Coordination Network Analysis: A Research Framework for Studying the Organizational Impacts of Service-Orientation in Business Intelligence

Mark Keith
Arizona State University
mark.keith@asu.edu

Haluk Demirkan
Arizona State University
haluk.demirkan@asu.edu

Michael Goul
Arizona State University
michael.goul@asu.edu

Abstract

Business intelligence (BI) technology and research is maturing. In evidence, some practitioners have indicated a shift in the nature of their data warehousing challenges from being technical in nature to organizational [15]. This research is based on a case study involving the BI services unit of a large, Fortune 500, financial services organization that is experiencing some of those “organizational” challenges. As a solution, this organization decided to implement a Service-Oriented Enterprise (SOE) structure to address some of those challenges such as collaboration and standardization in a complex, interdependent environment. However, because of the newness of SOE and the limited volume of rigorous research in the area, it is difficult understand or estimate the organizational and human impacts it will have. This paper proposes a coordination network analysis as a research methodology for estimating and optimizing the impacts of SOE at the individual and group level.

1. Introduction

As organizations initially began adopting data warehouses, there was a need for technical research and prescriptions for implementation. However, research in data warehousing and business intelligence has matured. Eventually, rigorous research emerged that modeled the critical success and failure factors of data warehouse projects [16] [30]. More recently, researchers and practitioners have cited the need to address the organizational and human challenges in data warehousing [15].

Herrmann [15] discovered in a practitioner-based survey that many of the perceived challenges in data warehousing up to that point were primarily technical in nature whereas the perceived future challenges concern organizational issues like interdepartmental coordination and the standardization of activities and processes. In general, a corporation’s demand for timely and accurate decision support information drives the need for a database environment that is flexible, adaptable, and accessible [16] [30]. Each of these needs and challenges concern the structural dimension of data warehouses. This research is based on a case study examining the potential of a service-oriented enterprise (SOE) structure to address some of the critical issues in data warehouses cited by Herrmann [15] like interdepartmental coordination and the standardization of activities and processes.

Service orientation is emerging at multiple organizational levels in response to the growing need for greater business integration, flexibility, and agility. Based on an IT infrastructure of web services, service-oriented architecture (SOA) has been viewed as, “…a technical architecture, a business modeling concept, a piece of infrastructure, an integration source, and a new way of viewing units of automation within the enterprise [12].” The SOE evolves as the various types
of SOAs are developed and integrated within an organization. SOA adoption offers promising benefits such as “end-to-end enterprise connectivity by removing redundancies, generating unified collaboration tools, and streamlining IT processes [4].” However, the SOA paradigm is still quite new and the technologies and practices involved are changing rapidly. As a result, research is difficult and mostly focuses on technical prescriptions of “best practices” for SOA integration in various contexts [4] [7] [9] [24]. Perpetuating this problem is the fact that there are a limited number of organizations that have successfully incorporated service-orientation, thus, providing few opportunities for academic researchers to cooperatively study the SOE concept in a practical setting. As a result, there is little research on the organizational and human impacts of SOA adoption like workflow patterns, knowledge sharing, team and organizational structures, and individual and group performance.

The authors of this paper have been cooperatively researching the impacts of SOA adoption in the business intelligence (BI) unit of a large, Fortune 500 financial services organization which has been transitioning its structure to an SOE over the last one-and-a-half years. The main contribution of this paper is a proposed research methodology for studying the organizational impacts of SOA at the individual and group level of analysis in a BI context. Termed as a “coordination network analysis,” it is based on prior research in network theory [14].

The remainder of the paper is organized as follows: 1) section 2 provides some background information on the case study and explains the organizational factors of data warehousing discovered from the case and literature, 2) section 3 discusses how a BI unit may become service-oriented from a business-modeling perspective and identifies critical issues that need addressing with SOA adoption, 3) the next section introduces and explains the concept of a “coordination network” and how it might influence the workflow and success of software development projects in an SOE-based BI unit, 4) section 5 describes the proposed research framework and methodology, and finally, 5) section 6 discusses the implications and contributions to research and practice of the method and future research directions.

