

Death-March Projects

ICSE conference — May 20, 1997

ed@yourdon.com

<http://www.yourdon.com>

An informal survey

- ★ In general, our projects are:
- ★ Under budget and ahead of schedule
- ★ About 10% over budget, 10% behind schedule
- ★ About 50-100% over budget, 50-100% behind schedule
- ★ Substantially more than 100% over budget, 100% behind schedule

Death-March definitions

★ **Definition 1:** Project parameters exceed the norm by >50%

- ✓ Schedule compression (most common)
- ✓ Staff reduction
- ✓ Budget/resource constraints
- ✓ Functionality/performance demands

★ **Definition 2:** risk assessment (technical, legal, political, etc.) indicates >50% chance of failure

★ **Observation:** this is now the norm, not the exception

Different kinds of Death-March projects

- ★ ***Small-scale:*** 3-6 people for 3-6 months
- ★ ***Large-scale:*** 100-200 people for 3-5 years
- ★ ***Mind-boggling:*** 1,000 - 2,000 people for 7-10 years

Why Do Death-March Projects Occur?

★ The technical answer: *we're dumb*

- ✓ We don't practice software engineering
- ✓ We don't use the right methods and tools
- ✓ We don't know how to estimate our projects

★ The “don't blame us” answer:

- ✓ it's all the fault of Machiavellian managers
- ✓ Managers are evil, and they cunningly force us to take on 12-month projects with a 6-month deadline

★ The sobering reality: whatever the reason, this has now become the “norm,” not the exception

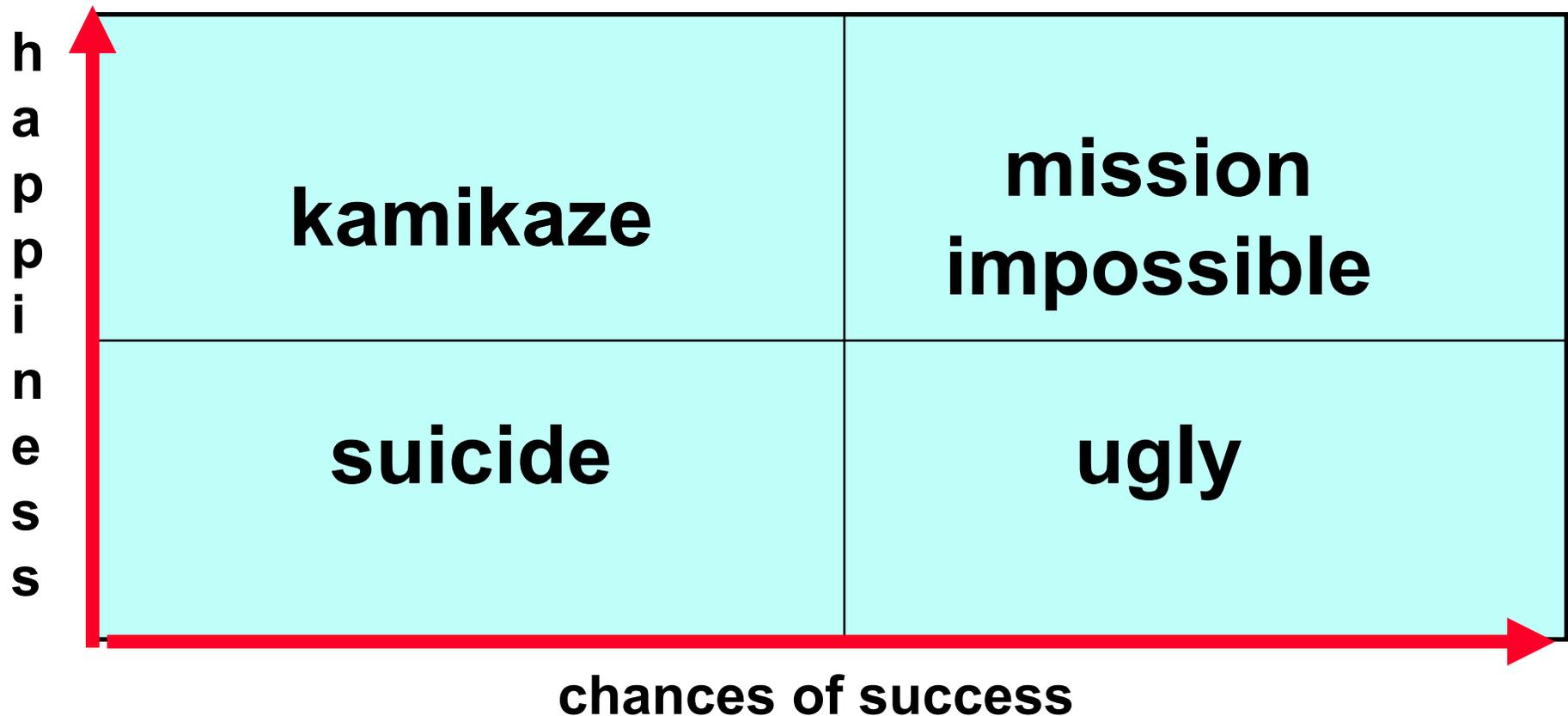
Why Do Death-Marches Occur?

