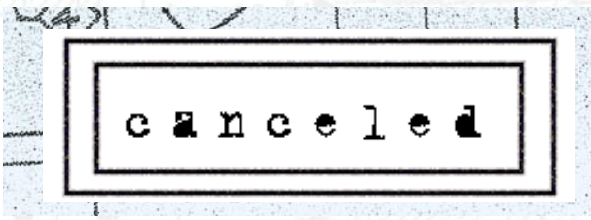
Kweku Ewusi-Mensah

Critical Issues in ABANDONED Information Systems Development Projects

What is it about IS development projects that make them susceptible to cancellations?

A medium-sized electronic distribution company with plans for becoming a billion-dollar company in the near future realized it needed to develop a new information system (IS) to cope with the increasing workload it was experiencing—a situation expected to continue. After more than two years of investing in information technology resources—including new IS staff and new management to develop the system—the company was forced to cancel the project. Later, they would start again with a new IS staff and management [3].



In 1988, the Intrico consortium was formed by Hilton Hotels, Marriott, Budget Rent-A-Car and American Airlines Information Services (AMRIS), a subsidiary of American Airlines (AMR). They teamed up to develop and market what was intended to be "the most advanced reservation system in the combined industry of travel, lodging, and car rental." Five years later, after a number of lawsuits and millions of dollars in cost overruns, the Confirm project was finally canceled amid bitter accusations from some of the top executives involved [10].

PC Week [11] reported in a study by the Standish Group that 31% of new IS projects are canceled before completion at an estimated combined cost of $81 billion. Furthermore, 52.7% of the projects completed are 189% over budget at an additional cost of $59 billion. These problems have led others to characterize the software industry as being in a state of crisis.

Our research shows that such cases are not isolated incidents; rather they occur with some regularity in companies of all sizes [3, 4]. Therefore, it is apparent that such cases are an industry-wide problem despite the significant progress made in IS development methodologies and tools since the early days of business computing almost four decades ago. These incidents lead us to ask: What is it about IS development projects that make them so vulnerable to such fiascoes?

This article highlights issues critical to aban-

doned IS development projects. What factors contribute to decisions to abandon IS development project? What characteristics do abandoned projects have in common? How are such abandonment decisions made, and what are the consequences of such decisions on companies and their IS staffs? What lessons, if any, should organizations derive from past decisions to abandon projects?

Our results indicate that the cancellation of projects can be attributed to a combination of several factors, including the following:

- *Project goals:* The lack of general agreement on a well-articulated set of project goals and objectives is the first critical issue.
- *Project team composition:* A weak or problematic project team.
- *Project management and control:* The management of the project, the lack of a measurement system to measure progress and identify potential risks in time to mitigate them, and the leadership responsible for making critical decisions at different phases of the project.
- *Technical know-how:* The level of expertise and experience, together with the relevant application-domain knowledge, of the team working on the project is not capable of the task.
- *Technology base or infrastructure:* The current information technology infrastructure within the organization was not carefully assessed and deemed satisfactory prior to undertaking the project.
- *Senior management involvement:* The active participation of corporate management in monitoring progress on the project and in making decisions at critical junctures is essential but was delegated or deferred to the technical experts.
- *Escalating project cost and time of completion:* Generally, these are symptomatic of more serious underlying problems calling for senior management attention before they get to the crisis stage.

We will use data from studies of abandoned projects, such as the electronic distribution company (EDC) and the AMRIS Confirm projects, to illustrate some of the pitfalls and management issues that often plague projects that end up being canceled. The incidents described here are replicated in many other companies that have experienced IS development project cancellations.

IS Project Characteristics

If we consider an IS project to be any information technology project intended to meet the information processing needs of an organization, one wonders what aspects of such projects make them different from non-IS projects that companies undertake. How do these aspects combine to make the projects vulnerable to cancellations?

IS projects are unique in that they require the intense collaboration of three groups of stakeholders, namely; IS staff, end users, and management. Hence IS projects are group-oriented activities, organized and executed in teams, and therefore subject to all the vagaries of group dynamics, interactions, coordination, and communication. The diverse backgrounds of the team members make the ability to communicate and coordinate the activities of the group an extremely important issue if the team is to work successfully.

