12-31-2006

Tracing Patterns of Large-Scale Software Reuse

Paul Witman
Claremont Graduate University

Follow this and additional works at: http://aisel.aisnet.org/amcis2006

Recommended Citation
http://aisel.aisnet.org/amcis2006/539

This material is brought to you by the Americas Conference on Information Systems (AMCIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in AMCIS 2006 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.
Tracing Patterns of Large-Scale Software Reuse

Paul D. Witman
Claremont Graduate University
witman@ieee.org

ABSTRACT

Much work has been done on software reuse “in the small” – objects, subroutines, and components. Relatively little work on very large-grained reuse is available in the literature. Such reuse might include reusing an entire automated teller machine (ATM) software and hardware system for banks across multiple continents; and reusing infrastructure across Internet, teller systems, and ATM systems, all with relatively little customization and rewriting as compared to commercial systems. This study focuses on the internal reuse practices of a very large banking corporation, with significant software development capabilities, and their experiences with reuse at fine-, medium-, and particularly the large-grained levels.

Keywords
Software reuse, case study, large scale

BACKGROUND

While there has been much research on software reuse at the fine- and medium-grained levels, there is relatively little published on actual case studies of very large-grained software reuse within corporate environments. This study focuses on a very large banking corporation, with significant software development capabilities, and their experience with reuse at fine-, medium-, and large-grained levels.

Much work has been done on software reuse “in the small” – objects, subroutines, etc. Relatively little work on very large-grained reuse is visible in the literature – e.g., reusing an entire ATM software system for banks in North America, Europe, and Asia – all with relatively little customization and rewriting as compared to commercial systems.

Software reuse has been defined as “the systematic practice of developing software from a stock of building blocks, so that similarities … between applications can be exploited” (Morisio et al., 2002). BigBank, and the BigBank Technology Center (BTC) in particular, have created a large number of software and hardware systems (large building blocks) that have been reused in multiple businesses and countries. BigBank and BTC thus provide a rich source of data for case studies to look at the characteristics of those projects and why they’ve been successful. Additionally, it may be possible to examine projects that have been less successful and to understand what has caused those results, and how to improve them.

BTC is a technology development unit of BigBank, operating primarily in the Western United States. Approximately 500 people are employed by BTC, and it is organized to deliver both products and infrastructure components to BigBank. BTC’s product line includes system infrastructure, consumer Internet banking services, teller systems, automated teller machine software, and network management tools.

Potential research questions were developed in preliminary discussions with BTC research sponsors. Some of those questions are noted below, as foundational to proposing a research design.

• How does the organization attempt to reuse their investment in software, and what models have been proven to be successful, and to what degree?
• From theory to practice for reuse – what did the organization learn, how does it decide on reuse options now, and how did its behavior change over time?
• What are the economic and technological benefits of reusing large-grained software components?
• What are the costs and challenges associated with such reuse (technical, organizational, economic, vendor, outsourcing, etc.)?
• What are the foundational technical elements required to support large-grained reuse (reusable abstracted interfaces, business rules, etc.)?

The value of reuse may be found in its direct economic value (e.g., costs saved), as well as in improved time to market, reduction of redundant effort, and deployment of more stable components. Cost elements of the reuse program include the potentially higher costs and slower time to market of initial delivery of a new product designed for reuse, as well as the added organizational friction inherent in having a single component serving multiple projects, and having to balance demands from those projects.

Numerous factors can contribute to the success of a reuse program. These include technical, organizational, and market factors, as well as the organization’s ability to properly consider in-house vs. purchase decisions (Karlsson, 1995). Similarly, technical activities also contribute heavily to the success of the reuse program, including commonality of methods and tools, investment in component reuse tools, and definition of interfaces to allow broad reuse.

Formal approval from both BTC and the University’s Institutional Review Board was received on March 17 & 24, 2006, respectively, so all procedural requirements are complete.

METHODS AND THEORETICAL FOUNDATIONS

Yin (2003) describes three conditions to be evaluated in determining the type of methodology to be used. These include the type of research questions to be asked, the control that the researcher has or needs over subjects’ behavior, and the degree of focus on contemporary vs. historical events.

Key research questions for this study are of the “how” and “why” form, along with “what” questions related to the organization’s learning. No control of subjects’ behavior is required to conduct the study, as the objective is to study what has happened relatively recently and what is happening – leading to a focus on contemporary events.

Yin (2003) provides a set of three criteria that characterize situations appropriate for a case study. First, where there are more variables of interest than data points available; second, where there is a need to examine multiple sources of evidence, and third, where there is preexisting theory on which the case might be based. Multiple sources of evidence, including archival documents, interviews, and physical artifacts, are available and can contribute to the data collection required for this research. Due to the ability, within BTC, to replicate the case study process across multiple cases, and to gain greater robustness of the results, a multiple-case study is appropriate.

Measures and theories will be developed a priori, to help propose causal relationships, determine domain for generalization, and demonstrating that the inquiry is value-free (Dubé and Paré, 2003). The intent of the study is to look deeply and from multiple angles at the processes and structures around reuse at BigBank, another motivator for a case study design (Benbasat et al., 1987).

Numerous other key authors have identified elements of a reference model that will be examined in the course of the case study. Various models (e.g., Nada and Rine, 1998, Davis, 1993, Bassett, 1997) have been proposed that capture key organizational practices to support software reuse at a variety of levels of granularity. Sanchez (1995) provides a great deal of work on the more general topic of innovation, especially with respect to interoperability and modularity in the physical world, but these concepts are also key to large-grained software reuse.

