# An Introduction to the HLA
# Part 1

Roger McFarlane
School of Computer Science
McGill University
Montreal, CANADA

# Overview

- Introduction
  - What is the HLA?
  - Motivation
  - Goals
  - History
- HLA Components
  - The RTI
  - HLA Rules
  - Object Model Templates
- For Next Time …

# Introduction

- What is the HLA?
- Motivation
- Goals
- History

# What is the HLA?

- A general framework facilitating interoperability and reusability of distributed simulation components
- Developed by the Defense Modeling and Simulation Office (DSMO)
- Developed for the United States Department of Defence (DoD)
- IEEE Standard 1516-2000

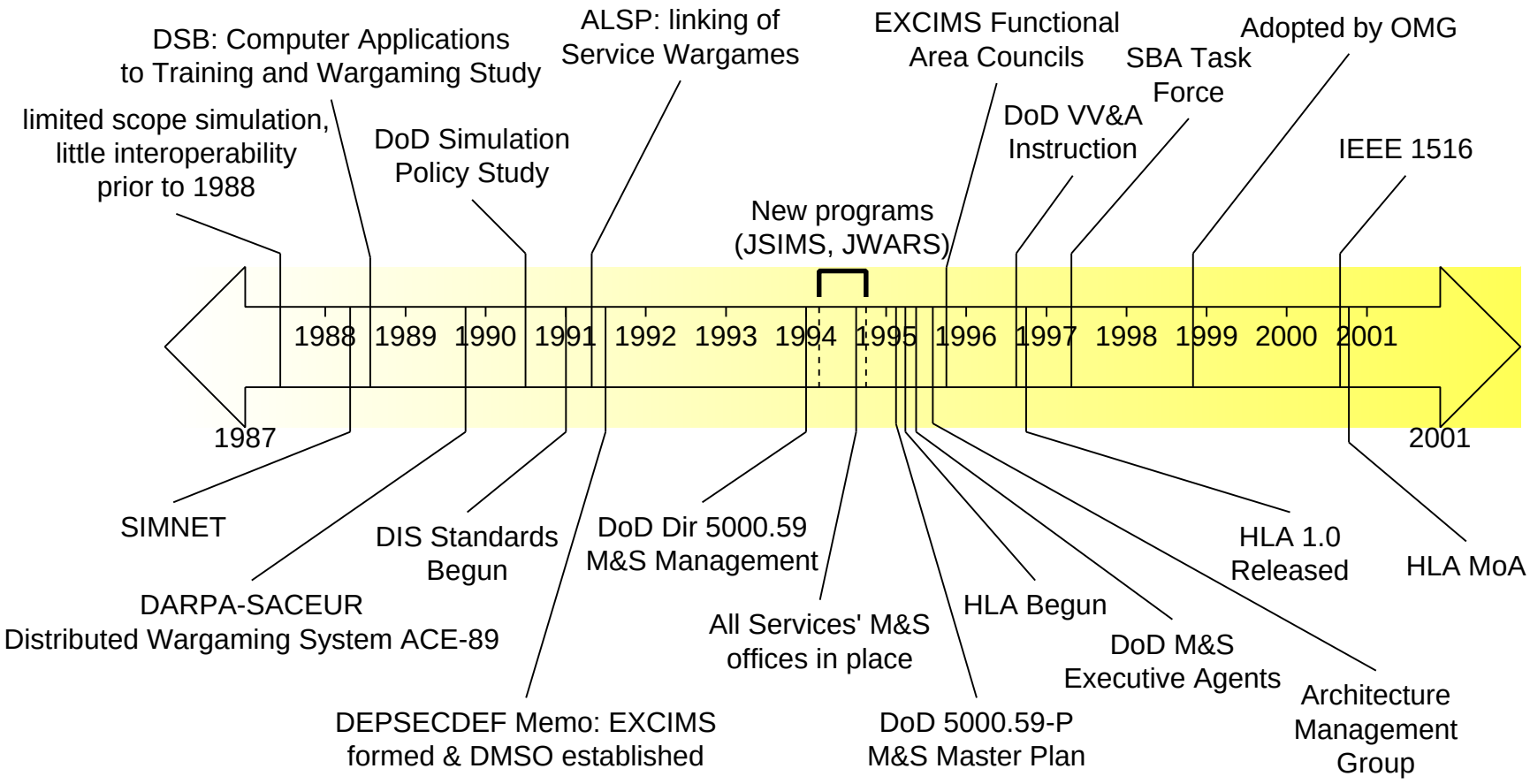
# Motivation

- Many large/complex simulations involve individual “sub-simulations” of components
- “Sub-simulations” are often heterogeneous (in the type of simulation and type of component)
- Simulators for the components may already exist
- Re-implementing or retrofitting a simulation system is risky and expensive

