Not Being Kathleen Sebelius
Minimizing Risk in High-Risk, High-Stakes Technology Projects
By Allen Klein and Anne Ackerman

Introduction

Are you concerned that a major technology rollout under your watch will fail and that your Board of Directors will ask you the same questions members of Congress have been asking Secretary Sebelius? If so, there is good reason for your concern. More than 60% of all major system development and implementation projects experience serious technical issues and/or come in materially late and substantially over budget.[i] The failure of the rollout of the federal healthcare exchange (the "Exchange") is merely the most public illustration that a major technology rollout is a high-risk activity.

Politicians, pundits and the press are all trying to understand what went wrong and no one knows for sure. Simply put, none of us has sufficient access to the system, Department of Health and Human Services ("HHS") personnel, contractor personnel, contracts between HHS and its contractors (the "Implementation Services Agreements" or "ISAs") and documentation to perform a forensic investigation. Nonetheless, publicly available documents and statements point to several key reasons for the failure of the Exchange.

First, development and implementation of complex systems require the sponsor to determine priorities and balance potentially conflicting objectives, including, among others, cost, timeline, flexibility (including the sponsor's flexibility to make changes to its requirements as the system is being built), functionality, system quality and project risk. While HHS's priorities and objectives are not entirely clear to us, it is clear that minimizing project risk was not at the top of HHS's list.

Second, a common denominator of successful projects is the appointment and empowerment by the sponsor of a project executive with sufficient (a) expertise and experience to track project progress, manage the timeline, communicate status to all interested parties on a regular basis, and to make difficult tradeoffs (e.g., extend the timeline to ensure that there is sufficient working functionality to at least minimally achieve project objectives); and (b) gravitas to address problems when they arise, including making, and persuading the sponsor's executive management to agree to, major changes to the project timeline if required. Apparently, HHS failed to put in place, or empower, such a project executive. Public testimony suggests that the project executive failed to successfully manage the timeline and make the proper tradeoffs. Moreover, she didn't (or wasn't able to) extend the project timeline to complete testing. After all, who is going to tell the President of the United States to announce to the country that the rollout of his signature project must be delayed due to technical glitches?

In this article we examine the lessons that sponsors developing and implementing complex systems and seeking to minimize project risk can learn from the failure of the Exchange. If sponsors keep these lessons in mind, they may be able to avoid having their Boards of Directors ask "How did this happen?" and "Who is responsible?"

Lessons Learned

1. Use a Prime Contractor

When the Prime Contractor Model is used, work is outsourced to one contractor which may subcontract some of the work to other contractors. The prime contractor is responsible for its own performance and that of the other contractors working on the project and, except to the extent failure is attributable to acts or omissions of the sponsor, the failure of the project. Given that high degree of accountability, a prime contractor has every incentive to, and can be expected to, enter into subcontracts that enable it to effectively manage the other contractors' performance. Finally, many major IT contractors have the resources, expertise and experience to manage complex projects.

When the Multi-Contractor Model is used, work is outsourced to multiple contractors and the sponsor manages the contractors' performance.[ii] Sponsors have every incentive to manage multiple contractors well and, because prime contractors charge for the management function, mark up the contractors' charges, and include risk premiums, use of the Multi-Contractor Model can produce significant savings. Moreover, a sponsor can be an effective manager if it has resources, expertise and experience. However, all too often, a sponsor underestimates the level of resources, expertise and experience required, or elects to retain the role in an effort to reduce costs, notwithstanding the fact that it does not have such resources, expertise and experience.

There are legitimate reasons for using, or not using, a prime contractor depending on the sponsor's priorities. However, effective management of all contractors working on the project is a critical part of risk mitigation and, if minimizing project risk is a critical priority, the use of a prime contractor should seriously be considered.

2. Clearly Define Project Tasks, Roles and Responsibilities

In some cases project tasks can be better identified, described and allocated as the project progresses and the tasks are better understood. Moreover, in some cases, the best use of project personnel is to participate in the coding, configuration and testing of software rather than in the preparation and negotiation of detailed contract documents and post-contract written deliverables. However, without such a detailed description and allocation, important tasks can "fall through the cracks" and not be performed in a timely fashion, putting pressure on the project timeline. Moreover, such pressure is exacerbated when, as is often the case, the need to perform additional tasks causes a re-negotiation and/or re-pricing of the ISA. Accordingly, if minimizing project risk is a critical priority, sponsors should carefully consider the risks that arise from failing to (a) identify project tasks at a granular level and allocate responsibility for such tasks among the sponsor and the various contractors working on the project; and (b) include detailed descriptions of the project tasks and the allocation of them in the ISA.[iii]

3. Lock Down System Requirements and Functional Specifications Early in the Process

In many cases, the evolution of the system requirements and functional specifications may produce a system that better meets the users' needs than one where the system requirements and functional specifications are fixed early in the project. Whatever development and implementation process is used, it is inevitable that project personnel will (a) miss, or fail to accurately define, some system requirements and some required functionality; and (b) improve their understanding of project drivers and increase their level of expertise during the course of the project. Indeed, it is for this reason that the agile methodologies "welcome" changes even late in the project.

