LEGS: A WSRF Service to Estimate Latency between Arbitrary Hosts on the Internet

R Vijayprasanth[1], R Kavithaa [2,3], and Rajkumar Kettimuthu[2,3]
[1]Department of Information Technology
Coimbatore Institute of Technology, Coimbatore, India 641014
[2]Computation Institute, The University of Chicago, Chicago, IL USA 60637
[3]Math and Computer Science Division
Argonne National Laboratory, Argonne, IL USA 60439

Abstract:
The ability to estimate the communication latency between arbitrary hosts on the Internet would enable the distributed computing applications to find the closest server or to find the closest replica of the data required for the computation. Internet content providers, who often place data and server mirrors throughout the Internet to improve access latency for clients, can use this as one of the metrics to find the appropriate mirror for the clients. It could be useful for constructing topologically sensitive overlay networks. In this paper, we present Latency Estimation Grid Service (LEGS), a Web Services Resource Framework (WSRF) service that is built on top of a tool called King to estimate latency between arbitrary hosts using recursive DNS queries.

1. Introduction:

With the advent of Grid computing to harvest the enormous potential of the Internet infrastructure, a new class of large-scale globally-distributed applications such as climate modeling[1], molecular modeling for drug design, earthquake simulation[2], ATLAS, CMS[3], ALICE[4], Sloan Digital Sky Survey[5], CMCS[6], LTER[7], SEEK[8] have emerged. Because these systems have a lot of flexibility in choosing their resources for computation and the sources/sinks for the input/output data, they can greatly benefit from the available bandwidth and latency between the various pairs of participating nodes. In this paper, we focus on providing the latency information through a WSRF[9] service. Such a service would also benefit distributed content hosting services, overlay network multicast[10][11], content addressable overlay networks[12][13], and peer-to-peer file sharing, as these services and applications have a lot of flexibility in choosing their communication paths.

This paper is organized as follows. Section 2 provides some background on tools to evaluate the network latency between arbitrary end hosts on the Internet. Section 3 gives some background on WSRF. Section 4 describes LEGS in detail. Section 5 contains experimental results and Section 6 provides the summary.

2. Latency Estimation Tools:

Tools like Ping[14], Synack[15] can be used to estimate the end-to-end delay but these tools need to be run on one of the two nodes between which the latency needs to be estimated. Researchers have developed some tools to calculate the round-trip time (RTT) between arbitrary hosts on the Internet, at least to certain extent. We will provide a brief overview of three such popular tools namely King[16], IDMaps[17] and GNP[18].

2.1 King:

King[16] estimates RTT between any two hosts in the Internet by estimating the RTT between their domain name servers. In order to do this, King depends on the following fact, which hold true for most of the domain name servers in the current Internet. Most domain name servers in the current Internet (~75%-80%) support recursive queries from any host in the Internet. The accuracy of King depends crucially on another fact about the domain name servers in the Internet. In many cases the name servers are located close (measured as network latency) to their hosts. While the first fact is the result of a default choice by many name server administrators the second fact arises more out of administrative convenience than anything else.

2.2 IDMaps:

IDMaps[17] is an infrastructural service in which special HOPS servers maintain a virtual topology map of the Internet consisting of end hosts and special hosts called Tracers. The distance between hosts 'A' and 'B' is estimated as the distance between 'A' and its nearest Tracer 'T1', plus the distance between 'B' and its nearest Tracer 'T2', plus the shortest path distance from 'T1' to 'T2' over the Tracer virtual topology. As the number of Tracers grows, the prediction accuracy of IDMaps tends to improve. Designed as a client-server architecture solution, end hosts can query HOPS servers to obtain network distance predictions. An experimental IDMaps system has been deployed.

2.3 GNP:

GNP[18] is a relatively new technique to estimate the latency between end hosts. A small set of dedicated hosts, called landmarks, first selects their own coordinates in a chosen geometric space. These coordinates are then disseminated to any host that wants to compute it's own coordinates. Hosts measure the latency between themselves and the landmarks, and from this extrapolate their own coordinates. Given the coordinates of two end hosts, one can then compute the latency between them as a function of the distance between their coordinates. An evaluation over a small number of hosts has demonstrated that GNP is often more accurate than IDMaps. GNP and IDMaps are similar to each other in that they measure latencies along a subset of Internet paths, and extrapolate these measurements to estimate the latency of an arbitrary Internet path. The primary architectural difference between them is that GNP shifts the complexity of computing distances between landmarks and end hosts from the landmark servers to the end hosts themselves.

