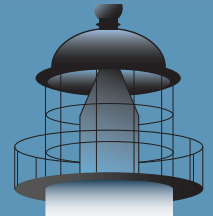


CHAOS REPORT: 21ST ANNIVERSARY EDITION



Our original *CHAOS Report* in 1994 started with the paragraph, *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 this 21st anniversary of the *CHAOS Report* we consider improvements that have been made. The answer lies in this explanation written more than 20 years ago: 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.

In the original *CHAOS Report* we set out three objectives to examine: 1) the scope of software project failures; 2) the major factors that cause software projects to fail; and 3) the key ingredients that can reduce project failures. In this 21st edition, we look at the scope of project failure, and then consider causes and cures. The chart shows the resolution of all software projects from 2009 to 2013 within the new *CHAOS* database. The total number of software projects exceeds 25,000, with an average of 5,000 per yearly period. The overall results show only minor improvement in success rate over the last 21 years of data collection. For example, in 1994, 31% of all projects were on time, on budget, and on target. In 2013, the number of projects that were on time, on budget, and on target had improved to 36%.

On the other hand, 21 years ago the failure rate was 16%. In 2013, the failure rate is also 16%. On the face of it the results have not changed much in the last 21 years. However, during that time period the management of software projects has bifurcated into a linear style and a logical style. The 21st anniversary *CHAOS Report* takes the traditional view of project success/failure by considering the "triple constraints":

- **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/or offers fewer features and functions than specified.
- **RESOLUTION TYPE 3, OR PROJECT IMPAIRED:** The project is canceled at some point during the development cycle or not used after implemented.

The *CHAOS Manifesto 2014: Value versus Success & the Orthogonals* presents an alternate view of project success—one that is based on value.

CHAOS Report is based on the collection of project case information from real-life IT environments and software projects. This version and past versions have used eight different instruments in the collection of this information, which includes project profiles, project tracking, individual project surveys, case interviews, general surveys, project postmortems, and other instruments. *CHAOS* research encompasses 21 years of data on why projects succeed or fail, representing more than 100,000 completed IT projects. However, for our new database we eliminated cases from 1994 through 2002, since they did not match the current requirements for analysis. The new *CHAOS* database has just under 50,000 projects.

CHAOS RESOLUTION FOR ALL PROJECTS

	1994	2009	2010	2011	2012	2013
SUCCESSFUL	31%	36%	38%	37%	41%	36%
CHALLENGED	53%	44%	40%	46%	40%	48%
FAILED	16%	20%	22%	17%	19%	16%



Challenged projects are projects that miss the cost estimate, time estimate, and feature/function target. Challenged projects are not failed projects. Many challenged projects provide value; just as many successful projects provide no value. The chart shows the cost and time overruns of challenged projects from 2009 to 2013 within the new CHAOS database. The chart also shows the percent of scope in the deliverable of the project. The total number of challenged software projects in the dataset is more than 11,000. Cost overruns and scope have been pretty consistent, with a slight rise in both items during the time period, from 75% to 80% and 49% to 54%, respectively, over the original estimate. However, time overruns doubled from 2009 to 2013. This indicates that a non-cost lapsed time element is being reported that we have not seen before.

Our very first project case review was the California Department of Motor Vehicles (DMV). In 1987, the California DMV embarked on a major project to revitalize its driver license and registration application process. By 1993, after \$45 million had already been spent, the project was canceled. Now 21 years later we reviewed the Massachusetts Registry of Motor Vehicles (RMV) modernization program. This program is a \$160 million driver license and registration application project. Our profile of this project shows that the project has no chance of coming in on time or on budget. In fact, the new time frame added three years to the original schedule, making it already a challenged project.

The primary reason for redeveloping both applications was the adoption of new technology. It is clear that both projects lacked monetary payback. The California DMV did not have the support of executive management, lacked user involvement, and had poor planning, poor design specifications, and unclear objectives. On the other hand, the Massachusetts Motor Vehicle program does have the support of executive management, and has intense user involvement, sophisticated planning, elaborate design specifications, and very clear objectives. The reasons this project has a high likelihood of failure include executive sponsor turnover, size, complexity, internal state politics, vendor conflicts, and lack of intermediate gratification with a big bang deliverable. On the other hand, it has a wonderful opportunity to provide great service to the citizens of Massachusetts by adopting a value-driven approach as outlined in the *CHAOS Manifesto 2014*.

