Transparent Optimization of Grid Server Selection with Real-Time Passive Network Measurements

Marcia Zangrilli and Bruce Lowekamp

Overview

Grid Services

- Grid resources modeled as services
- Define interface for interacting with services
  - Defines contract between service and client
  - Locate service on remote machines
  - Access to remote services via HTTP

Benefits of Grid Services

- Reduce programming complexities
- Create opportunities to help steer the decision of which service is best to invoke

Grid Services abstraction removes control from client and reduces client's ability to optimize its performance by monitoring the system

Research Problem

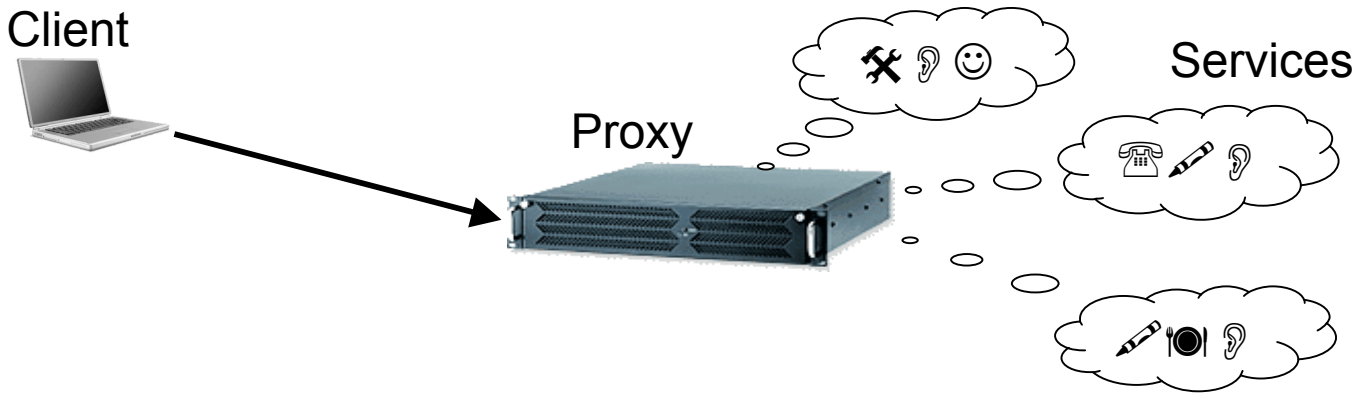
How can we provide transparent optimization services to grid services applications?

- No modification to or awareness from client application or service provider

Specifically, how can we incorporate network performance monitoring to better utilize network resources?

- Provide accurate measurements with minimal intrusiveness on other applications and network

Monitoring Grid Services Proxy



- Grid services proxy
  - Implements service interface
  - New level of abstraction that hides data or service location from client, while providing different services and datasets to the client and users
    - Service can change implementation with no affect on client
- Monitor Network Performance
  - Integrate Wren monitoring toolkit into Grid Services Proxy
  - Use network available bandwidth measurements to help proxy choose service replica for client applications

Why integrate Wren with Grid Services Proxy?

- Provide middleware with sufficient power to transparently make performance optimizations on behalf of the client
- No need to involve user or client application (unless qualitative decisions need to be made)
- Wren provides network performance information without actively probing the network.

Real-time measurements can be used by proxy to make runtime server selection decisions without modification to client application or service provider code/infrastructure.

Outline

- Wren monitoring system
  - Online Wren overview
  - Overhead of Wren Measurements
- Grid Services Proxy and Wren Integration
  - Accuracy and lag of Wren Measurements
- Grid Server Selection Problem
- Conclusions

