



SYST 101: Intro to Systems

Lecture 24

Apr 17, 2003

C. Wells, SEOR Dept.



Announcements

- Remaining Semester Schedule
 - Apr 22, 24 Project 2 demos and oral presentations
 - Apr 29, May 1 Review for final
 - May 2 SYST 490/495 presentations
 - May 13 Final Exam 10:30 – 1:15



Demo Schedule Apr 22

- Project 2 Demonstration Trials
 - 12:00 Team 1
 - 12:06 Team 2
 - 12:12 Team 3
 - 12:18 Team 4
 - 12:24 Team 5
 - 12:30 Team 6
 - 12:36 Team 7
 - 12:42 Team 8
 - 12:48 Team 9
 - 1:05 Team 1 Oral Presentation



Presentation Schedule Apr 24

- Project 2 oral presentations
 - 12:00 Team 2
 - 12:08 Team 3
 - 12:16 Team 4
 - 12:32 Team 5
 - 12:40 Team 6
 - 12:48 Team 7
 - 12:56 Team 8
 - 1:04 Team 9



Agenda

- Problem resolution



Testing Process

- Next class meet at Lab to run trials
 - All members of the team must be present at the start time
- Convene in classroom for first oral presentation after trial runs



Failure Analysis In Lego Robots

- Like many real systems, we have a combination of hardware and software.
- When behavior is not as expected, where is the problem?
- Typically, on large project, you have your hardware team and your software team.



Typical Team Failure Analysis





Software Can Be Tricky

- Validation and verification of software is problematic at best
 - Impossible to test all conditions
- Failure modes are harder to identify
 - Failure sources can be computer hardware, logic, or coding
- Compilation and optimization can exacerbate the problem



A Basic Problem

- Knowing you're right
- Leads to
 - Dead ends
 - Being stuck
 - Conflict with others who know they're right too.



The Pencil Point Analysis

- All the steps were right, so why isn't the answer right?
 - An assumption was invalid, but was overlooked.
 - An inadvertent error exists but is repeatedly missed because the mind sees what it expects to see



Human Nature

- It's human nature, so it happens to all of us
- Not really feasible to avoid it happening
- Learn to recognize and learn how to get unstuck.



You May Be Stuck If...

- You're sure you've done everything right, and it's not working
- You know there's this hidden thing that is messing you up that you can't get to.



Mental Models

- You have a mental model of your system in your head.
 - Especially when you designed it.
- As it operates, your mental model stays in step.
- Your assumptions about your mental model may lead you to overlook the actual behavior of your system.



Getting Unstuck

- Become an independent observer
 - Observe your system operating, without keeping a mental model.
 - Watch what it does, and only what it does.
 - Preconceptions about what is wrong can keep your attention away from the real difficulty.



Using Your Team

- Appoint parts of your team as developers, other parts as testers
- Example:
 - Team A designs mechanical, tests program.
 - Team B tests mechanical, writes program.



“Pride of Ownership”

- A phrase often heard in the real world.
- Meaning: “I built this, and if you criticize what I built, you’re criticizing me, and I will take it personally.”
- Effects: Criticisms dismissed as “stupid” or “ignorant”. “They really don’t understand what’s going on if they can make comments like that”.



“Pride of Ownership”

- Leads to rigid thinking, no adaptation, bad designs.
- Lesson: Don't take it personally!!



“NIH” - Not Invented Here

- Translation: “If we didn’t invent it, then it can’t be any good”.
- Allows a team to dismiss others’ ideas and criticisms without truly considering them.



Hardware Designs

- Symmetric Vs. Asymmetric?
 - Symmetric means that it's the same on either side of a line (usually left-right symmetry in our case)
- Asymmetric designs may be harder to make perform exactly as you want.
 - Propeller aircraft turn one way better than the other.



Separate Testing

- Disconnect the hardware from the software:
- Is is supposed to roll in a straight line?
Does it? Can it turn if it needs to?
 - A Test” Disconnect the motors from the wheels (remove gears) and roll it down a hill.
 - Try to make it turn



Assignments

- Prepare for Project 2 demonstration trials
- Homework
 - Give an example of an instance where you solved a problem or made a decision, outside of class, using system engineering processes, procedures, and techniques (due Thursday 4-24-03)
 - scope of SE consistent with the problem at hand
 - not end to end process, just part of the SE process
 - **must** include the observations that initiated the process and the observations made during the process
 - (to see if you use, and recognize you are using, SE tools)