

Leveraging Simulation to Create Better Software Systems in an Agile World

Jason Ard
Kristine Davidsen
4/8/2013

Agenda

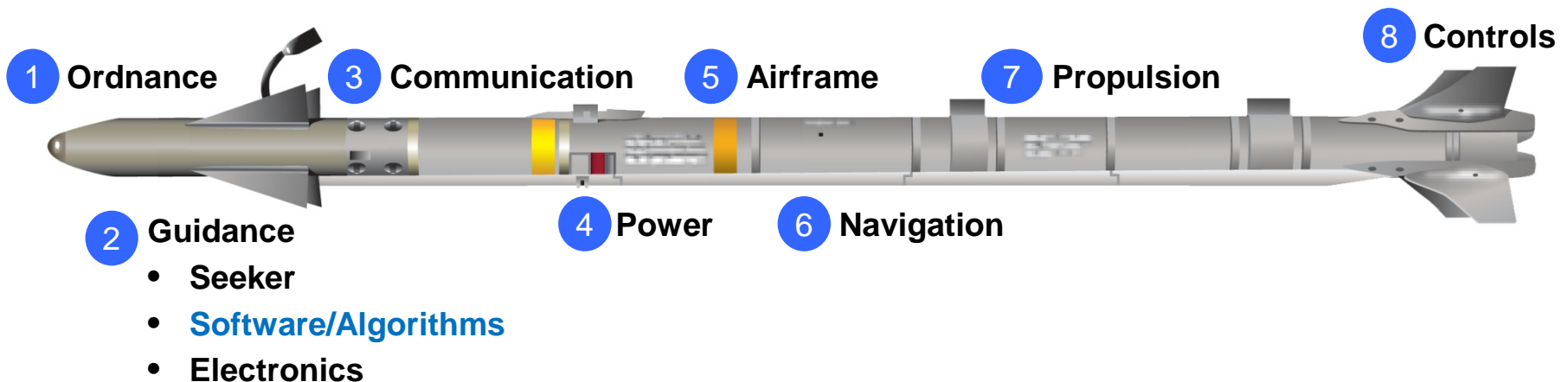
- Overview of our problem space
- Traditional spiral development turned Agile
- Leveraging simulation as a
 - System design tool
 - Software development tool
 - System assessment tool
- Summary
- Questions

