Misalignments in ERP Implementation: A Dialectic Perspective

Christina Soh
Siew Kien Sia
Wai Fong Boh
May Tang
Nanyang Business School
Nanyang Technological University

Enterprise resource planning (ERP) systems are often not fully aligned with the implementing organization. It is important to understand their sources of misalignments because they can have significant implications for the organization. From a dialectic perspective, such misalignments are the result of opposing forces that arise from structures embedded in the ERP package and the organization.

An intensive case study was conducted in one organization that has experienced significant misalignments with its ERP implementation. A typology of the misalignments and four pairs of dialectic forces were identified. Articulating deeper structural level misalignments enables organizations to examine their assumptions about the “permanent” characteristics of the organization and helps misalignments surface early to aid planning for resolution strategies.

1. INTRODUCTION

Organizations implementing enterprise resource planning (ERP) packages (or any other large-scale package) will find gaps between organizational requirements and package features. These gaps require the organization to decide whether to customize the package or to change organizational practices to fit the package. ERP vendors and consultants strongly advocate adoption of the package with minimal customization for a variety of reasons (Brehm, Heinzl, & Markus, 2000): to minimize implementation risk, reduce implementation cost, avoid negative impacts on system performance, facilitate adoption of future package upgrades, reduce maintenance costs, and foster adoption of process-oriented “best practices.” Organizational users, however, often demand to have the ERP package customized to meet

Requests for reprints should be sent to Christina Soh, Information Management Research Center, Nanyang Business School, Nanyang Technological University, Nanyang Avenue, Singapore 639798. E-mail: Acsoh@ntu.edu.sg
their operational needs, minimize disruption to established ways of doing things, and meet regulatory requirements and customer needs.

Where the system is not customized to the organization’s structures and processes, users either have to adopt the procedures embedded in the package or work around them (this usually requires additional procedural steps, but leaves current processes and structures fundamentally unchanged). The decisions to adopt, customize, or work around the package are important because they have significant impacts on the organization. Customization provides greater fit with organizational requirements but incurs significant costs (both current project and downstream maintenance and upgrades) and has system performance implications (Markus & Tanis, 2000). Adoption may result in improvements in process efficiency, but may also result in legitimate organization needs not being adequately met. Workarounds usually have negative impacts on productivity and organizational controls and undermine potential benefits from integration.

The misalignments can be severe. AeroGroup, for example, dropped the Apparel Footwear Solution of SAP/R3 halfway through implementation because of its inability to model the uniqueness and complexities of the footwear business (Deutsch, 1998). Similarly, Fox Meyer’s presumption that SAP could be easily tuned to suit its wholesale distribution business proved disastrous (Bulkeley, 1996). Thus, it is important to understand why these alignments arise early in the decision-making process. Although the existing ERP literature has highlighted the issue of misalignments, the blanket assertion that they arise from unmet organizational requirements masks the variety of sources of misalignment. Prior studies investigating the nature of the ERP misalignments are scarce. One exception is the misalignment categorization suggested by Soh, Sia, and Tay-Yap (2000). However, the schema adopts a software application perspective, classifying misfits superficially into data, process, and output categories. The framework does not surface the underlying rationale that gave rise to these manifested misalignments. We therefore seek to explode the “black box” to investigate the sources of misalignments. Understanding the sources of misalignments will enable ERP implementers to surface misalignments earlier. The identification of potential misalignments also allows related change management issues to be adequately planned for. Resolution strategies are less likely to be reactive.

The theoretical perspectives that we found most helpful in examining the issue of misalignments and their sources were from the field of organizational change. In particular, the dialectic perspective of organizational change (Van de Ven & Poole, 1995) provided a way to conceptualize the underlying forces that give rise to misalignments. These opposing forces (Robey & Boudreau, 1999) exist because of the dualistic nature of technology (Orlikowski, 1992). As a product of human action, it embeds developer intentions and assumptions. On the other hand, once a technology artifact (e.g., an ERP package) is created, it also has the potential to constrain and enable further human action.

In this article, we provide a simple typology of ERP misalignments, which we develop groundedly from our data. We also draw on theories of organizational change to develop a framework for understanding the sources of misalignments by identifying the opposing forces that are likely to arise in ERP implementation. We
use rich data from an in-depth study of an ERP implementation in a hospital to provide a concrete classification of misalignments and to provide an explanation of how misalignments arise from the opposing forces.

