CORBA Based Next Generation C2 Architecture Security Requirements

James E. Just, Teknowledge
Rick Hayes-Roth, Teknowledge
John Schill, DARPA
Sami Saydjari, DARPA

NSA/OMG CORBA Security Workshop
April 4-5, 1997
Agenda

• Architecture Definition
• JTF ATD
• Transition
• DARPA ISO Architecture Effort
• Information Assurance
• Issues
What do WE Mean by “Architecture”?  

• Architecture is a *flexible blueprint* for a family of applications  
  – Example definition from DMSO Master Plan  
    • An architecture is the structure of components in a program/system, their interrelationships, and principles and guidelines governing their design and evolution over time  
    • HLA = Major functional elements, interfaces, and design rules, pertaining as feasible to all DoD simulation applications, and providing a common framework within which specific system architectures can be defined  

• Architectures are increasingly critical for affordability and viability  
  – In both commercial and government arenas  
    • *How Architecture Wins Technology Wars*  
      – *Harvard Business Review, March-April 1993*
Architecture Essentials

- Architecture imposes constraints on how particular applications are built
  - Architects specify and employ design abstractions
- Architecture often defines
  - Components
  - Topologies of connections of component assemblies
  - Required “standards,” for example:
    - Information types
    - Communication protocols
    - User interface guidelines
  - Assumed “platform” characteristics, for example:
    - Operating systems, libraries, middleware
    - Resources and ways to access them
    - Hardware and other devices

The architecture is good if it assures the quality of assemblies
Motives: Old, Persistent, and New

• Constrain design and development to reduce unnecessary variability
• Encourage reuse
  – Components and successful design abstractions
• Standardize practices and interfaces
• Achieve “plug and play” interoperability
  • Example: DMSO Master Plan
    – “Interoperability and reuse are limited because DoD lacks a common high-level simulation architecture. There is now a consensus that DoD must establish such an architecture to facilitate the interoperability of all types of models and simulations among themselves and with C4I Systems, as well as to facilitate the reuse of M&S components.”

• Ensure a long and effective system lifetime
• Anticipate and exploit commercial trends
What’s been Tried? What’s been Learned?

- Interoperability is hard
- “Boxology” isn’t powerful enough
- Standards profiles are too weak
- The generic objective of “reuse” is too hard
- A product-line (“domain specific”) focus can work
- Reuse depends largely on prescient architecture
  - Complex systems incorporate multiple types of interactions
- We have poor tools to support architects
- Good architectures make possible high-productivity development environments
  - Good architectures can be supported by tools that enable customers to “generate” applications themselves
Lessons Newly Learned

• “Architecture”
  – An overused and under-understood term
  – Neither a panacea nor necessarily a good idea

• Successful “Architectures” are “Environmentally Adapted”
  – You have to “fit” the environment
  – You can’t avoid real commitments
  – Weak constraints aren’t very relevant, but strong constraints are risky
  – Overly particular solutions aren’t viable for long
    • “Evolve to occupy more space or die!”

• “Architectures” aren’t cheap or easy
  – They incorporate deep knowledge
  – They require careful specification
  – They need human and automation support
  – They must be enforced and validated

• The pace of technology makes architecting more important and more difficult
Architecture ° Implementation

• The architecture consists of
  – generic or abstract components
    • capabilities
    • interfaces
    • platform portability
  – connection topology and technology
  – shared information models
• Particular systems “instantiate” the architecture
  – specific component implementations
  – specific compositions, integration, messaging
  – specific schema, distribution, and access
• The goal is to co-evolve and weave these together
  – Stable but evolving architecture
  – Effective applications, with maximum reuse
  – Exploitation of new implementation technologies that afford great benefit, relative to assimilation cost
JTF ATD Vision: Anywhere, Anytime

Information Support

Execution Forces

Nat’l Resources

CJTF

PLAN OPS LOG INTEL

JTF Component Services

CINC

Distributed Interactive Simulation

• “Come as you are” crises
• Simultaneous Conflicts
• No-plan situations
• Resource constraints
• Coalition Forces
• Operations other than war
JTF Context (Who’s Involved)
Challenges

JTF Architectural Requirements / ISO Arch/PRR 970205 # 11

• Distributed collaboration
• Heterogeneous computers
• Bandwidth adaptivity
• Coalition support
• Interoperability -- application level
• Interoperability -- easy insertion of new technology
• Agility -- extensibility, evolvability
• Agility -- COTS / GOTS
• Reusability
• Coalition requirements
• Reusability

