



From: www.cio.com

The Tech Disaster Awards: What You Can Learn from IT's Biggest Project Failures

– Jake Widman, Computerworld

October 09, 2008

Every year, the [Improbable Research](#) organization hands out Ig Nobel prizes to research projects that "first make people laugh, and then make them think."

For example, this year's [Ig Nobel winners](#), announced last week, include a prize in nutrition to researchers who electronically modified the sound of a potato chip to make it appear crisper and fresher than it really is and a biology prize to researchers who determined that fleas that live on a dog jump higher than fleas that live on a cat. Last year, a team won for studying how sheets become wrinkled.

That got us thinking: Though the Ig Nobels haven't given many awards to information technology (see [No Prize for IT](#) for reasons why), the history of information technology is littered with projects that have made people laugh -- if you're the type to find humor in other people's expensive failures. But have they made us think? Maybe not so much. "IT projects have terrible track records. I just don't get why people don't learn," says Mark Kozak-Holland, author of [Titanic Lessons for IT Projects](#) (that's *Titanic* as in the ship, by the way).



When you look at the reasons for project failure, "it's like a top 10 list that just repeats itself over and over again," says Holland, who is also a senior business architect and consultant with [HP Services](#). Feature creep? Insufficient training? Overlooking essential stakeholders? They're all on the list -- time and time again.

A popular management concept these days is "failing forward" -- the idea that it's OK to fail so long as you learn from your failures. In the spirit of that motto and of the Ig Nobel awards, *Computerworld* presents 11 IT projects that may have "failed" -- in some cases, failed spectacularly -- but from which the people involved were able to draw useful lessons.

You'll notice that many of them are government projects. That's not necessarily because government fails more often than the private sector, but because regulations and oversight make it harder for governments to cover up their mistakes. Private enterprise, on the other hand, is a bit better at making sure fewer people know of its failures.

So here, in chronological order, are *Computerworld's* favorite IT boondoggles, our own Ig Nobels. Feel free to laugh at them -- but try and learn something too.

IBM's Stretch project

In 1956, a group of computer scientists at IBM set out to build the world's fastest supercomputer. Five years later, they produced the IBM 7030 -- a.k.a. Stretch -- the company's first transistorized supercomputer, and delivered the first unit to the Los Alamos National Laboratory in 1961. Capable of handling a half-million instructions per second, Stretch was the fastest computer in the world and would remain so through 1964.

Nevertheless, the 7030 was considered a failure. IBM's original bid to Los Alamos was to develop a computer 100 times faster than the system it was meant to replace, and the Stretch came in only 30 to 40 times faster. Because it failed to meet its goal, IBM had to drop Stretch's price to \$7.8 million from the planned \$13.5 million, which meant the system was priced below cost. The company stopped offering the 7030 for sale, and only nine were ever built.

That wasn't the end of the story, however. "A lot of what went into that effort was later helpful to the rest of the industry," said [Turing Award winner](#) and Stretch team member Fran Allen at a recent event marking the project's 50th anniversary. Stretch introduced pipelining, memory protection, memory interleaving and other technologies that have shaped the development of computers as we know them.

Lesson learned

Don't throw the baby out with the bathwater. Even if you don't meet your project's main goals, you may be able to salvage something of lasting value from the wreckage.

[Knight-Ridder's Viewtron service](#)

The Knight-Ridder media giant was right to think that the future of home information delivery would be via computer. Unfortunately, this insight came in the early 1980s, and the computer they had in mind was an expensive dedicated terminal.

Knight-Ridder launched its [Viewtron](#) version of [videotex](#) -- the in-home information-retrieval service -- in Florida in 1983 and extended it to other U.S. cities by 1985. The service offered banking, shopping, news and ads delivered over a custom terminal with color graphics capabilities beyond those of the typical PC of the time. But Viewtron never took off: It was meant to be the the "McDonald's of videotex" and at the same time cater to upmarket consumers, according to a Knight-Ridder representative at the time who apparently didn't notice the contradictions in that goal.

Related Stories

[Don't Believe the Hype: The 21 Biggest Technology Flops](#)

[Your Votes Tallied: The Biggest Tech Flop of All Time](#)

A Viewtron terminal cost \$900 initially (the price was later dropped to \$600 in an attempt to stimulate demand); by the time the company made the service available to anyone with a standard PC, videotex's moment had passed.