Also, IS projects tend to be conceptual in nature. As Brooks [1] suggests, software "is pure thought-stuff, infinitely malleable" as well as "invisible and unvisualizable." In addition to the conceptual nature of the systems, there are often inherent risks and uncertainties associated with projects that are difficult to assess with any degree of reliability prior to the start of the projects. Those risks may include the large size of the project, complexity of the problem domain, project members unfamiliar with new technology, unstable information requirements, and difficulties in integrating different component systems into a composite system.

Moreover, IS projects are capital intensive, requiring the investment of substantial capital and human resources. IS projects, as sociotechnical systems, have become increasingly critical to the survival and well-being of companies. Indeed, the data suggest most canceled projects were systems the companies considered vital to their way of doing business in the Information Age [4, 5].

IS projects possess these unique attributes which make them susceptible to the problems outlined. Consequently, unless proper planning and control tools are used with a commitment to satisfying the

requirements of the technology and the organizational components of the system, together with satisfactory interactions among the three stakeholder groups, project cancellations may be distinctly possible, particularly for large-scale projects.

Factors Contributing to Project Cancellation

What factors associated with IS development projects have a negative influence on the projects' outcome, or tend to exacerbate development problems and thus contribute to the projects' eventual cancellation? The available data, obtained through surveys of canceled projects in Fortune 500 companies in the U.S., case studies of abandoned projects in Los Angeles and Orange counties in California, and published reports in the literature, identifies a number of factors as suspects [3, 4, 5, 10].

Data from the EDC project suggests that although there was a general agreement within the organization regarding the need for a new IS, there was no clear statement of what the new system's goals and objectives should be in order to satisfy the specific information requirements of the company. As one of the project's participants, the VP of marketing, indicated, "There were no firm established goals and, moreover, no real consensus on what those goals should be. . . . " [3].

It is imperative that firm goals and objectives be established for the project to guide the information requirements phase of the development process. Failure to satisfy this aspect of the project is likely to lead to fragmented efforts and lack of team focus in assembling the facts to guide the rest of the development [2]. Projects may still be canceled if the goals articulated as attainable in the original requirements document prove elusive, especially if the requirements change, as was the case in the Confirm project [10]. Therefore, it is equally important the specified goals be achievable and within the capabilities of the project team.

The development of large IS projects is the work of a team drawn from the ranks of IS staff, end users, and senior management, each with specific responsibilities to ensure the project's success. The disparate backgrounds and goals of the team members make it critical that clear lines of communication be established within the team, with lines of authority and responsibilities of the team members clearly enumerated. Curtis et al. [2] point out that communication in project teams is "necessary to resolve misunderstandings about requirements or design decisions among project members" and emphasize team-building as essential to translate individual talent into project success.

In addition, lack of structure and organizational purpose in the team's efforts, aggravated by lack of leadership and active interaction among all three IS project groups, is likely to lead to problems at later stages of the system development lifecycle. For example, AMRIS alleged in its lawsuit that the three user partners "made poor staffing assignments that harmed the Confirm project" [9]. Who becomes part of the project team is another important issue. For example, Brooks [1] and Curtis et al. [2] state the selection of good (that is, competent) staff is critical to project success. Equally critical to the project's success is how the project is managed and controlled from its initial stages to final implementation.

The technical dimension of any IS development project demands that some structure be imposed on the development effort to help guide the system to successful completion. The most obvious advantage of using the phased-lifecycle approach is to help the project team realize what the deliverables for each stage are and to know if they have been satisfied. The iterative nature of systems development notwithstanding, the phased-lifecycle approach has been instrumental in helping to manage and control the development of large, complex systems successfully. The consequence of failing to provide any systems development methodology to guide a project may result in the project's being tackled as another systems maintenance problem, as was the case in the EDC project. The VP of marketing remarked, "There was no planning developed for the project; no meetings were held to discuss technical issues on how to satisfy the computing needs;

It is common knowledge in the computer industry that projects are still over budget and behind schedule in far more cases than IS professionals and management find acceptable.

no management review meetings were held to discuss systems development efforts; and the project never moved beyond the level of what we wanted to do and could the computer do it" [3].

