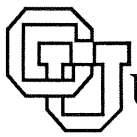
Resource Discovery in the Global Internet

Michael F. Schwartz

CU-CS-555-91 November 1991



University of Colorado at Boulder
DEPARTMENT OF COMPUTER SCIENCE

Resource Discovery in the Global Internet

Michael F. Schwartz

CU-CS-555-91 November 1991

Department of Computer Science
Campus Box 430
University of Colorado
Boulder, Colorado 80309
(303) 492-3902
schwartz@cs.colorado.edu

Abstract

Rapidly increasing wide area network interconnection promises vastly increased opportunities for remote collaboration and resource sharing. A fundamental problem that confronts users of such networks is how to discover the existence of resources of interest, such as documents, network services, and people. In this paper we overview efforts of the Networked Resource Discovery Project at the University of Colorado, Boulder, concerning resource discovery and a set of related problems.

1. Introduction

The global TCP/IP Internet is a powerful resource. Interconnecting millions of individuals at thousands of institutions worldwide, it offers the potential for significant inter-organizational collaboration and sharing of resources, such as documents, software, data, network services, and people. Yet, such sharing is currently relatively limited. The most widespread use is as a means of low-cost (government and industry subsidized) communication via electronic mail. Other less common uses include file transfer, remote login, and remote data acquisition. We envision a much more sophisticated *distributed collaboration* paradigm, where people accomplish tasks and share resources among many interrelated individuals across administrative boundaries. Collaboration across administrative boundaries is of particular importance for researchers because often a person's closest colleagues are at other institutions.

An increasingly important question for supporting such activities is how a user can discover the available resources, including machine resources (such as network services, databases and documents) as well as information about external resources (such as retail products, current events, and people). This *resource discovery* problem is important because it is an enabling aspect of distributed collaboration. Without the ability to discover resources of interest, users perceive only a very limited fraction of the full potential for sharing resources and collaborating with colleagues.

For the past four years, the Networked Resource Discovery Project at the University of Colorado has been investigating a number of problems associated with resource discovery. We impose three key goals on our approaches. First, we consider very large environments, spanning global internetworks. Such environments place stringent scalability requirements on the algorithms that can be used. Second, we want to avoid imposing artificial constraints on the resource space organization. Traditional directory services (such as the CCITT X.500 standard [CCITT 1988]) rely on hierarchical organization to achieve good scalability. Unfortunately, the organization of a hierarchy becomes convoluted as an increasingly wide variety of resources is registered, and requires users to understand how the increasingly deeply nested components are arranged. Finally, realistic approaches to the resource discovery problem must accommodate widespread administrative decentralization. To meet this objective, one must minimize the need for global agreement over protocols, information formats, and organizational structures. While standards are helpful, it is difficult to specify standards that are both globally adopted and technologically current. As an increasingly diverse collection of institutions contribute to the global information infrastructure, smooth evolution will require the ability to support multiple organizational structures, and to interoperate with a heterogeneous set of protocols and information formats.

In this paper we overview our research efforts. We begin in Section 2 by describing the range of problems that fall within the scope of our research. In Section 3 we overview our research efforts, and present measurements and other results from these projects. In Section 4 we discuss related work. Several of the projects discussed in 3 pertain to more than one problem introduced in Section 2. The problems addressed by each project are illustrated graphically in Section 5, where we review how our efforts relate to one another, and offer our conclusions.

2. Scope of Resource Discovery Investigations

There are a number of related problems that fit into the framework of our investigations. The first issue is how a user discovers the existence of some type of resource, such as a document about a particular topic. This problem is complicated by the fact that in many cases resources are not formally advertised. For example, many Internet hosts provide open access to a variety of software and documents, but administrators on these hosts do not announce them through any formal mechanism. At best, such resources are announced via ad hoc mail or news messages. At worst, they are not advertised at all. It is important that a resource discovery system be able to locate such resources, when doing so does not violate privacy or security boundaries.

Once a resource has been discovered, the next problem is how to locate an appropriate copy. For example, source code for various X window system applications is available from many different sites around the Internet, but it makes more sense to retrieve a copy from a site to which one has a high bandwidth network connection than a site connected by a heavily congested, low bandwidth, or unreliable network. Systems that support resource discovery typically ignore this issue, and simply provide the user with information about a list of a large number of copies of many different resources, in response to a query. It is left up to the user to sift through

this information, and find the responses relevant to the resource of interest, and then chose among the copies. Moreover, people often use simple heuristics to chose among copies (for example, preferring to retrieve files from sites in their country). However, as the global Internet becomes increasingly complex, making this determination manually will become difficult. Optimizing for network bandwidth, cost, and other factors will need support from the network layers responsible for routing, flow control, accounting, and policy considerations.

Another issue that arises once a resource has been discovered is that users often wish to keep track of information about the resource in a manner that makes it easier to recall the information in the future. While it might be argued that all resources should simply be organized according to a structure shared by all users, in practice users often have personal preferences concerning how resources should be organized, and what information should be kept close at hand.

More generally, resource discovery can be cast as a general framework for managing complex networked environments. There is little conceptual difference, for example, between locating overloaded network gateways in a network management system [Case et al. 1989] and locating nearby PostScript printers in a resource discovery system. Another problem that fits within this framework is supporting dynamic configuration for mobile hosts, so that a user could plug a computer into any part of the global Internet, and have it establish temporary addressing and routing support, contact mail and other network services, discover the location of needed physical devices (such as printers and fax modems), and register with the local accounting authority to be charged for these services.

3. Projects

Below, we overview our research efforts concerning a range of resource discovery problems. We discuss one project to support name based ("white pages") resource discovery; two different approaches to attribute-based ("yellow pages") resource discovery; two projects concerning the use of resource discovery techniques for supporting network and system management; a study of organizational properties in communication graphs, which has implications on resource discovery and distributed collaboration; a mechanism for supporting efficient distribution of information in a wide area network environment; and a measurement study of service connectivity in the global Internet.

3.1. Internet "White Pages" (Netfind)

An important special case of the resource discovery problem is providing "white pages" for the Internet, so that users my find each others electronic mail addresses. We have built and experimented extensively with a tool called "netfind", which uses a number of existing protocols and decentralized sources of simply structured information to support Internet white pages [Schwartz & Tsirigotis 1991a]. Using decentralized information avoids difficult problems of consistency and transfer of authority that are inherent in mechanisms that rely on building auxiliary databases to hold the information. The ability to use simply structured information is important in heterogeneous, administratively decentralized environments, where global agreement about structured information formats is difficult to achieve.

Search requests use the format "netfind *UserString InstString [InstString ...]*", where *UserString* identifies the user (typically by last name), and the conjunction of one or more *InstString*s identify the institution where the user works by name and/or geographic location. When a search is requested, netfind consults a *seed database* to obtain hints of potential machines to search, based on the specified institution keywords. This database is built by gathering organization names, city names, and host names from the headers of USENET [Quarterman & Hoskins 1986] news messages over time, and providing an inverted index into the data.

If the machines found in the seed database fall within more than three naming domains (an example of one domain being "cs.colorado.edu"), the user is asked to select at most three domains to search. The Domain Naming System (DNS) [Mockapetris 1987] is then contacted, to locate authoritative name server hosts for each of these domains. The idea is that these hosts are often central administrative machines, with accounts and/or mail forwarding information for many users at a site. Each of these machines is then queried using the Simple Mail Transfer Protocol (SMTP) [Postel 1982], in an attempt to find mail forwarding information about the specified user. If such information is found, the located machines are then probed using the "finger" protocol [Zimmerman 1990]. The results from finger searches can sometimes yield other machines to search as well. Ten

lightweight threads are used to allow sets of DNS/SMTP/finger lookup sequences to proceed in parallel, to increase resilience to host and network failures.

