

Risk Management for Laboratory Automation Projects

*R. D. McDowall
McDowall Consulting, Kent, United Kingdom*

Keywords:

risk management,
risk assessment,
risk analysis,
risk evaluation,
laboratory
information
management system,
LIMS,
laboratory
automation

This tutorial outlines some of the common risks that may be associated throughout the development and implementation of a laboratory automation project such as a laboratory information management system (LIMS) or another automation project. It presents a scheme for undertaking risk management to help assess and mitigate the degree of risk associated with each of these factors. In the case of high-risk factors, suggestions are presented to manage or help avoid the problem.

Risk management is an ongoing process. It begins at the start of a project and should be reassessed at intervals throughout the project to re-evaluate existing risks and to see if any factors have changed or new ones have emerged. (JALA 2004;9:72–86)

INTRODUCTION

There are many Laboratory Information Management System (LIMS) and laboratory automation projects that have collapsed or failed to deliver the expected benefits. Furthermore, surveys of information technology (IT) projects frequently show that many have run over budget, and nearly all projects end up with a changed specification from that originally described. When organizations used IT within laboratory areas in the 1980s and early 1990s,

any failures were either covered up or written off as “one of those things” that should be put down to experience. Since then, organizations are far more cost conscious and sensitive of failed projects. Although failure is a powerful learning experience, it is usually never incorporated into a corporate knowledge base for use by similar projects in the future.¹

A project is a single activity with a well-defined set of end results such as the successful implementation of a LIMS or another automation project. It follows a systems development life cycle (SDLC) from inception to completion.² A project does not exist in isolation and must often be coordinated or interfaced with other projects within the parent organization. Projects involve high levels of interdisciplinary communication and coordination with groups of specialists who are not usually used to such interaction. To aid the delivery of successful projects, project management provides an organization with the tools to plan, organize, implement, and control the activities necessary to achieve this.³ This tutorial is intended to be complementary to existing project management techniques and methodologies.

The complexities and multidisciplinary nature of projects require that the many tasks and deliverable parts of each be put together so that the prime objectives of performance, timescales, and cost are achieved when delivering the defined project endpoint. There is a relationship between these three factors that has to be traded off by the project manager. Some of these tradeoffs can involve risk management in varying degrees. This tutorial aims to discuss some general risks and the management of them to ensure a successful outcome of an automation project and is a revision and update of an earlier paper on risk management by the author.⁴

Correspondence: R. D. McDowall, McDowall Consulting,
73 Murray Avenue, Bromley, Kent BR1 3DJ, UK; Phone & Fax:
+44.(0)20.8313.0934; E-mail: R_D_McDowall@compuserve.com
1535-5535/\$30.00

Copyright © 2004 by The Association for Laboratory Automation
doi:10.1016/j.jala.2004.01.002

Background Reading

Although not directly referenced in this tutorial, the following books are useful background reading for computerized system failures (and the occasional success):

- *Crash: Learning from the World's Worst Computer Disasters*. Tony Collins with David Bicknell (1998), Simon and Schuster, London. ISBN 0-684-81687-3. The 10 deadly sins of computer failure are worth reading along with the case studies of many failed computer system projects—read and learn. However, as the authors note in the preface, the book has gone through two reprints and a second edition, but the book has not had the slightest beneficial effect.
- *Assessment and Control of Software Risks*. Capers Jones (1994), Yourden Press, Upper Saddle River, NJ. ISBN 0-13-741406-4. This book goes into more detail about project failure, but it is a more academic approach to the subject than *Crash*.
- *Patterns of Software Systems Failure and Success*. Capers Jones (1996), International Thompson Press, Boston, MA. ISBN 1-850-32804-8. Following the themes of his 1994 book, Jones looks at the reasons for successes and failures of software projects.
- *Managing Risk: Methods for Software Systems Development*. Elaine Hall (1998), Addison-Wesley, Reading, MA. ISBN 0-201-25592-8. A detailed approach for managing software development that can also be applied easily to automation and robotic projects.

Although the emphasis of these books is mainly on software, the principles outlined in them can be applied to other automation projects with little effort.

