Appendix

A

Medical Devices: The Therac-25 Story

A.1 Introduction

Between June 1985 and January 1987, a computer-controlled radiation therapy machine, called the Therac-25, massively overdosed six people. These accidents have been described as the worst in the 35-year history of medical accelerators [284].

A detailed accident investigation, drawn from publicly available documents, can be found in Leveson and Turner [187]. The following account is taken from this report and includes both the factors involved in the overdoses themselves and the attempts by the users, manufacturers, and governments to deal with them. Because this accident was never officially investigated, only partial information on the Therac-25 software development, management, and quality control procedures is available. What is included below has been gleaned from law suits and depositions, government records, and copies of correspondence and other material obtained from the U.S. Food and Drug Administration (FDA), which regulates these devices.

A.2 Background

Medical linear accelerators (linacs) accelerate electrons to create high-energy beams that can destroy tumors with minimal impact on the surrounding healthy tissue. Relatively shallow tissue is treated with the accelerated electrons; to reach deeper tissue, the electron beam is converted into X-ray photons.

In the early 1970s, Atomic Energy of Canada Limited (AECL)[1] and a French company called CGR went into business together building linear accelerators. The products of this cooperation were (1) the Therac-6, a 6 million electron volt (MeV) accelerator capable of producing X-rays only and later (2) the Therac-20, a 20 MeV, dual-mode (X-rays or electrons) accelerator. Both were versions of older CGR machines, the Neptune and Sagittaire, respectively, which were augmented with computer control using a DEC PDP-11 minicomputer. We know that some of the old Therac-6 software routines were reused in the Therac-20 and that CGR developed the initial software.

Software functionality was limited in both machines: The computer merely added convenience to the existing hardware, which was capable of standing alone. Industry-standard hardware safety features and interlocks in the underlying machines were retained.

The business relationship between AECL and CGR faltered after the Therac-20 effort. Citing competitive pressures, the two companies did not renew their cooperative agreement when scheduled in 1981.

In the mid-1970s, AECL had developed a radical new "double pass" concept for electron acceleration. A double-pass accelerator needs much less space to develop comparable energy levels because it folds the long physical mechanism required to accelerate the electrons, and it is more economical to produce. Using this double-pass concept, AECL designed the Therac-25, a dual-mode linear accelerator that can deliver either photons at 25 MeV or electrons at various energy levels.

Compared with the Therac-20, the Therac-25 is notably more compact, more versatile, and arguably easier to use. The higher energy takes advantage of the phenomenon of *depth dose*: As the energy increases, the depth in the body at which maximum dose build-up occurs also increases, sparing the tissue above the target area. Economic advantages also come into play for the customer, since only one machine is required for both treatment modalities (electrons and photons).

Several features of the Therac-25 are important in understanding the accidents. First, like the Therac-6 and the Therac-20, the Therac-25 is controlled by a PDP-11 computer. However, AECL designed the Therac-25 to take advantage of computer control from the outset; they did not build on a stand-alone machine. The Therac-6 and Therac-20 had been designed around machines that already had histories of clinical use without computer control.

In addition, the Therac-25 software has more responsibility for maintaining safety than the software in the previous machines. The Therac-20 has independent protective circuits for monitoring the electron-beam scanning plus mechanical interlocks for policing the machine and ensuring safe operation. The Therac-25 relies more on software for these functions. AECL took advantage of the computer's abilities to control and monitor the hardware and decided not to duplicate all the existing hardware safety mechanisms and interlocks.

[1] AECL was an arms-length entity, called a crown corporation, of the Canadian government. Since the time of the incidents related in this paper, AECL Medical, a division of AECL, was privatized and is now called Theratronics International, Ltd. Currently, the primary business of AECL is the design and installation of nuclear reactors.

Some software for the machines was interrelated or reused. In a letter to a Therac-25 user, the AECL quality assurance manager said, "The same Therac-6 package was used by the AECL software people when they started the Therac-25 software. The Therac-20 and Therac-25 software programs were done independently starting from a common base" [187]. The reuse of Therac-6 design features or modules may explain some of the problematic aspects of the Therac-25 software design. The quality assurance manager was apparently unaware that some Therac-20 routines were also used in the Therac-25; this was discovered after a bug related to one of the Therac-25 accidents was found in the Therac-20 software.

