

[alyzer](#)[Documentation](#)[License](#)[Blog](#)[About
us](#)[Try
for
free](#)[Support](#) [En](#)

Killer Bug. Therac-25: Quick-and-Dirty

Oct 10 2016

Author: Aleksey Statsenko

[The murderer](#)[The murder](#)[The investigation](#)[Fixes](#)[Additional resources on the Therac-25 and related accidents](#)[Conclusion](#)[UPD](#)

Program code started using machines to kill people as early as in 1985.



A standard one-time therapeutic dose of radiation is up to 200 [rads](#).

1000 rads is a lethal dose, and the revolted machine was burning the defenseless humans with 20 000 rads.

Let's look into the case of a system error - the worst software bug in history - that occurred as a result of incremental yet uncoordinated software improvements.

Hardware locks were removed in the Therac-25, and the safety-maintaining functions were passed to the software instead.

In this article, we will talk about how the investigation went and what lessons IT engineers, programmers, and testers should learn from this story not to let something like that happen again.

The Therac-25 is a radiation therapy machine, a medical linear accelerator produced by Atomic Energy of Canada Limited (AECL).



The plan of the facility is shown in the figure below.

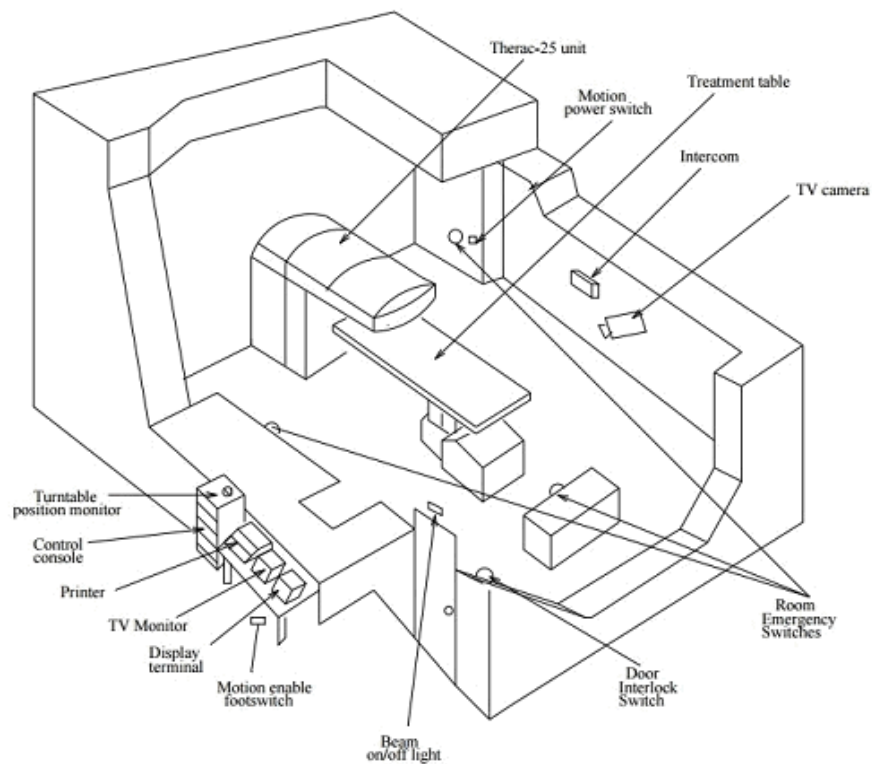


Figure 5: A typical Therac-25 facility after the final CAP.

And here's a commercial for housewives.

**HPS Presents: Therac 25**

Graham Caldwell

01:03

The murder

Between June 1985 and January 1987, this machine was the cause of six radiation-overdose accidents, when some of the patients were exposed to dozens of thousands of rads. At least two patients died of the direct consequences of the overdoses.

The technician recalled changing the command 'x' to 'e' that day. It was found that doing it quickly enough resulted in radiation overdose in almost 100% of cases.



The investigation



While prosecuting the cases against AECL, the Smith County District Attorney's office in Tyler, Texas, asked [Nancy Leveson](#) (who was a Computer Science professor at the University of California, Irvine, at the time) to assist as an expert in the investigation. She made a considerable contribution to system and software safety. Nancy and Clark Turner spent three years collecting the materials and reconstructing the events related to the Therac-25 accidents. This is an important result, as for most incidents involving safety, information appears to be incomplete, inconsistent, and incorrect.

AECL built three versions of their machine: Therac-6, Therac-20, and Therac-25. The versions 6 and 20 were manufactured in partnership with CGR, a French company. The partnership had dissolved before the Therac-25 was designed, but both companies maintained access to the designs and source code of the earlier models.

The Therac-20 codebase was developed from the Therac-6. All three machines used a [PDP-11](#) computer. Therac-6 and 20 didn't need that computer, though. Both were designed to operate as standalone devices. In manual mode, a radiotherapy technician would **manually** set up various parts of the machine, including the turntable to place one of three devices in the path of the electron beam.