Leverage COTS / GOTS
Meeting the Architectural Challenges

• **Inherent distributed collaboration support**
  – Model / Viewer / Controller (MVC) paradigm
  – Blackboard controllers for shared access & mediation
  – Objects and Webs
    • As human-computer language
    • Warplan object web as common link across echelons and between C2 and M&S

• **Inherent heterogeneity support**
  – Two of all COTS products

• **Inherent bandwidth adaptivity**
  – Bandwidth adaptive applications & collaboration
  – Information logistics (value based distribution)

• **Maximum reusability**
  – CORBA Components
  – Vertical Common Facilities / C4I Application Framework

• **Inherent coalition support, agility, flexibility**
  – Built-in internationalization
  – Agile objects
  – Built-in support for advanced HCI
  – Servers (with viewers and controllers) for major C4I constructs

• **Interaction / coexistence with legacy systems**
  – Data Server isolates other ATD products from access to legacy data problems
  – Security -- Equal to or better than fielded systems

• **Leverage COTS / GOTS**
  – Leverage commercial standards efforts (e.g., OMG & CORBA)
  – Able to use best COTS / GOTS technology and standards
**Example:**
JTF ATD Reference Architecture

- **User Environments**
  - Desktops: Viewers
  - Workplaces (Groups, Contexts): Controllers & Blackboards

- **Applications**
  - ANCHOR DESK
  - APPLICATIONS

- **Services Interface**
  - Model Server
  - Map Server
  - Message Server
  - Data Server
  - Situation Server
  - Plan Server
  - Comm. Server
  - Web Server

- **Object Repository**
  - JTF C² Schema
  - (C++) Core Object Schema
  - (C++) Object System
  - COE, Object Management (CORBA), & Communications

ISO Arch/RHR 970205 # 13
Example:

Architecture Essentials

Components
Topology
Information Types
Communication Protocols
User Interface Guidelines
OSs, Libraries, Middleware
Resources & Access
H/W & Other Devices
Operationally Significant JTF Activities

- Forming and reforming of COI during crisis evolution
- Accessing Open Source and Intel data from a COI
- Establishing CPT that spans Log, Ops, and Intel
- Separating sensitive planning activities and info from operations
- Controlled sharing with “coalition partners of the day”
- Role based access control with rapid changes or exceptions
- Automated production of lower level info from higher level sources
- Uniform access to CORBA and non-CORBA based systems
- Integrated security and system administration with reachback capability and enhanced automation for workload reduction
JTF ATD Development Guidelines

• Vendor independence and interoperability
  – Two of all COTS products -- HPUX and Solaris, Orbix and CORBUS (soon to be selected second commercial ORB) -- lots of pressure for NT development

• Security
  – C2 workstations, system high operations (e.g., physical and cryptographic separation), Radiant Mercury style release to coalition partners
  – Dependent of CORBASec yet to be implemented features
  – PM committed to demonstrating CORBASec features within four months of availability from Orbix
The Vision: From Components to Business Objects

Components are reusable, self-contained objects that are independent of:
- Languages
- Operating Systems
- Vendors
- Compilers
- Tools
- Address Spaces
- Networks
- Applications

Interoperable Components

Collaborating Business Suites
JTF Reference Architecture
“Application Framework for JTF C²”

User Environments

Applications

Services Interface

Object Repository

Infrastructures

Desktops: Viewers

Workplaces (Groups, Contexts): Controllers & Blackboards

Task-Force Process Management Applications
- Task Modeler
- Workflow Manager
- Monitors & Triggers

Situation Assessment & Planning Applications
- Planning Support Functions
- Decision Support Functions
- Associate Systems

Coordination, Communication, & Control Applications
- Comm. Support Functions
- Comm. Associate

Associate Systems

Coordination, Communication, & Control Applications

Task Modeler
Workflow Manager
Monitors & Triggers
Planning Support Functions
Decision Support Functions
Associate Systems
Comm. Support Functions
Comm. Associate

Desktops:

Viewers

Controllers & Blackboards

Model Server
Map Server
Message Server

Data Server
Situation Server
Plan Server

Comm. Server
Web Server

JTF C² Schema

(C++) Core Object Schema

(C++) Object System

COE, Object Management (CORBA), & Communications

ISO Arch/RHR 970205 # 18
Where are We?