AECL produced the first hardwired prototype of the Therac-25 in 1976, and the completely computer-controlled commercial version was available in late 1982.

Turntable Positioning. The Therac-25 turntable design plays an important role in the accidents. The upper turntable (see Figure A.1) rotates accessory equipment into the beam path to produce two therapeutic modes: electron mode and photon mode. A third position (called the field light position) involves no beam at all, but rather is used to facilitate correct positioning of the patient. Because the accessories appropriate to each mode are physically attached to the turntable, proper operation of the Therac-25 is heavily dependent on the turntable position, which is monitored by three microswitches.

The raw, highly concentrated accelerator beam is dangerous to living tissue. In electron therapy, the computer controls the beam energy (from 5 to 25 MeV) and current, while scanning magnets are used to spread the beam to a safe, therapeutic concentration. These scanning magnets are mounted on the turntable and moved into proper position by the computer. Similarly, an ion chamber to measure electrons is mounted on the turntable and also moved into position by the computer. In addition, operator-mounted electron trimmers can be used to shape the beam if necessary.

For X-ray (or photon) therapy, only one energy level is available: 25 MeV. Much greater electron-beam current is required for X-ray mode (some 100 times greater than that for electron therapy) [284] to produce comparable output. Such a high dose-rate capability is required because a "beam flattener" is used to produce a uniform treatment field. This flattener, which resembles an inverted ice cream cone, is a very efficient attenuator; thus, to get a reasonable treatment dose rate out of the flattener, a very high input dose rate is required. If the machine should produce a photon beam with the beam flattener not in position, a high output dose to the patient results. This is the basic hazard of dual-mode machines: If the turntable is in the wrong position, the beam flattener will not be in place.

In the Therac-25, the computer is responsible for positioning the turntable

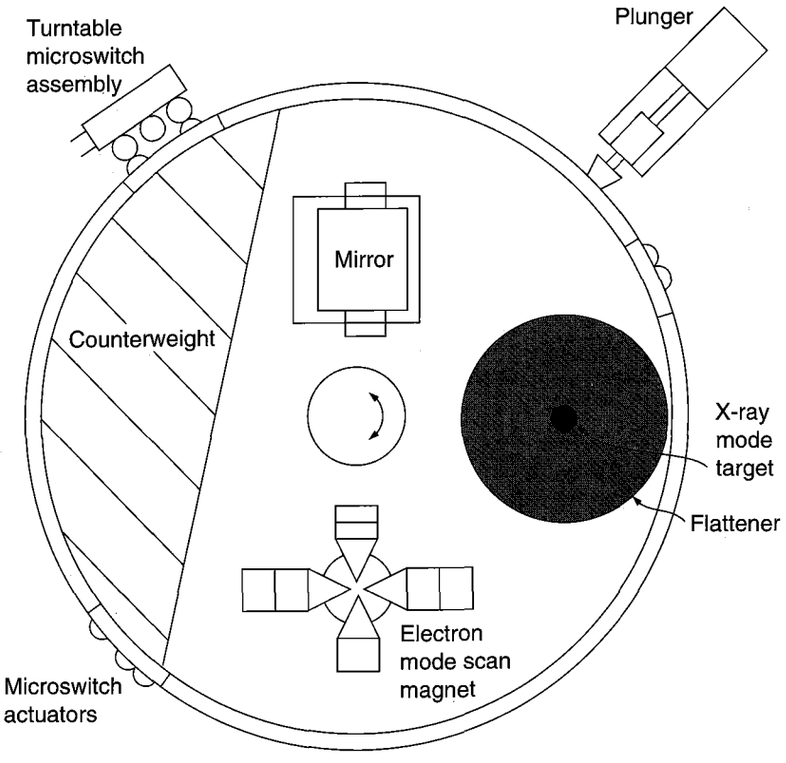

FIGURE A.1
Upper turntable assembly.

(and for checking the turntable position) so that a target, flattening filter, and X-ray ion chamber are directly in the beam path. With the target in place, electron bombardment produces X-rays. The X-ray beam is shaped by the flattening filter and measured by the X-ray ion chamber.

No accelerator beam is expected in the third or field light turntable position. A stainless steel mirror is placed in the beam path and a light simulates the beam. This lets the operator see precisely where the beam will strike the patient and make necessary adjustments before treatment starts. There is no ion chamber in place at this turntable position, since no beam is expected.