2. Organizational Challenges in Data Warehousing

Practitioners have been increasingly recognizing the organizational and “people” challenges of BI [8] [26] [22]. In 2004, Herrmann [15] surveyed data warehousing practitioners to understand their perceptions concerning the nature of past and future data warehousing issues. He found that although most of the past concerns dealt with technical challenges, practitioners were more concerned about the organizational challenges in the future. In particular, the following problems were identified by at least 75 percent of the respondents:

- Difficulty in collaboration among organizational units
- No transparency of time and costs of activities
- No standardized activities and processes

As stated previously, this research evolved out of a cooperative effort of academic researchers with an international financial organization which will be termed “ABC Corp” for non-disclosure purposes. The organizational issues in data warehousing raised in Herrmann’s survey are remarkably similar to those revealed from our investigation into ABC Corp.

ABC Corp’s financial and travel services have grown rapidly for years. As a result, their IT systems have grown equally in complexity. To provide flexibility amidst changing customer needs, ABC Corp divided the services it provides into separate business units. One of those units provides business intelligence support for corporate customers. It does this by making use of a large data warehouse. For example, if a customer would like a unique aggregated view of their spending or travel data, this BI service unit will develop a new module under that customer’s portal to provide the requested information. This type of B2B decision support is one of many services offered to attract new corporate customers at ABC Corp.

The BI unit’s resources include software, hardware, people, and knowledge in the form of both tacit knowledge held by individual employees and explicit knowledge as codified, reusable software components. The BI unit structure makes use of functional groups (e.g. an ETL team) and temporary project groups formed dynamically for each customer-requested project.

At the outset of the research engagement with ABC Corp, they identified several concerns with the speed and efficiency of their operations. For example, project cycle time and costs were frequently over budget. They suspect several causes for this problem. For one thing, client requirements were often vague and changing
throughout the course of a project. Also, information and strategies used in prior projects was difficult to obtain and utilize on similar projects. Just as other practitioners had indicated in Herrmann’s survey, there was difficulty in collaborating among functional and project groups. Another problem was that project schedules were not flexible enough be accommodating to other “priority” projects. In addition, the unit manager believed that significant efficiency gains could be made by reusing more software components, project plans, and knowledge obtained by experience on prior projects.

Based on these issues, we jointly agreed with ABC Corp to pursue research into the potential for service-orientation to accommodate several of their needs. SOA for data warehouses is one of the future trends predicted by some practitioners [1]. As we set out defining and developing an SOA for ABC Corp’s BI unit, it was clear that we would also need to develop a strategy for measuring and understanding the impacts. In particular, we are interested in the organizational rather than technical effects. Subsequent sections will describe our “top-down” approach to SOE in the context of ABC Corp’s BI unit and detail a research framework for studying the organizational impacts of service-orientation.

3. SOA for Business Intelligence

Most SOA initiatives begin as incremental efforts to expose legacy IT systems as web services which are indexed into some kind of resource ontology [4]. In addition, patterns of web service use are developed and stored based on the business processes the patterns are developed after [7].

Much of the current SOA research tends to focus on technical implementation issues such as ontology development for web services [29] and legacy system migration to web services [31]. Beginning SOA adoption at the underlying web services might be considered a “bottom-up” approach to SOA adoption. With the ABC Corp BI unit, we used a “top-down” strategy by first identifying the business processes that take place, and then, describing the steps involved in terms of the “services” they are comprised of. In BI, these processes are simply the software development projects which provide new data access, mining, and configuration required by the business or its customers. The term “service,” in this sense, no longer refers to the underlying web services used in applications or IT integration, but a more abstracted level of “business” service like those provided by individuals and teams within the BI unit. Think of a software development project schedule like a pattern of business services such as planning, analysis, design, development, testing, and deployment. The choreography of these “services” comprises a project pattern that can be stored and reused. Each service can be composed of multiple sub-level services like the many sub-steps within each software development phase – each performed by individuals or groups.

In a sense, a business service is quite comparable with a web service. Both represent units of work, or service, which can be loosely-couple with other units as building blocks of a business process. Both types of services can be stored similarly in an organization’s resource ontology and both can have patterns developed and reused consisting of them. Indeed, a fully service-oriented enterprise will integrate both business and web services to achieve end-to-end integration and agility. However, modeling your organization infrastructure for business services presents very different challenges than modeling your IT infrastructure for web services. This issue of understanding and researching the organizational impacts of service-orientation is the focus of this study’s approach.

