## System A’s View Of The World

<table>
<thead>
<tr>
<th></th>
<th>8025W</th>
<th>8010W</th>
<th>7955W</th>
<th>7940W</th>
</tr>
</thead>
<tbody>
<tr>
<td>02547</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Not observed</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2 observed; Only 1 real aircraft

## System B’s View Of The World

<table>
<thead>
<tr>
<th></th>
<th>8025W</th>
<th>8010W</th>
<th>7955W</th>
<th>7940W</th>
</tr>
</thead>
<tbody>
<tr>
<td>2547</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Looks like a friendly, but it’s not</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

303

306

3055N

3045N

403

1227

3055N

3045N
Joint Tactical BMC2 (today)

Common functionality, implemented and maintained many ways
Data obtained from sources outside the distributed system

Joint Tactical BMC2

Data exchanged among peers within the distributed system

Displays

Joint Tactical BMC2 domain

Sensors

System–specific Tactical BMC2

Joint Tactical BMC2 domain

Weapons

Help is needed in identifying and controlling interface (e.g., system-specific Tactical BMC2, sensors)

Service Combat and Command and Control System Program Manager’s domain

16 February, 2004
Joint Tactical BMC2 (future)

Communication Network (and Enterprise Services)

Common functionality, implemented and maintained commonly
Configuration Management

Paper Standard(s) and Specification(s)

- Gaps, overlaps, and conflicts
- Context-free
- Static
- Syntax

Integrated Architecture Behavior Model (“Platform” Independent Model”)

- Unambiguous
- Described in context
- Dynamic
- Syntax and semantics
- Strong typing

Shift from static, paper artifact to dynamic behavior model
Integrated Architecture Behavior Model

• Derived from JROC-validated requirements
• Unambiguously describes dynamic system behavior in an open source model
• Supports selection among alternative solutions
• Delivered to program managers with verification/validation data and JDEP technical framework

Idealized model of distributed system performance that shows industry what “good” looks like – automates the standards
### Precepts

#### Cornerstones
- Continuous Readiness
- Sensor Netting
- Battlespace Dominance
- Proven Lethality
- Coordinated Weapon Employment
- Joint Command Support
- Information Assurance

#### Architecture Quality Attributes

- **Performance (functionality)**
  - Correctness
  - Efficiency
  - Completeness
- **Reliability**
  - Survivability
  - Fault tolerance
  - Openness
- **Scalability**
  - Flexibility
  - Openness
- **Maintainability**
  - Openness
  - Expandability
  - Testability
- **Safety**
- **Security (Info. Assurance)**
  - Survivability
- **Verifiability**
  - Openness
- **Reusability and portability**
  - Equipment and OS independence
  - Openness

Source: IEEE-Std 1061-ISO Std 9126
MITRE Guide to Total Software Quality Control

#### Outcomes

- Reduce fratricide
- Employ weapons to design range
- Counter existing and emerging threats
- Increased performance
- Lifecycle cost avoidance
- Reduced time to field new and modified capability

---

16 February, 2004
Model Driven Architecture

1. Integrated Architecture Repository
2. Block 1 Engineering
3. "Platform" Independent Model
4. Object Oriented Analysis
5. Open Source executable UML

- Object oriented dynamic model
- Characterize BMC2 behavior of nodes in the distributed system
- Precise, durable, repeatable
- Subjected to rigorous consistency and conformance verification

Configuration Item

Isolate functionality from specific implementation technologies allows Design for Change
Model Driven Architecture

Targeted to High Level Architecture Run Time Infrastructure to support distributed development and test and evaluation.

- Integrated Architecture Repository
- “Platform” Independent Model
- “Platform” Specific Model
- Implementation
- Testing
- A Reference Implementation
- Verification & Validation
- Application
  - HLA RTI
  - Operating Systems
  - Equipment

Machine Translation

Block 1 Engineering
Model Driven Architecture

Integrated Architecture Repository

“Platform” Independent Model

“Platform” Specific Model

Implementation

Testing

Verification & Validation

Example, targeted to specific industry standards (e.g., TAO, POSIX), based on individual system needs
Model Driven Architecture

- "Platform" independent model is inherently "open", provides dynamic model of system behavior, and allows deferral of specific implementation technology decisions
- HLA-compliant model demonstrates distributed system performance
- One or more specific model(s) demonstrate distributed system performance in real system(s)
Verification and Validation