# Goals

- Reusability
  - A component simulation may be used in different scenarios and applications over its lifetime
- Interoperability
  - Aggregate simulations composed of multiple component simulations
  - Aggregate simulations distributed across heterogeneous hardware and software platforms
  - Reuse without significant code change or development cost
  - Combine component simulations with diverse models of computation and representation

# History


limited scope simulation, little interoperability prior to 1988
DSB: Computer Applications to Training and Wargaming Study
DoD Simulation Policy Study
ALSP: linking of Service Wargames
New programs (JSIMS, JWARS)
EXCIMS Functional Area Councils
DoD VV&A Instruction
SBA Task Force
Adopted by OMG
IEEE 1516
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
1987
2001
SIMNET
DARPA-SACEUR Distributed Wargaming System ACE-89
DIS Standards Begun
DEPSECDEF Memo: EXCIMS formed & DMSO established
DoD Dir 5000.59 M&S Management
All Services' M&S offices in place
DoD 5000.59-P M&S Master Plan
HLA Begun
DoD M&S Executive Agents
HLA 1.0 Released
Architecture Management Group
HLA MoA

# HLA Components

- Definitions & Terms
- Technical Architecture
- HLA Rules
- Object Model Templates
- Run-Time Interface Specification

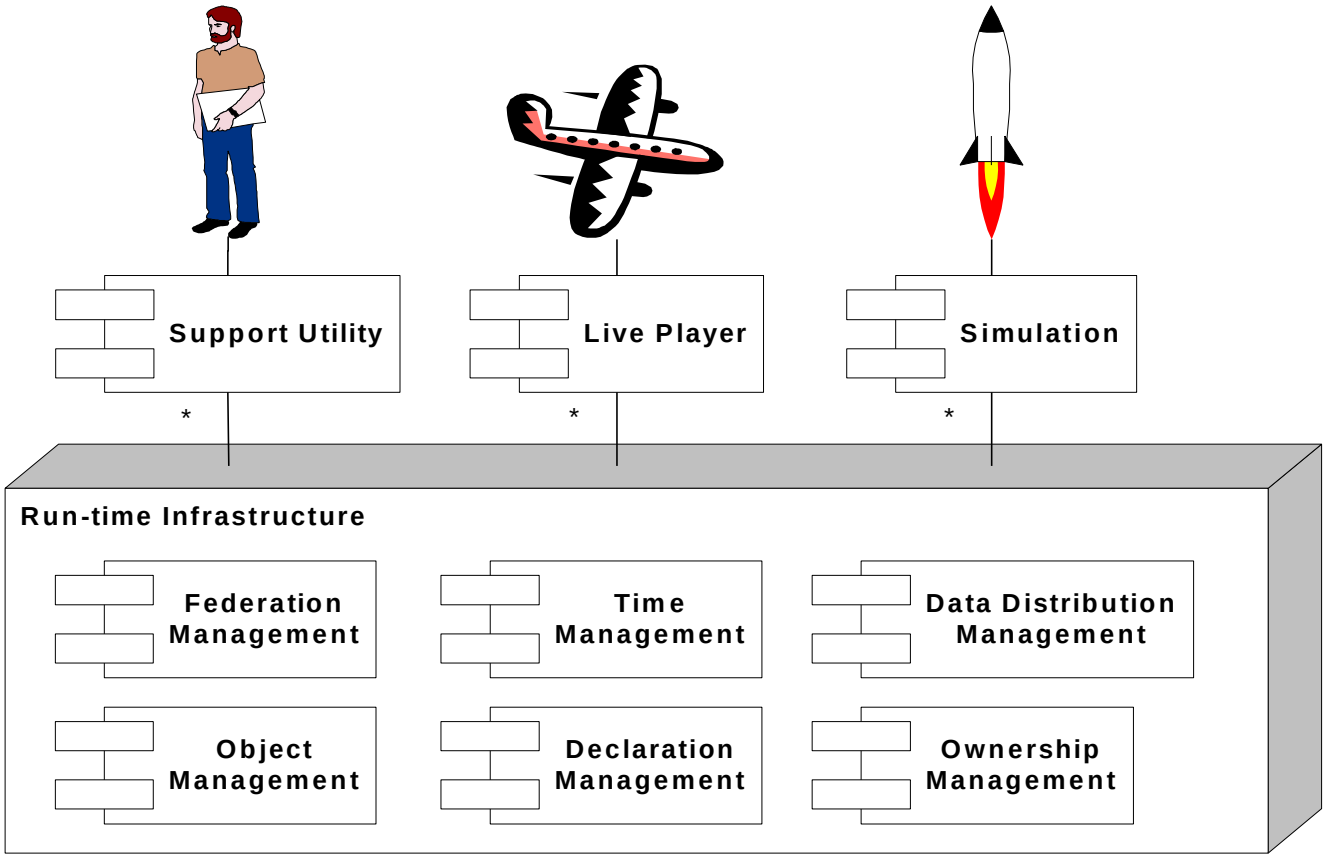
# Definitions & Terms (1)

