

***2160 Kharbanda on Project Management  
The Value of Analysis and Risk Assessment***

***By Daniel L. James***

***MBA Candidate at Rushmore University***

Submitted: December 2, 2001

Word Count: 7,909 net

**Executive Summary:**

The process of managing a project, large or small, is fraught with danger, especially for the project manager. Too many projects are undertaken without sufficient prior analysis of the needs of the ultimate customer, nor proper assessment of the anticipated risks. The odds of failure are enormous in any project. Knowing when to start, or when to cut-and-run, is as essential as understanding the features and cost elements of the project. This document utilizes a project the author is currently analyzing and addresses the functional and risk elements uncovered prior to the client's acceptance of the endeavor. It also provides a blueprint for managing and implementing the project, drawn from both personal experience and lessons from failed projects gleaned from various authors.

## Table of Contents

EXECUTIVE SUMMARY:.....	1
<b>TABLE OF CONTENTS.....</b>	<b>2</b>
<b>INTRODUCTION.....</b>	<b>3</b>
<b>PROJECT FRAMEWORK .....</b>	<b>4</b>
DISCOVER .....	4
DESIGN.....	6
DEVELOP.....	7
DOCUMENT .....	9
DEPLOY.....	9
FINAL LESSONS.....	10
<b>DISCOVERY OF “PROJECT RODEO” .....</b>	<b>11</b>
THE BIG PICTURE .....	11
FUNCTIONAL ANALYSIS .....	12
RISK ASSESSMENT .....	15
PROJECT METHODS & FEASIBILITY.....	22
<b>CONCLUSION.....</b>	<b>22</b>
<b>REFERENCES.....</b>	<b>23</b>

**2160 Kharbanda on Project Management**  
**The Value of Analysis and Risk Assessment**  
**By Daniel L. James**

## Introduction

In 1981, the Rand Corporation studied the phenomenon known as the “Concorde syndrome,” named for the infamous Concorde project to develop a commercially viable supersonic airliner. As history has shown, the Concorde is a technological marvel, but a commercial flop. Rand’s study found that 80 percent of typical “pioneer” (new development) projects failed to achieve their main objectives.<sup>1</sup>

It is axiomatic that everyone learns from mistakes, but there is no law I’m aware of that says they must be *our own* mistakes. The book, *What Made Gertie Gallop: Learning from Project Failures*, by Kharbanda and Pinto<sup>2</sup>, is the lead source of this Course Paper on project management. In a nutshell, the book discusses actual failed projects from around the world, from which valuable project management lessons are extracted. The authors set the tone in the Preface, “[T]hose who never made a mistake also never made a discovery.”<sup>3</sup>

I found that the authors validated many of the principles I have applied to project management efforts through my professional career, but I also benefited from new perspectives, especially related to multi-national projects. This was especially relevant because I was recently contracted to provide analysis and risk assessment services for a global information technology project by a client in San Antonio, Texas. Since I am under strict non-disclosure constraints, I cannot discuss strategic details nor reveal specific technologies associated with the project, however, a general description of the services and methods employed is possible. And I am able to divulge that the working title of the analysis is: “Project Rodeo.”

My intention is to show that the current client need not learn any lessons the “hard way,” but that the potential risks and failures associated with this project are identifiable and, thus, preventable. As Michael Eisner said:

“Everyone in business makes mistakes. If you didn’t take risks, you’d never get anything done. The sin would be if you made the same mistake more than once.”<sup>4</sup>

---

<sup>1</sup> Merrow, E., Phillips, K., Myers, C. 1981. “Understanding cost growth and performance shortfalls in pioneer process plants.” Santa Barbara, CA; Rand Corporation.

<sup>2</sup> Kharbanda, O.P. and Pinto, J.K. 1996. “What Made Gertie Gallop: Learning from project failures.” Van Nostrand Reinhold, New York, NY.

<sup>3</sup> Ibid, preface, p. x.

<sup>4</sup> Michael Eisner, Chairman and CEO, Disney Corporation, 1995; quoted in Henkoff, R. 1995. “Smartest and dumbest managerial moves of 1994,” Fortune, Jan 16, p. 59.

Through careful *Analysis* and *Risk Assessment*, the key prerequisites to any project, we can assist the client in avoiding mistakes and calculating the risks well in advance of any major financial commitment. But before we get into any specifics, we need to establish the framework of successful Information Technology (I.T.) project management.

## Project Framework

First of all, we must agree that there is no such thing as a “frozen specification” in the real world. Life and business are subject to change as the market changes or as technology evolves. Knowing this in advance forces a good project manager to adopt a reiterative approach to the design of the finished product or service. For I.T. projects, I have found the “Six Ds” to be a successful approach, consisting of the following deliverables: *Discover*, *Design*, *Develop*, *Debug*, *Document*, and *Deploy*. Figure 1 illustrates the sequence followed.

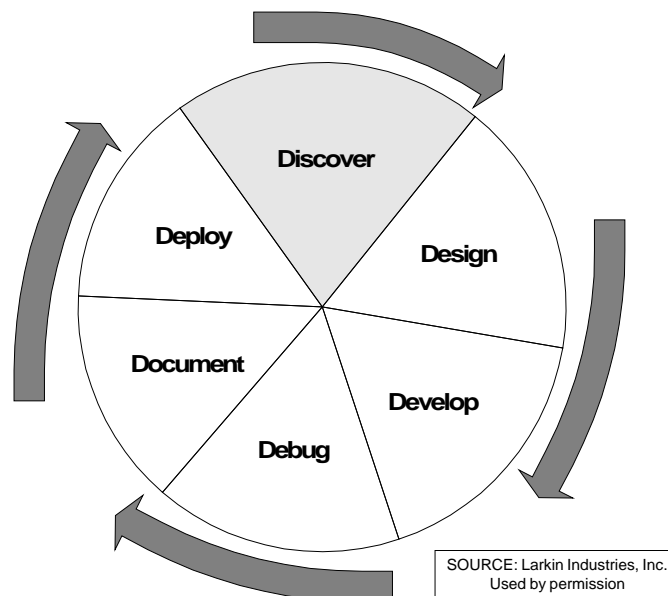


Figure 1. Reiterative I.T. Project Deliverables.