“Platform” Specific Model(s)

“Derived from consistent and conformant “Platform” Independent Model

Implementation ➔ Sensor(s) ➔ Communication Server ➔ Weapon(s)

Reference implementation

HLA RTI

Common Reference Scenario Driver ➔ Environment Driver ➔ Data Extraction

Demonstrate Correctness of Distributed System
Implementation in tactical systems

Consistent and conformant; built by joint consortium

“Platform” Independent Model

“Platform” Specific Model

Machine (or manual) translation done by System Program Managers (with help from joint consortium)

Verification and Validation by System Program Managers; Joint Independent Verification and Validation by JITC

Testing

Being developed in collaboration w/ Industry, FFRDCs & Gov. PMs (e.g., Navy Open Architecture & Air Force’s E-10A/MC2A)
Deliverables

• One “Platform” Independent Model

• Two or more example “Platform” Specific Model(s)
  - One HLA RTI-specific
  - At least one targeted to a specific communication environment and operating system

• Reference Implementation(s) derived from “platform specific model(s)"

• Unit and integration test scripts and results (verification)

• Validation test scripts and results

• **JDEP kit**
Funding Strategies

- Original Service estimates accounted for redundant development

\[ \text{Total} = \text{System A} + \text{System B} \]

- A change in business model should reduce total cost and help synchronize development

<table>
<thead>
<tr>
<th></th>
<th>Design</th>
<th>Code</th>
<th>Int</th>
<th>Test</th>
<th>Pgm Mgt</th>
</tr>
</thead>
<tbody>
<tr>
<td>System A</td>
<td>a</td>
<td>b</td>
<td>c</td>
<td>d</td>
<td>e</td>
</tr>
<tr>
<td>System B</td>
<td>a</td>
<td>b</td>
<td>c</td>
<td>d</td>
<td>e</td>
</tr>
</tbody>
</table>

Common Block 0 & Block 1

New Method

\[ \text{Total} = a + b + c + A_d + B_d + A_e + B_e + A_f + B_f \]
The message

- Integrated Architecture continues to be a key task force product
- Integrated Architecture behavior model supports dynamic analysis and improved communication with industry
- Approach changes configuration item(s) from paper specifications and standards to dynamic behavior model

The Integrated Architecture provides the basis for reducing development costs, reducing time required to field new and modified capability, and increasing operational effectiveness
Requirement sources

TAMD, CID, GIG CRDs
MIL-STD-6016B
STANAG 5516
JSLIR-16 (draft)
STANAG 5522
MIL-STD-3011
SIAP SE Technical Reports
Existing Architecture products
  - Views
  - Threads
Athena/Sea Athena/Common C&D
MSI
SRIG design
Navy OA materials

SGS/AC spec, source code
JDEP kit
SIAP Block 0 DSB
SIAP Block 1 DSB
USAF DLI/LCI/TDLCS
USAF COLE
USMC COLE

Large number of ways to describe expected performance creates gaps, overlaps, and conflicts...Integrated Architecture can force convergence and consistency
“Platform” Independent Model

• **Who**
  - Industry/University/FFRDC/Government Team

• **What**
  - Independent of computer, operating system, and “middleware”
  - Complete and correct model of an arbitrary distributed system peer
    • Syntax and semantics
    • Dynamically verifiable (unambiguous)
  - Tailored for specific implementations (e.g., AWACS, AEGIS) when “Platform” Specific Implementation is built
“Platform” Independent Model (cont.)

• Why

- Express the behavior of the distributed system in an industry standard language
- Allow verification and validation of the integrated architecture
- Change configuration management artifact from paper standard and source code to behavior model
- Support verification and validation of end product
“Platform” Independent Model (cont.)

- Where
  - Collocated team in Arlington, VA

- When
  - Block 1 System Engineering FY 02-03
  - Build and test model FY 03-05
  - Integrate and test FY 06-07
  - IOC FY 08
“Platform” Independent Model (cont.)

• How
  - Product of disciplined, but efficient system engineering process
  - Model developed by partnership of industry, university, FFRDC, government
  - Implemented and integrated by industry

• How much
  - Joint Tactical BMC2 functionality; extensible to service–unique functionality
JDEP kit contents

- Attributes Technical Reports
- Common Reference Scenarios
- Common Reference Scenario Driver
- ARCTIC
- PET
- Environment services
- Communications services
- Sensor representations
- Weapon representations