- Federate
  - An application which supports the HLA and is capable of participating in a simulation.
- Federation
  - A declaration between federates describing how and what will be simulated.
- Federation Execution
  - A run-time instantiation of a Federation; that is, an actual simulation execution.

# Definitions & Terms (2)

- The HLA provides the **Federation** formalism by which **Federates** can be modeled such that the framework can support **Federation Execution**
- This is really no different from any other type of modelling and simulation application!

# Technical Architecture



# Example

# Run-Time Infrastructure (1)

- Software layer providing common services to federates
- RTI Specification defines the interfaces federates must use to obtain services and interact with other federates
- RTI Specification defines interfaces to be exposed by federates in order to be recognizable by the services and by other federates

# Run-Time Infrastructure (2)

- Improvements on older standards
  - DIS
  - ALSP
- Provides efficient inter-federate communications
- Separate simulation concerns from communication concerns
- Language and platform independent

# Service Groups

- Federation management
- Declaration management
- Object management
- Ownership management
- Time management
- Data Distribution management
- Support services

# Federation Management

- Controls federation-wide activities during a federation execution
- Services offered:
  - Creation and destruction of federation executions
  - Joining and resigning of federates
  - Pause/Resume federation execution
  - Save/Restore federation execution

# Declaration Management

- Manages the publisher/subscriber model for information exchange
- Services Offered:
  - Publish Object/Interaction class
  - Subscribe to Object Class Attribute
  - Subscribe to Interaction Class
  - Control Updates
  - Control Interactions

# Object Management

- Manages the lifecycle and message passing for object instances
- Services Offered:
  - Register/Discover Object
  - Update/Reflect Attribute Values
  - Send/Receive Interaction
  - Remove Object
  - Manage Transport/Ordering

# Ownership Management

- Supports cooperative modelling by allowing attribute ownership to be transferred across instances
- Services Offered:
  - Assume/Divest Attribute Ownership
  - Acquire/Release Attribute Ownership
  - Notification of ownership changes

# Time Management (1)

- Coordinates federate time advancement along the federation time axis
- Attempts to preserve causality and ordering
- Mechanisms supported:
  - Conservative synchronization (with look ahead)
  - Optimistic synchronization (e.g., time warp)
  - Hybrid methods
  - Time-stepped
  - Real-time driven

# Time Management (2)

- Federates request permission to advance their local time
- Services offered
  - Request Time Advance
  - Notification of Granting of Time Advance
  - Request Next Event
  - Notification of Granting of Next Event
  - Queue Management

# Data Distribution Management

- Efficient data transmission between federates
- Uses routing spaces to direct data only to the interested parties
  - Publisher specifies the update region
  - Subscribes specify their interest region
  - Intersection define routing space

# Support Services

- Miscellaneous functionality useful to joined federates
  - Name-to-handle transformation
  - Handle-to-name transformation
  - Setting advisory switches
  - Manipulating regions
  - RTI start-up and shutdown

# HLA Rules

- Define the behaviour and capabilities of federates and federations
- Five rules for Federates
- Five rules for Federations

# Federation Rules

- Must have an Federation Object Model (FOM) documented using the OMT
- All object representation occur in the Federates, not in the RTI
- Data exchange between instances of objects in different Federates occurs via the RTI
- Federates must interact with the RTI in accordance with the HLA Interface Specification
- During Federation Execution, an instance attribute may be owned by at most one federate at any given time

