



## Sample Research Paper

**To purchase a newer version of either the *CHAOS Chronicles* or *CHAOS: A Recipe For Success*, please [click here](#).**

For more information on project failure and success click here for [Unfinished Voyages!](#)

### CHAOS

*"The Roman bridges of antiquity were very inefficient structures. By modern standards, they used too much stone, and as a result, far too much labor to build. Over the years we have learned to build bridges more efficiently, using fewer materials and less labor to perform the same task."*

Tom Clancy (The Sum of All Fears)

---

## INTRODUCTION

In 1986, Alfred Spector, president of Transarc Corporation, co-authored a paper comparing bridge building to software development. The premise: Bridges are normally built on-time, on-budget, and do not fall down. On the other hand, software never comes in on-time or on-budget. In addition, it always breaks down. (Nevertheless, bridge building did not always have such a stellar record. Many bridge building projects overshot their estimates, time frames, and some even fell down.)

One of the biggest reasons bridges come in on-time, on-budget and do not fall down is because of the extreme detail of design. The design is frozen and the contractor has little flexibility in changing the specifications. However, in today's fast moving business environment, a frozen design does not accommodate changes in the business practices. Therefore a more flexible model must be used. This could be and has been used as a rationale for development failure.

But there is another difference between software failures and bridge failures, beside 3,000 years of experience. When a bridge falls down, it is investigated and a report is written on the cause of the failure. This is not so in the computer industry where failures are covered up, ignored, and/or rationalized. As a result, we keep making the same mistakes over and over again.

Consequently the focus of this latest research project at The Standish Group has been to identify:

- The scope of software project failures
- The major factors that cause software projects to fail
- The key ingredients that can reduce project failures

## FAILURE RECORD

In the United States, we spend more than \$250 billion each year on IT application development of approximately 175,000 projects. The average cost of a development project for a large company is \$2,322,000; for a medium company, it is \$1,331,000; and for a small company, it is \$434,000. A great many of these projects will fail. Software development projects are in chaos, and we can no longer imitate the three monkeys -- hear no failures, see no failures, speak no failures.

The Standish Group research shows a staggering 31.1% of projects will be canceled before they ever get completed. Further results indicate 52.7% of projects will cost 189% of their original estimates. The cost of these failures and overruns are just the tip of the proverbial iceberg. The lost opportunity costs are not measurable, but could easily be in the trillions of dollars. One just has to look to the City of Denver to realize the extent of this problem. The failure to produce reliable software to handle baggage at the new Denver airport is costing the city \$1.1 million per day.

Based on this research, The Standish Group estimates that in 1995 American companies and government agencies will spend \$81 billion for canceled software projects. These same organizations will pay an additional \$59 billion for software projects that will be completed, but will exceed their original time estimates. Risk is always a factor when pushing the technology envelope, but many of these projects were as mundane as a drivers license database, a new accounting package, or an order entry system.

On the success side, the average is only 16.2% for software projects that are completed on-time and on-budget. In the larger companies, the news is even worse: only 9% of their projects come in on-time and on-budget. And, even when these projects are completed, many are no more than a mere shadow of their original specification requirements. Projects completed by the largest American companies have only approximately 42% of the originally-proposed features and functions. Smaller companies do much better. A total of 78.4% of their software projects will get deployed with at least 74.2% of their original features and functions.

This data may seem disheartening, and in fact, 48% of the IT executives in our research sample feel that there are more failures currently than just five years ago. The good news is that over 50% feel there are fewer or the same number of failures today than there were five and ten years ago.

