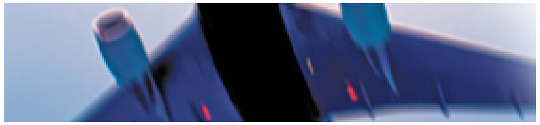
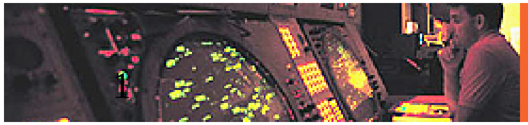
Stop the Pain: Effective Requirements Definition and Management for Project Success

Scott Derby, Esterline AVISTA

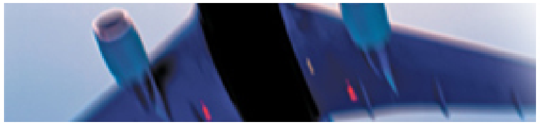
NDIA Systems Engineering Conference
October 20-23, 2008

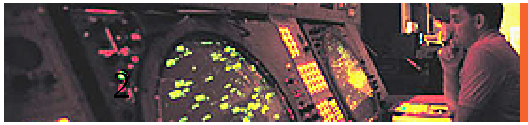




Agenda

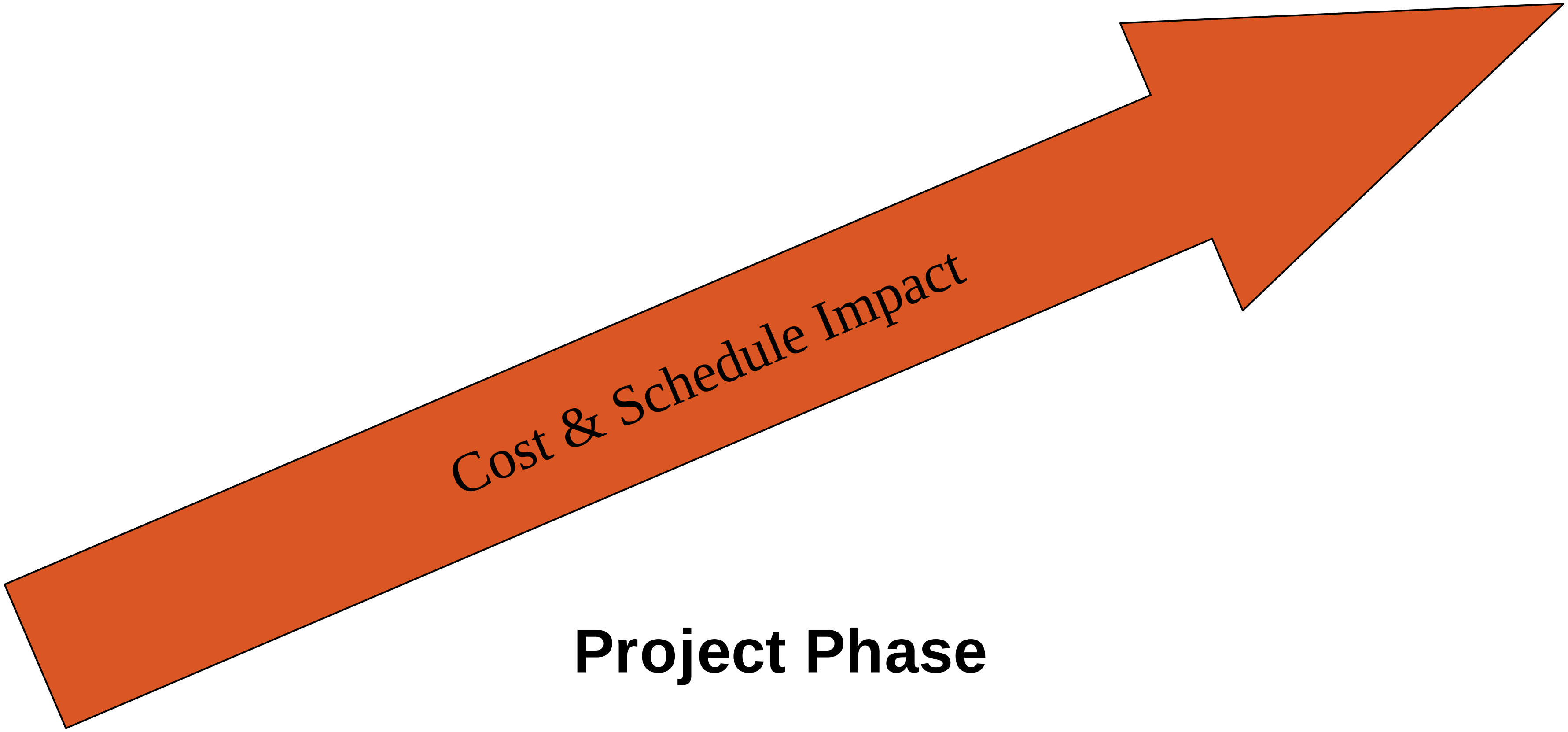
- Why are good requirements so important?
- What makes a good requirement?
- Requirements definition
- Managing change
- Advantages in modeling
- Effective prototyping
- Summary
- Q & A





Why Are Good Requirements So Important?