Not only should the entire project be managed in this sequence, but each sub-component, prototype, and sub-program should follow the same reiterative pattern. In the high-technology field, even after a product has been successfully deployed, it should be continuously revisited, re-analyzed and revised to stay ahead of the market. A discussion of each deliverable follows.

### ***Discover***

This is the analysis phase of the project, where we discover the needs of the client or end-user, in order to devise a blueprint for achieving the stated goals and to successfully complete the project. Is this project even feasible? The answer must be found during this initial phase. The more knowledgeable we become in the beginning, before large resources have been committed to

development, the more accurate and cost-effective the ensuing development will be. For this reason, Discovery is the most important deliverable.

For example, if one's goal is to develop an office building, it is not wise to start with a builder; one must first allow an architect to "discover" the needs and desires of the client, and to present a concept from which clear decisions can be made. A "golden" principle of project management for information technology development is: Never be in a hurry to start writing programs. The more time spent in analysis and design, the less time wasted rewriting programs later on. As Kharbanda says, "Research and development are twin pillars. One without the other often signals a sure recipe for disaster."<sup>5</sup>

In order to ascertain any preliminary cost estimates, the project deadlines, equipment and material requirements, and human resource needs must be estimated. In addition, we must determine the general areas of development in both software and hardware that are required to make the final solution workable. How are these elements analyzed? Kharbanda's approach is useful:

"Planning must always take into account the ultimate goal in terms of what is needed to obtain [client] approval. Once the final goal is set, one works backwards to complete the entire development plan."<sup>6</sup>

During discovery, we make our first attempt at a cost estimate, although this is usually understood to be a "rough guess." All parties to the project must understand that *legitimate* cost estimation can only arise when some basic design has been completed. Without such understanding and agreement, the project should go no further.

To arrive at a rough cost estimate, most American project consultants follow the procedure: design, estimate each resulting component, and add up the costs. Other cultures do it in reverse. For example, in Japan, they start with the market's likely selling price, then work backwards to the cost in order to calculate the project's internal rate of return (the Sony Walkman™ was developed this way). A "market price" is not always valid for non-consumer high-tech projects, but my experience indicates that, to approach the "real" cost of an I.T. project, I start with my normal cautious estimating, add more than the normal margins to cover contingencies, and then double it! Historically, this formula has kept my projects within a mere 10 percent error rate.

Additional rules-of-thumb I have used in I.T. projects where, after realistic time estimates have been presented, the client wishes to "compress" the project time: *For each one percent reduction in project duration, the project cost will rise by 1.75 to 2.0 percent.*

---

<sup>5</sup> Kharbanda, O.P. and Pinto, J.K. 1996. "What Made Gertie Gallop: Learning from project failures." Van Nostrand Reinhold, New York, NY, p. 265.

<sup>6</sup> Ibid, p. 10.

In assessing failed projects, Kharbanda nails this point home: "...[M]any problematic projects or outright failures were often handcuffed by utterly unrealistic scheduling and budgetary constraints at their conception."<sup>7</sup>

Finally, our analysis must identify the risks likely to be encountered during the project and suggest appropriate counter-measures to manage those risks. On this topic, Kharbanda spends a great deal of time. For example:

"...[I]n every field of activity, there is always an element of risk. The real test of good project management is to try to reduce these risks to the barest minimum and/or acceptable level through meticulous planning, execution, control, and above all, motivation of the entire project personnel."<sup>8</sup>

The principle issue that must be addressed as part of any project's risk analysis is the degree to which various forms of trouble are prevalent and predictable. The best approach is to expand risk analysis to take into consideration a greater degree of uncertainty. Other authors concur:

The number one cause of project failure was the inability or unwillingness of the project manager and team to engage in the 'What if?' troubleshooting guesswork before problems came to the surface.<sup>9</sup>

It should now be crystal clear that the "Discovery" phase is extremely important, and that considerable time should be devoted to this effort. In the process of analyzing all aspects of the project, the project manager will gain the additional benefit of discovering the political environments and individual personalities of all parties to the project.

### **Design**

Once we have done our homework, we fully understand the concepts of the project, and the client has approved the "big picture," the next step is to perform the necessary engineering and detailed analyses to prove the viability of the previously discovered solutions.

To continue our office-building example, after obtaining an architect's drawing, the next step is to enlist electrical, fluid, mechanical, and structural engineers to "flesh out" the concepts with precise drawings and specifications. The resulting documents should now be "portable," meaning that any builder could complete the project by following the engineers' instructions.

In I.T. parlance, this phase results in flowcharts, system diagrams, data models, user interface designs, and other technical information. The final component of the design phase is the *real* cost estimate for the project, which takes into consideration any new elements discovered during "engineering."

---

<sup>7</sup> Ibid, p. 22.

<sup>8</sup> Ibid, p. 81.

<sup>9</sup> Pinto, J.K. and S.J. Mantel, Jr. 1990. "The Causes of Project Failure," IEEE Transactions on Engineering Management, EM-37, no. 4, pp. 269-276.