The CEO of AMR is quoted as stating in a letter to the other three partners, "The individuals to whom we gave responsibility for managing Confirm have proven to be inept. Additionally, they have apparently deliberately concealed a number of important technical and performance problems" [12]. It is obvious that project management failure created an environment where the alleged concealment of activities could have taken place. But even if the alleged concealment did not take place, the IBM review commission spoke to the "need of more critical review and immediate corrective action by AMRIS management. Not doing so will almost assuredly result in failure" [12].

Clearly, management has an important role to play in ensuring that proper management and control practices are adhered to and enforced in IS development projects. However, the responsibility should not be entrusted entirely to the technical personnel without adequate safeguards and oversight by management to ensure the technical personnel's compliance with accepted industry standards for reporting and dealing with problems uncovered in any phase of systems development. The problem is further compounded by the lack of good measurement systems to gauge and monitor progress on the project, and thus to identify potential problems early before they blossom into major ones.

The technical competence of the IS staff should not be underestimated. This was at the core of a number of canceled projects. Lack of familiarity with an information technology that is new to the firm or its IS staff is quite typical in many companies. This is what the new project director at EDC hired to work on the new systems development said about the canceled project: "The major contributing factor was that most of the people hired to work on the project had *no real prior experience* [emphasis added] or knowledge of new systems development activities. Most of their experience had been involved with maintenance-related activities. There was a general lack of know-how on [gathering] . . . user information requirements, and providing formal guidelines on how new systems development activities should be conducted. . . ." [3].

In addition to the technical competence of the team, the scope of the project is also a significant concern in abandoned projects. Projects that are excessively grand in scope tend to have built-in difficulties, higher risks, and levels of complexity that may frustrate even some competent teams. For example, the then president of AMR Information Services is reported to have indicated that "the task of tying together Confirm's Transaction Processing Facility-based central reservation system with its decision support system proved to be overwhelming. . . . We found they were not integrable" [7].

Also, the technical complexity of a project requires effective coordination of the various interacting parts to ensure successful completion. Curtis et al. [2] noted that the role of a competent project leader with "both application-domain knowledge and software knowledge" to guide and coordinate the activities of the various subgroups is vital to the project's success.

EDC embarked on its systems development effort when the integrated database technology necessary for the project to succeed did not exist in the company. The technology infrastructure available in a company needs to be critically assessed to determine whether it is adequate for the kind of systems development effort being considered. Projects should not be approved for development unless and until senior executives are fully satisfied that the company's technology base is adequate. Failure to get that assurance from the MIS management unduly increases the risks and uncertainties normally associated with systems development work to an unacceptable level, thus often laying the foundation for the project's eventual cancellation. The new VP of MIS (at EDC) hired to manage the MIS department described what he found when he joined the company as follows: "The mainframe computer had reached its capacity; they were running three different operating systems none of which was current; IS personnel were not up-to-date on technology, they were preoccupied with maintenance and enhancements to the existing applications. . . ," among a host of other problems [6].

In the Confirm project, the failure of the database to recover in the event of a crash was, in the words of the VP of Operations, due to the fact that "in the development of the DB2-based decision-support system, the company mistakenly implemented a version of Texas Instruments' Information Engineering Facility (IEF) computer-aided software engineering tool in which IEF generates its own database structure." Also, the VP is reported to have suggested that for Confirm's size, they "should have implemented a version of IEF in which the structure is dictated [because] . . . the system was so big that what IEF generated would have been impossible to maintain" [8].

It is vital for senior management to be supportive of a project and to provide the necessary resources to carry it out. However, if senior management fails to become actively involved in monitoring progress on the project or fails to inform itself on what is going on with the project on a regular basis, it is likely that small, unresolved problems will compound over time and eventually contribute to the project's termination.