Figure 1 illustrates the flow of data and events in a single thread of the optimal case, when the Domain Naming System, SMTP, and finger searches are all successful. Netfind can often find a user even if the remote site does not support all of the above protocols, or if some steps in the protocol sequence fail. For example, if finger is disabled because of security concerns, mail forwarding information may sometimes still be found. Or, if no mail forwarding information is found, netfind attempts to finger some of the machines matched from the seed database. Similarly, netfind can proceed without information about authoritative name servers. This ability to function in the presence of failures or partial remote protocol support is an example of a technique for supporting fault tolerant resource discovery without global agreement. We utilize this technique to a more significant extent in our network visualization project, described later in this paper.

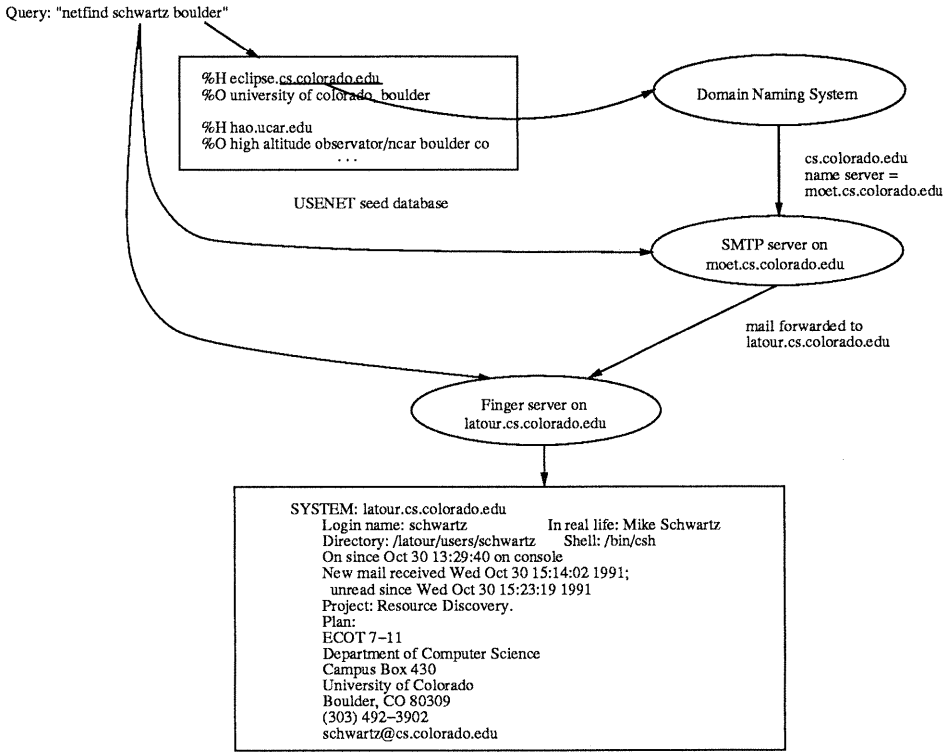


Figure 1: Single Thread of Netfind's Optimal Case Search Algorithm

Because many different institutional keywords will lead to the same seed database records and Domain information, it is usually quite easy to "guess" keywords that will succeed for any particular search. Moreover, netfind's tolerance of partial remote protocol support allows it to locate information about a large proportion of Internet users. Measurements indicate that netfind can locate information for approximately 1,501,000 users in 2,524 sites worldwide, broken down as indicated in Table 1.† This scope is significantly larger than other existing Internet directory services, which require that users register with an administratively centralized service (as with the SRI Network Information Center WHOIS service [Harrenstien, Stahl & Feinler 1985]), or that special directory servers be run at many sites around the Internet (as with X.500).

Netfind's ability to use highly decentralized information sources allows it to locate very timely information about users. Unlike services that use an auxiliary database that must be updated by a separate administrative

† These estimates are based on measurements done 6 months later than those reported in [Schwartz & Tsirigotis 1991a]. The set of reachable sub-domains (and users for which white pages can be located) has grown by 31% since the original measurements, because of Internet growth. By now the scope is significantly larger still.

Top-Level Domain Name	Description	Reachable Sub-Domains	Top-Level Domain Name	Description	Reachable Sub-Domains
edu	U.S. Educational	1184	dk	Danish	14
arpa	ARPANET names	245	it	Italian	11
com	Commercial	322	nz	New Zealand	11
au	Australian	114	ch	Swiss	10
ca	Canadian	113	at	Austrian	5
gov	U.S. Government	93	il	Israeli	5
mil	U.S. Military	89	us	U.S.	5
de	German	49	is	Icelandic	4
se	Swedish	46	uk	British	4
nl	Dutch	37	kr	Korean	3
org	Non-profit	33	mx	Mexican	2
jp	Japanese	28	es	Spanish	1
fi	Finnish	25	gr	Greek	1
fr	French	25	in	Indian	1
net	Named by network	23	pr	Puerto Rican	1
no	Norwegian	20			

Table 1: Breakdown of Top-Level Domains Reached by Netfind

procedure, netfind probes the machines on which users do their daily computing. To the best of the author's knowledge, all other white pages services (including WHOIS and X.500) depend on some form of auxiliary database. Populating and keeping such a database up-to-date are difficult tasks. The potential downside of searching for such timely information is increased search costs. To gauge these costs, we built a mechanism into a research version of netfind so that after each use, it would transmit network load and other measurements to a logging server at the University of Colorado. We distributed this version of netfind to researchers at 24 institutions worldwide, and recorded data for six months. During this time, 119 users (excluding the author and related researchers) used netfind 2,302 times. The mean network load per search was 136.33 packets. Combining this with usage frequency distribution measurements, we estimate that widespread use of netfind would contribute only a fraction of a percent to Internet load. While the per search cost of netfind is higher than a registration-style directory like X.500, we believe this is quite a reasonable price to pay for providing timely information without requiring global cooperation, particularly when one considers the capacity of next-generation high speed networks.

Netfind is in active use at over 60 institutions worldwide, and is being developed further commercially.

3.2. Probabilistic "Yellow Pages"

Netfind supports name-based searches. Attribute-based ("yellow pages") searches represent a more difficult problem, because the breadth of resources is larger, and there are many different ways that users would like to describe resources in search requests. The typical approach to this problem is to impose a standard global taxonomy on the resource space. The choice of this taxonomy is important, since searching for resources is vastly more efficient if the searches correspond to the way the system is organized. For example, it would be prohibitively expensive to search for information about commercially available bulk data storage devices in a worldwide system organized by institutional bureaucracy, since that would require examining the records of each institution. In contrast, it would be efficient to search if storage media were an explicit subtree of the taxonomy.

Because it is difficult to reach consensus and maintain meaningful organization over time in a global taxonomy, we took a different approach to the problem. We chose to focus on a means for disseminating resource information and distributing search effort in such a fashion that information migrates to where it is needed, reducing the costs of popular searches. Our approach involves the use of probabilistic algorithms for constructing and searching a resource graph [Schwartz 1989, Schwartz 1990]. We assume it is acceptable to find a small

number of instances of a moderately large class of resources. For example, in searching for a supplier of a particular piece of computer hardware, finding 5 out of 100 suppliers in a metropolitan area would often suffice. We also assume that it is acceptable to return different answers to the same query across search sessions. If consistent responses to queries are desired, one could build a front-end user interface that cached results, and provided identical responses across searches.