Wren: Watching Resources from the Edge of the Network

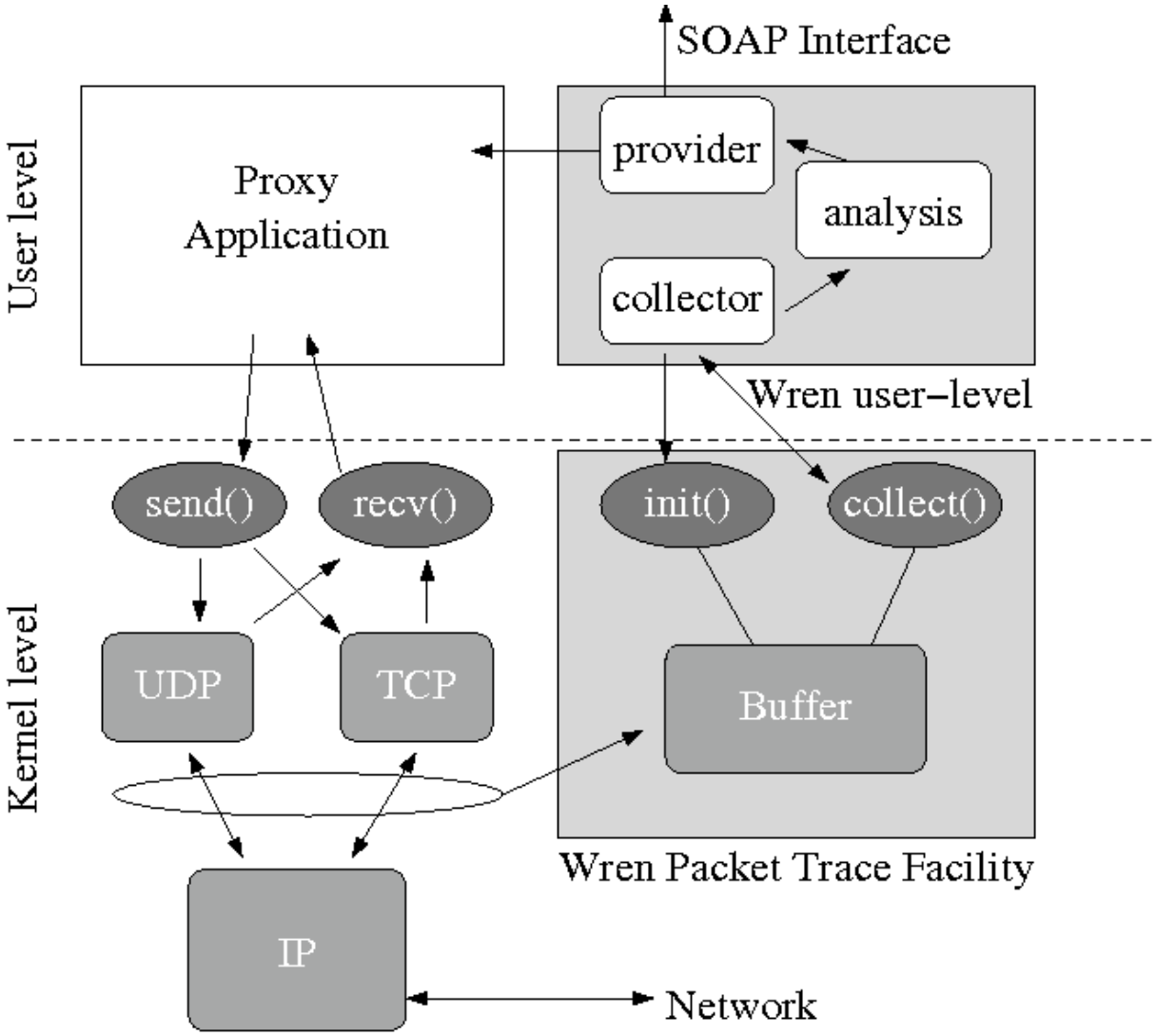
Use passive traces of application traffic to measure available bandwidth.

- No traffic injected, no burden on network!