The failure of senior management to request and enforce regularly scheduled management review meetings to monitor progress on a project is a major cause of failure. The consequence of this failure in the EDC project resulted in the following:

- Senior management had no way of knowing first-hand at what stage in the systems development life cycle the project was at any time or, even less, when the project would be completed;
- The new systems development project was treated as another series of fixes to the current system to satisfy the identified needs of the company; and
- Far more resources were expended on the project than could be justified before senior management realized the project could not be delivered as expected.

When active senior management involvement is lacking on a project, it helps create an environment where concealment of important performance problems can occur. A marketing director noted this concern in terms of "the inability of senior management to recognize they had a problem and delayed decision on it for so long. Pete should have been let go sooner and the MIS area should have been reorganized to deal with the project and its related problems earlier" [3].

The failure of senior management in the Intrico consortium to be actively engaged on a more frequent basis in the development of Confirm is cited by one AMRIS executive as partly to blame for the failure of the project. The AMRIS executive stated: "Confirm's 'fatal flaw' was a faulty management structure in which no one group had ample authority over the project. . . . You cannot manage a development effort of this magnitude by getting together once a month. . . . Had they allowed the president of Intrico to function as CEO in a normal sense and empowered their senior reps [to] work together with a common goal and objective, it would have worked. . . . A system of this magnitude requires quintessential teamwork. We essentially had four different groups. . . . It was a formula for failure" [9]. In this case, senior management was not only lax in managing the development; in addition, the lines of authority and responsibility for the success of the project were blurred, leading unavoidably to the passing of blame to others and failing to take decisive actions in situations where such actions would have advanced the course of the project. In addition, "the project's management was perceived as promising anything to keep the project moving" [12].

Executives should seize every opportunity to become fully informed and knowledgeable about all the major issues and aspects of the development process.

Senior managers are generally recognized as the beneficiaries of formal management procedures in the software development process and are expected to use the procedures to gain feedback and exercise control over the process. In the two cases cited, it is apparent that the requisite senior management involvement was absent, contributing inevitably to the projects being canceled for a variety of reasons, including cost overruns and delays in completion schedules.

It is common knowledge in the computer industry that projects are still over budget and behind schedule in far more cases than IS professionals and management find acceptable. Severe cases often end up being canceled. However, these facts may only mask more fundamental problems in the development process, such as those outlined here. Although information on a project's being grossly over budget and behind schedule will force management to act, the search for the underlying factors should begin elsewhere in the project's history.

In the Confirm project, the cost of the project was originally estimated at $55.7 million in April 1988 with a completion date of June 1992. It was revised to $72.6 million in September 1989. This trend in escalating project cost continued till the project was canceled in July 1992—after 3 1/2 years and $125 million in costs. During this period, Oz [10] reports, "the

client-partner teams met with the developer's representatives *just once a month* (emphasis added)." This apparent shortcoming was later cited by AMRIS executives as a major contributor to the problems leading to the eventual cancellation of Confirm.

In summary, the pattern that emerges from a synthesis of the data from our research and that of others is quite clear: IS project cancellation is a multidimensional and multifaceted issue with different interacting parts. Therefore, the cancellation of a project can be attributed to several factors detailed in this article. Which of these factors dominates the cancellation decision is contingent on the particular organizational and IS environments of the project.

In order to ensure successful development of IS projects, senior executives should be prepared to take specific actions to guide the process along and, when all else fails, to be prepared to cut the company's losses by terminating the project.

Minimizing Damage Due to Project Cancellations

Once the decision to cancel a project is made, it should be communicated sensitively to the entire project team, preferably by the executive directly in charge. Our data shows that in most cases, even if the team is aware of the possibility, communicating the termination decision directly to the project's participants will have a significant impact in reassuring the team of management's intentions and will help minimize the blame game and the recriminations that sometimes follow [3, 6]. The impact of the decision on individual careers should be minimized whenever possible so as not to create an atmosphere in which individuals would be unwilling to discuss with management what went wrong and why in the aftermath of the decision. Frequently, abandonment decisions are so badly handled by companies, culminating in the firing and/or demotion of some key IS staffers—as was the case in the two examples cited—that even those left unscathed feel intimidated and so refrain from voicing their opinions. This is often the basis for the "code of silence" that exists within the computer industry with respect to discussing project failures [6]. However, if we are to move beyond the current state of IS practice, we need to come to grips with the need to examine systems failures and shortcomings in order to gain insights that will significantly improve the technology and the art and practice of IS development projects in companies. We believe executives can play an important and constructive role in this learning process.

