SE4831 Software Quality Assurance

- Objectives
  - Quantify the impact of software failure
  - Explain Why Because Analysis and how it can be used to reach the root causes for a software problem
  - Recognize and discuss the complex nature of modern software failure
  - Explain the importance of organizational culture on quality

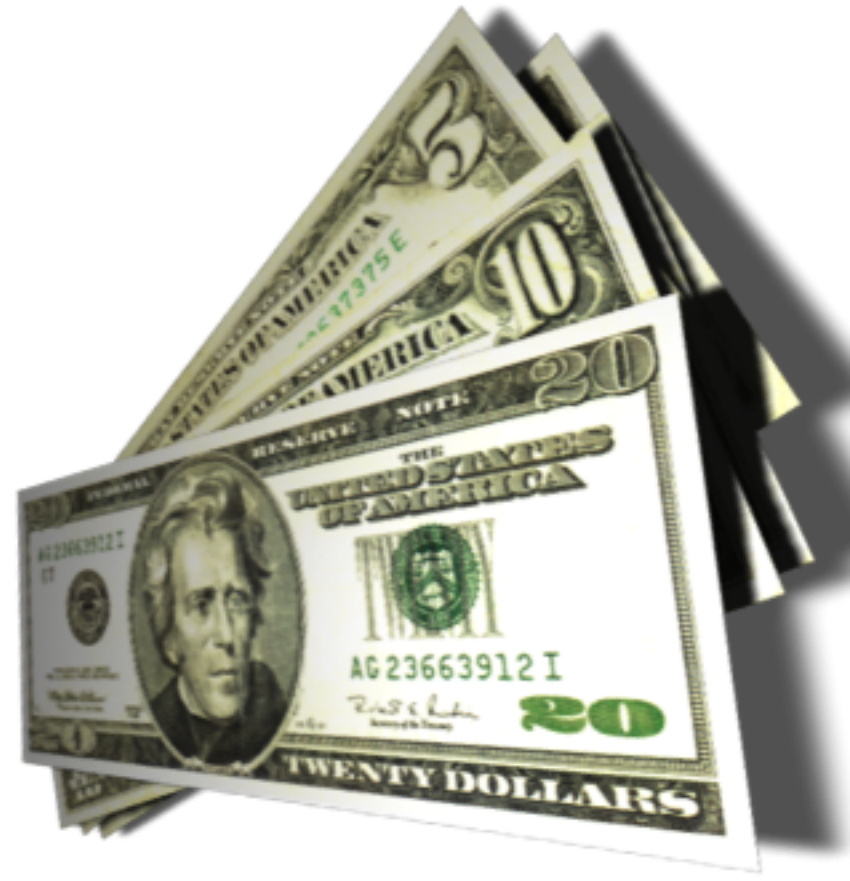
Introduction

"The most significant problem facing the data processing business today is the software problem that is manifested in two major complaints: software is too expensive and software is unreliable."

-Glenford J. Myers: **Software Reliability: Principles and Practices**, 1976. *[Mye76, page 3]*

Introduction – Impact of Failure

- Large economic impact to software failure
  - $59.5 billion annual cost to economy.
- Fiscal year 2003 DOD spent $21 billion on software development
  - $8 billion (40%) spent to fix reliability problems in software



Software Failure is expensive!

Toyota's killer firmware: Bad design and its consequences

Toyota settles acceleration lawsuit after $3-million verdict

Toyota heads off punitive damages after a $3-million jury verdict pointed to software defects in a fatal crash. The case could fuel other sudden acceleration lawsuits.

Q. What is that conclusion?

A. My conclusion is that a software defect has caused the unintended acceleration which could not be stopped through the pumping of the brakes and the braking. Not in time anyway to avoid the crash.

- Barr's ultimate conclusions were that:
  - Toyota's electronic throttle control system (ETCS) source code is of unreasonable quality.
  - Toyota's source code is defective and contains bugs, including bugs that can cause unintended acceleration (UA).
  - Code-quality metrics predict presence of additional bugs.
  - Toyota's fail safes are defective and inadequate (referring to them as a *"house of cards" safety architecture).*
  - Misbehaviors of Toyota's ETCS are a cause of UA.