Traditionally, electromechanical interlocks have been used on these types of equipment to ensure safety—in this case, to ensure that the turntable and attached equipment are in the correct position when treatment is started. In the Therac-25, software checks were substituted for many of the traditional hardware interlocks.


```
PATIENT NAME      : TEST
TREATMENT MODE : FIX          BEAM TYPE: X      ENERGY (MeV): 25

                              ACTUAL        PRESCRIBED
      UNIT RATE/MINUTE          0             200
      MONITOR UNITS           50  50          200
      TIME (MIN)              0.27            1.00

GANTRY ROTATION (DEG)          0.0             0      VERIFIED
COLLIMATOR ROTATION (DEG)    359.2           359      VERIFIED
COLLIMATOR X (CM)             14.2            14.3    VERIFIED
COLLIMATOR Y (CM)             27.2            27.3    VERIFIED
WEDGE NUMBER                     1               1    VERIFIED
ACCESSORY NUMBER                 0               0    VERIFIED

DATE  : 84-OCT-26   SYSTEM : BEAM READY   OP.MODE : TREAT   AUTO
TIME  : 12:55:8     TREAT  : TREAT PAUSE            : X-RAY  173777
OPR ID : T25V02-R03 REASON : OPERATOR     COMMAND:
```

FIGURE A.2
Operator interface screen layout.

The Operator Interface. The description of the operator interface here applies to the version of the software used during the accidents. Changes made as a result of an FDA recall are described later.

The Therac-25 operator controls the machine through a DEC VT100 terminal. In the general case, the operator positions the patient on the treatment table, manually sets the treatment field sizes and gantry rotation, and attaches accessories to the machine. Leaving the treatment room, the operator returns to the console to enter the patient identification, treatment prescription (including mode or beam type, energy level, dose, dose rate, and time), field sizing, gantry rotation, and accessory data. The system then compares the manually set values with those entered at the console. If they match, a *verified* message is displayed and treatment is permitted. If they do not match, treatment is not allowed to proceed until the mismatch is corrected. Figure A.2 shows the screen layout.

When the system was first built, operators complained that it took too long to enter the treatment plan. In response, AECL modified the software before the first unit was installed: Instead of reentering the data at the keyboard, operators could simply use a carriage return to copy the treatment site data [225]. A quick

series of carriage returns would thus complete the data entry. This modification was to figure in several of the accidents.

The Therac-25 could shut down in two ways after it detected an error condition. One was a *treatment suspend*, which required a complete machine reset to restart. The other, not so serious, was a *treatment pause*, which only required a single key command to restart the machine. If a *treatment pause* occurred, the operator could press the Ⓟ key to "proceed" and resume treatment quickly and conveniently. The previous treatment parameters remained in effect, and no reset was required. This feature could be invoked a maximum of five times before the machine automatically suspended treatment and required the operator to perform a system reset.

Error messages provided to the operator were cryptic, and some merely consisted of the word MALFUNCTION followed by a number from 1 to 64 denoting an analog/digital channel number. According to an FDA memorandum written after one accident:

> The operator's manual supplied with the machine does not explain nor even address the malfunction codes. The Maintance [sic] Manual lists the various malfunction numbers but gives no explanation. The materials provided give no indication that these malfunctions could place a patient at risk.
>
> The program does not advise the operator if a situation exists wherein the ion chambers used to monitor the patient are saturated, thus are beyond the measurement limits of the instrument. This software package does not appear to contain a safety system to prevent parameters being entered and intermixed that would result in excessive radiation being delivered to the patient under treatment.

An operator involved in one of the accidents testified that she had become insensitive to machine malfunctions. Malfunction messages were commonplace and most did not involve patient safety. Service technicians would fix the problems or the hospital physicist would realign the machine and make it operable again. She said,

> It was not out of the ordinary for something to stop the machine. . . . It would often give a low dose rate in which you would turn the machine back on. . . . They would give messages of low dose rate, V-tilt, H-tilt, and other things; I can't remember all the reasons it would stop, but there was a lot of them.

A radiation therapist at another clinic reported that an average of 40 dose-rate malfunctions, attributed to underdoses, occurred on some days.

The operator further testified that during instruction she had been taught that there were "so many safety mechanisms" that she understood it was virtually impossible to overdose a patient.