- ★ **Politics, politics, politics**
- ★ **Naive promises made by marketing, senior executives, project managers, etc.**
- ★ **Naive optimism of youth: “we can do it over the weekend”**
- ★ **“Startup” mentality**
- ★ **“Marine Corps” mentality: “real men don’t need sleep”**
- ★ **Intense competition caused by globalization of markets**
- ★ **Intense competition caused by appearance of new technologies**
- ★ **Intense pressures caused by unexpected government regulations**
- ★ **Unexpected and/or unplanned crises — e.g., your vendor went bankrupt, or your 3 best programmers just died of Bubonic Plague**

Why Would Anyone Want to be Involved in a Death-March?

- ★ **Risks may be high, but so are the rewards**
- ★ **The thrill of the challenge**
- ★ **The naivete and optimism of youth**
- ★ **The alternative is unemployment**
- ★ **The alternative is bankruptcy or some other calamity**
- ★ **The project will provide training in valuable new technologies and skills, and then you can quit!**
- ★ **Revenge**

Determining the Basic Nature of the Death-March Project



Key point: get the project team members to indicate where they think *they* fit into this grid.

How to Survive?

- ★ “What is the *one thing* you feel would be most important advice for a project manager to do when involved in a “mission impossible” project?”
- ★ “What is the one thing you feel would be most important for a project manager to *avoid doing* when involved in a mission impossible project?”
- ★ It’s usually a combination of:
 - ✓ negotiating techniques
 - ✓ peopleware
 - ✓ processes (and project management)
 - ✓ tools/technology

Tools for rational Death-March negotiations

- ★ **Estimating tools — e.g., SLIM, ESTIMACS, and other commercial products**
- ★ **System dynamics models, e.g., Tarek Abdel-Hamid's model in iThink**
- ★ **Copies of *The Mythical Man-Month* for all concerned**
- ★ **Time-boxing to see how feasible/infeasible the project constraints really are**

Rational negotiations, cont'd

- ★ **Beware the temptation to give up... e.g.,**
- ★ **“We have no idea how long this project will really take, and it doesn't matter, since they've already told us the deadline...**
- ★ **...so we'll just work 7 days a week, 24 hours a day, until we drop from exhaustion. They can whip us and beat us, but we can't do any more than that...”**

Negotiating games

- ★ **Doubling and add some...**
- ★ **Reverse doubling**
- ★ **Guess the Number I'm Thinking of...**
- ★ **Double Dummy Spit**
- ★ **The X-Plus Game**
- ★ **Spanish Inquisition**
- ★ **Low Bid**
- ★ **Gotcha — throwing good money after bad**
- ★ **Chinese Water Torture**
- ★ **Smoke and Mirrors/Blinding with Science**
 - ✓ thanks to Rob Thomsett, “Double Dummy Spit, and Other Estimating Games”, *American Programmer*, June 1996

