Service Provisioning for HLA-based Distributed Simulation on the Grid

Yong Xie
SMA Programme, Singapore

Yong-Meng Teo
NUS, Singapore

Wentong Cai & S. J. Turner
NTU, Singapore

Outline

- Background & motivation
- Objective
- Design
- Implementation
- Experiments
- Discussion
- Conclusion & future work
Distributed Simulation

- Provides a way of linking simulation components (federates) of various types at possibly different locations to create a common virtual environment (federation).

High Level Architecture

- Federation
- Passive Viewers
- Simulations
- Simulation Surrogates
- Interface

Run-Time Infrastructure (RTI)
- Federation Management
- Object Management
- Time Management
- Declaration Management
- Ownership Management
- Data Distribution Management
Motivation: Provision of Services

- Distributed Simulation over the Wide-Area-Network using IEEE HLA/RTI requires:
  - Hardware and software
  - Arrangements beforehand
  - Increase in scale and complexity of simulation -> large amounts of resources, provision of services are more difficult
  - Resources: RTI execution services, simulation model, underlying DS infrastructure
The Grid

OGSA
OGSI/WSRF
Grid Service
Web Service
Globus Toolkit

Objective

► To support RTI service to be used on demand
► To support dynamic discovery of federations
► To provide a standard HLA API: for interoperability and reusability
► To overcome the limitation of firewalls in traditional HLA/RTI implementation (e.g. DMSO HLA/RTI)
► To support hierarchical federation
Design

Grid Network

Client 1
...
Client n

Federation 1

Resource

RtiExec

Proxies...

FedExec1...m

Federation m

Design (Client-Proxy-RTI)

Grid Services: indexing, discovery, resource management, monitoring services ...

Globus

Simulation Code

Grid-enabled HLA API

Globus

Proxy

Grid-enabled HLA API

HLA API

Globus

LAN

Globus

LAP

Proxies & Federates & RTI

HLA API

LAN

LAN

Globus

Grid Network

Client

Resource
Design: Client-Proxy

Federate code

Embed RTIamb. Method calls into Grid service invocations

MyFederateAmb. Grid Service

ProxyRTIamb. Grid Service

Embed ProxyFedAmb callbacks into Grid service invocations

Proxy

Grid Network

Design: Proxy-RTI

Proxy

ProxyRTIamb. Grid Service

Embed ProxyFedAmb callbacks into Grid service invocations

Grid Network

RTI

FedAmb.

RTIamb.

Local Area Network (LAN)
Implementation

- Prototype of proposed framework is implemented in Java using DMSO RTI 1.3NG v6
- Grid system runs the Globus Toolkit v.3
- Implmented the RTIambassador services’ API for Federation Management, Time Management, Object Management, Declaration Management, and Ownership Management
- Data Distribution Management (DDM) implementation in process

Implementation: Client Side

```
Federate Code

NullFederateAmbassador

HLAGridNullFedAmb OperationProvider

MyFedAmb ClientProvider

HLAGridRTIamb

Grid Service invocation

Invoke RTIamb Grid Service
```
### Implementation: Proxy Side

- **OperationProvider**
  - **ProxyServiceProvider**
    - **ProxyRTIComponent**
      - **RTIambassador**
    - **ProxyFedComponent**
    - **NullFederateAmb.**
  - **Invoke RTIamb Grid Service**
  - **Invoke Client’s Grid Service**

### Implementation: Client → Proxy

- **Client side**
  - RTIamb method call
  - Encode parameters
  - Invoke Grid service
  - Decode parameters
  - Actual RTIamb call
  - Decode result
  - Resume execution

- **Proxy side**
  - Decode parameters
  - Encode result

---

28 June, 2005  PADS 2005  15

28 June, 2005  PADS 2005  16
Implementation: Proxy → Client

Experiments

- Investigate the overhead incurred in the proposed framework
- Converted the benchmark programs from DMSO’s HLA/RTI package into Java, and tested under different network configurations
- Two main benchmarks: Latency and Time Advancement
- Testing environments:
  - Linux cluster in Parallel and Distributed Computing Center at Nanyang Technological University
  - Linux workstations in School of Computer Science at Birmingham University
Experiments Environment

Experiments (Hardware)

<table>
<thead>
<tr>
<th></th>
<th>$M_{proxy1}$</th>
<th>$M_{proxy2}$</th>
<th>$M_{proxy2}$</th>
<th>$M_{proxy2}$</th>
<th>$M_{proxy2}$</th>
<th>$M_{proxy2}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>4x Pentium III 500MHz</td>
<td>Pentium III 733MHz</td>
<td>2x Pentium III 733MHz</td>
<td>AMD Athlon 1.5GHz</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Memory</td>
<td>1 Gbyte</td>
<td>1 Gbyte</td>
<td>1 Gbyte</td>
<td>2 Gbyte</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OS</td>
<td>Redhat Linux 7.0</td>
<td>Redhat Linux 7.0</td>
<td>Redhat Linux 7.0</td>
<td>Redhat Linux 7.3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc</td>
<td>3.0.2</td>
<td>3.0.2</td>
<td>3.0.2</td>
<td>3.0.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HLA</td>
<td>DMSO NG 1.3 V6</td>
<td>DMSO NG 1.3 V6</td>
<td>DMSO NG 1.3 V6</td>
<td>DMSO NG 1.3 V6</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Experiments (Results)

<table>
<thead>
<tr>
<th></th>
<th>HLA</th>
<th>HLAGrid</th>
</tr>
</thead>
<tbody>
<tr>
<td>Latency</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cluster</td>
<td>10 ms</td>
<td>50 ms</td>
</tr>
<tr>
<td>WAN</td>
<td>305 ms</td>
<td>1,200 ms</td>
</tr>
<tr>
<td>Time Advancement</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cluster</td>
<td>680 s</td>
<td>150 s</td>
</tr>
<tr>
<td>WAN</td>
<td>2 s</td>
<td>0.41 s</td>
</tr>
</tbody>
</table>

- Overhead in cluster: latency = 40 millisecond
- Use of GT3, encoding/decoding of parameters/results, and the communication costs
- Overhead in WAN: latency = 895 millisecond
  - Mainly caused by the increase in communication using SOAP messages over long distances -> increase number of packets
Benefits of the Architecture

► Decoupling
  ► Security
  ► Heterogeneity
  ► Transparency

Benefits of the Architecture

► Flexibility
  ► Multi-federation
  ► Hierarchical Federation
  ► Hybrid Federation
Benefits of the Architecture

- Flexibility
  - Multi-federation
  - Hierarchical Federation
  - Hybrid Federation

![Diagram of a client federate connecting to RTI proxy through client federate and federate.]
Conclusion & Future Work

- Design and implement a framework to extend HLA to support Grid-wide distributed simulation using Federate-Proxy-RTI architecture
- Provision of services through Grid Service invocations
- Prototype using Glogus Toolkit v3 and DMSO RTI 1.3NG v6
- Overhead incurred in the prototype suggests coarse-grained application
- Future work: federate migration, fault tolerance, integration with security

Questions?