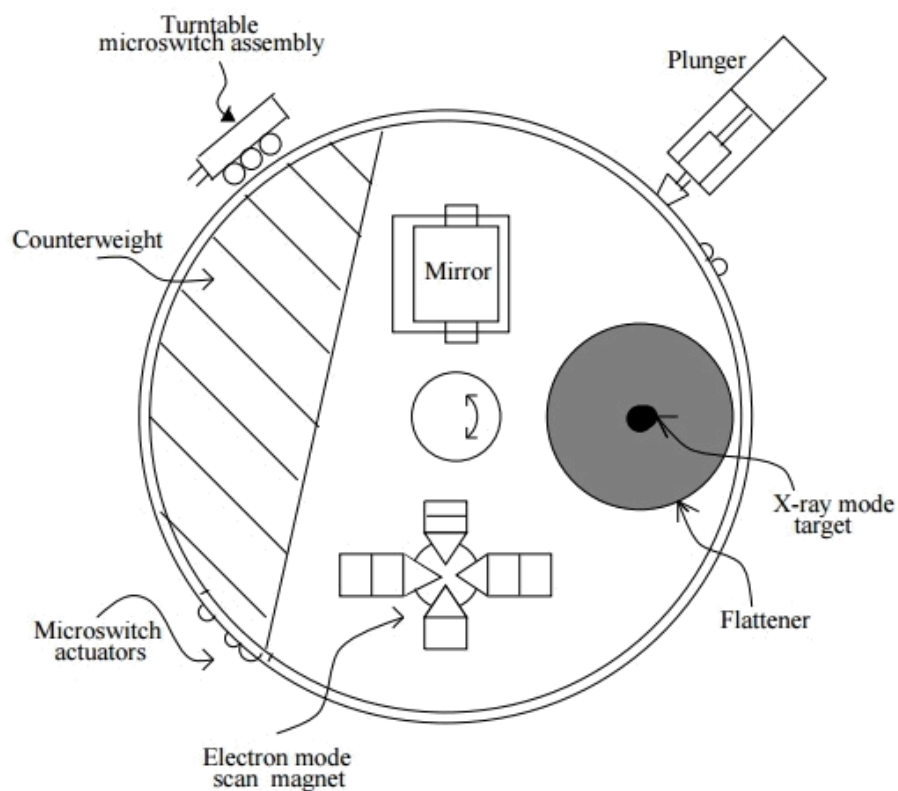


Figure 1: Upper turntable assembly.

In electron mode, scanning magnets would be used to spread the beam out to cover a larger area. In X-ray mode, a target was placed in the electron beam with electrons striking the target to produce X-ray photons directed at the patient. Finally, a mirror could be placed in the beam. The electron beam would never switch on while the mirror was in place. The mirror would reflect a light which would help the radiotherapy technician to precisely aim the machine.

On the Therac-6 and 20, hardware locks prevented the operator from doing something dangerous, say selecting a high power electron beam without the x-ray target in place.

Attempting to activate the accelerator in an invalid mode would trigger a protector, bringing everything to a halt. The PDP-11 and associated hardware were added as a convenience. The technician could enter a prescription in on a VT-100 terminal, and the computer would use servos to position the turntable and other devices.

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
COLLIMATOR ROTATION (DEG)		359.2	359
COLLIMATOR X (CM)		14.2	14.3
COLLIMATOR Y (CM)		27.2	27.3
WEDGE NUMBER		1	1
ACCESSORY NUMBER		0	0
DATE	: 84-OCT-26	SYSTEM	: BEAM READY
TIME	: 12:55: 8	TREAT	: TREAT PAUSE
OPR ID	: T25V02-R03	REASON	: OPERATOR
		OP. MODE	: TREAT AUTO
			X-RAY 173777
		COMMAND:	

Figure 2: Operator interface screen layout.

Hospitals loved the fact that the computer was faster at setup than a human. Less setup time meant more patients per day.

When it came time to design the Therac-25, AECL decided to **go with computer control only**. Not only did they remove many of the manual controls, they also removed the hardware locks. The computer would keep track of the machine setup and shut things down if it detected a dangerous situation.

Well, well...

At least four bugs were found in the Therac-25 software that could cause radiation overdose.

One shared variable was used both for analyzing input values and tracking turntable position. Quickly entering the data on the terminal could, therefore, result in leaving the turntable in the wrong position ([race condition](#)).

It took about 8 seconds for the bending magnets to set in place. If the operator changed the beam type and power within that time and moved the cursor to the final position, the system would not detect those changes.

Division by the value of the variable controlling the beam power in some cases led to a zero-division error and, as a result, power increase up to the largest value possible.

Setting a (one-byte) Boolean variable to "true" was done through the "x=x+1" command, so pressing the "Set" button would result in the system failing to identify the message about incorrect turntable position 1 time out of 256.

A number of potential bugs were also found: the multitasking operating system lacked any synchronization.

Fixes

All interruptions related to the dosimetry system would halt the treatment process instead of suspending it. Operators would need to reenter all parameters.

A software single-pulse shutdown was added.

An independent hardware single-pulse shutdown was added.

