

TROUBLED AND FAILED ERP/SCM/ eCOMMERCE IMPLEMENTATIONS: CAUSES, CAUTIONARIES AND SOLUTIONS

A White Paper

by **Richard Luetngen**
ThoughtWing

This paper was developed to provide general background to assist clients in understanding the causes of troubled and failed ERP (Enterprise Resource Planning), SCM (Supply Chain Management) and eCommerce (such as CRM, or Customer Relationship Management) implementations, as well as approaches to solutions. ERPs primarily automate the transaction processing requirements of financial, distribution and manufacturing business processes, SCM applications facilitate efficient interaction between customers and suppliers, and CRM applications support sophisticated interaction between companies and their customers.

We should note that many of the matters discussed in this paper can be equally applicable to large custom-development initiatives.

Please note as well that this paper presents professional opinions intended to apply generally and that clients must take appropriate care to evaluate them in light of their specific needs. ThoughtWing makes no representations, warranties or guarantees of any sort as to the applicability of the opinions presented in this paper to the specific needs of any client.

The paper is organized as follows:

1. Overview
2. Causes and Cautionaries
3. Re-Staging a Troubled or Failed Implementation
4. Conclusion.

1. Overview

Implementing ERPs/SCMs/eCommerce products may be the most complex and risky endeavors undertaken by a company. This is due primarily to the fact that the efforts touch, perhaps drastically affect, almost every business process within the enterprise. Change on such a scale is very difficult to accomplish, even in highly cooperative environments.

It is therefore not surprising given the complexity and the risk involved that such implementations sometimes encounter difficulties and sometimes fail. Difficulties alone can require expenditure of many millions of dollars that might have been saved had the difficulties been avoided. Outright failure can result in writing off the entire investment, which, for large companies, could run higher than \$100 million. Severe difficulties or failure also could poison the atmosphere for other projects or for re-staging of the implementation: political capital, once lost, can be very difficult to reacquire.

A well considered implementation plan in the right hands anticipates potential causes of difficulty or failure and provides means of addressing them should they be observed to develop during plan execution.

The remainder of this paper seeks to:

- Identify major causes of difficulties encountered during packaged software implementations, and what can be done to treat them so that they have minimum chance of actually arising; and
- Should the implementation already be in trouble or acknowledged to have failed, how can the investment made to date be salvaged to the greatest extent possible by re-staging the implementation.

2. Causes and Cautionaries

This section identifies our view of the major causes of derailed implementations, which are:

2.1 Incomplete Planning

2.2 Poor Change Management

2.3 Poor Expectation Management

2.4 Altered Internal or External Conditions

2.5 Problems with Consultants.

Each subsection describes the cause and why it precipitates problems, and what can be done to manage it as a set of symptoms before it becomes a problem.

2.1 Incomplete Planning

Any significant packaged software implementation project should be planned employing a structured methodology that specifies activities, tasks, deliverables and assigned human resources, in order to minimize the chance of missing key steps.

When inadequate attention is paid to rigor in planning the result can be a series of unpleasant surprises during execution of the plan, surprises that may:

- Require additional capital
- Require additional elapsed time
- Require more intense participation by internal personnel than had been anticipated by them.

With too many such surprises and their effects, political capital with both management and key personnel will disappear. With the loss of political capital often comes effort on participants' or management's part to find reasons for failure, or at least to change project management, which can be highly destabilizing and further fuel the difficulties.

Some key areas in which methodological rigor is particularly important in planning an implementation include the following:

- *Project Organization* -- often, in companies with highly collaborative cultures, project organizational structures are constructed that are too highly matrixed at all levels. Complex package software implementation organization structures should be formed with an eye to effectively controlling thousands of interrelated tasks, many requiring little or no broad consensus. Effective command and control in support of timetables and budgets is the actual organizational objective. When legitimate and delicate disputes regarding policy arise, those who resolve them should be highly placed. Distinctions should be made between substantive policy disputes and less important ones, and people at each level authorized to *decide* them.

Too highly a matrixed structure can result in serious slippage in schedules and budgets as time is wasted trying to forge broad consensus in all areas. Too much slippage of this sort and confidence in the team's ability to achieve its objectives will erode.

Tip: Spend enough time during planning to adequately describe authority at each level of the organization, and then spend the time necessary to sell the idea throughout the company, explaining the reasons for the rules.