• We have a critical mass of capability in JPS 2 (Oct 96)
  – It is usable by DARPA-types
  – It is like a commercial alpha release of a v1 object-oriented planning tool suite
  – We have accomplished some truly hard technical results
  • Almost everything is object-web based using CORBA 2.0 as intelligent and coherently as possible
  • We have created a solution to the OO granularity problems

• The architecture is partially implemented, works & has “mind share”

• The architecture is the starting point for several new programs throughout ISO and elsewhere

• However, we are weak in “product” qualities and support
  – Will be addressed going forward by ISO & DDJPO
The DARPA ISO Architecture

• **Initial scope (the JTF RA)**
  - Distributed collaborative situation assessment & planning
  - Essential ingredients:
    - Object webs for sharing
    - A common schema for semantic interoperability
    - Bandwidth adaptivity
    - Agility
    - Emerging OMG CORBA services

• **Significant extensions**
  - C⁴IS common foundation for C² and simulation
  - Battlefied Awareness functions
  - C³I Schema
  - DII co-evolution
  - Security and availability (Info Assurance)
  - Intell, logistics, air ops, other components
  - Java and code mobility
  - Microsoft contingency
  - Distributed systems management
  - Component-based user interface
A Top-Level “System of Systems” Architecture for Battlefield Awareness

User Needs Profiling

User Mission Goals, Plan & Cdr’s Intent

Inconsistency Detection & Management

Others’ Situation Assessments

Assimilation & Interpretation

Dissemination

User Situation Assessment

Product Manager

Packaging

Fusion

Exploitation

Collection

Inconsistency Detection & Management

Others’ Situation Assessments

User Needs Profiling

User Mission Goals, Plan & Cdr’s Intent

Product Manager

Change & Condition Monitoring

Data & Product Repositories

Other Users & Collaborators

Repository Search & Retrieval

Data & Product Repositories

KEY

● Object webs & object databases

□ Principal Function

ISO Arch/RHR 970205 # 21
C⁴IS Enhanced Architecture for “Training & Fighting in the Same Seat”

World Context

- Time Controllers
- Resource Controllers
- Script Players
- World Configuration
- Scripts

Worlds

- Viewers
- Sim. Banner
- Controllers & Blackboards
- Task Trainers

Exercise Management Applications

- Script Writer
- Configurer

Anchor Desks

- Applications

Desktops:

- Model Server
- Map Server
- Message Server
- Data Server
- Situation Server
- Plan Server
- Comm. Server
- Web Server
- Event Server
- Redirectors

C⁴IS Schema

- (C++) Core Object Schema

(C++) Object System

COE, Object Management (CORBA), & Communications
JFACC Program Goals

Develop a New Process

- An objectives-based planning, execution and assessment process for enhanced responsiveness, efficiency, effectiveness, and flexibility.

Enable the Process with Advanced Technologies and Systems

- A common plan representation provides the foundation for strategy-to-task, cross-domain (Ops-Intel-Support) integration in a continuous, dynamic, event-driven operational tempo.

Empower the JFACC with Next Generation Capabilities

- Transition advanced decision-making technologies and information processing systems to integrate and synchronize forces in time, space, and purpose.

Knowledge-based Decision Support Environment for the JFACC

Objectives-based … Cross-functionally integrated … Object-oriented
JFACC Fundamental Concepts

- Objectives-based planning, execution & assessment
- Cross-functional collaboration (horiz & vertical)
- Common Plan Representation
- Continuous planning, execution, and assessment
- Workflow management
- Continuous event monitoring
The JFACC “After Next”

Unifies Efforts of Forces in Time, Space & Purpose

Component CCs
Warfighters
Joint Force CC
Ops/Intel/Support
Allies
Support Agencies

Integrates Planning, Execution & Assessment Systems

Force Support
Air & Space Control
Force Enhancement
Planning
Force Application
Execution
Assessment

Using Advanced Applications & Services

Dist-Collab
Visualizers
Workflow Mgr
Planners
Schedulers
Integrated DBs

GCCS Leading Edge Services >> DII+

Strategy-to-Task
- Guidance derived priorities
- Common Plan Representation

Situation Driven
Continuous
- Planning
- Execution
- Assessment

Collaboration (horiz & vert)
- Human-to-Human
- Applications-to-Application
- Cross Functional
- Cross Echelon

C2 Schema Tailored to Planner Needs
- Web-based
- Object-Oriented
- Server feed
Why an Information Assurance Program?