The Problem Space

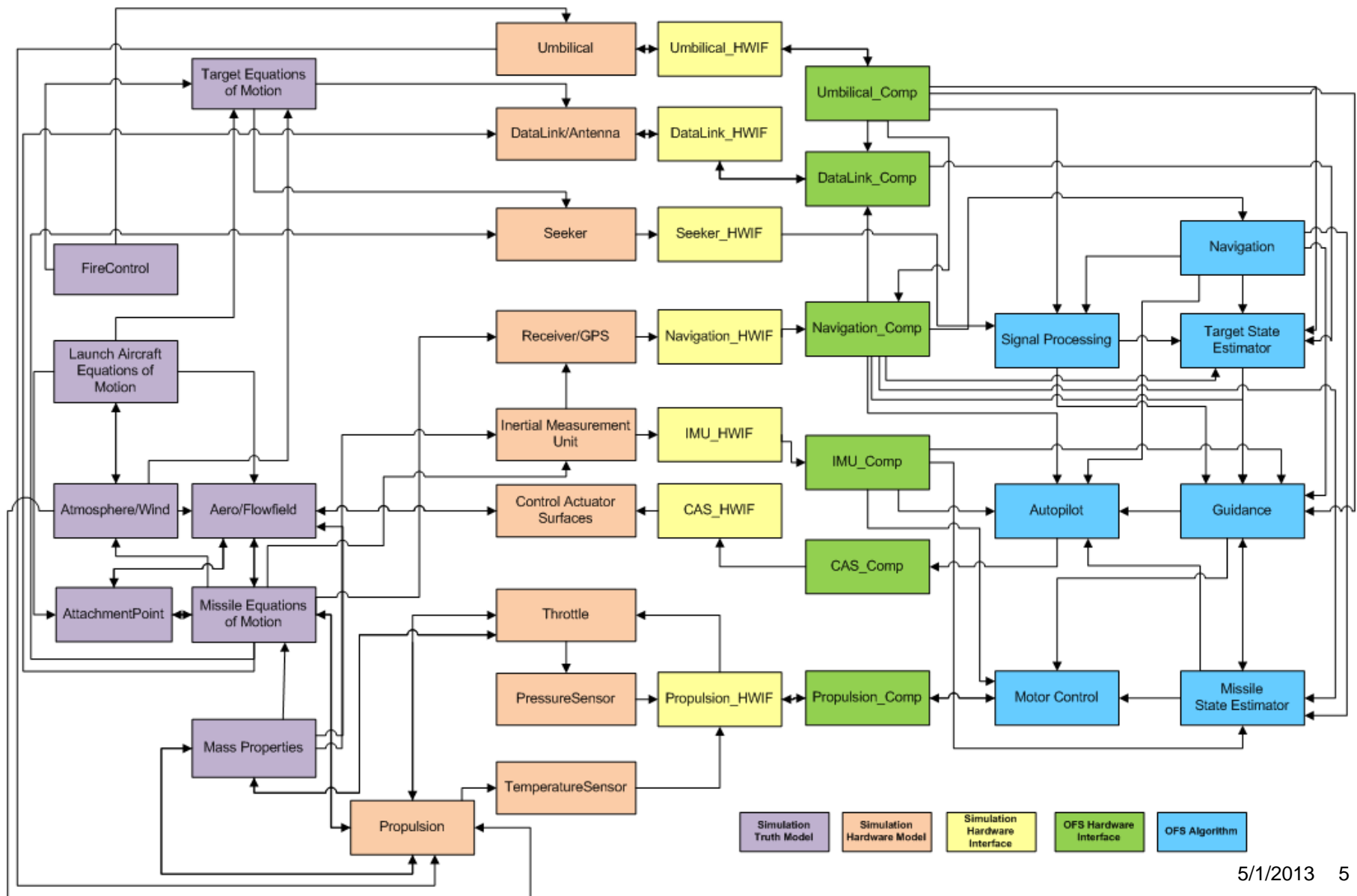


Missile System

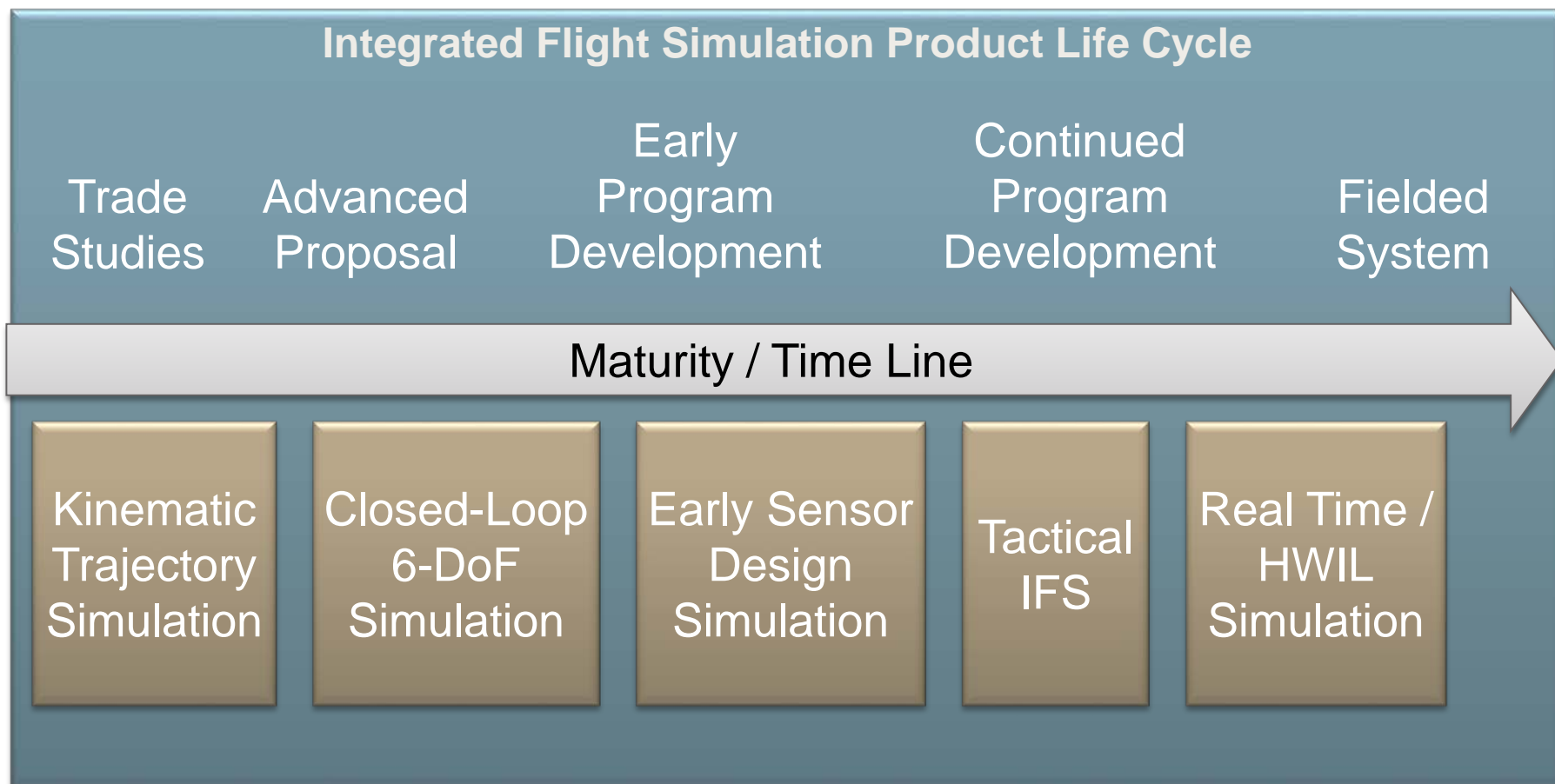
- Embedded software:
 - Controls hardware components (fins, gimbal, motor, radio, etc.)
- Simulation:
 - Design Studies
 - Software Development
 - Performance Prediction
 - Flight Test Analysis & Support
- Challenges:
 - Limited assets
 - Cost
 - Availability of testing environment
 - Safety