Project management is also important and plays a major role in determining if a project will be successful or not. Two key books that should be read are:

- *Software Project Management, A Unified Framework*. Walker Royce (1998), Addison-Wesley, Reading MA. ISBN 0-201-30958-0. Excellent book on managing software projects. Read and follow the principles and advice in this book and you won't need the next one.
- *Troubled IT Projects: Prevention and Turnaround*. John Smith (2001), Institute of Electrical Engineers, London. ISBN 0-85296-104-9. A good practical approach to project salvage and resurrection.

RISK MANAGEMENT

To overcome possible poor implementation or failure of a LIMS or laboratory automation project, risk management should be carried out at most stages of the system development life cycle. Risk management should be used in conjunction with project management techniques to manage the overall project. Therefore, identification of the risk factors should allow better management of a project and

identify specific areas where additional expertise or care should be taken.

Definitions

Risk is defined for the purposes of this article as the chance or probability of an event occurring that may alter the progress or outcome of a LIMS or laboratory automation project.⁴

The following definitions are taken from ISO 14971:⁵

- Risk management is the systematic application of management policies, procedures, and practices to the tasks of analysing, evaluating, and controlling risk. From Figure 1, this is the overall process that is the subject of this tutorial.
- Risk assessment is the overall process of a risk analysis and risk evaluation. This is the major subprocess and comprises analysis and evaluation of risk as shown in Figure 1.

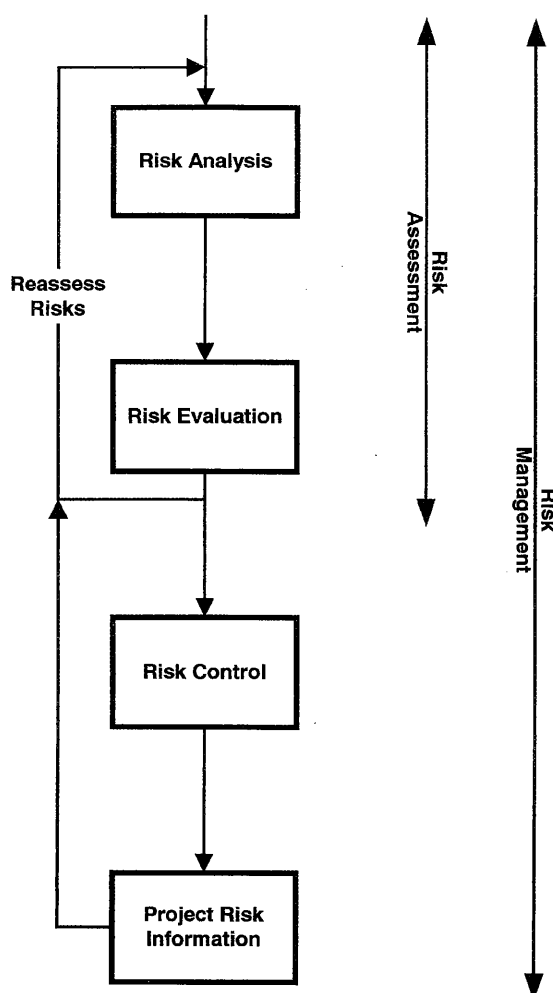


Figure 1. Risk Management Process Flow (Adapted from Ref. 5).

- Risk analysis is the systematic use of available information to identify hazards and estimate the risk.
- Risk evaluation is judgment, based on risk analysis, of whether a risk that is acceptable has been achieved in a given context.
- Risk control is the process through which decisions are reached and protective measures are implemented for reducing risks to, or maintaining risks within, acceptable levels.

Risk Management as a Process

There is little written in the scientific literature on risk management. Most risk analysis and management is intuitive and undertaken informally by project managers or project teams as a result of their experience or common sense. However, inexperienced individuals or project teams may have problems that could be mitigated or eliminated by the advance knowledge or experience of the common risks associated with LIMS and automation projects. The overall approach is shown in Figure 1.