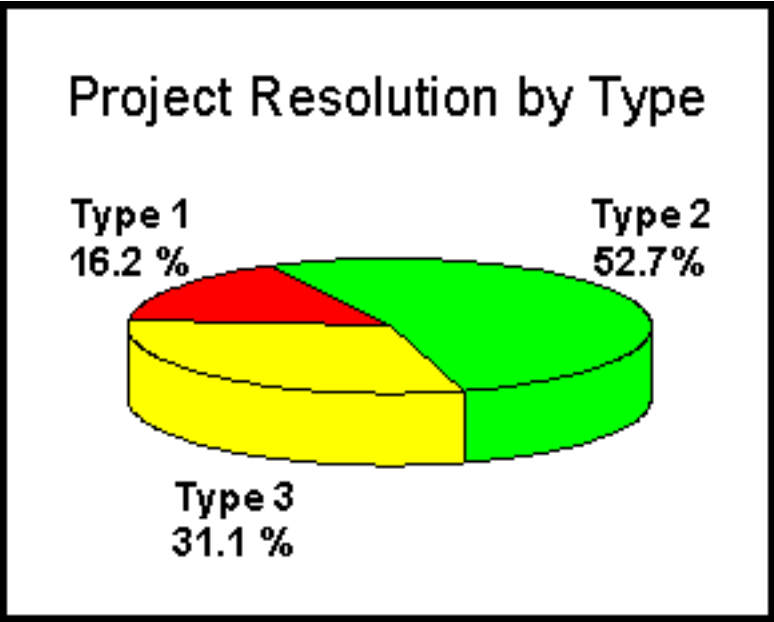
METHODOLOGY

The survey made by The Standish Group was as thorough as possible, short of the unreachable goal of surveying every company with MIS in the country. The results are based on what we at The Standish Group define as "key findings" from our research surveys and several personal interviews. The respondents were IT executive managers. The sample included large, medium, and small companies across major industry segments, e.g., banking, securities, manufacturing, retail, wholesale, heath care, insurance, services, and local, state, and federal organizations. The total sample size was 365 respondents and represented 8,380 applications. In addition, The Standish Group conducted four focus groups and numerous personal interviews to provide qualitative context for the survey results.

For purposes of the study, projects were classified into three resolution types:

- Resolution Type 1, or project success: The project is completed on-time and on-budget, with all features and functions as initially specified.
- Resolution Type 2, or project challenged: The project is completed and operational but over-budget, over the time estimate, and offers fewer features and functions than originally specified.
- Resolution Type 3, or project impaired: The project is canceled at some point during the development cycle.

Overall, the success rate was only 16.2%, while challenged projects accounted for 52.7%, and impaired (canceled) for 31.1%.



FAILURE STATISTICS

The Standish Group further segmented these results by large, medium and small companies. A large company is any company with greater than \$500 million dollars in revenue per year, a medium company is defined as having \$200 million to \$500 million in yearly revenue, and a small company is from \$100 million to \$200 million.

The figures for failure were equally disheartening in companies of all sizes. Only 9% of projects in large companies were successful. At 16.2% and 28% respectively, medium and small companies were somewhat more successful. A whopping 61.5% of all large company projects were challenged (Resolution Type 2) compared to 46.7% for medium companies and 50.4% for small companies. The most projects, 37.1%, were impaired and subsequently canceled (Resolution Type 3) in medium companies, compared to 29.5% in large companies and 21.6% in small companies.

Restarts

One of the major causes of both cost and time overruns is restarts. For every 100 projects that start, there are 94 restarts. This does not mean that 94 of 100 will have one restart, some projects can have several restarts. For example, the California Department of Motor Vehicles project, a failure scenario summarized later in this article, had many restarts.

Cost Overruns

Equally telling were the results for cost overruns, time overruns, and failure of the applications to provide expected features. For combined Type 2 and Type 3 projects, almost a third experienced cost overruns of 150 to 200%. The average across all companies is 189% of the original cost estimate. The average cost overrun is 178% for large companies, 182% for medium companies, and 214% for small companies.

Cost Overruns % of Responses

Under 20%	15.5%
21 - 50%	31.5%
51 - 100%	29.6%
101 - 200%	10.2%
201 - 400%	8.8%
Over 400%	4.4%

Time Overruns

For the same combined challenged and impaired projects, over one-third also experienced time overruns of 200 to 300%. The average overrun is 222% of the original time estimate. For large companies, the average is 230%; for medium companies, the average is 202%; and for small companies, the average is 239%.

Time Overruns % of Responses

Under 20%	13.9%
21 - 50%	18.3%
51 - 100%	20.0%
101 - 200%	35.5%
201 - 400%	11.2%
Over 400%	1.1%

Content Deficiencies

For challenged projects, more than a quarter were completed with only 25% to 49% of originally-specified features and functions. On average, only 61% of originally specified features and functions were available on these projects. Large companies have the worst record with only 42% of the features and functions in the end product. For medium companies, the percentage is 65%. And for small companies, the percentage is 74%.

