Leveraging IT in Slack Resources Redeployment: A Development towards Resource Fluidity


Abstract

Slack resources have important strategic and managerial implications in an organization and IT plays an important role in redeploying them. Yet, this area has largely been overlooked by the IS community. This study, through a novel theoretical lens of resource fluidity, aims to answer how organizations leverage IT to redeploy slack resources. We conduct a case study at two leading manufacturers that have just gone through a strategic transformation, and interviewed 78 informants from them. Analysis of the data unveils two process models of fluidity development. The first process, which is guided by a centralization strategy and led by IT, features a small-win cycle. The second process, which is guided by a decentralization strategy and led by business, features a big-win cycle. The study contributes to the IS literature by bringing in the slack-resource perspective and unveiling how IT redeploy slack resources by enabling resource fluidity.

Keywords: Slack Resources, Resource Fluidity, IT-enabled Transformation, Case Study

Acknowledgement:

This research is supported by the National Natural Science Foundation of China (71402187)
**Introduction**

Excess resources available to an organization in a specific period of time are known as slack resources (Nohria and Gulati 1996; Sharfman et al. 1988). Slack resources are an important managerial and strategic topic, because they carry substantive implications on organizational performance (Love and Nohria 2005; Tan and Peng 2003). These implications differ based on the different forms of slack resources. Unabsorbed slack resources, such as excess cash and credit line, which are not committed to existing operations and can be readily redeployed (Sharfman et al. 1988), are positively associated with organizational performance, because they can act as buffer in managing changes in a turbulent environment (Cheng and Kesner 1997; Tan and Peng 2003) and act as an inducement to experiment, take risks, and make proactive strategic choices (Mishina et al. 2004). Absorbed slack resources such as excess inventory and production capacity, which are committed to existing operations and cannot be readily redeployed (Sharfman et al. 1988), are negatively associated with organizational performance, because they breed inefficiency (Tan and Peng 2003), increase coordination overheads (Rai et al. 2006) and reduce organizational flexibility (Mishina et al. 2004).

Organizations with high-level absorbed slack are dubbed fat. These organizations have been found to be slow and less competitive in the market (Love and Nohria 2005). As the competition grows intense, it becomes imperative for organizations to redeploy absorbed slack for productive use (Love and Nohria 2005; Mellahi and Wilkinson 2010). This imperative is prevalent in both large organizations that are adequately capitalized (Cheng and Kesner 1997) and small organizations that are undercapitalized (George 2005), and prevalent among organizations in both developed economies that enjoy a munificent economic foundation (Greenley and Oktemgil 1998) and developing economies that suffer from a sparse economic foundation (Tan and Peng 2003). However, despite its importance and prevalence, the question of how to redeploy slack resources for productive use remains unknown in the extant literature (Love and Nohria 2005, p.1088).

An important reason behind the emergence of and the accumulation of absorbed slack is that resources do not flow well; for example, when excess resources are separated by fragmented operations or disconnected routines, they become absorbed slack (Sharfman et al. 1988; Voss et al. 2008). Therefore, the key to redeploy absorbed slack is to create resource flows so that slack resources can move from one place to another and combine to yield better outputs. Information system (IS) has been known for its strengths in coordinating resources across boundaries (Dong et al. 2009; Rai et al. 2006) and recombining resources for better outputs (Kumar and Van Dissel 1996; Pavlou and El Sawy 2006). For example, in a manufacturer’s ERP initiative, a central material management is used to integrate material procurement across multiple production lines; this integration allows raw material to be shared across multiple product lines and prevents them from being in short supply at some production lines and redundant in others (Banker et al. 2006; Zhu 2004). Yet, sometimes IS tools, especially those legacy systems, are known for being the structure that locks resources in obsolete operations (e.g. Martinsons and Martinsons 2002) and prevents cross-border resource sharing (Pan et al. 2012). For example, in an IT outsourcing vendor’s knowledge management initiative, the knowledge management systems (KMS) hosted by the individual subsidiaries become a major obstacle for knowledge circulation across subsidiaries and as a result, much valuable knowledge is locked within the walls of the subsidiaries (Mayasandra et al. 2010).

Despite its both positive and negative implications towards deploying absorbed slack, few IS studies have looked at IS development and management from this perspective. This in fact contributes to the enduring criticism on IT department that the department consumes many resources but brings little material contribution, that IT investment is a technical imperative rather than a business one, and that CIO is a technical person who is not business oriented (Earl and Feeny 1994). In summary, IS scholars need a framework to explore how to redeploy absorbed slack using IS and practitioners need a framework to stimulate relevant discussions toward slack redeployment.
Theoretical Development

Resource Fluidity Development towards Slack Resources Redeployment

While early research struggles with mixed empirical evidences that suggest both positive and negative effects on organization (e.g. Cheng and Kesner 1997; Greenley and Oktemgil 1998), recent studies have reached a consensus that the effects are contingent on the types of slack resources (e.g. George 2005; Voss et al. 2008). Because of its availability and flexibility, unabsorbed slack improves organizations’ financial performance (Tan and Peng 2003), environment response (George 2005), competitive position (Love and Nohria 2005); as a buffer that protects organizations from resource shortage, unabsorbed slack stimulates innovations and encourages exploration (Mishina et al. 2004; Voss et al. 2008); but over a certain level, unabsorbed slack starts to bring in negative implications such as laziness (Greenley and Oktemgil 1998) and escalation of commitment (Nohria and Gulati 1996; Singh 1986). Absorbed slack, because of its immobility and rigidity, complicates operations and reduces efficiency (Greenley and Oktemgil 1998); as a major cost center, absorbed slack also makes an organization risk aversion and thus drives down its innovation (Voss et al. 2008). Yet, how to redeploy slack resources remain unknown, especially the redeployment of absorbed slack, which is costly to maintain and detrimental to organizational performance at all levels (Love and Nohria 2005). Some scholars advocate downsizing (e.g. Love and Nohria 2005; Mellahi and Wilkinson 2010). That is an effective approach but not a fundamental resolution, because it does not address the root cause of absorbed slack. The reviews done by Bourgeois (1981) and Sharffman et al. (1988) allude to the development of resource fluidity as a more fundamental resolution, because absorbed slacks emerge and accumulate when excess resources do not flow well.