2. A DIALECTIC APPROACH TO MISALIGNMENTS IN ERP

In technology implementation, misalignments occur when features of the technology do not fulfill the requirements of the organization. Robey and Boudreau (1999) suggested that it is fruitful to take a dialectic perspective of information technology (IT) implementations, because explicit recognition of the opposing forces underlying the implementation often leads to creative ways to manage the tensions. Although all four motors of change—teleological, life cycle, evolutionary, and dialectic (Van de Ven & Poole, 1995)—are likely to be present in any situation of organizational change (e.g., the implementation of a major system), the dialectic perspective, in particular, captures the essence of misalignments. Intuitively, we sense that misalignments arise from the conflict between opposing forces. What are these opposing forces that can give rise to misalignments? One set of forces arises from the technology itself.

If we view technology as a material artifact (Orlikowski, 1992)—in this case, a software package with its embedded features and functionality—then it is the product of human action, reflecting developers’ assumptions about business rules, norms, and values. These assumptions, norms, and rules (or structures) are built into the technology. These structures embedded in the technology have the potential to shape the organization in various ways, as demonstrated in studies by Barley (1986), DeSanctis and Poole (1994), and Orlikowski (1996). For example, Orlikowski (1996) showed organizational users who appropriated the new Lotus Notes technology into their work practices, resulting in changes to the nature of work, workload, patterns of interaction and coordination, performance evaluation, and accountability. Clearly, the technology exerts forces that can influence the organization in a variety of ways. The impacts are not deterministic, as much depends on decisions made during implementation, as well as appropriations made subsequently in use.

Most of the studies that recognize the structures embedded in technology focus on the use phase (Volkoff, 1999, is a notable exception). However, these forces surface and begin to impinge on the consciousness of organizational participants as early as the implementation phase. Organizational participants in the technology implementation project begin to understand the potential implications for how work will be done for organizational structure, controls, and decision making. As a result, misalignments are identified during implementation and important decisions are made regarding the misalignments that affect how the technology will impact the organization when it is subsequently deployed.

Just as one set of forces arise from structures (reflecting developers’ assumptions, norms, and values) embedded in the technology, another set of forces arise from structures in the implementing organization. These structures reflect the assumptions, norms, and values of the organization’s members.
when the organizational structures are in opposition to the structures embedded in the technology. Therefore, when an organization decides to implement an ERP package, it is necessary for it to first understand the structures that are built into the package.

2.1. Structures Embedded in ERPs

Structures are the essential set of shared rules or norms that have been institutionalized over time in a specific context. These values and beliefs are often the product of historical development, interaction between the key stakeholders/social actors, and the competitive environment. Within this socially constructed structure, the legitimacy of an action (i.e., its desirability and appropriateness) is determined (Suchman, 1995). Applying it to an ERP context, these embedded structures can be interpreted as the “spirit” of the technology (Majchrzak, Rice, Malhotra, King, & Ba, 2000); that is, the general intent of the ERP with regard to the developers’ assumptions, norms, and values. Our review of the ERP literature suggests that developers’ have assumptions, norms, and values about integration, process orientation, flexibility, and domain specificity.

One of the key selling points of ERPs is that they offer integration across the enterprise (Davenport, 1998). This is achieved through an “underlying integrated database that stores master and transactional data in a consistent way and with controlled redundancy” (Klaus, Rosemann, & Gable, 2000, p. 143). The main ERP package, SAP, had its origins in materials resource planning (MRP) systems that were highly data driven. As these manufacturing systems grew to encompass more functions, the use of an integrated database became one of the norms of the increasingly multimodule packages.

The structures that support integration have significant implications for the implementing organization’s job scope, organizational controls, decision making, and organizational structure. For example, norms of integration in ERP require data capture at source and subsequent sharing of data among departments. Issues of data ownership and responsibility for data capture arise during implementation. Often, the workload of front-line operational staff increase and downstream users becomes more dependent on front-line staff for accurate entry of data. Integration also seeks to make more information available to more people in a more timely fashion. This has implications for management control, as well as the ability of peers to view others’ work (Sia, Tang, Soh, & Boh, 2000).

ERPs also offer implementing organizations the promise of “best practices.” Enterprise system vendors have designed “best practices” by consulting with customers (Connolly, 1999) and drawing on the work of academics (Markus & Tanis, 2000). Many of these best practices reflect principles from the business process re-engineering (BPR) movement. BPR has been practiced by most major organizations sometime in the past 15 years. Although it has often been oversold and its detractors rightly point out its many shortcomings, it has made an impact on the values of practitioners and academics with regard to the characteristics of a “good” business process. The norms of efficient, cross-functional processes are widely ac-
accepted prescriptions today and have been embedded in ERP systems (Markus & Tanis, 2000).