% of Features/Functions % of Responses

Less Than 25%	4.6%
25 - 49%	27.2%
50 - 74%	21.8%
75 - 99%	39.1%
100%	7.3%

Currently, the 365 companies have a combined 3,682 applications under development. Only 431 or 12% of these projects are on-time and on-budget.

SUCCESS/FAILURE PROFILES

The most important aspect of the research is discovering why projects fail. To do this, The Standish Group surveyed IT executive managers for their opinions about why projects succeed. The three major reasons that a project will succeed are user involvement, executive management support, and a clear statement of requirements. There are other success criteria, but with these three elements in place, the chances of success are much greater. Without them, chance of failure increases dramatically.

Project Success Factors	% of Responses
1. User Involvement	15.9%
2. Executive Management Support	13.9%
3. Clear Statement of Requirements	13.0%
4. Proper Planning	9.6%
5. Realistic Expectations	8.2%
6. Smaller Project Milestones	7.7%
7. Competent Staff	7.2%
8. Ownership	5.3%
9. Clear Vision & Objectives	2.9%
10. Hard-Working, Focused Staff	2.4%
Other	13.9%

The survey participants were also asked about the factors that cause projects to be challenged.

Project Challenged Factors	% of Responses
1. Lack of User Input	12.8%
2. Incomplete Requirements & Specifications	12.3%
3. Changing Requirements & Specifications	11.8%
4. Lack of Executive Support	7.5%
5. Technology Incompetence	7.0%
6. Lack of Resources	6.4%
7. Unrealistic Expectations	5.9%
8. Unclear Objectives	5.3%
9. Unrealistic Time Frames	4.3%
10. New Technology	3.7%
Other	23.0%

Opinions about why projects are impaired and ultimately canceled ranked incomplete requirements and lack of user involvement at the top of the list.

Project Impaired Factors	% of Responses
1. Incomplete Requirements	13.1%
2. Lack of User Involvement	12.4%
3. Lack of Resources	10.6%
4. Unrealistic Expectations	9.9%
5. Lack of Executive Support	9.3%
6. Changing Requirements & Specifications	8.7%
7. Lack of Planning	8.1%
8. Didn't Need It Any Longer	7.5%
9. Lack of IT Management	6.2%
10. Technology Illiteracy	4.3%
Other	9.9%

Another key finding of the survey is that a high percentage of executive managers believe that there are more project failures now than five years ago and ten years ago. This despite the fact that technology has had time to mature.

	Than 5 Years Ago	Than 10 Years Ago
Significantly More Failures	27%	17%
Somewhat More Failures	21%	29%
No Change	11%	23%
Somewhat Fewer Failures	19%	23%
Significantly Fewer Failures	22%	8%

FOCUS GROUPS

To augment the survey results, The Standish Group conducted four focus groups with IT executives of major companies. The attendees were from a cross section of industries, including insurance, state and federal government, retail, banking, securities, manufacturing and service. Two of the focus groups were in Boston. The other two, in San Francisco. Each focus group had an average of ten participants with an overall total of forty-one IT executives. The purpose of these particular focus groups was to solicit opinions on why projects fail. In addition, The Standish Group conducted interviews with various IT managers. Some of their comments are enlightening about the variety of problems besetting project development.

Many of the comments echoed the findings of The Standish Group survey. "We have 500 projects. None are on-time and on-budget. This year, 40% will get canceled," said Edward, Vice President of MIS at a pharmaceutical company.

Other comments went directly to the reasons for failure. Jim, the Director of IT at a major medical equipment manufacturer, said: "Being that it's a mindset, it's very difficult to get all of the management -- it's even on the local level, not even on a worldwide level -- to get all of the management to agree on a set of rules.... That's a challenge in itself because you have to, in some cases, convince them that this is best for the company, not necessarily best for them, but best for the company. And you have to have that buy-in. If you don't have that buy-in, you're going to fail. I don't care how big or how small the project is."

John, Director of MIS at a government agency added: "Probably 90% of application project failure is due to politics!" And Kathy, a programmer at a telecommunication company, offered an even more scathing comment on politics: "Sometimes you have to make a decision you don't like. Even against your own nature. You say well, it's wrong, but you make that decision anyway. It's like taking a hammer to your toe. It hurts."