What to do when rational negotiation breaks down

- ★ **Quit (the project or the company)**
- ★ **Appeal to a higher authority**
- ★ **Determine your own constraints**
- ★ **Redefine the project as a kamikaze, suicide, etc., and make sure entire project team knows it.**
- ★ ***Key point:* project leader has to believe in the possibility of achieving project goals**
- ★ **...and must be able to convince team members without “conning” them**

Quitting and the “social contract”

- ★ **Traditional corporate culture used to be based on “a job for life” — like marriage, till death do us part...**
- ★ **Which meant that we were expected to put up with a lot of grief in death-march projects**
- ★ **But many employers have already indicated that the “social contract” is no longer valid...**
- ★ **If the employer threatens to fire you if death-march project fails, then you should be equally cold-blooded if you’re given impossible constraints for the project**

DEATH-MARCH PEOPLEWARE ISSUES

- ★ *Hiring and staffing issues: putting the best possible people on the project*
- ★ **Identifying loyalty and commitment issues: oneself, family, project, company, etc.**
- ★ **The importance of communicating urgency, priorities, constraints, risks**
- ★ **Team-building issues: team roles, “gel”, keeping teams together, etc.**

Hiring and Staffing Issues

- ★ **Strategy #1: hire superstars and turn them loose**
- ★ **Strategy #2: insist on a well-honed “mission impossible” team that has worked together before**
- ★ **Strategy #3: choose mere mortals, but make sure they know what they’re getting in for**
- ★ **Strategy #4: take whoever you’re given and convert them into a mission-impossible team**

Hiring and Staffing, cont'd

- ★ Risk increases *substantially* if project manager can't choose his/her team members
- ★ Crucial to avoid losing people during the project; highly desirable *not* to add new people during project
- ★ What to do if you can't choose your own team:
 - ✓ Quit
 - ✓ Appeal to a higher authority
 - ✓ Determine your own constraints
 - ✓ Redefine the project as a kamikaze, suicide, etc., and make sure entire project team knows it.

Question:

- ★ You're half-way through a death-march project, and the latest status report makes you realize that the odds of failure have increased significantly — to perhaps 90%.
- ★ What is the *one thing* you feel would be most important advice to do at this point, from a peopleware perspective?
- ★ What is the one thing you feel would be most important for to *avoid doing* at this point?

My answers:

- ★ **Do communicate the status to the key players. If the project is allowed to continue, then triage mercilessly — but cut non-critical “features,” not key processes (like testing).**
- ★ **Don't lie to your project team. They're not idiots — they read “Dilbert.”**

DEATH-MARCH PROCESSES

- ★ *Formal vs informal processes*
- ★ *Getting the team to “own” the process*
- ★ **SEI models vs. “mad-world” models**
- ★ **Prototyping**
- ★ **Using simulation models to explore the impact of different process strategies**
- ★ **Best practices, worst practices, and breathalyzer test**

Formal vs. Informal Processes

- ★ **Formal processes are great if you know what you're doing...**
- ★ **...and if you've done the same thing several times before**
- ★ **Watts Humphrey: "if a process can't be used in a crisis, it shouldn't be used at all."**
- ★ **But many death-march projects involve doing things that have never been done before — with teams that have never worked together before.**
- ★ **Nevertheless, team needs to agree on what processes will be formalized (e.g., change management, source code control, testing(?)), and what processes will be done on a completely ad hoc basis.**

Getting the team to “own” the process

- ★ In a death-march project, it's pointless for Methodology Police to mandate a formal software process if it's not going to be followed
- ★ Which either means that project manager must impose it in a dictatorial fashion...
- ★ ...or the team must *sincerely* agree to adopt it, because they believe in it.
- ★ A corollary: it's usually a disaster to introduce a new, unfamiliar process in an death-march project.