3. WSRF:

A web service is an Internet hosted application that is described via Web Services Description Language (WSDL)[19] and is capable via standard network protocols such as but not limited to SOAP[20] over HTTP[21]. Web services must often provide their users with the ability to access and manipulate state, i.e., data values that persist across, and evolve as a result of, Web service interactions. Grid service [22] is a Web service that conforms to a set of conventions for such purposes as service lifetime management, inspection, and notification of service state changes. Grid services provide for the controlled management of the distributed and often long-lived state that is commonly required in distributed applications. Global Grid Forum's (now it is called

Open Grid Forum) Open Grid Services Infrastructure (OGSI) [23] working group defined a Minimum set of standards and behaviors with which Grid Services must comply. OGSI is concerned primarily with creating, addressing, inspecting, and managing the lifetime of stateful Grid services [22].

The WS-Resource Framework is inspired by the work of the Global Grid Forum's Open Grid Services Infrastructure (OGSI) Working Group. WSRF appropriately factors the OGSI interfaces to produce a framework of independently useful Web service standards. The Web Services Resource Framework (WSRF) defines conventions for managing state so that applications discover, inspect, and interact with stateful resources in standard and interoperable ways. It does so within the context of established Web services standards. It provides specifications to make grid resources accessible within the web services architecture. In addition, the specifications partition OGSI functionality into distinct functionality that allows flexible composition in a mix-and-match manner. The factoring, composition capability and greater reliance on broadly accepted Web service concepts provide a simpler, more familiar and incremental path for developers wishing to exploit OGSI functionality

4. LEGS:

LEGS (Latency Estimation Grid Service) is a Grid service implemented on top of the Globus toolkit [24]. The Globus Toolkit is a software toolkit, developed by The Globus Alliance [25], which can be used to program grid-based applications. The toolkit includes quite a few high-level services that we can use to build Grid applications. The Globus Toolkit 4 includes a complete implementation of the WSRF specification. By implementing it on top of the Globus toolkit, we use the built-in security and the monitoring and discovery services provided by Globus. LEGS exposes an operation called latency, which takes the two fully qualified domain names or IP addresses as arguments and returns the network latency between them. Relevant portions of the WSDL is given below:

```
<operation name="latency">
  <input message="tns:LatencyInput"/>
  <output message="tns:LatencyOut"/>
</operation>
<message name="LatencyInput">
  <part name="parameters"
          element="tns:latencyRequest"/>
</message>
<message name="LatencyOut">
  <part name="parameters"
    element="tns:latencyResponse"/>
</message>
<xsd:element name="latencyRequest">
   <xsd:complexType>
   <xsd:sequence>
     <xsd:element name="a1"
                  type="xsd:string"/>
     <xsd:element name="a2"
                  type="xsd:string"/>
   </xsd:sequence>
   </xsd:complexType>
</xsd:element>
<xsd:element name="latencyResponse"
                  type="xsd:int"/>
```

LEGS uses King to find the network latency between the given hosts. Even though IDMaps and GNP are better than King in terms of Speed of estimation and the estimation overhead, we use King because of its ease of use. King is simpler to use than IDMaps, in that it leverages the existing DNS and does not require the deployment of any additional infrastructure. While GNP requires lesser infrastructure, GNP does require the cooperation of end hosts to estimate and share their coordinates relative to an agreed- upon set of landmarks. Further King is shown to have a better accuracy than IDMaps[16] and consistency identical to that of traceroute measurements made using public traceroute servers accessible from www.traceroute.org. Resources within WSRF services are used to maintain state across service calls through message exchanges between a client and a Web service. LEGS uses a resource to store the result of a latency operation. Thus, LEGS uses King to determine the latency between two end points only once. For subsequent requests to find the latency between the same pair of endpoints, LEGS just pulls the value stored in its resource. As the API provided by King requires IP addresses as input, LEGS take care of converting the hostname to IP address, if need be. LEGS uses IP addresses of the endpoints while storing the latency results in its resource. If the client request has hostnames, LEGS converts the hostnames into IP addresses before looking it up on the resource.

5. Experimental Results:

Src	Dest	RTT	L1	L2
ANL	OSU	10ms	5.2s	4.2s
ANL	USC	60ms	5.7s	4.2s
Maine	SDSC	105ms	6.0s	4.2s

Table 1: Time taken by LEGS o compute the RTT between end hosts

A useful general service for the Internet should enable a host to quickly and efficiently learn the distance between any two hosts. To be widely useful, such a service should provide an answer with a delay overhead less than the speed-up gained using the service. Table 1 shows the time taken by LEGS to find the latency for the endpoints that have Round Trip Times (RTTs) of 10ms, 60ms and 105ms. We developed a simple client that takes in two hostnames/IP addresses as arguments and talks to the LEGS service to get the latency between the hosts. We measured the total time taken by the client to send in the request and get back the result. Column L1 in Table 1 refers to the time taken to get the latency between a pair of hosts from LEGS for the very first time. Column L2 refers to the time taken to get the latency between the same pair of hosts from LEGS for the subsequent requests.