Bob, the Director of MIS at a hospital, commented on external factors contributing to project failure. "Our biggest problem is competing priorities," he said. "We just had a reorganization today. So now that's going to sap all the resources. And explaining to senior management that, 'Well, it's really taking us the time we said it was going to take. But because you've

reorganized the company, I'm going to take another six months on this other project, because I'm doing something else for you.' That's the biggest issue I have." Bill, the Director of MIS at a securities firm, added: "Changes, changes, changes; they're the real killers."

Some of the comments were darkly humorous. "Brain-dead users, just plain brain-dead users," said Peter, an application analyst at a bank. "When the projected started to fail," said Paul, a programmer at a personal products manufacturer, "the management got behind it -- way behind."

The comment most indicative of the chaos in project development came from Sid, a project manager at an insurance company. "The project was two years late and three years in development," he said. "We had thirty people on the project. We delivered an application the user didn't need. They had stopped selling the product over a year before."

CASE STUDIES

For further insight into failure and success, The Standish Group looked carefully at two famous Resolution Type 3 (canceled) projects and two Resolution Type 1 (successful) projects. For purposes of comparison, the project success criteria from the survey of IT executive managers was used to create a "success potential" chart. The success criteria were then weighted, based on the input from the surveyed IT managers. The most important criterion, "user involvement," was given 19 "success points". The least important -- "hard-working, focused staff" -- was given three points. Two very important success criteria -- "realistic expectations" and "smaller project milestones" -- were weighted at ten and nine points respectively. Finally, as presented later in this report, each of the case studies was graded.

California DMV

In 1987, the California Department of Motor Vehicles (DMV) embarked on a major project to revitalize their drivers license and registration application process. By 1993, after \$45 million dollars had already been spent, the project was canceled.

According to a special report issued by DMV, the primary reason for redeveloping this application was the adoption new technology. They publicly stated: "The specific objective of the 1987 project was to use modern technology to support the DMV mission and sustain its growth by strategically positioning the DMV data processing environment to rapidly respond to change." Also, according to the DMV special report "The phasing was changed several times, but the DMV technical community was never truly confident in its viability...."

The project had no monetary payback, was not supported by executive management, had no user involvement, had poor planning, poor design specifications and unclear objectives. It also did not have the support of the state's information management staff.

The DMV project was not rocket science. There are much harder applications than driver licenses and registrations. But because of internal state politics, unclear objectives, and poor planning, the project was doomed from the start.

American Airlines

Early in 1994, American Airlines settled their lawsuit with Budget Rent-A-Car, Marriott Corp. and Hilton Hotels after the \$165 million CONFIRM car rental and hotel reservation system project collapsed into chaos.

This project failed because there were too many cooks and the soup spoiled. Executive management not only supported the project, they were active project managers. Of course, for a project this size to fail, it must have had many flaws. Other major causes included an incomplete statement of requirements, lack of user involvement, and constant changing of requirements and specifications.

Hyatt Hotels

While Marriott and Hilton Hotels were checking out of their failed reservation system, Hyatt was checking in. Today, you can dial from a cellular airplane telephone at 35,000 feet, check into your Hyatt hotel room, schedule the courtesy bus to pick you up, and have your keys waiting for you at the express desk. This new reservation system was ahead of schedule, under budget, with extra features -- for a mere \$15 million of cold cash. They used modern, open systems software with an Informix database and the TUXEDO transaction monitor, on Unix-based hardware.

Hyatt had all the right ingredients for success: user involvement, executive management support, a clear statement of requirements, proper planning, and small project milestones.

Banco Itamarati

A year after a strategic redirection, Banco Itamarati, a privately-held Brazilian bank, produced an annual net profit growth of 51% and moved from 47th to 15th place in the Brazilian banking industry. Three fundamental reasons account for Banco Itamarati's success. First, they had a clear vision with documented specific objectives. Second, their top-down level of involvement allowed Banco Itamarati to stay on course. And finally, the bank produced incremental, measurable results throughout the planning/implementation period.

Banco Itamarati's clear business goal is to be one of Brazil's top five privately-held banks by the year 2000. Their objectives include maintaining a close relationship with their customers to improve and maintain an understanding of their needs, offering competitive financial solutions, guaranteeing customer satisfaction, and finally producing balanced results for the Itamarati Group. Banco Itamarati's objectives were incorporated into a strategic plan that clearly identified measurable results and individual ownership.