Viewtron only attracted 20,000 subscribers, and by 1986, it had been canceled. But not before it cost Knight-Ridder \$50 million. *The New York Times* business section wrote, with admirable understatement, that Viewtron "tried to offer too much to too many people who were not overly interested."

Nevertheless, *BusinessWeek* concluded at the time, "Some of the nation's largest media, technology and financial services companies ... remain convinced that some day, everyday life will center on computer screens in the home." Can you imagine?

Lesson learned

Sometimes you can be so far ahead of the curve that you fall right off the edge.

DMV projects -- California and Washington

Two Western states spent the 1990s attempting to computerize their departments of motor vehicles, only to abandon the projects after spending millions of dollars. First was California, which in 1987 embarked on a five-year, \$27 million plan to develop a system for keeping track of the state's 31 million drivers' licenses and 38 million vehicle registrations. But the state solicited a bid from just one company and awarded the contract to Tandem Computers. With Tandem supplying the software, the state was locked into buying Tandem hardware as well, and in 1990, it purchased six computers at a cost of \$11.9 million.

That same year, however, tests showed that the new system was slower than the one it was designed to replace. The state forged ahead, but in 1994, it was finally forced to abandon what the *San Francisco Chronicle* described as "an unworkable system that could not be fixed without the expenditure of millions more." In that May 1994 article, the *Chronicle* described it as a "failed \$44 million computer project." In an August article, it was described as a \$49 million project, suggesting that the project continued to cost money even after it was shut down. A state audit later concluded that the DMV had "violated numerous contracting laws and regulations."

Lesson learned

Regulations are there for a reason, especially ones that keep you from doing things like placing your future in the hands of one supplier.

Meanwhile, the state of Washington was going through its own nightmare with its License Application Mitigation Project (LAMP). Begun in 1990, LAMP was supposed to cost \$16 million over five years and automate the state's vehicle registration and license renewal processes. By 1992, the projected cost had grown to \$41.8 million; a year later, \$51 million; by 1997, \$67.5 million. Finally, it became apparent that not only was the cost of installing the system out of control, but it would also cost six times as much to run every year as the system it was replacing. Result: plug pulled, with \$40 million spent for nothing.

Lesson learned

When a project is obviously doomed to failure, get out sooner rather than later.

FoxMeyer ERP program

In 1993, FoxMeyer Drugs was the fourth largest distributor of pharmaceuticals in the U.S., worth \$5 billion. In an attempt to increase efficiency, FoxMeyer purchased an SAP system and a warehouse automation system and hired Andersen Consulting to integrate and implement the two in what was supposed to be a \$35 million project. By 1996, the company was bankrupt; it was eventually sold to a competitor for a mere \$80 million.

The reasons for the failure are familiar. First, FoxMeyer set up an unrealistically aggressive time line -- the entire system was supposed to be implemented in 18 months. Second, the warehouse employees whose jobs were affected -- more accurately, threatened -- by the automated system were not supportive of the project, to say the least. After three existing warehouses were closed, the first warehouse to be automated was plagued by sabotage, with inventory damaged by workers and orders going unfilled.

Finally, the new system turned out to be less capable than the one it replaced: By 1994, the SAP system was processing only 10,000 orders a night, compared with 420,000 orders under the old mainframe. FoxMeyer also alleged that both Andersen and SAP used the automation project as a training tool for junior employees, rather than assigning their best workers to it.

In 1998, two years after [filing for bankruptcy](#), FoxMeyer sued Andersen and SAP for \$500 million each, claiming it had paid twice the estimate to get the system in a quarter of the intended sites. The suits were settled and/or dismissed in 2004.

Lesson learned

No one plans to fail, but even so, make sure your operation can survive the failure of a project.

Apple's Copland operating system

It's easy to forget these days just how desperate [Apple Computer](#) was during the 1990s. When Microsoft Windows 95 came out, it arrived with multitasking and dynamic memory allocation, neither of which was available in the existing Mac System 7. [Copland](#) was Apple's attempt to develop a new operating system in-house; actually begun in 1994, the new OS was intended to be released as System 8 in 1996.

