



*Real-Time Middleware*

## **The Middleware NetMARK-RT**

**A quantitative metric for  
middleware performance with  
emphasis on RT-relevant  
characteristics**

**Edwin de Jong, Ph.D.  
Gerardo Pardo-Castellote, Ph.D.**

# Overview

---

- What is benchmarking? Why?
- Middleware benchmarking
- Proposal for three middleware benchmarks
  - NetMARK-raw: average performance
  - NetMARK-RT: performance predictability
  - NetMARK-scale: performance scalability
- Illustrative results
- Concluding remarks

# Middleware Benchmark

- Benchmark (Webster)
  - “A standard of measurement or evaluation”
- Middleware benchmark:
  - Distributed application executing a workload on the interconnect and returning a result (metric)
- Benchmarks typically measure
  - Throughput (speed; units per second)
  - Latency (duration; seconds)
- Running benchmarks on different platforms allows objective comparison

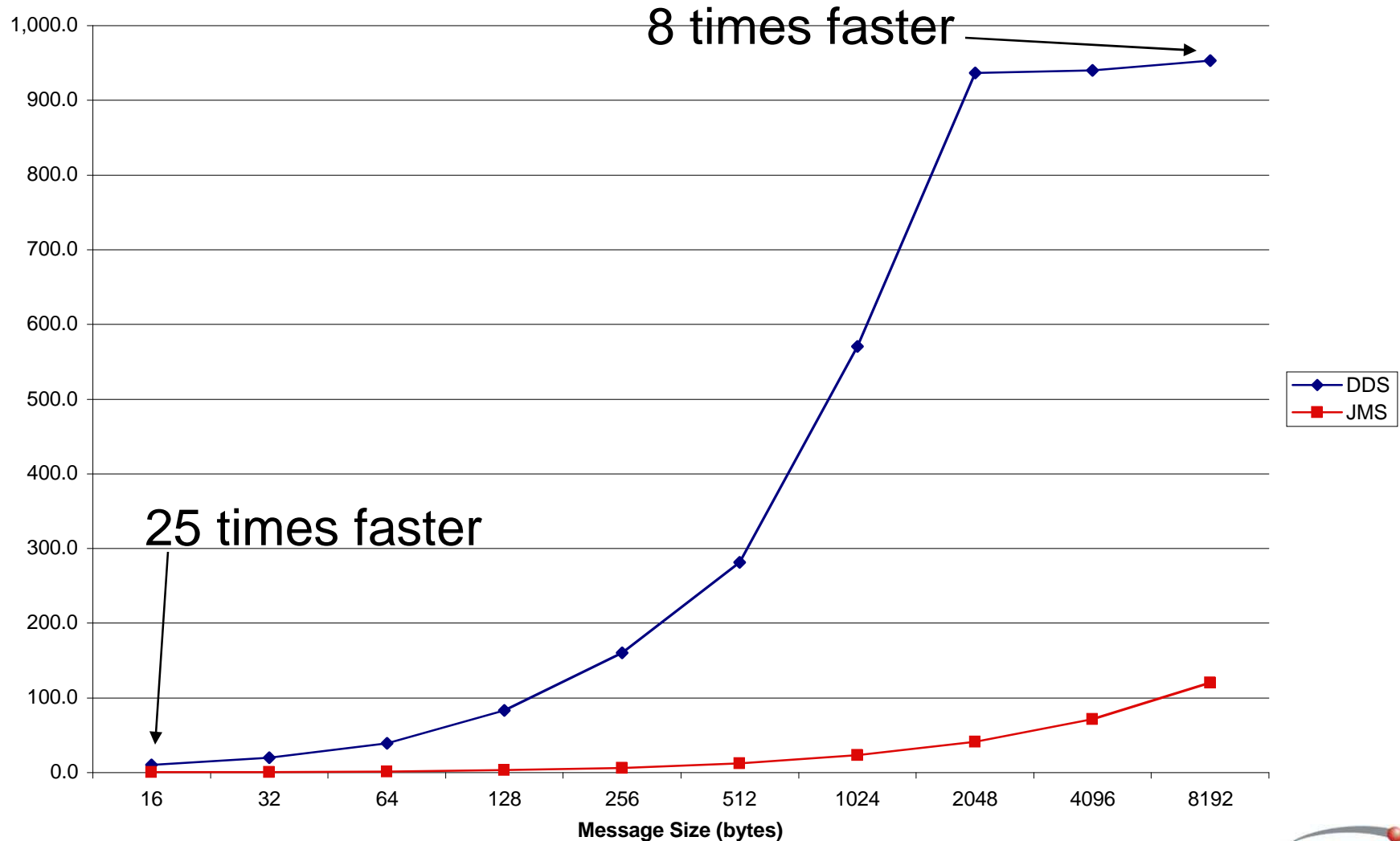
# Why middleware benchmark?

