# ARеNA: Adaptive Distributed Catalog Infrastructure Based On Relevance Networks


Vladimir Zadorozhny
University of Pittsburgh
Pittsburgh, PA
vladimir@sis.pitt.edu

Avigdor Gal
Technion - IIT
Haifa, Israel
avigal@ie.technion.ac.il

Louiqa Raschid
University of Maryland
College Park, MD
louiqa@umiacs.umd.edu

Qiang Ye
University of Pittsburgh
Pittsburgh, PA
qye@sis.pitt.edu


## 1 Introduction

Wide area applications (WAAs) utilize a WAN infrastructure (*e.g.*, the Internet) to connect a federation of hundreds of servers with tens of thousands of clients. Earlier generations of WAA relied on Web accessible sources and the `http` protocol for data delivery. Recent developments such as the PlanetLab [8] testbed is now demonstrating an emerging class of data- and compute- intensive wide area applications.

One of the many challenging characteristics of WAA [14] involves the availability of several alternative data sources for a given query. This may be due to mirror sites, or more commonly, the overlap of content in specific application domains. For example, in scientific environments users may choose to access raw datasets, or they may be able to re-use the cached results of prior queries. Choosing among alternative data sources depends on access costs, coverage, rate of data delivery, outpt quality, *etc.* For example, a Performance Target (PT) [13] query optimizer may optimistically allow the selection of potentially noisy sources whose latencies range from excellent to poor since there is some expectation of meeting a performance target when there is no delay. A conservative strategy would avoid such a source since there was some expectation of missing the target.

Access costs are know to exhibit *transient* behavior. The unpredictable behavior of a dynamic WAN [9, 11] results in a wide variability in access latency (end-to-end delay). Variation in access latency may depend on many factors including the topology of the client and the remote server, the network and server workloads, which are often affected by the time of day and day of week, and points of congestion between clients and servers [9]. While there are many accurate models for predicting latencies based on network topology and characteristics, *e.g., Internet distance* and *points of congestion* [3, 7, 9, 11], these models were typically designed to test network behavior. They were not designed to support scalable mediators in the presence of a large number of clients and servers. To this end, there is a need for comprehensive information on replica sites, coverage at these sites, and access latencies. Maintaining such information Should be scalable to the large number of clients and servers for WAA. Therefore, we propose a distributed catalog and in this demonstration, we discuss collecting and managing latency information with *AReNA*. *AReNA* is complementary to the BibFinder project [6] which learns statistics on source coverage information for the bibliographic domain, but does not provide cost or delay information. We expect to integrate these two catalogs in the future.

*AReNA* obtains latency information through passive performance monitoring. Individual *Latency Profiles (iLPs)* represent time-dependent latency distributions between individual clients and servers. *AReNA* aggregates performance information into aggregate *Latency Profiles (aLPs)* in order to make latency predictions scalable for groups of clients and servers. We use measures such as *mutual information* and *correlation* to find similarity relationships among iLPs. Once similarity relations have been established, we allow either automatic or manual constructions of aLPs. To allow efficient manual intervention, *AReNA* offers a unique feature of dynamically analyzing and visualizing meaningful relationships using *Relevance Networks* (RN) [1]. We adopt RNs as a management tool, to manage large numbers of iLPs, and to allow the tuning of size and relationship strength in grouping iLPs, so as to make predictions and to provide scalable maintenance.




**Proceedings of the 31st VLDB Conference, Trondheim, Norway, 2005**


## 2 Architecture: iLPs and aLPs

Figure 1 presents the *AReNA* (Adaptive Relevance Network Architecture) environment. There are three types of nodes, namely clients, content servers, and performance monitors (PMs). Clients continuously download data from content servers and passively construct individual iLPs. Given a