Ariane 5 : June 4th, 1996

- Total failure of the Ariane 5 launcher maiden flight
- Direct result of a software failure.
  - Failure occurred when typecast of a 64-bit floating point number to a signed 16-bit integer overflowed.
  - No exception handler associated with the conversion
    - System hander shut down computer.
  - Backup system was identical in all regards
    - 37ms later, backup system failed.
- Software Developed in Ada.
  - Had code been developed in C, problem most likely would not have occurred.

Inquiry Board Report

On 4 June 1996, the maiden flight of the Ariane 5 launcher ended in a failure. Only about 40 seconds after initiation of the flight sequence, at an altitude of about 3700 m, the launcher veered off its flight path, broke up and exploded.

The failure of the Ariane 501 was caused by the complete loss of guidance and attitude information 37 seconds after start of the main engine ignition sequence (30 seconds after lift-off). This loss of information was due to specification and design errors in the software of the inertial reference system.

The internal SRI software exception was caused during execution of a data conversion from 64-bit floating point to 16-bit signed integer value. The floating point number which was converted had a value greater than what could be represented by a 16-bit signed integer.*

- - ARIANE 5, Flight 501 Failure, Report by the Inquiry Board
- *SRI stands for Système de Référence Inertielle or Inertial Reference System.

Ariane 5 Source Code

```
...
declare
  vertical_veloc_sensor:  float;
  horizontal_veloc_sensor: float;
  vertical_veloc_bias:  integer;
  horizontal_veloc_bias: integer;
  ...
begin
  declare
    pragma suppress(numeric_error,  horizontal_veloc_bias);
  begin
    sensor_get(vertical_veloc_sensor);
    sensor_get(horizontal_veloc_sensor);
    vertical_veloc_bias  := integer(vertical_veloc_sensor);
    horizontal_veloc_bias  := integer(horizontal_veloc_sensor);
    ...
  exception
    when numeric_error  => calculate_vertical_veloc();
    when others => use_irs1();
  end;
end irs2;
```

Does this tell the whole story?

Why Because Analysis

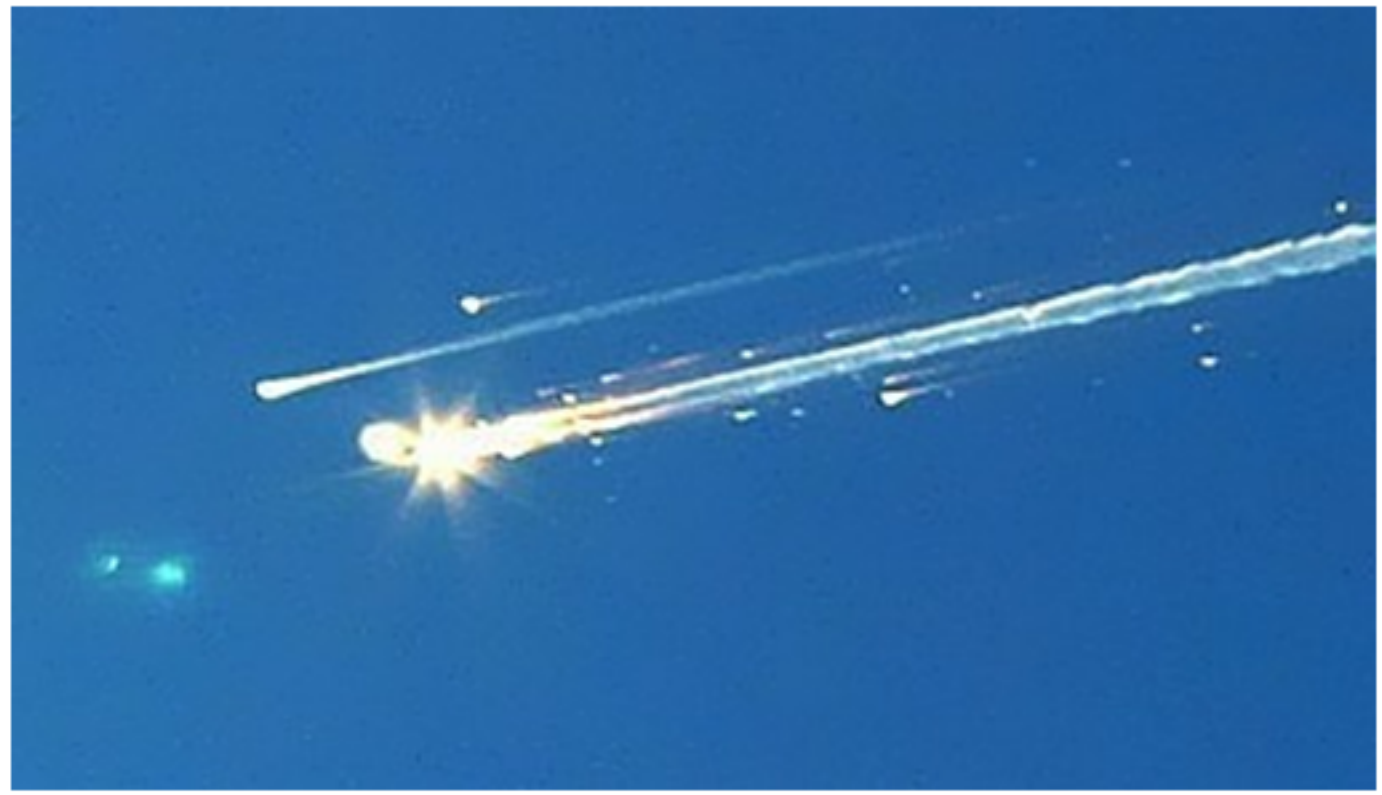
- Why-Because Analysis (WBA)
  - rigorous technique for causally analysing the behaviour of complex technical and socio-technical systems.
  - Primary application is in the analysis of accidents, mainly to transportation systems (air, rail and sea).

