Retrieving Highly Dynamic, Widely Distributed Information

M F Wyle, H P Frei
Department of Computer Science
Swiss Federal Institute of
Technology (ETH) Zurich, Switzerland
wyle@inf.ethz.ch

Abstract: Wide area networks provide a variety of information sources which can be exploited only by appropriate information retrieval techniques such as repeated automatic query of remote databases and bulletin boards. Distinctive features of the content and access methods of information on wide area nets are discussed from an IR perspective. The development, algorithms, and analysis of a functioning system are also presented.

1. Introduction

1.1. Wide Area Networks (WANs)

Modern advances in computer communications technology have spawned the growth of wide area networks (WANs). WANs link thousands of computers together, providing a distributed computer-mediated communications system to hundreds of thousands of users. The information is delivered quickly over large geographical distances; information sources for retrieval are no longer limited to a single machine or local area network (LAN). Researchers wanted, and have now created, a communications medium which is faster and somewhat less formal than journals or conferences [1]. New challenges in retrieving information are emerging as a result of the unique nature of WAN systems and their data.

Research in the retrieval of large amounts of textual information is well established [6, 11]. In contrast to the predominantly static, well-organized nature of such information, the information available on WANs is dynamic, timely, often "noisy" and thus difficult to index, or filter. Data formats, indexes, archive and query methods differ among systems on the same network. The topology of WANs is also dynamic; new machines join the network and member machines drop off of the network continuously. Implementing a mechanism to query an entire WAN is therefore problematic. Much of the information on these networks is publicly available for query, but not actively disseminated to member sites.

Quarterman and Hoskins define a notable network [5] to be one which provides at least mail or news service to its users and interconnects to other networks which provide such services. Notable WANs span the globe, with nodes on most continents and gateways into one another. We shall concentrate on the information content of such networks and the related retrieval aspects. We shall further restrict ourselves to a discussion of networks used by the research community as opposed to commercial services or "Value Added Networks" (VANs), although some principles discussed are applicable to both.

WANs have the interesting quality that the component systems from which they are formed vary widely in hardware, operating system software, and connection media. Even though the real number of nodes on a WAN varies, a typical one might have several thousand leaf nodes, hundreds of intermediate sites, and tens of main network traffic carriers.

1.2. WAN Abstraction Layers

A WAN is an extremely abstract entity. There are many layers of hardware and software between a user and the network. Different users have completely different concepts of what constitutes a WAN.

We are not concerned with the lower levels of WAN technology. Instead, our discussion assumes three higher abstraction layers, namely distribution programs, network management software and user-interface software. Distribution programs handle network connectivity maps, network routing, program scheduling, etc. The management software logs statistics, creates and removes network news categories, deletes old news information, changes routing tables, and so on. Finally the user interface software allows users to read information from and submit information to a WAN.

The distribution programs include the so-called Message-Transfer Agents (MTAs) of electronic mail systems, network news transfer software, some remote execution programs, and database software for maintaining connectivity, routing, and scheduling information. It is at this level of abstraction that a WAN ceases to be a conglomeration of many different software systems, and starts to appear as a single abstract entity.

The management software in WANs automatically creates and removes data files on the component systems, and responds to commands issued from any system on the WAN. Many cumbersome tasks are completely automatic, requiring almost no human intervention [1, 8].

1.3. Access Methods in WANs

There are three basic methods by which information is exchanged on WANs. Electronic mail allows users to compose and distribute messages to individuals or groups of WAN users. Bulletin-boards provide the facility for users to send and access messages submitted to the entire WAN. On-line conferences give WAN users the ability to "meet," in real-time, to ask questions, brain-storm, and share technical data. Such sessions can be archived and distributed later to other interested network users either by bulletin-board or electronic mailing lists.

People who use such networks gain access by virtue of having access to a WAN. The majority of users with accounts on time-sharing computers attached to WANs do not use any WAN services at all, either because of ignorance, WAN information overload, or lack of interest.