COST, TIME, AND TARGET

	2009	2010	2011	2012	2013
Cost Overrun	75%	74%	75%	76%	80%
Time Overrun	53%	57%	103%	105%	106%
Scope	49%	48%	51%	49%	54%

CHAOS DEMOGRAPHICS: CHAOS results provide a global view of project statistics but do tend to have a heavier concentration on the United States and Europe. For each reporting period, about 60% of the projects are U.S. based, 25% are European, and the remaining 15% represent the rest of the world. A little more than half of the companies are considered Fortune 1000-type companies; another 30% are midrange in size; and 20% are in the small-size category. They span a diverse number of vertical industries and organizations. Participants are made up of a variety of CIOs, VPs, directors, and PMO project managers.

Challenged projects are projects that miss the cost estimate, time estimate, and feature/function target. Challenged projects are not failed projects. Many challenged projects provide value, just as many successful projects provide no value.



Size has always been a major element in the CHAOS research. It was clear from the very beginning of the CHAOS research that size was the single most important factor in the resolution of project outcome. Here we show two charts: 1) resolution of all software projects by size; and 2) size of the software projects by resolution. The first chart shows the resolution of all software projects by size from 2009 to 2013 within the new CHAOS database. The total number of software projects is more than 25,000. The results show that only 2% of successful projects were over \$10 million in labor costs, 54% of successful projects were under \$1 million in labor costs, and 76% of failed projects were over \$3 million in labor costs.

The second chart shows the size of the software projects by resolution from 2009 to 2013 within the new CHAOS database. The total number of software projects is more than 25,000. The results show that only 8% of large projects (over \$10 million in labor costs) were successful, 51% of large projects were challenged, and 41% of large projects failed. The results show that 70% of small projects were successful. Only 12% of projects under \$3 million to \$1 million in labor costs had a record of failure. While, only 5% of projects under \$1 million in labor cost had a record of failure. Taking some of the \$1 to \$3 million projects and reducing them to under \$1 million projects will more than double the chances the projects will not fail.

In our *CHAOS Manifesto 2014* we stated that almost every big, expensive, complex, slow project is made up of hundreds of SAFE (simple, absorbent, fast, and economical) projects trying to break free. While it is sometimes hard to break up large projects into multiple small projects and deliverables the rewards of such effort are immense. It is much easier in today's technology environment with standard infrastructure software— such as middleware, databases, and system management—to implement small projects on a continuous flow. Current software techniques such as object-oriented programming, agile, and DevOps methodologies as well as service-oriented architecture all aid in using a small project philosophy.

Size has always been a major element in the CHAOS research.

RESOLUTION OF ALL SOFTWARE PROJECTS BY SIZE

	SUCCESSFUL	CHALLENGED	FAILED
Over \$10M	2%	9%	16%
\$6M to \$10M	8%	18%	25%
\$3M to \$6M	11%	28%	35%
\$1M to \$3M	25%	29%	16%
Under \$1M	54%	16%	8%
TOTAL	100%	100%	100%

SIZE OF THE SOFTWARE PROJECTS BY RESOLUTION

	SUCCESSFUL	CHALLENGED	FAILED	TOTAL
Over \$10M	8%	51%	41%	100%
\$6M to \$10M	19%	51%	30%	100%
\$3M to \$6M	18%	54%	28%	100%
\$1M to \$3M	39%	49%	12%	100%
Under \$1M	70%	25%	5%	100%



Looking at project resolution by industry provides another view of the CHAOS database. The chart shows the resolution of all software projects by industry from 2009 to 2013 within the new CHAOS database. The total number of software projects is more than 25,000. The results show that retail projects had the highest success rate at 46%. The results also show that government projects had the highest failure rate at 24%. In addition, the results show that banking projects had the highest challenged rate at 48%. For the last 21 years retail has always led in the top industries for project success, while government has always been far behind.

Retail as a project industry has a number of valuable attributes. First, time to market is paramount in the retail sector. Second, competitive pressures force early deliverables. Third, margins are small and costs are contained. Compliance and governance are generally minimal. Decisions are made swiftly. Projects are small with restrained scope. On the other hand, government projects are typically large. The budget process and project start-up costs are normally high and intense. Decisions are made slowly with multiple inputs and meetings. Compliance and governance are generally at the maximum. Time and money are not factors. Projects are large, with the hope of pleasing all stakeholders and constituents. If we look at these two extremes we can visualize the other industries along these dimensions and the results.