These features have strong implications for the implementing organization’s structures in areas such as nature of jobs, workload, and interdependence among departments, management controls, and management information. For example, efficient, cross-functional processes seek to increase organizational efficiency and turnaround time by eliminating or combining steps in the workflow (Connolly, 1999). The result is usually a lower number of direct checks on every transaction. This has implications for the implementing organization’s control orientation. In addition, for employees, these ERP embedded norms usually require at least an increase in knowledge about other departments to support cross-functional processes, and usually also an increase in work scope.

ERPs are packages that are designed to be adopted by many different organizations. As such, they need to have a degree of flexibility to meet the diverse requirements of multiple organizations. In ERPs, flexibility is built into the system through features such as multiple screen paths and a multitude of configuration options (Klaus et al., 2000). The price of flexibility, however, is complexity. For example, though icon symbols and menus provide users with flexibility in entering data (Mallett & Newson, 1998), users find that they have to go through many screens to complete a transaction. The chances that unfamiliar users will take the wrong path are also higher because there are fewer restrictions on screen flow. The complexity that results from the configurability of the package leads many organizations to depend heavily on consultants during implementation.

Finally, the procedures embedded in ERPs encode the rules, regulations, and assumptions of the countries/regions, sectors (e.g., manufacturing vs. service, public vs. private), and organizations that the package developers primarily interact with. For example, it has been noted that norms related to western names are different from those for Asian names and that Asian users have problems with apparently simple input fields such as first name and last name (Soh et al., 2000). Another well-known example is the Fox Meyer case where procedures embedded in the system were developed from a manufacturing environment and did not fit a wholesale distribution business, contributing to the overall failure of the system implementation (Bulkeley, 1996). Such domain-specific assumptions can lead to severe misalignments when the implementing organization is in a different region and sector. The organization will be subject to the regulatory and business practice norms of its environment. The organization itself will also have added its own norms over time.

In summary, the prior literature on ERPs suggests an embedded structure that favors integration, process-orientation, flexibility, and the specificity of the originating domain. However, the aspects of the implementing organization’s structures that are most likely to be opposed to the embedded ERP structures are less well understood. Figure 1 presents our conceptual view of the problem of misalignments. We seek to clarify and populate the right side of the diagram; that is, the structural elements in the organizations that interact with the embedded ERP structures, as well as the types of misalignments that typically arise from the interaction of the opposing forces.
3. METHODOLOGY

The objective of this study was to understand the types of misalignments and the underlying forces. This required that we understand the structures of the implementing organization and how they interacted with the structures embedded in the ERP package. An intensive case study approach was appropriate as it enabled us to gather contextual information on organizational structures—some of which were specific to the organization and others that arose from the organization’s external environment. A variety of data sources—interviews, documents, and seminars—was sought to increase the reliability of the results.

**FIGURE 1** Conceptual view of misalignments in ERP implementations.
3.1. Site Selection

We chose a large hospital that was experiencing a significant number of misalignments in their implementation of an ERP package, as this allowed us to collect sufficient data points to test the validity and completeness of the proposed misalignment schema. The hospital was also implementing several modules, thus allowing us to examine the misalignments arising from the package structures that supported integration and efficient, cross-functional processes. Finally, the hospital’s Asian context also allowed us to examine issues related to domain specificity.

3.2. Site Description

The hospital studied is one of seven government-funded hospitals in Singapore. These hospitals together account for over 80% of hospital beds in the country. These government hospitals have been increasingly “privatized” since the 1980s with the aim of providing more effective and efficient healthcare through increasing accountability and flexibility to respond to trends in patient needs. Although a significant proportion of the hospitals’ budget comes from government subsidies for patients treated, over the years hospitals have put in place a layer of professional healthcare management and independent boards of governors and have acquired considerable independence in the provision of services.

The hospital we studied has over 800 beds (the other hospitals have between 800–1,500 beds) and provides specialist healthcare services. It is over 100 years old and has about 2,000 staff, many of whom have been with the hospital for many years. The hospital is functionally organized—the main functional units being nursing and operations, followed by medical record, human resources, and corporate development. Within the nursing area, there are further functional subdivisions—wards for various medical specialties, operating theatre, and nursing administration. Operations are further divided into support services (laboratory, pharmacy), financial services, and so on.