# Federate Rules

- Must have a Simulation Object Model (SOM) documented using the OMT
- Must be able to update/reflect instance attributes and send/receive interactions as specified in their SOM
- Must be able to dynamically transfer/accept ownership of attributes during federation execution as specified in their SOM
- Must be able to vary the conditions under which they provide attribute updates as specified in their SOM
- Must manage their local time in a manner which allows them to coordinate data exchange with other federates

# Object Model Templates

- Provide a mechanism for specifying data exchange and coordination within a federation
- Provide a mechanism for describing the capabilities of federate
- Facilitates design and implementation of common tools for building HLA compliant objects

# Types of Object Models

- Simulation Object Model (SOM)
- Federation Object Model (FOM)
- Management Object Model (MOM)

# SOM – Simulation Object Model

- Information exposed/consumed by a federate
  - Objects
  - Interactions
  - Attributes (of Objects and Interactions)
  - Parameters (of Objects and Interactions)

# FOM – Federation Object Model

- Specifies data exchange between federates
  - Objects
  - Interactions
  - Attributes (of Objects)
  - Parameters (of Interactions)
- Provides the “information model contract” which governs the simulation
- Provides the foundation for interoperability

# MOM – Management Object Model

- A predefined set of information elements to be included in the FOM
- Contains data relevant to Federation Execution
- Federates may also include referenced to the MOM if they may influence Federation execution.

# OMT Components (1)

- Object model identification table
- Object class structure table
- Interaction class structure table
- Attribute table
- Parameter table
- Dimension table
- Time representation table

# OMT Components (2)

- User-supplied tag table
- Synchronization table
- Transportation type table
- Switches table
- Datatype tables
- Notes table
- FOM/SOM lexicon

# Object Model Identification Table

- Describes object model's identity
- Useful for developers seeking reusable object models
- Why the object model was constructed
- How the object model was constructed
- Who knows about the object model
- Where to look for more information

# Example – Object Model Identification Table

| Category | Information |
|---|---|
| **Name** | Object Model Name |
| **Type** | “SOM” or “FOM” |
| **Version** | Version Identifier |
| **Modification Date** | Last Modified Date (YYYY-MM-DD) |
| **Purpose** | Why was this object model developed |
| **Application Domain** | Type of Application |
| **Sponsor** | Name of Sponsoring Organization |
| **POC** | Point of Contact’s Name |
| **POC Organization** | Point of Contact’s Organization |
| **POC Telephone** | Point of Contact’s Telephone Number |
| **POC Email** | Point of Contact’s Email Address |
| **References** | Where to look for further information |
| **Other** | Any other relevant data |

# Object Class Structure Table

- Defines super/sub-class relationships
- For a SOM, classes may be tagged …
  - P: The federate is capable of publishing at least one attribute of the object class.
  - S: The federate is capable of subscribing to at least one attribute of the object class.
  - PS: Both publish and subscribe
  - N: The federate is neither capable of publishing nor subscribing to any attributes of the object class.
- For a FOM, the same tags indicate if least one federate is capable of publishing or subscribing to any attribute of the object class

# Example – Object Class Structure Table

| | | | | | |
|---|---|---|---|---|---|
| HLA Object Root (N) | Customer (PS) | | | | |
| | Bill (PS) | | | | |
| | Order (PS) | | | | |
| | Employee (N) | Greeter (PS) | | | |
| | | Waiter (PS) | | | |
| | | Cashier (PS) | | | |
| | Food (S) | Main Course (PS) | | | |
| | | Appetizer (S) | Soup (S) | Clam Chowder (PS) | Manhattan (P) |
| | | | | | New England (P) |
| | | | | Beef Barley (PS) | |
| | | | Salad (S) | | |
| | | Entrée (S) | Seafood (S) | Shrimp (PS) | |
| | | | | Salmon (PS) | |
| | | | Pasta (PS) | | |

# Interaction Class Structure Table

- Specific actions which a federate may perform
- Hierarchy similar to Object Class Structure Table
- SOM Interactions may be tagged
  - P: The federate is capable of publishing the interaction class
  - S: The federate is capable of subscribing to the interaction class
  - PS: Both publish and subscribe
  - N: The federate is neither capable of publishing nor subscribing to the interaction class
- Same tags used for a FOM meaning there does (not) exist a federate capable of publishing/subscribing to the interaction class.