One of the reasons projects get big and complex is because of the advent of the formal requirements process and the business analysis profession. Both the formal requirements process and the business analysis profession have become commonplace and popular over the last 21 years with IT management. The reason for the popularity is that many projects failed to deliver the needed business functionality. Prior to this only the obvious needs were part of the specifications and changing to accommodate additional functionality was often traumatic. This was especially important before the age of object-oriented programming and service-oriented architecture.

In many cases the formal requirements document becomes the basis for the specifications and the specifications turn into the scope of the project. Often the requirements and specifications are segmented by must have, need to have, and nice to have. Very little effort is made to segment by capability, risk, cost, and value. Very little effort is made to drain or remove requirements and specifications off to reduce the size of the project or break up the large project into microprojects. This is the key feature of The Standish Group's Value Portfolio Optimization and Management Service.

CHAOS RESOLUTION BY INDUSTRY

	SUCCESSFUL	CHALLENGED	FAILED
Banking	36%	48%	16%
Financial	38%	47%	15%
Government	29%	47%	24%
Healthcare	40%	43%	17%
Manufacturing	35%	46%	19%
Retail	46%	38%	16%
Services	38%	43%	19%
Telecom	36%	44%	20%
Other	38%	39%	23%



The type of project has a major effect on resolution. The chart on this page shows the resolution of all software projects by project type from 2009 to 2013 within the new CHAOS database. The total number of software projects is more than 25,000. Projects using a purchased application with modification had the second highest success rate at 54%. Projects that were developed from scratch using modern methodologies had a 22% failure rate. This is the highest failure rate other than the “other” category. The results also show that projects that were done with some developed components and purchased components had the highest challenged rate at 52%. This is followed closely by projects that purchased components and assembled the application with 50%.

Modernization projects have the highest success rate at 70%. The Standish Group has a very specific definition and development method for modernization projects. In fact, we modified “modernization” by adding “in place” so as not to confuse the general modernization of applications by the other techniques such as developing from scratch using modern methodologies or purchasing components. The IRS, California DMV, and the Massachusetts RMV are labeled as modernization projects but they are not or would not be part of the 70% since they are developed from scratch.

The definition of “modernization in place” is using the current operating application as the base of the project. Changes and replacements are made to the running application and users continue to use the existing parts and the new developments without interruption. Only this type of modernization project is included in the 70% success rate. The most successful and valuable modernization-in-place projects use our SAFE or microproject approach. In addition, we have developed a six-step modernization-in-place method. The six steps include updating a modern database technology, improving the user experience, and replacing the code with up-to-date languages. This method is outlined in our series *CHAOS Tuesday* with Harry Scott of Carr Scott Software.

Over 20 years ago Hyatt Hotels modernized its existing reservation system using many of the modernization-in-place techniques. As described in the original *CHAOS Report*, the project was ahead of schedule, under budget, with extra features—for a mere \$15 million of cold cash. Hyatt had all the right ingredients for success: user involvement, executive management support, a clear statement of requirements, proper planning, and small project milestones. The Standish Group’s Value Portfolio Optimization and Management Service can help develop strategies for modernization-in-place projects.

CHAOS RESOLUTION BY PROJECT TYPE

PROJECT TYPE	SUCCESSFUL	CHALLENGED	FAILED
Developed from scratch using traditional languages and methods	27%	55%	18%
Developed from scratch using modern methodologies	31%	47%	22%
Developed some components & purchased others	30%	52%	18%
Purchased components & assembled the application	32%	50%	18%
Purchased application & modified	54%	27%	19%
Purchased application & performed no modifications	51%	28%	21%
Modernization	70%	21%	9%
Other	36%	39%	25%

The chart compares the resolution of all software projects from 2009 to 2013 within the new CHAOS database, segmented by the agile process and waterfall method. The total number of software projects is more than 10,000. The results for all projects show that agile projects have four times the success rate as waterfall projects. The results also show that waterfall projects have three times the failure rate as agile projects. The results are also broken down by size: large, medium, and small. The overall results clearly show that waterfall projects do not scale well, while agile projects scale much better. However, note that the smaller the project, the smaller the difference is between the agile and the waterfall process.



One of the other reasons projects get big and complex is because of the common Byzantine budget and approval process. Many organizations we know will start the project budget process in early August and finish it the beginning of January. During this five-month budget process, the business units, the project management office, financial services, and executive committees review project budgets. Throughout this process many small projects are combined into larger projects to gain approval. There is much horse trading, feature enhancements, and compromises to get executives to approve projects. In fact, the number of projects is artificially kept small to help in prioritization, management, and tracking. Therefore, projects become large through consolidation and homogenization.