Risk assessment and management is not a one-step operation but should be carried out at key stages of the SDLC of any project, and it is iterative. A project starts with a high degree of uncertainty and hence high risk. As it progresses, uncertainty in some areas is reduced but in others it can increase, hence the need for repeating the risk analysis and plan approaches to counter any newly identified risks.

Risk Analysis. At the top of Figure 1, the input should be at key stages of the project such as when a project definition document is written, system selection, or before implementation and rollout. From the project plan, the individual tasks can be identified for this portion of the work and analysed for potential risk factors. Using the knowledge and experience of the project team members, risk analysis can be

carried out and potential risks identified; alternatively, some of the risk management tables in this tutorial can be used (see Tables 1–6).

For any stage of the project, the risk analysis process is to:

- Identify the known or foreseeable factors or hazards that could pose risk to the project, i.e., risk factors that can impact a project can be organizational, financial, or technological.
- Estimate the risks associated with each factor. For example, what could happen if the factor occurred and what would be the impact on the project?
- Estimate the probability of the risk occurring.

Risk Evaluation. The evaluation process is very simple—it asks the question, Does the risk need to be mitigated or not?

If the answer is “no,” then the risk is accepted and nothing further is required. However, if there is a requirement for mitigation, then the risk moves into the next stage of the process: risk control. Typically, only high-risk factors will pass to the next stage; however, this decision depends on the criticality of the project in question.

Risk Control. Once the high-risk tasks have been highlighted, then it is possible to prepare plans and countermeasures to overcome the risk. “Risk Factors during Project Definition and Start-up,” “Evaluation and Selection of the System,” and “Risks Associated during Development and Rollout” discuss the risk and some of the possible approaches to mitigation. Note that it is not always possible to eliminate a risk, as this may be impossible or require too much effort; however, sufficient work needs to be done to ensure that the impact of the risk is managed and is acceptable.

The project manager then implements these approaches within the updated project plan. Milestones of the project

Table 1. Risk factors associated during project definition stages

Risk factor	Low risk	High risk
Project scope	Well defined	Undefined
Project deliverables	Well defined	Undefined
Benefits of the system	Quantified business impact	Undefined
System requirements	Straightforward, using standard components and technologies	Complex, using custom components and new technologies
Personnel providing application knowledge	Knowledge of both IT and user areas	Lack user area knowledge
Project members who have experience of business area	All project members	None
Status of documented procedures in user area	Complete and current	Nonexistent or outdated documentation
Number of computer applications that will interface with the system	None	Two or more computer applications
Status of these applications	Operational	Under development
Resource required	<5 resource years	>15 resource years
Time to develop the system	<12 months	>3 years

Table 2. Risk factors associated with system sponsorship and user commitment

Risk factor	Low risk	High risk
Project sponsor	Identified and has a strong user influence	Unknown
Attitude of user management	Fully support the project	Resistant and skeptical
Attitude of the users	Understand and support the project	Resistant and skeptical
Organizational maturity	Able to use the system effectively	Unable to understand rationale for the system or use it
Relationship of the project to the strategic IT plan	Included in the plan	Not included in the plan

can be identified and progress of the project reviewed against these; the same is true of risk factors. Once the progress of the project has been evaluated, this can be fed back into the system for an updated risk analysis. As can be seen, risk analysis is linked very closely with project management, and the two approaches should operate intimately.

Project Risk Information. As the project progresses, a body of information is collected. It describes how the project risk has been managed and how effective the approaches have been. It is important that this information is not forgotten or ignored. Reuse, cross-reference, and update the information; it can be used to feed back into the risk cycles as shown in Figure 1. Also, experience from failed projects, discussed briefly in “Learning from Failure,” should also be incorporated in the organization’s experience of automation projects as lessons to be learnt for the future.

Areas of Risk in the System Development Life Cycle

As laboratories depend so much on automation, and LIMS in particular, it is essential that management, users, and the project team should do as much as possible to minimize the risks to ensure a successful implementation. If an LIMS is not functioning effectively, then work within the

laboratories will be disrupted and productivity will suffer. However, when an LIMS is operating badly or not working at all, then the customer does not obtain the information and the reputation of the laboratory suffers.

