

# CS 335 — Software Development

Introduction

Jan 13, 2014

*Healthcare.gov must verify a would-be applicant's eligibility from Homeland Security, IRS and Social Security systems, in real time, plus enroll members electronically to any registered insurance company in any of the 34 states that don't have their own state health insurance exchange. Matthew Heusser, "6 Software Development Lessons from Healthcare.gov's Failed Launch", in CIO, Nov 18, 2013.*

*C.M.S. is the Centers for Medicare and Medicaid Services, which [a Healthcare.gov contractor senior VP testifying before congress] described as the “systems integrator, or ‘quarterback,’ on this project, and. . . the ultimate responsible party for the end-to-end performance of the overall Federal Exchange.” She then noted that full testing of the complete Healthcare.gov site only occurred “in the last two weeks of September.” Healthcare.gov involved fifty-five different contractors that each delivered a piece of the final system to C.M.S. for integration and testing. Rusty Foster, “Don’t go chasing waterfalls: A more agile healthcare.gov”, in The New Yorker, Oct 28, 2013.*

*If you've worked on a waterfall project with a defined test phase, you know it's not really a "test" phase at all. Once the first show-stopper bug is found, it's actually a fixing phase. To state that Healthcare.gov wasn't fully tested implies that the test group either never found any show-stopper bugs or didn't have the ability to halt the project. Given the legally required go-live mandate, the second option seems realistic.*

*One thing we do know for sure: The organization rushed ahead with coding before it knew answers to questions. That's no recipe for success.*

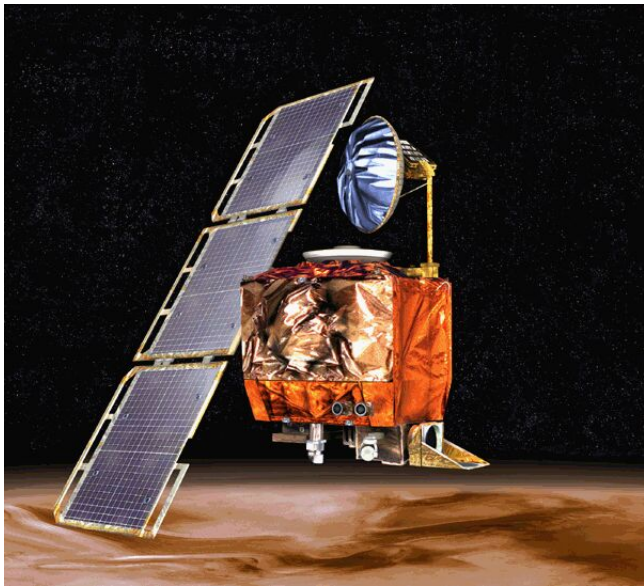
*We know because at go-live the JavaScript code itself was still full of filler text that a non-decision maker typically insert onto a page or pop-up.*

*People in the organization must have known the site was not ready. Perhaps they spoke up and were overruled.*

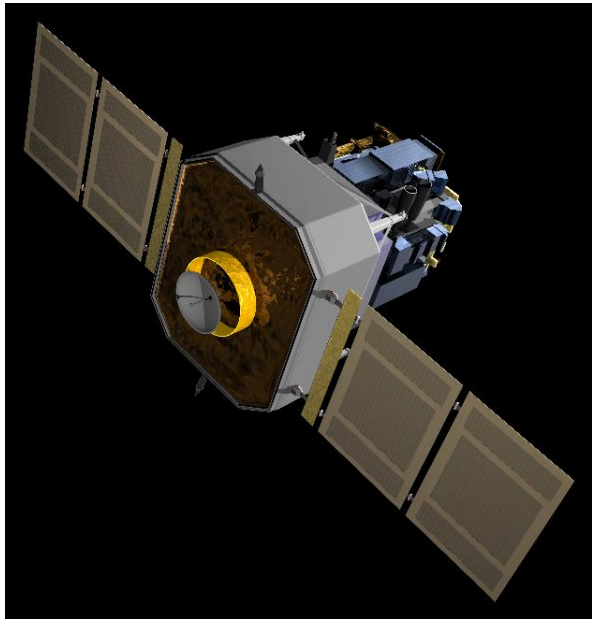
*We don't know. Matthew Heusser, "6 Software Development Lessons from Healthcare.gov's Failed Launch", in CIO, Nov 18, 2013.*



[http://www.esa.int/SPECIALS/Launchers\\_Access\\_to\\_Space/ASEVLU0TCNC\\_1.html](http://www.esa.int/SPECIALS/Launchers_Access_to_Space/ASEVLU0TCNC_1.html)



[http://en.wikipedia.org/wiki/Mars\\_Climate\\_Orbiter](http://en.wikipedia.org/wiki/Mars_Climate_Orbiter)



<http://soho.esac.esa.int/gallery/Spacecraft/SOHOLower2.html>

## JPL's 10 rules

1. Control flow must be simple. No goto statements or *recursion* allowed.
2. All loops must have a fixed upper-bound on the number of iterations.
3. No dynamic memory allocation (after initialization).
4. No function may be longer than a single sheet of paper.
5. Minimum of two assertions per function.
6. Declarations at the smallest possible level of scope
7. All return values must be checked by the calling function.
8. Preprocessor-use is restricted.
9. No more than one level of dereference is allowed.
10. All code is compiled from day 1 with no warnings.

G Holzman, "The Power of 10: Rules for Developing Safety-Critical Code."  
*IEEE Computer*, June 2006