A better technique is to separate and prioritize small projects. The problem with small projects is also the common Byzantine budget and approval process. This procedure increases the costs and decreases the value of small projects. Some organizations have created frameworks and streamlined the approval process for small projects to reduce the overhead. However, this technique, while reducing some of the costs, is not good enough to create real value. Our recommendation is to create activity budgets such business analysis, design creation, software developments, idea management, and other software development activities. We call this breadbasket budgeting. Feed each of these activities with small projects and concrete deliverables.

The major benefit to breadbasket budgets is time to market and delivery. If it takes half a year to get a project started and then another year or two to get it delivered, the business changes will dramatically decrease the value of any project. The Standish Group's Value Portfolio Optimization and Management Service can help develop strategies for breadbasket budgets and small projects.

CHAOS RESOLUTION BY AGILE VERSUS WATERFALL

SIZE	METHOD	SUCCESSFUL	CHALLENGED	FAILED
All Size Projects	Agile	46%	44%	10%
	Waterfall	14%	58%	28%
Large Size Projects	Agile	29%	49%	22%
	Waterfall	4%	56%	40%
Medium Size Projects	Agile	42%	47%	11%
	Waterfall	12%	62%	26%
Small Size Projects	Agile	52%	42%	6%
	Waterfall	47%	45%	8%



Project resolution differs slightly by most of the areas of the world. The chart on this page shows the resolution of all software projects from 2009 to 2013 by the four major areas of the world. The total number of software projects is more than 25,000. The results show that for most of the world the success rate is 37% to 39% and failure rates are between 18% to 20%. Asia has the lowest success rate and both the highest challenged (46%) and failure (22%) rates. So the question is, why is Asia the outlier? Asia also has the lowest emotional maturity scores according to our emotional maturity appraisals and benchmarks. It is our opinion that they have a hard time saying “no” and confronting authority—even in the face of disaster.

Lack of emotional maturity is one of the top five reasons projects will either overrun their resources or fail outright. Emotional maturity is the ability and capacity to perceive, assess, manage, and direct the emotions and actions of the project executive sponsor, project stakeholders, and the project team. It is the skill to recognize and deal with the Five Deadly Sins of project management, which are overambition, arrogance, ignorance, abstinence, and fraudulence. Emotional maturity is the ability to promote and institutionalize the CHAOS Commandments, which are community, honor, awareness, objectivity, and superiority. Finally, emotional maturity is the ability to deliver and deal with bad news, set expectations, and create and maintain fair rules of engagement.

Ego is one of the other reasons projects get big and complex. Status is often associated with larger projects. Bigger budgets and larger staff often mean higher position and authority. This also increases the pay grade of the personnel on the project. The larger the organization, the larger the project needs to be to get visibility. No one is going to pay attention to a successful million-dollar software project by a federal agency. Even the billion-dollar failure of the Virtual Fence got little notice from the congressional and executive branches of the United States government. Even in the Commonwealth of Massachusetts a million-dollar project gets little notice.

A collection of small projects reduces the effect of the lack of emotional maturity. There is no ego in small projects. There is no overambition in the collection of small projects. A collection of small projects mitigates the influence of arrogance, ignorance, abstinence, and fraudulence. The most important factor for success is a skilled executive sponsor who can inspire the project team and stakeholders. One of the problems with large projects is it is difficult to kept the inspiration going for a long period of time. Small projects are done rapid-fire and therefore inspiration is only needed for a short period of time before results are evident.

CHAOS RESOLUTION BY AREA OF THE WORLD

	SUCCESSFUL	CHALLENGED	FAILED
North America	38%	43%	19%
Europe	37%	45%	18%
Asia	32%	46%	22%
Rest of World	39%	41%	20%

In the original CHAOS Report in 1994 we wrote: 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.



Complexity is one of the main reasons for project failure. The chart on this page shows the resolution of all software projects by complexity from 2009 to 2013 within the new CHAOS database. The total number of software projects is more than 25,000. The results show that 47% of very easy projects were successful. Very complex projects have both the highest challenged (56%) and failure (29%) rates. In the *CHAOS Manifesto 2014*, Rule #14 of the radical orthogonal states: *Complexity causes confusion and costs. Always keep it simple.* The best way to solve a complex problem is to find a simple solution. Inside of every complex problem are simple solutions. Complexity is often caused by size, conflicting goals, large budgets, and executive sponsor egos.