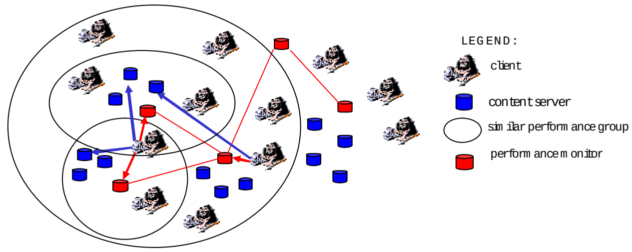


Figure 1: *AReNA* environment

client $c$ and a server $s$, an $iLP$ characterizes the end-to-end delay for a request from server $s$. Due to the stochastic nature of the network, $iLP$ is a random variable represented as a latency probability distribution. An $iLP$ is a time varying distribution that can still show some regularity, such as a repetitive latency pattern, where similar latencies may be observed at the same time of day and day of week. For example, consider Figure 2, illustrating the result of passive latency information collection, grouped by day of week (Day) and time of Day (Time). Based on this figure, it seems that the iLP should be modeled to have time or day significance. In our experiments we observed iLPs with varying levels of noise and time and day significance.

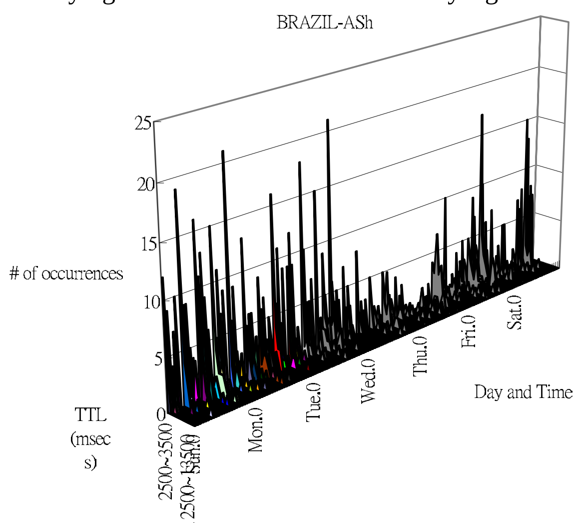


Figure 2: Latency distribution grouped by Day and Time

PMs manage large collections of iLPs; this is done by aggregating iLPs into a smaller number of $aLP$s. These aggregated profiles combine a set of $n$ $iLP$s. PMs then manage aLPs and their associated iLPs. PMs form an overlay network similar in spirit to the control and measurement overlays of the MCoop architecture [10]. Clients consult PMs to obtain a prediction. The scope of an aLP is depicted in Figure 1 by ellipses, where each ellipse contains clients and servers for which an aLP could be constructed. It is worth noting that some aLPs overlap. This illustrates a situation where a client may belong to one aLP for some servers and to another aLP for a different set of servers. The number and placement of PMs should maximize scalability of performance monitoring and minimize uncertainty in latency estimation.

*AReNA* constructs an $aLP$ using non-randomly associated with each other; this ensures that the grouping benefits the prediction ability of the aLP. For this grouping, *AReNA* relies on information theoretic and statistical similarity measures computed for the pair-wise associations of iLPs. *AReNA* utilizes two similarity measures, namely *mutual information* [2] and *correlation* [5]. A higher mutual information (or correlation) between two $iLP$s means that these $iLP$s are non-randomly associated. Conversely, a mutual information of zero means that the joint distribution of $iLP$s holds no more information than their individual distributions. In general, there is no straightforward relationship between correlation and MI [4]. While correlation captures linear dependence, mutual information is a general dependence measure.

## 3 The *AReNA* Demonstration System

Our demonstration emphasizes the data management aspects of providing a scalable latency estimation. We demonstrate that *AReNA*: **1)** captures significant dynamic non-random associations between $iLP$s; **2)** provides a tool to construct aLPs to predict latency, and **3)** uses RNs to effectively manage a large number of iLPs and aLPs.

Figure 3 represents a snapshot of the *AReNA* control panel. Users can observe the evolution of the distributed environment that is being monitored via animated Relevance Networks. The animation reflects both the changing topology and the changing strength of non-random association between pairs of $iLP$s. The *AReNA* Visualizer was implemented using Zoomgraph Graph Analyzing and Visualization Software [15]. In addition to the Visualizer, *AReNA* supports data gathering and preparation to populate and maintain the $iLP$ catalog. It also constructs $aLP$s and participates in latency prediction.

### 3.1 Experiment Data Collection

During the demo presentation we shall use the following two data collections.