This is also an important opportunity for the client to cancel the project based upon the detailed evidence and updated cost estimates obtained. Some clients may consider this option a sign of failure. I disagree, as does Kharbanda; to wit:

“...[C]ancellation is a realistic option. A company that is bold enough to cancel a project and acts early enough, will indeed cut its losses. Such a decision, however, requires as much, or even more, rigour [sic], analysis, and discipline as the decision to launch a new project. Project cancellation should not be seen as a failure, but rather as a key decision in the company’s strategy.”<sup>10</sup>

While this stage in the project is not a “point of no return,” the client must ensure by whatever means open to it that early estimates are sound, based on a completed basic design. Once Development has started, it is crucial that every change in scope is identified, scrutinized, recognized, authorized, and publicized, in order to answer the question: “must we have it?”

The client should clearly understand that the larger the change, the higher the impact on the fate of the project. Each change request should come with a complete and clear description, justification for the change, and the expected impact on both time and budget constraints. In most cases, a change in project scope will necessarily require additional time and budget allowances.

### ***Develop***

While the previous deliverables are typically documents and diagrams, the “Develop” phase appears more tangible – hardware is being purchased and installed; software is being written; the full force of the project team is onboard; but the potential risks are mounting. At this point, if either the Discover or Design steps have been skipped by the client, the risks of project failure are multiplied by an order of magnitude, including missed deadlines, cost overruns, missed features, inadequate testing, and so forth.

Assuming we can begin Development, often, it is necessary to first develop small-scale prototypes of the most critical technologies before committing to major systems, thus reducing risks along the way. The key is that the most crucial – and difficult - steps (discovery and design) have been completed, reducing any further questions about the scope and content of the overall project.

Any specification changes at this point, if not properly managed, further multiply the risks of failure. It is a scientific fact that engineers, if not time-constrained, will by nature continue to improve and perfect their designs. Project managers are often damned if they allow it, and damned if they don’t. In order to get the technology perfect, we sacrifice time. In order to accelerate a project, we sacrifice budget money.

The most important ingredient in the project management mix, at this stage, is communication. Kharbanda sums it up succinctly: “Effective communications are

---

<sup>10</sup> Kharbanda, O.P. and Pinto, J.K. 1996. “What Made Gertie Gallop: Learning from project failures.” Van Nostrand Reinhold, New York, NY, p. 323.

essential to the successful execution of any project.”<sup>11</sup> All parties must know what is expected of them, and by when those expectations are to be realized. The project manager must be acutely aware of both time and budget, and must enforce the quality of the project output according to (or exceeding the spec’s).

My favorite method of managing a complex project is to “show up and walk through it.” Problems are often not communicated effectively in memos and phone calls. By seeing and hearing them firsthand, problems can be resolved on the spot. One problem faced by project managers during development is external interference.

Interference, whether political or bureaucratic, encroaches on both time and budgetary resources, and is a primary cause of failure to meet deadlines or cost limitations. Statistically and historically, projects and politics do not mix, and any attempt to mix them often leads to disaster. When political expediency continues to outweigh pragmatic decisions, the end result is waste.

Projects often fail “due to management’s inability to refocus from the micro approach that consumes much of the project manager’s day, to a strategic, overall macro focus on the direction the project is taking.”<sup>12</sup> This is the project equivalent to rearranging deck chairs on the *Titanic*.

In order for oversight to remain objective while eliminating the need for daily “micro-management,” larger projects should be reviewed (audited) periodically, both before and during each major milestone, by representatives from five disciplines appointed by the client: research, marketing, strategy, operations, and finance.<sup>13</sup> This is the least intrusive and disruptive to the project team while providing the necessary feedback between the client and the project manager.

## **Debug**

Thorough testing and benchmarking are imperative, especially when developing new technologies. It is dangerous to assume technology that works well in one setting will work equally well in another, particularly when other variables in the equation are subject to change.

During this phase of the project, each component – hardware or software - is stress-tested against real-world environments, first as separate units and then in the context of the entire project. Benchmarking and field evaluations ensure that components are performing according to specification, or in line with customer acceptance. Only after thorough testing should any component, no matter how trivial, be signed off to production. Once all components have been tested individually, the integrated solution can be tested against client-determined performance needs and acceptable rates of failure.

---

<sup>11</sup> Ibid, p. 86.

<sup>12</sup> Ibid, p. 67.

<sup>13</sup> Lambrix, R.J. and Singhvi, S.S. 1984. “Preapproval audits of capital projects,” Harvard Business Review, Vol 62, March/April, pp. 12-14.

All “debugging” and system integration testing should be performed in such a way as to apply the real-time stresses of the production environment to the subject system. The entire project, and its management, will live or die by how effective its components survive this evaluation. Kharbanda agrees: “Monitoring and evaluation represent the core of any system whereby project management can be assessed in ‘real time.’”<sup>14</sup>

### **Document**

There are many categories of documentation. *Technical* documentation is used internally to memorialize the designs of each product, and to provide assistance in maintenance and troubleshooting efforts.

*User* documentation teaches the end-user, or customer, how to use the technologies and how to resolve problems encountered in the field (problems we should have determined in prior analysis and testing). All user documentation should be written for the lowest level of literacy amongst its potential readers. In other words, if 6<sup>th</sup>-graders are using the product, the documentation should be written so that 6<sup>th</sup>-graders can understand and benefit from it.

*Management* documentation is the medium for communicating overall project status, financial reports, resource requirements, and other high-level topics.

Documentation can take many forms, and tends to be most conveniently published online these days, however, printed documentation will often be required in the field where Internet connections are unavailable. For global projects, all user documentation should be provided in multiple languages. This further supports the requirement that complete Discovery, Design, Development and Debugging be performed *prior* to writing user documentation – we want to translate the text into multiple languages only *once*.

The final component of “Documentation” is the availability of live, technical user support and field training, if applicable, without which final deployment of the project should not be allowed to proceed. The client should be made aware of the costs of such support and training well in advance of deployment.