• Situation – Threat outpacing security
  – Today: Frequent DoD break-ins 250K/yr
  – Warfighting timelines compressing
  – Forcing integration and net connections

• Future – Advanced Battlespace Information Systems
  – Almost complete reliance on information systems
  – Multilevel/multinational processing = norm
  – Integration of SCI thru foxhole inevitable
  – Distributed computing – mobile code
Information Assurance Concept

Providing Base Security Services in GCCS LES Framework

JTF ATD/GCCS-LES Baseline Architecture
IA Investment Strategy

Balanced Protection

- Risk reduction is the name of the game
- Need tools and techniques to map landscape
- Need model of adversary behavior
- Take game theory view
  — Min-max chess problem
Unified Protection

**Meshing Prevention and Detection**

- Prevent what you can
  - Firewalls
  - End-system security services
  - Know where you have holes
- Detect residue
  - Intrusion detection – attack signatures
  - Generalize to symptom finder
  - Auto-respond eventually
Network Security

Network Security Management Tools. Network management security tools will facilitate the DII supporting infrastructure.
Security Questions and Problems

- Will CORBA based C2 systems be ready for operational deployment in 18 months without compromising security?
- How will CORBA based security mechanisms interoperate with DCE and MISSI based security within GCCS and with alternative mechanisms of coalition partners?
- How will CORBAsec interoperate with Microsoft OLE/DCOM/ActiveX and evolving internet security mechanisms and technologies?
- How can CORBAsec be made more useful
  - Fault tolerant single login that can coexist with legacy logins and stronger authentication
  - Better authorization
  - Labeling
  - Easier and more affordable security administration
Security Questions and Problems (cont.)

- How make Orbs non-bypassable and tamperproof
- Can we combine near term security services based on COTS/GOTS to field relatively secure CORBA based systems
- What can be done to facilitate negotiation of trust and security privileges between untrusted enclaves, e.g., as envisioned in CIIF Security Report 960930
Conclusions

• The entire history of software has been an inexorable march from low-level, machine-oriented, physical specifications to higher and higher-level, more abstract characterization of desired behavior of systems
• The predominance of software today and in the future makes it critical to improve our control of quality, time to use, & life-cycle cost
• Architecture has been determined to be the single highest-leverage focal point on these concerns
• Architecture can provide the “context” required to make the development process predictable, repeatable, controllable, and affordable

(more)
Conclusions (cont.)

• The ISO Reference Architecture has numerous strengths
  – Exploits major technical & commercial trends
  – Encouraging synergies across programs
  – Promises reuse and savings
• The Architecture, being distributed, open & OO, lacks several capabilities
  – Adaptive performance in heterogeneous, unfriendly, and mobile environments
  – Easy tailorability and configurability
  – Configuration and systems management
  – Comprehensive security
• This is a good “case” to focus efforts on
  – The need and payoff are great
  – The tie to commercial technology is strong
  – This is the foundation for DARPA and the DARPA-DISA AITS JPO
  – Incremental progress can and must be made
• Information Assurance is key problem for current/future systems
Backup
Sharing Distributed Structured Objects Over Constrained Comms

- Structured webs of interdependent ideas and services – underlie complex plans and analyses; human / computer understandable
- Evolving webs – the essential information structure for C² collaboration
- Webs – must be easy to create, update, view, share, and exchange, even in distributed, variable, and degraded comms environments

NOTE: WWW (HTTP, HTML, etc.) does not support most of these objectives.
Replication & Consistency Management: Nearly Every Decision Product is a Compound Object that is Replicated

Bandwidth limitations make selective replication and inconsistency management essential
Common Objects Will Support Multiple Missions & Perspectives

Common Schema (Heavyweight Objects)
- Facilities
- Platforms
- Organizations
- Environment
- Intentions & Plans

Intell
- Where are they?
- What are their capabilities?
- What are their intentions?
- What is the evidence?
- What is the uncertainty?