---

- Middleware specifications not helpful
- Less than 1 ms latency...
  - Zero latency?!
- Over 1 million messages per second...
  - What message size?
- As low as 100  $\mu$ s latency...
  - But what on average?
  - How predictable (jitter)?
- And of course many, many more questions:
  - What processor, what network, what compiler, what middleware configuration settings,...?

# Looking at graphs can be misleading

Throughput over Gigabit Ethernet without Message Aggregation



# The Best Performance Comparison

- Your own application with your own workload and metric
- Often not feasible because of
  - Schedule
  - Cost
- Standardized benchmark provides alternative off-the-shelf comparison
  - Should run on your specific platforms and configurations
  - Results may even already have been published
- Important to understand correlation between standardized benchmark and application

# What does a middleware benchmark measure?

---

- Computers'
  - Processors
  - Memory architectures
- Operating systems
- Compilers
- Interconnects
- Middleware implementation
  
- Middleware performance is more than just the middleware implementation!

# Platform parameters affect performance

---

- OS and compiler configuration
- Benchmark settings:
  - Payload size
  - Reliability
  - Persistence
- Middleware parameters:
  - Multicast
  - Reliable protocol configuration
  - DDS has many QoS parameters
  - In addition there are vendor-specific parameters

# Three proposed benchmarks

---

- NetMARK-raw
  - Measures average middleware performance
- NetMARK-RT
  - Measures predictability of middleware performance
- NetMark-scale
  - Measures network scalability of middleware performance

# NetMARK-raw

- Measures raw middleware performance
- Three types of raw tests:
  - Throughput measured as Messages Per Second (MPS)
    - Measure for application performance
  - Throughput measured as Megabit per second (Mb/s)
    - Measure for network utilization
  - Roundtrip (latency) measured as microseconds ( $\mu\text{s}$ )
- Run between two hosts
  - Reliable delivery
  - Most other QoS settings are benchmark parameters

## NetMARK-raw (cont.)

- Benchmark is based on three payload size categories:
  - Small (e.g. between 16Byte and 512 Byte: 16B, 32B, ..., 512B)
  - Medium (e.g. between 1K and 64K: 1K, 2K, ..., 63K)
  - Large (e.g. over 64KB: 64K, 128K, ..., 1MB)
- Benchmark code executes homogeneously distributed mix of payload sizes for each category
- Metric averages results, giving a 3x3 benchmark matrix
  - **Non-weighted (to be discussed)**

	MPS	Kb/s	μs
small	62704	115	92
medium	36820	850	145
large			
SCORE	<b>49762</b>	<b>482</b>	<b>118</b>

Not easily combined!

# NetMARK-raw: Illustrative Results

- Experimental benchmark results for:
  - RTI Data Distribution Service
  - Broker-based MOM
- Results for illustration purposes of proposed benchmarks only
- Benchmark configuration:
  - Two 2.0GHz Ghz Dual AMD Opteron Processor 270 machines
  - 1MB cache; 4GB main memory
  - Interconnect: Gigabit Ethernet
  - OS: Red Hat Enterprise Linux 4.0 (32 bit)
- Full report should disclose all platform details:
  - Processor
  - Memory architecture
  - Operating system
  - Compiler
  - Network infrastructure
  - Middleware product and vendor
  - Benchmark and middleware configuration

# NetMARK-raw: Illustrative Results (cont.)

- Benchmark configuration:
  - Two 2.0GHz Ghz Dual AMD Opteron Processor 270 machines
  - 1MB cache; 4GB main memory
  - Interconnect: Gigabit Ethernet
  - OS: Red Hat Enterprise Linux 4.0 (32 bit)
- Note: medium benchmarks contain messages up to 8KB
- Illustrative results for RTI:
  - NetMARK-raw-MPS-small: 62,704
  - NetMARK-raw-MPS-medium: 36,820
  - NetMARK-raw-latency-small: 69
  - NetMARK-raw-latency-medium: 127
- Illustrative results for broker-based MOM:
  - NetMARK-raw-MPS-small: 3,133
  - NetMARK-raw-MPS-medium: 2,342
  - NetMARK-raw-latency-small: 326
  - NetMARK-raw-latency-medium: 461