- *Business Process Reengineering* -- Too often companies overestimate (or are too effectively sold on) the adaptability of products (ERP and SCM products particularly) to unique business process requirements.

When the need to either modify business processes to satisfy the constraints of the package, or modify the package code to satisfy the requirements of the process, becomes very great and was unanticipated, then inadequate requirements definition is usually the cause. This suggests that the wrong packaged software product was selected, which in turn usually reflects badly on those who did the selecting, who are often the same people who must address implementation problems. Consequently, it can be a very difficult political problem to address.

Yet, if left unaddressed the problem can add so much cost and time to the implementation that failure could seem to be a blessing.

Tip: During product *selection*, spend time identifying the business process areas that may contain unique elements, and particularly areas where the company has interest in improving processes. Do this with knowledgeable consultants from candidate vendors who can advise on product constraints, and who often can be engaged at cost because their companies are trying to close a software license deal. During product implementation planning, take the results of such sessions in to account when estimating related implementation tasks.

- *Report Definition, Interface Development and Historical Data Conversion* -- The time and effort required to perform these implementation activities is often very greatly underestimated. The impact of underestimating them is particularly severe because it usually is felt close to the end of the ordeal, when these activities, if not performed to schedule, could cause the roll-out to halt.

Tips:

- With respect to Report Definition, make clear to management that this is a moving target, since users will approach the team many times over the course of a long implementation to add reports they had not considered before or complicate already-requested reports. Often, refusal is not an option. Identify consulting resources that can perform the commodity tasks of report definition and development, and have them ready to appear as needed. In this way, schedules might be minimally affected by adding the short-term hands necessary to perform the work within the overall schedule. Do not give management hard estimates as to the cost of this activity; rather, provide a *range* of most likely cost.
- What is most often underestimated in Interface Development is the required *complexity* of the interfaces, rather than their number. Assume for estimating purposes that major interfaces require *some* re-design of the interfaced elements, which should result in safer *over*-estimation overall.

- Often, when converting historical data across years it is determined but was not anticipated in planning that the same data entities had different meanings depending on the year: software was enhanced and the entity took on a different meaning. This affects the complexity of the conversion, since the data must be leveled across years in order to be converted. Assume for estimating purposes that five percent of the data entities will require leveling, which should result in safer *over-estimation* of the activity.
- *Equipment Sizing* -- companies often underestimate the amount of computing power required to effectively support complex packages (ERPs and SCMs particularly, which are nothing if not hogs). If this happens, the need to upgrade processor, memory or disk at very short notice to support an implementation can be a very expensive and time-consuming surprise.

Tip: While modeling throughput characteristics and implications, also require software vendors to introduce you to at least one, preferably several, customer(s) with similar transactional volume and other sizing-sensitive characteristics. Ask why they sized equipment requirements as they did, and what experiences they had with sizing at implementation. Plan accordingly.

2.2 Poor Change Management

Change Management is the discipline that seeks to identify barriers to change in an organization and to formulate remedial programs that lower the barriers, thereby maximizing chances for success of the implementation.

Such barriers can include major cultural attributes, such as:

- fear of change because the organization has been stable for many years with employees possessing substantial tenure; or
- as simple as the outright refusal of a key individual to participate in the change; or,
- again as complex as the clear and unavoidable obsolescing of skills possessed by large numbers of employees who may leave but are necessary through transition to the new environment.

In some highly complex and sensitive cases industrial psychologists are used to provide specialized insight to assist the implementation.

The primary objective of *Change Management* is to identify the substantive issues that could delay or complicate the implementation, or cause it to fail, to identify the implications of each change barrier, and to develop programs to mitigate the risks and lower the barriers. Contingencies also should be developed in the event that the primary programs do not succeed, and measurement schemes also should be developed to determine if the programs are on track through implementation.

A secondary but important objective of *Change Management* is development of a Communication Plan whose intent is to communicate implementation status to all stakeholders in the implementation.

When little or no *Change Management* is performed as part of implementation planning or as part of the actual implementation, significant risk is incurred. Some change barrier could surface to trouble or derail the implementation with no plan in place to manage the risk with sufficient speed to be effective.

The most important advice that can be given with respect to *Change Management* is to take it seriously. Many do not.

Inability to understand that people who feel threatened by change can derail an implementation in ways that are not easily identified or countered can result in barriers that become so high that they *cannot* be countered except in a planned manner.