Based on these assumptions, we designed a protocol to support a set of *agents* in organizing and searching the resource space. Agents maintain pointers to sources of resource information, and access these sources via intermediary *brokers*, which enforce the access control policies and encapsulate the heterogeneity of the information repositories. While agents are intended to be part of the network infrastructure, each broker belongs to the organization whose resource information it exports. This functional breakdown is illustrated in Figure 2

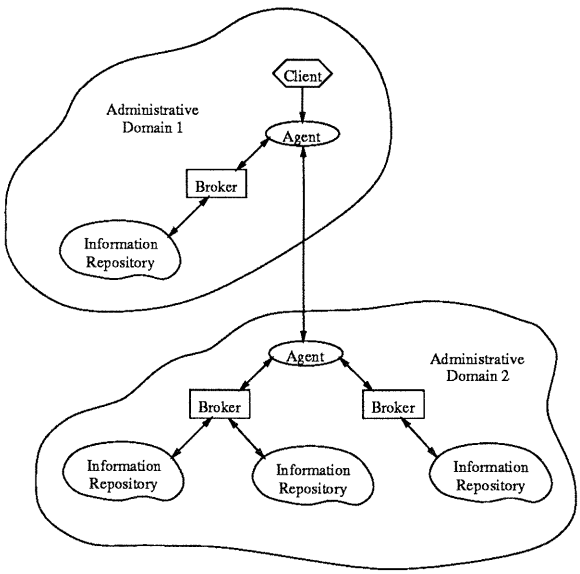


Figure 2: Architecture of Probabilistic Yellow Pages

While brokers are an important part of the model, the main focus of this research is on the agent protocols for organizing and searching the resource space. Rather than having an administrative body specify how the space is organized, agents organize the space dynamically, according to what resources exist and the types of searches users make. Agents use a probabilistic *Sparse Diffusion Multicast* primitive to disseminate information about resources at uniformly distributed, randomly chosen nodes around the network, and likewise to route search requests randomly around the network. Sparse Diffusion Multicast can be implemented with simple modifications to previous work in Internet multicast [Cheriton & Deering 1990].

Randomly disseminating resource information is intended to place the information within a reasonably small neighborhood of any agent in the network, so that during searches it is likely that the information can be found using simple random probes. Since the types of resources that exist and the searches users request are not random, a cache management policy is used to prefer graph edges between agents that maintain related information, to form *Specialization Subgraphs (SSGs)*. Using this policy, a search initiated at a random agent may cause some random search behavior at the start of the search, until a member of an appropriate Specialization Subgraph is reached. If such a subgraph is reached, searches proceed in a more directed fashion. If a user continues to use the same agent, over time that agent will maintain pointers to sources of the type of information for which that user often searches.

To analyze the effectiveness of this approach, we built a detailed simulation, which computed the proportion of sites originating resource information that could be found, as a function of a number of different variables, including the number of agents in the network, the number of different resources, the fan out of sparse diffusion multicasts and the depth of recursion used during searches, cache replacement policy, cache size, and number of cache exchanges before a search was initiated. The results indicate that this probabilistic approach can support a non-hierarchical resource space for an environment roughly the size of a country, with several thousand sites

participating in resource registration and searches. For example, Figure 3 indicates the differential effectiveness of First In First Out and Least Frequently Used cache replacement policies, as a function of the number of cache exchanges before a search was initiated. This plot shows that search effectiveness grows approximately linearly until all caches fill under LFU caching, at which point caches begin to overflow and decrease search effectiveness. FIFO caching performs quite poorly here because it has no basis for making good use of an increasing number of cache exchanges.

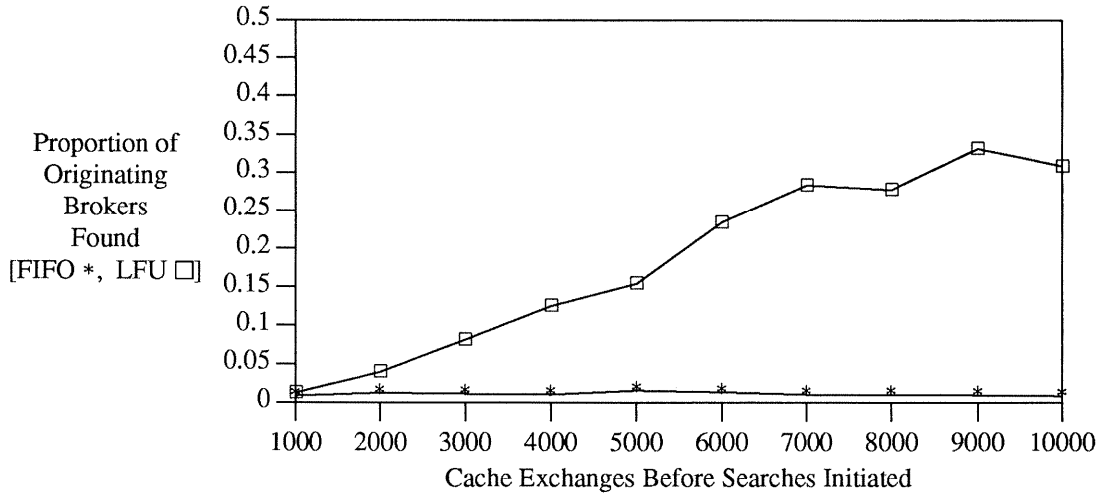


Figure 3: Probabilistic Yellow Pages Search Effectiveness as a Function of Cache Exchanges

The probabilistic nature of this approach also supports fair access among competing information providers, an important issue in commercial environments such as the U.S. telecommunications industry [Greene 1988], computer reservation systems [Gifford & Spector 1984], or the future Internet [Kahin 1990].

3.3. Internet Resource Mapping/Discovery Project

We explored a second approach to the yellow pages problem, which emphasizes the potential for a distributed collection of people to contribute to a shared map of the global resource space. A fundamental assumption of this project is that the resource space is so large that it cannot be completely organized. For this reason, mechanisms are needed to support incremental organization of the resources, based on the efforts of many geographically distributed individuals, and a range of different information sources of varying degrees of quality. Our approach to this problem is to use mechanisms that "tap into" existing network protocols and information sources to provide an immediately useful tool (much as netfind did), supplemented by mechanisms that allow users to superimpose additional organization on the resource space in an incremental fashion [Schwartz et al. 1991b]. As a concrete test case, we developed a prototype that focuses on public Internet archive sites accessible via the "anonymous" File Transfer Protocol† [Postel & Reynolds 1985]. This is an interesting test case, because it encompasses thousands of administratively decentralized sites containing a collection of resources of considerable practical value.

In the prototype implementation, three levels of information quality are supported. At the highest level, resources are described using archive-site-resident databases, with individual resources described according to their conceptual roles. An example of the contents of one such database is shown in Figure 4. The fields in each record contain file names, path names, the FTP command needed to cause the file transfer, a description of the resource, and keywords describing the resource. This database is constructed in large part with automated tools,

† FTP is an Internet standard protocol that supports file transfers between interconnected hosts. Anonymous FTP is a convention for allowing Internet users to transfer files to and from machines on which they do not have accounts, for example to support distribution of public domain software.

```
%A /pub/X/contrib
%B andrew
%C get andrew
%D Andrew -- X windows interface prototyping tool, from Carnegie Mellon University.
%K user interface source code cmu tool prototype X windows browse
```

Figure 4: Example Internet Mapping Site Archive Database Contents

and can be modified manually to improve keys. The next lower level information quality is provided by per-user and per-user-site caches, which record resources that have been found by individual users during their explorations. At the lowest level, the system scans USENET electronic bulletin board articles using a simple set of heuristics to recognize announcements about public archive sites, to provide a simple keyword-based index of resources throughout the Internet.