Copland's development could be the poster child for feature creep. As the new OS came to dominate resource allocation within Apple, project managers began protecting their fiefdoms by pushing for their products to be incorporated into System 8. Apple did manage to get one developers' release out in late 1996, but it was wildly unstable and did little to increase anyone's confidence in the company.

Before another developer release could come out, Apple made the decision to cancel Copland and look outside for its new operating system; the outcome, of course, was the purchase of NeXT, which supplied the technology that became OS X.

Copland did not die in vain. Some of the technology seen in demos eventually turned up in OS X. And even before that, some Copland features wound up in System 8 and 9, including a multithreaded Finder that provided something like true preemptive multitasking.

Lesson learned

Project creep is a killer. Keep your project's goals focused.

Sainsbury's warehouse automation

[Sainsbury's](#), the British supermarket giant, was determined to install an automated fulfillment system in its Waltham Point distribution center in Essex. Waltham Point was the distribution center for much of London and southeast England, and the barcode-based fulfillment system would increase efficiency and streamline operations. If it worked, that is.

Installed in 2003, the system promptly ran into what were then described as "horrendous" barcode-reading errors. Regardless, in 2005 the company claimed the system was operating as intended. Two years later, the entire project was scrapped, and Sainsbury's wrote off £150 million in IT costs. (That's \$265,335,000 calculated by today's exchange rate, enough to buy a lot of groceries.)

Lesson learned

A square peg in a round hole won't fit any better as time goes on. Put another way -- problems that go unaddressed at rollout will only get worse, not better, over time.

Canada's gun registration system

In June 1997, [Electronic Data Systems](#) and U.K.-based SHL Systemhouse started work on a Canadian national firearm registration system. The original plan was for a modest IT project that would cost taxpayers only \$2 million -- \$119 million for implementation, offset by \$117 million in licensing fees.

But then politics got in the way. Pressure from the gun lobby and other interest groups resulted in more than 1,000 change orders in just the first two years. The changes involved having to interface with the computer systems of more than 50 agencies, and since that integration wasn't part of the original contract, the government had to pay for all the extra work. By 2001, the costs had ballooned to \$688 million, including \$300 million for support.

But that wasn't the worst part. By 2001, the annual maintenance costs alone were running \$75 million a year. A 2002 audit estimated that the program would wind up costing more than \$1 billion by 2004 while generating revenue of only \$140 million, giving rise to its nickname: "the billion-dollar boondoggle."

The registry is still in operation and still a political football. Both the Canadian Police Association and the Canadian Association of Chiefs of Police have spoken in favor of it, while opponents argue that the money would be better spent otherwise.

Lesson learned

Define your project scope and freeze specifications before the requests for changes get out of hand.

Three current projects in danger

At least Canada managed to get its project up and running. Our final three projects, courtesy of the U.S. government, are still in development -- they have failed in many ways already, but can still fail more. Will anyone learn anything from them? After reading these other stories, we know how we'd bet.

[FBI Virtual Case File](#)

In 2000, the FBI finally decided to get serious about automating its case management and forms processing, and in September of that year, Congress approved \$379.8 million for the Information Technology Upgrade Project. What started as an attempt to upgrade the existing Automated Case Support system became, in 2001, a project to develop an entirely new system, the Virtual Case File (VCS), with a contract awarded to [Science Applications International Corp.](#)

That sounds reasonable until you read about the development time allotted (a mere 22 months), the rollout plans (a "flash cutover," in which the new system would come online and the old one would go offline over a single weekend), and the system requirements (an 800-page document specifying details down to the layout of each page).

By late 2002, the FBI needed another \$123.2 million for the project. And change requests started to take a toll: According to SAIC, those totaled about 400 by the end of 2003. In April 2005, SAIC delivered 700,000 lines of code that the FBI considered so bug-ridden and useless that the agency decided to scrap the entire VCS project. A later audit blamed factors such as poorly defined design requirements, an overly ambitious schedule and the lack of an overall plan for purchases and deployment.

The FBI did use some of what it learned from the VCF disaster in its current Sentinel project. Sentinel, now scheduled for completion in 2012, should do what VCF was supposed to do using off-the-shelf,

Web-based software.

