Adaptation of the Agile Scrum Methodology for the Implementation of Multi-Mission Relay Coordination Software

Daniel Allard
Jet Propulsion Laboratory (JPL), Caltech

March 16, 2013
Biography

• Daniel Allard, Senior Software Engineer
  – daniel.a.allard@jpl.nasa.gov
  – B.S. Engineering Tufts University, 1991
  – Mission systems software developer for 22 years at JPL
  – Supported missions include Cassini, Mars Exploration Rovers (MER), Phoenix, Mars Odyssey Orbiter, Mars Reconnaissance Orbiter (MRO), Mars Science Laboratory (MSL), Mars Express Orbiter

• Expertise in web service and mission support software development
Mars Relay Operations Service (MaROS) Project Overview

- Web service supporting all relay planning and coordination activities between landers and orbiters at Mars
  - Hosted at JPL, supports all active Mars Network missions
    - MSL, MER, Odyssey, MRO, MEX
- Implementation from late 2008 through today, in operations since 2010
  - Supported by weekly user group meeting including representatives from each mission
- Small task, 3-4 part time developers mostly half time or less
Service Context

March 16, 2013
Agile Development of Mars Relay Operations Service (MaROS)
User Interfaces

Relay Functions Portal

Visualization

Relay Request Management
Adopted Aspects of Scrum Methodology

• Sprint development phases
  – “Locked” development task list sprint-by-sprint
  – Pre-sprint planning meetings
  – Burn-down charts using Google Docs
  – End-of-sprint user demos

• Daily video tag-ups via Google+
  – Supports our distributed development
Sprints Within Waterfall
Typical Major Phase

Start of new major phase

~6 Month Development Cycles

Development Sprint Development Sprint Development Sprint Testing Sprint

~Three sprints, three to four weeks per sprint

Week of downtime for sprint planning and user demo between sprints

Proper sprinting during test phase can be problematic (can’t manage a burn down, etc.)

March 16, 2013
Agile Development of Mars Relay Operations Service (MaROS)
MaROS Burn Down:
Part-Time Development Signature

“Waves” of development as developers partner to complete shared tasks, then move on to other (non-MaROS) assignments.
Lessons Learned

• Scrum adoption challenges
  – Not institutionally mandated, so no institutional support
    • The process can ‘grow stale’ if the dev team isn’t on top of it
  – Difficult to justify some Scrum overhead (e.g. Burn-down setup) when the team size is around ~1.5 developers
  – Fared poorly during test-and-bug-fix cycle, due to constant sprint content updates

• Scrum value added
  – Sprint themes added focus to work, generally better quality end-product
  – Users very happy with regular feedback loop via demos
  – Made the most difference with multiple developers working together on a shared sub-task
References and Acknowledgements

References:

Acknowledgements:
• Current development team: Roy Gladden (Task Lead), Dan Allard (Lead Developer), Frank Hy, Hayk Arutyunyan

*The research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.*

Copyright 2013 California Institute of Technology. Government sponsorship acknowledged.