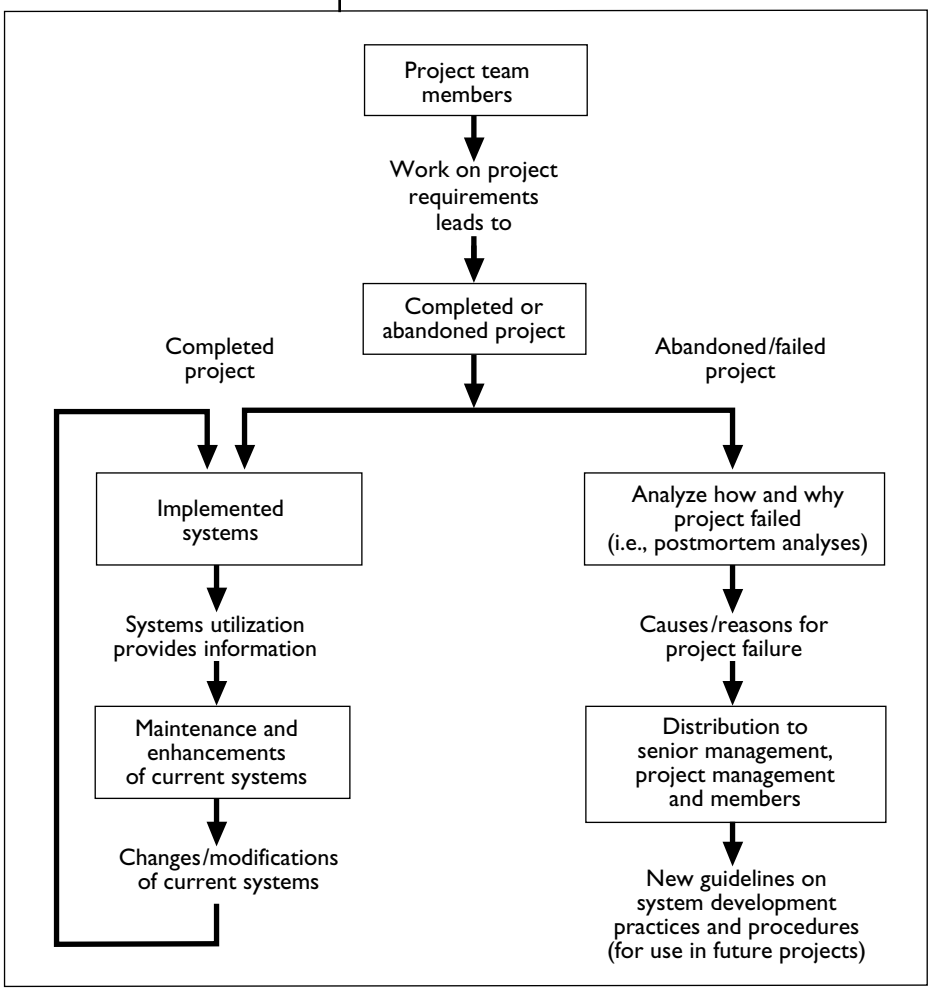


Figure 1. Learning cycle for IS project

We strongly advocate that soon after the decision to cancel the project is made, and even before it is

communicated to the team, executives should begin the process of determining what went wrong. A reputable senior executive within the company, or perhaps an outside consultant, should be appointed to examine all aspects of the development effort with a mandate to uncover the underlying root causes and reasons for the failure. Assurances of nonrecriminations must be fully extended to all project participants, who should be encouraged to talk to the person(s) in charge of the investigation with the sole purpose of helping the company benefit from the experience. This may prove valuable in future systems development work and will enable the company to recoup some of its investment in the abandoned project. The complexity and conceptual nature of systems development projects contribute to the difficulties of understanding "all the possible states" of the system that may in part contribute to "product flaws, cost overruns [and] schedule delays" [1]. Consequently, taking steps to understand the potential root causes of the project failure, and reporting on them in industry publications, will increase our knowledge and understanding of the systems development process and help minimize the recurrence of similar problems in the future.

