Emerging Trends in Ground Station Software Architectures

Vincent J. Kovarik Jr., Ph.D.
Software Technology Inc.
Melbourne, FL
Survival

• Ground station software is becoming market-driven.
• Monolithic or custom solutions cannot compete in a schedule and cost driven world.
• Solutions need to be:
  – Open
  – Modular
  – Reusable
  – Distributed
  – Multi-Mission/Platform
  – Client/Server
• The future belongs to plug-n-play products.
Evolving Software Architectures

- **Architectures based on frameworks (aka patterns)**
  - Model-View-Controller
  - Asynchronous Event-driven applications

- **Languages are supporting higher levels of abstraction.**
  - Representation is moving further away from the machine
  - More direct representation of the system entities.

- **Distributed computing demands more interoperability.**
  - Higher-level protocol between applications must be defined.

- **Open/Industry standards are significant criteria.**
  - Increased longevity
  - Interoperable with new products
  - SuperMOCA
Open Architecture for Ground Systems
Open Architecture for Ground Systems

[Diagram of Open Architecture for Ground Systems]

- Network Element
- Satellite
- Device Network Transport
- Telemetry Transport
- Command Transport
- Device Command Language
- Telemetry Conversion
- Logical Device Interfaces
- Transport Services
- Physical device(s)
- Open Architecture for Ground Systems
Open Architecture for Ground Systems
Open Architecture for Ground Systems

- **Planning, scheduling and system-wide resource management**
- **Device model and logical control**
- **Logical Device Interfaces**
- **Transport Services**
- **Physical device(s)**

**Device Network Transport**

**Telemetry Transport**

**Device Command Language**

**Command Transport**

**Telemetry Conversion**

**Structural Model**

**System Model**

**Behavioral Model**

**Scripting Language**

**Planner/Scheduler**
Open Architecture for Ground Systems

Human Computer Interface

- Planner/Scheduler
- System Model
- Scripting Language
- Behavioral Model
- Structural Model

Telemetry Conversion
- Device Command Language

Telemetry Transport
- Command Transport

Device Network Transport
- Satellite

Logical Device Interfaces
- Transport Services

Device model and logical control
- Planning, scheduling and system-wide resource management

Physical device(s)

Distributed HCI
Open Architecture for Ground Systems

- Human Computer Interface
- Planner/Scheduler
- System Model
- Scripting Language
- Behavioral Model
- Structural Model
- Telemetry Conversion
- Device Command Language
- Telemetry Transport
- Command Transport
- Device Network Transport
- Satellite
- Network Element

- Infrastructure and common services
  - Software BUS
  - Recording and Logging
  - Persistence / Databases
  - Orbit Determination

- Distributed HCI
  - Planning, scheduling and system-wide resource management
  - Device model and logical control
  - Logical Device Interfaces
  - Transport Services
  - Physical device(s)
• Architected as a UNIX-based distributed system.
  – Employs Client/Server architecture
• Recognized the need for and implemented a distributed software bus.
• Follows vertical partitioning and encapsulation.
  – Independent applications communicating over SW bus
• Being applied to significant satellite programs:
  – Navy Space
  – IRIDIUM
  – GPS
  – INTELSAT
Next Generation OS/COMET

• Provide a more robust model of satellite, devices, and network.
  – Support complex relationships.
  – Model-based reasoning

• Expand use of distributed computing standards.
  – Software bus, open standards

• Increased interoperability with third-party software.
  – G2, Satellite Tool Kit, Orbix, Nexpert

• Autonomous monitoring and control
  – Fault detection, mission planning

• Increased use of graphical user views.
  – Consolidate information to reduce operator overload.
The Road Ahead

- Standards in satellite industry.
- Distributed computing driven by standards.
- Message-based (context independent) interaction between applications.
- Plug-and-play software “machines” rather than software “reuse.”
- Robust representation of satellites, constellations, networks, and ground station assets.
- Winners will be those who provide significant value-added capabilities in one or more of the architecture components.