The hospital’s IT history is tied to that of the other hospitals because prior to the ERP implementation, they shared a set of core systems that ran on a central mainframe. In the mid-1990s, these public hospitals identified the need to replace the mainframe-based financial, administrative, and patient management systems, chiefly to resolve the year 2000 problem. Each hospital carried out its own independent assessment, but was eventually influenced by issues of risk minimization to opt for the same ERP package.

Prior to the adoption of the ERP system, the hospital used one customized mainframe package for patient management and another package for accounting. It also had a mini-computer package for materials management and a PC-based package for human resource management. Presentations to the board noted three reasons for adopting the ERP package. The main motivation was the year 2000 problem, but it was also noted that the mainframe systems were old and costly to maintain, and finally, that there were benefits from having integrated functional modules.
Eventually, the hospital adopted the financial, materials management, and inpatient management modules of the ERP package.

The implementation began in early 1998 and lasted for 1.5 years. Implementation consultants were hired, as there was little in-house expertise with the package. Project teams were constituted for each module. Each module comprised one or two consultants, one or two information systems department (ISD) personnel, and several user representatives. User representatives included the department manager as well as operational users. There was also an oversight team comprising the senior consultant (account manager), a senior operations manager, and the ISD manager. The implementation process followed the methodology recommended by the consultant organization and was divided into the following phases: as-is (requirements analysis), to-be (design), configuration, data conversion, testing, training, and rollout.

### 3.3. Data Collection

We began our study about two thirds of the way through the hospitals’ ERP implementation. Two modules (finance and materials management) had been implemented and another (patient management) was a few months away from cut-over. To provide us with an understanding of the wider institutional context, we attended a day-long seminar organized by the Health ministry on the use of technology in healthcare. We also read the papers presented to the hospitals’ Board that sought approval for the project funding and the minutes and presentation slides for the project kickoff meeting. Finally, we interviewed the senior operations manager and information systems (IS) manager for an overview of the conditions prior to initiation of the project.

To identify the misalignments that arose in the ERP implementation, we used multiple sources of data—logs of outstanding issues, request for change forms, meeting minutes, and interviews. Each data source had its strengths and weaknesses. The most “objective” sources of data were the logs of outstanding issues and requests for customization forms (RFCs). These provided concrete evidence of instances where the package capabilities were perceived by project team members not to meet the organization’s requirements. However, these two sources omit certain misalignments—those that users may not have pushed harder for because of power issues or time and resource constraints, and those where project team members had agreed were to be resolved through workarounds or through adopting the package features. To provide a more comprehensive source of misalignments, we examined the meeting minutes, particularly of the “as-is” (requirements analysis) and “to-be” (design) phases. These provided documentation of discussions about why the misalignments arose, suggestions, and counter-suggestions for their resolution. Finally, interviews provided a more in-depth source of data on users’ concerns, institutional norms, and values.

In all, we read through the minutes of over 50 meetings, documenting discussions between user representatives, ISD, and implementation consultants.
throughout the 1.5-year span of the implementation. These project meetings were held, on the average, once every 1 to 2 weeks and the minutes were routinely prepared by a project team member and distributed to the other meeting participants. We also conducted 30 interviews with implementation participants—including 20 with user representatives on the project teams, 6 with ISD staff, and 4 with implementation consultants. Each interview lasted about 1 hr, and at least two research team members were present. All interviews were transcribed and the transcripts were subsequently read and cross-checked by a second research team member. The interview questions were anchored in the phases of the implementation life cycle—requirements analysis, design, configuration, testing, training, and cut-over. For each phase, the interviewees were asked whether there were any major issues that surfaced and how these were handled. Interviewees provided a great deal of contextual information to explain what the issues were and why they arose.

3.4. Data Analysis

As our research team began examining the data gathered, we realized that the analysis would have to be iterative, with each cycle of analysis probing deeper into the data. We moved from a simple “surface” identification of the misalignments toward uncovering the forces underlying the misalignments, through successive iterations aimed at peeling back the layers of meaning. We identified all instances of misalignments by reading through the minutes, interviews, issue logs, and RFCs. A misalignment was defined as any instance where project team members (users, ISD, or consultants) identified an organizational requirement that they felt was not being addressed by the package. The output from this first phase was a list of misalignments typically surrounding issues like workflow changes, complexity of data entry, inappropriate reports, and inadequate revenue computation functionality. Each misalignment was supported by a document that brought together all statements from various data sources about that misalignment. For example, there may be statements about a particular misalignment in meeting minutes, an interview, or in the issue log and request for change form.