Hazard Analysis. In March 1983, AECL performed a safety analysis on the Therac-25. This analysis was in the form of a fault tree and apparently excluded

the software. According to the final report, the analysis made several assumptions about the computer and its software:

1. Programming errors have been reduced by extensive testing on a hardware simulator and under field conditions on teletherapy units. Any residual software errors are not included in the analysis.
2. Program software does not degrade due to wear, fatigue, or reproduction process.
3. Computer execution errors are caused by faulty hardware components and by "soft" (random) errors induced by alpha particles and electromagnetic noise.

The fault tree resulting from this analysis does appear to include computer failure, although apparently, judging from the basic assumptions above, it considers hardware failures only. For example, in one OR gate leading to the event of getting the wrong energy, a box contains "Computer selects wrong energy," and a probability of 10^{-11} is assigned to this event. For "Computer selects wrong mode," a probability of 4×10^{-9} is given. The report provides no justification of either number.

A.3 Events

Eleven Therac-25s were installed: five in the United States and six in Canada. Six accidents occurred between 1985 and 1987, when the machine was finally recalled to make extensive design changes. These changes include adding hardware safeguards against software errors.

Related problems were found in the Therac-20 software, but they were not recognized until after the Therac-25 accidents because the Therac-20 includes hardware safety interlocks. Thus, no injuries resulted.

A.3.4 East Texas Cancer Center, March 1986

More is known about the Tyler, Texas, accidents than the others because of the diligence of the Tyler hospital physicist, Fritz Hager, without whose efforts the understanding of the software problems may have been delayed even further.

The Therac-25 had been at the East Texas Cancer Center (ETCC) for two years before the first serious accident, and more than 500 patients had been treated. On March 21, 1986, a male patient came into ETCC for his ninth treatment on the Therac-25, one of a series prescribed as followup to the removal of a tumor from his back.

This treatment was to be a 22 MeV electron beam treatment of 180 rads on

the upper back and a little to the left of his spine, for a total of 6,000 rads over six and a half weeks. He was taken into the treatment room and placed face down on the treatment table. The operator then left the treatment room, closed the door, and sat at the control terminal.

The operator had held this job for some time, and her typing efficiency had increased with experience. She could quickly enter prescription data and change it conveniently with the Therac's editing features. She entered the patient's prescription data quickly, then noticed that she had typed "x" (for X-ray) when she had intended "e" (for electron) mode. This was a common mistake as most of the treatments involved X-rays, and she had gotten used to typing this. The mistake was easy to fix; she merely used the ⓘ key to edit the mode entry.

Because the other parameters she had entered were correct, she hit the return key several times and left their values unchanged. She reached the bottom of the screen, where it was indicated that the parameters had been VERIFIED and the terminal displayed BEAM READY, as expected. She hit the one-key command, Ⓑ for *beam on*, to begin the treatment. After a moment, the machine shut down and the console displayed the message MALFUNCTION 54. The machine also displayed a TREATMENT PAUSE, indicating a problem of low priority. The sheet on the side of the machine explained that this malfunction was a "dose input 2" error. The ETCC did not have any other information available in its instruction manual or other Therac-25 documentation to explain the meaning of MALFUNCTION 54. An AECL technician later testified that "dose input 2" meant that a dose had been delivered that was either too high or too low. The messages had been expected to be used only during internal company development.

The machine showed a substantial underdose on its dose monitor display—6 monitor units delivered whereas the operator had requested 202 monitor units. She was accustomed to the quirks of the machine, which would frequently stop or delay treatment; in the past, the only consequences had been inconvenience. She immediately took the normal action when the machine merely paused, which was to hit the Ⓟ key to proceed with the treatment. The machine promptly shut down with the same MALFUNCTION 54 error and the same underdose shown by the dosimetry.

The operator was isolated from the patient, since the machine apparatus was inside a shielded room of its own. The only way that the operator could be alerted to patient difficulty was through audio and video monitors. On this day, the video display was unplugged and the audio monitor was broken.

After the first attempt to treat him, the patient said that he felt as if he had received an electric shock or that someone had poured hot coffee on his back: He felt a thump and heat and heard a buzzing sound from the equipment. Since this was his ninth treatment, he knew that this was not normal. He began to get up from the treatment table to go for help. It was at this moment that the operator hit the Ⓟ key to proceed with the treatment. The patient said that he felt like his arm was being shocked by electricity and that his hand was leaving his body. He went to the treatment room door and pounded on it. The operator was

shocked and immediately opened the door for him. He appeared visibly shaken and upset.