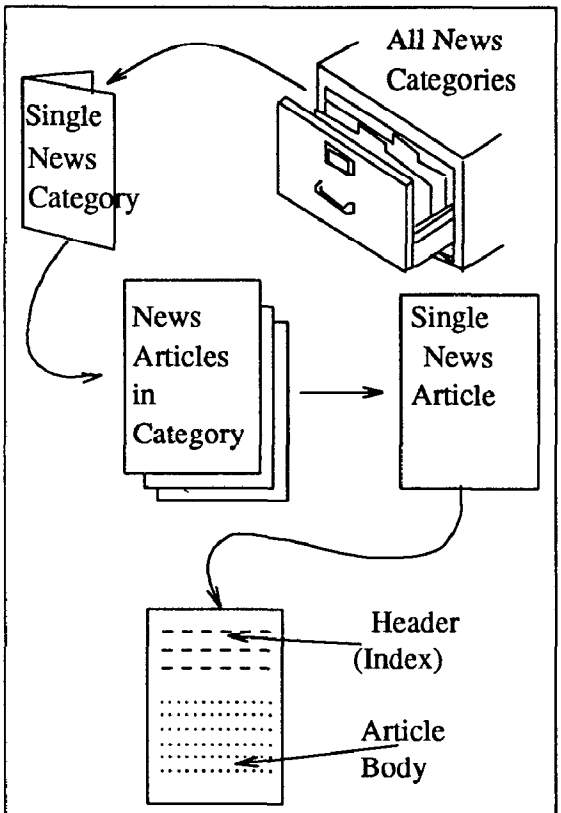

Figure 1 News Article Organization

A user's view of a WAN is the user-interface provided by an electronic mail, news-reading program, and / or conferencing system. These programs are normally lacking in the sophisticated "search" features of traditional document retrieval systems because they are predominantly designed with the assumption that the user wants to examine each and every item delivered by the system. Some simple stop list filtering methods have been developed.

These programs also provide methods of sending mail or news items over the WAN. A helpful analogy is to consider a collection of news items as an electronic magazine. Some magazines are edited by a moderator who collects articles via electronic mail, and sends the collections to a bulletin board or mailing list. Others are completely unregulated such that anyone can send information to every subscriber without editorial intervention.

1.4. Bulletin Boards and Mail Accessible Databases

The current standard for data transmitted on WANs is 7-bit ASCII text. At the user application level, digitized images, diagrams, sound, executable programs, and other binary data can be encoded into text, transmitted, and later decoded into their original form at the receivers' machine(s) [12]. For indexing and retrieval purposes, we assume that all WAN data are text.

In WAN terminology, one refers to a unit information item as a *message*. It is our thesis that IR principles used for automatic indexing in IR systems can be successfully applied to WAN messages. Just as a document often has a formal bibliographical description, WAN messages have "headers." Message headers contain meta-information such as transmission date, as well as descriptive information such as subject. Most WAN messages headers adhere to a standard categorization scheme to assist recipients in access [3]. However, this standard assists only as far as the formal part of a message (the header) is concerned. Retrieving on content is hardly ever supported.

We shall refer to network news and distributed bulletin board messages synonymously. This type of message has a very large distribution and contains many different indexing fields. Network news messages are always indexed by the author, which is the reason that many indexing fields are either left blank, or contain information of little utility. The date, group and subject fields are never left blank; they are used as the basis for retrieval by most E-mail User Agents and news reading software.

The major taxonomy scheme of network news articles is the group indexing field. News groups are arbitrarily categorized in a tree-structure, whose contents reflect the interests of the users who read and submit articles. Administrators of the news software decide on changes to the structure, and also which news groups their machines will accept and pass along to their neighbors on the WAN. Each machine, local area, or community may have local news groups whose distribution on the WAN is limited. Users specify the distribution scope of their messages when they are submitted. Figure 1 illustrates the news article organization scheme for network news [1].

In addition to mail, conferences, and distributed bulletin board services, WANs provide electronic mail-accessible databases. Such systems accept queries from remote machines via electronic mail and serve as depositories for many different kinds of information. Source code and binaries of application programs, mailing lists, bibliographical references, conference protocols, and other information are all available to WAN users by sending queries to a program at a remote site [2].

1.5. Heterogeneous Information Sources

It is important to note the differences between information and services available on WANs and those on direct-access databases and IR systems. The literature in IR and distributed databases [7, 10] addresses issues in a *cooperating* network of systems, or systems in which direct, fast communications to a few (less than 100) systems are available. WANs contain several orders of magnitude more systems, users and sources. The information available is subject to a high rate of change. There are very few direct connections, and the message delays are longer than a few seconds. The issues in which we are concerned here, namely extraction, filtering, and selective dissemination of WAN information, are quite unique and not well covered by the literature to date.

We have developed software for the query, extraction, indexation, selection, archival, and dissemination of WAN-based information. The goal of our system is to extend the utility of WAN services via periodic, automated retrieval, indexing, and filtering of WAN data. We hope to provide a service which amplifies the utility of WAN information and which provides a more comfortable access to the information.

2. WAN Information Models

2.1. WAN Information Variety

Traditional models of distributed systems, distributed databases, and communications systems are at a different level of abstraction than that required for the analysis of WAN information flow and retrieval. However, we can expand on some of the basic principles from these research areas to model information on WANs.

As we noted in the last section, WAN information is unique in its volume, distribution, audience, and duration. First of all, the information is dynamic; millions of items arrive daily at many thousands of computers. Secondly, the sources and topics of WAN information items are diverse; anyone with access to a WAN can send information to the entire network, and the research community served by WANs is large. A third important new aspect of WAN information is speed; information is disseminated quite quickly. Finally, as a result of this speed, WANs are often used to disseminate information which are valid or interesting only for a short time; such items make unique demands on a WAN information retrieval system. How can one successfully use this new variety of information?