After piecing together all information regarding a misalignment, we then attempted to understand the underlying rationale that led to the problems. The process of probing led us from superficial reasons to deeper, structural reasons. Sometimes, that required us to consciously ask a series of “why” questions (e.g., Why do the users need such a feature? Why isn’t the feature included in the ERP system?). Where necessary, follow-up discussions were arranged with the users and consultants for clarifications. For example, the lack of counter-collection functionality was eventually traced to the western revenue collection practices that tend to go through the institutional insurance mechanisms, in contrast to the predominant individual payments in this region. As we understood the context of each misalignment, we then categorized them by the underlying dialectic forces at play.
4. RESULTS

The analysis of the misfits revealed several recurring types of misalignments that were consistently cited by interviewees, regardless of their functional area (nursing, admissions, materials management, finance, IS), and by the implementation consultants. The misalignments clustered around the following themes: problems with workflow changes, complexity of data entry, lack of useful reports, and inappropriate revenue computation and collection procedures. As we probed for the sources of each misalignment, we surfaced the underlying structures in the hospital that were in opposition to those embedded in the ERP package. Table 1 summarizes the main pattern of our findings. In the following sections, we present the dialectic forces and illustrate each pair of forces with representative misalignments. We also briefly present the resolution decision made (adopt, customize, workaround) and the impacts experienced by the organization.

4.1. Integration—Differentiation

Integration—differentiation conflicts arise from the tension between the embedded ERP structure that promotes integration and the localized need for differentiation. Manifestations of these misalignments often come in the form of muddled data ownership and tighter workflow interdependency. For example, ERP packages require that departments share a common standardized database. These integration forces were in opposition to other forces in the hospital that arose from each departmental area previously having its own systems. These legacy systems had evolved over time, and each had become highly customized to the needs of a department.

One such misalignment involved the areas of materials management and finance. As explained by the implementation consultant:

Previously, Materials Management [MM] and Finance each maintained their own vendor master, and they may not have been consistent. There were some discussions on who should be the owner. We get the two parties together to fight it out. The implication is that the owner will get to make the changes to the vendor master. MM should logically be the owner because they are the first to create the vendor, but there is also some finance data.

The misalignment revolved around issues of common data structure, data ownership, responsibility for data entry, and related changes in workflow. Previously, when each department had its own differentiated systems, there was much less need to harmonize data requirements and each department had their own procedures for creating and maintaining departmental data.

In the aforementioned situation, the implementation consultants were able to offer a workaround solution within the system that maintained the benefits of sharing a common database while accommodating the functional orientation of users. Some changes in workflow, however, were still necessary.
<table>
<thead>
<tr>
<th>Opposing Forces</th>
<th>Example</th>
<th>Misalignments</th>
<th>Impact</th>
</tr>
</thead>
<tbody>
<tr>
<td>Integration</td>
<td>Differentiation</td>
<td>Materials management group and finance now have to share a common database.</td>
<td>Data ownership workflow</td>
</tr>
<tr>
<td>Process orientation</td>
<td>Functional orientation</td>
<td>Nurses expanded job scope to include capturing patient data at transfers.</td>
<td>Workflow job scope</td>
</tr>
<tr>
<td>Flexibility</td>
<td>Restrictiveness</td>
<td>Some fields to be made mandatory for input.</td>
<td>Data entry</td>
</tr>
<tr>
<td>Package domain specifics</td>
<td>Organization domain specifics</td>
<td>Revenue computation and collection.</td>
<td>Processing/billing</td>
</tr>
</tbody>
</table>
We divided the vendor master into two views—accounting and purchasing. MM will maintain the purchasing view and initiates the process for finance to update the financial view. Finance will have to update the vendor master for the invoice to be processed. Now MM is the main owner. We create a process to force the accounting people to enter the data, because if not, they cannot do their posting.

Sometimes, the attempts to accommodate the integration–differentiation conflicts have added more complexity. For example, the creation of universal codes that can also meet the respective needs of individual departments and physicians has generated a long list of charging codes, complicating data entry and report formatting.

4.2. Process Orientation—Functional Orientation

Another source of misalignment springs from the opposing process—functional orientation. The structures within the ERP package tend to support a process view, in contrast to the functional setup of the hospital. Among other things, the ERP package required data about a transaction be captured as it is moved through the organization. This meant that data capture could not be relegated to back-office, administrative staff, but often had to be performed by front-line, operational staff. The hospital had (and continues to have) a traditional, functional structure with clear demarcation between medical and administrative staff and with further functional specialization within these two broad functions. For example, medical staff are organized into functional areas such as operating theater, wards, emergency units, and so forth.