Coined by Doz and Kosone, resource fluidity refers to an organization’s ability to reconfigure internal operations so as to channel resources across business units for productive use (Doz and Kosonen 2008; Doz and Kosonen 2010). IS plays an important role in developing resource fluidity: inventory management systems can pool and disperse similar resources through a central platform (Banker et al. 2006; Zhu 2004); production scheduling systems can streamline resource flow to reduce its idle time on the supply chain (Dong et al. 2009; Rai et al. 2006); and collaboration systems can mobilize resources from different units or functions and combine them for the best complementary outcome (Kumar and Van Dissel 1996; Pavlou and El Sawy 2006). Therefore, our study aims to find out how organizations use IT to develop resource fluidity in order to redeploy absorbed slack resources. Although Doz and Kosone (2010) have provided a list of tactics such as decoupling, switching, and grafting, for developing resource fluidity, little is known about the development process. Our study intends to fill the gap by exploring a process model on IT-enabled resource fluidity development. The exploration is guided by a theoretical lens comprising resource interdependence and resource interoperability, since resource fluidity across different business units needs both: interdependence functions as a conduit for resources and interoperability acts as a protocol for the flow (Ford 2008).

Interdependence and Interoperability Development towards Resource Fluidity

Thompson (1967) introduces three types of interdependence that are commonly referenced in interdependence analysis. Pooled dependence forms among units that share and use common resources otherwise independent. An example is the different production lines sharing the same pool of raw materials (Rai et al. 2006). Sequential dependence forms among units that exchange resources and where output of one unit is the input of another. An example is the connection between production and delivery, where production’s output becomes delivery’s input (Banker et al. 2006). Reciprocal dependence forms among units feeding resources back and forth between each other without a particular direction. An example is a cross-sectional coalition circulating resources across boundaries to cope with an emergent situation (Pan et al. 2012).

Adopted from system engineering literature, interoperability refers to the ability of diverse systems to work together and to share resources (IEEE 1990) and a lack of interoperability creates islands of information systems. Adapted in the social systems, interoperability has also been used in the management literature and the relationships between business units (Carlile 2002; Carlile 2004). Ford (2008) reviews and summaries three major types of interoperability that are commonly used. Syntactic interoperability forms when interoperating units are able to exchange data through the same lexicon;
without the lexicon, resources can be recognized differently across boundaries and resource sharing could be impeded (Carlile 2004). An example is the central inventory system, which codifies inventories based on the same standard (Dong et al. 2009). Syntactical interoperability is the basis for more advanced interoperability as it lays the basic communication structure. Semantic interoperability forms when interoperating units are able to interpret the outputs of each other, often via a shared reference model; without this shared model, resources can be stuck on the disconnected value chain (Carlile 2004). An example is the Electronic Data Interchange (EDI) system between companies and their suppliers, which translates focal companies’ production plan to suppliers’ delivery plan (Teo et al. 2003). Pragmatic interoperability forms when interoperating units are able to understand each other’s practice and collaborate on a clearly defined working context; without a mutual understanding, business units may not negotiate effectively and falter in face of conflicts (Carlile 2004). An example is the cooperative work systems, which helps business units brainstorm, initiate new projects, and circulate resources within the project team (Kumar and Van Dissel 1996). In summary, the development of resource fluidity relies on both resource interdependence and resources interoperability. Interoperability builds a shared space among different business units, based on which, emergent interdependent relationships are formed and function as the resource conduit that moves resources from one place to another to create the fluidity.

## Research Methodology

The nature of this study is explorative, since little research has been conducted in the area of IT-enabled resource fluidity. We choose case study as the research methodology as it fits this explorative nature (Eisenhardt 1989). We identified two multinational manufacturers based in China as our case organizations: Carta (pseudonym) is the world’s 7th largest construction machinery manufacturer with over 60 product lines; Magma (pseudonym) is world’s largest white appliances manufacturer with over 90 product lines. Both organizations started a strategic transformation between 2008 and 2010. The case selection is theoretical sampling (Eisenhardt 1989), as both organizations are particularly suitable for illuminating constructs, relationships and logics in slack redeployment. First, manufacturing industry has more absorbed slack than others given its complex supply chains (Tan and Peng 2003). Second, large companies like Carta and Magma tend to have more absorbed slack than their smaller counterparts (Sharfman et al. 1988). Third, major transformation redeploy slack resources more effectively than minor organizational changes (Love and Nohria 2005).

Data collection lasted 2 years and 10 months, from June 2010 to April 2013 and included multiple visits to the case sites. Primary data came from face-to-face interviews with 78 informants, 32 from Carta and 46 from Magma. Some informants participated in multiple sessions. Early interview sessions lasted between 45 minutes to 1 hour, and follow-up sessions are shorter, on average less than 30 minutes. Informants came from both business and IT operations, and spread across top management, middle management, and staff on the field. Earlier interviews comprised mostly open-ended questions, which are targeted to encourage informants cover as much as possible regarding resource redeployment. Later interviews became semi-structured as the central themes emerged and were focused on topics relevant to these themes (Glaser and Strauss 1967). Onsite observation and archives supplemented the interview. Onsite observation consisted of photos and videos taken at critical sites, such as production line, procurement and delivery center, and system training center, and notes taken during informal chats. Archives came from both internal and external resources. Internal archives included internal publicity like the progress reports, business cases like the cost-benefit analysis, and relevant presentation slides. External archives

