Systems Development in State Government

CHAPTER 4

Although the primary focus of our evaluation was the Statewide Systems Project (SSP), we wanted to obtain a broader perspective so we could determine whether the state's experience with SSP was unusual. We also wanted to draw lessons from a variety of computer software development projects. Therefore, we spoke with national experts, reviewed a wide range of literature, and interviewed a number of Statewide Systems Project participants about what, in retrospect, they would have done differently. In addition, we examined five other recent state of Minnesota systems development projects, including:

- Maxis (Department of Human Services' recipient eligibility project);
- Project Delta (a Pollution Control Agency project to modernize their technology and improve permitting and enforcement);
- Project Daedalus (a Department of Labor and Industry imaging and document retrieval project);
- the Department of Revenue's sales tax project; and
- MMIS2 (an update of the Medicaid Management Information System in the Department of Human Services).

A project synopsis of each project is included as Appendix B.

To focus this part of our evaluation, we asked the following questions:

- **What are the typical characteristics of successful and unsuccessful information technology projects?**
- **What strategies should the state follow to maximize the chance of success with future computer development projects?**
- **What lessons can be learned from the Statewide Systems Project?**

OVERALL OBSERVATIONS

In reviewing the relevant literature and consulting with a variety of computer software development experts as well as state system development project managers and project participants about successful and failed systems development projects, we found that:

- **Neither the private sector nor the public sector is consistently good at computer software development.**

There is much uncertainty associated with systems projects.

The experts we consulted told us that software development is relatively new and constantly changing. Unlike constructing a building or a highway where the state of the art has developed over thousands of years, the standards and technology of systems development are relatively new and rapidly evolving. Because the sophistication of computer software development is not well developed, there is much more uncertainty associated with systems development projects than with constructing a building or repaving a highway.

According to a 1994 study of a sample of 365 public and private sector organizations by the Standish Group, a technology consulting firm, almost "one third of all [information] systems development projects are canceled before they are ever completed," and "only sixteen percent of all IT (Information Technology) projects were considered successful."

> The Standish Group research shows a staggering 31.1% of projects will be canceled before they ever get completed. Further, results indicate 52.7% of projects will cost 189% of their original estimates. The cost of these failures and overruns are just the tip of the proverbial iceberg. The lost opportunity costs are not measurable, but could easily be in the trillions of dollars. One just has to look to the City of Denver to realize the extent of this problem. The failure to produce reliable software to handle baggage at the new Denver airport is costing the city $1.1 million per day.
>
> Based on this research, The Standish Group estimates that in 1995 American companies and government agencies will spend $81 billion for canceled software projects. These same organizations will pay an additional $59 billion for software projects that will be completed, but will exceed their original time estimates. Risk is always a factor when pushing the technology envelope, but many of these projects were as mundane as a drivers license database, a new accounting package, or an order entry system.
>
> On the success side, the average is only 16.2% for software projects that are completed on-time and on-budget. In the larger companies, the news is even worse: only 9% of their projects come in on-time and on-budget. And, even when these projects are completed, many are no more than a mere shadow of their original specification requirements. Projects completed by the largest American companies have only approximately 42% of the originally-proposed features and functions. Smaller companies do much better. A total of 78.4% of their software projects will get deployed with at least 74.2% of their original features and functions.[1]

1 The Standish Group, *Chaos: A White Paper* (Cape Cod, 1995).

According to the Gartner Group, another worldwide technology consulting firm that tracks information technology projects, "over 80 percent of large systems development projects fail to come in on-time, on-budget, and meeting user expectations."[2] According to the Gartner Group, large system development projects, defined as over $6 million in cost, have a failure rate over 90 percent.[3]

The vast majority of large systems projects have problems.

Although the exact percentage of computer development "failures" varies depending on the study (as does the exact definition of "failure"), the vast majority of large systems projects clearly have some sort of significant problem. There have been many notable cases in both the private and public sector where many millions of dollars were spent -- in some cases hundreds of millions of dollars -- and no computer system was ever turned on. The General Accounting Office notes that this fact "highlights the reality of the complexity in planning, designing, and managing successful IT (Information Technology) projects."[4]

Even large state governments like California have not had much success with systems development. According to the California Legislative Analyst, many major computer systems developed by the State of California have experienced serious problems (see Figure 4.1).

