An Adaptable and Market-Driven Command and Control Architecture

GSAW - 2003
March 5, 2002

Scott Norcross
L-3 Storm Control Systems, Inc.
Scott.Norcross@L-3com.com
(703) 713-6200 x1030
COTS Solutions: Market Demands

• COTS products offer significant benefits to complex system implementations

• While benefits are acknowledged, concerns and criticism still exists regarding some COTS-based approaches

• Top concerns center around some basic attributes
  – **Functionality**
    • Ability to support necessary functions (either too much or too little function)
  – **Flexibility**
    • The ability of the product to easily adapt to mission specific requirements and end user ops concepts
  – **Cost**
    • The cost to maintain and evolve the COTS products
  – **Control**
    • The level of dependency of the end user on the COTS vendor
L-3 Storm Approach & Lessons Learned

• Long term COTS product provider
  – *L-3 Storm has been a COTS Command and Control product provider for more than a decade*

• Mission Unique Development/Integration
  – *During this time, L-3 Storm has adapted baseline COTS products to support numerous missions with mission unique requirements and ops concepts*

• Lessons Learned
  – *Sometimes “learned the hard way” how to adapt COTS products to support mission unique requirements*

• Extensive R&D
  – *Decided to invest R&D to define a comprehensive architecture capable of evolving to support future requirements.*
InControl-NG Architecture Characteristics

• A Software Product AND a Comprehensive Architecture
  – *InControl-NG designed to serve as an operational software product and a comprehensive system architecture to support evolution*

• Use of proven tools and processes
  – *Architectural design & implementation initially based on Rational processes and tools*
  – *Migration to RM-ODP to support more rigorous architectural views and evolution*

• Component-based design
  – *Well defined components and to support flexibility and encapsulation*

• Use of well-defined interfaces
  – *IDL and well structured/documentated interfaces to support third party product integration and system evolution*
  – *XML file formats to support data definition*

• Support for and use of industry standards and broad market technologies
  – *CORBA, Java, C++, XML, UML, RM-ODP, CCSDS, CCSDS/SLE, etc.*

• Careful use of open source products
  – *Careful selection and use of accepted open source products*

• Emphasis on platform independence
Support for Mission Unique Requirements

- Significant goal to satisfy mission unique requirements with minimal NRE

- Use of “layers”, “strata”, and design patterns to support unique requirements

- Goal to minimize/eliminate impact to “core” system components
  - Minimize development/integration
  - Simplify maintenance

- Support the ability to add unique processing
  - Use of well known design patterns for dynamic functions
    - Use of Chain of Responsibility design pattern to support any spacecraft specific on-board command queue
    - Use of interceptors in the telemetry engine to allow mission specific processing to be added with no impact on the core functionality
Modular and Adaptable User Interface

• Support for multiple system “views”
  – Configurable displays and ops concept support
  – Independent implementation (from baseline functions) – supporting flexibility
Support for Multiple Display Views

An Adaptable and Market Driven Command and Control Architecture
Scott Norcross
Storm Control Systems, Inc.

GSAW 2003
Summary

• Creation of a product and an architecture
• Support for industry standards for system expansion and evolution
• Strong component based design
• Well defined API’s
• Use of layers, strata, and design patterns to minimize NRE and support evolution
• Integrated and flexible user interface design