Integrated Flight Simulation (IFS) Block Diagram

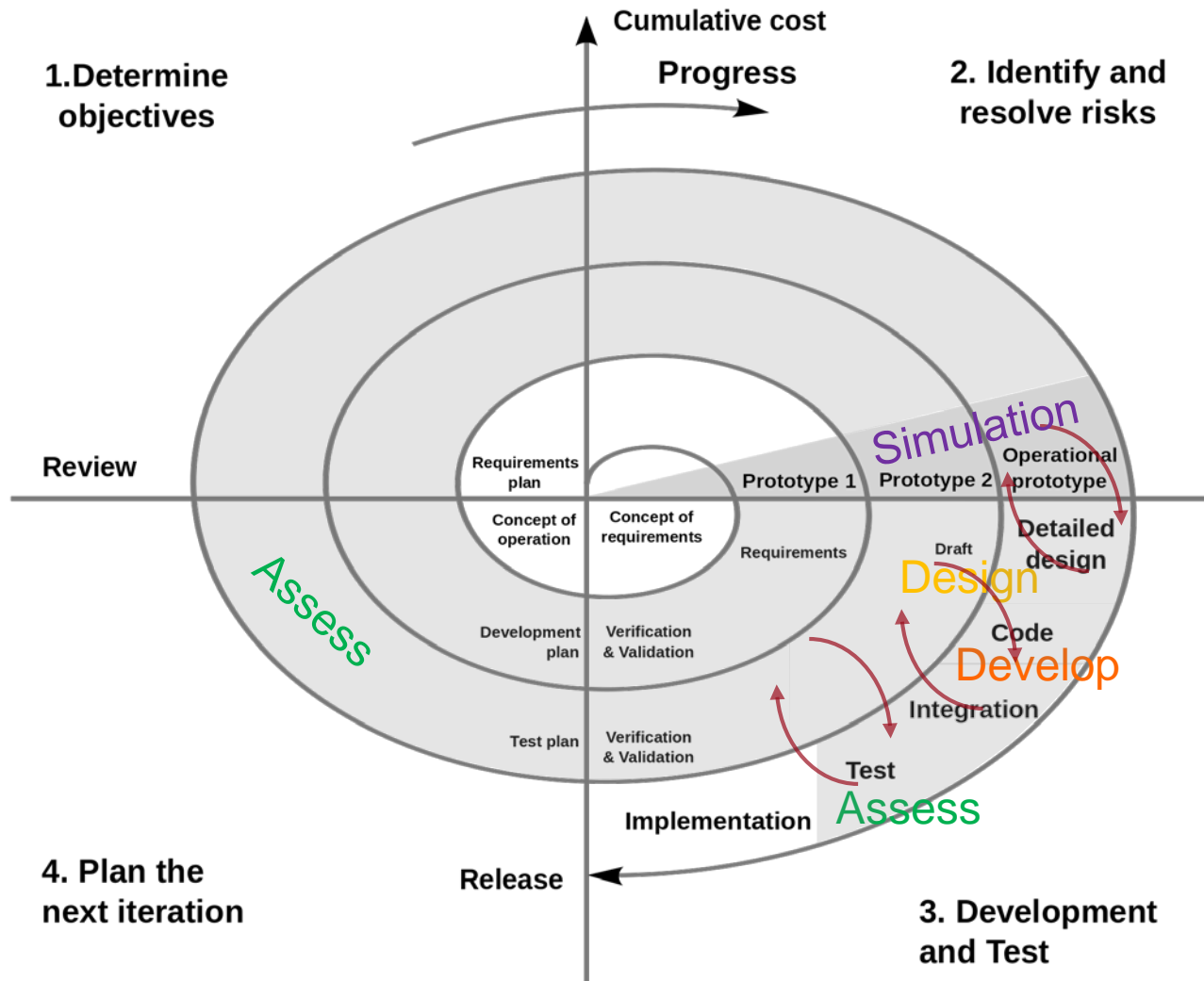


Integrated Flight Simulation Evolves



Simulation grows as the product grows

Simulation in the Traditional Spiral Model



Simulation Supports Agile Development

- Deliver the most valuable software frequently
- Use feedback to make constant adjustments to adapt quickly
- Determine what software to develop
- More effective face-to-face communication
- Inspect and adapt for the sake of the program