**CNRI Handle testbed.** The International Digital Object Identifier (DOI) Foundation (`www.doi.org`) and the community of publishers facilitate the identification and exchange of intellectual property over the Internet. Their application exploits the Handle protocol [12] for identifying and locating digital objects. Our testbed included the DOI server and publisher's Handle repositories and Web servers. We gathered latency data over CNRI Handle testbed during December 2003. We report on the performance of 22 clients (2 each on 11 client ASes) accessing 10 servers, yielding 220 *iLPs*.

***PlanetLab* testbed.** PlanetLab [8] is a globally distributed wide area testbed for deploying various network services at the Internet scale. PlanetLab currently consists of 350 machines, hosted by 150 sites, spanning 20 countries. The services experience all the behaviors of the Internet in terms of paths taken, latency, available bandwidth, connection properties, network presence and geographical location. All the PlanetLab machines run a common software package. Our

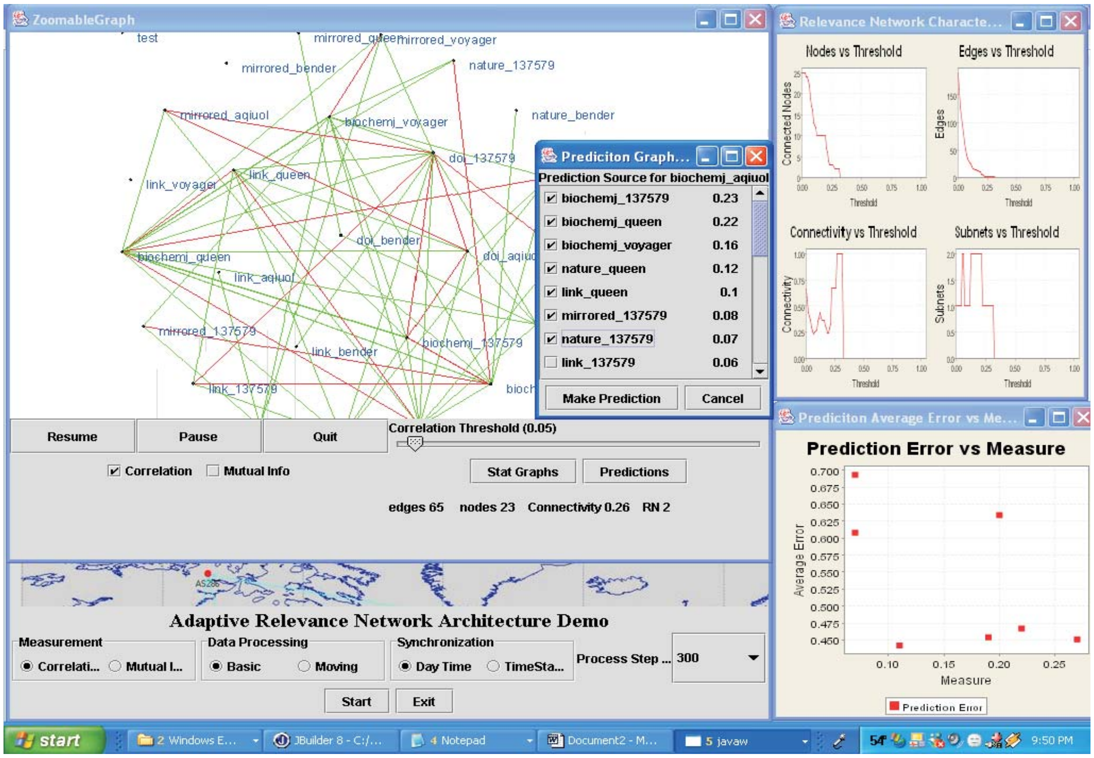


Figure 3: *AReNA* Control Panel

experiment gathered latency data in Summer 2004 for approximately 1600 *iLP*s.

### 3.2 Data Preparation

Based on the timestamp associated with the latency measurement of one $iLP$, we identify the corresponding latency in the pair, where correspondence is taken to be the latency with the closest timestamp. Then, we process the $iLPs$ and require that they share the same number of aligned samples (approximately 1000 samples for each $iLP$). This phase is aimed at ensuring comparable statistical measures when comparing among large numbers of $iLP$ pairs. Finally, for each pair of $iLP$s, we calculate correlation and mutual information (MI), resulting in two measures, ($iLP_1$, $iLP_2$, *correlation*) and ($iLP_1$, $iLP_2$, *MI*).