The patient was immediately examined by a physician, who observed intense reddening of the treatment area, but suspected nothing more serious than electric shock. The patient was discharged and sent home with instructions to return if he suffered any further reactions. The hospital physicist was called in, and he found the machine calibration within specifications. The meaning of the malfunction message was not understood. The machine was then used to treat patients for the rest of the day.

In actuality, but unknown to anyone at that time, the patient had received a massive overdose, concentrated in the center of the treatment location. After-the-fact simulations of the accident revealed possible doses of 16,500 to 25,000 rads in less than 1 second over an area of about 1 cm.

Over the weeks following the accident, the patient continued to have pain in his neck and shoulder. He lost the function of his left arm and had periodic bouts of nausea and vomiting. He was eventually hospitalized for radiation-induced myelitis of the cervical cord causing paralysis of his left arm and both legs, left vocal cord paralysis (which left him unable to speak), neurogenic bowel and bladder, and paralysis of the left diaphragm. He also had a lesion on his left lung and recurrent herpes simplex skin infections. He died from complications of the overdose five months after the accident.

User and Manufacturer Response

The Therac-25 was shut down for testing the day after this accident. One local AECL engineer and one from the home office in Canada came to ETCC to investigate. They spent a day running the machine through tests, but could not reproduce a Malfunction 54. The AECL engineer from the home office reportedly explained that it was not possible for the Therac-25 to overdose a patient. The ETCC physicist claims that he asked AECL at this time if there were any other reports of radiation overexposure and that AECL personnel (including the quality assurance manager) told him that AECL knew of no accidents involving radiation overexposure by the Therac-25. This seems odd since AECL was surely at least aware of the Hamilton accident that had occurred seven months before and the Yakima accident, and, even by their account, learned of the Georgia lawsuit around this time (which had been filed four months earlier). The AECL engineers then suggested that an electrical problem might have caused the burn.

The electric shock theory was checked out thoroughly by an independent engineering firm. The final report indicated that there was no electrical grounding problem in the machine, and it did not appear capable of giving a patient an electrical shock. The ETCC physicist checked the calibration of the Therac-25 and found it to be satisfactory. He put the machine back into service on April 7, 1986, convinced that it was performing properly.

A.4 Causal Factors

Many lessons can be learned from this series of accidents. A few are considered here.