The federal government has also had problems with systems development.

The General Accounting Office (GAO) has found similar difficulties with federal government software development. According to the GAO, "The management of IT (information technology) projects has long been a significant problem for many federal agencies. Federal information systems often cost millions more than expected, take longer to complete than anticipated, and fail to produce significant improvements in the speed, quality, or cost of federal programs."[5]

Management information professors Kenneth and Jane Laudon sum up the situation:

> In nearly every organization, information systems take much more time and money to implement than originally anticipated, or the completed system does not work properly. Because so many information systems are trouble-ridden, designers, builders, and users of information systems should understand why they succeed or fail.[6]

---

2 Richard Hunter, Gartner Group, teleconference with audit staff, August 30, 1996.

3 *Ibid.* The Gartner Group told us studies have found over 90 percent of large system development projects fail to come in on-time, on-budget, and meeting user expectations.

4 General Accounting Office, *Information Technology: Best Practices Can Improve Performance and Produce Results*, Testimony before the Subcommittee on Government Management, Information and Technology, Committee on Government Reform and Oversight, U.S. House of Representatives (Washington, February 26, 1996), 7.

5 General Accounting Office, *Information Technology Investment: Agencies Can Improve Performance, Reduce Costs, and Minimize Risks* (Washington, September 24, 1996), 1.

6 Jane Laudon and Kenneth Laudon, *Essentials of Management Information Systems* (Upper Saddle River: Prentice Hall, 1995), 297.

Figure 4.1: Legislative Analyst's Assessment of State of California Systems Development Projects Which Have Experienced Significant Problems

- *Department of Motor Vehicles Database Re-design* - $40 million spent and little to show.
- *Department of Corrections Corrections Management Information System* - Continued schedule slippage and cost increases ($101 million is the latest estimate of project cost).
- *Department of Social Services Statewide Automated Welfare System* - Cost increases, delay and reduced net benefits (project cost now estimated at $800 million, to be implemented over 12 years).
- *Department of Social Services Child Welfare System* - Three years behind schedule with implementation difficulties anticipated to result in a change in project scope and/or a significant cost increase.
- *Department of Social Services Statewide Automated Child Support System* - Cost increase (from $140.8 million to $152.2 million) and significant schedule slippage.
- *Student Aid Commission Financial Aid Processing System* - Cost increases and contract management problems.
- *Board of Equalization Conversion to State Data Center* - Cost increases and delays.
- *Department of Health Services Vital Records Improvement Project* - Implementation delays related in part to cost concerns.
- *Secretary of State Imaging Technology* - New system failed and was abandoned.
- *Department of Housing and Community Development Mobile Home Registration and Titling* - Repeated difficulties over several years in efforts to implement an effective system.
- *Department of Transportation New Database Structure* - Delays and difficulties implementing a new database structure for departmental applications.

Source: Legislative Analyst, State of California, *Information Technology: An Important Tool for More Effective Government*(Sacramento, June 1994), 10-11.

Characteristics of Project Success and Failure

The literature and expert testimony set forth a number of factors that influence software implementation success and failure. According to the Gartner Group, successful projects have:

- Effective executive sponsorship,
- User involvement and influence,
- Manageable technology and complexity risk, and
- Good project management.

Other experts and the literature tend to agree with these characteristics of successful projects, although a number of other lesser factors can also contribute to project success, such as:

- Realistic expectations,

- Project ownership by users,
- Smaller project milestones,
- Competent staff, and
- Clear vision and objectives.

According to the literature and the experts we consulted, the characteristics of project failure are almost the inverse of the characteristics of success. The Gartner Group, in fact, regards the lack of the top four characteristics of successful projects as the top four causes of project failure. Gartner regards effective executive sponsorship as essential for project success. The Standish Group, on the other hand, found in its survey of information executives that the top reasons for project failure were: incomplete specifications, lack of user involvement, lack of resources, and unrealistic expectations, followed by lack of executive support.

The Statewide Systems Project was moderately successful.

