An Evaluation of
The High Level Architecture (HLA)
as a Framework for NASA Modeling and Simulation

Michael R. Reid

Computer Sciences Corporation
Mike.Reid@gsfc.nasa.gov
Introduction

• Mission-oriented organizations increasingly rely on simulators to reduce risks and costs
• Need to reduce the costs and increase the capabilities of simulations
  – Move to component-based architectures
  – Integrate disparate simulators into larger simulation systems
  – Distribute computing
  – Separate infrastructure from simulation operations
• The DoD developed the HLA to address these issues
• NASA has simulation needs similar to the DoD’s
What is the HLA?

• The High Level Architecture (HLA) is a DoD and IEEE standard (IEEE-1516) framework that supports modeling and simulation

• “The HLA is the glue that allows you to combine computer simulations into a larger simulation.”

What the HLA Is

- Object-based software architecture
- Component-based software architecture
- Integration system
- Communications and data sharing system
- Synchronization and time management system
- Distributed computing environment
What the HLA Is Not

- Not a simulator
- Not a modeling tool
- Not a test tool
- Not a telemetry generator
- Not a data display and analysis tool
- Not a user interface
- Not a “plug and play” system
Role of the HLA

- Provides a standard architecture for simulation systems
- Facilitates distributed and multi-platform computing in simulation systems
- Integrates separate and remote applications
- Supports collaboration
- Facilitates reuse
HLA Terminology

• Federate
  – An individual HLA-compliant simulator application

• Federation
  – A simulation system composed of two or more (often many more) federates that “play” together
Components of the HLA

• A set of rules
  – Govern the overall simulation
  – Govern each participating simulator

• A Runtime Infrastructure (RTI) specification
  – COTS/GOTS software that manages the simulation and integrates the simulators

• An Object Model Template (OMT)
  – A standard for defining and documenting the form, type, and structure of data shared within a simulation
The Runtime Infrastructure (RTI)

- Part of the IEEE-1516 specification
- Software system
  - Object-based API
  - Set of daemons
- Provides six types of services to the federation
  - Federation Management
  - Declaration Management
  - Ownership Management
  - Object Management
  - Time Management
  - Data Distribution Management (DDM)
Object Modeling

- **Federation Object Model (FOM)**
  - Describes the “universe”
  - The common object model that defines the data that federates share within the federation
  - Documented in accordance with the OMT

- **Simulation Object Model (SOM)**
  - The object model which defines the data that an individual federate shares with a federation
  - Also contains some other interfacing information
  - Documented in accordance with the OMT
HLA Objects

- Analogous to objects in an OOP language
- Have state (in attributes)
- Represent the “actors” or entities that the simulation models
- Containers for persistent data
- Mechanism through which federates share data
  - Uses a publish and subscribe paradigm
- Federates create and destroy them at will
- Defined in the FOM and SOMs
What the HLA Will Not Do

- Won’t build your simulation for you
- Won’t increase the fidelity of simulators
- Won’t generate your data
- Won’t display or analyze your data
- Won’t provide a user interface
- Won’t make “plug and play” a reality
- Won’t eliminate programming
The R&D Prototype

• Developed a common HLA-interface library
  – Using C++ abstract base classes

• Created four fully HLA-compliant simulators
  – Spacecraft Controller
  – Orbit Calculator
  – Tracking Station
  – Earth

• Made the four simulators “play” together as federates in a federation

• A clock-driven simulation
The Federation

Spacecraft Controller

Landsat 7

Orbit Calculator

Earth

MagneticField Display

Magnetic Field Data

Terra

Tracking Station

L7 Orbital Elem. File

SC Tracks on Earth Map Display

RTI
The HLA and NASA

• Integrating numerous specialized simulators into larger overall space mission simulations
  – Spacecraft and instrument simulators
  – Science data generators
  – Ground system simulators

• Formation flying
  – Multiple spacecraft and ground stations

• Modeling natural Earth or planetary geophysical systems
  – Weather systems and climatic factors
  – Geological, biological, and anthropogenic activities
  – Cosmic influences
Conclusions on the HLA

• A viable technology for certain types of simulations and modeling
  – Game-like simulations
  – Modeling a “universe” where numerous actors interact with each other and come and go
  – Distributed systems
  – Systems involving different computing platforms

• Not appropriate for all simulation applications
  – Single component or single task simulators
  – Integrating hardware components