Scalable and reusable models for HLA-Ptolemy cosimulation framework

2016 Ptolemy Miniconference

David Côme, Janette Cardoso, Pierre Siron

October 15, 2015
HLA three main goals

Interoperability
Plug and play federates inside a federation

Reusability
A federate can easily be used in another federation

Reproductability
Same results from one run to another
The HLA world

- Federate 1
  - register objects
  - publish updates

- HLA Objects

- Federate 2
  - discover object
  - receive updates

- RTI
  - actual data exchange

- FOM
  - Classes: Attributes
Reusability with the previous version

Goal

Reuse models from one simulation to another without any changes

Can’t make any assumption on the number of HLA objects to handle

Non scalable model
Key idea number 1

Use Ptolemy classes and objects to simplify the design phase

One HLA class -> One Ptolemy class with HLASubscriber inside
Key idea number 2

Instantiate a new actor when an object is discovered if needed

Minimal model design required

Previous model handling 3 objects
Federation:

- 3 bouncing balls, each own by a federate
- The display shown previous slide
What to do with new objects

Main idea
Use existing instances like role model

Visual explanation of the policy used
ChangeRequest Semantics

Removing an actor

\[ CR^-_{A_j} : H \mapsto \{ A_i \in H, i \neq j \} \]  \hspace{1cm} (1)

Adding an actor

\[ CR^+_{A} : H \mapsto H \cup \{ A \} \]  \hspace{1cm} (2)

An atomic actor \( A \) is extended in

\[ A' = (I, O, S', s'_0, F', P', D', T') \]  \hspace{1cm} (3)

with

\[ S' = S \cup \{ CR \} \]  \hspace{1cm} (4)
Extended actor

\[ s'_0 = s_0 \cup \{ CR \mapsto \emptyset \} \] (5)

\[ F'(s, x) = \begin{cases} F(s, x) & \text{if } CR = \emptyset \\ \forall o \in O, o = \bot & \text{otherwise} \end{cases} \] (6)

\[ P'(s, x) = \begin{cases} P(s, x) & \text{if } CR = \emptyset \\ s & \text{otherwise} \end{cases} \] (7)

\[ D'(s, x) = \begin{cases} D(s, x) & \text{if } CR = \emptyset \\ 0 & \text{otherwise} \end{cases} \] (8)

\[ T'(s, x, d) = \begin{cases} T(s, x, d) & \text{if } CR = \emptyset \\ (s, CR \mapsto \emptyset) & \text{otherwise} \end{cases} \] (9)
\[
\bigcirc_{f \in A} f(H) = \begin{cases} 
    f_1 \circ \cdots \circ f_n(H) & \text{if } H \neq \emptyset \\
    \text{Identity} & \text{otherwise}
\end{cases}
\]

(10)

\[
CRs = \bigcup_{i=1}^{n} CR
\]

(11)

\[
P'(s, x) = \{ P(s, x) | H \mapsto \bigcirc_{f \in CRs} f(H) \}
\]

(12)
Mechanism implementation

For each iteration
execute Change Requests
initialize new actors
pre = director.prefire
    processHLAMessages()
if (pre = true)
director.fire
for each actor A to fire
    if (A.prefire)
        A.fire
        A.postfire
director.postfire
Conclusion

- HLA and Ptolemy: both great technologies
- HLA-CERTI: tries to bring the best of them to model designers

- Tried to eased the designer life
- First good step, more work to do