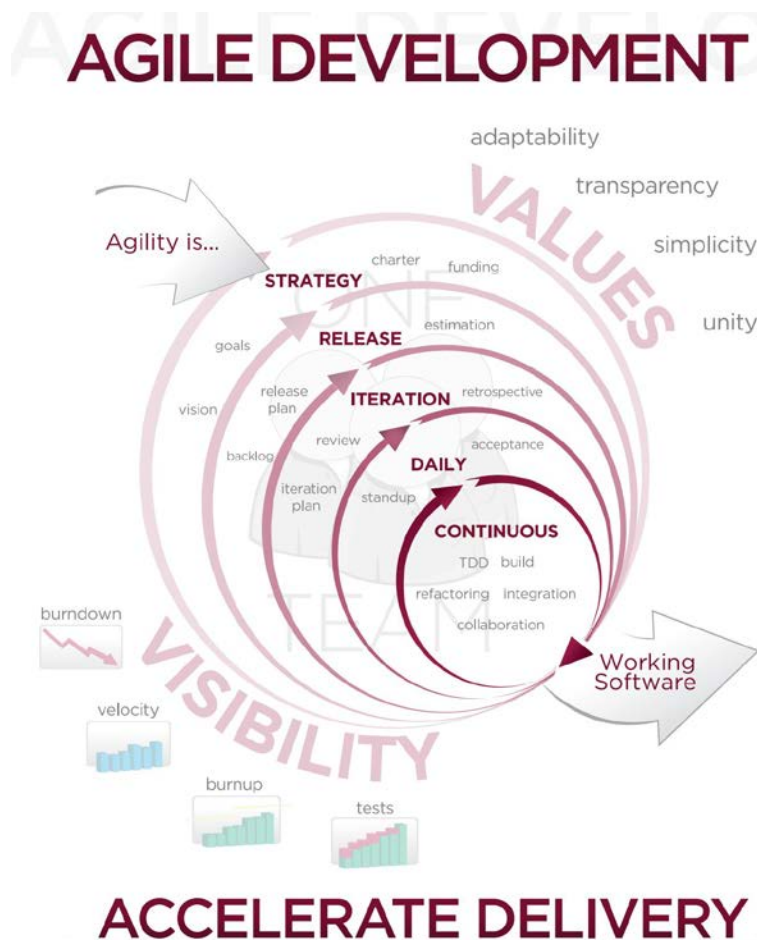
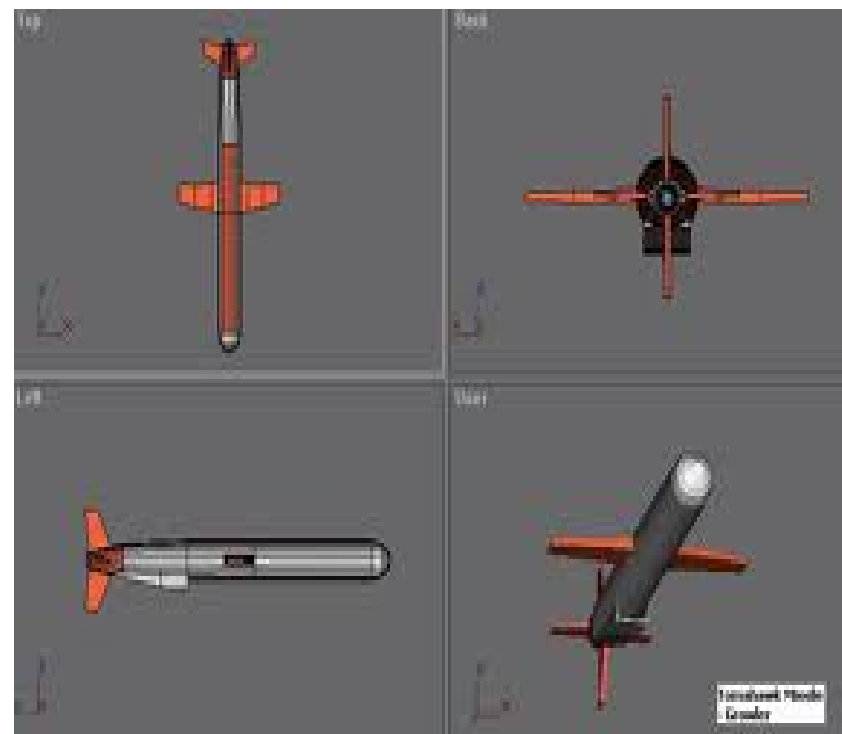


Image Attribution: Dbenson and VersionOne, Inc.
Licensed under the Creative Commons Attribution-Share Alike 3.0 Unported license
<http://commons.wikimedia.org/wiki/File:Agile-Software-Development-Poster-En.pdf>

Simulation as a System Design Tool

- System design decisions can be made using information gathered through simulation
 - Decide what type of rocket fuel to use
 - Select which algorithms should be used in embedded software
 - Determine what rates the embedded algorithms can be executed at
 - Demonstrate the feasibility of executing on an alternative processor



Simulation feedback aids system design

Example: Tracker Trade Study

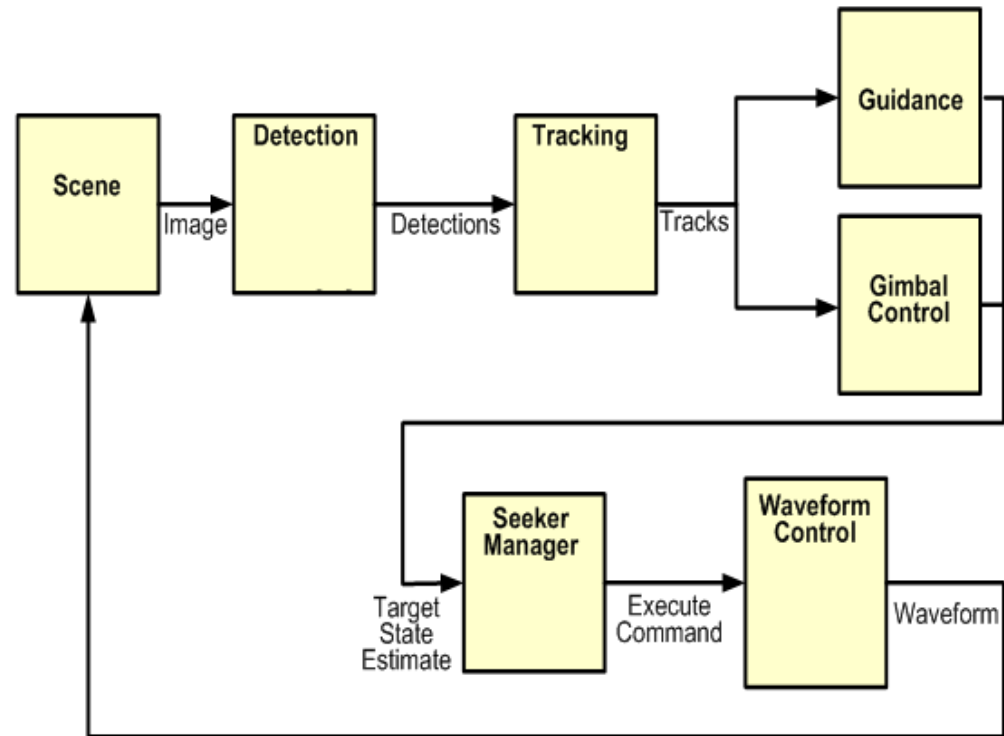
Objective & Response

■ Study Objective:

–Allocate system tolerance requirements to assure optimal performance of the tracker algorithm

■ Response Measure:

–Circular Error Probable (CEP)
i.e. median miss distance



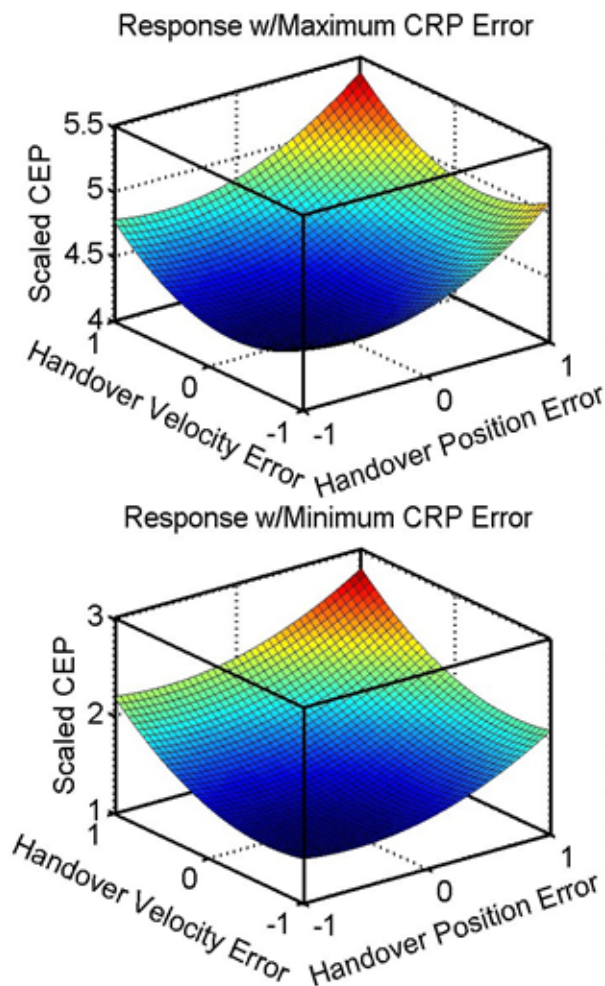
Example: Tracker Study Results

■ Results

- Discovered system was most sensitive to three system parameters
 - coherent reference point error
 - handover position error
 - handover velocity error
- Characterized system tolerances (based on CEP) as a function of those control factors

■ Potential Design Impacts

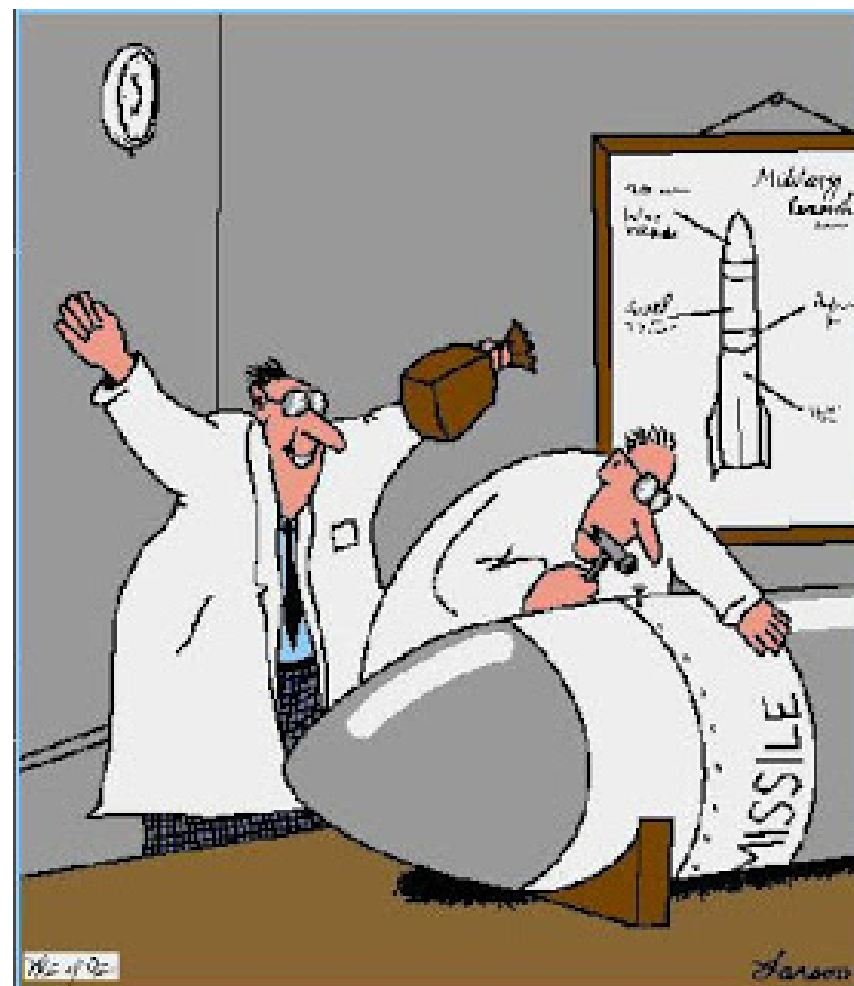
- Sensor hardware selection
- Software execution/update rates
- Selection of possible launch platforms
- Tracker usage scenarios (lock on before vs. after launch etc.)