We use two tables to determine complexity. There are five attributes in each table. We then add up the points based on the attributes of the project to determine the complexity level. Our Size-Complexity Matrix provides a guideline for categorizing a project either by size or complexity. We also use a 5-point scale for both sizes. The lowest-point project is a simple, small project, and the highest-point projects are the most complex and large projects. Organizations can use the Size-Complexity Matrix to help with the decision of whether to take on a project or break up a project into SAFE projects. It also can be used to determine the workload of team members. The Size-Complexity Matrix should be used to build a portfolio of SAFE projects. The Standish Group’s Value Portfolio Optimization and Management Service can help develop strategies for reducing complexity.

CHAOS RESOLUTION BY COMPLEXITY

	SUCCESSFUL	CHALLENGED	FAILED
Very Complex	15%	56%	29%
Complex	20%	52%	28%
Average	39%	44%	17%
Easy	43%	41%	16%
Very Easy	47%	38%	15%

In the original CHAOS Report in 1994 we included this comment: *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.”*



The Law of the Roads states that it does not matter which road everyone comes from as long as they end up in the same place. Clarity and focus are essential to a successful project, but not necessarily value. The CHAOS law supports the success factor of clear business objectives. Every stakeholder will have his or her own agenda that needs to be fulfilled by the project. It only stands to reason that 10 stakeholders will equal 10 individual goals. This is a major issue because these variable self-interested needs will often conflict. Please note that the clear business objectives success factor is a cornerstone of success-driven projects, but it is not a factor in value-driven projects.

In the *CHAOS Manifesto 2014*, Rule #6 of the radical orthogonal states: *Action is better than clarity*. Project teams should reduce or give up control of the business objectives to encourage and promote innovation. Standish research shows that some of the most valuable projects are ones that are not aligned with the organization’s strategy. In the value-based approach the business objectives are dynamic as the project progresses and evolves. The idea is that action promotes more ideas and these new ideas turn into more action. The road to oblivion is paved with projects that are strategically aligned with the current corporate strategy and vision. Each stakeholder will have his or her own ideas about the project, which the project team needs to embrace and exploit.

The chart shows the resolution of all software projects by goal from 2009 to 2013 within the new CHAOS database. The total number of software projects is more than 25,000. The results show that 46% of projects with vague goals were successful. Both close and distant projects had the highest challenged (45%) rate. Precise projects had the highest failure (25%) rate. It also should be noted that the precise and close projects had the lowest value rating as stated in the *CHAOS Manifesto 2014*.

CHAOS RESOLUTION BY GOAL

	SUCCESSFUL	CHALLENGED	FAILED
Precise	31%	44%	25%
Close	34%	45%	21%
Loose	37%	44%	19%
Vague	46%	40%	14%
Distant	43%	45%	12%



Successful projects need smart, trained people. Not surprisingly, one of the key project success factors identified in Standish Group’s CHAOS research is a competent staff. There are five key fundamentals to ensure staff competency. First, identify the required competencies and alternative skills. Second, provide a good, continuous training program to enhance the staff skills. Third, recruit both internally and externally to provide a balance of experiences. Fourth, provide incentive to motivate the staff. Finally, ensure the staff is project-focused. When a project has both teamwork and skilled resources, it can prevail under even the direst of circumstances. To ensure a competent staff you must match the skills of the team to correspond with the needed skills of the project. This is one of the seven constraints we use to prioritize your project portfolio.

Here we look at project resolution by capability. The chart on this page shows the resolution of all software projects by capability from 2009 to 2013 within the new CHAOS database. The total number of software projects is more than 25,000. The results show that 46% of gifted resources were on successful projects. Projects that had unskilled people had both the highest challenged (63%) and failure (25%) rates. The decisions around project priority include: Do you go forward with a project if you lack skilled capability? This decision is especially pertinent for large projects with a large staff who have a mix of good and poor resources.

In the *CHAOS Manifesto 2014*, Rule #8 of the radical orthogonal states: *Eagles don’t flock*. A project should have a small team of highly talented people who understand both the business and the technology. One good person is equal to 10 mediocre people. One highly talented person is equal to 100 mediocre people. The skills of the team should always correspond with the needed skills of the project, but change as new innovations become apparent. More important, the use of small teams actually improves competency since the team works closely together to solve a discrete solution. In addition, since team members can see the results of the work early they get great feedback and satisfaction for a completed job.