3.1. SOA as a Job Shop

A useful analogy for understanding the impacts and critical issues of SOA in a BI setting is to compare it to a job shop environment (See Figure 1). This analogy was first made by Leonhard and Davis [21] who applied the job shop concept to software development. We use it specifically in the BI context in an SOE environment like ABC Corp.

![Figure 1: Job shop analogy of ABC Corp BI unit](image)

In a job shop, work-in-progress (WIP) flows from machine to machine as it reaches completion. The flexibility and agility of the job shop depends on the ease of reconfiguring the machines and the pattern of
work flow. The job shop model is similar to the BI unit of ABC Corp. Each machine in the job shop is like a functional group in the BI unit. Each team performs a particular set of functions described as services. Projects patterns are composed by invoking these services performed by one or more individuals from multiple functional groups. In Figure 1, project “A” requires the services of person “a” and “c” of group 1, “A”, “b”, and “c” of group 2, “b” of group 3, and all of group 4. Each project brings together a different group of individuals who may or may not be familiar with each other. Even if project patterns are reused for similar projects, there’s no certainty that the same individuals will be grouped together again when you consider that people, unlike web services, will take vacations, leave their job, or call in sick.

Several aspects of a job shop environment are analogous to an SOA (See Figure 2). Bottlenecks in a job shop are similar to subject matter experts (SME) in software development. As an individual becomes “expert” in a particular area, he or she may become known as the best or only source of knowledge for that topic. Other developers come to depend on the SME for help. If the SME decides to take a vacation, leave the company, or is simply inundated with requests for help, they can become a bottleneck in the system.

Machine set-up time is similar to project switching costs. Often times, individuals will become assigned to multiple projects at once. Each time an individual stops working on one project and begins work on another, they incur switching costs from activities like setting up access to different resources, changing physical locations, cognitive stress, and time delays.

Figure 2: Job shop to SOE comparison

In a job shop, batching takes place when the same process for multiple jobs can be performed at once on a particular machine. In an SOE structured BI, an individual might be able to reuse a software component or ETL process for multiple service requests.

Workflow coordination between machines in a job shop is similar to the knowledge coordination required to complete software projects in an SOE. And finally, reusable workflow paths in job shops are related to the reusable patterns, or choreographies, of services.

4. Coordination Networks in Software Development Environments

The job shop analogy is useful because software development activities can be modeled and calibrated using traditional job shop heuristics. However, the analogy is insufficient when trying to identify specific process breakdowns and inefficiencies that increase the cost and length of BI projects. In a job shop, it is fairly transparent where the workflow is getting frustrated when the WIP piles up at a particular station. In software development, however, people do not always follow the simple patterns of workflow outlined in a project plan. Often, and particularly in difficult or poorly defined projects, individuals seek knowledge, information, and resources from others they know and trust to augment their own performance [5] [13]. Informal relationships are developed and dependencies can be created in these instances of organizational citizenship behavior [20].

4.1. Social Networks in Software Development Environments

Hidden beneath the formal project workflow pattern is a social advice or helping network that is developed informally as individuals seek knowledge and help each other in software development [10]. Research has demonstrated that these networks of informal relationships can greatly influence workflow, knowledge sharing, learning, and innovation [6] [11] [27]. Ultimately, the connectivity in a social network, whether “good” or “bad,” affects project performance in terms of meeting cost and time estimates. Figure 3 demonstrates what the social network of the ABC Corp BI unit diagram in Figure 1 might look like.
The mock diagram in Figure 3 reveals several potential problems that might otherwise go unnoticed without a social network analysis. Person “d” in team 4 has been assigned to all three projects. Perhaps that situation was planned for, but what the project managers who sourced person “d” did not know, is that he or she is also the SME for a particular topic whom many other people go to for knowledge and information. That person “d” from team 4 may likely become a bottleneck to several BI projects.