The three main areas of risk within the SDLC are:

- Project definition and start-up (see “Risk Factors during Project Definition and Start-up”)
- Evaluation and selection of the proposed system (see “Evaluation and Selection of the System”)
- Implementation of the system (see “Risks Associated During Development and Rollout”)

Risk Factor Tables

The factors highlighted in the accompanying tables will allow continuing assessments of risk to be made of individual projects at various stages of the SDLC. As there is usually an underestimate made of the technical complexity of systems development, risk can never be eliminated totally from a project. However, the more that significant factors can be contained or contingency plans made to manage them at the start and throughout a project, then, the greater the chances of success. Over optimism, especially in the planning stage, is a chronic problem, resulting in projects being over time and over budget. Therefore, contingency time and money should be included in all plans.

Risk can be assessed as probable (high risk), possible (low risk), and improbable (negligible). Another approach is to assign to each factor a numerical value, say 10, for the highest risk, and 1, for negligible risk. Risk can now be evaluated on a continuum, which can be useful for an assessment of risk for a number of projects such as a prioritization exercise. However, in the tables presented in this tutorial, only high and low risks have been evaluated, as it is preferable, in the author’s view, to keep the scheme as simple as possible.

Table 3. Risk factors associated with the impact of the system on the organization

Risk factor	Low risk	High risk
The new system	N/A	Replaces an existing one A completely new system
Effect of the system on the IT department of the organization	Little change	Much change
Procedural changes required by new system	Little change	Much change
Organizational structures	None required or no changes to existing ones	Not considered
Policy changes required to support the new system	None	Extensive

RISK FACTORS DURING PROJECT DEFINITION AND STARTUP

The risk factors that may be encountered during the start-up phase of a project can be divided into four main areas:

1. Project definition
2. Sponsorship of the system and user commitment

- 3. Impact of the system on the organization
- 4. Management of the project

Each is discussed in more detail below. This section is the longest in this tutorial since, if this is not defined and agreed at the start, the whole project is worthless and has a low probability of success.

Risk Management during Project Definition

Outlined in Table 1 are some of the key issues that should be considered for risk assessment and management during the proposal definition and writing of any automation project. The areas of low and high risk (relating to possible and probable risk, respectively) are highlighted in the two right-hand columns for each factor. Every factor is considered below with suggested ways of managing the risk posed. The main effort in this phase of a project is commonsense management. There are no excuses.

Project Definition and Deliverables

Before starting a project, it is common sense to define the overall scope and tasks the new system will replace or carry out. It is important for all concerned that this is achieved in a project proposal or definition document. The content should explain, in non-technical language, what the system is to achieve when it is delivered. As this is the baseline for all future work on the project, it is an essential deliverable.

Moreover, the users and management must accept and underwrite the content of this document. The alternative is a poorly defined project with no focus. Thus, it is easy to introduce trivial or non-essential functions at the whim of an individual, which can waste time and effort, or worse, functions with little practical use. Moreover, a poorly defined project can select the wrong equipment or application to meet business needs.

Table 4. Risk factors associated with project management

Risk factor	Low risk	High risk
Experience of project manager	3 or more projects	No prior experience
Managing the project	Full time	Part time
Project team assigned	Full time	Part time
Experience of team members as a team	All worked together before	All are strangers
Number of times application/system implemented	More than once	No experience of this application
Team location	In one place	In several locations

The deliverables expected throughout the SDLC should be outlined at the start of the project to avoid not meeting user, management, and, where appropriate, regulatory expectations. Therefore, time should be spent discussing with the users, management, and especially the project team members the importance of any deliverables. Within a regulatory environment, these deliverables will form the basis of the quality development and validation of any automated system.