# NetMARK-raw: Illustrative Results

	MPS	Kb/s	$\mu$ s
small	62704	115	92
medium	36820	850	145
large			
SCORE	<b>49762</b>	<b>482</b>	<b>118</b>

	MPS	Kb/s	$\mu$ s
small	3133	5.3	326
medium	2341	65	460
large			
SCORE	<b>2737</b>	<b>35</b>	<b>393</b>

DDS

Unit change  
is not  
intuitive

Broker-based  
MOM

# NetMARK-RT

- Measures predictability of middleware performance
- Focuses on latency, measuring:
  - Mean latency (same as in NetMARK-raw)
  - Mean + standard deviation
  - 99 percentile (*near real-time*)
  - 99.99 percentile (*real-time*)
- Metric averages the results, giving a 3x4 benchmark matrix
- Overall Score averages everything

usec	mean	Mean +std	NRT	RT	SCORE
small	92	95	99	161	112
medium	145	148	154	252	174
large					
SCORE	118	121	126	206	143

# NetMARK-RT: Illustrative Results

USEC	<i>mean</i>	<i>Mean +std</i>	<i>NRT</i>	<i>RT</i>	SCORE
<i>small</i>	92	95	99	161	112
<i>medium</i>	145	148	154	252	174
<i>large</i>					
SCORE	<b>118</b>	<b>121</b>	<b>126</b>	<b>206</b>	<b>143</b>

DDS

USEC	<i>mean</i>	<i>Mean +std</i>	<i>NRT</i>	<i>RT</i>	SCORE
<i>small</i>	326	362	544	20939	5542
<i>medium</i>	460	470	678	7379	2247
<i>large</i>					
SCORE	<b>393</b>	<b>416</b>	<b>611</b>	<b>14159</b>	<b>3895</b>

Broker-based  
MOM

# NetMARK-RT: Illustrative Results

- Same benchmark configuration as previously
- Illustrative results for RTI:
  - NetMARK-RT-mean-small: 69
  - NetMARK-RT-mean-medium: 127
  - NetMARK-RT-stdev-small: 8
  - NetMARK-RT-stddev-medium: 2
  - NetMARK-RT-99%-small: 81\*
  - NetMARK-RT-99.99%-small: 86\*
- Illustrative results for broker-based MOM:
  - NetMARK-RT-mean-small: 326
  - NetMARK-RT-mean-medium: 461
  - NetMARK-RT-stdev-small: 36
  - NetMARK-RT-stddev-medium: 15
  - NetMARK-RT-99%-small: 539\*
  - NetMARK-RT-99.99%-small: 19,655\*

\*128-byte messages only

# NetMARK-scale

- Measures network scalability of middleware performance
- How do results scale as the number of subscribed hosts increases?
- Benchmark code measures performance for specific number of subscribed hosts
  - Actual number is benchmark parameter
- Metric is ratio of multi-host and single-host results
  - Ratio of 1 means linear scalability
- Multicast is benchmark parameter; must be reliable

# NetMARK-scale: Sample Results

- Benchmark configuration:
  - Up to 24 1.8Ghz AMD Opteron Processor 244 machines
  - 1MB cache
  - Interconnect: Gigabit Ethernet
  - OS: Red Hat Enterprise Linux 4.0
  - Reliable multicast
- Illustrative results for RTI with 10 subscribed nodes
  - NetMARK-scale-MPS-small: 0.98
  - NetMARK-scale-MPS-medium: 0.98
- Illustrative results for RTI with 24 subscribed nodes
  - NetMARK-scale-MPS-small: 0.93
  - NetMARK-scale-MPS-medium: 0.93

# Concluding Remarks

---

- Proposed middleware benchmarks measure
  - Average performance
  - Performance predictability
  - Performance scalability
- Benchmarks allow objective off-the-shelf comparison of middleware platforms
- Important to understand correlation between benchmarks and application
- Critical to make results simple and easy to interpret
- Still work in progress:
  - Experimental results needed to validate the benchmarks
  - RT Community consensus will be critical