News in HLA 4 Object Modelling for Developers

Mikael Karlsson, Pitch Technologies, Sweden
Abstract

HLA Evolved to HLA 4

This presentation provides a detailed look at the changes from HLA Evolved to HLA 4 from a developer perspective.

We look at changes in Object Modelling, including merge rules, and how they can be used to improve FOM modelling.

This presentation does not cover the history of simulation, policy making, etc. This is for federate (and RTI) developers.
Unsigned Integers

Unsigned data types (that match existing signed types) have been added to FOM and Encoding Helpers.

<table>
<thead>
<tr>
<th>In HLA Evolved</th>
<th>Added in HLA 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>HLAinteger8</td>
<td>HLAUnsignedInteger8</td>
</tr>
<tr>
<td>HLAinteger16BE</td>
<td>HLAUnsignedInteger16BE</td>
</tr>
<tr>
<td>HLAinteger16LE</td>
<td>HLAUnsignedInteger16LE</td>
</tr>
<tr>
<td>HLAinteger32BE</td>
<td>HLAUnsignedInteger32BE</td>
</tr>
<tr>
<td>HLAinteger32LE</td>
<td>HLAUnsignedInteger32LE</td>
</tr>
<tr>
<td>HLAinteger64BE</td>
<td>HLAUnsignedInteger64BE</td>
</tr>
<tr>
<td>HLAinteger64LE</td>
<td>HLAUnsignedInteger64LE</td>
</tr>
</tbody>
</table>
HLA Evolved allows adding new classes.

**Base FOM**
- Car

**Extension FOM**
- Train

**Result**
- Car
  - Train
HLA Evolved allows adding new subclasses.
HLA 4 allows adding new attributes to existing object classes, even at runtime.
HLA 4 allows adding new parameters to existing interaction classes, even at runtime.

Base FOM

WeaponFire
MunitionType

Extension FOM

WeaponFire
CustomParameter

Result

WeaponFire
MunitionType
CustomParameter
FOM Merge – Extendable Enums

Add values to existing enums.

Base FOM

```
FuelTypeEnum
Gasoline = 1
Diesel = 2
Hydrogen = 3
```

Extension FOM

```
FuelTypeEnum
Gasoline = 1
Diesel = 2
Hydrogen = 3
Ethanol = 4
Electric = 5
```

Result

```
FuelTypeEnum
Gasoline = 1
Diesel = 2
Hydrogen = 3
Ethanol = 4
Electric = 5
```
FOM Merge – Extendable Variant Records

Add variants to existing variant records

New encoding **HLAextendableVariantRecord**

**Base FOM**

**FuelCapacityRecord**
Gasoline : Liter
Diesel : Liter
Hydrogen : Kilogram

**Extension FOM**

**FuelCapacityRecord**
Ethanol : Liter
Electric : kWh

**Result**

**FuelCapacityRecord**
Gasoline : Liter
Diesel : Liter
Hydrogen : Kilogram
Ethanol : Liter
Electric : kWh
In HLA Evolved, dimensions are specified on each attribute.

```
BaseClass
attrX Dim1, Dim2
attrY Dim2

SubClass
attrZ Dim3
```

getAvailableDimensionsForClassAttribute(BaseClass, attrX) => ( Dim1, Dim2 )

getAvailableDimensionsForClassAttribute(BaseClass, attrY) => ( Dim2 )

getAvailableDimensionsForClassAttribute(SubClass, attrZ) => ( Dim3 )

Dim3 is not available on attrX and attrY so we cannot filter them using Dim3.
In HLA 4, dimensions are specified on the object class, and they are inherited.

- All dimensions are available for all attributes
- This makes it possible to use all the dimensions to filter all the attributes.
- For example, on a SubClass object, you can use Dim3 to filter attrX and attrY.
With HLA 4, you can add dimensions to standard FOMs.

Base FOM

**Car**
Dimensions: X, Y

Extension FOM

**Car**
Dimensions: Color

Result

**Car**
Dimensions: X, Y, Color
In HLA 4, attributes can be marked in the FOM as **Required**. This does not affect the FDD or the behavior of the RTI.

“any federate that registers an instance of this class shall either publish and provide a value for this attribute or allow another federate to take ownership of the attribute and provide a value”
In HLA Evolved, you could refer to objects using a name, a unique id, etc. But this was ad hoc.

Car
- Name = "Red Car 14"
- Driver = "John"
- Passengers = ["Jack", "Jill"]

Person
- Name = "John"
Object References

HLA 4 formalizes references using a Reference Datatype.

The Reference Datatype defines the type of the value, what kind of object it references, and which field of that object that shall match.

Person

Name = “John”

Driver : PersonReference = “John”

Car

Name : String = “Car #1”

PersonReference

Representation: String

Referenced class: Person

Referenced attribute: Name
Send Directed Interaction

Similar to Send Interaction but takes an object instance as parameter. Directed interaction is delivered to all federates that own attributes on the specified object instance.

<table>
<thead>
<tr>
<th>Object</th>
<th>Directed Interaction</th>
<th>P/S</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ambulance</td>
<td>MedevacRequest</td>
<td>S</td>
</tr>
<tr>
<td>Soldier</td>
<td>MedevacRequest</td>
<td>P</td>
</tr>
</tbody>
</table>

publishDirectedInteraction(Soldier, MedevacRequest)
subscribeDirectedInteraction(Ambulance, MedevacRequest)
sendDirectedInteraction(MedevacRequest, ambulance1, parameters, …)
receiveDirectedInteraction(MedevacRequest, ambulance1, parameters, …)
HLA Evolved
HLAinteger32BE y;
y.set(value);
auto encodedBuffer = y.encode();

HLA 4
auto encodedBuffer = HLAinteger32BE()
  .set(value)
  .encode();
In HLA Evolved, MOM logging was a free-form, implementation-dependent text format.

In HLA 4, MOM logging uses a documented json-like format.

```
[  
  [  
    "AttributeHandle<1001>",  
    "xFDcMa"  
  ],  
  [  
    "AttributeHandle<1002>",  
    "R6lkh1a"  
  ]
]
```
Improved Switches

In HLA Evolved, switches were managed by a confusing mix of FOM settings, API calls, and MOM attributes and interactions.

In HLA 4, switch handling is unified.

- All switches are available in the FOM
- All switches have get (and set) methods in the API
- All switches can be accessed through the MOM
Summary of FOM Changes

- Unsigned Integers
- Add attributes to existing classes
- Add parameters to existing interaction classes
- Extendable enums and variant records
- Dimensions defined per class, not per attribute
- Dimensions are now inherited
- Add dimensions to existing classes and interactions
- Required attributes
- Object references
- Directed Interactions (to owner of attributes)
- Improved Encoding Helpers
- Improved MOM logging
- Improved switches
Simulation Interoperability Standards Organization

“Simulation Interoperability & Reuse through Standards”

Q&A / Discussion