Something else to notice from the diagram is that each team has very distinct helping patterns within their functional group. Perlow et al. [25] have identified 3 distinct helping patterns in software development – team-centered, expertise-centered, and managerial-centered. Each pattern influences, and is influenced by, the organization reward systems, company cultures, and group performance. For example, team 3 seems to follow a team-centered pattern where each member freely interacts with any other member in the group for help. Team 2 follows a managerial-centered pattern because all members only refer to only the team leader, person “e,” for advice and information. Understanding helping patterns becomes critical when project leaders are deciding which individuals to select from each team in order to source a service-oriented project pattern.

Keith et al. [18] found evidence of significant differences in knowledge sharing cultures and patterns between software engineering groups within the same business unit. Entrainment theory predicts that the set of norms, cultures, and habits that become “entrained” within groups will be carried with team members even after they leave the group and the original purpose for those cultures no longer exist [2]. This implicates that people selected from different functional groups could possibly bring either clashing or compatible cultures into a BI project. Because of this, discovering the within group helping patterns as well as the social network between groups is critical to streamlining the software development project.

4.2. From “Social” to “Coordination” Network

Many types of networks exist within organizations. In software development, helping networks that emerge from social interactions are quite salient because of the knowledge-intensive nature of the work. In the ABC Corp BI unit, another critical factor to project success is the coordination of reusable software components developed during past projects. Successful coordination of these components significant reduces the extent of “re-inventing the wheel” that takes place and improves project efficiency and reduced costs. Also, it is likely that the coordination of these components influences the informal helping relationships that develop. When people at ABC Corp share a component they have built, a certain amount of explanation, training, and gratitude takes place. In addition, the successful coordination of reusable components depends greatly on the informal relationships already in place. For example, if person “e” in team 5 of Figure 3 has built a valuable component, it may end up getting unused and re-invented because nobody currently looks to him or her for information. On the other hand, if person “d” in team 4 develops a valuable component, it is likely that many others will benefit from it because of the established helping relationships in place.

Gittell and Weiss [14] has proposed the concept of “coordination” networks to describe a type of network that exists across organizations such as the coordination of patient care between hospitals, nurses, and private doctors. An analogy might be made in an SOE to the coordination network of services that may also cross organization boundaries in software development. For example, a project manager in ABC Corp may discover that a new project requires capabilities that can only be found outside of the organization and outsource one or more of the services required by the project. However, we use the term “coordination network,” rather, to refer to the informal network of resource coordination like the one that exists in ABC Corp with reusable software components. In this case, the coordination network of...
resources may also cross organization boundaries as in Gittell and Weiss’s definition.

By performing a coordination network analysis, researchers can understand how SOA adoption is influencing the knowledge sharing, workflow, and project performance in software development environments like the BI unit of ABC Corp.

5. Coordination Network Analysis

There are several advantages to basing an SOA research methodology in network analysis. For example, network analyses are conducive to multiple levels of examination like individuals, groups, and organizations [14] [28]. It is particularly useful for pinpointing external group relationships and understanding the impacts of between-group ties [17]. Also, there is a well-established set of constructs in the social network literature to accurately portray the existing relationships including individual and group performance [10] [27].

Because informal networks evolve as the indirect result of management decisions, network analyses are useful for investigating the indirect impacts of these decisions on organizational performance at multiple levels of analysis. This makes network analysis an appealing choice for studying the impacts of SOA adoption as a business modeling concept where it can be difficult to readily recognize the value-added. As a result of the case study described in this paper with ABC Corp, this next sub-section describes an application of how a coordination network analysis can best be administered in a BI context within an SOE.

5.1. Coordination and Social Network Variables

Social network analyses typically investigate the relationship of the network variables of density and centrality to individual and group performance. In addition, a topic of interest among researchers has been to discover the antecedents of network structures. We frame a coordination network in the same way as social networks, but with the addition of a few variables to discover antecedents that are specific to a resource network. This sub-section reviews each of the variables involved and offers propositions for those contributed by this study.

5.1.1. Density. Density is a group variable that equates to the overall number of relationships in a network. In Figure 3, team 3 is “denser” than team 2 and team 5 because of the greater number of relationships between members. In a coordination network, a high-density group would know who has developed which components and be able to more freely access each other’s components relative to a low-density group. Density has been found to be positively related to group performance [27] [10].

