Open Architecture Publish-Subscribe Benchmarking

Bruce McCormick, EG&G
Leslie Madden, NSWCDD
Objectives

- To gain familiarity with existing publish-subscribe middleware
- Understand ability of products to support bounded latencies and sustained throughput required by combat system applications
- Understand architectural differences in products that may affect ability of products to scale to large numbers of processors & processes
- Provide a baseline of comparison as products migrate toward DDS compliance
THALES SPLICE 2

Node

Application

Publish/Subscribe API

Pub/Sub M/W

Data Store

SPLICE Daemon

Network
RTI NDDS 3.0M

Node

Application \textit{callback}

Application \textit{poll for data}

Publication
Subscription

Publish/Subscribe API

Publish/Subscribe M/W

Network

Publish/Subscribe API

Publish/Subscribe M/W
## Architectural Features Affecting Results

<table>
<thead>
<tr>
<th>SPLICE</th>
<th>NDDS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stores data into in-memory data store for future access</td>
<td>Receives data into message queues</td>
</tr>
<tr>
<td>Implements reliable but not FIFO – most recent message issue (based on timestamp) overwrites previous issue</td>
<td>Implements Reliable/FIFO – all issues delivered. Improperly sized queues can lead to flow control behavior</td>
</tr>
</tbody>
</table>
Mett Tool

Middleware Evaluation Test Tool

- Baseline
- Environment Ctrl
- TimeSync Verification
- Tool Select
- Utility Select
- Service Select

Help
Quit
Mett Stubs

Test Stub Window

Test Configuration Window
Test Environment

- All nodes in the same subnet
- Use multicast
- Privileged User
  - Mett sets high (Max – 5) RT thread and process priorities
  - Not bound to any CPU (SMP)
## Host Configuration

<table>
<thead>
<tr>
<th>Host</th>
<th>OS</th>
<th>Description</th>
<th>Network Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alphe1</td>
<td>Solaris 8</td>
<td>SUN Ultra2 w/ Dual 200 MHz UltraSparc - sun4u 128 MB RAM</td>
<td>Fast Ethernet</td>
</tr>
<tr>
<td>Alphe2</td>
<td>Solaris 8</td>
<td>SUN Ultra2 w/ Dual 200 MHz UltraSparc - sun4u 128 MB RAM</td>
<td>Fast Ethernet</td>
</tr>
<tr>
<td>Alphe3</td>
<td>Solaris 8</td>
<td>SUN Ultra2 w/ Dual 200 MHz UltraSparc - sun4u 128 MB RAM</td>
<td>Fast Ethernet</td>
</tr>
<tr>
<td>Alphe4</td>
<td>Solaris 8</td>
<td>SUN Ultra2 w/ Dual 200 MHz UltraSparc - sun4u 128 MB RAM</td>
<td>Fast Ethernet</td>
</tr>
<tr>
<td>Serpens</td>
<td>Solaris 8</td>
<td>SUN Ultra-Enterprise 8 processor UltraSparc – sun4u</td>
<td>Fast Ethernet</td>
</tr>
<tr>
<td>Alphe4</td>
<td>Solaris 8</td>
<td>SUN Ultra2 w/ Dual 200 MHz UltraSparc - sun4u 128 MB RAM</td>
<td>Fast Ethernet</td>
</tr>
<tr>
<td>Alphe5</td>
<td>Solaris 8</td>
<td>SUN Ultra2 w/ Dual 200 MHz UltraSparc - sun4u 128 MB RAM</td>
<td>Fast Ethernet</td>
</tr>
<tr>
<td>Alphe6</td>
<td>Solaris 8</td>
<td>SUN Ultra2 w/ Dual 200 MHz UltraSparc - sun4u 128 MB RAM</td>
<td>Fast Ethernet</td>
</tr>
</tbody>
</table>
**Test Setups**

**Single topic with one publisher and one subscriber**

Host 2

Host 3

Host 4

Host 6

Host 6

Host 1

**Single topic with one publisher and 5 subscribers**

Host 2

Host 3

Host 4

Host 6

Host 6

Host 1

**Single topic with 5 publishers and one subscriber**

Host 1

Host 2

Host 3

Host 4

Host 6

Host 6

Configuration Setup
Methodology

- NTP for clock sync
- NDDS 3.0m configuration
  - Reliable queue size 1
- Splice 2
- Vendor tool code generation
  - C libraries

One-way latencies

Time \_\_recv - Time \_\_sent

Time Synchronization

- Max Offset

Slide 12 6/30/2005
Latencies of 200 Byte Message Delivery at 10 Hz
Average Latencies of Different Message Sizes at Different Rates
Maximum Latencies of Different Message Sizes at Different Rates
Maximum Latencies for SPLICE Reliable, NDDS Reliable/FIFO, and UDP at 2000 Hz
Simultaneous NDDS & SPLICE – Average Latencies

Average Latencies of Different Message Sizes at Different Rates with Collocated Hosts
Maximum Latencies of Different Message Sizes at Different Rates with Collocated Hosts
5 Publishers to 1 Subscriber - Average Latencies

Average Latencies of Five Publishers Sending Data To One Subscriber
Maximum Latencies of Five Publishers Sending Data to One Subscriber.
Maximum Latencies of One Publisher Sending Data to Five Subscribers – NDDS Best Effort
Maximum Latencies of One Publisher Sending Data to Five Subscribers – NDDS Reliable/FIFO
Maximum Latencies of One Publisher Sending Data to Five Subscribers – SPLICE Best Effort
Maximum Latencies of Publisher Sending Data to Subscribers – SPLICE Reliable
Middleware Solaris  CPU Usage For Subscribers
Take and read
Publisher CPU Usage

Middleware Solaris CPU Usage For Publishers
Tool overhead included
Middleware Message Overhead

- IPC Message Overhead
- Publish
- Subscribe
- Splice
- NDDS

19,000 Content Messages
Summary

- Overall, both products perform well within the range of performance requirements of typical US Surface Navy Domain combat systems.

- Current indications are that both will provide good performance and scalability as a publish-subscribe middleware for combat system applications.

- Based on the results to date, it appears likely that performance of the products may not be a major differentiator, but other features not examined in this evaluation, such as life-cycle cost and support may influence the selection of one product over the other.
Testing of DDS-compliant products as they become available

Evaluation of marshalling overhead associated with different, domain applicable IDL representations

Evaluation of cross-platform performance issues, including marshalling

Performance with large message sizes

Performance on real-time operating systems such as LynxOS, RT Linux