Their strategic plan made technology a key component of the business strategy. Itamarati used Itaotec's GRIP OLTP monitor as a basic tool for integrating software components. According to Henrique Costabile, Director of Organization Development, "We are one of the first banks to implement a client-server architecture that maximizes the potential of this architecture." Executive leadership, a well-communicated plan, and a skilled diverse team provided the foundation for Banco Itamarati to achieve their long-term goal, potentially ahead of schedule.

CASE STUDY CONCLUSIONS

The study of each project included adding up success points on the "success potential" chart.

Success Criteria	Points	DMV	CONFIRM	HYATT	ITAMARATI
1. User Involvement	19	NO ( 0)	NO ( 0)	YES (19)	YES (19)
2. Executive Management Support	16	NO ( 0)	YES (16)	YES (16)	YES (16)
3. Clear Statement of Requirements	15	NO ( 0)	NO ( 0)	YES (15)	NO ( 0)
4. Proper Planning	11	NO ( 0)	NO ( 0)	YES (11)	YES (11)
5. Realistic Expectations	10	YES (10)	YES (10)	YES (10)	YES (10)
6. Smaller Project Milestones	9	NO ( 0)	NO ( 0)	YES ( 9)	YES ( 9)
7. Competent Staff	8	NO ( 0)	NO ( 0)	YES ( 8)	YES ( 8)
8. Ownership	6	NO ( 0)	NO ( 0)	YES ( 6)	YES ( 6)
9. Clear Vision & Objectives	3	NO ( 0)	NO ( 0)	YES ( 3)	YES ( 3)
10. Hard-Working, Focused Staff	3	NO ( 0)	YES ( 3)	YES ( 3)	YES ( 3)
TOTAL	100	10	29	100	85

With only 10 success points, the DMV project had virtually no chance of success. With 100 success points, Hyatt's reservation project had all the right ingredients for success. With only 29 success points, the CONFIRM project had little chance of success. With 85, Itamarati, while not as assured as Hyatt, started with a high success probability.

THE BRIDGE TO SUCCESS

Notwithstanding, this study is hardly in-depth enough to provide a real solution to such a daunting problem as the current project failure rates. Application software projects are truly in troubled waters. In order to make order out of the chaos, we need to examine why projects fail. Just like bridges, each major software failure must be investigated, studied, reported and shared.

Because it is the product of the ideas of IT managers, the "Success Potential" chart can be a useful tool for either forecasting the potential success of a project or evaluating project failure.

Research at The Standish Group also indicates that smaller time frames, with delivery of software components early and often, will increase the success rate. Shorter time frames result in an iterative process of design, prototype, develop, test, and deploy small elements. This process is known as "growing" software, as opposed to the old concept of "developing" software. Growing software engages the user earlier, each component has an owner or a small set of owners, and expectations are realistically set. In addition, each software component has a clear and precise statement and set of objectives. Software components and small projects tend to be less complex. Making the projects simpler is a worthwhile endeavor because complexity causes only confusion and increased cost.

There is one final aspect to be considered in any degree of project failure. All success is rooted in either luck or failure. If you begin with luck, you learn nothing but arrogance. However, if you begin with failure and learn to evaluate it, you also learn to succeed. Failure begets knowledge. Out of knowledge you gain wisdom, and it is with wisdom that you can become truly successful.

Copyright 1995

This report is the property of The Standish Group International, Inc. and is made available to a restricted number of clients only upon these terms and conditions. The Standish Group International, Inc. reserves all rights herein. Reproduction or disclosure in whole or part to parties other than The Standish Group International, Inc. client, which is the original subscriber to this report, is permitted only with the written and express consent of The Standish Group International, Inc. No part of this report may be reproduced, stored or distributed via an electronic retrieval system, or transmitted in any form or by any means without express written permission from the publisher. Please respect intellectual rights! This report shall be treated at all times as a confidential and proprietary document for internal use only. The information contained in this report is believed to be reliable, but cannot be guaranteed to be correct or complete.