While we think the Statewide Systems Project can be called moderately successful, it had some of the characteristics of a project at high risk of failure. For example, it was extremely complex and large; it tried to implement all its components at once; it had no single person in charge; it had incomplete specifications (necessitating significant midcourse changes); and it won approval in the Legislature and among state agencies based on unrealistic expectations.

Executive Sponsorship

The Statewide Systems Project had a significant amount of executive sponsorship, but it was weakened by high turnover among members of the steering committee. During the course of the project, the committee had a turnover of at least 10 members, and the state project manager changed as well. However, key high level managers in all of the sponsoring agencies remained in place throughout the project. In addition, the SSP management team had no single person to whom they reported, slowing decisionmaking. At various times, the Commissioner of Finance stepped to the forefront to champion the project, but overall, the turnover on the steering committee and the lack of "one person in-charge" put the project at serious risk of failure.

User Involvement

The Statewide Systems Project strongly emphasized user involvement. SSP involved more than 700 state employees in some capacity during the course of its development. Many state employees took mobility assignments to work directly for the project, and many more were released by their employing agency to spend time working on the project's development. Still, as we saw in Chapters 2 and 3, there are complaints that user involvement was not emphasized more. Users, and non-sponsoring agency steering committee members, felt that their input was heard clearly early in the development process, but less clearly as the project approached implementation.

Manageable Technology

SSP was an "unprecedented" project.

The client-server technology of the human resources (SEMA4) portion of the SSP project was new and untested anywhere in as wide an implementation as Minnesota planned.[7] It was as one consultant told us "an unprecedented project." The risks associated with large projects that have not been done before are especially high, according to the experts we consulted. Similar risks were faced by the Maxis project implemented by the Department of Human Services in 1991.

The result of this complexity was a large number of changes to the scope and specifications of the computer systems as the project proceeded. As we have seen, the changes contributed to higher costs than were originally anticipated. Overall, the ambitious and complex nature of the Statewide Systems Project put the whole project at a greater risk of failure.

Project Management

The Statewide Systems Project took steps to follow "best management practices" for systems projects by:

SSP followed many systems development "best practices."

- Having state managers as co-project leaders,
- Using steering committees,
- Having users review specifications,
- Having users involved in the design of the system,
- Utilizing a variety of change management techniques to aid in the transition between the new and old systems,
- Using a structured systems development methodology,
- Conducting internal and external risk assessments, and
- Having an active communications component.

One "best practice" that SSP did not completely follow was to complete the "re-engineering," or redesign, of state agencies' "business processes" before the project development started. Re-engineering is best completed at a project's beginning. Several of the other large systems projects we examined, such as the Department of Revenue's sales tax system and the Department of Labor and Industry's Daedalus project, did re-engineer their business processes before designing the system approach with good results. The Statewide Systems Project performed some limited re-engineering in the middle of the project's design stage. However,

7 Client-server technology, put simply, utilizes the user's computer (the client) to accomplish some of the information processing after downloading some information from the server (in this case a mainframe computer).

SSP staff performed the re-engineering work so late in the development cycle that its utility was limited.

As mentioned in Chapter 3, user training is a critical part of any successful software implementation. Training for the SSP project was problematic because the training and the training materials for the two MAPS components (accounting and procurement) did not always match the way the system actually worked.

Risk assessments can be beneficial.

Another best practice in systems development is ongoing risk assessment both by the project team and by external reviewers. The SSP had internal risk assessment procedures and also had an external risk assessment near the project's end at the Legislature's direction. Although risk assessments are somewhat disruptive to the project's development, we believe they offer a valuable outside perspective on the progress of development. Another project we reviewed, PCA's Project Delta, also used risk assessment to good effect. The Information Policy Office has recently negotiated a contract with three outside consultants that offers risk assessment services, making it easier for state agencies to engage their services. We believe external risk assessment is a beneficial part of any large systems project. We recommend:

- **The Legislature should require an external risk assessment as a part of any large systems project.**

The systems development literature, confirmed by several consultants we spoke with, recommends that systems projects should be done in phases or increments. Smaller scale projects generally result in less uncertainty about cost and development time. Many users of the systems also told us that, in hindsight, it would have been much better to have brought the systems on line in phases. Many state employees thought the phased implementation of the SEMA4 human resources component of the project worked smoother than the all-at-once "big bang" implementation of the accounting and procurement systems.