Design trade-offs understood through simulation

Simulation as a Software Development Tool

- Develop device drivers using simulated device(s)
- Common continuous integration environment for
 - embedded SW developers
 - algorithm developers
 - performance analysts
 - hardware developers
 - systems testers
- Simulation empowers Agile development practices



Developing within the simulation saves time

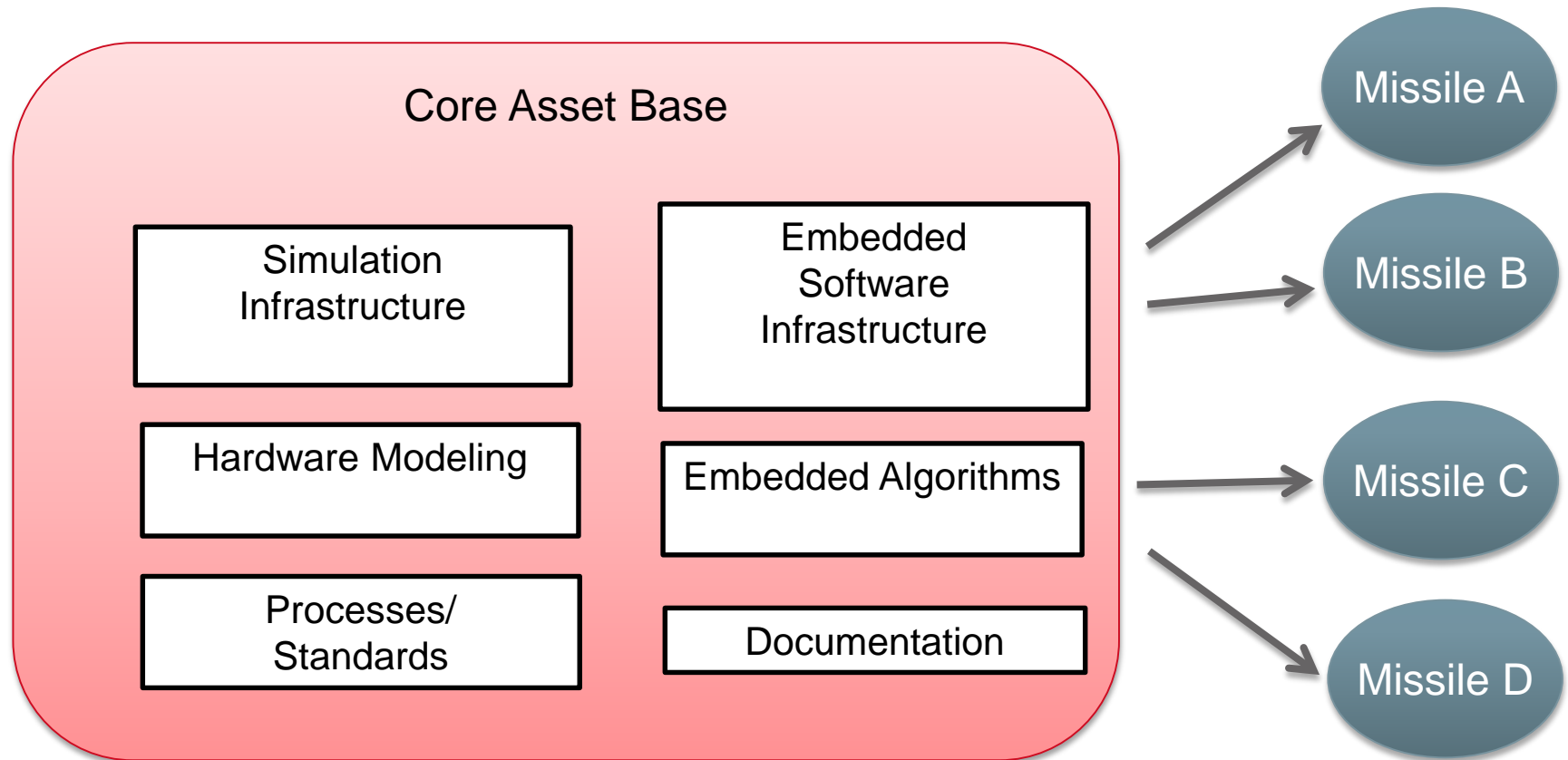
Example: Paper to Flight in 10 Months

- Protect our Troops ASAP
- No hardware available in beginning
- Software development start immediately
- Agile framework using Software in the Simulation (SiS)