Do Not Reproduce

For more information on project failure and success click here for [Unfinished Voyages!](#)



[About us](#)

[Events](#)

[Sample research](#)

[Services](#)

[Jobs](#)

[Contact us](#)

THE

STANDISH  
GROUP

*sample research*



Sample Research Paper

# Unfinished Voyages

Cobb's Paradox:

*"We know why projects fail, we know how to prevent their failure -- so why do they still fail?"*

Martin Cobb  
Treasury Board of Canada Secretariat  
Ottawa, Canada

## Introduction

Off the coast of the eastern seaboard in 1906 there were 177 shipwrecks. These shipwrecks were caused by accidents, bad management, carelessness, errors in judgment, and pilot errors --all of which can be attributed to human (as opposed to technological) failings. So too were the results of The Standish Group report on development project failures.

From November 6th through the 9th, 1995, The Standish Group held CHAOS University in Chatham, Massachusetts. CHAOS University was a follow-up to the CHAOS study published in January 1995. The Standish Group estimates that almost 80,000 projects were cancelled in 1995. While The Standish Group identified the ten main causes of these failures (along with possible solutions), it was unclear whether these solutions could be implemented. In response to this and by invitation only, CHAOS University brought together 60 IT professionals for the purpose of digging down to create two additional levels of detail in the implementation of the success factors.

Achieving the answers to solving project failure often lies in developing written communication such as problem statements, project plans, and detail specifications. However, one of the problems with any written communication is the participant's (reader's) level of understanding. As technologists, we think, write, and talk in a manner that is not readily grasped by many people outside our industry. Aside from sounding intimidating, you run the danger of the reader actually thinking they understand what you are saying, while your meaning may in fact be entirely different. To paraphrase the words of the English poet, Samuel Taylor Coleridge "Until you understand a reader's ignorance, presume yourself ignorant of his understanding". In other words, write the document devoid of all technical terms and pseudo technical terms. This includes words used by our industry, but rarely used outside our industry. Words like paradigm, metric, abstraction, and orthogonal, should not be used in any document if you want the normal reader to understand. Remember it is your job make the reader understand the plan. It is not your job to show how smart you are or to demonstrate that you can use big words.

At CHAOS University Martin Cobb, Treasury Board of Canada Secretariat, Ottawa, Canada outlined his paradox: ***"We know why projects fail, we know how to prevent their failure -- so why do they still fail?"*** Throughout the next year via both magazine articles and our home page ([www.standishgroup.com](http://www.standishgroup.com)), Standish will explore the next level of detail and work to solve the mystery of Cobb's paradox.

The CHAOS Report surveyed IT executive managers. For the purposes of project comparison, The Standish Group used the project success criteria from the surveyed IT managers to create a success potential chart. The success criteria were then weighted based on the input from the surveyed IT managers. The most important criteria, "user involvement", was given 19 success points while the least important, "hard-working, focused staff" was given 3 success points. The table below lists the criterion in order of importance with their appropriate success points.

SUCCESS CRITERIA	POINTS
1. User Involvement	19
2. Executive Management Support	16
3. Clear Statement of Requirements	15
4. Proper Planning	11
5. Realistic Expectations	10
6. Smaller Project Milestones	9
7. Competent Staff	8
8. Ownership	6

9. Clear Vision & Objectives	3
10. Hard-Working, Focused Staff	3
TOTAL	100

CHAOS University attendees broke down each success criteria into another level of detail. Attendees of CHAOS University were asked to problem solve and come up with five positive ways to achieve each of the success criteria. As you read the five reasons, consider whether or not you have considered them in a real project. After each explanation of the five positives, they are listed again in question form. Ask yourself each question with a specific project in mind and allot yourself the number of points specified for each YES answer. After reading all ten success criterion and the positive ways to achieve them, you should be able to calculate a score which can be a maximum of 100 and a minimum of 0. This number represents your project's success potential.

PROJECT SUCCESS POTENTIAL

In this COMPASS Report we look at the first level of detail.

USER INVOLVEMENT

First, find the right user or users. Look for users up and down the organization. Second, involve the user (or users) early and often. Third, establish a quality relationship with the user(s) by keeping open lines of communication throughout the life of the project. Fourth, make it easy for them to be involved in the project. Last, but not least, talk to them and find out what they need. After all, the only reason the project exists in the first place is because someone needs to use the business application when it is finished.