### **Deploy**

Now that all products and technologies have been researched, developed, fully tested and documented, the initial users have been trained, management has approved the results, and the project manager is prepared, the project is ready for deployment.

Depending on the scope of a project, a few components may be “phased in” over time, or many components deployed to a small geographic region before expanding the project to larger areas. The process of deployment is subject to vulnerabilities, requiring strict project management to reduce further risks.

---

<sup>14</sup> Kharbanda, O.P. and Pinto, J.K. 1996. “What Made Gertie Gallop: Learning from project failures.” Van Nostrand Reinhold, New York, NY, p. 55.

At this juncture, if there is anything we have “discovered” in the finished product that could be improved, the information is communicated to the client in preparation for a reiteration of the “Discovery” process, under a *new project framework*! With software projects, this reiteration results in a new “version” of the software, or a maintenance upgrade. In many cases, parallel projects – following the same reiterative pattern as we have established – can be used to develop new releases of the original product. The key is to actually have succeeded with the initial project before a larger scope or version is deployed.

In my experience, “deployment” is not complete until the customer is satisfied and all bills are paid. A project is no better than its ultimate use. Thus, a completed project that goes unused is a failure, and is indicative of a lack of prior analysis.

### ***Final Lessons***

From reading Kharbanda and other sources, and after my own project management experiences, I have learned some valuable lessons, which can be applied generally to projects large and small.

First, projects have four common dimensions:

1. They are constrained by a finite budget and timeframe to completion;
2. They comprise a set of complex and interrelated activities that require effective coordination;
3. They are directed toward the attainment of a clearly-defined goal or set of goals; and
4. To some degree, each project is unique.

Second, the “art” of project management can be condensed to two words: *cost containment*. That is, if you look at it from the client’s perspective. From a project manager’s view, we are subject to many forces and are constrained by four general “specifications:”

1. Time;
2. Money;
3. Performance; and
4. Customer satisfaction.

The last item, “customer satisfaction,” has not always received the importance it deserves by project managers, especially since it often requires additional time and attention maintaining close ties with and satisfying the demands of “external” clients. Some managers prefer to include the concept of “customer satisfaction” within the “performance” specification. However, after studying the lessons from past failed projects, I am convinced that customer satisfaction should be separate and equal to the other concerns.

Finally, business leaders should also learn from failed projects. The majority of mistakes identified from project disasters throughout the world can be attributed

to a violation of one or more of the seven lessons, described by Kharbanda<sup>15</sup> below:

1. Put professionals in charge and allow them to operate as the real champions;
2. Organizations must be flexible and adaptable to respond to their environment;
3. Relative decentralization, with short and rational decision-making processes, is a strong impetus for innovation;
4. Satisfactory coordination and integration between all parties is vital;
5. Roles must be well-defined with a shared view of organizational goals;
6. Conflicts should be resolved quickly within the organizational structure; and
7. In the ultimate analysis, the paramount factors were the end results and the “bottom line.”

### **Discovery of “Project Rodeo”**

Based on the name of our project, you have probably guessed that it has something to do with ranch animals. If so, then you have guessed right. In Autumn, 2001, I was subcontracted by a software development firm<sup>16</sup> (hereinafter “contractor”) to assist them with the “Discover” phase of a global project proposed by a local ranch solutions company (hereinafter “client”). My role in this phase of the project was to perform a feasibility analysis, feature analysis and risk assessment, and the contractor’s role was to provide resource requirements, timelines, and preliminary cost estimates, based on the results of the analysis.

What follows is a highly-condensed and “sanitized” version of the analysis and risk assessment; condensed due to the limited scope of this course paper, and “sanitized” to comply with a non-disclosure agreement between the parties. The objective is to describe the thought process required to perform such discovery, and to illustrate the importance of clear analysis prior to starting any information technology (I.T.) project.

#### ***The Big Picture***

There are dozens of euphemisms for the phrase, “understanding the big picture.” My favorite is “having a firm grasp of the obvious.” The first thing a project manager or analyst must do is establish credibility, and this is best accomplished by proving to the client that he or she (hereinafter “he”) understands what the project is supposed to accomplish as well as its scope. After spending time with the client and the contractor, performing interviews with technical experts, and doing some homework on livestock, I was able to describe the big picture for *Project Rodeo* as follows:

---

<sup>15</sup> Ibid, pp. 209-212.

<sup>16</sup> R&D Technologies, Inc., San Antonio, Texas. 210-543-8700

- **Objective** - The Client proposes to collect and track the data related to the existence, production, movement, medication, and consumption of livestock around the world. This task will involve certain technological innovations, combining the use of radio-electronic tagging, electronic vaccination devices, highly secure data transfer and storage methods, and the World Wide Web to facilitate tracking and reporting.
- **Goal** - The technological goal of this project is to effectively secure and manage livestock data obtained in the field, within various countries, at all stages of each animal's lifecycle. The solution will require the interfacing of multiple hardware and software components and programs, forming a reliable and fluid data stream, culminating in a central storage and processing facility in the United States.
- **Scope** - In order to support the global implementation of this project, the required technologies must be designed from the outset to be scalable. The solution must be able to ultimately provide data collection, storage, tracking, web access, security, backup and reporting for over 1.5 Billion animals at any given time. The solution must also adapt to varying communications capabilities within each target country, considering Internet access and connection speeds. Further, all user interfaces in the field must have multi-language capabilities, applicable to the respective countries targeted.

### ***Functional Analysis***

Once the overall project is defined, the next task is to analyze the functional characteristics of the proposed solution. In other words, how will the finished product operate? This is the crucial part of the analysis, from which all future details will flow. If the analyst doesn't get this concept right, then 1) the analyst has failed, and 2) the project will fail, if allowed to proceed.