<table>
<thead>
<tr>
<th>Type of Informants</th>
<th># in Carta</th>
<th># in Magma</th>
<th>Total #</th>
</tr>
</thead>
<tbody>
<tr>
<td>Managing director (e.g. CFO and COO)</td>
<td>1</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Business operations manager (e.g. supply chain manager)</td>
<td>7</td>
<td>15</td>
<td>22</td>
</tr>
<tr>
<td>Business operators (e.g. supply chain operators)</td>
<td>3</td>
<td>12</td>
<td>15</td>
</tr>
<tr>
<td>IT program manager/CIO</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>IT project manager</td>
<td>9</td>
<td>7</td>
<td>16</td>
</tr>
<tr>
<td>IT developer</td>
<td>10</td>
<td>7</td>
<td>17</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>32</strong></td>
<td><strong>46</strong></td>
<td><strong>78</strong></td>
</tr>
</tbody>
</table>
included major business periodicals like China IT Weekly, academic publications like Harvard business case on Magma, and business books written by 3rd party journalists.

Data analysis, also known as coding (Glaser and Strauss 1967), consists of two steps: open coding and selective coding. Both were conducted in tandem with data collection (Eisenhardt 1989): open coding is in tandem with the open-ended interviews, and selective coding is in tandem with the semi-structured interviews. During open coding, every line of the interview transcripts, field notes, and archives was summarized into a code, and four research assistants were recruited to do this. Because all assistants had no prior knowledge of our theoretical stance, coding bias was alleviated and codes emerged from data rather than being forced into existing cognitive models (Corbin and Strauss 1990). Codes were collated and then grouped into categories. Emergence of the high-level categories marked the end of open coding. The following selective coding focused chiefly on variables related to those categories, e.g. components of spatial fluidity. Each variable must be supported by at least two sources of data and if a variable lacked sufficient data to support, follow-up interviews were conducted toward it. This iteration between data analysis and data collection continued until the emergent model reached theoretical saturation, when newly collected data began to repeat and failed to dispute the existing model or to reveal new themes (Eisenhardt 1989).

While simultaneous data collection and analysis leveraged the flexibility of case study, stringent measurements must be put in place to ensure its reliability and validity. Reliability concerns the extent to which findings are repeatable, while validity concerns the extent to which findings match reality. Measurements used in this study were adopted from either principles of established methodology work (e.g. Klein and Myers 1999; Pan and Tan 2011) or best practices shared by sample case studies (e.g. Kirsch 2004). Examples of reliability measurements include the development of case study protocol and the build of primary and secondary databases. Examples of validity measurements include multiple interpretations and the incorporation of dialogic reasoning.

Case Description and Analysis

The description of each case is followed by the analysis on it, and juxtaposing them makes it easier to perceive how analytical models emerge from the data. Data is presented in narratives and the quotations are organized in tables, in order to create a structured view. The cross-case analysis and further abstraction are included in the discussion.

**Carta’s IT-lead Transformation towards an Integrated Enterprise System**

Founded in 1989, Carta expanded rapidly in the ensuing two decades, with a compound growth rate of 27.8% per year. In 2008, the company became the largest Chinese construction manufacturer, employing over 20-thousand employees and generating $13.8 billion revenue. The revenue was contributed by 17 subsidiaries specialized in heavy lifting, road construction, earthmoving, concrete, and oil extraction, among others. This achievement was attributed to both external and internal factors. The rising Chinese economy boosted the country’s ambitious infrastructure development and raised its appetite for construction machinery. To ride on this market trend and fuel its expansion, Carta adopted an autonomous management structure, whereby subsidiaries were given the discretion to make decisions based on their local interests. IT played a major role to support the local operation. Subsidiaries had their own IT teams. IT teams of matured subsidiaries like heavy lifting and road construction were full-fledged. The autonomy while carrying merits brought up issues as well. First, business processes tended to fragment. This is common inter-and intra-subsidiaries. For example, procurement functions of two subsidiaries were isolated, and production and welfare functions of the same subsidiary could be independent as well. As a result of this fragmentation, resources did not flow well inter-and intra-subsidiaries. IT systems also grew fragmented, since they were not architected based on the enterprise-level interests, but the local interests.

While fast growth brought prosperity and profitability, it also brought growing pains. Slack resources were one. Some resources were simply redundant. For example, the heavy-lifting division once spent five million RMB building a new warehouse, while its neighbor, the road-construction division, had much excess warehouse space to rent out. Some resources were not redundant, but were idle. An extreme example was a stack of special steel plate that the road-construction division order five years ago. A more
common example was excess product inventories that waited to be shipped to customers. “When you have large stocks, you feel safe, because if there is a surge in customer demand or hiccup in the production, you can deliver; when the stock is low, you just throw in a few more shifts.” A production manager explained the rationale for keeping those inventories. Some resources were neither redundant nor idle, but were underutilized. One example is the customer resources. For example, many customers of the road construction had expressed the desire for maintenance service at rural areas where many of the construction projects took place, but the division could not profit from this opportunity, because it lacked a widespread service network and failed to access other subsidiaries’ networks.

However, as the company grew rapidly and enjoyed premium profits, all these issues were masked. In 2008, the growth hit a speed bump, not only because of the global financial crisis, but also because international competitors started to enter the domestic market with more advanced management practices and more efficient and effective operations. It was thus imperative to address the fragmented issues and redeploy excess resources that were not in productive use, as the CEO commented “We saw the gap between us and the international players. To remain competitive, we need to transform. This crisis, to some extent, is an opportunity, as it gave us time and space to carry out big changes.” Carta launched an integrated enterprise system, the center piece of which was a SAP ERP. Carta intended to leverage the renowned best practices embedded in the software to upgrade its current management practices to match international standards. This implementation, unlike those at the subsidiaries, required collective actions across local IT teams. To coordinate these teams, Carta founded a central IT team at the headquarters and recruited an industry veteran as the CIO.