# Example – Interaction Class Structure Table

| | | | |
|---|---|---|---|
| HLA Object Root (N) | Customer Transaction (P) | Customer Seated (PS) | |
| | | Order Taken (P) | From Kids Menu (P) |
| | | | From Adult Menu (P) |
| | | Food Served (P) | Drink Served (P) |
| | | | Appetizer Served (P) |
| | | | Main Course Served (P) |
| | | | Dessert Served (P) |
| | | Customer Pays (P) | By Credit Card (P) |
| | | | By Cash (P) |
| | | Customer Leaves (P) | |

# Attribute Table

- Properties of an object
- May be published by the object
- Other objects may subscribe to an attribute
- Declare how/when an attribute value changes
- Declares if attribute ownership may be transferred between objects
  - DA = Divest & Acquire
  - N = Neither
- The transport used to communicate the attribute

# Example – Attribute Table

| Object | Attribute | Data Type | Update Type | Update Condition | D/A | P/S | Available Dimensions | Transportation | Order |
|---|---|---|---|---|---|---|---|---|---|
| Root | PTDO | NA | NA | NA | N | N | NA | HLAReliable | Timestamp |
| Employee | PayRate | Dollars | Cond. | Merit | DA | PS | NA | HLAReliable | Timestamp |
| | Seniority | Years | Periodic | +1/year | DA | PS | NA | HLAReliable | Timestamp |
| | Phone | Text | Cond. | Empl. Req. | DA | PS | NA | HLAReliable | Timestamp |
| | Address | Text | Cond. | Empl. Req. | DA | PS | NA | HLAReliable | Timestamp |
| Employee. Waiter | Efficiency | WaiterValue | Cond. | Perf. Rev. | DA | PS | NA | HLAReliable | Timestamp |
| | Manner | WaiterValue | Cond. | Perf. Rev. | DA | PS | NA | HLAReliable | Timestamp |
| | State | WaiterTask | Cond. | Work Flow | DA | PS | NA | HLAReliable | Timestamp |
| Food. Drink | Cups | DrinkCount | Cond. | Cust. Req. | N | PS | BarQuantity | HLAReliable | Timestamp |
| Food. Drink. Soda | Flavour | FlavourType | Cond. | Cust. Req. | N | PS | BarQuantity, SodaFlavour | HLAReliable | Timestamp |

# Parameter Table

- Additional information to characterize an interaction
- Identify the transport used to deliver the parameter
- Identify the ordering constraints for the parameter
  - Timestamp
  - Receive (indeterminate  order)

# Example – Parameter Table

| Interaction | Parameter | Datatype | Available Dimensions | Transportation | Order |
|---|---|---|---|---|---|
| Customer Seated | NA | NA | NA | HLAReliable | Timestamp |
| FoodServed. MainCourse Served. | TemperatureOK | ServiceStat | WaiterID | HLAReliable | Timestamp |
| | AccuracyOK | ServiceStat | | | |
| | TimelinessOK | HLABoolean | | | |

# Dimension Table

- Maps domain specific data values onto integer values ranging from zero to some upper bound
- Specifies the legal values which may be transmitted across the RTI
- Enables Data Distribution Management (DDM) and Declaration Management (DM)
- Used to specify update and subscribe regions to the RTI

# Example – Dimension Table

| Name | DataType | Upper Bound | Normalization | Value If Not Specified |
|---|---|---|---|---|
| SodaFlavour | flavourType | 3 | LinearEnumerated( Flavour, {Cola, Orange, Grape}) | [0..3) |
| BarQuantity | DrinkCount | 25 | Linear( NumberCups, 0, 25) | [0..1) |
| WaiterId | EmpId | 20 | Linear(WaiterId, 0, 20) | Excluded |

# Time Representation Table

- Declares the format used to represent time
  - For a federate
  - Across a federation
- Declares the semantics of time
  - For a federate
  - Across a federation
- Used by the RTI to coordinate federates during federation execution

# Example – Time Representation Table

| Category | Datatype | Semantics |
|---|---|---|
| Timestamp | TimeType | Floating point value expressed in minutes |
| LookAhead | LAType | Non-negative floating point value expressed in minutes |

# User-Supplied Tag Table

- Extensible mechanism for specifying auxiliary data
- Provides additional control and coordination of services provided by the HLA

# Example – User-Supplied Tag Table

| Category | Datatype | Semantics |
|---|---|---|
| Update/Reflect | NA | NA |
| Send/Receive | NA | NA |
| Delete/Remove | HLAascii | Reason for deletion |
| Divestiture Request | PriorityLevel | High value for immediate transfer |
| Divestiture Completion | NA | NA |
| Acquisition Request | PriorityLevel | High value for immediate transfer |
| Request Update | NA | NA |