Based on our prior analogy about the duties of an architect (page 5), our analysis should help the client to clearly understand the solution, or be able to visualize its functionality. Note that, at this level of analysis, it is not necessary to delve into highly-technical concepts (although the analyst should understand the components involved). After reviewing the functional analysis, the client must have 100-percent confidence that 1) the project goal has been solved, 2) the analyst truly understands the solution, and 3) all parties know how to implement the solution.

What I call "functional analysis" was presented to the client as a "concept of operations" followed by descriptions of both hardware and software components (and their "functions"). I have summarized these sections as evidence that the analyst could verbalize the solution:

- **Concept of Operations** - The business of livestock production involves a series of "events" which are monitored and managed for the highest possible efficiency. The ability to accurately record these events and track each animal being processed adds a significant amount of value to the

business. For example, if a rancher can reliably demonstrate that a marketable herd has been properly vaccinated, a higher return can be expected. Moreover, if the tracking of animal events can indicate the ideal breeding cycle of each female, the rancher can maximize offspring production. Strategically, accurate event tracking can open new markets to an entire country's livestock production, due to increased consumer confidence in the health of the animals being purchased. The most important tracked events are listed below:

1. **Tagging.** Usually performed at or soon after an animal's birth. This step is a prerequisite to the tracking of other events, since the "tag" contains an electronic serial number unique to the animal with which all future data is associated.
2. **Dosing.** Most animals will receive vaccinations and medications between two and six times per year. Each "dose" of a drug is a trackable event.
3. **Weighing.** This event is designed to help the producer detect an animal's most "marketable" condition.
4. **Moving.** Upon selling, transferring, switching corrals, slaughtering, or loading onto vehicles, producers can track each animal being "checked in" or "checked out" of the location. This is especially important to feedlot operators, where herds from many ranches are managed at a single location.
5. **Breeding.** The key to maximizing production rates is the ability to predict the optimal breeding opportunities for each animal, by accurately capturing breeding and calving cycles, allowing ranchers to "facilitate" breeding at peak times.
6. **Slaughter.** For animals that are bred for meat consumption, the processing of tags is important to link the butchered meat with the medication records of the animals.
7. **Death.** This event captures animal demise by accidental or natural causes, and, as such, is rarely a "planned" event.
8. **Other Events.** There may be other events that are found to be useful for tracking purposes, however, most would be simply variations of the events we've already identified.

The final steps in the operation occur when the "event" data captured on various field devices are uploaded to the Central Server in the United States, which processes the global data. During all such transmissions, encryption and data compression will be employed. Security and reliability is further assured by automatically "mirroring" the data at each server site, and by maintaining a final archive server at an offsite location within the United States.

- **Field Components** – This section analyzed each livestock data-collection or tracking device proposed for the project, describing both hardware and software modules, complete with pictures and high-level flowcharts. Since

virtually every such component is proprietary to the client, I cannot share any of the information. Suffice it to say that the analysis was many pages long and had some beautiful pictures and charts!

- **Connectivity and Transmission Components** – This section, too, contained many proprietary items. In a nutshell, each field device accumulates events until such time as it is connected to a “download station” for transmission to the Central Server via either phone line or the Internet. The analysis included plans to encrypt and decrypt the data, as well as compress and decompress it, thus securing it during worldwide transmittal.
- **Server Components** – This section analyzed the server requirements, identifying each hardware component and the software needed to operate each one. Many pictures were provided, illustrating the relationships between components, and the worldwide connectivity of the overall solution. Issues of server data redundancy as well as electrical redundancy were covered, along with data firewalls and environmental controls. Data protection schemes and procedures were also discussed.
- **Interface Components** – This analysis included hardware interfaces, software device drivers, and user interfaces (on-screen programs), including multi-language issues.
- **Business Components** – This section covered proposed business applications that both support the data collection efforts as well as offer extensive global, regional, and producer reporting capabilities. An excerpt from the overview of this section might be helpful, since it is very boring to read that you won’t be able to read about things.

“Other than internal system management and operations reports, most business applications in this project will provide dynamic, interactive reporting on the World Wide Web. The reporting required will depend on the User. For example, a government agency would be interested in countrywide reports and summaries, or specific production numbers for individual ranches within the country. Ranch managers would want to see results of their data uploads, summarize their production and medication records, and manage the records of individual animals. Slaughterhouses would want to look up the vaccination records of animals from multiple ranches. [The client] would use global data for reporting to agencies of the United States and/or the United Nations.”

- **Methods of Use** – Again, I am constrained from revealing any secrets. This analysis, however, illustrated with workflow diagrams exactly how field devices were to be used, and how each component in the data-chain would process and/or transmit the precious livestock data.
- **Data Flow** – This analysis concentrated on how the data actually flowed from one “layer” of the solution to another, until finally reaching its destination at the Central Server site. Each time the data is encrypted or compressed, transmitted or processed, it adds bottlenecks to the

performance. Methods of improving data-flow efficiency were also discussed.

- **Data Backup** – This section recapped all the ideas heretofore presented on the topic of data protection and offered additional suggestions on safeguarding the database.
- **Scalability** – This analysis provided guidance on a clear “growth path,” allowing the project to start small and to grow as each target country is added to the system, without causing obsolescence to the primary components. Items covered included additional field components, server memory, mass storage options, multiple server processing units, server processing speed, distributed server concepts, contracting third-party “server farms,” and server “clustering” (processing redundancy).
- **Security** – Finally, a thorough discussion of physical and virtual security issues was presented. Topics included the methods used by “hackers” (cyber-terrorists) and other forms of “thuggery,” and how to defend against their attacks. Damage due to internal threats was also covered, including sabotage, corruption, and incompetence. This section was purposely saved for last in order to act as a transition to the “Risk Analysis” section of the report.