- Do I have the right user(s)? \_\_\_\_\_
- Did I involve the user(s) early and often? \_\_\_\_\_
- Do I have a quality user(s) relationship? \_\_\_\_\_
- Do I make involvement easy? \_\_\_\_\_
- Did I find out what the user(s) needs? \_\_\_\_\_

For each question with a YES answer, add 3.8 points to the total project success potential score.  
Total Points (not to exceed 19) \_\_\_\_\_

EXECUTIVE MANAGEMENT SUPPORT

First, find a key executive with a vested interest in the successful outcome of the project. Second, the key executive must have a bottom line responsibility to his/her personal career. Third, the consequence of failure is acceptable. Fourth, show the key executive a well-defined plan. Fifth and final, show the project team has a stake in the project's success. In addition, allow for continuous communication with the key executive in all aspects of the project as it moves through its life cycle.

- Do I have the key executive(s)? \_\_\_\_\_
- Does the key executive have a stake in the outcome? \_\_\_\_\_
- Is failure acceptable? \_\_\_\_\_
- Do I have a well defined plan? \_\_\_\_\_
- Does the project team have a stake? \_\_\_\_\_

For each question with a YES answer, add 3.2 points to the total project success potential score.  
Total Points (not to exceed 16) \_\_\_\_\_

DEVELOPING CLEAR STATEMENT OF REQUIREMENTS:

First, write a concise definition of the vision in the short-term, the mid-term and the long-term. Second, write a functional cross-section analysis and allow for re-iteration. Third, develop a functional risk assessment and management document. Develop a business case statement outlining return on investment. Finally, define metrics, measurements, and milestones to determine success and/or the completion of the project. Additionally, define what is not to be included in the project.

- Do I have a concise vision? \_\_\_\_\_
- Do I have a functional analysis? \_\_\_\_\_
- Do I have a risk assessment? \_\_\_\_\_
- Do I have a business case? \_\_\_\_\_
- Can I measure the project? \_\_\_\_\_

For each question with a YES answer, add 3 points to the total project success potential score.  
Total Points (not to exceed 15) \_\_\_\_\_

PROPER PLANNING:

First, develop a brief formal problem or concept statement. This document should include a statement describing the problem and the resulting benefit to the organization if the problem is solved. Second write a requirements definition or concept solution document. This document is not the full specifications report, but outlines possible solutions to the problem. Third, identify the proper personnel. In this regard, name names and let these people know what their role is in the project. Have a firm start date and require the personnel to be available on that date. Develop a firm functional

specification. Do not let this document promote scope creep, however it should allow for changing business requirements. Finally, develop a project plan with attainable milestones and prioritization. One the attendees said that users want so much out of their projects it is like "prioritizing an avalanche."

- Do I have a problem statement? \_\_\_\_\_
- Do I have a solution statement? \_\_\_\_\_
- Do I have the right people? \_\_\_\_\_
- Do I have a firm specification? \_\_\_\_\_
- Do I have attainable milestones? \_\_\_\_\_

For each question with a YES answer, add 2.2 points to the total project success potential score.  
Total Points (not to exceed 11)\_\_\_\_\_

**SETTING REALISTIC EXPECTATIONS:**

First, write a firm and clear specification document outlining a project that is attainable. You need to take the time to write this document. To invoke the old saying "If you fail to plan ... then plan to fail." Second, prioritize project needs. Eliminate most of the features that are "wanted" or "nice to have". One of the attendees said that when a user insists on wanting too much she just nods, knowing she will deliver what she can. We need to be clearer and sometimes braver, concentrating on and delivering only features that are required. Third, develop smaller project milestones. Fourth, provide for change and manage the change. Fifth, prototype the project.

- Do I have clear specifications? \_\_\_\_\_
- Do I have prioritization of needs? \_\_\_\_\_
- Do I have small milestones? \_\_\_\_\_
- Can I manage change? \_\_\_\_\_
- Can I prototype? \_\_\_\_\_

For each question with a YES answer, add 2 points to the total project success potential score.  
Total Points (not to exceed 10) \_\_\_\_\_

**SMALL PROJECT MILESTONES:**

In the majority of cases, 20% of a project's features will provide 80% of user benefits. The first thing is to concentrate on the twenty percent. Second, create a top-down design by understanding the "big picture" and breaking the project into manageable parts. Third, set a time limit on project milestones and deliver what is completed in that time limit. Fourth, use prototyping tools. Fifth, measure, quantify and account for the results.

