6.033
COMPUTER SYSTEMS ENGINEERING
RECITATION 1

JOHN WANG

1. Reactions to Therac-25 Paper

Takeaways: stories keep the reader enthralled. Even in technical papers which don't have sensational things to tell, you should tell a story.

However, this technical paper didn't really give a good description of the key technical point. There is no clear description of what the fundamental problem was. Even though papers need a good story, they also need to clearly describe the technical point.

1.1. **Using Old Software.** Pros:

- Saves time.
- Safer because they had been tested. If the core software is good, then you can easily and safely expand.

Cons:

- Does not fit new context.

1.2. **Problems.**

- No reporting of bugs. Since there were hardware locks, no bugs were reported for Therac-6 or Therac-20.
- Didn't check software. Thought new software was close enough to previous software.
- Did not accept idea that bugs exist.
- No meaningful error messages.
- No consistent bug tracing. Didn't collect bugs together and try to understand what's going on.

2. Fundamental Problem

Invalid states:

- Pass through and x-ray mode.
- *X-ray beam with magnets.* This is what usually happened.
- Pass through and electron therapy mode.

Variables: cursor variable turns on if the cursor has been brought to the right place (0 or 1). When cursor goes to bottom, the keyboard thread starts working. The table thread and beam thread are also running. These threads are all spinning on a shared variable (waiting for that shared variable to change). Keyboard thread spins on the cursor variable. It takes the information on the screen and puts it into a buffer for the other two threads to read. Once the keyboard thread puts all the information into the buffer, it sets the data ready bit to 1.

The table thread and beam notice that data ready bit is set. The table thread sets things to x-ray mode and the beam thread sets the beam to the correct position. The table thread works very quickly, but the beam thread has to do a lot of work and set a lot of other variables (takes about 8 seconds).

If the technician quickly changes the mode (say to electron therapy mode), then the cursor bit changes to 1. The keyboard thread puts electron therapy into the buffer and sets data ready bit to 1. The table thread (very fast) notices this change and sets the state to electron therapy. The table moves to the magnets. The other thread, however, doesn't notice this because it's still running.

Essentially the problem is a *concurrency problem* which only occurs when a highly trained technician uses the machine very quickly.

2.1. **Takeaways.** *Complex systems have complicated bugs.* Example: The Mars Pathfinder rover. They had a bug that they found on Earth which they couldn't reproduce, so they sent the rover up to Mars. The first thing that happened was the bug and the rover couldn't be contacted for 2 days.

This is a common thing: when bugs appear and it is difficult to repeat a bug, people usually tend to forget it. You want to have great error recording in your software design so that you can know exactly what the state was for which the error occurred.

2.2. **Potential Solutions.**

- Block the operator.
- Locks among threads.
- Check state before starting beam.
- Fixed table with predefined pairs.