Often, internal personnel are too close to the underlying problems to be effective in identifying them. Independent consultants are best for this task. However, formulation of programs to manage the risk and lower the barriers absolutely require attention by senior internal personnel, guided by consultants, for only internal people possess the required organizational and cultural knowledge.

Tips:

- Communicate early with your people regarding the goals of the change, how those goals will be secured, what their participation will need to be, and what roles they will have in the new environment.
- Be ruthlessly honest with yourselves about what affect the change will have on your people. They are probably intelligent and, sooner or later, will figure it out themselves. If the affect is negative and you create the perception that you care so little about their interests that you have not thought the matter through, they may react with anger and do what they can to derail the implementation. They may very well panic and seek other employment.

If their active participation in the company's operations is key, at least through implementation, then such a reaction could pose serious danger to those operations.

2.3 Poor Expectation Management

This problem comes in two flavors, poor expectation *setting*, which involves the goals of the implementation, and poor expectation *management*, which is an on-going implementation challenge. If unaddressed, either can derail the implementation.

The expectation manager in either case is the prime company mover of the implementation or a consultant who acts as his surrogate. Those whose expectations are both set and managed are senior management, who approve and fund action, and who expect business benefits to accrue from the investment.

If, in an attempt to secure management backing of an expensive and difficult project such as an ERP, SCM or eCommerce implementation, the prime mover successfully presents potential benefits that are too aggressive, then expectations have been poorly set. Sooner or later, usually later, after a great deal of capital has been consumed, it becomes evident that the benefits will not be secured, despite what might otherwise be a successful implementation. In management's mind, this is a failure: an investment that did not produce anticipated benefits.

If this is discovered at a point at which management believes that halting the implementation can prevent further loss, then it can fail. This is particularly so if project participants have been complaining about the extent of required involvement, the extra hours and pressure, all normal characteristics of complex implementations but additional ammunition to an unhappy management who feel betrayed.

When this happens, poor expectation setting might be traced to a consultant surrogate or to a software vendor, who may be overzealous in pursuit of his own interests, such as placement of a consultant team to assist in the implementation or closing of a software license deal.

The most important factor in *managing* expectations is usually project scope. Initial expectations of the implementation may have been set reasonably, but over time additional goals may be added, one by one, without subjecting them to the same achievability tests as the initial objectives. In such cases a cardinal rule of any already-complex undertaking is violated: do not try to do too much at once; if you do, you risk loss of focus, or the undertaking may develop beyond your span of effective control, and you may fail to achieve *any* objective.

If management is led to believe that additional goals are reasonable, and then realizes at some point that they will not be achieved, the same problems described above could develop.

Tips:

- When considering initial expectations of a major action such as an ERP/SCM/CRM implementation, particularly if consultants or vendors set such expectations, management should take great care to assure that the rational underpinnings of arguments used are tested thoroughly.
- ERP/SCM/CRM program management should create a "phase awareness" in corporate management's collective mind, to better *manage* expectations. In other words, when additional goals are presented, test them to determine that they can reasonably be added to the program. If they cannot, consider creating subsequent implementation phases to deal with them.

2.2 Altered Internal or External Conditions

Changes in internal or external conditions can trouble or derail an implementation because such changes can invalidate the premises that justified the implementation.

Significant change in a company's management structure is a good example of altered internal conditions. Such change can fundamentally question the validity of the premises on which the complex software implementation was justified, because a different set of perceptions and values may rule. Additionally, new managements sometimes seek reasons to question initiatives taken by prior administrations.

While this may sound irresponsible when millions of dollars are at stake, it is nevertheless very real.

Another example of altered internal conditions is divestiture of a significant piece of the business that was to be automated by software, or acquisition of another company. Divestiture could call into question the cost/benefit value since a piece of the business that contributed the benefits is gone, while acquisition potentially introduces new requirements which could call into question the appropriateness of the software product selection, or the feasibility of the implementation plan.

Changed market or technology conditions are good examples of altered external conditions. For instance, a competitor might introduce a powerful new feature of customer service that threatens to bind customers more strongly to them, and the service is application software based.

In such a case, pressure could build to abandon the implementation to find and implement products that more effectively counter the competitive threat.

This is an example of a situation in which both market and technology conditions have been altered. A simpler example of altered technology conditions might be introduction of a new manufacturing approach that is automated by another ERP, which also could cause pressure to abandon an implementation and begin fresh with other products.