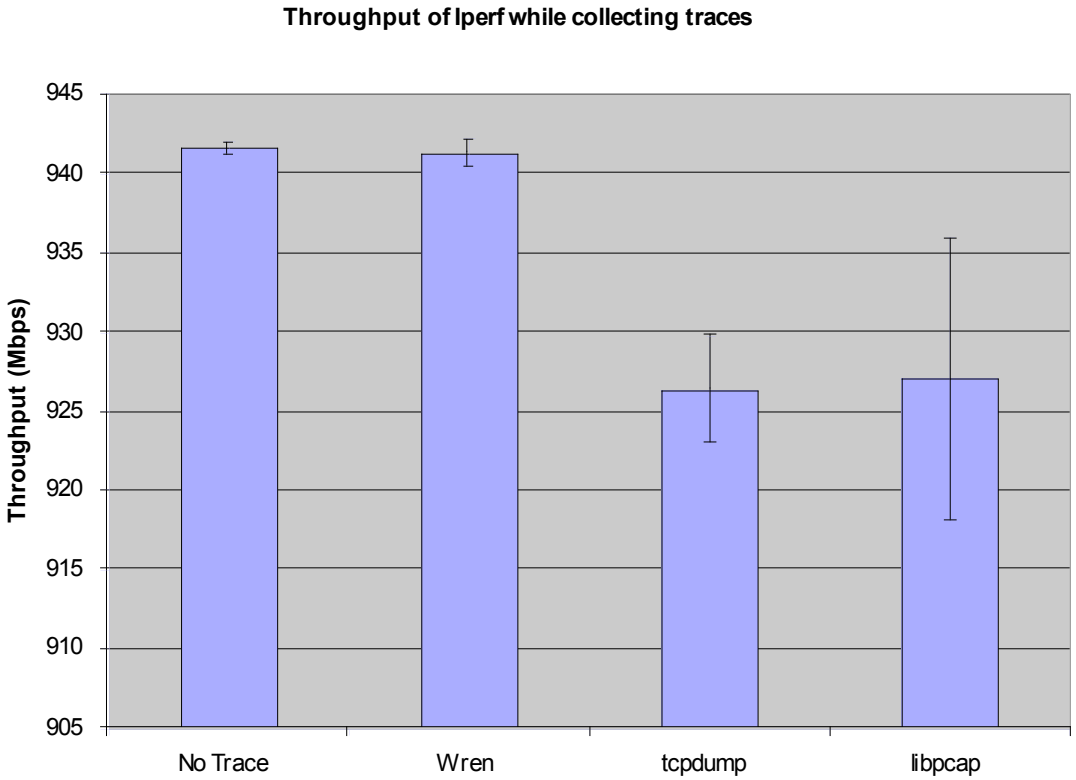
Wren Design:

- Kernel-level instrumentation to collect traces of application traffic.
- Analysis and management of traces handled in user-level.

Wren Architecture



Wren Tracing Efficiency



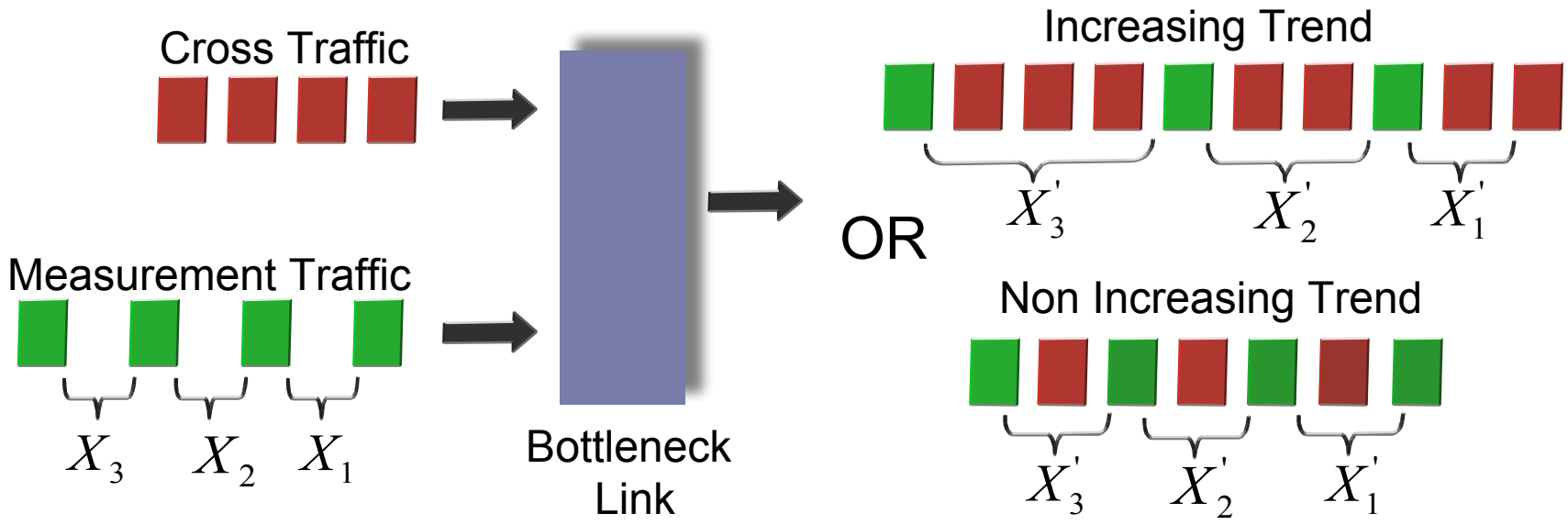
Tracing between 2.8GHz Hyperthreaded P4s running Linux 2.4.29 with 1GB of RAM and an Intel CSA gigabit Ethernet NIC. Nodes connected using Cisco 3750 gigabit switch.