Cryptic malfunction messages were replaced with meaningful messages and dose-rate messages were displayed on the monitor.

A potentiometer was added to monitor the turntable location.

A motion-enable footswitch (deadman switch) was added so that the turntable and other parts of the machine could move only while the operator was holding this switch closed.

In X-ray mode, interlocking with the 270-degree bending magnet was added to ensure that the target and beam flattener were in position.

Complete list of fixes in English:

- All interruptions related to the dosimetry system will go to a treatment suspend, not a treatment pause. Operators will not be allowed to restart the machine without reentering all parameters.
- A software single-pulse shutdown will be added.
- An independent hardware single-pulse shutdown will be added.
- Monitoring logic for turntable position will be improved to ensure that the turntable is in one of the three legal positions.
- A potentiometer will be added to the turntable. The output is used to monitor exact turntable location and provide a visible position signal to the operator.
- Interlocking with the 270-degree bending magnet will be added to ensure that the target and beam flattener are in position if the X-ray mode is selected.
- Beam-on will be prevented if the turntable is in the field light or any intermediate position.
- Cryptic malfunction messages will be replaced with meaningful messages and highlighted dose-rate messages.
- Editing keys will be limited to *cursor up*, *backspace*, and *return*. All other keys will be inoperative.
- A motion-enable footswitch (a type of deadman switch) will be added. The operator will be required to hold this switch closed during movement of certain parts of the machine to prevent unwanted motions when the operator is not in control.
- Twenty three other changes will be made to the software to improve its operation and reliability, including disabling of unused keys, changing the operation of the *set* and *reset* commands, preventing copying of the control program on site, changing the way various detected hardware faults are handled, eliminating errors in the software that were detected during the review process, adding several additional software interlocks, disallowing changes in the service mode while a treatment is in progress, and adding meaningful error messages.
- The known software problems associated with the Tyler and Yakima accidents will be fixed.
- The manuals will be fixed to reflect the changes.

Source: Nancy G. Leveson, [Therac-25 Accidents](#)

The manufacturer said that the hardware and software had been tested over many years. However, the investigation found that a minimum amount of tests had been run on a simulator, while most of the effort had been directed at the integrated system test. It means that the developers neglected unit testing and did integration testing only.

A naive assumption is often made that reusing software or using commercial off-the-shelf software increases safety because the software has been exercised extensively. Reusing software modules does not guarantee safety in the new system to which they are transferred due to the development

specifics of that system. Rewriting the entire software may be safer in many cases.

In this case, the manufacturer chose to reuse the program code from the Therac-6 and Therac-20, though the Therac-6 did not provide X-ray mode at all, while the Therac-20 was equipped with hardware locks.

Since the Therac-25 events, the [FDA](#) has changed their attitude to many of the issues involving safety-critical systems and moved to improve the reporting system and to augment their procedures and guidelines to include software. It was an important lesson not only for FDA, but for all industrial safety-critical systems.

Additional resources on the Therac-25 and related accidents

[My professor investigated the Therac-25 incident and was a part of the prosecution. Got any questions for me to ask him?](#)

[What is the name of the programmer who wrote the Therac-25 software?](#)

[Fatal Defect: Chasing Killer Computer Bugs](#)

Nancy Leveson, Clark S. Turner [An Investigation of the Therac-25 Accidents](#)

Nancy G. Leveson, [Therac-25 Accidents](#)

Nancy G. Leveson [Safeware: System Safety and Computer](#)

Popular related articles

[Infusion Pump Software Safety Research at FDA](#)

The University of California, Berkeley: Computer Science 61A — Lecture 35: Therac-25

<https://www.youtube.com/watch?v=nxX-aAvZbmM>

Conclusion


A
b
b
fi
ti
c
v
p
L
T
T
h
s

Bugs that buzzed a lot

Date: Apr 12 2024
Author: Anastasiya Vorobeva

A real bug, a bug in the code, or distractions can all affect your project and lead to many different consequences. In this article, we've collected a number of notorious and fascinating bugs. Let's....

6.01 K 0 1 0

 SHARE

Get notifications about comments to this article

We can email you a selection of our best articles once a month

☐ By clicking this button you agree to our [Privacy Policy](#) statement

Your Email

Comments (0)

Subscribe

Guest



B

I

U



Leave a comment

Want to try PVS-Studio for free?

Get free trial

Achievements

Blog

Checked projects

Detected errors

Customers

Early access program

PVS-Studio

About PVS-Studio

Download

Request a trial key

Documentation

Online Examples

Troubleshooting

Licensing

Purchase a license

Choose a license

For clients

For resellers

For students

For Open Source

For Microsoft MVP

Company

About us

Jobs

Contacts

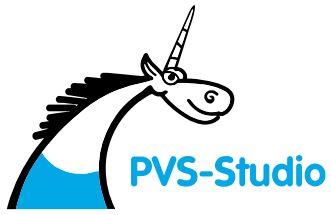
Feedback

Subscribe to newsletter

Contact us for technical information
or other questions

Contact us

Search



[Sitemap](#)

[Terms of use](#)

©2008 - 2025, PVS-Studio LLC