However, changes to the system requirements and functional specifications often have cascading downstream effects that may, among other things, change the scope of a contractor's work and/or the skill sets required by such contractor and require re-negotiation and/or re-pricing of the ISA, which can disrupt progress on the project. In addition, such changes often result in substantial rework and/or additional work, further disrupting project progress and preventing the orderly and timely completion of project tasks. Accordingly, project executives are called upon to balance the benefits from the evolution of system requirements and the functional specifications against the costs and risks arising from the downstream effects of changes. If minimizing project risk is a critical priority, serious consideration should be given to locking down the system requirements and functional specifications before beginning to prepare the technical specifications. Requirements identified after the lock-down would be implemented in a subsequent phase.

4. Have in Place and Adhere to a Reasonable Project Timeline

Failure to have in place or to adhere to a well-structured project plan may cause the sponsor and the contractors to rush sensitive tasks or eliminate important steps that are consistent with best practices. Moreover, such a failure often puts the sponsor in a position in which it (like HHS) must either (a) postpone elevation of the system to production; or (b) elevate the system to production without sufficient opportunity to subject the system to robust testing and remediate any deficiencies.

All too often sponsors set deadlines for the completion of technology projects that are not driven by the complexity of the project and the availability of resources. Rather, these deadlines are driven by factors that are extraneous to the project. While many, if not all, of these factors are legitimate and important to the sponsor in some respect, they may not be achievable or be achievable only if sensitive tasks are rushed or important steps are eliminated. Accordingly, if minimizing project risk is a critical priority, sponsor management should avoid including such deadlines in project timelines.

5. Don't "Go Live" Until Testing is Completed and Material Deficiencies are Resolved

Given the complexity of systems development and implementation work, initial versions of a system will almost never be free of deficiencies (including, in many cases, material deficiencies). For this reason, systems development and implementation methodologies typically provide for an iterative process pursuant to which the system is, or system components are, designed, built,

installed, subjected to testing, and fixed and retested until all material deficiencies that can be discovered during testing are discovered and fixed. This includes load or stress testing, during which the system is required to process transactions well in excess of maximum anticipated volumes. Load or stress testing ensures that, when the system is elevated into production, actual transaction volumes do not cause it to crash, which is what occurred with the Exchange.

As noted in Lesson 4, sponsors often set deadlines for project completion that are not driven by the project complexity and the availability of resources, and these deadlines may cause the parties to take shortcuts. However, if reducing project risk is a priority, the testing and remediation processes should never be cut short.

6. Sponsor Project Leadership is Critical

Finally, it is critical to appoint and empower a project executive with the experience, expertise and gravitas to fill the role as described above. Although in most cases lives are not at stake, the elevation of a system to production by a corporation or even a government department (e.g., HHS) is like the launch by NASA of a rocket carrying astronauts into orbit or to the space station. There is a great deal of work that must be done and many tests that must be performed before the rocket can safely launch. If there is a problem with a task along the way, that problem must be resolved promptly so that subsequent work dependent on the proper performance of the task can be performed properly, regardless of the personal or political consequences of addressing that problem. Final testing occurs just before and during the countdown and, if that testing is not complete or a problem is discovered, the project executive must delay the launch. The consequences of failing to do so are just too great.

The views expressed in this article are those of the authors and do not necessarily represent the views of, and should not be attributed to, DisputeSoft.

---

i *See* "The CHAOS Manifesto 2012: The Year of the Executive Sponsor," The Standish Group International, Inc., 2012. Available at: http://versionone.com/assets/img/files/CHAOSManifesto2012.pdf.
ii When a Multi-Contractor Model is used, management of the contractors can also be outsourced to a third party retained solely as a manager. However, a third-party manager will not have a contractual relationship with the other contractors and may not have sufficient control over such third parties to effectively manage them. Moreover, a third-party manager is not accountable for the work of the other contractors, and the other contractors are not accountable to the third-party manager. Accordingly, a third-party manager may not have the tools or incentives required to effectively manage the project.
iii The description of the tasks typically appears in a Statement of Work describing the contractors' responsibilities and in a Schedule of Retained Responsibilities describing the sponsor's responsibilities. Both of these documents are Schedules to, and incorporated in, the Implementation Services Agreement.

About the Authors

Allen Klein, Esq., is a globally recognized commercial, outsourcing and technology transactions lawyer. Mr. Klein is highly ranked by many top legal industry publications and rating services, including Chambers USA, Chambers Global, The Legal 500 US and Martindale-Hubbell. Before starting his own firm, Mr. Klein was a senior equity partner at a leading global law firm where he was a prominent figure in its top-ranked Outsourcing and Technology Transactions Practices. Mr. Klein can be reached at allen@kleincommerciallaw.com.

Anne Ackerman assists DisputeSoft on cases related to software failure. She has researched the legal issues regarding failed software projects and has helped to prepare documents in software failure cases. Prior to joining DisputeSoft, Ms. Ackerman was a co-founder of a midtown Manhattan general law practice where her work included litigation, arbitration, contract negotiations and settlement negotiations. She also spent 8 years as a legal recruiter, working closely with AmLaw 100 firms to negotiate the placement of individual partners as well as practice groups. Ms. Ackerman can be reached at aackerman@disputesoft.com.