

Software Failure in Aerospace Accidents

Lessons learned from past mistakes



developing software in mission or life critical domains. . . which is where the fun begins!

Randy Winters has over 25 years experience designing and developing real-time software systems. He has successfully started several companies and has sold mission critical systems to many large corporations including Microsoft, HP, Apple and Epson. Recently he served as Director of Novell's Advanced Development Group where he helped design Novell's Internet Proxy Cache Technology.

He has received many major awards for his product designs and also holds a patent in the US and abroad in multi-tasking Artificial Intelligence. He is currently working with the faculty at Carnegie Mellon University's Master of Software Engineering Program.

Mr. Winters is an avid fixed-wing and hot air balloon pilot, horseback rider and scuba diver, and also likes boating and mountain biking and of course, coding all night long.

**Date: Tuesday
March 25, 2003
Time: 3:00-4:00 PM
Place: GCB 140**

He also invites you stop by and meet him Tuesday, March 25 in GCB 101/103 from 9:30AM - noon.

He will have a variety of high tech gadgets to show you. Did you know that US Vice President Dick Cheney has an implantable heart defibrillator that can be programmed over the Internet? Come see it. Randy has brought one along!

Unfortunately, software errors and bugs in mission-critical systems can have disastrous consequences costing billions of dollars and sometimes the loss of life.

Aerospace systems have become more and more complex, placing greater challenges on the software that makes them run.

How do you insure that critical software will run reliably? How do you test systems that can only be run in space? How do you develop code that will protect these projects and guarantee human life?

Randy Winters will show some of the worst software-related aerospace accidents and close calls using pictures, video clips and slides. He will then discuss what went wrong and what should have been done to prevent each disaster.

In addition, he will also address what you will need to know if you find yourself working as a software engineer