- Am I using the 80/20 rule? \_\_\_\_\_
- Am I using a top-down design? \_\_\_\_\_
- Am I setting time limits? \_\_\_\_\_
- Am I using a prototype tool? \_\_\_\_\_
- Can I measure progress? \_\_\_\_\_

For each question with a YES answer, add 1.8 points to the total project success potential score.  
Total Points (not to exceed 9) \_\_\_\_\_

**COMPETENT STAFF:**

First, identify the skills required; such as an understanding of the business, leadership experience, and technical knowledge. Second, recruit appropriately skilled people from both inside and out of the company. Third, develop a well-structured and continuous training program. Fourth, provide and present incentives such as career advancement, skill expansion, and of course money, either in the form of bonuses and/or raises. Such incentives will insure the staff will be focused on the project and willing to see it through to the end.

- Do I know the skills required? \_\_\_\_\_
- Do I have the right people? \_\_\_\_\_
- Do I have a training program? \_\_\_\_\_
- Do I have incentives? \_\_\_\_\_
- Will the staff see it through? \_\_\_\_\_

For each question with a YES answer, add 1.6 points to the total project success potential score.  
Total Points (not to exceed 8) \_\_\_\_\_

**PROJECT OWNERSHIP:**

First, clearly define roles and responsibilities for each member of the project team. Second, define an organizational model which supports the above accountability. Third, communicate the defined roles throughout the corporation. Fourth, tie incentives to the project's success. Finally, get commitments from each project participant, including users and key executives.

- Do I have defined roles? \_\_\_\_\_
- Do I have a defined organization? \_\_\_\_\_
- Does everyone know their role? \_\_\_\_\_
- Are incentives attached to success? \_\_\_\_\_

- Is everyone committed? \_\_\_\_\_

For each question with a YES answer, add 1.2 points to the total project success potential score.  
Total Points (not to exceed 6) \_\_\_\_\_

**CLEAR VISION AND OBJECTIVES:**

First, share the vision with all people and groups affected by the project. Second, be sure the objectives align with corporate goals. Third, make sure the objectives are achievable. Fourth, create measurable goals. Finally, install honest and continuous sanity checks.

- Is the vision shared? \_\_\_\_\_
- Is the vision aligned with company goals? \_\_\_\_\_
- Are the objectives achievable? \_\_\_\_\_
- Are the objectives measurable? \_\_\_\_\_
- Do I have honest sanity checks? \_\_\_\_\_

For each question with a YES answer, add 0.6 points to the total project success potential score.  
Total Points (not to exceed 3) \_\_\_\_\_

**HARD WORKING, FOCUSED STAFF:**

First, provide incentives for the staff such bonuses, raises or promotions. Second, have the staff concentrate on quantifiable deliverables. Third, convey to each individual that they have part ownership in the project. Fourth, communicate that each individual has a role, which is inclusive of functioning as a team. Build a team that works well together. Finally, a proper plan with attainable results and milestones will build confidence in the staff and keep them focused.

- Are there incentives? \_\_\_\_\_
- Are we concentrating on quantifiable deliverables? \_\_\_\_\_
- Does each member have part ownership? \_\_\_\_\_
- Does everyone work together? \_\_\_\_\_
- Are we building confidence? \_\_\_\_\_

For each question with a YES answer, add 0.6 points to the total project success potential score.  
Total Points (not to exceed 3) \_\_\_\_\_

Calculate all of the points to achieve the final score.  
The Success Potential for Project \_\_\_\_\_ is \_\_\_\_\_.

**Copyright 1996**

This report is the property of The Standish Group International, Inc. and is made available to a restricted number of clients only upon these terms and conditions. The Standish Group International, Inc. reserves all rights herein. Reproduction or disclosure in whole or part to parties other than The Standish Group International, Inc. client, which is the original subscriber to this report, is permitted only with the written and express consent of The Standish Group International, Inc. No part of this report may be reproduced, stored or distributed via an electronic retrieval system, or transmitted in any form or by any means without express written permission from the publisher. Please respect intellectual rights! This report shall be treated at all times as a confidential and proprietary document for internal use only. The information contained in this report is believed to be reliable, but cannot be guaranteed to be correct or complete.

**Do Not Reproduce**