Tips:

- The potential for altered conditions sometimes can be anticipated, which could suggest re-scheduling the implementation for more stable times. For example, the management levels that normally approve major expenditures related to ERP/SCM/eCommerce implementation often are aware of imminent organizational changes before they happen. In such cases, it could be better to withhold approval until the new management team can take ownership of the effort.

- On-going threat analysis is a useful technique for altered external conditions. As part of implementation planning, potential threats such as those described above could be identified and tracked through the implementation. Contingency plans in the event of threat materialization could be developed and could be ready for use to mitigate the threat or change direction in an orderly manner that salvages as much investment as possible.

Clearly, long implementation periods magnify the risk of these problems.

2.5 Problems with Consultants

Large consulting/integration teams can cause problems in a complex implementation, including the following:

- Unwieldy program management structure, where span of control at critical levels is insufficient, often due to inadequate complex program management skills on the part of consultant management. This can be particularly damaging when the consulting team is substantially comprised of relatively inexperienced or junior personnel, who need a high degree of structure. Should this happen, expect to see patterns of missed deadlines, poor deliverable quality and inadequate translation of requirements into plans and deliverables, resulting in the need to continually revisit requirements and redevelop deliverables.
- Revolving door consulting management. The market for consulting talent is such that turnover in consulting firms, particularly among experienced technical supervisors and managers, is very high.

Over a long implementation period this can result in the loss of consulting resources in pivotal positions, which in turn can create great discontinuity in the effort and even cause it to fail. Again, this syndrome is particularly damaging when the consulting team has a high proportion of junior personnel.

- Consultants can sometimes have agendas that are not precisely aligned with their clients' agendas. (Imagine that.) For example, the larger the consulting implementation team, the greater the exposure to the consulting firm of unanticipated lessening of need for as many of them. This might result in a pattern of taking the more time-consuming and complex approach when a simpler one would suffice, so as to keep the maximum number of consultants optimally utilized. This can extend project duration and cost to the point where pressure builds to abandon the effort.

The ideal use of consultants in an implementation is a minimalist one. These threats are lessened tremendously in their likelihood of occurring and in their impact should they occur when consulting teams are small in number, very senior and used as facilitators and gray eminences rather than as providers of commodity services that would be provided better by internal personnel.

However, extensive/intensive use of internal personnel in large implementations, while highly advisable, is usually difficult, always painful for the personnel and sometimes not practicable. In such cases the following *tips* can be useful:

- Closely examine all consultant team leaders for competency and experience, retaining the right to reject any that do not measure up. If you do not possess the experience yourself to perform this vetting task, hire independent specialists who can do it for you: it is a very worthwhile investment.
- Treat potential revolving door consulting management very strictly. Inform the consulting firm at the first instance that you take the matter very seriously and that another such defection could be cause for termination of their involvement. And mean it. Require them to inform you of how they intend to minimize the discontinuity and assure that it have minimal impact on implementation duration, cost and quality.

“Meaning it” will require that you have alternative sources of consulting talent identified and pre-vetted.

- Do not abrogate your responsibility to test the recommendations of consultants. Rather than simply accepting recommendations regarding approach, require that consultants present the alternatives and provide the rationale for adopting the recommended alternative. “Sanity check” all recommendations in order to assure adoption of those that reflect *your* priorities and *your* interests.
- Develop a set of guidelines to determine how senior consulting personnel need to be in order to provide optimal value at a given activity. The more critical the activity to the success of the implementation, or the less commodity in nature the task, the more senior the consultant performing it might need to be. Not surprisingly, having a higher proportion of team members who are senior usually results in fewer consultants needed.

* * * * *

Any or all of the problems described above could cause severe software implementation problems or cause the implementation to fail. Should either potential outcome occur, sufficient capital could have been expended that a simple write-off may be unacceptable, or at least highly unpalatable. There exists an available option, which is to attempt re-staging of the implementation. That option is the subject of the remainder of this paper.

3. Re-staging a Troubled or Failed Implementation

A practicable re-staging usually begins with changing the faces. This usually means the consultants. In this manner, internal acrimony can be channeled away from the plans and people on whom success of the re-staging must depend.

This action might need to be taken for that reason even if the culpable faces belong to internal personnel, if their value to the company is too great to allow more direct action. However, if such is the case, the internal personnel at question should be finessed into less directly influential roles in the re-staging.

The next major step is to re-validate the premises, which is to say that management should determine that a successful implementation is possible if managed well, and worth the effort to achieve.