A process orientation requires changes in workflow, as handling of transactions is no longer limited by functional boundaries. The system and employees now see a transaction through from start to finish. Where the organization has a strong functional orientation, however, opposing forces arise that resist the expanded work scope required by the process orientation.

An example of workflow changes brought about by the process orientation occurred in the wards. It was noted during the implementation phase that ward nurses would now have to key data into the system (e.g., when a patient is transferred between wards). Nurses’ job scope would be expanded to include capturing information about patient location, attending physician, and treatment department because they were the “person on the spot” at certain important points of the transaction process (in this case, patient movement). Although these changes resulted in more current and detailed process information on patients, as well as more efficient workflow, project team members were concerned that the inclusion of this new task in ward nurses’ duties would give rise to certain problems. Nurses were unfamiliar with data entry and found it difficult to remember the many physician and department codes. The organizational norms were that nurses were to attend to patients and interacting with systems was not previously part of their job.
Nurses felt that the new system-related tasks detracted from the time they should have spent in direct patient contact.

In this case, despite the system-imposed process orientation, the hospital did little to change the functional focus in nurses’ job (e.g., handling tasks from end-to-end along some medical specialties). As a result, the manifested issues are complexity, unfamiliarity, and the constant feeling of compromised patient care. The repercussion on the hospital is large. First, others downstream, such as finance, were dependent on the accuracy of the nurses’ input.

If they don’t key in the correct OU [operating unit], it will be a major problem for finance. Now, nurses play a greater role, but nurses are not used to doing much administration work. If nurses don’t key in the correct OU, the impact goes to different cost centers.

Errors that occurred not only had financial impacts, but also impact operations.

There was a case this morning where admissions admitted a patient thinking that there was an empty bed. But in actual fact, the patient had been transferred elsewhere then back, but the nurses did not record the transfer of the patient back to the bed, and the admissions person thought the bed was empty. There was thus the problem of double-decking.

The hospital responded to these impacts with more training for nurses, as well as devoting more resources to checking transactions both at the wards and in the backroom departments such as the business office that deals with billing.

Initially, we tried to minimize errors by getting the clerks to check the entire case keyed into the system by the nurses. We would then point out the mistake to the nurses, who were not very happy. But we have to highlight the mistake to her, because it impacts on other divisions. There is always the problem of lateral transfer (between wards), but it is important to capture the transfers of the patients because it affects revenue charging.

4.3. **Flexibility—Restrictiveness**

In attempting to meet the diverse needs of many organizations, ERP packages have built in a high degree of flexibility. For example, flexibility is promoted through offering users a large number of screen options and many possible paths in navigating among the screens. The underlying assumptions are that users are sophisticated and the variety of tasks performed requires flexibility. The hospital, however, had a relatively low level of IT literacy among users. Moreover, the nature of tasks is simple and routine. Administrative users had previously interacted with systems that were highly customized to their specialized requirements, and that had simple and restrictive interfaces. Medical staff hardly interacted with the systems
at all. Users therefore perceived the ERP package’s flexibility features as contributing to complexity.

Project team members voiced concerns about the likelihood of incomplete and inaccurate data input (both in the project meeting minutes and in interviews). Many of these concerns surfaced as requests for certain fields were made compulsory; that is, project team members wanted to make the system more restrictive so that users could not leave the current input screen unless certain data fields were completed. For example, in the situation where clerks in the admissions area were capturing patient data:

We wanted to make the number of living children a mandatory field, as the figure is needed at the back end calculation as well as medisave claims [a government-supported medical saving scheme].

In some cases, the requests for more restrictive input screens led to customization. However, in other cases, due to time and resource constraints, no customization was made. Users were left to deal with the problems arising from incomplete or inaccurate data entry. Some of the impacts were seen in the examples provided by exasperated frontline supervisors:

For example, the guarantor field (person guaranteeing payment of patient’s bills where patient is a minor or unemployed) is not compulsory, so sometimes, the [check] refunds are made to a baby 6 months old!

The same pair of opposing forces are again the source of many of the misalignments concerning reports. The ERP package, seeking to meet the needs of a wide variety of organizations, provides a large number of reports and provides some tools within the package itself for the implementers and users to customize the report. However, this hospital’s relatively low user IT literacy means that the report customization tools were almost never used by users. Users did not value having a large number of generic reports to choose from; instead, they desired a much smaller set of reports tailored to their specific needs.