The system implementation took three phases featuring vertical, lateral and dynamic integration respectively (see Table 2). The vertical integration phase lasted from Nov 2008 to 2009 Chinese New Year. This phase aimed to consolidate resources that could be shared across subsidiaries, such as raw material and warehouse capacity. Vertical integration is less disruptive as compared to the other two types of integration. The CIO recalled the rationale for doing vertical integration first “At the beginning, people were skeptical about our ability to lead this transformation. That was not a good time to throw up big changes.” At the same time, the headquarters IT lacked resources, as the CIO recounted, “When I first arrived, I was the only one at 6th floor (where headquarters IT resided) and had no subordinate reporting to me.” External recruitment seemed to a viable and straightforward option, but the CIO rejected it for two reasons. First, subsidiaries had abundant and often underutilized IT professionals and second, existing employees knew the business better. The headquarters team took a comprehensive stock of the existing IT resources. Infrastructure-related resources such as data storage and network were later outsourced to 3rd party vendors. Business-related resources such as system analysts and implementation best practices were consolidated and centrally managed by the headquarters team. These resources were strategic assets and core to this transformation. Although many analysts remain posted at the subsidiaries, they were under direct supervision and control of the headquarters team and could be reassigned to other subsidiaries if the headquarters deemed it necessary. As resources were shared across subsidiaries, local IT teams did not need to stock their own resources and many excess resources were recouped. These resources were then deployed to the business side to carry out the vertical integration there.

Vertical integration at the business side focused on the development of a virtual resource pool. Through this resource pool, similar resources such as raw materials and production capacity can be shared across subsidiaries. Underlying this pool is a universal coding scheme which the subsidiary used to code those resources. Previously, those resources are named differently within different subsidiaries, which make the sharing difficult. To recode the resources using the same scheme was an arduous task, because it required changes to the fundamental data structure, as a headquarters IT manager commented “if it was not for the resources recouped earlier, this task could take a lot of time.” The resource pool also made the central procurement possible, which significantly increased the organization’s bargaining power against suppliers. Cost saving was thus evident and business units start to become receptive to the new IT initiatives. The positive feedback also alleviated some initial resistances from the local IT managers, who objected the idea of managing IT resources centrally, as it reduced their power.

Lateral integration started in Feb 2009, right after the Chinese New Year break. The objective was to speed up resources flow along value chains. A prominent value chain was among design, production, procurement and delivery. Subsidiary IT teams were the main driving force of lateral integration since they understood the local context better. The headquarters team coordinated the integration by
establishing enterprise-wide system architecture. The architecture is a set of policies and rules that govern the value chain system development in all subsidiaries. Based on this shared architecture, the headquarters team connected the upstream and downstream projects by sending people responsible for the downstream projects to the upstream projects sites. This benefited both parties, as an IT manager commented, “While we get extra manpower, they (the downstream project members) get their hands dirty and that is the best training.” To encourage teams of the downstream projects to get involved earlier and send their best people to the upstream projects, a policy was enacted that any architectural change made at the upstream projects would be automatically inherited at the downstream projects. As a result, the IT community sustained a healthy momentum in the value chain system deployment. This momentum was a key factor behind its success, as the IT program manager explained “That was a painful stage (for the business side), since their daily work was seriously disrupted. We had to get this over quickly and if dragged, we lost their support.” Major changes to the business side were the establishment of virtual interfaces and virtual workflow. The interfaces connected business units on the value chain. For example, the interface between design and production translated the 3D drawing of CAD into the production schedule of MES (i.e. manufacturing execution system). The workflow ensured synchronized workflow along the value chain. For example, a design change was reflected in the plan of production, procurement and delivery, and if a change in one unit upset others’ plan, it prompted a need for adjustment.

The vertical and lateral integration made it increasingly obvious that cross-business collaboration could yield better outputs. Road construction’s sales division and heavy lifting’s delivery division once formed a joint venture to explore the opportunity of remote services. The former had rich experience in remote services as its machineries were often deployed at rural places, but it lacked the geographic coverage. The latter had no experience in remote services, but had a wide geographic service coverage. This collaboration combined the best bits of the two parties and delivered outstanding results. To spark collaboration like this, a dynamic integration was initiated. Unlike the two linear integrations, dynamic integration may involve business units from different value chains or functional divisions. The integration is not permanent. Featuring experiments and exploration, this integration may be called off if the results are not promising or change as the context changes. This was not the first time Carta promoted cross-business collaboration. In fact, the top management had been encouraging the subsidiaries to collaborate, but collaboration was not popular.

<p>| Table 2: Carta’s IT-lead Transformation towards an Integrated Enterprise System |</p>
<table>
<thead>
<tr>
<th>IT Side</th>
<th>Business Side</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Phase 1 Vertical Integration (Nov 2008 ~ Jan 2009)</strong></td>
<td></td>
</tr>
<tr>
<td>Classify IT resources</td>
<td>“My first task was to take stock of what we had. Broadly, they were either related to hardware, like desktop and network maintenance or software like business and system analysts.” – CIO</td>
</tr>
<tr>
<td>Centralize IT resources management</td>
<td>“IT assets used to be idle when managed separately. Now by putting them together, we can have a holistic view and move them across boundaries when needed.” – Manager, Headquarters IT</td>
</tr>
<tr>
<td>Establish system architecture</td>
<td>“We set up an enterprise-level system structure that all local teams should follow in developing value chain systems for their subsidiaries. Gradually, this structure becomes habits in the development” – Architect, Headquarters IT</td>
</tr>
<tr>
<td>Overlap upstream and downstream projects</td>
<td>“We have a policy that architectural changes made at the upstream projects were inherited in the downstream ones. This encouraged people of the downstream projects to participate earlier.” – Program Manager, Headquarters IT</td>
</tr>
<tr>
<td><strong>Phase 2 Lateral Integration (Feb 2009 ~ Sep 2009)</strong></td>
<td></td>
</tr>
<tr>
<td>Establish virtual interfaces</td>
<td>“The key of mapping out the existing business process is the creation of virtual interfaces between functions, which allow different functional teams to understand each other.” – Manager, Accounting, Headquarters</td>
</tr>
<tr>
<td>Establish virtual workflow</td>
<td>“Sales, production, and delivery are tightly linked by the system. Changes in one will ripple to others. If one function cannot accommodate the change, there’ll be a pop-up error messages.” – Manager, Production, Heavy Lifting Business</td>
</tr>
<tr>
<td><strong>Phase 3 Dynamic Integration (Sep 2009 ~ Feb 2010)</strong></td>
<td></td>
</tr>
<tr>
<td>Establish strategy-oriented identity</td>
<td>“People no longer associate us with PCs or only think of us only when something breaks. We now take more responsibilities in solving strategic issues.” – Developer, Finance Module</td>
</tr>
<tr>
<td>Promote collective</td>
<td>“When you’re confident, you become creative and ambitious. Taking on ambitious goals needs others’ support.” – CFO</td>
</tr>
</tbody>
</table>