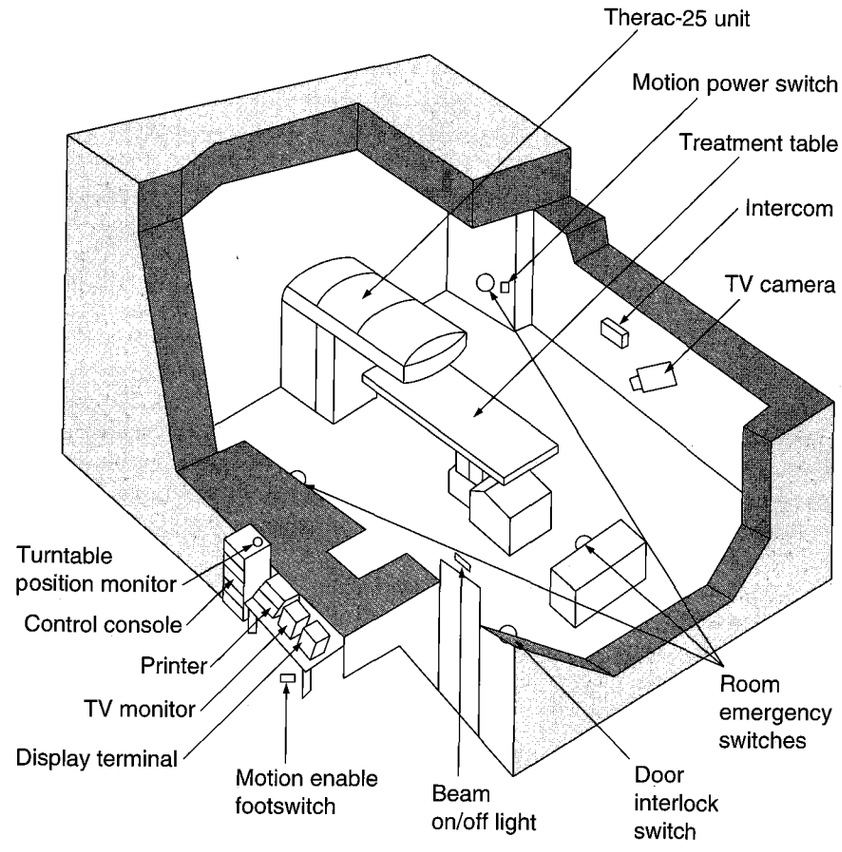

FIGURE A.5
A typical Therac-25 facility after the final CAP.

Overconfidence in Software. A common mistake in engineering, in this case and in many others, is to put too much confidence in software. There seems to be a feeling among nonsoftware professionals that software will not or cannot fail, which leads to complacency and overreliance on computer functions.

A related tendency among engineers is to ignore software. The first safety analysis on the Therac-25 did not include software—although nearly full responsibility for safety rested on it. When problems started occurring, it was assumed that hardware had caused them, and the investigation looked only at the hardware.

Confusing Reliability with Safety. This software was highly reliable. It worked tens of thousands of times before overdosing anyone, and occurrences of erroneous behavior were few and far between. AECL assumed that their software was safe because it was reliable, and this led to complacency.

Lack of Defensive Design. The software did not contain self-checks or other error-detection and error-handling features that would have detected the inconsistencies and coding errors. Audit trails were limited because of a lack of memory. However, today larger memories are available and audit trails and other design techniques must be given high priority in making tradeoff decisions.

Patient reactions were the only real indications of the seriousness of the problems with the Therac-25; there were no independent checks that the machine and its software were operating correctly. Such verification cannot be assigned to operators without providing them with some means of detecting errors: The Therac-25 software "lied" to the operators, and the machine itself was not capable of detecting that a massive overdose had occurred. The ion chambers on the Therac-25 could not handle the high density of ionization from the unscanned electron beam at high beam current; they thus became saturated and gave an indication of a low dosage. Engineers need to design for the worst case.

Failure to Eliminate Root Causes. One of the lessons to be learned from the Therac-25 experiences is that focusing on particular software design errors is not the way to make a system safe. Virtually all complex software can be made to behave in an unexpected fashion under some conditions: There will always be another software bug. Just as engineers would not rely on a design with a hardware single point of failure that could lead to catastrophe, they should not do so if that single point of failure is software.

The Therac-20 contained the same software error implicated in the Tyler deaths, but this machine included hardware interlocks that mitigated the consequences of the error. Protection against software errors can and should be built into both the system and the software itself. We cannot eliminate all software errors, but we can often protect against their worst effects, and we can recognize their likelihood in our decision making.

One of the serious mistakes that led to the multiple Therac-25 accidents was the tendency to believe that the cause of an accident had been determined (e.g., a microswitch failure in the case of Hamilton) without adequate evidence to come to this conclusion and without looking at all possible contributing factors. Without a thorough investigation, it is not possible to determine whether a sensor provided the wrong information, the software provided an incorrect command, or the actuator had a transient failure and did the wrong thing on its own. In the case of the Hamilton accident, a transient microswitch failure was assumed to be the cause even though the engineers were unable to reproduce the failure or to find anything wrong with the microswitch.

In general, it is a mistake to patch just one causal factor (such as the software) and assume that future accidents will be eliminated. Accidents are unlikely to occur in exactly the same way again. If we patch only the symptoms and ignore the deeper underlying causes, or if we fix only the specific cause of one accident, we are unlikely to have much effect on future accidents. The series of accidents involving the Therac-25 is a good example of exactly this problem: Fixing each individual software flaw as it was found did not solve the safety problems of the device.

Complacency. Often it takes an accident to alert people to the dangers involved in technology. A medical physicist wrote about the Therac-25 accidents:

> In the past decade or two, the medical accelerator "industry" has become perhaps a little complacent about safety. We have assumed that the manufacturers have all kinds of safety design experience since they've been in the business a long time. We know that there are many safety codes, guides, and regulations to guide them and we have been reassured by the hitherto excellent record of these machines. Except for a few incidents in the 1960's (e.g., at Hammersmith, Hamburg) the use of medical accelerators has been remarkably free of serious radiation accidents until now. Perhaps, though we have been spoiled by this success [284].