- Ariane 501 Explodes 37 seconds into flight
  - Guidance System Fails Completely
    - 37.72 ms into flight, Backup guidance system failed
      - 37 s into flight primary guidance system fails
        - Floating point number is typecast to an integer without exception handling
          - Ada Programming Language used for Guidance System
          - Programmers thought that the values could never overflow
            - Ariane 4 took off with lower launch velocity
        - Default exception handling disables system
          - Assumption made that program was correct until it was proven faulty.
        - Unnecessary Process was running
          - Ariane 4 required process to continue running after launch in order to handle launch scrubs properly
        - Performed Tests did not detect the problem
          - Behavior not documented in the requirements
        - Complete Revalidation of the software did not occur.
          - Software was reused from Ariane 4
      - 37.72 ms into flight Backup System has Integer Overflow
        - Default exception handling disables system
          - Assumption made that program was correct until it was proven faulty.
        - Floating point number is typecast to an integer without exception handling
          - Programmers thought that the values could never overflow
            - Ariane 4 took off with lower launch velocity
          - Ada Programming Language used for Guidance System
        - Complete revalidation of the software did not occur.
          - Software was reused from Ariane 4
        - Unnecessary Process was running
          - Ariane 4 required process to continue running after launch in order to handle launch scrubs properly
          - Deemed to risky to remove an unnecessary process from validated software
      - Primary and Backup Systems used identical software
        - Redundancy Intended to solve hardware failure, not software design failure.
        - Easier to develop and test 1 version of software instead of 2 versions.

What went wrong?

How could we have prevented this?

Space Shuttle Columbia



- February 1, 2003
  - Space shuttle Columbia Disintegrated on re-entry
  - All astronauts were killed
- Root Cause of failure: Foam wing during launch

- "Organizational culture refers to the basic values, norms, beliefs, and practices that characterize the functioning of a particular institution. At the most basic level, organizational culture defines the assumptions that employees make as they carry out their work; it defines "the way we do things here." An organization's culture is a powerful force that persists through reorganizations and the departure of key personnel."
- ***CAIB Report, Vol. 1, p. 101***

- **"In our view, the NASA organizational culture had as much to do with this accident as the foam."**
- ***CAIB Report, Vol. 1, p. 97***

What does this have to do with software?

- Software Quality Assurance IS NOT Just testing

What will be talking about in this course?

- Best Practices for quality assurance
  - Organizational
  - Formal Inspections
  - Static Analysis with a Tool
  - Risk management