Question:

- ★ **Howard Rubin and I conducted a World-Wide Benchmarking Survey for the government of Canada in 1995, in which we found productivity differences of 200-to-1 at the organizational level**
- ★ **What is the *one thing* you feel would be the most important way of accomplishing this in terms of changing/adapting/improving *processes*?**
- ★ **What is the *one thing* you feel would be the most important thing to avoid in terms of changing/adapting/improving *processes*?**

My answers:

- ★ **Do reuse**
- ★ **Do triage — early and often, with full user involvement**
- ★ **Don't implement any software process that buries the developers in paper, or activities that are not part of the “by-product” of the work they need to accomplish.**

DEATH-MARCH TOOLS

- ★ *Identifying a minimal toolset*
- ★ **A checklist of tools for prototyping, CM, groupware, testing, etc.**
- ★ **The risks of choosing new tools in an death-march project.**

Identifying a minimal toolset

- ★ **Death-march projects *must* be allowed to choose its own tools, regardless of whether it conforms to organizational standards**
- ★ **...but team members need to agree on common tools *within* the project — otherwise, chaos will occur**
- ★ **Unless team has worked together before on several projects, this implies a “minimal” set of tools that everyone will use**

Tool checklist

- ★ **Email, groupware tools**
- ★ **Prototyping/RAD development tools**
- ★ **Configuration management/version control**
- ★ **CASE tools for analysis/design?**
- ★ **Requirements management tool!!**
- ★ **Testing, debugging tools**
- ★ **Project management (estimating, scheduling, PERT/GANTT, etc)**
- ★ ***toolbag of reusable components***

Risks of choosing new tools

- ★ **Some death-march projects grab new tools as a “silver bullet” to accomplish much higher levels of productivity than would otherwise be possible...**
- ★ **...but they ignore the learning curve, confusion, and political debates associated with the introduction of new tool**
- ★ **And the tools are often so new that they don't even work properly yet**
- ★ **An irony: new tool sometimes is the straw that breaks the camel's back — and project failure is then blamed on the tool**

DEATH-MARCH AS A WAY OF LIFE

- ★ **What if this is the first of many death-march projects?**
- ★ **Establishing an death-march “culture” in the organization**
- ★ **Death-march training and annual visits to the death-march “flight simulator”**

What if this is the first of many Death-March projects?

- ★ **Because the company is in the midst of ongoing crises...**
- ★ **...or management/customers have adopted this as their negotiating position**
- ★ **Or (in software/consulting firms) it's part of the company's "strategic advantage"**
- ★ **Key question: having survived one death-march project, would you do it again?**
- ★ **Important to ask this question *early***

Establishing a death-march “culture” in the company

- ★ Presumes that death-march is a conscious strategy
- ★ May have an impact on hiring strategies — e.g., preference for young, unmarried, anti-social workaholic techno-nerds
- ★ May have an impact on formal career advancement policies — e.g., “if you survive a death march for 7 years, we’ll make you a partner.”
- ★ Also impacts project management strategy — e.g., should managers plan to “burn out” their team members and discard them at the end of the death-march project?
- ★ Should be accompanied by formal training, so that new recruits understand it’s *proactive* rather than *reactive*

Death-March Training

- ★ **Currently consists of OJT and osmosis — if you survive one death-march project, you’ve become a veteran**
- ★ **Management training consists of two words: “good luck”**
- ★ **Suggestion: consider annual visits to a death-march “flight simulator”**

Conclusion

- ★ For many of us, death-march projects are inevitable in today's crazy times
- ★ Interesting question is whether your company acknowledges it...
- ★ Succeeding with death-march projects is obviously desirable, but *surviving* them is also important!
- ★ Recognize that younger generation of software people may have different attitudes about this than older generation

Death-March Projects

ICSE conference — May 20, 1997

ed@yourdon.com

<http://www.yourdon.com>