Industry Challenges in Embedded Software Development

Don Wilson
Raytheon Company
dwilson4@raytheon.com
(520)545-9429

7 July 2005
Industry Challenges in Embedded Software Development

• Embedded software development at Raytheon

• What are key challenges in embedded software development?

• Thoughts on research and advances to address these challenges
Raytheon – Embedded Systems

- 80,000 employees
- $20B+ sales
- 8-10,000 software developers
- Broad spectrum of embedded systems
  - Commercial, military, space, international
- Distributed workforce
Real World Development @ Raytheon

• Large gap between research and industry development
  – Typical embedded software development @ Raytheon
    Limited modeling, limited component use
    C/C++, Commercial RTOS, Custom hardware, hand coded and integrated
  – Productivity growth is slow
  – Small groups using current research and technology

• Non-Technical Challenges in industry development
  – Developers of varying skill levels
  – Long product lifetimes influence technology
  – Risk aversion is extreme
  – Related domains – but we don’t share solutions well
  – Complexity growth is relentless

Reducing the gap between research and industry is critical
What advances would be most critical to our future?

• Components that are inherently real-time and configurable

• Mature real time architectures

• Dependability and quality in embedded systems
Components that are inherently real-time and configurable

- Today’s development incorporates RT performance based on design experience
  - Actual performance is tested into the system and varies over time
  - Nearly all systems must operate in multiple target environments

Research has not reached US industry
Mature real time architectures

- There is little commonality in representation and structure of software architectures across industry and Raytheon

- We lack a language to discuss, document, research and compare embedded system architectures
  - This reduces leverage from past successes
  - It is difficult to cross boundaries (industry, academia, different domains)

- Research, publication and standardization of embedded software architectures would
  - Improve architectural use and evolution
  - Support maturation of tools and techniques
  - Allow development of components that are composed into known architectures

What should a SW architecture of an embedded system consist of?
What are known solutions to different domains of embedded systems?
How do we compose components into a successful SW architecture?
Dependability and quality in embedded systems

• Raytheon customers want “no doubt” system performance
  – Even early tests must be low risk
  – Reliable, real world correctness and testing methods are critical

• Formal methods for dependability and correctness are insufficient today
  – Must encompass real world complexity levels
  – Methods for specifying quality attributes are insufficient
    Availability, resource consumption, reliability

• Expanding formal methods to entire systems (not just software) would be valuable
Other Challenges

• There is a shortage of software developers with real time embedded education and experience

• Tools coming out of research are limited in use unless adopted by commercial tool vendors

• Standards that would support real-time modeling, formal methods, etc are needed
Raytheon Initiatives

• Develop and support a variety of model driven software development programs
  – Internal Research and development, University partnerships, ESCHER

• Shape embedded systems software architecture standards, representation and communication
  – Standardize across Raytheon where possible

• Support embedded real time standards to facilitate commercial tool development
Conclusion

• Raytheon is interested in closing the gap between research and industry
  
  RT standards, partnerships with vendors, partnerships with academia, Maturation and risk reduction via testbeds, research support

• Formal methods for complex embedded systems
  
  Raytheon supports the development of techniques or methodologies that can dramatically improve system dependability, correctness and quality