Change Impact vs. Project Phase

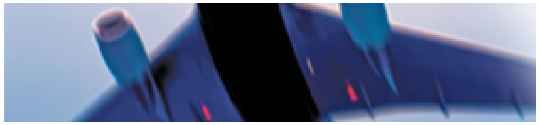


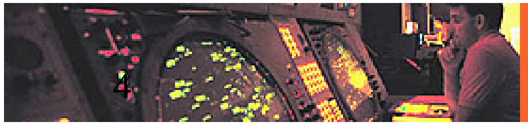
Cost & Schedule Impact

Project Phase

Why Are Good Requirements So Important? (cont.)

- Requirements can be:
  - Unrealistic
  - Incomplete
  - Ambiguous
  - Contradictory
  - Un-testable
  - Poorly managed
- This leads to:
  - Rework, delays, budget over-runs, unhappy customers





What Makes a Good Requirement?

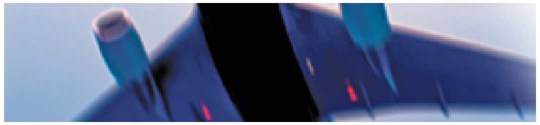
- Be S.M.A.R.T.*
  - Specific (concise, clear, unique)
  - Measurable
  - Achievable
  - Relevant
  - Testable
- What vs. How
- This leads to:
  - Less rework, shorter schedules, lower costs, happy customers
- *http://www.win.tue.nl/~wstomv/edu/2ip30/references/smart-requirements.pdf

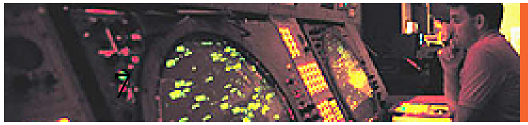
Requirements Definition

- Consider interests of ALL stakeholders
- Include all users in reviews
  - End user
  - Development/Safety Team
  - Production/Maintenance Team
  - Verification/Validation Team
- Don’t forget:
  - Traceability
  - Interface requirements