Thirty Fifth International Conference on Information Systems, Auckland 2014
This time, the digital platform played a decisive role. The platform consisted of systems that could be used to unveil collaborative insights. For example, business intelligence (BI) through analyzing data collected from multiple subsidiaries unveiled business opportunities, which subsidiaries can collectively work on. The platform also consisted of systems that promoted communication. Examples were internal social network and wiki. The popularity of cross-business collaboration was not possible without a universal data-oriented mentality. The intensive use of IT and prevalence of systematic data had made this mentality perpetuate easily and employees had developed a habit of using data to support their decisions and launch collaborations. Supporting this digital platform was a vibrant IT community that strived for IT innovation. The earlier successes built confidence into the IT community and led it to assume an identity as a strategic partner to business rather than a support agent. “This new identity was crucial to the IT community, because only when they acted in a strategic mindset, could their collaboration deliver strategic innovations.” the CFO explained. To spark IT innovation, the headquarters returned some discretions to the local IT teams, as a senior IT manager explained the rationale “While central management was good for painstaking tasks, it did not fare well in getting new initiatives.” This time, new initiatives from the local IT teams must seek approval from the headquarters before implementation. The headquarters team decided its priority, line-up, and development paths. This practice prevented new initiatives from degrading into chaos or a vector of isolated systems, like what happened in the early years of autonomous management.

After implementing the integrated enterprise systems through the three phases of integration, the company achieved much success in redeploying slack resources. This is evident in the statistical index relevant to slack redeployment: the raw material usage rate was increased by 15%, the inventory turnover rate was increased by 25%, and the production lifecycle was reduced by 40%; as a result, sales after a slight decline during the ERP implementation achieved continuous growth for 5 consecutive months from 2009 August to 2009 December; in 2009 December, the sales was 60% higher as compared to that in 2008; meanwhile, the profit rate also increased from 5% before transformation to 9% after transformation.

**Magma’s Business-lead Transformation towards a Nodular Organization**

Magma was founded in 1984 and expanded rapidly in the past two decades. In 2008, its annual revenue reached $17.4 billion, triumphing over all the competitors and making the company world’s largest white appliance manufacturer. Magma’s transformation took place at the same period as that of Carta and was motivated by similar reasons such as business fragmentation and underutilized resources. For example, due to the bureaucratic systems and slow market responses, many business lines were producing products that were no longer desired by customers. Many of these products became inventories that were hard to move and eventually sold under serious discounts. Another example of slack resources was the production capacity. Because there were over 200 product lines, it was not uncommon to observe some production lines fully occupied while others underutilized. This was most evident in the production for air-conditioners, where the line was very busy during summer and largely idle during winter. During the slow seasons, its capacity could have been shared with other product lines, but before the transformation, there was not a formal channel to do so and the capacity was wasted. Slack not only came from the warehouse or factor, but also human resources, which were in fact the most valuable asset of Magma. Previously, the sales teams, for example, were separated according to the products that they were responsible for. This arrangement created many independent sales teams, which not only wasted the cross-selling opportunities but also the opportunities to consolidate sales channels and increase bargaining power with distributors.

But, instead of consolidating the fragmented business units, Magma went the another direction by breaking the business into small pieces and assigning each piece to an individual business unit, also known as a node. The organization was broken into over 2000 nodes, each responsible for its own profit and lost. The internal IT community was expected to take on an important role in this nodular transformation. But in 2008, the community was not in a good shape. The two-decade expansion had left a legacy of over 600 systems, most of which were isolated from each other. These legacy systems not only
consumed resources but also prevented the existing businesses from changes. Instead of adopting a large commercial software package, Magma developed most of the systems internally and when a development exceeded the internal capacity, 3rd party vendors would be brought in. Many Chinese organizations are like Magma, which preferred developing customized IT systems on their own over acquiring commercial software packages. These companies had often accumulated unique business practices that they viewed as part of competitive advantage, and aspired to retain them in the new systems. As the CEO of Magma once commented, “We had accumulated many unique business practices through restless experiments and reflections; they are very dear to us and really, part of our soul. Therefore, IT should retain them instead of adjusting them to the so-called standards.”