To support users in superimposing additional organization on the resource space, the architecture also supports a mechanism in which any group of users who share common interests can build a structure (called a *view*) that superimposes organization on the resource space according to their particular interests.† For example, a group interested in graphics might build a view that organizes the world according to PostScript, Tools, Window Systems, Images, and Discussions, with pointers to network accessible resources of these various types nested into this structure. A view is intended to be a simple structuring mechanism for loosely integrating an administratively decentralized pool of resources, with properties much like those of Specialization Subgraphs. Views can include pointers to parts of other views, so that related interest groups (such as people interested in operating systems and people interested in data communications) can cross reference each others' views. Views are not constrained to be hierarchical, although in practice many links will probably be tree-like.

We place particular focus on supporting searches. Our approach is to build a simple flat index of each view. While flat searching has not worked well in some situations involving large scale (such as library information systems), searching a view in this manner should work well because any particular view will be fairly small and highly focused in scope. Because of this, it should be relatively easy to "guess" appropriate keywords for searching a view, without the difficulty of keywords matching many unrelated subjects. Moreover, the underlying space in a view is structured. Hence, a user could examine a subtree in a view once a match occurs, unlike the corresponding situation with the flat underlying spaces supported by information retrieval systems [Salton 1986].

Of course, the problem remains that a user must discover appropriate views to search when trying to find a resource. To address this problem, an area of future research we intend to pursue involves experimenting with a means of automatically interelating views, based on the interest clustering algorithm developed in connection with our electronic mail study (see Section 3.5). Doing so will provide a dynamically evolving set of links between related views, and allow users to search for resources without having to know what views exist.

3.4. Network Visualization Project

Our network visualization project focuses on discovering information about networks, such as topology, congestion, routing, and protocol usage [Schwartz et al. 1991a]. As with netfind, we use a number of protocols and information sources, to support discovery in the absence of global agreement on any one protocol or information source. For the visualization project, however, we use a much more extensive collection of protocols and information sources, including the Address Resolution Protocol [Plummer 1982], the Internet Control Message Protocol [Postel 1981], the Domain Naming System [Mockapetris 1987], the Simple Network Management Protocol [Case et al. 1989], and a dozen others.

---

† Views are conceptually similar to the Specialization Subgraphs introduced in Section 3.2.

An explicit component of our approach is the recognition that different sources of network information have different characteristics with respect to timeliness of discovered information, discovery expense, danger of generating network problems (such as broadcast storms), and completeness of discovered information. In contrast, network management systems based on a single standard are limited to the characteristics provided by that standard. Usually the standard focuses on a particular perspective of network management, which therefore limits its scope. SNMP, for instance, takes the perspective that the network is essentially a collection of devices that can be instrumented and measured, to determine packet flow rates, routing table contents, etc. A different set of characteristics may be detected if passive packet monitoring servers are installed in the network, or if the network management system supports directed probes into a network. For example, passive monitoring would allow broadcast storms to be detected without imposing added processing load on a gateway. Active probes would allow networks to be examined even if no monitoring server were installed on some network segments. This might be useful for network management when it is infeasible to install monitors on all segments. Our architecture supports all of these perspectives.

The architecture uses a collection of servers distributed around portions of the internet being instrumented, as illustrated in Figure 5. Each server periodically executes a set of discovery protocols to maintain information about certain segments of the network. Typically, a server will reside on a particular segment of a local internet, and passively gather information about that segment (monitoring routing table update traffic on a broadcast LAN, for example). In the figure, passive monitoring is indicated by thin arcs with arrowheads directed into the discovery server. The architecture also allows active probes (indicated in the figure by thin arcs with arrowheads directed out from the discovery server). This capability can be used to monitor multiple network segments, to reduce the number of discovery servers that must be installed to instrument an internet. This active probe capability can also be used by clients to "launch" a probing network discovery operation to a portion of the internet where no discovery server has been installed. The scope of a discovery server is specified in terms of a set of networks, as indicated by the dashed cloud surrounding three networks in the figure.

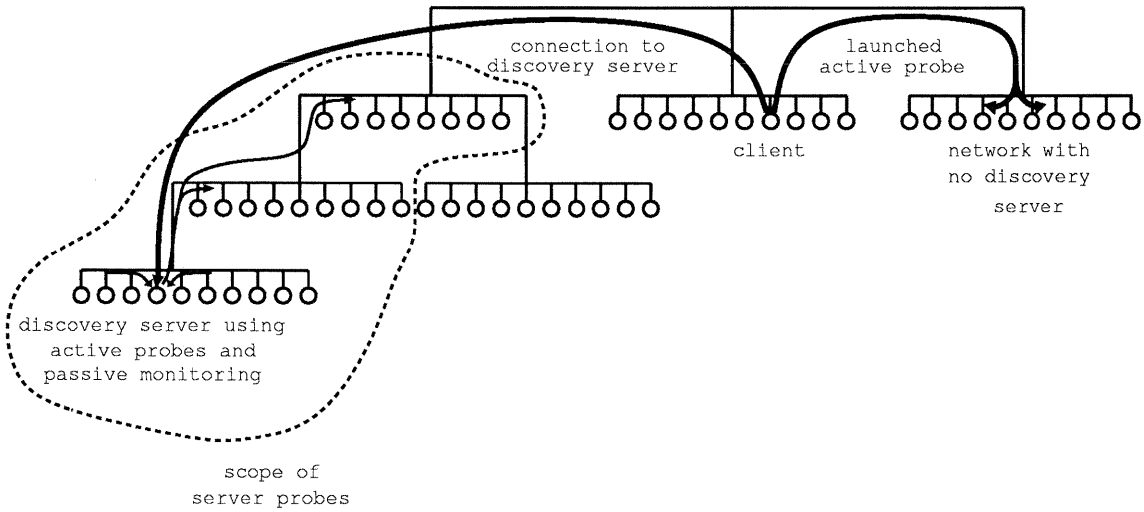


Figure 5: Network Visualization Architecture

The architecture also provides a discovery protocol scheduler and state manager that allows a system administrator to specify which discovery protocol modules will be scheduled and how frequently, according to his/her perceptions about the relative importance of the various protocol characteristics (timeliness of discovered information, etc.). Network protocols can be specified in a table that describes each protocol, where the executable code to use it resides, and its characteristics. In this fashion, the set of network protocols that are supported is easy to change without rebuilding the system.

To support scalable operation, the architecture includes a mechanism to cache discovered network information. In addition, queries may specify predicates, to avoid retrieving unchanged or unneeded information. For example, a query may request information about all of the hosts on a particular network segment that have been

discovered since the most recent cache entry. In addition to enhancing scalability, this mechanism can improve the responsiveness of browsing operations, for the case where users are browsing parts of an internet that have been viewed recently.

The prototype is implemented in C on top of UNIX,† and runs on the University of Colorado local area internet. We have implemented three different discovery mechanisms. The first mechanism sequences through a range of addresses on a local network segment, to discover the Ethernet addresses of each node on the network, and then probes the ARP cache to determine the Internet address associated with each of these hosts. It then performs a name lookup to determine the host name associated with each node. The second mechanism uses the Routing Information Protocol [Hedrick 1988] to determine the gateway topology of the network. The third mechanism uses broadcast Internet Control Message Protocol [Postel 1981] "echo" messages to discover the hosts on a remote network segment. To reduce the problem of broadcast storms, we are modifying this protocol to use a sequentially incremented Time-To-Live (TTL) mechanism similar to Van Jacobson's traceroute program, so that a broadcast directed at a remote network will reach that network with a TTL value of 1.