### 3.3 RN Generation and Analysis

*AReNA* computes pair-wise MI and Correlation values for all $iLP$ pairs. Each $iLP$ pair represents an edge in a complete $iLP$s relationship graph. *AReNA* produces a sequence of such graphs by varying the MI and correlation thresholds, starting from $th = 0$ with fixed increments. For each such threshold, *AReNA* modifies the $iLP$ relationships graph by discarding edges with $MI < th$ ($Corr < th$). Each transformation generates a group of connected subnetworks. The subnetworks correspond to the MI or Correlation Relevance Networks with respect to a given threshold. The evolution of these graphs and the correponding RNs are visualized by *AReNA*.

*AReNA* maintains several metrics of the RNs representing the associations among the $iLP$ pairs, for self-assesment. The metrics include the number of associations (edges that surpass the threshold), the number of participating nodes, the number of relevance networks and the connectivity (a ratio of the number of edges that surpassed the threshold to the number of all possible edges). Figure 4 illustrates two of these four metrics for different values of the MI threshold for CNRI Handle dataset.

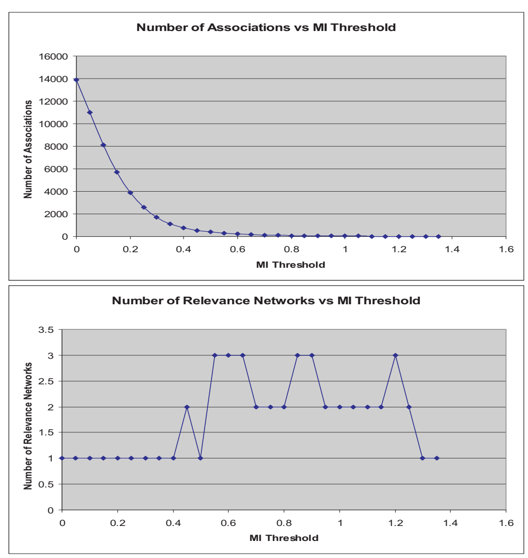


Figure 4: Characteristics of MI Relevance Networks as a Function of MI Threshold

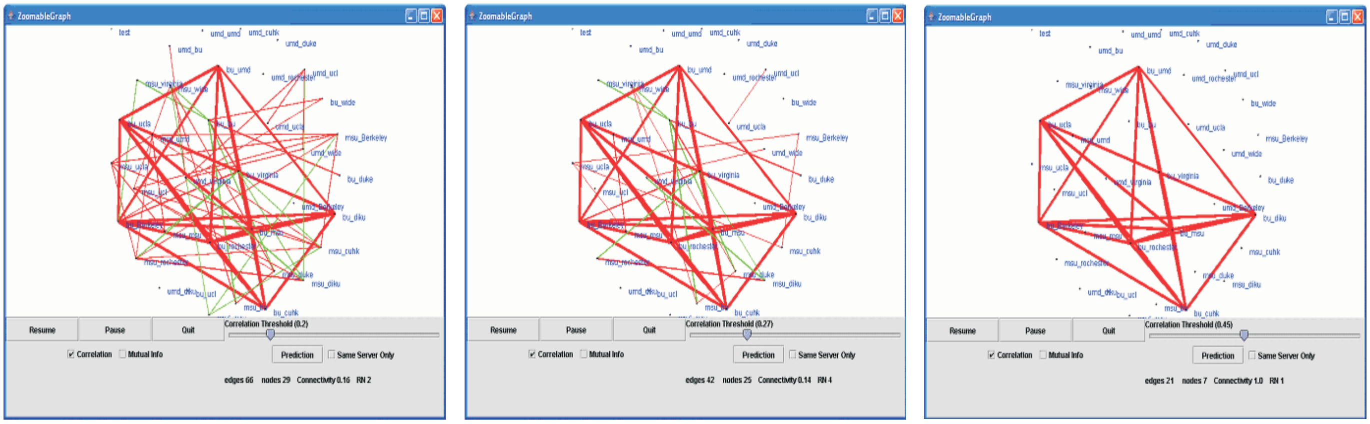


Figure 5: Examples of Correlation Relevance Networks generated by *AReNA*

As the threshold increases, the number of associations decreases, as is expected. However, the shape of the RN is surprisingly stable, compared to the change of the MI threshold. As we increase the threshold from 0.0 to 0.5, while we observe a steep decrease in the number of associations (from 14000+ to 1000+) there is only a *single* RN. As we increase the threshold further, the number of relevance networks increases, reaching $3$. To summarize, the key observation is that the MI relevance network is stable and is characterized by a small number of dominant $iLP$ clusters.

### 3.4 Managing Aggregate Latency Profiles with Relevance Networks

*AReNA* uses RNs to provide a birds-eye view of potential *aLPs*. In addition, by observing the changes of the RN, as the threshold is changed, one can also observe if a cluster is *strongly associated*, compared to the entire graph, or to other clusters. Such management features are valuable to provide scalable network management tools.

Figure 5 gives an example of correlation RNs generated by *AReNA* for three increasing treshould values and the Hndle data. Each node is a client-server pair (*e.g.*, `pubs-qew`) and an edge between two nodes represents a non-random association. The thickness of an edge reflects the strenth of the association. As we increase the threshold, only the *strongly associated* RNs survive. Such RNs can be used to construct an $aLP$.