The transformation towards a nodular organization took three phases that featured dynamic, lateral and vertical clustering respectively (see Table 3). Dynamic clusters were formed to exploit the best opportunities emerged from the market. The clusters were not stable, because it would be dismissed when the market situations changed or when the collaboration failed to deliver the expected results. Driving this dynamic clustering was the entrepreneurial context set up by the top management, where individual nodes were responsible for its own profit and lost and were expected to initiate collaborations among themselves rather than being assigned to work together. Therefore, it becomes common to observe nodes competing with each other for a spot in a promising project. However, no spots were permanent. An incumbent node could be voted out by others if it underperformed. This entrepreneurial context organized nodes into a vibrant community like an invisible hand and sparked creative synergies among them. For example, A US refrigerator sales node once received an order from a local nursing home, which had very specific requirements and an urgent deadline. The sales node uploaded the requirement to the intranet and soon formed a project team comprising a design house in Germany, a factory in Vietnam, and a local shipping partner. Together, they delivered a highly customized product within the time frame and shared the premium profits generated. As the sales representative commented, “After splitting the organization into nodes, people start to act like wolves; they are active in searching new opportunities, quick in forming new alliances and ready to rejoice the spoils after victory.”

A variety of IT tools were built to support this dynamic clustering. Examples included a virtual discussion forum built to circulate real-time information, a knowledge repository built to spread knowledge, and a market intelligence system built to unveil collaborative opportunities. Despite the different functionalities, the tools aim at a common goal. That was to build an IT platform conducive for collaboration. All tools are built by the IT community instead of a single team. The development process can be seen as an ensemble of multiple IT teams that come from different backgrounds and used to belong to different subsidiaries. Collaboration within the IT community was also proactive. The question that IT leaders frequently asked each other was: How could we break down the walls and become more collaborative and innovative? These leaders also recognized that what differentiated them from the external vendors was their in-depth understanding of the business and the ability to launch innovative solutions based on this understanding; otherwise, they could be replaced by the IT outsourcing vendors and cease to exist. To this end, the IT community assumed a strategy-oriented identity and started to take ownership of business issues.

While dynamic clustering circulated resources for the best combination possible, resource transfer between connected nodes could still go awry as the workflow might be disrupted. Lateral clustering was invoked to address this issue. An illustrative example was the value chain created for the jumbo washing machine: a Pakistan sales node once detected a market gap for a large washing machine catered to heavy Muslim robes, and quickly formed a project team with two US design houses, an India factory, and a local delivery center; the product turned out to be a huge success and established a busy value chain; and to ensure the workflow on the chain, the nodes signed an internal contract, which stipulated punishments for late delivery and hiccups, as a sales manager recalled “Workflow disruption attenuates the final output and affects everyone’s evaluation, ones who cause it should be held responsible.” Since contract negotiation took place in nearly every value chain that a node participated in, an interface role emerged. “Interfaces are important and they must be politically savvy to understand and coordinate interests of other nodes that are connected with it.” A strategy department manager explained.

To support lateral clustering, IT reconfigured existing systems to form a virtual value chain that connected individual business processes. Because of the new evaluation mechanism, system users (i.e. business nodes) were increasingly impatient towards system development that inevitably disrupted their daily operations. It became clearer that the key difference between a successful and failed system
implementation was the duration. One senior IT manager elaborated, “Gone were the good days (laughter) when a system could be developed phase by phase and module after module. Now, everything is broken down into pieces and assigned to different parties to work on them simultaneously.” To coordinate different parties, the headquarters IT established system architecture like that in Carta, which defined the properties of each subsystem and the dependence among them. Like that in Carta, this architecture later became a communication protocol in the IT community.

Vertical clustering was proposed to share resources across nodes. At the heart of this clustering was the development of service nodes responsible for their own profit and lost. A strategy department manager explained the motivation, “Each service node specializes in a specific area, such as procurement and delivery, which other business nodes can access.” Outsourcing daily minutia to service nodes focuses the business nodes’ attention on their core competences. To consolidate resources at the service nodes, service resources like storage space and delivery capacity must be universally coded; otherwise, they would remain scattered across the organization. This was not Magma’s first attempt to consolidate shared services. The previous attempt failed, because different business units defined service resources differently. Inspired by resource consolidation at the business side, IT community followed suit and recouped many redundant resources. These resources were then redeployed to support the recoding efforts, which were an arduous task as it required changes in data structure of many existing systems.

Similar to Carta, after the transformation towards a nodular organization, the company achieved much success in redeploying slack resources. As the statistical index showed: idle inventory in the warehouse was reduced by 80%, the warehouse space was reduced by 85%, inventory turnover rate was increased from 4% to 14%; as a result, the revenue and the profit in 2009 increased by 9.2% and 20.8% as compared to those in 2008.

<table>
<thead>
<tr>
<th>Table 3: Magma’s Business-lead Transformation towards a Nodular Organization</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>IT Side</strong></td>
</tr>
<tr>
<td>---------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Phase 1 Dynamic Clustering (Aug 2008 ~ Jan 2009)</strong></td>
</tr>
<tr>
<td>Establish strategy-oriented identity</td>
</tr>
<tr>
<td>Promote proactive collaboration</td>
</tr>
<tr>
<td><strong>Phase 2 Lateral Clustering (Feb 2009 ~ Aug 2009)</strong></td>
</tr>
<tr>
<td>Establish system architecture</td>
</tr>
<tr>
<td>Connect upstream and downstream development</td>
</tr>
<tr>
<td><strong>Phase 3 Vertical Clustering (Aug 2009 ~ Dec 2009)</strong></td>
</tr>
<tr>
<td>Classify IT resources</td>
</tr>
<tr>
<td>Centralize IT resources management</td>
</tr>
</tbody>
</table>
Discussion

Interoperability and Interdependence towards Resource Fluidity Development

Both organizations develop resource fluidity through the creation of resource interoperability and resource interdependence (See Figure 1) and this development applies to both IT and business resources. In Carta, resource fluidity is developed through vertical, lateral and dynamic integration. All three types of integration start from the IT side. Vertical integration starts an establishment of a headquarters IT team. An important objective of the team is to coordinate IT resources across subsidiaries. Those resources used to reside in local subsidiaries and do not flow across boundaries. The central team first creates a shared scheme that local IT teams use to classify their resources. The scheme acts as a common lexicon towards IT resources and creates a syntactic interoperability in the IT community with regards to resources (2008). This interoperability affords a shared space where interdependent relationships can be created. A pooled interdependence is created at this stage as IT resources are consolidated at the headquarters and local teams instead of maintaining their own resources share resources from the same pool (Thompson 1967). Because IT resources are released from their local environments and moved across team boundaries, the organization develops a capability that we term spatial resource fluidity. The slack IT resources recouped here are then used to develop a resource lexicon at the business side and to forge interdependent relationships there, as a result cultivating a spatial resource fluidity. Positive outcomes from the business side in turn justify the vertical integration and stabilize the spatial fluidity at the IT side.