3.5. Global Electronic Mail Study

A question central to both resource discovery and distributed collaboration is how to organize a large, administratively decentralized, constantly evolving system. As a point of departure, we became interested in the organization of human social networks. Such networks use a non-hierarchical organizational structure that scales well. Rather than forming contacts with each other based on a hierarchy, people often establish more direct "networks", by contacting knowledgeable intermediaries who can quickly refer them to other relevant people, cutting across bureaucratic boundaries. For example, by contacting a computer science professor or network manager, someone interested in high-speed networking technology can quickly meet other people who share this interest. These people can, in turn, introduce the person to others who perhaps more closely share his/her particular interests. At the same time, the newcomer can be instrumental in pointing out individuals who share other interests with the people he/she meets. This graph structure is conceptually similar to the Specialization Subgraphs introduced in Section 3.2. An SSG is a subset of nodes that share common attributes, and that has a small diameter. As an example, in a graph of relationships among people, one SSG could connect individuals based on a shared interest in a particular computer science speciality, a second SSG could connect individuals based on shared responsibilities at a place of employment, and a third SSG could group individuals based on shared cultural/recreational interests. Any individual can belong to many different SSGs, and can search for information about a particular topic by consulting the appropriate SSG.

To study the organizational properties of SSGs, we collected mail logs from 15 sites around the U.S. and Western Europe for two months, and analyzed this data using graph theory and traffic analysis [Schwartz & Wood 1991]. The data collection sites are summarized in Table 2.

Geographic Regions	Types of Institutions	Institution Sizes
• 5 on U.S. West Coast • 2 in U.S. Mountain Region • 4 in U.S. Central Region • 2 on U.S. East Coast • 2 in Western Europe	• 11 universities • 3 research laboratories • 1 product development firm	• 4 with 50-200 people • 7 with 200-1,000 people • 4 with 1,000+ people

Table 2: Data Collection Sites

The data we collected constituted a graph containing approximately 50,000 users in 3,700 different sites around the world. Applying graph theoretic analyses yielded a number of insights about how users collaborate by electronic mail. We found that the average path between people is short (5.96 ± 6.6%), which is a statistically rendered version of the small diameter postulated by the so-called "small world" phenomenon [Travers &

† UNIX is a trademark of AT&T Bell Laboratories.

Milgram 1969]. This property indicates that the graph can support rapid information dissemination. Furthermore, we found that the graph edges are highly redundant, indicating that the graph can support reliable information dissemination. These properties are highly sought in computer networks, yet arise naturally in human social networks.

To exploit properties of SSGs, we developed an algorithm to cluster individuals by shared interests. The algorithm works without access to the contents of the mail messages, by computing properties of the mail interconnection graph using traffic analysis techniques [Callimahos 1989]. We found it necessary to apply this algorithm to a subgraph derived by a graph reduction technique that eliminated "noise" caused by our statistical sampling. The graph reduction involved first building a subgraph whose edges consisted of the edges from the original graph that spanned administrative boundaries, and then iteratively constructing a sequence of subgraphs, each obtained by removing all of the nodes with degree 1 and their incident edges from the previous subgraph, until no such nodes remained. (The reader interested in the reasons for these reductions is referred to [Schwartz & Wood 1991].)

Given this reduced graph, we experimented with a variety of functions for ranking the "interest distance" between two individuals. Applying such a function between a particular starting node and every other node in the graph yields a list of node interest distances, which can be sorted to provide a ranking of nodes by closeness of interests to the starting node. By applying various interest distance functions starting from nodes we knew personally, we eventually selected an function that could isolate many individuals who shared interests with the starting nodes. As a concrete example, Table 3 shows the closest 12 entries computed around the author. As can be seen, in most cases the isolated individuals had interests related to the author's, whose interests lie primarily in networks, distributed systems, privacy/security, and performance. The individuals in this list were not trivially derivable from the communication graph. Several of the individuals were not known personally by the author, indicating that the algorithm uncovered relevant individuals with whom the author had never exchanged electronic mail. In fact, Table 3 includes one individual whom the author did not know at the time of this study, to whom the author was later introduced by a third party, because their work was related. Moreover, the list omits a number of people with whom the author did communicate, whose interests are not closely related to the author's. Finally, many of the individuals in the list were not at any of our data collection sites. The algorithm was able to derive information about these individuals given data about only a tiny proportion of the total electronic mail community (15 sites around the world).

Relationship to Schwartz	Distance from Schwartz
Schwartz	0.0000
Graduate student at university where Schwartz studied, involved in networking research	0.7692
Theory professor who studied where Schwartz studied	0.8462
Unknown student at a Southwestern U.S. university	0.8571
Industrial systems researcher who studied where Schwartz studied	0.8824
Schwartz's Ph.D. advisor (interested in performance and distributed systems)	0.9048
System administrator at an East Coast U.S. university	0.9048
Systems and security researcher at a Midwest U.S. university	0.9130
Ph.D. advisor of Schwartz's Ph.D. advisor (interested in performance and systems)	0.9167
Performance and Systems researcher at a West Coast U.S. university	0.9167
System administrator at an East Coast U.S. university	0.9167
Head system architect, government research laboratory	0.9167

Table 3: Top of Computed Aggregate Specialization Graph Surrounding Schwartz

This clustering algorithm does not directly indicate how individuals are related, since individuals can be related by many different shared interests. However, by starting with several individuals known to have a common interest and intersecting the graphs formed by running the algorithm on each individual, one can derive a

set of individuals who likely share that particular interest. Experimenting with this idea produced very promising lists of persons related to chosen starting people, concerning shared interests in such closely related areas as distributed computing, networks, and naming. By specifying only a few "seed" users, many other highly relevant individuals were found.

Applying this algorithm to each of a randomly chosen subset of 500 nodes within the graph provided measurements of how people collaborate, indicating the existence of a large number of different but heavily interrelated groupings of individuals based on shared interest, and underscoring the importance of supporting a number of different organizational structures for distributed collaboration.

Clearly, the clustering algorithm raises significant privacy issues. At the same time, it offers intriguing possibilities for supporting distributed collaboration, by extracting implicit organizational information from a communication graph. We believe there are situations in which the algorithm could be applied without invading privacy. One example is interrelating views from our Internet Resource Mapping/Discovery Project (see Section 3.3). As a second example, we intend to experiment with the algorithm to provide an implicit organizational index of the recently deployed online version of the CCITT Blue Book standards documents on one of our file servers [Malamud 1991], based on logged accesses to the various documents.

3.6. Wide Area Demand Resource Distribution Project

Our Probabilistic Yellow Pages Project (Section 3.2), Internet Resource Mapping/Discovery Project (Section 3.3), and experimental deployment of International Telecommunications Union standards documents all underscore the need for an effective means of disseminating information around a large network. Motivated by this need, we are exploring mechanisms for widely distributing files without broadcasting (as is used by the Network News Transfer Protocol [Kantor & Lapsley 1986]), and without causing files to be retrieved multiple times across individual network links (as is the case with anonymous FTP). The basic idea, suggested by Phil Karn of Qualcomm, Inc., is to distribute files in response to requests for them, caching them at intermediate nodes along a dynamically developed spanning tree of the Internet. Doing so can potentially reduce the load on the Internet substantially, since FTP currently accounts for 45% of the bytes transmitted on the NSFNET backbone [NSF Network Service Center 1989]. Moreover, this mechanism could underly a network-transparent mode of resource sharing, whereby resources are named independently of the particular hosts that hold them. This technique will be best supported by an Internet multicast mechanism [Cheriton & Deering 1990], but will also function without such specialized support.

A number of issues must be addressed in developing such a mechanism, including accountability of cache contents (to protect against accidental or malicious replica modification) and cache consistency. Before addressing these issues, however, we are working on two more basic problems: measurements of duplicate traffic in the current Internet, and efficient bulk data transport.