5.1.2. Centrality. Individual centrality in a network refers to the extent to which an individual is connected to, or has some type of relationship with, others in the network. In a social network, centrality is an indication of the number of people who come to an individual for advice or knowledge of some type. In Figure 3, person “d” from team 4 is a highly-central individual – which is indicated by the many incoming connections from other individuals. In a coordination network, a highly-central person is one who knows about, and has access to, reusable software components that are desirable by others in the network. Researchers have found that centrality in advice networks is positively related to individual performance [27] [3].

Group centrality equates to the variance of the individual centrality in the group. The effect of group centrality on group performance is mediated by task complexity. When task complexity is high, centrality is negatively related to group performance. Centrality is positively related to group performance when complexity is low [10].

5.2. Antecedents of Centrality in an SOA Coordination Network

An important topic of interest in network research has been discovering the causes or antecedents of individual centrality in a network. If a manager wants to influence how the informal networks develop, he or she must understand what factors lead individuals to develop network relationships. There are several established variables in the literature and we offer additional antecedents specifically for coordination networks in an SOE.

5.2.1. Individual Characteristics. Several individual characteristics have been examined as antecedents to centrality [19]. These include: 1) demographic characteristics like gender, race, age, and education (similar individuals tend to develop relationships), 2) values like work ethic, hedonism, and tradition, and 3) personality traits like extraversion, conscientiousness, neuroticism, openness, and agreeableness. In a
longitudinal study, Klein et al. found many of these individual characteristics to be significant antecedents to centrality in advice, friendship, and adversarial networks.

5.2.2. Individual Resources. As already discussed from the case study with ABC Corp, we suspect that informal networks within software development environments are influenced by the access individuals have to valuable resources like reusable software components. Besides components, software documentation and applications could also be resources that drive centrality if they are not fully available and acknowledged by the network. In addition, ABC Corp had trained SMEs and other recognized training certifications. If known about by others, these knowledge resources may lead the SMEs to become more central in the network.

Proposition 1: An individual’s access to, and knowledge of, software resources (reusable components, documentation, applications) is an antecedent to centrality in coordination networks of software development environments.

Proposition 2: An individual’s knowledge resources (subject-matter expertise and certifications) are antecedents to centrality in the coordination networks of software development environments.

5.2.3. Organizational Structure. Although managers cannot directly enforce the connectivity of their organization networks, they can indirectly influence them by the structures and environments which they can directly enforce. There are many factors of organization structure that might influence an individual’s centrality such as geographic location, organization hierarchy, and task type. However, our case study of ABC Corp has revealed several salient factors unique to the BI projects and SOE context via interviews with key personnel and a detailed documentation of work patterns.

In the ABC Corp’s BI unit, there are several teams grouped by the functional services they provide. In addition, two other teams are grouped divisionally (meaning they are project-oriented rather than functionally-oriented) and act as “integration teams” for each project. Each new project assigned to the BI unit is first given to one of these two integration teams which then select individuals or groups from various functional teams to join them in providing services as part of a project pattern.

Moon et al. [23] discovered that when team members switch between groups (as functional team members of ABC Corp do as they leave their home group to join project groups) they bring their collaboration cultures with them. In addition, Moon et al. found that functional groups tended to foster better collaboration practices than divisional groups – likely because functional groups stand more to gain from knowledge sharing since members are performing similar functions. As a result, the structure of the home group that an individual is sourced from may impact their propensity for developing new relationships in the project group.

Proposition 3: Individuals from functionally-structured home groups in a service-oriented software development environment will become more central in the coordination network than individuals from divisionally-structured home groups.

Another characteristic of ABC Corp’s service-oriented BI environment was that most individuals, regardless of their home team membership, were concurrently assigned to multiple projects. This makes sense because, in an SOE, groups and individuals are business service providers. In other words, they are assigned to provide a particular service to the project rather than being assigned to work on an entire project through to completion. As a result, employees may end up switching between projects at a rapid pace depending on the needs and requirements of each assignment. Obviously, the more projects an individual is assigned to means more opportunities to build relationships with new people. However, there is likely a point at which “too many” project assignments could be a detriment and actually decrease performance, morale, and future network centrality. Additionally, people need to have enough consecutive time working on a particular project to be able to build meaningful relationships. If people switch between projects too rapidly, they may not spend time making new connections with those on each project team.