Lateral integration starts from developing a system architecture shared by the entire IT community. Because this architecture acts as a shared reference model that different IT teams use to interpret each other’s outputs, it builds a semantic interoperability in the IT community (Ford 2008). This interoperability affords a space where a sequential interdependence is created, because the organization links its upstream and downstream IT projects and makes sure outputs of the upstream project become inputs of the downstream project (Thompson 1967). Because IT resources can flow rapidly from one phase to another without much delay and idleness, the organization develops a capability that we term temporal resource fluidity. Dynamic integration is supported by IT community’s newly assumed strategic identity. This new identity gives different IT teams a shared goal and a common ground to understand each other, and thus creates a pragmatic interoperability (Ford 2008). In-depth collaboration emerges based on this common ground and creates reciprocal interdependence among different IT teams (Thompson 1967). Shared identity does not mean the collaboration is without friction. In fact, friction of competing ideas sparks innovation (Vera and Crossan 2005). Because IT resources can be moved from their original usage context to a new situation that promises better yields, the organization develops a capability that we term situational resource fluidity.

In Magma, resource fluidity is developed through a transformation towards a nodular organization. From a resource perspective, each node can be seen as a combination of resources. Because nodes can freely move around and are encouraged to take initiatives, resources residing in them are released from a fixed context and mobilized across boundaries. This business-lead transformation starts from dynamic clustering, where clusters are formed and dismissed based on the dynamic market conditions. Entrepreneurial spirit plays an important role in channeling nodes towards value co-creation and high-
yield collaboration (Levitt 1960), because it acts as a shared mental space for collaboration and creates a pragmatic interoperability among different nodes (Ford 2008). Based on the shared mental space, collaborations become commonplace and a reciprocal interdependence is created (Thompson 1967). Since resources are mobilized based on the latest situation, a situational resource fluidity is created. This new development at the business side raises the bar for IT support. IT teams should no longer work independently and produce only automation tools. If so, they could be replaced by outsourcing vendors (Ang and Slaughter 2001). Therefore, IT teams proactively engage each other and explore opportunities for strategic innovations (Sabherwal and Chan 2001). Contextual fluidity is thus developed at the IT side. An important outcome is the digital platform, which sparks collaborations (Pavlou and El Sawy 2006) and stabilizes contextual fluidity at the business side.

Lateral clustering starts from the creation of business interfaces. The interfaces help different nodes to interpret each other’s outputs and create a semantic interoperability between them (Ford 2008). With the help of the interfaces, internal contracts are created between two adjacent nodes in the value chain. The contracts aim to make sure outputs from the preceding nodes are suitable inputs of the succeeding nodes. Because resources are mobilized across the value chain with little delay or idleness, a temporal resource fluidity is developed. Vertical clustering takes place at the very end as a means to consolidate similar resources across nodes. Similar to that in Carta, a resource classification is created as a common lexicon for nodes to interoperate syntactically (Ford 2008); and because common resources are consolidated at service nodes and shared by all the business nodes, a pooled interdependence is created (Thompson 1967). As a result, business community develops the ability to circulate resources from one place to another, substantiating the spatial resource fluidity.

Resource Fluidity Development towards Slack Redeployment

Transformations at Carta and Magma share similar starting points. At the onset, both organizations are plagued by absorbed slack resources. These resources reduce the efficiency and increase the operational cost (Bourgeois 1981). But, they are the necessary evil during a period of rapid expansion, when new initiatives tend to be spontaneous and business process tend to be fragmented (Miner et al. 2001; Moorman and Miner 1998). Fragmented business processes prevent resources from circulating well and eventually lead to slack (Sharfman et al. 1988; Tan and Peng 2003). Carta and Magma redeploy slack resources through two distinct paths of fluidity development. These two paths can be attributed to the different transformational strategies adopted by the two organizations. In Carta, the implementation of an integrated enterprise system, as reflected in the vertical, lateral and dynamic integration, reflects a centralization strategy, because their key objectives are to consolidate the management of fragmented businesses under a central unit (Hutchcroft 2001). Magma takes the opposite direction. Instead of integrating fragmented business under one central unit, the business adopts a decentralized strategy, by dividing the fragmented business into self-organizing nodes, each responsible for their own resources (Jarvenpaa and Ives 1994); these nodes, through forming lateral, vertical and dynamics clusters, develop fluidity that mobilizes resources across their boundaries. The distinct paths also lead to different outputs. Although in both organizations, absorbed slack resources are transformed into productive use. The use of resources in Carta has become more efficient, as resources are organized through a central unit, which coordinates resources under the saving mentality. The most common example used by Carta as an achievement of slack redeployment is that redundant resources are either eliminated or shared. The use of resources in Magma has become more flexible, as resources are organized not by a central unit, but by self-organizing nodes on their own. This bottom-up approach gives resource deployment more flexibility and the key objective of these nodes in their collaboration is to make more effective use of the resources and identify new opportunities that afford high-yield combination of these resources.