Internet Duplicate Transmission Measurements

Before implementing a demand distribution mechanism, we are beginning a measurement experiment to gauge the extent to which duplicate network traffic flows across various Internet gateways. These measurements will indicate how much a demand distribution mechanism could reduce network traffic. The measurement software samples 20 dispersed bytes from each FTP file transfer across a particular network, and uses this data as a probabilistic test of file identity.† The program will track the names and access counts of the 10,000 most frequently transferred files. Collecting this data at various levels of the Internet (such as a LAN, regional network gateway, and perhaps an NSFNET backbone router) will allow us to determine the expected data transmission savings that could be realized by implementing an Internet-wide demand distribution mechanism.

---

† Equality cannot be established by file names, since they are not unique across Internet nodes, and relative names are not even unique across a single node.

Bulk Data Transport Protocol for Future High Bandwidth-Delay Networks

To support efficient data transfer, we have designed and implemented a transport protocol optimized for low loss, high bandwidth, long haul networks [Schwartz & Schaefer 1991]. While the bandwidth supported by future networks will exceed one gigabit per second, the delay will be limited by the finite propogation speed of electromagnetic waves. Because of this, typical windowed flow control protocols (such as TCP/IP) will suffer performance degradation, as the sender must wait relatively long periods of time before proceeding when packets are dropped. For example, the minimum round trip time across the U.S. is approximately 30 milliseconds. A sender forced to wait that long for an acknowledgement will lose the opportunity to transmit over 32 megabytes of data in a network operating at one gigabit per second. Our protocol optimizes for such networks by increasing the proportion of time that the sender can proceed asynchronously from the receiver. To do this, we specialize the protocol to the case of bulk data transfer. Our protocol would not be well suited to a stream-based application (such as remote login), since the protocol does not provide any guarantees about the order in which packets arrive. It guarantees only that all packets arrive, tagged with sequence numbers that allow the transmitted file to be reconstructed.

The first step towards the goal of increasing asynchrony is to maximize the information contained in an acknowledgement packet by using an acknowledgement vector, such that each bit of the vector acknowledges a transmitted packet. As acknowledgement packets are received, the sender computes the proportion of packets that have been lost, and adjusts its packet flow rate accordingly. The sender does not retransmit any packets until it has progressed through the entire file. It then begins retransmitting based on the most recently received acknowledgement vector. Because of this, packets that were in transit but have not yet been received (and hence checked off in an incoming acknowledgement vector) may be marked as received in a vector that arrives later. We also implemented slow-start and slow-stop mechanisms to allow the sender and receiver to approach maximum network bandwidth without pushing the network into congestive collapse [Jacobson 1988].

Our protocol runs on top of UDP [Postel 1980]. Because it does not run in the kernel, it is easy to install, but could potentially realize poor performance. Yet, as illustrated in Table 4, the performance of our prototype implementation is quite good. These measurements compare the cost of transmitting a 200k byte file across the Internet (between Boulder and Seattle) using our ("AckVec") protocol, MIT's NETBLT protocol [Clark, Lambert & Zhang] and TCP.† Since it turns out to be easy to increase network throughput by transmitting packets without regard to network congestion, we also measured wasted packets. The minimum number of packets needed is the file size divided by the maximum packet size, plus overhead associated with packet headers, transmission set-up and shut-down, acknowledgements, and dropped packets caused by network congestion. NETBLT wastes packets in these measurements because it uses a flow control policy tuned for a particular network speed. AckVec and TCP dynamically adapt to network speed. We also show code sizes of the 3 protocol implementations, as a rough indication of protocol complexity. We expect that the simplicity of AckVec will allow it to run at lower CPU utilization than other bulk data transfer protocols, increasing its attractiveness for heavily utilized archive file servers.

Protocol	Throughput [bytes/sec]	Protocol Inefficiency [% wasted packets]	Protocol Complexity [lines of C code]
AckVec	28,743	24	589
NetBlt	27,562	46	5,375
TCP	16,767	26	3,442

Table 4: Preliminary Measurements of AckVec Protocol

We are currently tuning the AckVec protocol for higher performance. We are also implementing a version of FTP capable of using our protocol as well as TCP, with a negotiation step at the start. In this fashion, our FTP

† The version of TCP we have includes only Jacobsen's congestion control improvements [Jacobson 1988], without his header prediction modifications and other performance improvements. We are currently attempting to build and measure a version of TCP with these performance improvements.

client and server will be backwards compatible with servers and clients that support only TCP-based file transfers, and will support higher performance for the situation where both a client and a server support AckVec. We plan to distribute this software to sites around the Internet. Before doing so, however, we plan to install (compile time disablable) support for resource discovery, to help infuse that mechanism into the Internet infrastructure at the same time as we distribute our performance improved FTP software.

3.7. Measurements of Internet Service Reachability

Implicit in the projects described so far is the assumption that the Internet will continue to grow and evolve as a medium for supporting wide area distributed applications. However, the same connectivity that offers collaboration potential also threatens security. In response to a number of well publicized events over the past few years, many sites have imposed a range of mechanisms to limit their exposure to security intrusions. While these measures are preferable to the damage that could occur from security violations, taken to their logical extreme they could eventually reduce the Internet to little more than a means of supporting certain pre-approved point-to-point data transfers. Such diminished functionality could hinder or prevent the deployment of important new types of network services, impeding both research and commercial advancement.

To understand the evolution of this situation, we are carrying out a study to measure changes in Internet service-level reachability over a period of one year. The study considers upper layer service reachability instead of basic IP connectivity because the former indicates the willingness of organizations to participate in inter-organizational computing, which will be an important component of future wide area distributed applications. The data we gather will be useful for both Internet research and engineering planning activities. They will also be of general interest, as they represent direct measurements of the evolution of a global electronic society.

The study consists of a set of runs of a program over the span of one to two days each month, repeated monthly for a period of one year. Each program run attempts to connect to 13 different TCP ports at each of 12,865 Internet domains worldwide, recording the failure/success status of each attempt. The program attempts no data transfers in either direction. If a connection is successful, it is closed immediately. The machines on which connections are attempted are selected at random from a large list of machines in the Internet, constrained such that at most 1 to 3 machines is contacted in any particular domain. The list of ports was chosen to span a representative set of services that can be expected to be found on any machine in a domain (so that probing random machines is meaningful). Only TCP ports are used, since they allow one to determine if a server is running in an application-independent fashion.

Clearly, a study of this nature raises a number of potential concerns regarding privacy, security, and network/remote site load. A study plan overviews our experimental design, considerations of network and remote site load, mechanisms used to control the measurement collection process, and efforts to inform administrators at sites measured by this study, along with concomitant privacy and security issues [Schwartz 1991a].

4. Related Work

WHOIS Service

The Defense Data Network Information Center provides a centralized TCP-based Internet directory facility called the WHOIS service [Harrenstien, Stahl & Feinler 1985]. This directory is helpful, but the database contains only the small fraction of Internet users who have registered with the NIC. Moreover, the information is often out of date, since people who register often forget to update the NIC when their information changes (e.g., when they change work addresses).

X.500

CCITT has developed a directory service standard called X.500, which involves a hierarchical collection of servers running at participating sites, each of which maintains structured directory information about that site [CCITT 1988]. Browsing and searching operations are supported. Performance Systems International has deployed a prototype implementation of X.500 [Rose & Schoffstall 1989]. A difficulty with standards such as X.500 is that they require that a large number of standard-conformant servers be deployed in order to achieve a reasonably extensive directory.

Nomenclator

Ordille has developed a system called Nomenclator for supporting relational queries, and a prototype that operates in conjunction with the PSI X.500 prototype [Ordille 1991]. The architecture involves the use of a query planning mechanism called catalog functions, which limit the scope of searches. Catalog functions are cached and reused at each query site, tailoring the query processing environment to the needs of users at that site.