Dissatisfaction with reports appeared in a large number of meeting minutes and was voiced by almost every single project team member interviewed. The misalignments were usually about inappropriate format or difficulty in getting all the desired pieces of information in one report.

The key problem with the [ERP] package: report deficiency. Initially, we did not match our reports to the package reports. [We] thought that with so many reports, over 1,000 provided by the package, the requirements would surely be met. Most [package] reports have no header, no page numbers, no dates, etc. (IS personnel)

Main problem is the reports. Users are used to the reports they have been traditionally receiving. What they received in one report in the past, they now need to run three or four reports to get the information. (Consultant)
For example, we used to show the monthly profit and loss (P&L), from the beginning of the year to the current month, in side-by-side columns. [The package] only provides the previous and current month, and year-to-date P&L. If you want monthly P&L, you have to cut and paste. The information is available, but not in the required format. (Accountant)

The small and over-burdened IT department were also not able to provide much support for report customization. The consultants’ favored adoption of the standard reports:

Report development is not included in the contract because we are aware that most clients will not like the reports in [the package]. We tend to exclude report development because it is like a black hole, we don’t know how many reports they require, could be 100–200! (Consultant)

In most cases, the users “lived with it.” However, about 40 reports were identified that were required routinely because of reporting requirements to various regulatory authorities such as the Ministry of Health. For these, a separate contract was subsequently struck with the vendor to write a program that would extract the necessary data at specified intervals and to generate the required reports.

4.4. Domain-Specific Forces

The last set of dialectic forces springs from the processing rules embedded in ERP that reflect the environment of the countries and sectors that the developers based their reference models on. Country-specific factors include national culture, the regulatory environment, level of national wealth, the degree of government involvement in the economy, and the level of education. Sector-specific factors would include revenue generation (different for public vs. private sector), and cost allocation metrics (different for manufacturing vs. service sectors).

In this study, country-specific structures in terms of government involvement in healthcare and sector-specific structures in terms of mechanisms for computing and collecting revenue were found to be in opposition to the assumptions embedded in the ERP package.

A major source of misalignments was the difference in the basis of computing revenue. The local hospitals’ institutional structures differed from that embedded in the ERP package in two key respects: the concept of bed-class and significant revenue collection from patients and government. All local hospitals have three major bed-classes: “A” class wards (single bed per room), “B” class wards (two to four beds per room), and “C” class wards (six or more beds per room). The bed-class concept affected revenue computation and collection as the amount charged to the patient per day of hospital stay, as well as for use of facilities (such as the operating theatre), for medication, and for professional services, differ depending on the bed-class. In addition, the amount claimed from the government was greater for “B” and “C” class patients, as the subsidies for these classes are higher. In many
western countries, there is no bed-class concept. Most of the revenue computation/allocation problems surfaced as requests for functionality required to compute patient bills and to compute claims from the government subvention fund.

This group of misalignments were resolved by having the implementation consultants develop a billing module that interfaced with the ERP system via a user exit. This add-on module incorporated the hospitals many and complex rules for revenue computation.

In addition to subvention from the government for “B” and “C” class wards, a portion of the costs are also collected directly from individuals. The amounts are significant for “A” class wards, which receive very little government subsidy. In the United States and western Europe (referent countries providing the embedded institutional structures), hospital revenue is largely collected from insurance companies. In the Singapore context (as is the case in Asia and other countries where health insurance is less well established), a significant portion of the costs is collected directly from the individual as well. For day surgery and other minor procedures, for example, it is common for the patient to simply pay over the counter. The ERP module had only rudimentary counter-collection capability. For example, it could not handle part-payment satisfactorily.

We wanted to have a different installment module. (The package) is able to break the bill amount into, say, 10 installments, but treats it as 10 bills. This is clumsy in terms of monitoring. If the clerk recording the payment chooses the wrong bill, a bill can remain outstanding forever. Moreover [the package] could not handle interest computations [for late payment].

In the case of this misalignment, the decision was to develop a counter-collection module and link it to the ERP system via a user exit.