Wren tracing does not reduce application throughput and is the lowest overhead of the trace collection tools.

Wren User-Level

- Collect Traces from kernel
- Apply SIC analysis to traces
- Provide Measurements
  - Rather than one measurement, provide series of available bandwidth lower bounds
  - Interested applications can use measurements to adapt runtime strategies
  - Measurements can be stored in central repository for later use.

Self-Induced Congestion (SIC) Analysis



Increasing Trend indicates queuing in the path. $X'_3 > X'_2 > X'_1$

Queuing occurs when traffic is sent at rate greater than available bandwidth

Wren applies SIC to passive traces of existing TCP traffic.

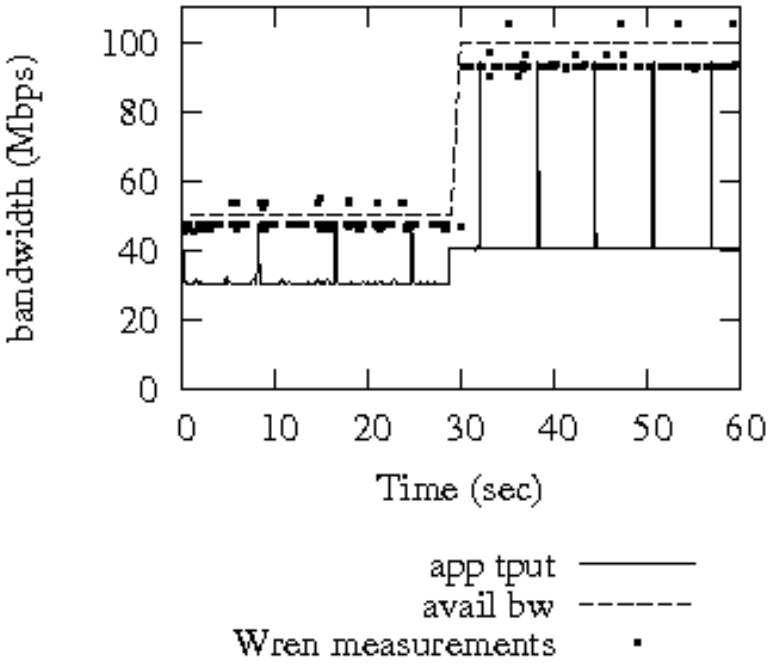
Wren Proxy Implementation

- HTTP-level proxy service
  - Parses message and selects server based on available bandwidth information associated with requested service
  - Does not modify message
- Wren component
  - Provides series of lower bound available bandwidth measurements for each connection
  - Proxy stores available bandwidth information for each service registered with the proxy