Profile and Univers

Peterson et al. developed a system called Profile that supports queries over general types of objects, based on their Universal Naming Protocol [Peterson 1988], which they later evolved into the Univers system [Bowman, Peterson & Yeatts 1990]. Like several of the Networked Resource Discovery Project systems, Profile and Univers support a non-hierarchical name space. However, Profile and Univers focus more effort on the structure of query mechanisms for supporting directory services, whereas our work focuses on supporting resource discovery in the absence of global cooperation.

Prospero

Neuman has built a prototype file system called *Prospero* based on a notion called "user centered naming", which allows users to construct their own views of the accessible files [Neuman 1990]. This system is based on three primary mechanisms. The *union link* provides a means of building a directory that contains references to files and other directories that are stored on remote nodes. The *filter* is an interpreted-language based specification that allows one to alter the set of entries that are seen in directories whose paths pass through it. *Closure* is used to explicitly associate a view with each name, to resolve the differences in names used to refer to objects from different views. Unlike the views provided by our Internet Resource Mapping/Discovery Project, Prospero's views do not support flat searches.

HNS and KIS

Netfind's use of information where it naturally resides is a principle carried forward from our earlier Heterogeneous Name Service work [Schwartz, Zahorjan & Notkin 1987]. However, netfind uses much more decentralized information than the HNS did. The HNS was essentially a framework for supporting users in specifying the semantic operations needed to incorporate new auxiliary database-style name services into a global name service. Moreover, the HNS was used for mapping named objects to data about those objects (such as the network address of a host), rather than for discovering resources. More recently, Droms used an architecture similar to the HNS in his Knowbot Information Service, to provide a white pages service [Droms 1990].

Archie

The Archie project maintains a list of over 900 anonymous FTP archive sites worldwide, and retrieves a recursive directory listing at each site once per month. The data are made available to Internet users on a server that can be searched using regular expressions [Emtage 1991]. While it provides a very useful service, it has no architectural support that will allow it to scale indefinitely, as the number of resources, FTP archive sites, and users increase. (There are manually replicated servers to help distribute the load.) Moreover, Archie does not separate the class discovery and instance location problems. Users must sift through irrelevant data in many cases, and make ad hoc decisions about which file copies are most appropriate to retrieve (see Section 2). Nonetheless, it is a fascinating example of the potential for improving distributed collaboration in the global Internet.

Hypertext and Digital Library Systems

The Telesophy system provides a hypertext medium for information shared by a community [Schatz & Caplinger 1989], albeit of considerably smaller scale than the global collection of network accessible resources. There are also a number of efforts devoted to the problems of supporting access to a digital library systems,

intended to provide access to a range of documents [Kahn & Cerf 1988, Lynch 1990].

5. Conclusions

Resource discovery encompasses a range of problems that confront users of wide area networks in realizing the potential for remote collaboration and resource sharing. In addition to discovering the existence of a resource of interest, users must locate appropriate instances and keep track of information according to personal organizational preferences. Beyond these problems, resource discovery can also support network and system management, network integration and dynamic host configuration. More generally, we see resource discovery as a paradigm to support network-based collaboration and other types of wide area distributed applications [Schwartz & Tsirigotis 1991b].

The Networked Resource Discovery Project is exploring a number of approaches to these problems. We focus on techniques appropriate for global networks, and the concomitant issues of scalability, administrative decentralization, and organizational flexibility. Our efforts involve wide area distributed prototypes and measurement studies, focusing particularly on the global TCP/IP Internet as an experimental testbed. The projects discussed in this paper each address several of the problems described above, as illustrated in Table 5.

Project	Issue					
	Class Discovery	Instance Location	Information Organization	Network and System Mgmt.	Information Dissemination	Loose Coupling in Admin. Decen. Envs.
Internet "White Pages" (Netfind)		X				X
Probabilistic "Yellow Pages"	X	X	X		X	
Internet Resource Mapping/Discovery	X	X	X		X	X
Network Visualization			X			X
Electronic Mail Study	X	X	X			
Demand Distribution				X		
Internet Service Reachability Measurements						X

Table 5: Focal Issues of Resource Discovery Subprojects

Supporting resource discovery raises some difficult issues concerning privacy of information. While security mechanisms may be imposed to preserve privacy in some cases, in many cases such mechanisms are either difficult to provide, or of questionable merit. We believe that privacy is essentially a social issue, and as such requires careful consideration about the policies that will manage the technical solutions, in addition to technical solutions to security problems. Moreover, we believe that the best way to understand the tension between privacy and resource discovery is to explore the issues raised by building and deploying resource discovery prototypes.

Acknowledgements

I would like to thank the students who have contributed to this project: Roger Bacalzo, Kris Farnes, David Goldstein, Darren Hardy, Trent Hein, Bill Heinzman, Glen Hirschowitz, Darren Kalmbach, Rich Neves, Gregory Schaefer, Mike Smith, Stephen Strebel, Panos Tsirigotis, Alex Waterman, and David Wood.

A preliminary version of this paper appeared in [Schwartz 1991b].

This material is based upon work supported in part by the National Science Foundation under grants DCR-8420944 and NCR-9105372, a grant from Sun Microsystems' Collaborative Research Program, and a grant from AT&T Bell Laboratories.

6. Bibliography

Papers about the Networked Resource Discovery Project are available by anonymous FTP from latour.cs.colorado.edu, in the directory pub/RD.Papers. The author of this paper can be reached by electronic mail at schwartz@cs.colorado.edu.

[Bowman, Peterson & Yeatts 1990]
M. Bowman, L. L. Peterson and A. Yeatts. Univers: An Attribute-Based Name Server. *Software — Practice & Experience*, 20(4), pp. 403-424, Apr. 1990.

[CCITT 1988]
CCITT. The Directory, Part 1: Overview of Concepts, Models and Services. ISO DIS 9594-1, CCITT, Gloucester, England, Dec. 1988. Draft Recommendation X.500.

[Callimahos 1989]
L. D. Callimahos. *Traffic Analysis and the Zendian Problem.* Aegean Park Press, Laguna Hills, CA, 1989.

[Case et al. 1989]
J. Case, M. Fedor, M. Schoffstall and C. Davin. A Simple Network Management Protocol (SNMP). Req. For Com. 1098, Apr. 1989.

[Cheriton & Deering 1990]
D. R. Cheriton and S. E. Deering. Multicast Routing in Datagram Internetworks and Extended LANs. *ACM Trans. Comput. Syst.*, 8(2), pp. 85-110, May 1990.

[Clark, Lambert & Zhang]
D. Clark, M. Lambert and L. Zhang. NETBLT: A Bulk Data Transfer Protocol. Req. For Com. 998.

[Droms 1990]
R. E. Droms. Access to Heterogeneous Directory Services. Proc. 9th Joint Conf. of IEEE Computer and Communications Societies (InfoCom), June 1990.

[Emtage 1991]
A. Emtage. Personal Communication. McGill Univ., Montreal, Canada, Mar. 1991. Electronic bulletin board posting on comp.archives about version 2.0 of Archie anonymous FTP site directory server.

[Gifford & Spector 1984]
D. Gifford and A. Spector. The TWA Reservation System. *Commun. ACM*, 27(7), pp. 650-665, July 1984.

[Greene 1988]
H. H. Greene. United States of America, Plaintiff, v. Western Electric Company, Inc., et al., Defendants. Civil Action No. 82-0192, U.S. District Court, District of Columbia, Mar. 1988. Triennial review of Modified Final Judgement divesting AT&T of the Regional Bell Operating Companies.

[Harrenstien, Stahl & Feinler 1985]
K. Harrenstien, M. Stahl and E. Feinler. NICName/Whois. Req. For Com. 954, Oct. 1985.

[Hedrick 1988]
C. Hedrick. Routing Information Protocol. Req. For Com. 1058, Rutgers Univ., June 1988.