### ***Risk Assessment***

Any systems or project analyst worth his salt can determine the requirements to complete a project under “ideal” circumstances, but not all project managers are prepared for Murphy’s Law.<sup>17</sup> Evidence for that statement is contained in the Rand Corporation study referenced earlier in this document (page 3), where 80 percent of projects failed. In my experience and observation, any pre-project analysis that has omitted a thorough risk analysis, has failed in one or all of its goals. Sometimes, just briefly listing all the potential risks – meaning that the analyst has acknowledged that certain risks do exist – is often enough to prevent them.

As the great Chinese general, Sun Tzu, stated in “The Art of War,” originally written more than 2,500 years ago:

“If you know the enemy and know yourself, you need not fear the result of a hundred battles. If you know yourself but not the enemy, for every victory gained you will also suffer a defeat. If you know neither the enemy nor yourself, you will succumb in every battle.”<sup>18</sup>

While the proprietary nature of the “how to” (functional) portion of the analysis of Project Rodeo has prevented me from discussing many details, I am not

---

<sup>17</sup> Murphy’s Law: “If something can go wrong, it will.” (Most software developers believe that Murphy was an optimist.)

<sup>18</sup> Clavell, James, editor. 1983. “The Art of War: Sun Tzu” Dell Publishing, New York, NY; edited from translation by Lionel Giles, 1910, Shanghai.

prevented from discussing the 22 potential risks discovered for this project. Especially since most of these risks are applicable to any global I.T. project. Therefore, I have summarized below the entire risk assessment that was presented to the client, which should provide some insight into the very “pessimistic” attitude required to create it. In some cases, I have omitted suggested solutions if they, too, involve proprietary methods or devices.

- **Risk Analysis** – Although the potential revenues from this project are quite large, once implemented in the primary target countries, the risks of failure are also substantial and should be considered. Most risks can be “managed,” not prevented, so the importance of addressing these risks early in the project cannot be over-stressed. The bottom line: we must be proactive, not reactive. Our risk analysis has uncovered the following issues that must be addressed in order for this project to succeed.
  1. **Animal Tag Loss** – Once an animal has been tagged, under the premise of this project, it should be tracked “for life.” Should the animal lose its RF tag for whatever reason, the data history accumulated thus far under the original tag number becomes “orphaned.” Simply retagging the animal with a new number does not automatically link the “lost” data with the new tag. A few lost tags here and there may not seem like a severe problem, however, when multiplied over thousands of ranches throughout the world, data integrity and accuracy can be negatively affected. A procedure should be developed to mitigate this risk.
  2. **Tag Uniqueness** – Tag numbers must be globally unique. The “risk” arises when live animals are exported to other countries with tag numbers that duplicate animals in the destination countries. A secondary risk arises if tags are recycled. Relational databases rely on unique “keys” to manage the data. Finally, user tags (used by human field operators) should also be unique, and their numbers easily distinguishable from animal tags, simplifying the programming of field data collection devices.
  3. **Equipment Failure** – What is the procedure when a [field data collection] device or other unit fails or jams in the field, is accidentally damaged, batteries expire, or power sources for recharging stations are unavailable? What if any such failure causes the device to lose any accumulated transaction data? What if the device is not adequately protected from moisture or dust? What if the port used to connect the device to a download station is damaged beyond repair, preventing attachment? What secondary strategies exist to manage these risks? These risks are magnified when considering the international scope of this project.
  4. **Human Failure** – The key to managing most risks associated with human error is to make a product that is completely intuitive and simple to operate, and to invest heavily in operator training. Since [client] personnel will likely not attend every data-collection site

simultaneously, the risks associated with inadequate human training are real. Preventative solutions may include training certifications and other incentives. Ensuring that field equipment is engineered to be “dummy-proof” is also helpful. Such a goal, however, will require extensive testing with real livestock in order to eliminate potential design or manufacturing flaws.

5. **Device Initialization** – Let’s suppose that the [field data collection] device is engineered flawlessly, and its operator is an expert. The unit is submitted to the ranch manager who attaches it to a download station. If the download fails to complete and the manager initializes the device before realizing the problem, then all previous data will be lost. Rigid procedures, both human and software-driven, must be enforced to prevent such a disaster. One preventative idea would be to design all data-collection devices with sufficient storage capacity to preclude the need to erase any prior data. Since all data is date-time stamped, extracting the “latest” transactions from among many others is relatively simple.
6. **Data Backup** – The data has been completely and accurately downloaded from a [field data collection] device, and the download station has an operating system error that both destroys data and “hangs” the machine. The operator has not yet performed a backup, or worse, had no intention of performing a backup. What is the procedure? Should the software *require* that a backup be performed before any data is uploaded to the server, or upon downloading each batch (not very realistic in practice)? As we have mentioned earlier, data integrity is most vulnerable in the field where adequate oversight and process enforcement is lax.
7. **Data Synchronization** – Since tag numbers are the “glue” holding all data together in the system, the risks associated with data synchronization are many. For instance, if tags are affixed to a herd of cattle, but no data is uploaded for the herd, and then a vaccination run is executed, the database will have hundreds of vaccination records without any “base” information about the animals. In such a case, the database would not know that these animals were in fact “cows,” nor the parentage of each animal. The resulting gaps in the data would tend to skew any meaningful reports related to the herd in question and its ranch. What procedures can be developed to manage this risk?
8. **Unauthorized Access** – The proposed system can provide for user identification and passwords before access is granted to the data over the Internet. There are certainly ways to circumvent such precautions. For example, a device operator other than the one whose [animal] tag was scanned may be vaccinating. Or, an unauthorized individual has logged into the “Rodeo” Website using someone else’s identification and password, and is marking

hundreds of animals “deceased.” What can we do to detect and manage these situations?