Wren Measurements of Proxy Traffic

- Focus on LAN testbed results
  - Capacity on testbed is 100Mbps
  - Can control amount of congestion / available bandwidth on path
  - WAN results are difficult to validate
- Monitored Traffic and Measure available bandwidth for 60 seconds
- Monitored Application
  - Traffic generator that sends TCP traffic through proxy with average throughput of 45Mbps on uncongested LAN
- Cross Traffic
  - Send varying amounts of uniform (constant bit rate, CBR) cross traffic
  - 50 Mbps of cross traffic for first 30 seconds, and 0 for remainder of experiment
  - Available bandwidth on path is 100 Mbps – avg cross traffic rate

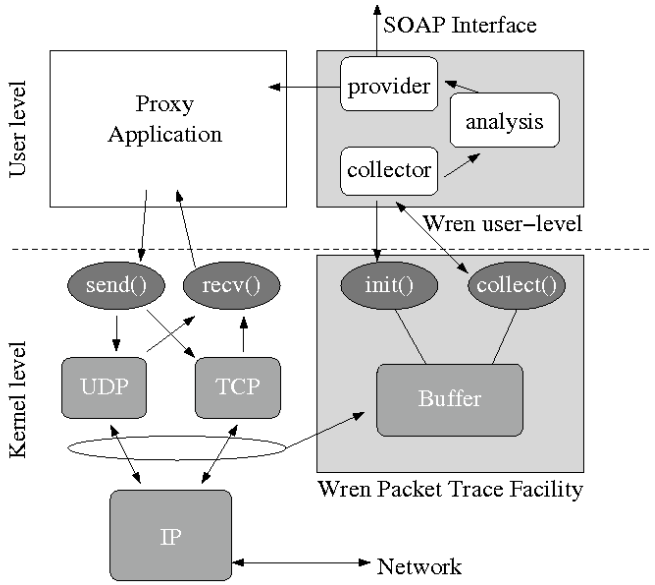
Wren Measurements of Proxy Traffic Results



Wren produces accurate available bandwidth measurements that reflect the changes in available bandwidth even when the throughput of the application is not saturating the path

Wren Measurement Lag

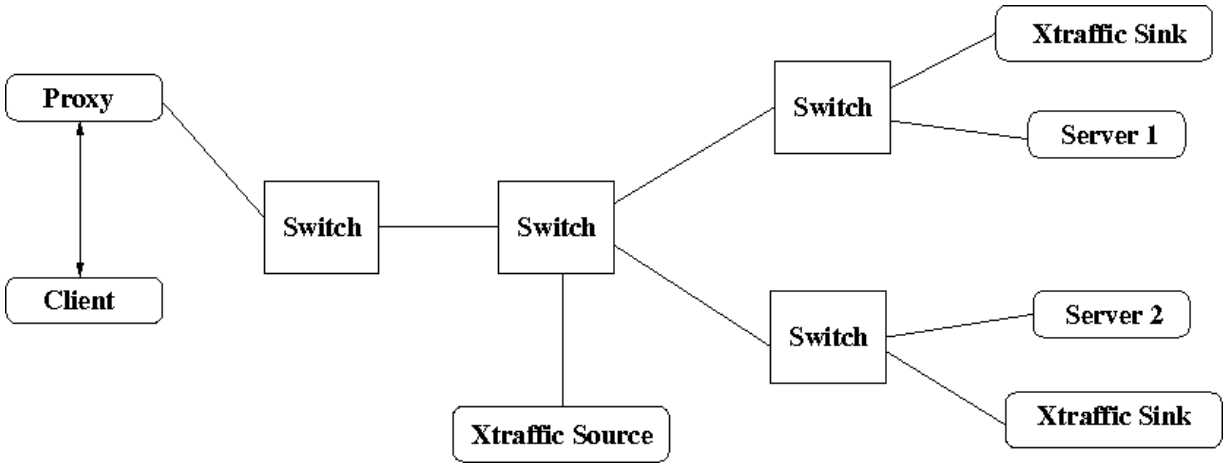
Collector Thread processing time (excluding polling time)	Analysis Thread processing time
0.02852 ± 0.00029 s	0.21741 ± 0.00216 s



0.25 second processing lag acceptable for proxy relaying a lot of data or application nodes that don't saturate link constantly.

