Improving GridFTP Performance Using The Phoebus Session Layer

Radu Nitu
rnu200@student.vu.nl

Feb 2012
Table of Contents

- Phoebus and Dynamic Networks
- Globus XIO Driver for Phoebus
- Phoebus Services
- Experimental Results
- Conclusion
• Phoebus and Dynamic Networks
• Globus XIO Driver for Phoebus
• Phoebus Services
• Experimental Results
• Conclusion
Phoebus and Dynamic Networks 1/2

- Phoebus - infrastructure for improving end-to-end throughput in high-bandwidth, long-distance networks

- Phoebus utilizes a “session layer” protocol and “gateways” in the network.

- Phoebus can fit in hybrid network environments

- Dynamic Networks - DCNs

![Internet2](image1.png) ![ESnet](image2.png) ![GÉANT](image3.png)
• Adapt the data transfer at application runtime, based on available network resources and conditions.

• Provides interface to the common DCN infrastructure.
• Phoebus and Dynamic Networks
• **Globus XIO Driver for Phoebus**
• Phoebus Services
• Experimental Results
• Conclusion
• Grid FTP - data movement application in high-performance and Grid computing from the Globus project

• Globus XIO is an extensible input/output library within the Globus Toolkit

• Phoebus Transport driver is based on XIO TCP driver distributed in the Globus Toolkit.
• Phoebus and Dynamic Networks
• Globus XIO Driver for Phoebus
• **Phoebus Services**
• Experimental Results
• Conclusion
• Phoebus model - end-to-end connection, articulated via a series of Transport protocol adapting Session gateways

• Data transfer crosses over multiple links in an end-to-end connection which are divided into a series of transport layer connection dynamically adapted to the characteristics of each link.

• Phoebus Gateways (PGs) are deployed inside the networks, performing protocol adaptation and translation.
• Basic TCP Adaptation – change the settings used in the TCP connection
  – Buffer Sizes
  – Use of Congestion Control Algorithms

• Adaptation to Non-TCP Protocols
  – User-space implementation protocols and kernel – space implementation
  – Take form of a library and are based on UDP
  – Congestion control and the correct and in-order reception is handled on the far side of the connection
• Protocol Abstraction Layer
  – Functions to return new connections
  – Functions to return the connection objects themselves

• UDT Adaptation
  – Protocol used for the abstraction layer
  – Designed for transfers over wide-area, high speed networks
  – Phoebus UDT implementation emulates the TCP shutdown function with a session-layer framing.
• Phoebus and Dynamic Networks
• Globus XIO Driver for Phoebus
• Phoebus Services
• Experimental Results
• Conclusion
• Testbed
  – Phoebus infrastructure
  – GridFTP
  – Variety of network conditions using netem (Linux module)

• Setup
  – 7 machines
  – 2 end hosts
  – 2 PGs
  – 3 dedicated hosts as netem fw nodes
Experimental Configuration

- A set of conditions expected on current real world networks.
- LAN packet loss rates of %0.001, %0.001, %0.1
- WAN latencies of 25ms, 50ms, 100ms, 150ms
- 4ms of latency for each LAN segment
Experimental results 3/4

• CPU Utilization
  – high CPU loads for using UDT
  – CPU overhead moved from the edge node to PGs

<table>
<thead>
<tr>
<th>Direct UDT</th>
<th>Phoebus w/ UDT</th>
<th>Direct UDT</th>
<th>Phoebus w/ UDT</th>
</tr>
</thead>
<tbody>
<tr>
<td>no loss</td>
<td></td>
<td>no loss</td>
<td></td>
</tr>
<tr>
<td>62.1%</td>
<td>15%</td>
<td>38.4%</td>
<td>15.7%</td>
</tr>
</tbody>
</table>

• Performance over bottleneck links
Experimental results 4/4

- **Connect Latencies**

<table>
<thead>
<tr>
<th>Type</th>
<th>Latency</th>
<th>Connect Mean (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct</td>
<td>None</td>
<td>0.46</td>
</tr>
<tr>
<td>Phoebus-TCP</td>
<td>None</td>
<td>2.33</td>
</tr>
<tr>
<td>Phoebus-UDT</td>
<td>None</td>
<td>8.03</td>
</tr>
<tr>
<td>Direct</td>
<td>Moderate</td>
<td>64.02</td>
</tr>
<tr>
<td>Phoebus-TCP</td>
<td>Moderate</td>
<td>74.41</td>
</tr>
<tr>
<td>Phoebus-UDT</td>
<td>Moderate</td>
<td>143.33</td>
</tr>
<tr>
<td>Direct</td>
<td>High</td>
<td>114.38</td>
</tr>
<tr>
<td>Phoebus-TCP</td>
<td>High</td>
<td>125.27</td>
</tr>
<tr>
<td>Phoebus-UDT</td>
<td>High</td>
<td>245.32</td>
</tr>
</tbody>
</table>

- **Transfer Latencies**

<table>
<thead>
<tr>
<th>Type</th>
<th>Latency</th>
<th>Transfer Mean (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct</td>
<td>None</td>
<td>0.38</td>
</tr>
<tr>
<td>Phoebus-TCP</td>
<td>None</td>
<td>0.45</td>
</tr>
<tr>
<td>Phoebus-UDT</td>
<td>None</td>
<td>0.6</td>
</tr>
<tr>
<td>Direct</td>
<td>Moderate</td>
<td>64.04</td>
</tr>
<tr>
<td>Phoebus-TCP</td>
<td>Moderate</td>
<td>64.07</td>
</tr>
<tr>
<td>Phoebus-UDT</td>
<td>Moderate</td>
<td>64.04</td>
</tr>
<tr>
<td>Direct</td>
<td>High</td>
<td>114.6</td>
</tr>
<tr>
<td>Phoebus-TCP</td>
<td>High</td>
<td>112.16</td>
</tr>
<tr>
<td>Phoebus-UDT</td>
<td>High</td>
<td>112</td>
</tr>
</tbody>
</table>
• Phoebus and Dynamic Networks
• Globus XIO Driver for Phoebus
• Phoebus Services
• Experimental Results

• Conclusion
• Controlled experiments:
  – Phoebus improves throughput and provides improved single-stream application performance
  – Integration with GridFTP brings improved performance with no tuning
  – Phoebus Gateways can more effectively utilize bottleneck links

• Phoebus is under experimentation in Internet2’s network
Thank you