Defined Business Benefits. To avoid premature cancellation of the project due to budget cuts or management change, define and, if possible, quantify the business benefits that the system is anticipated to deliver. Senior management will need this information for project approval if over a preset spending limit, and continued management support is essential for longer-term projects. The advantages or direct benefit to the organization and the user should be outlined. To obtain the information to write this authoritatively,

Table 5. Risk factors associated with system selection

Risk factor	Low risk	High risk
New or non standard hardware or system software required	No	Yes
Team has experience of this application/system	Expertise available	Used for the first time
New language(s) required by the project	None	Used for the first time
Database used in the application	Well established in the organization	New in the organization
The system processing	Batch processing	Distributed system
On-line response required	>7 seconds 90% of time	2 seconds or less 90% of time
System availability	95%	>99%
Technology mix (database, network, etc)	Existing or simple architecture	New or complex architecture
Team's knowledge of the package	Previous experience	No knowledge
Organization has worked with the vendor	Three or more times before	Never
The package matches the system requirements	Well (little customization required)	Poor (major customization required)
Computer department involvement in package selection	High involvement	Not involved
User department involvement in package selection	High involvement	Not involved
Vendor reputation	Good	Poor or unknown

Table 6. Risk factors associated with system development and implementation

Risk factor	Low risk	High risk
System scope fixed and prioritized	Yes	No
Changes in organization or policy implemented	Yes	No
Change control procedures in place	Yes	No
Scope matches existing or proposed working practices	Yes	No
Documented roll-out plan available, detailing who, when and where	Yes	No
Sympathetic users identified for prototyping and testing	Yes	No
Quality Assurance involved for pre-operation audit and GMP advice	Yes	No
Development documentation to be produced identified	Yes	No
Performance of development system matches expectation	Yes	No
Contingency plan available if performance not good enough	Yes	No
Focused training planned and agreed with the users	Yes	No
Documentation available for users, backup and support staff	Yes	No
Assistance arranged for new users in period after training	Yes	No
Work schedules altered to cope with post training productivity loss	Yes	No
Plan for management of user expectations	Yes	No

involve experienced staff from the user environment as well as laboratory customers.

The business benefits are useful for defining, in another way, the target of the system to be developed. This definition can be of positive use in helping to make decisions concerning which functions are to be evaluated during selection and development.

System Requirements and Documentation of Current Procedures. At the start of a project, the system requirements are relatively vague and can hide a number of complex technical, procedural, and even organizational issues within them. However, even at this early stage, the requirements of a project and the operations carried out within the target laboratory should give an understanding of the degree of complexity involved. If the system to be implemented appears to be complex, a number of approaches to reduce the risk can be suggested:

- A complex system could benefit from a detailed systems analysis to understand the information and data inputs, internal operations, and outputs. This should give

a better understanding of the requirements and may help the new system support decision-making.

- As users often find it difficult to explain exactly what functions they do or are uncertain what they want the system to do, this may suggest a prototyping development with engineering or software projects. This approach should help the users debate and develop what they require and reduce the risk of the overall project, as the target can be scoped and the functions defined based on the experience of the prototype.
- A review of the working practices of the laboratory should also reveal if the processes undertaken should be changed prior to the introduction of a new system. One area where this is important if electronic signatures are to be used: an electronic process has to be defined, as the existing paper-based process will be inefficient.⁶

If current procedures are documented, these will help define the current practices and system. In contrast, poor, outdated, or no documentation can cause assumptions, perhaps wrong ones, to be drawn and requirements defined from incorrect information. Great care must also be taken not to assume that even if practices are defined, say in standard operating procedures (SOPs), that the current working practices in the laboratory match them. No assumptions should be made in this respect, as these documents may be major works of artistic fiction. Enlist the help of expert users to help define the current system.

It is essential to define current working practices, modify them where necessary, and map them onto a proposed system to help selection.

Knowledge of the Project Team Members and Users. The skills of all of the project team members should be assessed before the start of the project. Clearly, where the IT and laboratory automation members have been involved successfully with similar projects in the past, especially within the same area, there will a high degree of confidence and technical skill. In contrast, a team that is new and has little experience will require team building and technical training, ideally, before the project starts.

Equally, the experience of the user representatives working with the project team members may not be adequate to get a high-performance team working immediately. Therefore, some on-the-job training in project team membership may be appropriate.

An issue of major concern is the degree and depth of understanding and knowledge each member has in the other team members' disciplines. When there is none, a level of common understanding has to be developed. Equally important is the degree of knowledge and understanding the computer staff has in the laboratory. In the author's experience, this understanding takes much effort to acquire but is a worthwhile investment. A corollary is that carefully trained IT staff must be retained by the project; otherwise, momentum will be lost. To reduce risk before the project is

fully underway, management has the responsibility to ensure that the team is formed and has the required level of knowledge and understanding to do the job.