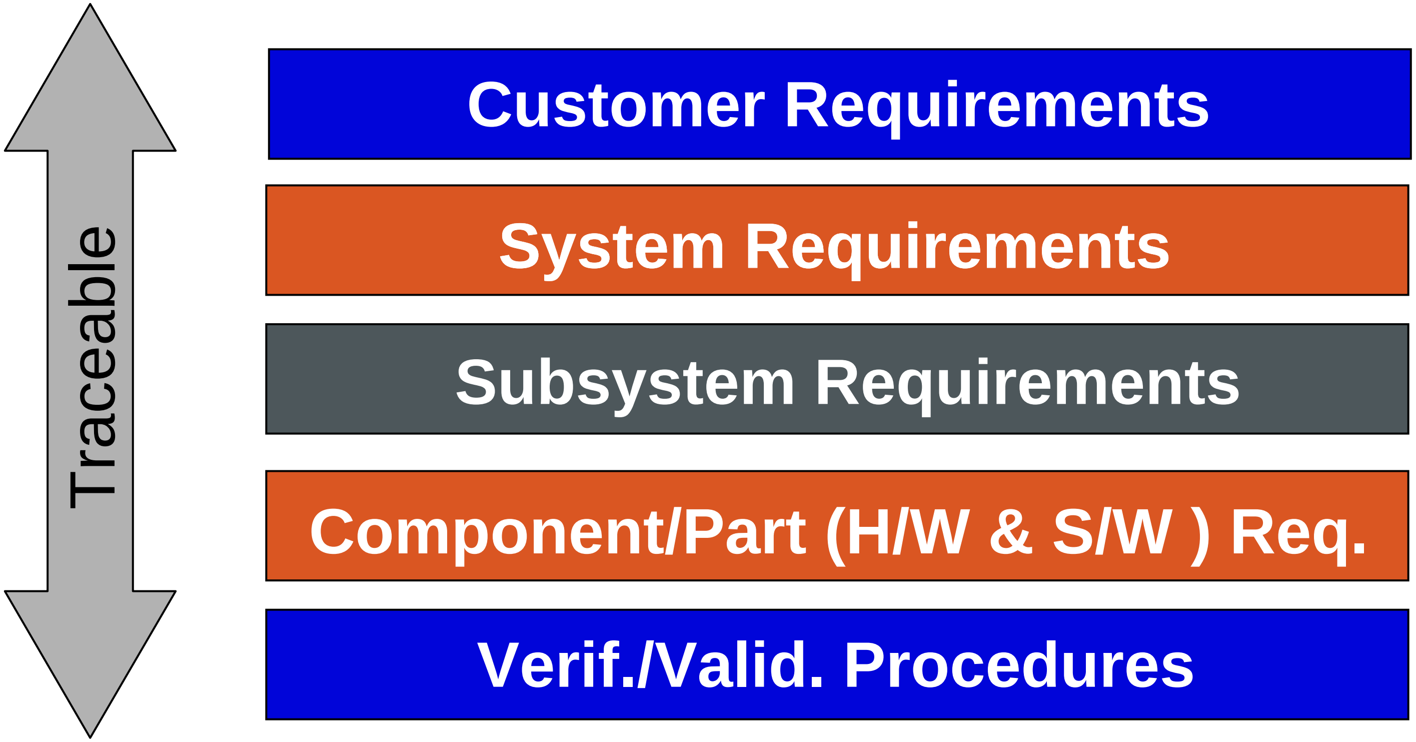
Requirement Layers

- Start with high level concept and technical requirements
- Drill down adding more detail with each layer
  - Highest level – capabilities
  - Next n levels – subsystems, architecture, high level design, low level design
  - The number is subjective - depends on complexity
  - Stop when you have enough detail to build it, buy it, code it, and test it



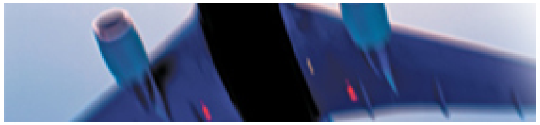


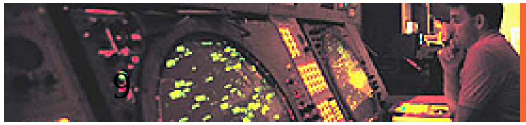
Requirements Layers



Managing Change

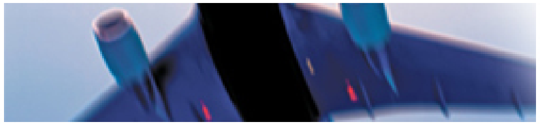
- During initiation:
  - Define and formalize change control process (internal and external)
  - Define how legacy issues will be handled
- Get to know the “customer” and learn their true priorities
- Good communications with stakeholders is key (include Contract Administrators)

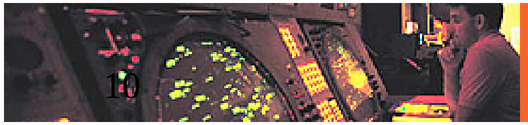




Managing Change (cont.)

- » Effectively and formally evaluate and control proposed changes
- » Hold the line even on small impact changes
- » Requirements vs. desirements (what is in the contract?)
- » Identify and address errors/issues as early as possible





Advantages of Model Based Development

» Early detection of errors in requirements and design

» Proof of concept

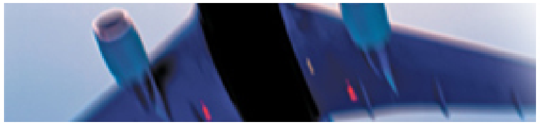
» Repeatable

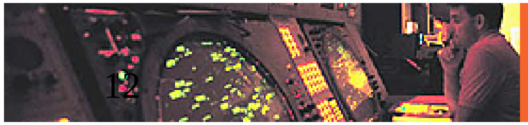
» Reduces impact of changes

» Reduces cost of downstream activities (design, code)

Rapid Prototyping

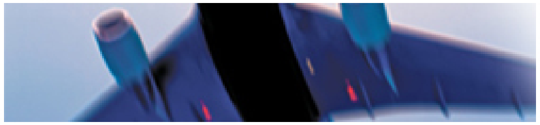
- » Formalize the process to provide proof of concept
- » Make it repeatable – what if it works?
- » Emphasis of testing on core functionality, doesn't address capabilities such as operational environment

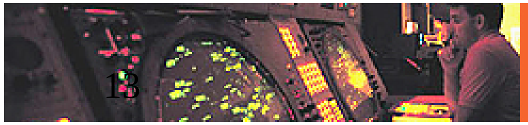




Summary

- » Create S.M.A.R.T. requirements
- » Communicate with stakeholders and dig deeper for clarification of requirements
- » Formalize the change management process
- » Identify legacy issues at the start of the project
- » Leverage modeling to detect errors early and reduce downstream costs
- » Use prototyping to help test functionality





Questions?

Contact Info:

Scott Derby
Programs Manager
Esterline AVISTA
Phone (608)348-8815
Fax (608)348-8819
Email Scott.Derby@avistainc.com
www.avistainc.com