This problem seems to be common in all fields.

Unrealistic Risk Assessments. The first hazard analyses initially ignored software, and then they treated it superficially by assuming that all software errors were equally likely. The probabilistic risk assessments generated undue confidence in the machine and in the results of the risk assessment themselves. When the first Yakima accident was reported to AECL, the company did not investigate. Their evidence for their belief that the radiation burn could not have been caused by their machine included a probabilistic risk assessment showing that safety had increased by five orders of magnitude as a result of the microswitch fix.

The belief that safety had been increased by such a large amount seems hard to justify. Perhaps it was based on the probability of failure of the microswitch (typically 10^{-5}) AND-ed with the other interlocks. The problem with all such analyses is that they typically make many independence assumptions and exclude aspects of the problem—in this case, software—that are difficult to quantify but which may have a larger impact on safety than the quantifiable factors that are included.

Inadequate Investigation or Followup on Accident Reports. Every company building safety-critical systems should have audit trails and incident analysis procedures that are applied whenever any hint of a problem is found that might lead to an accident. The first phone call by Tim Still should have led to an extensive investigation of the events at Kennestone. Certainly, learning about the first lawsuit should have triggered an immediate response.

Inadequate Software Engineering Practices. Some basic software engineering principles that apparently were violated in the case of the Therac-25 include the following:

- Software specifications and documentation should not be an afterthought.
- Rigorous software quality assurance practices and standards should be established.
- Designs should be kept simple and dangerous coding practices avoided.
- Ways to detect errors and get information about them, such as software audit trails, should be designed into the software from the beginning.
- The software should be subjected to extensive testing and formal analysis at the module and software level; system testing alone is not adequate. Regression testing should be performed on all software changes.
- Computer displays and the presentation of information to the operators, such as error messages, along with user manuals and other documentation need to be carefully designed.

The manufacturer said that the hardware and software were "tested and exercised separately or together over many years." In his deposition for one of the lawsuits, the quality assurance manager explained that testing was done in two parts. A "small amount" of software testing was done on a simulator, but most of the testing was done as a system. It appears that unit and software testing was minimal, with most of the effort directed at the integrated system test. At a Therac-25 user's meeting, the same man stated that the Therac-25 software was tested for 2,700 hours. Under questioning by the users, he clarified this as meaning "2700 hours of use." The FDA difficulty in getting an adequate test plan out of the company and the lack of regression testing are evidence that testing was not done well.

The design is unnecessarily complex for such critical software. It is untestable in the sense that the design ensured that the known errors (there may very well be more that have just not been found) would most likely not have been found using standard testing and verification techniques. This does not mean that software testing is not important, only that software must be designed to be testable and that simple designs may prevent errors in the first place.

Software Reuse. Important lessons about software reuse can be found in these accidents. A naive assumption is often made that reusing software or using commercial off-the-shelf software will increase safety because the software will have been exercised extensively. Reusing software modules does not guarantee safety in the new system to which they are transferred and sometimes leads to awkward and dangerous designs. Safety is a quality of the system in which the software is used; it is not a quality of the software itself. Rewriting the entire software in order to get a clean and simple design may be safer in many cases.

Safe versus Friendly User Interfaces. Making the machine as easy as possible to use may conflict with safety goals. Certainly, the user interface design left much to be desired, but eliminating multiple data entry and assuming that operators would check the values carefully before pressing the return key was unrealistic.

User and Government Oversight and Standards. Once the FDA got involved in the Therac-25, their response was impressive, especially considering how little experience they had with similar problems in computer-controlled medical devices. Since the Therac-25 events, the FDA has moved to improve the reporting system and to augment their procedures and guidelines to include software. The input and pressure from the user group was also important in getting the machine fixed and provides an important lesson to users in other industries.

SAFEWARE

System Safety and Computers

Nancy G. Leveson

University of Washington

SAD

ADDISON-WESLEY PUBLISHING COMPANY

Reading, Massachusetts • Menlo Park, California • New York • Don Mills, Ontario
Wokingham, England • Amsterdam • Bonn • Sydney • Singapore
Tokyo • Madrid • San Juan • Milan • Paris