Interfaces to Other Systems. This aspect is not always identified in project proposals, but for integration to form an efficient organization, IT and automation projects should not exist in a vacuum but interface with each other to provide an integrated information environment. If the system is a standalone, no interfaces with other systems are deemed necessary, and development can proceed unhindered. In contrast, if the system has to interface with one or more systems, this adds complexity and risk. The interfaces, especially the data inputs and outputs, must be carefully defined and documented along with the responsibilities of who does what. This is acceptable if the systems already exist and are functional, as the interfaces are tangible.

Problems can arise if the interfaces are with proposed or partially developed systems, since interfacing with these applications increases the risk assessment. Now, additional time is required to identify where the projects overlap and how they should interface; liaison between projects is essential. Liaison may include sharing project team members, planning inter-project dependencies, or identifying the other project's deliverables.

Resources for Project Development. Smaller projects carry less risk. Therefore, to minimize the risk for larger projects, there are a number of measures that can be used to reduce the risk to acceptable proportions:

- Ensure formal project planning and monitoring with clearly identified deliverables and milestones, although this can be time consuming, and project team members could lose enthusiasm for the project. It is essential to focus members on the original aim of the project.
- Large projects can be broken down into smaller ones with discrete endpoints. These smaller projects, when complete and aggregated together, constitute the overall system. Alternatives are to reduce the original project scope and produce a minimum working system that can prove its effectiveness before additional functions are added or by using a phased development approach.
- Large projects developed over long time periods can cause problems in maintaining enthusiasm and user commitment. Moreover, any organizational changes could result in changes in sponsorship and less commitment or resources for the project.
- Large projects could justify the use of application development tools, such as computer-aided software engineering (CASE), to enhance productivity and decrease the time taken for various phases of the project. If the team has used this approach successfully in the past, this would be beneficial to the project. However, if this approach means introducing new technology, it may increase risk instead of reducing it.

Project Timescales. In today's organizations, a timescale of 2 to 3 years is a long time; in some companies, this can be the expectancy of an organizational structure or time between mergers. Therefore, if the timescale exceeds this, the project is unlikely to complete before the next change and is very unlikely to bring business benefit. Therefore, projects that last longer than 18 to 24 months are high risk. As described above, reduce the scope or break the project into a number of clearly defined phases. However, in doing this, it is essential that each phase provides defined and quantifiable business benefits of itself, or it is not worth doing.

Risk Factors Associated with Sponsorship and Commitment

Presented in Table 2, and described below, are the risk factors associated with sponsorship of the proposed system and the commitment of the users to accept it. Although presented here as factors associated with the start-up of a project, they must be reassessed during the project in the light of any changes in senior personnel, organizational rearrangements, and influences on the users. This is especially so in a project that has a long timescale or has been delayed.

Sponsorship of the Project. The best way of identifying a project sponsor is to ask the question, Who provides the money? Active sponsorship of large projects is important to persuade people to use the new system. A sponsor who is just a figurehead is a green light to wasting a large amount of money, as there will be few questions asked if the project fails to deliver. Senior management's understanding of automation and computer projects is usually based on two premises:

- The project is expensive
- It won't work

These perceptions must change.

If the project sponsor is not strong, political battles within the organization units under this individual can result in project delays due to a lack of decision or management commitment, especially in large projects. Therefore, to avoid these problems, procedures for resolving disputes should be instigated by the user management.

Attitude and Commitment of User Management. The attitude and commitment of the user management is essential for the success of any project. Apparent lack of commitment may indicate that they are unaware of the potential benefits that the system may bring or of plans to change the laboratory's direction. The manager of the user area should be briefed regarding the benefits that the system should deliver and its ability to enhance the business objectives of the specified area.

Winning the hearts and minds of user management is one option. If the project sponsor links a performance bonus to a successful project implementation, this adds the dimension of the wallet or purse to the equation.