### 3.5 Latency Prediction

After constructing an $aLP$ from a set of $iLP$s, *AReNA* improves the prediction quality of an $iLP$ using observations of other $iLP$s from the same $aLP$. We omit details due to lack of space but we note that our experiments have confirmed that high MI and high Correlation can indeed be used to identify exactly those $aLP$s (composed on $iLP$ pairs) with low relative error of prediction.

## References

[1] A.J. Butte and I.S. Kohane. Mutual information relevance networks: Functional genomic clustering using pairwise entropy measurements. In *Proc. Pacific Symposium on Biocomputing*, 2000.

[2] F.Reza. *An Introduction to Information Theory*. McGraw-Hill, 1961.

[3] S. Jamin, C. Jin, Y. Jin, D. Raz, Y. Shavin, and L. Zhang. On the placement of internet instrumentation. In *Proceedings of IEEE InfoComm*, 2000.

[4] W. Li. Mutual information functions versus correlation functions. *Journal of Statistical Physics*, (60), 1990.

[5] W. Mendenhall and T. Sincich. *Statistics for Engineering and the Sciences*. Macmillan Publishing, 1985.

[6] Zaiqing Nie and Subbarao Kambhampati. A frequency-based approach for mining coverage statistics in data integration. In *ICDE*, pages 387–398, 2004.

[7] V.N. Padmanabhan and L. Subramanian. An investigation of geographic mapping techniques for internet hosts. *Proceedings of the SIGCOMM*, 2001.

[8] The PlanetLab home page. http://www.planet-lab.org.

[9] D. Rubenstein, J. Kurose, and D. Towsley. Detecting shared congestion of flows via end-to-end measurement. *Proceedings of the ACM SIGMETRICS Conference*, 2000.

[10] S. Srinivasan and Ellen Zegura. M-coop:a scalable infrastructure for network measurement. *The Third IEEE Workshop on Internet Applications*, June 2003.

[11] M. Stemm, S. Seshan, and R. Katz. A network measurement architecture for adaptive applications. In *Proceedings of IEEE InfoComm*, 2000.

[12] S. Sun and L. Lannom. Handle system overview. *IRDM/IRTF Draft, 2001, http://www.idrm.org/idrm_drafts.htm*, 2001.

[13] V. Zadorozhny and L. Raschid. Query optimization to meet performance targets for wide area applications. *Proceedings of International Conference on Distributed Computing Systems (ICDCS)*, 2002.

[14] V. Zadorozhny, L. Raschid, M.E. Vidal, L. Bright, and T. Urhan. Efficient evaluation of queries in a mediator for websources. *Proceedings of ACM SIGMOD*, 2002.

[15] Zoomgraph: Visualizing zoomable data driven graphs. http://www.hpl.hp.com/research/idl/projects/graphs/.