[Jacobson 1988]
V. Jacobson. Congestion Avoidance and Control. *Proc. ACM SIGCOMM Symp.*, pp. 314-329, Stanford Univ., Stanford, CA, Aug. 1988.

[Kahin 1990]
B. Kahin, editor. Commercialization of the Internet Summary Report. Req. For Com. 1192, Harvard Univ., Nov. 1990.

[Kahn & Cerf 1988]
R. E. Kahn and V. G. Cerf. *The Digital Library Project - Volume 1: The World of Knowbots.* Corp. for National Research Initiatives, Mar. 1988.

[Kantor & Lapsley 1986]
B. Kantor and P. Lapsley. Network News Transfer Protocol - A Proposed Standard for the Stream-Based

Transmission of News. Req. For Com. 977, U.C. San Diego and U.C. Berkeley, Feb. 1986.

[Lynch 1990]
C. A. Lynch. Information Retrieval as a Network Application. *Library Hi Tech*, 32(4), pp. 57-72, 1990.

[Malamud 1991]
C. Malamud. The ITU Adopts New Meta-Standard: Open Access. *ConneXions - The Interoperability Report*, Interop, Inc., 1991. To appear.

[Mockapetris 1987]
P. Mockapetris. Domain Names - Concepts and Facilities. Req. For Com. 1034, USC Information Sci. Institute, Nov. 1987.

[NSF Network Service Center 1989]
NSF Network Service Center. NSF Network News. July 1989.

[Neuman 1990]
B. C. Neuman. The Virtual System Model: A Scalable Approach to Organizing Large Systems. Tech. Rep. 90-05-01, Comput. Sci. Dept., Univ. Washington, Seattle, WA, May 1990. Ph.D. thesis proposal.

[Ordille 1991]
J. Ordille. Nomenclator Descriptive Query Optimization in Large X.500 Environments. *Proc. SIGCOMM Symp.*, Zurich, Switzerland, Sep. 1991.

[Peterson 1988]
L. L. Peterson. The Profile Naming Service. *ACM Trans. Comput. Syst.*, 6(4), pp. 341-364, Nov. 1988.

[Plummer 1982]
D. C. Plummer. An Ethernet Address Resolution Protocol -- Or -- Converting Network Protocol Addresses to 48.bit Ethernet Address for Transmission on Ethernet Hardware. Req. For Com. 826, Nov. 1982.

[Postel 1980]
J. Postel. User Datagram Protocol. Req. For Com. 768, USC Information Sci. Institute, Aug. 1980.

[Postel 1981]
J. Postel. Internet Control Message Protocol. Req. For Com. 792, USC Information Sci. Institute, Sep. 1981.

[Postel 1982]
J. B. Postel. Simple Mail Transfer Protocol. Req. For Com. 821, USC Information Sci. Institute, Aug. 1982.

[Postel & Reynolds 1985]
J. Postel and J. Reynolds. File Transfer Protocol (FTP). Req. For Com. 959, USC Information Sci. Institute, Oct. 1985.

[Quarterman & Hoskins 1986]
J. S. Quarterman and J. C. Hoskins. Notable Computer Networks. *Commun. ACM*, 23(10), pp. 932-971, Oct. 1986.

[Rose & Schoffstall 1989]
M. T. Rose and M. L. Schoffstall. An Introduction to a NYSERNet White Pages Pilot Project. Tech. Rep., NYSERNet Inc., Dec. 1989.

[Salton 1986]
G. Salton. Another Look at Automatic Text-Retrieval Systems. *Commun. ACM*, 29(7), pp. 648-656, July 1986.

[Schatz & Caplinger 1989]
B. R. Schatz and M. Caplinger. Searching in a Hyperlibrary. *Proc. 5th IEEE Int. Conf. Data Eng.*, pp. 188-197, Feb. 1989.

[Schwartz, Zahorjan & Notkin 1987]
M. F. Schwartz, J. Zahorjan and D. Notkin. A Name Service for Evolving, Heterogeneous Systems. *Proc. 11th ACM Symp. Operating Syst. Prin.*, pp. 52-62, Nov. 1987.

[Schwartz 1988]
M. F. Schwartz. Autonomy vs. Interdependence in the Networked Resource Discovery Project. Position paper, ACM SIGOPS European Workshop, Cambridge, England, Sep. 1988.

[Schwartz 1989]
M. F. Schwartz. The Networked Resource Discovery Project. *Proc. IFIP XI World Congress*, pp. 827-832, San Francisco, CA, Aug. 1989.

[Schwartz 1990]
M. F. Schwartz. A Scalable, Non-Hierarchical Resource Discovery Mechanism Based on Probabilistic

Protocols. Tech. Rep. CU-CS-474-90, Dept. Comput. Sci., Univ. Colorado, Boulder, CO, June 1990. Submitted for publication.

[Schwartz 1991a]
M. F. Schwartz. A Measurement Study of Changes in Service-Level Reachability in the Global TCP/IP Internet: Goals, Experimental Design, Implementation, and Policy Considerations. Tech. Rep. CU-CS-551-91, Dept. Comput. Sci., Univ. Colorado, Boulder, CO, Oct. 1991.

[Schwartz & Tsirigotis 1991a]
M. F. Schwartz and P. G. Tsirigotis. Experience with a Semantically Cognizant Internet White Pages Directory Tool. *J. Internetworking: Research and Experience*, 2(1), pp. 23-50, Mar. 1991.

[Schwartz & Wood 1991]
M. F. Schwartz and D. C. M. Wood. A Measurement Study of Organizational Properties in the Global Electronic Mail Community. Tech. Rep. CU-CS-482-90, Dept. Comput. Sci., Univ. Colorado, Boulder, CO, Aug. 1990; Revised July 1991. Submitted for publication.

[Schwartz et al. 1991a]
M. F. Schwartz, D. H. Goldstein, R. K. Neves and D. C. M. Wood. An Architecture for Discovering and Visualizing Characteristics of Large Internets. Tech. Rep. CU-CS-520-91, Dept. Comput. Sci., Univ. Colorado, Boulder, CO, Feb. 1991. Submitted for publication.

[Schwartz & Schaefer 1991]
M. F. Schwartz and G. W. Schaefer. A Bulk Data Transfer Protocol Optimized for Low Loss, High Delay-Bandwidth Networks. In preparation, 1991.

[Schwartz et al. 1991b]
M. F. Schwartz, D. R. Hardy, W. K. Heinzman and G. Hirschowitz. Supporting Resource Discovery Among Public Internet Archives Using a Spectrum of Information Quality. *Proc. 11th IEEE Int. Conf. Distrib. Comput. Syst.*, pp. 82-89, May 1991.

[Schwartz 1991b]
M. F. Schwartz. Resource Discovery and Related Research at the University of Colorado. *ConneXions - The Interoperability Report*, pp. 12-20, Interop, Inc., May 1991.

[Schwartz & Tsirigotis 1991b]
M. F. Schwartz and P. G. Tsirigotis. Techniques for Supporting Wide Area Distributed Applications. Tech. Rep. CU-CS-519-91, Dept. Comput. Sci., Univ. Colorado, Boulder, CO, Feb. 1991; Revised Aug. 1991. Submitted for publication.

[Schwartz 1991c]
M. F. Schwartz. The Role of Resource Discovery in Support of a National Software Exchange. Position paper, RIACS National Software Exchange Workshop, Mar. 1991.

[Travers & Milgram 1969]
J. Travers and S. Milgram. An Experimental Study of the Small World Problem. *Sociomety*, 32(4), pp. 425-443, 1969.

[Zimmerman 1990]
D. Zimmerman. The Finger User Information Protocol. Req. For Com. 1194, Center for Discrete Mathematics and Theoretical Computer Science, Nov. 1990.