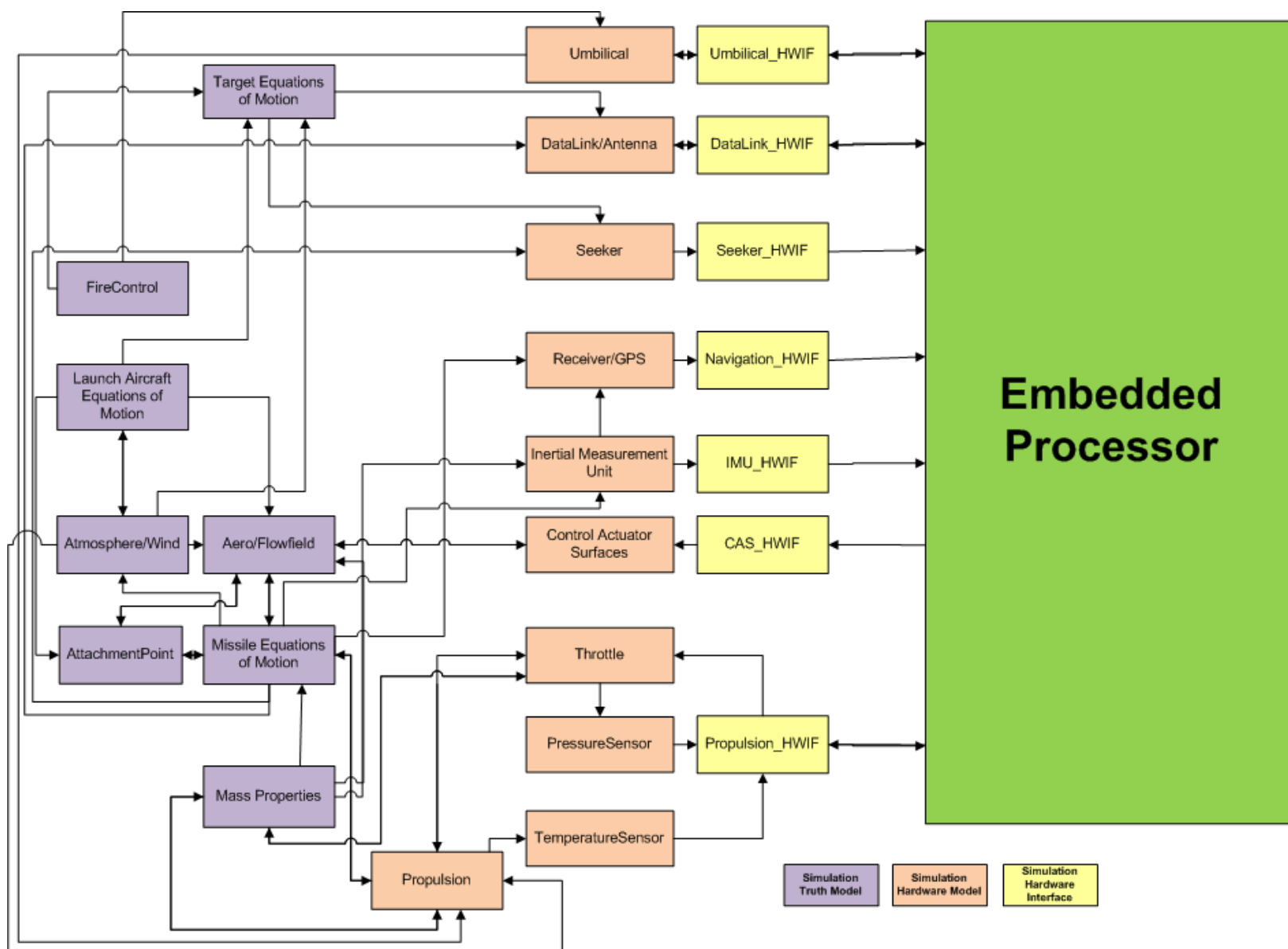
Simulation needed for rapid development

Development Using Missile Software Product Lines



Product lines allow strategic reuse

Integrated Flight Simulation Matured



Simulation Benefits Development

- Less dependency on hardware assets
- Early detection of performance issues
- Enables parallel development
- Integration is continuous



Simulation makes large development efforts manageable

Development Example: Simulation Based Hardware Interface Development

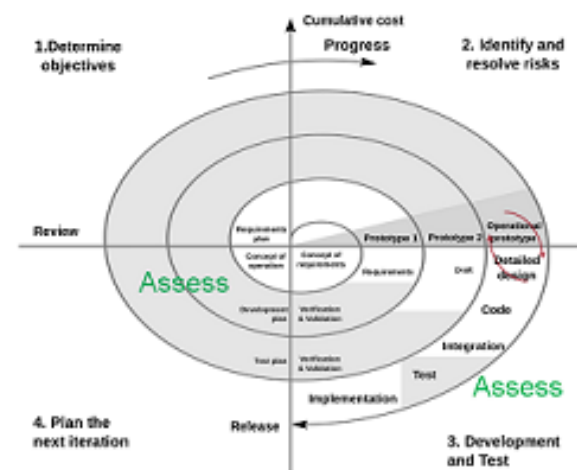
- Simulated hardware layer communication (encoding/decoding of bit streams) detected a flawed scale factor that caused simulated flight failures
- Because the interface would work under most conditions, our usual suite of software unit tests would not have detected it
- By developing the hardware communication layer in the simulation, we were able to detect this problem early and fixed it at very low cost



