decision to upgrade or further customize the system.

Our goals are to develop an ES package evaluation methodology based on ontological distance and to create software tools to support use of our methodology. For example, such tools might facilitate clustering selection criteria used during an ES selection project and provide default values for each criteria based on the ontological classification. We are also currently conducting an
ontological evaluation of parts of a human resources management solution of an enterprise system. The software tool we will develop to calculate ontological distances will use the outcomes of this evaluation.

References