9. **Insufficient Project Resources** – One of the most common risks associated with any large-scale project is lack of adequate funding or resources to fully develop and/or test the systems prior to implementation. Whether the problem stems from under-production of field devices, under-equipping the server resources, or under-estimating the personnel requirements to implement the solution, the result is the same: failure. One strategy to minimize this risk is to implement a “bare-bones” initial solution, low on features and cost, and perfect its processes before moving forward. This would mean resisting any changes to the specifications of the project along the way, waiting until later project phases to add functionality (see “Scope Creep”).
10. **Scope Creep** – This is the term used to describe a project that begins with limited specifications, but that “improves” or “evolves” over time, resulting in features that far exceed the original scope or budget. The typical result of such a situation is failure to meet project deadlines and costs well beyond available funds. The cause of “scope creep” conditions is usually the lack of adequate analysis during the early stages of the project design. Since little, if any, analysis has been done, the developers have not accurately defined what the client needs, resulting in massive changes along the way. Another cause is poor project management, allowing ideas to continue to flourish well beyond the “sign-off” of the project specifications. The solution is to perform detailed analysis and extensive research prior to doing any development, and to then fiercely resist any changes to the specifications after-the-fact, without commensurate changes in budget and deadline.
11. **Technical Support** – [The client] will be required to provide both training and ongoing technical support for its worldwide customers. Considering language and geographical logistics for such an endeavor, the cost of these services will be substantial, especially in the early stages of the project. Regional telephone support centers would assist with any direct “handholding” required, and multi-lingual Web pages could provide additional support centrally. Failure to address technical support issues and training prior to implementation of this project would expose [the client] to “mission critical” risk.
12. **Adequate Testing** – Most people understand the need for testing a product before its release to market, however, risks resulting from insufficient or inadequate testing are potentially high with projects containing multiple components. As we have outlined in this document, in order for this project to succeed, many technology layers, including hardware and software components, must interact flawlessly and seamlessly. Any change to the specifications of one

component may adversely affect the entire data-flow, thus requiring extensive re-testing before results are certifiable. Therefore, it is imperative that adequate time and resources be allocated for thorough field testing of all hardware and software components.

13. **Version Control** – All data-collection devices in the field should be upgraded regularly with the latest firmware and software versions in order to ensure trouble-free performance and the least technical-support volume. The potential risks arising from non-conforming systems in the field are many, but primarily result in the failure to accumulate data. Wherever possible, all field software and firmware upgrades should be performed automatically at the time any device is attached to a download station for processing. In that way, the users will not need to manage the upgrades and consistent versioning can be attained in the field.
14. **Software Failure** – While software systems are to be written with built-in error controls, sooner or later a system will fail. The more system layers (“moving parts”) exist in the chain, the higher the risk of failure. We have stressed the importance of hardware redundancy and failsafe environments for servers, and likewise should address the need for safeguards for software at all levels. First, all software should be backed up regularly; it should contain extensive fault-tolerant error routines; it should properly communicate with other software components up and down the data chain so that timely failure alerts can be managed; and it should be serviceable remotely from the Central Server site. Upon experiencing software failure, minimizing downtime is only one concern – preventing data loss or data corruption is the primary mission.
15. **Global Deployment** – Implementing a system in the global marketplace requires the management of many risks, both tangible and intangible, resulting from political and cultural differences. Comprehensive analysis and adequate planning can minimize these risks, however, each country will pose its own unique deployment problems. Language is the first, and most obvious obstacle. Training, technical support, user support, and interfaces with local infrastructure providers will severely complicate the process. [The client] will be required to expend the energy and resources necessary to “part the political waters” well in advance of deployment within each target country.
16. **Client Performance** – Most projects are initially embraced with enthusiasm and beneficial intentions, some continuing to receive support until their successful completion. Other projects are subject to risks associated with failures by their “owners” to follow through with necessary resources or information. A project of this scope is no exception, and will require absolute commitment from [the client] throughout the development and deployment process. It is

understood that, due to circumstances beyond control, there may be delays or obstacles that would prevent optimum performance, however, [the client] must ensure that it remedies these situations and maintains its commitment to the project. Specific areas of concern may include failure to provide necessary information for successful development of the project; lack of communication between developers of hardware, software, and the project managers; failure to coordinate revisions of specifications between all parties; failure to maintain strict version control of software and hardware components; failure to perform under the financial commitments of this project; and failure to provide necessary human resources.

17. **Government Intrusion** – Historically, all governments at one time or another have suffered from incompetence, inefficiency, and/or corruption. In some cases, the problem has been temporary, in others, permanent. The potential risks of governmental intrusion – for whatever reason – is very real in a global project. Should data encryption safeguards be breached by a government entity, the resulting data could be modified to suit a political agenda or to cover up negative medication histories, and so on. Field devices could be confiscated at the whim of an over-zealous enforcement department, at best delaying the receipt of valid data, at worst preventing its collection. The transition from one “friendly” administration to a less accommodating one could be disruptive to the implementation or maintenance of the project. Efforts should be made to obtain governmental guarantees toward the project’s unfettered deployment and ongoing execution within each target country.
18. **Intentional Damage** – We have already addressed the risks of human error in the field, and accidental damage of field devices, but not the risks of intentional damage or sabotage of data. Since field operations are the most vulnerable in the chain of data collection, and security in the field not always assured, this risk can only be managed with the cooperation of both producers and government entities within the target countries. How well are field operators screened and what oversight is available to enforce the proper care and protection of the data collection devices?
19. **Natural Disasters** – This risk is universal. No amount of preparation can prevent loss due to a natural catastrophe, however, backup procedures and redundancies in the system can minimize any potential threat. In the field, however, a natural disaster could wipe out data stored in devices before that data has been extracted. Beyond waterproofing and shock-proofing the data collection devices, and establishing procedures designed to securely store them between each use, there are few measures available.