5. DISCUSSION

The findings thus show that misalignments that emerge in ERP implementation can be traced to a few fundamental incompatibilities between the embedded structures of ERP and the implementing organization; namely, the tensions between the forces of integration and differentiation, process-orientation and functional specialization, flexibility and restrictiveness, and packaged versus organizational domain specificity (Figure 2). The dialectic perspective thus suggests a preliminary typology of misalignment based on the underlying opposing forces. With an appreciation of the embedded structures of ERP, one can examine them against corresponding structural forces in the implementing organizations to surface the potential misalignments. As seen in the case of the hospital, the mapping efforts require conscious surfacing of the specificity about the organizational structure, the nature of tasks, the assumptions on user competency, and its business model of funding and resource allocation.

Recognizing the tensions between the underlying structures helps us conceptualize the ways to deal with misalignments. At one extreme, an organization can
choose to restructure the embedded forces within ERP to forge alignment (e.g., by heavy customization of ERP features like setting up functional identifiers and restricting transaction paths). However, the organization’s inability to control the evolution of the structures of ERP packages in future versions renders this option transient, inefficient, and costly. On the other hand, an organization can choose to change its own structures to match those of ERPs (integrative, process-oriented, etc.). The decentralized, functional setup of many organizations, however, demands significant organizational transformation.

The hospital focus was largely on the implementation of the package, and less on the redesign of organizational processes and structures. This approach to technology implementation has been noted in many other studies (Venkatraman, 1998),
and has been shown to produce a lower level of organizational benefits than when organizational transformation is explicitly sought. Organizations too often assume that the technology itself will be the “magic bullet” (Markus & Benjamin, 1998) that delivers organizational change, and therefore do not engage in the change management behaviors needed to bring about the transformation.

Even with a transformational intent and aggressive change management, most organizations are unlikely to be able to fully align the structural forces at play between the ERP and the organization. The dialectic view of misalignments sensitizes us to the trade-offs that must be made when dealing with the misalignments. The hospital we examined, for example, has chosen to retain its functional structure (e.g., the centralization of wards to be shared among all medical specialties), despite the process-oriented forces of ERP (e.g., end-to-end processing by medical specialty). Consequently, additional efforts have to be incurred to repackage information that the system collects and processes by medical specialty for reporting to individual wards. The trade-off issue is illustrated by the prioritization of functional specialization over the process-orientation. Similarly, the hospital’s reluctance to expand the job scope along more process-oriented lines (except in limited cases) causes the enhanced flexibility of the ERP to be perceived as “cumbersome” or “silly” by hospital staff. The greater screen maneuverability and input variety has “added extra steps to the process” and introduced more data entry errors.

Thus, the dialectic perspective helps us to assess the trade-offs inherent in solutions to misalignments. Having full integration will tend to compromise differentiation issues (e.g., restricted access) simultaneously. Similarly, flexibility in screen navigation will inevitably add to input complexity. The issue management has to deal with is to decide on the prioritization of these conflicting structural forces at play and once a choice is made, factor the trade-off issues into change management. In general, our observation is that domain-specific misalignments often necessitate customization or the institution of significant workarounds whereas the other three dialectic conflicts tend to result in adoption or workaround resolutions.

The dialectic perspectives also remind us that these embedded structural forces will continue to operate in the future. Over the long run, an organization may aim to influence the development of its own structures to converge with those embedded within ERPs. As it does so, it may reap greater benefits from integration and process efficiencies. It is perhaps harder to shape the assumptions, norms, and values among the ERP developers. Organizations may attempt to influence ERP vendors by banding together. This was seen in this case where the various restructured, public sector Singapore hospitals built some bargaining strength by negotiating for certain “local” features with the vendor. Organizations should also keep up-to-date with the development in the ERP industry and monitor the extent of convergence and divergence between the two sets of structural forces.

6. CONCLUSION

In conclusion, by framing misalignments as resulting from opposing structural forces embedded in ERP packages and the implementing organization, we have
helped to unpack the dimensions on which these forces interact. Moving beyond the surface issues like cumbersome workflow, complex data entry, inappropriate reports, and missing or inadequate functionality and articulating these misalignments at the deeper structural level allows us to examine our fundamental assumptions about the more “permanent” characteristics in our organizations. The imposition of the structural forces embedded in an ERP package on the organizational characteristics generates the inherent tensions between integration and differentiation, flexibility and restrictiveness, process-orientation and functional specialization, and conflicts between package—organizational domain specificity. Although this dialectic conceptualization of ERP misalignments should be further refined and validated across multiple sites and different modules, the dialectic forces we have identified serve as a preliminary typology on which critical misalignments can be surfaced early in the implementation process. This will enable related change management issues to be adequately planned for.

REFERENCES