Another "best practice" recommended in the systems development literature and by the experts we consulted is to measure the benefits of the project after implementation. There was little assessment of the benefits of any of the Minnesota systems development projects we reviewed. We recommend that:

- **The state should carefully review the likelihood that benefits will result from a proposed project and require that the project sponsors establish measurement systems to evaluate the benefits after implementation. The Information Policy Office would be the logical place for this review to occur.**

Scope and Cost Changes

In examining the Statewide Systems Project and other large state systems development projects, we found that:

- **Changes in the scope and specifications of computer systems during development have been common in Minnesota state government.**

We found scope changes and change orders, in some fashion, on all of the projects we reviewed. We found that it is common that system requirements have not been specified precisely at a project's start. This lack of specification, in SSP's case, resulted in conflict between the state team and the consultant as they constantly negotiated what work was "in-scope" or "out-of-scope" of the original contract. On other projects the scope changed just due to the multi-year nature of the project; that is, there were changes in the program during the time period that the system was being developed. For example, on the Maxis project the eligibility for the Medicaid program became an issue for project developers several years into the project.

Experts told us it is impossible to accurately estimate costs at a project's start.

Problems also exist because it is difficult to estimate precisely the costs of a project at its beginning. According to the experts, final cost should not be estimated until after the system has been designed. The Gartner group told us that it is impossible to estimate accurately the costs of a systems project at the project's inception. A Gartner consultant told us that:

> [E]stimating an entire project at the very beginning of a project can very easily lead to variations between estimated and actual of 100 percent or more. Simply put, it is a practical impossibility for a project manager to estimate at the very beginning of a project what the entire project will take to complete, unless that project manager has done a number of projects that are exactly the same in type and scope.[8]

The Gartner Group told us that: "[The most capable information technology organizations'] best practice is to estimate costs for a project on a phase-by-phase basis, and they commit the funding for a project on a phase-by-phase basis, and they recommit the funding at the end of every phase when the estimates for the previous stage are done.[9] Experts from the Software Evaluation Institute, a federally funded institute to promote quality software development, gave us similar advice. Agencies should be held to cost estimates to design the system, the project should then be re-estimated for development, and the process should be repeated before implementation. This represents somewhat of a dilemma for the Legislative branch both in appropriating funds and in holding the Executive branch agencies accountable. One of the observations we found about the software development process in state government was that the development cycle is not synchronized with the appropriation process. Agencies are forced to begin the budget process before they are far enough advanced in the development to have a good idea of what the project will really cost. There is no easy solution to this problem in a government setting. On the Statewide Systems Project, development had to stop for four months because projected costs exceeded the appropriation. This was costly to the project's schedule and budget.

The software development cycle is not synchronized with the budget process.

8 Richard Hunter, Gartner Group, teleconference with audit staff, August 30, 1996.

9 *Ibid.*

In our view, the state of Minnesota should avoid computer development projects of this scope in the future. Projects that are developed in stages probably offer a greater chance of success, and smaller scaled projects present less uncertainty about costs. We recommend that:

- **In the future, the state should undertake large computer development projects only in more carefully planned stages, rather than trying to implement a large, multi-component project all at once.**

Future Benefits

As we saw in Chapter 2, some of the project's objectives might be met in the future if, for example, the EDI (electronic data interchange) module of the procurement system and the workers' compensation, recruitment, scheduling, and training modules of the human resources system are implemented. These modules would significantly reduce the need for paper documentation for many transactions. The Department of Administration plans to start a pilot test of the EDI subcomponent of the procurement system in January 1997, and the Department of Employee Relations has plans to implement the workers' compensation and training modules shortly thereafter.

Many system enhancements are still on the development "wish list."

Many of the enhancements to the system necessary for agencies to fully use the system are still on the development "wish list." We think that a continuing investment should be made in the systems in order to increase their functionality and increase future benefits. The sponsoring agencies should periodically assess needed improvements and report to the Legislature.

SUMMARY

We found that software development is rapidly evolving and that no one in the private or public sector does it consistently well. Scope changes and cost overruns are common. Successful development projects almost always have effective executive sponsorship, user involvement and influence, manageable technology and complexity, and good project management.