20. **Project Delays** – What provisions are in place to manage the risks associated with project delays (avoidable or otherwise), especially in the deployment to the first target country? Should technology vendors, political entanglements, or any other mission-critical element be delayed, the risk of failure to meet proposed deadlines is high. For example, if necessary investment capital needed by [the client] to move this project forward is delayed, certain hardware purchases, or the hiring of development personnel, could also be delayed, causing a chain reaction of failures. Most systems developers are familiar with “Murphy’s Law” and, therefore, they plan for the worst. [The client] should also have contingency plans available to manage events that would cause project delays.
21. **Acceptable Failure Rates** – Some failures are expected in any project. On a larger scale, failures may doom the project. Failures at the detail level may not. For example, if dosing transactions collected by a [field] device contain no date or time, is it a failure? Probably not, since the missing elements can be filled in manually to “salvage” the batch. On the other hand, if the medication data from an entire country is missing the drug [information], then the problem is indeed serious. How many incomplete transactions will constitute “failure?” Once that calculation is made, any number of incomplete transactions below that “failure” rate might be considered “acceptable.” The key to quantifying risks is knowing the threshold of failure. If a succession of failures remains “below the threshold,” then you may still need to fix the exceptions, but it is not yet a failure. As a further example, if the Central Server suffers hardware failure and is down no more than one hour per week, is that an “acceptable rate?” These calculations will determine the Quality of the resulting product. [The client] must define the “Acceptable Failure Rates” for this project, keeping in mind that higher standards often translate to higher costs.
22. **Initial Deployment** – [The client] desires to deploy this project to its first target country [in] 2002. This means that all field data collection devices and animal tags must be designed, manufactured, programmed, and field tested; server technologies purchased, installed, programmed and secured; regulatory issues resolved; documentation prepared and translated; users trained; and full testing and deployment completed [by mid] 2002. This is an ambitious goal, requiring very tight coordination of development tasks, and narrowing the margin of error to near-negligible levels. The risks of failure are high – a single delay in the development or procurement of any component in the system, or the miscommunication of any specifications or client needs, could result in failure to meet the given deadline. In order to minimize these risks, strict project management and coordination of all required information should be enforced.

### ***Project Methods & Feasibility***

After the component specifications and risks have been analyzed, the next step is to identify exactly how the project will be managed. This includes development and testing methods, benchmarks for performance evaluation, milestone dependencies, resource requirements, estimated costs, and a preliminary project schedule. Since most of this information is proprietary, there is little to sample here.

Finally, a “feasibility statement” – or analysis conclusion – is offered, often containing input from all concerned parties to the analysis.

Most of these analysis sections were (or are being) authored by members of the primary contractor group, and thus, are outside of my control (and definitely outside the scope of this course paper). However, I can share the portion of the “feasibility statement” that I contributed:

- **Project Feasibility** - Based on considerable discovery and analysis, we have determined that [the client's] “Project Rodeo” is technologically feasible. Considerable risk management will be required, pursuant to the potential risks itemized herein, specifically related to the proposed initial deployment by [mid] 2002. Without such short-term deadlines, future deployments to other countries should pose lower risks of failure. The costs of implementing the project are, of course, dependent on the level of risk the client is willing to endure, and are inversely proportional to the time allotted to meet the first deadline. The features available for the initial deployment must also be limited due to the short-term deadline, but should be adequate to “prove” the concept in the field.

### **Conclusion**

In this course paper, I have attempted to illustrate the application of my study and experience in I.T. project management. The actual analysis for *Project Rodeo* was beyond 40 pages of text and graphic documentation. Putting this effort into perspective, the complete “blueprint” for this project will likely be 200-300 pages of detailed design and development specifications. While most analysts will agree that, “size isn’t everything,” the important issue is that the project will have been thoroughly designed before any large capital outlays are made.

The true value of analysis and risk assessment efforts, therefore, is to help the project manager contain costs, prevent disasters, meet deadlines, and satisfy the ultimate customer. Without either effort, the project is very likely doomed to repeat history. The lessons from the failed projects in Kharbanda and Pinto’s “What Made Gertie Gallop,” clearly indicate that most project managers do not learn from history.

As for this I.T. professional, I would much rather learn from others’ mistakes than my own.

## References

Kharbanda, O.P. and Pinto, J.K. 1996. "What Made Gertie Gallop: Learning from project failures." Van Nostrand Reinhold, New York, NY.

Merrow, E., Phillips, K., Myers, C. 1981. "Understanding cost growth and performance shortfalls in pioneer process plants." Santa Barbara, CA; Rand Corporation.

Michael Eisner, Chairman and CEO, Disney Corporation, 1995; quoted in Henkoff, R. 1995. "Smartest and dumbest managerial moves of 1994," Fortune, Jan 16, p. 59.

Pinto, J.K. and S.J. Mantel, Jr. 1990. "The Causes of Project Failure," IEEE Transactions on Engineering Management, EM-37, no. 4, pp. 269-276.

Lambrix, R.J. and Singhvi, S.S. 1984. "Preapproval audits of capital projects," Harvard Business Review, Vol 62, March/April, pp. 12-14.

Clavell, James, editor. 1983. "The Art of War: Sun Tzu" Dell Publising, New York, NY; edited from translation by Lionel Giles, 1910, Shanghai.

R&D Technologies, Inc., San Antonio, Texas. 210-543-8700

Larkin Industries, Inc., Carson City, Nevada. 775-883-9032