When LEGS receives a request to find latency between a pair of hosts, it converts the hostnames to IP addresses (if need be) and looks for the IP address pair in its resource. If its finds the IP address pair in its resource, it returns the latency value stored in the resource. Otherwise it uses King to estimate the latency between the IP address pair, stores it in its resource and returns it to the client. Column L2 in Table 1 represents the time taken to get the latency value stored in LEGS resource. That is the reason why the values in column L2 are same and less than that of the values L1. As the LEGS service and client run on the same node for these experiments and there is no other on the host, the values in column L2 are same. Otherwise, there may be little variations in those values.

We also measured the time taken by a simple Math Service running on top of the Globus Toolkit to add a value to the value stored in resource and return back the result. It took 4.3 seconds. This implies that LEGS does not introduce any additional delays. We believe that majority of this time is used in security message exchanges and SOAP processing.

6. Summary:

We developed a Grid service called LEGS to find the latency between two arbitrary hosts on the Internet. It would be useful for a lot of applications and high-level web and Grid services. Implementing it as a Grid service allows us the store the results and optimize the performance for subsequent requests. We also presented experimental results that show the time taken by LEGS to determine the latency between various pairs of end hosts.

Acknowledgment:

This work was supported in part by the Mathematical, Information, and Computational Sciences Division subprogram of the Office of Advanced Scientific Computing Research, Office of Science, U.S. Dept. of Energy, under Contract DE-AC02-06CH11357 and in part by National Science Foundation under contract OCI-0534113.

References:

[1] The Earth System Grid: Supporting the Next Generation of Climate Modeling Research. D. Bernholdt,et. al.. Proceedings of the IEEE, 93:3, March, 2005, 485-495.

[2] NEESgrid: A Distributed Collaboratory for Advanced Earthquake Engineering Experiment and Simulation. B. Spencer Jr. et. al, 13th World Conference on Earthquake Engineering, August 2004.

[3] http://uscms.fnal.gov/

[4] http://aliceinfo.cern.ch/index.html

[5] http://www.sdss.org/

[6] C Pancerella et. al. Metadata in the Collaboratory for Multi-Scale Chemical Science. In 2003 Dublin Core Conference: Supporting Communities of Discourse and Practice-Metadata Research and Applications, Seatle, WA, 28 September - 2 October 2003.

[7] http://www.lternet.edu/

[8] http://seek.ecoinformatics.org/

[9] http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsrf

[10] Y. Chu, S. Rao, andH. Zhang, "Acasefor endsystemmulticast," in Proceedings of ACM Sigmetrics, June 2000.

[11] J. Liebeherr, M. Nahas, andW. Si, "Application-layer multicast with delaunay triangulations," Tech. Rep., University of Virginia, Nov.2001.

[12] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, "A scalable content-addressable network," in Proceedings of ACM SIGCOMM'01, San Diego, CA, Aug.2001.

[13] I. Stoica, R. Morris, D. Karger, F. Kaashoek, and H. Balakrishnan, "Chord: A scalable peer-to-peer lookup service for Internet applications," in Proceedings of ACM SIGCOMM'01, San Diego, CA, Aug. 2001.

[14] http://www.ping127001.com/pingpage.htm

[15] http://www-iepm.slac.stanford.edu/tools/synack/

[16] K. Gummadi, S. Saroiu, S. Gribble, King estimating Latency between Arbitrary Internet End Hosts, In Proceedings of the SIGCOMM Internet Measurement Workshop (IMW 2002).

[17] P. Francis, S. Jamin, C. Jin, Y. Jin, D. Raz, Y.Shavitt, and L. Zhang. IDMAPS: a global Internet host distance estimation service. In IEEE/ACM Trans. on Networking, October 2001.

[18] Eugene Ng and Hui Zhang. Predicting Internet network distance with coordinates-based approaches. In Proceedings of IEEE INFOCOM 2002, NewYork, NY, June 2002.

[19] http://www.w3.org/TR/wsdl

[20] http://www.w3.org/TR/soap/

[21] http://www.w3.org/Protocols/rfc2616/rfc2616.html

[22] The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration. I. Foster, C. Kesselman, J. Nick, S. Tuecke, Open Grid Service Infrastructure WG, Global Grid Forum, June 22, 2002.

[23] http://xml.coverpages.org/OGSI-SpecificationV110.pdf

[24] http://www.globus.org/toolkit/

[25] http://www.globus.org/alliance/