One of the major reasons small projects work better than large is the ability of the users to assimilate features and functions. Large projects with many functions tend to frustrate users and cause much upheaval. In fact, the whole school of change management is built around a large crop of code with major changes. The reason is that your capable resources get feedback much quicker and can fix and take advantage of new opportunities.

CHAOS RESOLUTION BY CAPABILITY

	SUCCESSFUL	CHALLENGED	FAILED
Gifted	46%	37%	17%
Talented	40%	44%	16%
Competent	38%	43%	19%
Able	31%	45%	24%
Unskilled	12%	63%	25%



Twenty-one years ago we developed the factors of software project success. In the first 10 years the factors changed and moved around in rank quite a bit. However, most of the important factors are still on top of the list. During the last 10 years the factors have been relatively stable with slight movement up and down in position and point value. This chart shows our current thinking. It should be noted that while there has been no major improvement in project success, the investment profile has not changed much either. If you consider the current project management investment it is upside down with the factors of success. Much money is spent on tools and process improvement, which is at the bottom of the table, and little money is spent on improving the executive management support and emotional maturity, which are both at the top of the table. The table below shows the 1994 and 2014 versions of project success.

The big three pillars of project success are executive support, user involvement, and clear business objectives. Organizations have ignored these realities—and project resolutions clearly demonstrate the results. On Standish Group’s point scale, these factors account for almost 50% of the points. Therefore, by just focusing on these three success factors you can reduce your risk of failure by as much as 50%. Improvements to these factors can be assessed, tested, taught, and implemented. The first step in our Value Portfolio Optimization and Management Service is to assess and test the factors of success.

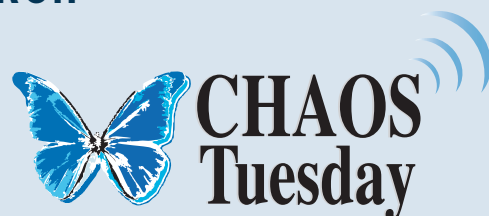
CHAOS FACTORS OF SUCCESS

CHAOS REPORT 1994		CHAOS REPORT 2014	
FACTORS OF SUCCESS	POINTS	FACTORS OF SUCCESS	POINTS
Involvement	19	Executive management support	20
Executive management support	16	User involvement	15
Clear statement of requirements	15	Clear business objectives	14
Proper planning	11	Emotional maturity	13
Realistic expectations	10	Optimization	12
Smaller project milestones	9	Agile process	10
Competent staff	8	Project management expertise	6
Ownership	6	Skilled resources	6
Clear vision and objectives	3	Execution	3
Hard-working, focused staff	3	Tools and infrastructure	1

ADDITIONAL RESOURCES AND RESEARCH

Similar research reports are

- CHAOS Manifesto 2014
- CHAOS Manifesto 2013
- CHAOS Knowledge Center
- CHAOS Charts



Many of the subjects within this report are subjects of CHAOS Tuesday, our Internet radio program. These shows include:

CHAOS Tuesday: On Fire Series

CHAOS Tuesday: Why Large Project Fail Series

CHAOS Tuesday #62: Agile Trends

CHAOS Tuesday #34: True Cost of a Project



VALUE PORTFOLIO OPTIMIZATION AND MANAGEMENT SERVICE

Our Value Portfolio Optimization and Management Service is a forward-thinking and predictive visualization of the value of your software investments.

STEP 1: GETTING TO KNOW YOU

By focusing your project portfolio on value, our service frees your organization to create value. Our service offers the following benefits:

- High Returns on Investment
- More Innovations
- Greater Stakeholder Satisfaction
- Less Management Frustration
- Reduced Project Overhead

STEP 2: PROJECT PROFILES

The Value Portfolio Optimization and Management Service offers the following features:

- Rapid
- Simple
- Comprehensive
- Inexpensive
- Comprehensible

STEP 3: OPTIMIZATION CLINIC

The one thing we are not going to do is immediately change your process or try to sell you new and cumbersome tools. It really does not matter where you are in project management maturity. Our aim is to reduce or minimize the burden placed on you by expensive tools and complex processes. We do this through our three unique items:

- Research database of 50,000 projects
- Patented optimization formula
- Our insight into a broad set of projects

Our three-step method helps you focus on things that really count.

- **STEP 1:** Project Skills and Environmental Orientation
- **STEP 2:** Individual Project Optimization & Assessments
- **STEP 3:** Value Optimization Clinic

FOLLOW-UP: Each quarter The Standish Group will visit your organization to update your portfolio. Our Value Portfolio Optimization and Management Service helps you exceed and create value.