# Synchronization Table

- Provides a federate synchronization mechanism
- Federates declare the synchronization points they support
- Federations describe the synchronization points to be used

# Example – Synchronization Table

| Label | Tag Datatype | Capability | Semantics |
|---|---|---|---|
| InitialPublish | NA | Achieve | Achieved when all classes are published and subscribed, and all initially present objects are registered |
| InitialUpdate | NA | Achieve | Achieved when instance attribute values for all initially present objects are updated |
| BeginTimeAdvance | NA | Achieve | Achieved when time management services are invoked |
| PauseExecution | TimeType | Register<br>Achieve | Achieved when the time advance after the time in the user-supplied tag is attained; time advance requests should then cease |

# Transportation Type Table

- The RTI provides different mechanisms for transport of interactions and attributes between federates
- Allows a federate designer to describe the transports supported by the federate
- Allows federation designers to describe the transportation contracts between federates

# Example – Transportation Type Table

| Name | Description |
|---|---|
| HLAreliable | Provide reliable delivery of data in the sense that TCP/IP delivers its data reliably |
| HLAbestEffort | Make an effort to deliver data in the sense that UDP provides best-effort delivery |
| LowLatency | Choose the delivery mechanism that results in the lowest latency from service initiation to callback invocation at the receiving federate |

# Switches Table

- Configuration of RTI activities performed on behalf of a federate
- A few services are configured globally for the federation
  - Auto Provide, Convey Region Designator Sets
- Most services are configured per federate
  - Attribute Scope Advisory, Attribute Relevance Advisory, Object Class Relevance Advisory, Service Reporting
- Services may be either enabled or disabled

# Switch Definitions (1)

- Auto Provide
  - (Global) Should the RTI automatically solicit updates from instance attribute owners when an object is discovered.
- Convey Region Designator Sets
  - (Global) Should the RTI provide the optional Sent Region Set argument with invocations of Reflect Attribute Values and Receive Interaction.
- Attribute Scope Advisory
  - Should the RTI advise federates when attributes of an object instance come into or go out of scope.

# Switch Definitions (2)

- Attribute Relevance Advisory
  - Should the RTI advise federates about whether they should provide attribute value updates for the value of an attribute of an object instance.
- Object Class Relevance Advisory
  - Should the RTI advise federates about whether they should register instances of an object class.
- Interaction Relevance Advisory
  - Should the RTI advise federates about whether they should send interactions of an interaction class.
- Service Reporting
  - Should the RTI report service invocations using MOM.

# Example – Switches Table

| Switch | Setting |
|---|---|
| Auto provide | Disabled |
| Convey region designator sets | Disabled |
| Attribute scope advisory | Enabled |
| Attribute relevance advisory | Enabled |
| Object class relevance advisory | Enabled |
| Interaction relevance advisory | Enabled |
| Service reporting | Disabled |

# Data Type Tables (1)

- Globally define data types referenced in other tables
- Basic Data Table
  - Name, Size in Bits, Interpretation, Endian, Encoding
- Simple (Scalar) Data Table
  - Name, Representation, Units, Resolution, Accuracy, Semantics
- Enumerated Data Table
  - Name, Representation, Enumerator, Values, Semantics

# Data Type Tables (2)

- Array Data Table
  - Name, Element Type, Cardinality, Encoding, Semantics
- Fixed Record Data Table
  - Record Name, Field-{Name,Type,Semantics}*, Encoding, Semantics
- Variant Record Data Table
  - Record Name, Encoding, Semantics, Discriminant-{Name, Type, Semantics}*, Alternative--{Name,Type,Semantics}*

# Notes Table

- Named annotations may be attached to any OMT entry
- A set of name/value pairs
- Value is free form explanatory text
- Name uniquely identifies the corresponding explanatory text
- Notes may be referenced multiple times

# FOM/SOM Lexicon

- Name/Value pairs
- Dictionary tables associating every class, attribute, interaction, parameter, etc (by name) with a free form text description (value)

# For Next Time …

- A deeper look at the RTI

# References (1)


- IEEE Std 1516-2000, IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) - Framework and Rules.
- IEEE Std 1516.1-2000, IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) - Federate Interface Specification
- IEEE Std 1516.2-2000, IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) - Object Model Template (OMT) Specification.

# References (2)


- Roy Crosbie and John Zenor, “High Level Architecture, Module 1 – Basic Concepts, Parts 1-6.” California State University, Chico. http://www.ecst.csuchico.edu/~hla
- <Steffen Strassburger's text>