OSGi Framework in Ground Systems Product Lines

GSAW 2010

Chris Newton
Ground Systems Product Line Manager
Ubiquitous S/W Challenges

- Software complexity constantly increasing
- Shorter development cycles demanded by customers
- Requirements creep as software becomes more and more complex
- Management of multiple variations for different customers
- Of course re-use will help... but two major hurdles stand in our way
  - The “one-offs” culture
    - Design philosophies vary
    - Coding styles differ
    - Protectionism (It’s my code)
    - Communications layers and decisions differ
  - Technology disagreements
    - How do we implement? Which framework? Which protocol?
- So what do we do?
  - We spend too much time integrating one-off’s instead of solving “hard problems”
- Bottom line... Large companies have a hard time maintaining a consistent culture
  - This inability breeds culture clashes, head butting, and, in general, prevents us from solving truly Hard Problems
What Is OSGi?

• A software technology framework that enables:
  • A software framework for Java that provides a robust dynamic component model
  • The product of the OSGi Alliance – a world-wide consortium of s/w organizations
    • Established in 1999
    • Members from diverse industries, including automotive, telecom, consumer electronics, etc.
  • A specification, reference implementations, test suites and certification processes
First bullet is really still "philosophy/problem statement" - not solution. Probably should go elsewhere.

Can probably drop second bullet.

s087759, 2/4/2010
The OSGi Value Proposition

- A software technology framework that enables:
  - Quick integration of software components
  - A focus on solving hard mission problems, rather than integrating infrastructure
  - A shorter software time-to-market cycle
  - Reduces complexity at the highest level by decomposing the hard problems
  - Loosely-coupled components

- Features
  - Provides a component based framework architecture
  - Dynamic removal and addition of components
  - Automatic dependency management amongst software components

- Benefits
  - Reduces time-to-market and development costs
    - Integration of pre-built and pre-tested modules
  - Reduces long-term maintenance cost
    - Test once, re-use many times
  - Enables application creation from a federation of services
    - Simpler to implement a SOA
OSGI is a flexible, effective glue that wires software components together.
How we use OSGI...