We illustrate the type of organizational learning we feel is needed to obviate the frequent abandonment decisions experienced industrywide on IS development projects in Figure 1 [6]. It is generally accepted that even successfully implemented systems need to undergo maintenance to correct errors detected after implementation as well as to continue to adapt the system to meet current requirements. This kind of organizational learning is readily accepted throughout the information technology industry. We hope that executives can force the industry to break the code of silence surrounding failed projects and to share the knowledge and information gained through formal postmortems to the benefit of the industry as a whole. The model of the IS learning cycle presented is intended to be applied, in particular, to abandoned projects through formal postmortems performed on those projects.

Conclusion

Every IS development project must be viewed as both a learning experience and an experimental process, the outcome of which may not be fully predetermined because of the risks and uncertainties associated with it. Executives should seize every opportunity to become fully informed and knowledgeable about all the major issues and aspects of the development process. They should keep an open mind and not be so tied up in the process that cancellation does not become an option. Cancellation decisions should be handled sensitively to minimize damage to the morale and careers of all participants. Executives should follow the cancellation decision with a systematic examination of what went wrong and why as soon as possible, because participants' memories fade with time and because of the difficulties inherent in tracking down people after they have left the company. Executives must be willing to share any information and knowledge gained in the postmortem exercise with current and future project participants. Finally, the postmortem exercise should be made standard practice for all canceled projects in the company.

Hopefully, the knowledge, experience, and insights gained in the process will help diminish the frequency of abandoned development projects and improve the practice of information systems development throughout the industry. C

References

1. Brooks, F.P. Jr. No silver bullet: Essence and accidents of software engineering. *IEEE Comput. 20*, 4 (Apr. 1987), 10–19.
2. Curtis, B., Krasner, H. and Iscoe, N. A field study of the software design process for large systems. *Comm. ACM 31*, 11 (Nov. 1988), 1268–1287.
3. Ewusi-Mensah, K. Why IS development projects are abandoned: A diagnosis from user perspectives. Working Paper, CBA, Loyola Marymount University, 1994.
4. Ewusi-Mensah, K., and Przasnyski, Z. H. On information systems project abandonment: An exploratory study of organizational practices. *MIS Q 15*, 1 (March 1991), 67–85.
5. Ewusi-Mensah, K., and Przasnyski, Z. H. Factors contributing to the abandonment of information systems development projects. *J. Inf. Tech.* 9 (1994), 185–201.
6. Ewusi-Mensah, K., and Przasnyski, Z. H. Learning from abandoned information systems development projects. *J. Inf. Tech.* 10 (1995), 3–14.
7. Halper, M. Outsourcer confirms demise of reservation coalition plan. *Computerworld 26*, 31 (Aug. 3, 1992).
8. Halper, M. IS cover-up charged in system kill. *Computerworld 26*, 32 (Aug. 10, 1992).
9. Halper, M. Too many pilots. *Computerworld 26*, 41 (Oct. 12, 1992).
10. Oz, E. When professional standards are lax: The CONFIRM failure and its lessons. *Commun. ACM 37*, 10 (Oct. 1994), 29–36.
11. *PC Week 16* (Jan. 1995), 68.
12. Zellner, W. Portrait of a project as a total disaster. *Bus.Week* (Jan. 17, 1994).

KWEKU EWUSI-MENSAH (kewusi@lmumail.lmu.edu) is a professor of computer and information systems at the College of Business Administration, Loyola Marymount University, in Los Angeles.