Notions of organizational learning (e.g., Argyris, 1992) will be explored to determine whether BTC is able to operate with Model II (double-loop) learning. This would enable it to avoid the self-limiting learning boundaries that organizations often place upon themselves, and gain greater value from the reusable software assets that they have created. The notion of knowledge levels (Sein et al., 1999) will also be explored, to gain an understanding of how broadly and deeply the organization understands the concepts and value of software reuse.

LITERATURE REVIEW AND CONCEPT MODEL

One obvious requirement of the conceptual model (see Figure 1, below) is to define the measurement of success of the system and of the level of reuse achieved. DeLone & McLean note that “IS success is a multidimensional concept” (DeLone and McLean, 1992) and as such there will likely be several components to measure to evaluate the success of reuse for any particular project or product. Like DeLone & McLean’s model, we attempt here to capture notions of the drivers of intent to reuse, as well as the drivers of the tactics used for reuse, to identify elements that are most likely to contribute to successful large-scale software reuse.
Reuse success may be characterized as consisting of the notions of reuse efficiency and reuse proficiency (Davis, 1993). Reuse efficiency is defined as the ratio of the value of assets that were actually reused divided by the reuse the organization attempted to make. Reuse proficiency uses the same numerator, but divides instead by the total amount of reuse the organization might possibly have made. These two measures will be used as an initial proxy for reuse success.

Supporting the notion of tactics for reuse are such elements as small-scale reuse successes, reuse metrics, business case elements, reuse maturity, organizational incentives, reuse technology, and architecture, such as interoperability and modularity. Reuse on a smaller scale has been shown to be supportive of larger scale reuse (Henry and Faller, 1995). The success of small-scale reuse programs seems likely to build credibility for future proposals for larger scale reuse.

BTC may make the business case for reuse both formally and informally. Numerous models have been proposed to capture the economic elements, and to go beyond to the intangibles as well (e.g., Malan and Wentzel, 1993). A solid business case should be supportive of large-scale reuse. Business value can be defined by total cost of ownership, time to market, quality, functionality, and consistency across business units, to the extent that BigBank values those measures.

The various measures of organizational reuse maturity (Bassett, 1997, Davis, 1993, Karlsson, 1995) provide a framework by which to measure the propensity of the organization to be successful in software reuse. This overall measure, which includes within it some of the more detailed elements of this model, should provide an indication of the likelihood of reuse success.
As the organization has changed and matured through the years, it is likely that it has learned from its successes and failures to reuse software components. This learning can be expected to feed back into its success in both small- and large-scale reuse. The organization may provide implicit and explicit incentives and disincentives to its managers and staff to support software reuse. These incentives can be expected to contribute to reuse success (Karlsson, 1995).

Numerous barriers to reuse are likely to exist or to have existed in the past in the organization. Existence and impact of those barriers is likely to contribute negatively to the success of reuse, both on small and large scales (Sherif and Vinze, 2003). It will be important to look at BigBank’s results vs. their potential, and understand why reuse perhaps doesn’t occur as much as it could.

The organization’s track record of success, at the large scale, should reasonably be expected to drive the intention to reuse. Likewise, the business value of reuse, as represented by lower costs, a better fit of product requirements to business needs, and better control for the business, should also contribute not only to the development organization’s intent to reuse, but also to the business units’ agreement to accept reused and reusable software.

The final component of the business value element is the evaluation of build vs. buy decisions. As the software industry matures, it often reaches a state where a component that a company once had to build for itself is now acceptably produced by the industry. At such time that the organization chooses to acquire and integrate those components (whether open-source or COTS), reuse is taking place, albeit in a different form. These evaluations likewise contribute to the overall business value.

RIVAL THEORIES

One potential rival theory is that reuse decisions are based on capricious management approaches based on BigBank and BTC’s power structure (Salancik and Pfeffer, 1977). For example, the decision to reuse software may be driven by elements of the power structure that favor BTC’s development organizations, possibly at the expense of the business units involved. This of course requires rigorous definition of “best” for the business unit, as decisions optimized for an individual business unit may be suboptimal for units further up the organization hierarchy, and for the organization as a whole.

In a panel discussion, Cox noted the potential impact of market forces on the reuse process. Software engineers will presumably make rational free-market choices about whether to reuse components from a repository or to build them from scratch, absent other forces driving the decisions (Seidewitz et al., 1993). Cox argues that market forces like those that shaped the industrial manufacturing market should similarly impact the market for software components, but that the low cost of internal copying and modifying causes fundamental changes in that market. It will be important to understand BigBank’s economic models used to evaluate buy vs. build vs. reuse decisions.

CONCLUSION

The research will deliver new knowledge about large-grained system reuse, and the nature of the elements required to support it. These elements include technology, organization, and economic factors, as well as in-house vs. vendor product evaluation. Much has been written about these elements independently, and in relation to finer-grained reuse, but this project will focus on larger-grained reuse than have been previously described in the literature.

The researcher will document “theory to practice” for reuse. This will look at questions of what the organization learned over time, how it decides on reuse options now based on that learning, and what it foresees changing in the years ahead in terms of technology, external competition, and internal organizational impacts.

The researcher will endeavor to develop a model that describes BTC’s success, and that can be potentially generalized to other banking organizations. In addition, the model should be generalizable outside of the banking industry, and will attempt to document the critical success factors of large-scale reuse, including organizational, technical, and economic factors.

ACKNOWLEDGMENTS

The author appreciates the support and guidance of his doctoral committee, Drs. Terry Ryan, Lorne Olfman, and Ron Riggio. The author is especially grateful to BTC for the anticipated access to valuable research data.
REFERENCES