The remaining steps to a re-staging are the following:

- Determine what went wrong with the implementation;
- Remedy the problem(s);
- Determine the most sensible point of departure for the re-staging; this may require backtracking a significant distance from the point of implementation failure;
- Develop plans to manage the possibility of other major problems occurring;
- Approach all internal participants and prepare them for the re-staging; emotional preparation after a troubled or failed implementation is far more important than any other type of preparation, since much political capital will have been lost already and they likely will be very cynical;
- Re-commence the implementation at the determined point of departure.

While these are the actual steps that we recommend that clients take, and they are the steps we take when re-staging a troubled or failed implementation, they lack a great deal of context when presented as they are above.

For instance, if the problem that caused failure was poor planning, it is important that the corrected plans be presented to management in a way that explains why the implementation encountered difficulties, and how the new plans will avoid the same difficulties.

Similarly, selecting the point of departure for the re-staging is a delicate matter. It needs to be back far enough in the train of events that occurred to assure that work that contributed to the failure is avoided and redone. It also needs to be far enough back to reassure participants that management is serious in its commitment to success, not simply forcing a dubious re-staging at the expense (long hours, frustration, etc.) of participants. Yet, it also needs to salvage enough of the work performed to make a re-staging worthwhile.

Often, participants and management may perceive that a hard implementation failure was caused by a poor ERP/SCM/eCommerce product selection. This would suggest that the implementation needs to be scrapped and the cycle begun anew with a more informed selection process. On careful reflection, this indeed may be the case. However, this perception might also be due to exhaustion on the part of participants and the conviction that only a fresh start can succeed. If management is guided by this conviction without appropriate due diligence, this solution could turn out to be unnecessarily expensive.

When such hard failures are initially ascribed to poor product selections, management should determine:

- A. Was the failure due to extensive, unanticipated and undesired business process reengineering (BPR) due to product constraints?, or
- B. Was it due to modifications to the software so extensive that the project spun out of control?

In either case, management should consult independent product specialists to determine if alternate methods would be possible. Often, the product vendor or reseller is willing to provide such specialists, either at or below cost, or free of charge, in order to avoid the damage to their reputation that a failure would entail.

If too-extensive required BPR was the problem, it is probably more likely than not that product selection was the culprit, and that another product could better suit the company's requirements. In any event, expectations were not well set or managed, which is a problem that should be addressed as a part of any other action.

However, if too-extensive software modification was the problem, the opposite may be true. Often in an implementation, processes are not well developed to screen requested software modifications for value to the company. This may result in politically motivated modifications (modification to avoid the *inconvenience* of adjusting a business process to satisfy product constraints).

Just as often, the first politically motivated modification has the affect of opening the floodgates to many others (you accepted *his* modification, now you have to accept *mine*, too). If this were determined to be the case, setting up the appropriate screening processes and choosing a point of departure before the first unnecessary modification could provide justification for a re-staging. Effectively setting and managing of expectations also should be addressed in this scenario.

If poor change management was the cause of the trouble, an even more extensive and careful change management effort than usual will need to be mounted for the re-staging. This is due to the fact that much political capital has been lost in the failure, with participants who believe their interests (perhaps their livelihoods) have been actively threatened or damaged, and with management, who are decidedly underwhelmed by the prime mover's failure to anticipate the threat.

Problems with consultants are perhaps the easiest cause to rectify, for the following reasons:

- They can be blamed for the failure, channeling acrimony away from internal staff who must participate in the re-staging;
- They certainly can be replaced, or at least reorganized to operate more effectively in the re-staging; and

- They can be sued to recover monetary damages if their performance is less than professional and competent. Of course, likelihood of *recovering* damages depends on how contracts and other forms of agreements were written.

4. Conclusion

Given the high risk and cost of ERP/SCM/eCommerce implementations, it is clearly better to anticipate and effectively manage potential problems *before* trouble or failure occurs.

Doing so usually requires highly experienced professional help, at least to review plans and comment on potential gaps in appropriate safeguards.

Employing senior specialist consultants in this role allows leverage of a technique known as “pattern matching”. This technique requires professionals who have seen troubled and failed implementations, and who know what worked in a re-staging and what did not. Your problems fit a pattern, whose alternative actions and outcomes can be extrapolated from experience.

Actual re-staging of a troubled or failed complex package software implementation absolutely requires the deft hand of a highly experienced professional. Regardless of whom you choose.