Proposition 4a: Individual switching characteristics will have either a positive or negative affect on their centrality in the coordination network of a service-oriented software development environment depending on the
Proposition 4b: Individual switching characteristics will have either a positive or negative affect on their centrality in the coordination network of a service-oriented software development environment depending on the rate of switching between projects.

![Diagram](Image)

**Figure 4: Coordination network framework for antecedents of centrality**

In summary, the coordination network framework proposed here combines elements of a social network model with a resource network including structural variables that are significant to a software development environment of an SOE.

6. Discussion

Testing the proposed framework is beyond the scope of this paper, but some suggestions can be made for future testing based on our initial case study with ABC Corp. Since it would be difficult to analyze the coordination network for each software resource, interviews should be performed with project managers and key members to identify a small number of the software components that are of the most interest to management. The network could then be measured using traditional surveys asking each member which components they know of and who they go to for access to them. At the same time, the social helping network should be measured. Then, it could be overlaid with the coordination network to reveal the dependencies between the two. It would be interesting to find out if individuals who are highly central in the social helping network are also central in the resource coordination network. A longitudinal analysis taking multiple network “snapshots” could reveal if one network appears to be driving the other. In addition, a longitudinal analysis would make a particular contribution to network theory considering the limited number of such studies [17].

The antecedents of centrality can be measured by administering surveys to discover individual characteristics and resources. Collecting time cards could be used estimate switching characteristics and project load. Individual and group performance measures can be recorded from team leader and unit manager evaluations of individuals and groups. In a longitudinal study, the comparison of project time and cost estimates to actual results would provide an interesting measure of project group performance.

6.1. Contributions to Practice

There are several benefits to organizations from using a coordination network analysis. In software development organizations – where SOA adoption has mainly been driven from – management can benefit from discovering both the social and coordination networks that exist. This would allow them to discover the “hot” components that are driving project workflows. Limited access to these resources may be what slows project lifecycles and reduces efficiency. These analyses allow managers to identify and resolve bottlenecks. They can balance the network load so there is minimal loss when highly-central people become unavailable. Also, managers can identify expertise gaps and provide more targeted and efficient training meetings. Project managers become better informed when sourcing projects so they can select the right people with the right connections and resources for the right tasks.

6.2. Contributions to Research

The coordination network framework contributes to the growing body of literature based in network theory by establishing new, testable antecedents to centrality in a resource-based network. Individual project load and switching characteristics are important variables, not only in software development environments, but in all project-based environments within organizations. Also, it adds the construct of “individual resources” as an antecedent of centrality in the context of an SOE where pattern and resources reusability is fundamental.

To contribute to the academic research on SOA, this paper defines a network-based research methodology for understanding the impacts of service-orientation, in the business-modeling sense, to organization performance at both the individual and
group level of analyses. This is significant because it provides a theory-based research approach to the SOE paradigm.

For BI research, this framework provides a methodology for examining some of the organizational issues in data warehousing suggested by Herrmann [15]. Specifically, the difficulty in collaboration among organizational units can be at least partially addressed by a coordination network analysis. This would help to reveal the communication, resource, and knowledge gaps and dependencies that might be inhibiting collaboration. Coordination networks are just one of many network types that can be critical to BI performance. Discovering additional networks of interest that are specific to BI units would be a valuable contribution to the research in data warehousing. However, one of the advantages to using a coordination network analysis to study the impacts of SOE is that it is not necessarily specific to BI services. Although the variables and propositions were developed based on the BI environment within ABC Corp, the framework would apply in any project-based, SOE task environment that values the reuse of software components, is based on knowledge-intensive work, and is characterized by medium to high task uncertainty – meaning there is some amount of differentiation between project types. For example, any software development or other product engineering environments that make use of temporary and dynamic groups can use the framework. Future research should explore other antecedents of centrality and density and their effects on performance in an SOE.

7. References