Server Selection Experiment

- Available bandwidth only one factor in server selection problems
  - This experiment ignores other factors such as CPU power or cost
  - In real server selection process, Wren measurements would be used in conjunction with other factors



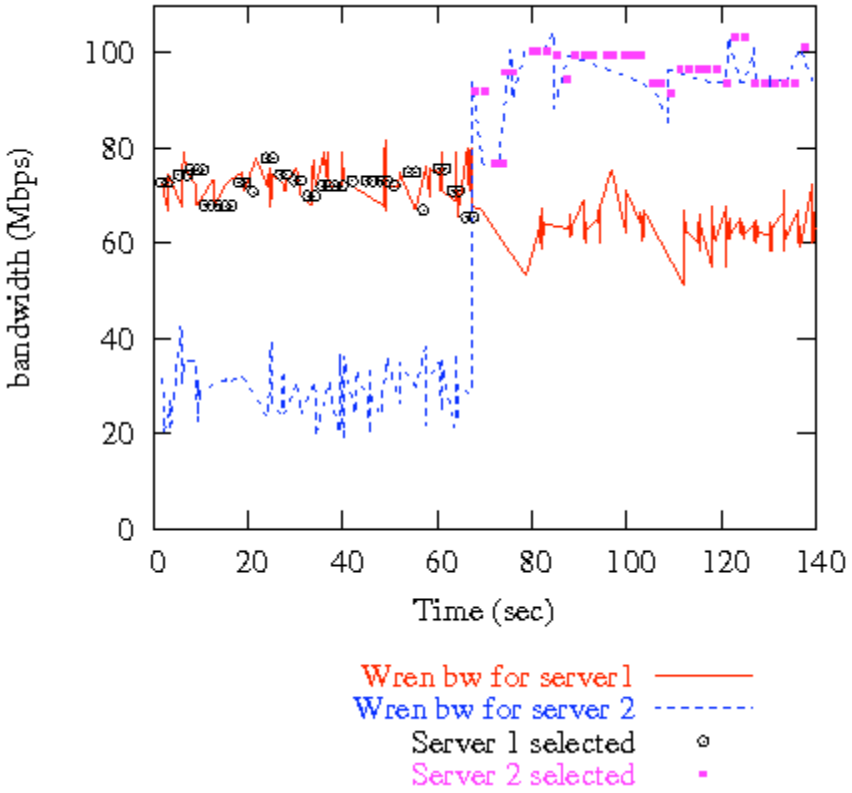
Server Selection Experiment Details

Cross Traffic

- 25 Mbps of CBR traffic between proxy and server1
- 75 Mbps of CBR traffic between proxy and server2 for first 70 seconds, 0 cross traffic after 70 seconds

- Application-Generated Traffic
  - Traffic generator that sends at average rate of 40Mbps on uncongested LAN
- Proxy server selection
  - Pick server that has highest bandwidth on path between server and proxy

Proxy Server Selection Results



- Wren measurements used by proxy to correctly select the server with the highest available bandwidth path between proxy and server
  - No modification to client code or server code/infrastructure
  - Detect changes in available bandwidth and change server appropriately

Contributions

- Online Wren monitoring system
  - System that provides real-time measurements to application
  - Quantification of overhead associated with Wren packet tracing
  - Comparison of Wren packet tracing to tcpdump and libpcap
- Application of passive monitoring to server select proxy that functions in a grid services environment without modifying any other component
  - Accuracy of Wren measurements using grid services requests
  - Quantification of lag associated with providing real-time measurements
  - Demonstration that Wren measurements can aid the grid services proxy in server selection

Thank you

For more information:

- Email: mazang@cs.wm.edu or
- Visit http://www.cs.wm.edu/~lowekamp/wren.html