C²
- What’s the objective?
- What’s the situation?
- What are the options?
- What are likely results?
- Select, convey & execute
- Monitor & adapt

M&S
- Starting conditions, object attributes, other assumptions
- Interaction & time management
- Instrumentation
- Inference & projection

Measurement
- English
- Metric

Belief
- Probability
- Uncertainty

Time
- Absolute
- Relative

Space
- 2D
- 3D

Base Ontologies & Notations

Schema Maintenance

Object Factory

Slim Objects
Situation Server Ideas

• Basic situation assessment facilities
  – Accept user conditions of interest to monitor
  – Monitor legacy data sources including message content
  – User notified via trigger service when COI met
  – Map Viewer-based display of entities of interest

• Status
  – Situation assessment object web attribute values accessible via WWW browsers
  – ACPT integration
    • Battle Damage Assessment target status update notification to ACPT using triggers
  – Model Server integration
    • Simple dead-reckoning model
  – Data Server trigger integration
    • Evaluation of legacy data source updates for selective notification to user applications
OPS Intel Interface in DMIF

"Requests" specify:
1) time, 2) quality, 3) content, 4) priority

- JPS  - JFACC  - ALP  - BADD COP

Planning or Execution Application

Plan Generation  Replanning

Explicit Plan Representation

ID Conditions of Interest  Update

Request Commander’s Critical Intel Requirements

Abstract Query and Mission Context

Threat estimate and Drill-down links

Product Finisher - MCS

Product Finisher - IPB

Product Finisher - Targeting

Query Decomposer

Situation Server

DDB Services

Situation Data

Dissemination Svcs

Collection Mgt Svc

Data Svcs

Info Production

Legacy DBs
e.g., MIDB, EPL

Dynamic DB
Sit Object Base
C3I Schema

Fusion Engines

Fusion Mgr and Agile Modeling

Situation Server

DMIF

“Products”
1) one time,
2) standing request,
3) continuous iterative dialog

Info Utilization

- Identity
- Location
- Capabilities
- Status
- Projections

Requirements

Utilization

Situation Object Base

Situation

“Requests” specify:
1) time, 2) quality,
3) content, 4) priority
AITS-LES Integrated Testbed Concept

Applications Services

Advanced Information Technology Services (AITS)

Information Services

COE Services

Communications Services

Battlefield Awareness

C2 Planning & Decision Making

Simulation

Applications

DII

Advanced Infrastructure

DISN

GCCS, GCSS, TBMCS, etc

Leading Edge Services (LES)
Testbed Functional Areas - integration focus on ACTDs

Battlefield Awareness
- C2/CS Planning & Decision Making
  - JFACC
  - JTF ATD
  - ALP
  - AJP
  - Joint Log

Simulation
- Communications
  - GloMo
  - TCDL
  - Warfighter’s Internet
  - Info Assurance
  - Infosec
- Information Security
  - Human Behavior Sim
  - Natural Environment Sim
  - Scenario Generation

C4ISR Integrated Testbed
- Plan Svcs
- Sitn Svcs
- Model Svcs
- Comm Svcs
- Info Svcs
- Infosec Svcs

Information Mgmt
- COA Assessment
- STOW
- SAIP
- HSI
- DMIF
- HPKB
- I3

ISO Arch/RHR 970205 # 44
### The JFACC Server Suite

**Existing Servers**

<table>
<thead>
<tr>
<th>Server</th>
<th>Service Provided</th>
</tr>
</thead>
<tbody>
<tr>
<td>Map</td>
<td>Geographic display services</td>
</tr>
<tr>
<td>Plan</td>
<td>Plan representation, storage, retrieval</td>
</tr>
<tr>
<td>Communications</td>
<td>Bandwidth adaptation</td>
</tr>
<tr>
<td>Data</td>
<td>Access to legacy data</td>
</tr>
<tr>
<td>Web</td>
<td>Storage &amp; management of webs of related information</td>
</tr>
<tr>
<td>Model</td>
<td>Modeling &amp; simulation services</td>
</tr>
<tr>
<td>Situation</td>
<td>Situation assessment services</td>
</tr>
</tbody>
</table>

**New Servers**

<table>
<thead>
<tr>
<th>Server</th>
<th>Service Provided</th>
</tr>
</thead>
<tbody>
<tr>
<td>Collaboration</td>
<td>Distributed collaboration services</td>
</tr>
<tr>
<td>Visualization</td>
<td>Common plan visualization &amp; display</td>
</tr>
</tbody>
</table>
Leverage Point Has Changed