Simulation detects issues early to avert disaster

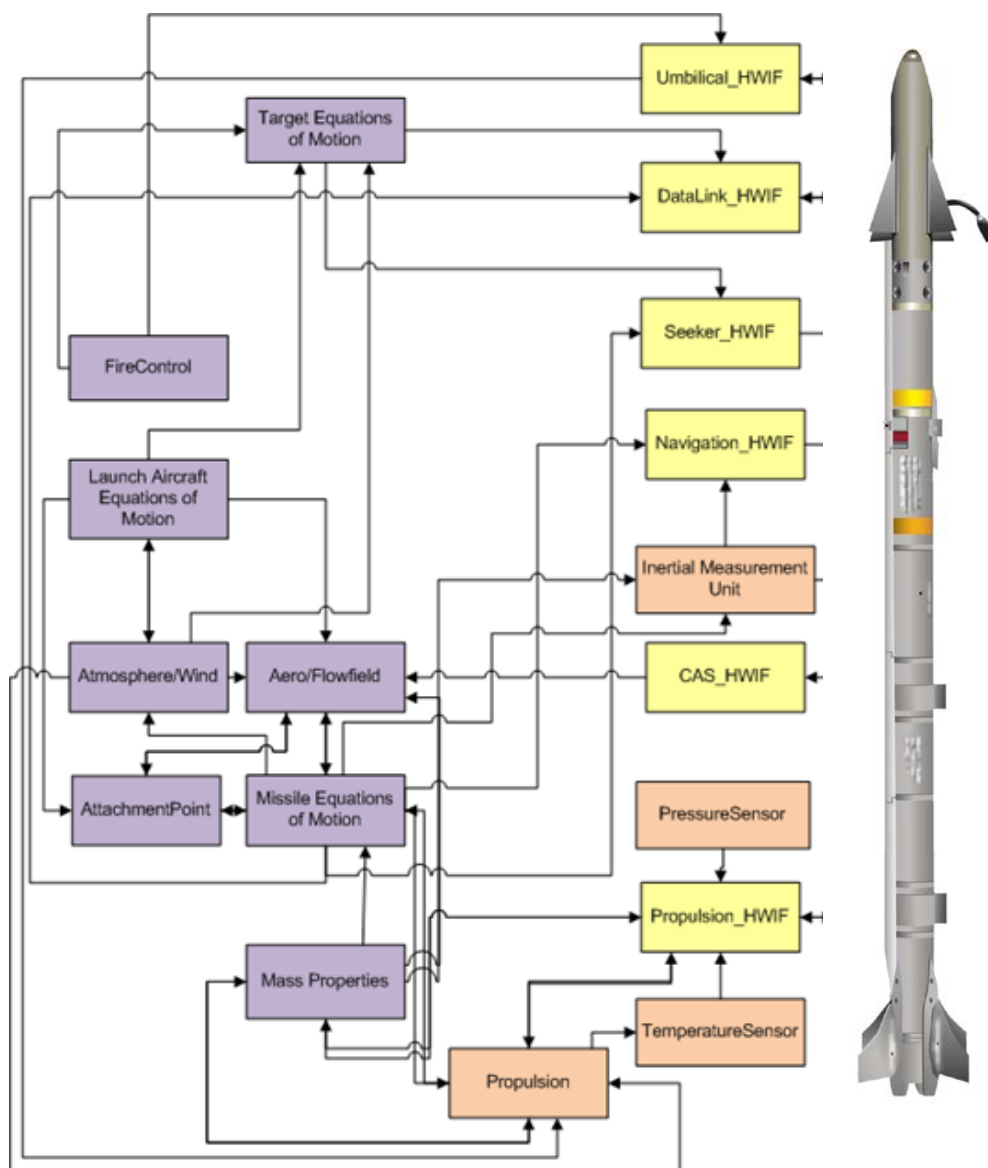
Simulation as an Assessment Tool

- Virtually unlimited assessment areas
- Low cost testing solution
- Immediate feedback
- Easily respond to changing conditions
- 1200 simulation runs in 30 minutes
- Formal Qualification Testing



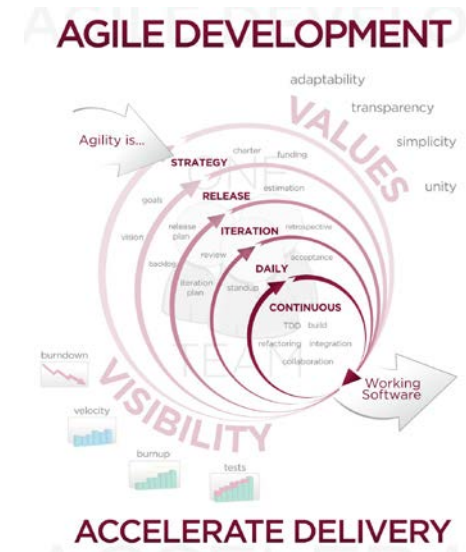
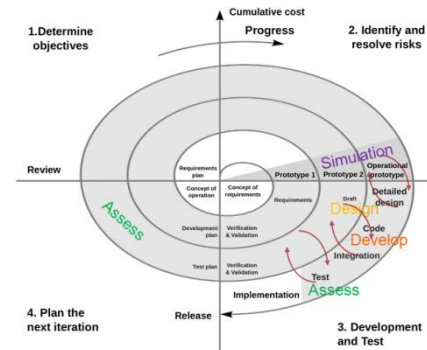
Simulation supports many forms of assessment

Hardware in the Loop Testing



Summary

- Simulations play a central role in Agile development of embedded software
- Provide inspect and adapt mechanism to produce a better product
- Simulation is a vital tool for
 - Design
 - Development
 - Assessment tool



410697-03

Biography

▪ Jason Ard

- Jason Ard joined Raytheon Missile Systems in 2007. He is a Senior Systems Engineer currently working out of the Performance Simulations Department in Tucson. Since joining Raytheon, he has developed Integrated Flight Simulations (IFS), Computer in the Loop (CIL) simulations, image processing algorithms, and software architectures on numerous programs including MRM (Mid-Range-Munition) JAGM (Joint-Air-To-Ground-Missile) and T3 (Triple Target Terminator). He received BS degrees in Applied Physics and in Computer Science from Brigham Young University.

Biography

▪ Kristine Davidsen

- Kristine has been a software engineer at Raytheon for 13 years spanning across two business units: Missile Systems (MS) and Integrated Defense Systems (IDS). She started her career developing test software at MS on Special Access Programs. Then later she transferred to IDS where she contributed to the Navy Zumwalt Destroyer Infrastructure. While at IDS, Kristine was introduced to Agile software development in 2008 when she worked on a company funded research and development program. Upon her return to MS in December 2008, Kristine joined the Software Engineering Center where she has been an embedded software developer and/or ScrumMaster on multiple programs including (Joint-Air-To-Ground-Missile), SM-3 Block IIA (Standard Missile-3) and T3 (Triple Target Terminator). In addition, she helped author Raytheon's Scrum courses and is the main instructor for the *Scrum for Practitioners* course.
- Kristine has a BS degree in Mathematics from the University of Arizona and is a Certified ScrumMaster.
- In her spare time, Kristine spends time with family, bicycling, raising lizards, cooking, and keeping antique vehicles on the road.

Appendix

List of Terms

- CIL: Computer in the Loop
- GNC: Guidance, Navigation, and Control
- GUI: Graphical User Interface
- HWIL: Hardware In The Loop
- RMS: Raytheon Missile Systems
- SEI: Software Engineering Institute (Carnegie Mellon University)
- SiS: Software in the Simulation
- SP: Signal Processing
- SPL: Software Product Line