Among the three types of resource fluidities, situational fluidity is the most challenging to develop. First, the development requires the combination of pragmatic interoperability, a more advanced and complex interoperability (Carlile 2004), and a likewise challenging reciprocal interdependence (Kumar and Van Dissel 1996). Second, the development of situational fluidity resembles an explorative expedition confronted by the unknowns and the other two types of resource fluidity embody exploitation fed by what is already known (March 1991). In most cases, explorative tasks are more challenging than exploitative tasks (Groysberg and Lee 2009). Third, situational fluidity features unstructured interactions and the other two feature structured interaction. Unstructured interactions make application of IT more difficult (Markus et al. 2002). Commensurate with its challenge, situational fluidity is also most rewarding. While
spatial and temporal fluidity aim at efficient use of resources and operational values, situational fluidity aims at effective use of resources that aligns with the environment and strategic values.

Figure 2: Resource Fluidity Development towards Slack Redeployment

Carta starts from the basic level and gradually moves up. This follows a small-win logic. This logic is similar to the piecemeal approach introduced by Robey et al. (2002) in ERP implementation, which refer to cases where incremental changes are undertaken each time. This logic aligns Carta’s IT-lead transformation. The company uses IT to lead transformation, because it intends to use the best practices embedded in IT systems to substitute its existing managerial practices (Davenport 1998; Lavie 2006). Modifying a rigid commercial software package like SAP ERP can be very difficult and suboptimal, because it foregoes the advantages associated with process integration (Luo and Strong 2004). But, because IT units have an inferior power in system deployment, as compared to the business users (Markus and Mao 2004) and power is important for the leading agents to put through changes (Pfeffer 1981), it is wise to start small and gain power through accumulative small wins (McCarter et al. 2011). Otherwise, major changes at the earlier stage engender skeptics and fierce resistances, which are the key recipes for failed system implementation (Gattiker and Goodhue 2005). This small-win logic also aligns Carta’s centralization strategy. The company intends to integrate all the fragmented businesses under a central platform and such a platform can only be built from the bottom up, whereby the more advanced level of fluidity uses the basic level of fluidity as the foundation.

Magma starts from the advanced level and trickles down. This follows a big-bang logic. This logic fits the IT-support or business-lead nature of Magma’s transformation. Business takes the lead in this IT-enable transformation, because the company expects to retain its unique business processes and prevent them from being neutralized by the standard software package (Luo and Strong 2004). Unlike the IT community, the business community has the influencing power (Brown and Vessey 2011). Changes sponsored by powerful agents create a sense of urgency and enthusiasm (Brown and Eisenhardt 1997), which wanes as the development goes by (Ancona et al. 2001). It is thus optimal to start from big changes and devote the highest enthusiasm to the most critical challenges. If Magma started from small changes, employees’ enthusiasm could have been wasted on the less challenging tasks and when the real challenge comes, they may no longer have the energy to tough it through. This small-win logic also aligns Magma’s de-centralization strategy. The company intends to further break the fragmented business into more flexible self-organizing groups and foster clusters among these groups. Among all the clusters, the dynamic ones are most strategic and the lateral and vertical clusters can be seen as auxiliary clusters built to support the dynamic ones. In other words, if the dynamic clusters fail, there is not point to establish the lateral and vertical clusters.
Contributions and Limitations

The study makes several important theoretical contributions. To the IS literature, the study brings in the perspective of slack resources, which have been overlooked by the IS research community. Slack resources are not only an important managerial and strategic topic (Bourgeois 1981; Sharfman et al. 1988), but a topic that IT can play an important role, because effective use of IT can redeploy absorbed slack into productive use (Kumar and Van Dissel 1996; Rai et al. 2006) while ineffective use of IT can entrench the slack and make the redeployment more difficult (Martinsons and Martinsons 2002; Ravishankar and Pan 2008). Through the theoretical lens of resource fluidity and a comparative case study, this research unveils how to use IT to redeploy absorbed slacks. Moreover, few IS studies have considered business resources and IT resources simultaneously. This gap confines our understanding to either IT’s impact on business (e.g. Pavlou and El Sawy 2006; Zhu 2004) or business’s impact on IT (e.g. Lavie 2006; Ravishankar and Pan 2008), but not their mutual influence. Our study through juxtaposing business and IT resource in the context of slack redeployment and fluidity development unveils two opposing interactions between business and IT resources. The study also makes theoretical contributions to the slack resource literature by offering a process model on how to redeploy absorbed slack resources, which has remained a gap in the extant literature despite the significant negative impacts on performance (Love and Nohria 2005; Mishina et al. 2004). The study also makes important practical contributions, especially to IS managers. First, the process model offers CIOs a new perspective about the strategic values of IT and how to devise IT strategies that are aligned with important business issues, such as deployment of slack resources. The study also offers general IT managers a guidepost to develop specific types of resource fluidity through combining different types of interoperability and interdependence, and through developing resource fluidity, redeploy both IT and business slack.

Despite the contributions, findings of this study must be considered in the light of limitations, which also point to future research directions. First, the case study is conducted in the background of a financial crisis, when austerity is the buzz word and organizations are extremely conscious about resource usage. Therefore, findings from this study may not be commonly observed in regular times when slack is often masked by good financial chats. But, it is also because of this special, hostile situation that both case organizations have gone extra miles and introduced effective tactics. We believe these tactics are still valid in the good years and should be recommended to organizations. Second, our findings are most relevant to large manufacturers that are characterized by rich resources and complex operations, by obvious excess inventories and production capacity, and sufficient resources to implement complicated IT systems.

Although all these characteristics have made the IT-enable slack redeployment and IT-enabled fluidity development more salient and therefore, renders the theoretical phenomenon more significant (Eisenhardt 1989), it overlooks the small and medium companies which have less slack resources and insufficient resources to implement sophisticated IT projects and also overlooks companies in other industries such as service, whose slack resources could be less obvious and whose slack redeployment needs a different strategy. Therefore, the findings may not be readily apply to these conditions. This limitation also points a future research direction. In the future, it will be interesting to study how small companies and companies in other industries use IT to redeploy slack, and conduct comparative studies across companies of different scales and in different industries.

References


30th International Conference on Information Systems, Auckland 2014