Homeland Security's virtual fence

The [U.S. Department of Homeland Security](#) is bolstering the U.S. Border Patrol with a network of radar, satellites, sensors and communication links -- what's commonly referred to as a "virtual fence." In September 2006, a contract for this [Secure Border Initiative Network](#) (SBI-net, not to be confused with Skynet) was awarded to Boeing, which was given \$20 million to construct a 28-mile pilot section along the Arizona-Mexico border.

But early this year, Congress learned that the pilot project was being delayed because users had been excluded from the process and the complexity of the project had been underestimated. (Sound familiar?) In February 2008, the Government Accountability Office reported that the radar meant to detect aliens coming across the border could be set off by rain and other weather, and the cameras meant to zoom in on subjects sent back images of uselessly low resolution for objects beyond 3.1 miles. Also, the pilot's communications system interfered with local residents' WiFi networks -- not good PR.

In April, DHS announced that the surveillance towers of the pilot fence did not meet the Border Patrol's goals and were being replaced -- a story picked up by the Associated Press and widely reported in the mainstream media. But the story behind the story is less clear. The DHS and Boeing maintain the original towers were only temporary installations for demonstration purposes. Even so, the project is already experiencing delays and cost overruns, and in April, SBI-net program manager Kirk Evans [resigned](#), citing lack of a system design as just one specific concern. Not an auspicious beginning.

Census Bureau's handheld units

Back in 2006, the [U.S. Census Bureau](#) made a plan to use 500,000 handheld devices -- purchased from [Harris Corp.](#) under a \$600 million contract -- to help automate the 2010 census. Now, though, the cost has more than doubled, and their use is [going to be curtailed](#) in 2010 -- but the Census Bureau is moving ahead with the project anyway.

During a rehearsal for the census conducted in the fall of 2007, according to the GAO, field staff found that the handheld devices froze or failed to retrieve mapping coordinates (see [Hard questions needed to save projects](#) for details). Furthermore, multiple devices had the same identification number, which meant they would overwrite one another's data.

After the rehearsal, a representative of [Mitre Corp.](#), which advises the bureau on IT matters, brought notes to a meeting with the bureau's representative that read, "It is not clear that the system will meet Census' operational needs and quality goals. The final cost is unpredictable. Immediate, significant changes are required to rescue the program. However, the risks are so large considering the available time that we recommend immediate development of contingency plans to revert to paper operations."

There you have it, a true list of IT Ig Nobels: handheld computers that don't work as well as pencil and paper, new systems that are slower and less capable than the old ones they're meant to replace. Perhaps the overarching lesson is one that project managers should have learned at their mothers' knees: Don't bite off more than you can chew.

No Prize for IT

Information technology has rarely won an [Ig Nobel award](#) in the 18 years the prizes have been doled out by the [Improbable Research](#) organization.

Should we take the snub personally?

Marc Abrahams, the editor of *Improbable Research*, the organization's blog, says he thinks IT's relative absence is simply because the field is younger than other disciplines. "Certainly IT offers the same level of absurdity as other areas of research," he says comfortably.

He points out that Murphy's Law, whose three "inventors" (John Paul Stapp, Edward A. Murphy, Jr. and George Nichols,) were honored with an Ig Nobel in 2003, sprang from an IT-like project in the late 1940s. Murphy was an electrical engineer who was brought in to help the Air Force figure out why safety tests they were conducting weren't producing any results. Murphy discovered that the electronic monitoring systems had been installed "backwards and upside down," according to Abrahams, which discovery caused him to mutter the first version of the law that bears his name.

Other Ig Nobels drawn from the world of technology include:

2001: John Keogh of Hawthorn, Victoria, Australia, won in the Technology category for patenting the wheel; he shared the award with the Australian Patent Office, which granted him [Innovation Patent #2001100012](#) (pdf) for a "circular transportation facilitation device."

2000: Chris Niswander of Tucson, Ariz., won a Computer Science Ig Nobel for his development of [PawSense](#), software that can tell when a cat is walking across your keyboard and make a sound to scare it off.

1997: Sanford Wallace -- yes, that Sanford Wallace -- of Cyber Promotions takes the Communications Ig Nobel for being the [Spam King](#).

The Ig Nobels, it must be remembered, aren't into value judgments.

San Francisco-based Widman is a frequent contributor to Computerworld.

© 2007 Computerworld Inc.



Get up to 34% more performance per watt* with the new Six-Core AMD Opteron™ processor.

▲ Buy today.

