

1 Therac-25

Therac-25 was a radiation therapy machine produced by Atomic Energy of Canada Limited. It was involved with at least six known accidents between 1985 and 1987, in which patients were given massive overdoses of radiation, which were in some cases on the order of tens of thousands of rads. At least five patients died of the overdoses. These accidents highlighted the dangers of software control of safety-critical systems.

The machine had two treatment modes: Direct electron-beam therapy, which used high doses of energy over short periods of time, and Soft X-ray therapy, which used X-rays derived from the electron beam via a “target”, a device which converts electron beams into X-rays.

The accidents occurred when the high-energy electron-beam was activated without the target having been rotated into place; the machine’s software did not detect that this had occurred, and did not thenceforth determine that the patient was receiving a potentially lethal dose of radiation, or prevent this from occurring. The very high energy electron-beam directly struck the patients causing the feeling of an intense electric shock and the occurrence of thermal and radiation burns. In some cases, the injured patients died later from radiation poisoning.

Researchers who investigated the accidents found many contributing causes:

- * The software code was not independently reviewed.
- * The software design was not documented with enough detail to support reliability modeling.
- * The system documentation did not adequately explain error codes.
- * AECL personnel were at first dismissive of complaints.
- * The design did not have any hardware interlocks to prevent the electron-beam from operating in its high-energy mode without the target in place.
- * Software from older models had been reused without properly considering the hardware differences.
- * The older models had included hardware interlocks; when the bug manifested in these models, they shut down, which was seen as a mere annoyance and never investigated.

- * The software assumed that sensors always worked correctly, since there was no way to verify them.
- * The equipment control task did not properly synchronize with the operator interface task, so that race conditions occurred if the operator changed the setup too quickly. This was evidently missed during testing, since it took some practice before operators were able to work quickly enough for the problem to occur.
- * Arithmetic overflows could cause the software to bypass safety checks.
- * The software was written in assembly language. While this was more common at the time than it is today, assembly language is harder to debug than high-level languages.

2 DIA Baggage System

Denver International Airport's computerized baggage system which was supposed to reduce flight delays, shorten waiting times at luggage carousels, and save airlines in labor costs, turned into an unmitigated failure. An opening originally scheduled for October 31, 1993 with a single system for all three concourses turned into a February 28, 1995 opening with separate systems for each concourse, with varying degrees of automation. The system's \$186 million in original construction costs grew by \$1 million per day during months of modifications and repairs. Incoming flights never made use of the system, and only United, DIA's dominant airline, used it for outgoing flights. The system never worked well, and in August 2005, it became public knowledge that United would abandon the system, a decision that would save them \$1 million in monthly maintenance costs.

3 Ariane 5 Rocket

Ariane 5 Rocket Flight 501, which took place on June 4, 1996, was the first test flight of the European Space Agency's Ariane 5 expendable launch system. It was not successful; the rocket tore itself apart 40 seconds after launch because of a malfunction in the control software, making the fault one of the most expensive computer bugs in history. The breakup caused the loss of the payload: four Cluster mission spacecraft.

The Ariane 5 software reused the specifications from the Ariane 4, but the Ariane 5's flight path was considerably different and beyond the range for which the reused code had been designed. Specifically, the Ariane 5's greater acceleration caused the back-up and primary inertial guidance computers to crash, after which the launcher's nozzles were directed by spurious data. Pre-flight tests had never been performed on the re-alignment code under simulated Ariane 5 flight conditions, so the error was not discovered before launch.

Because of the different flight path, a data conversion from a 64-bit floating point to 16-bit signed integer value caused a hardware exception (more specifically, an arithmetic overflow, as the floating point number had a value too large to be represented by a 16-bit signed integer). Efficiency considerations had led to the disabling of the software handler (in Ada code) for this error trap, although other conversions of comparable variables in the code remained protected. This led to a cascade of problems, culminating in destruction of the entire flight.

4 Duke Nukem Forever

Duke Nukem Forever (DNF) is a yet-to-be-released first-person shooter video game being developed by 3D Realms, and is the next game in the popular Duke Nukem series. It is notorious for its protracted development, which has been ongoing since 1997. Wired News awarded Duke Nukem Forever the Vaporware Lifetime Achievement Award in 2003. While there has never been an official release date, developers originally hinted that the intended release date was 1998. The game has been jokingly referred to as “Duke Nukem If Ever” or “Duke Nukem Taking Forever.”

Many speculate that this title will never see the light of day. Internet forum comments made by lead designer George Broussard in 2004 suggested that development was progressing reasonably well, even though he later said that almost all of the previous generation of game content had been scrapped as of early 2003. 3D Realms cites several factors which have contributed to the game’s late release. They primarily blame the delays on several project “restarts” (starting the project from scratch), as well as engine changes, in order to take account of the swiftly-advancing pace of home computer development. They also lay some early blame on attempting multiple in-house projects, which split internal focus too much for such a small developer.

5 FBI Virtual Case File

Modern case-management capabilities remain atop the Federal Bureau of Investigation's IT Most Wanted List as the bureau's \$170 million Virtual Case File (VCF) project faces imminent termination. A forthcoming report by the Justice Department's Inspector General will read like an indictment of FBI executive and IT management, according to Government Computer News. GCN quotes a Dec. 20 draft as saying that VCF, designed to manage documentation for investigations and prosecutions, "will not meet FBI needs," that "the FBI has no clear timetable or prospect for completing VCF" and that the planned replacement, the Federal Investigative Case Management System, is "unlikely to benefit substantially from the VCF [project] from a technical or engineering standpoint." VCF would have been the third leg of the bureau's late and overbudget Trilogy IT modernization program. The Government Accountability Office (GAO) told Congress that the FBI's weak IT management contributed to Trilogy's 21-month delay and \$120 million cost overrun.

The bureau's plan for a "flash cutover" from the old Automated Case Support system without testing or provisions for reversion is a risky approach that's "nearly guaranteed to cause mission-critical failures and further delays, with implications for training, performance, coherence, internal morale, public image and cost to recovery."

FBI application development processes exclude users. "It is essentially impossible ... to anticipate in detail and in advance all [user] requirements and specifications." The bureau should adopt extensive prototyping and usability testing with real users. Senior FBI management improperly delegated project leadership to contractors, lacking its own adequate skill base for IT modernization. Architectural incoherence, poor planning, mission creep, unrealistic development processes and deployment expectations, improper delegation and inadequate management also plague private industry. But missed profit targets hardly compare with unsupported law-enforcement and counterterrorism needs.

The FBI reportedly will rely on commercial or existing government-owned systems to replace VCF, salvaging little from the \$170 million effort beyond a better understanding of requirements. There's no word yet whether the bureau will be making fundamental changes to the flawed IT management and contracting processes at the root of the VCF debacle.