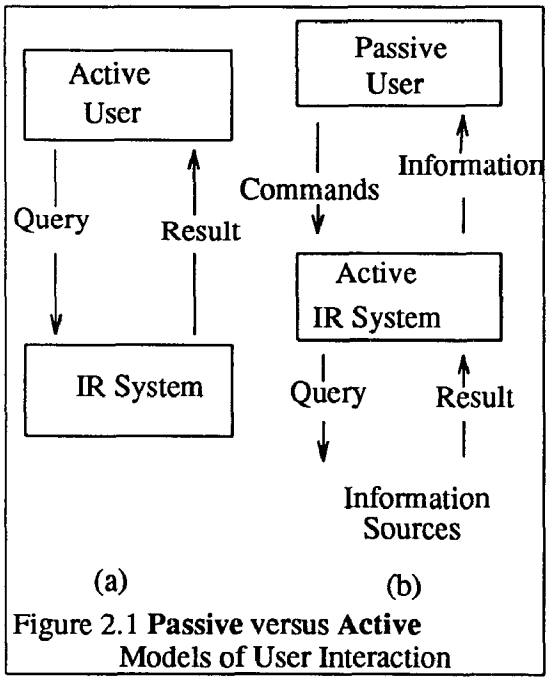

Figure 2.1 **Passive versus Active** Models of User Interaction

2.2. Passive User Model

Our research is based on the assumption that this new form of dynamic information is (or will become) too voluminous to sift manually. The system which we have developed (see section 4) therefore assumes a *passive* user, and a sophisticated, *active* software system which performs routine and algorithmic information gathering, filtering and dissemination. The filtering task is accomplished by the application of modern information retrieval methods such as statistical text analysis, semantic indexing and information trace comparison.

In contrast to the traditional (active) user model, our passive user model (figure 2.1) contains an intermediary system which queries several information sources. The system accepts commands from the user, stores these commands in the sense of an interest profile, and automatically supplies information to the user. For this reason, the system periodically queries information sources, then sends the retrieved information to the user(s) whose profile(s) match the new information items. In this model, the user *passively* receives information from the system, without having to re-submit queries periodically.

Obviously, sophisticated methods for matching a user interest profile with in-coming WAN information are required. User profiles must be well-formulated and periodically updated. Additionally, methods for automatic, repeated query generation must be developed in order to get the desired information to users of the system in a timely fashion. Finally, such a highly automatic system should be robust, and capable of recovering from network failures. Algorithms for error recovery are therefore critical.

2.3. WAN Database Query Scheduling

Part of our information server is an extraction system called "Selective Extractor of Information" (SEI), which automatically and repeatedly issues queries to WAN database servers such as [2]. In order to schedule the queries sent by SEI to the different WAN database servers, we must take several factors into consideration:

- The rate that new information arrives at the remote server;
- The delay between issuing a query and obtaining the result;
- The probability that a query message or its result message will be lost.

Let us assume that information flows from different sources to a WAN database server.
An optimal query schedule will allow the extractor program to obtain information at the same rate as it arrives at the server. Obviously some delay is inevitable since the source information is not instantly available. The rate of information *flow*, however, should be approximately the same over some time interval. Querying too often would result in many empty or redundant result answers. Querying too infrequently would result in large, infrequent "bursts" of information at the extractor, and large intervals of time where the extractor's information are out of date. Since WAN communications bandwidth can be quite expensive, and since the clients of every Selective Dissemination of Information (SDI) service prefer current information, both of these situations are unacceptable.

In order to quantify this model somewhat, we shall define some values which can be used in a generalized query scheduling algorithm.

Information flows from different sources to a server. Although the arrival rate is important, from the extractor's point of view, a more important parameter is the rate that the information becomes available for querying. We shall call this *update rate* Λ and define it to be the number of information items which the server makes available for query per unit time. This rate may indeed be different from the arrival rate, but since the extractor has no control over the indexing process this parameter must take precedence. Thus, from the Extractor's perspective we shall take a "black box" approach.

"Server" here refers to a remote, passive WAN database system. A database server should not to be confused with our information server!

Consider the relationship between Λ and query scheduling. Let's call the query rate Γ and give it units of queries per unit time. An ideal query rate would, on average, retrieve precisely one information item per query. If more than one information item is retrieved, the rate is too slow. If no information items are retrieved, the rate is too fast. Since the extractor has no control or knowledge about the update rate at the remote server site, the extractor system should adjust Γ using an algorithm which somehow predicts the value of Λ at each moment in time.

A functioning extractor which has logged some statistics has the advantage of knowing some past values of Λ and Γ and exact times it issued queries. Using this history information, a future value of Λ can be predicted, and a new Γ can be calculated:

$$\Gamma = \frac{\Lambda}{I}, \qquad (2.1)$$

$$\text{where } I = \frac{\text{number of information items}}{\text{query}}$$

Information items are often queued at a server and updated in "bursts;" in this case, it is impossible for a single query to return exactly one information item.

2.4. Accounting for Network Failures

This analysis obviously assumes that no anomalies or errors occur. Unfortunately experience indicates that such assumptions are never the case. Network failures are far too frequent to ignore; we are therefore forced to add factors for error recovery. Consider the space-time diagram in figure 2.3. The information sources are grouped into one abstract entity on the right. Time flows from the bottom to the top, and events are shown as circles inscribed on the entity lines at which they occur. Source events S1 and S2 travel to the Server where the information items are updated by Update events (U1, U2).

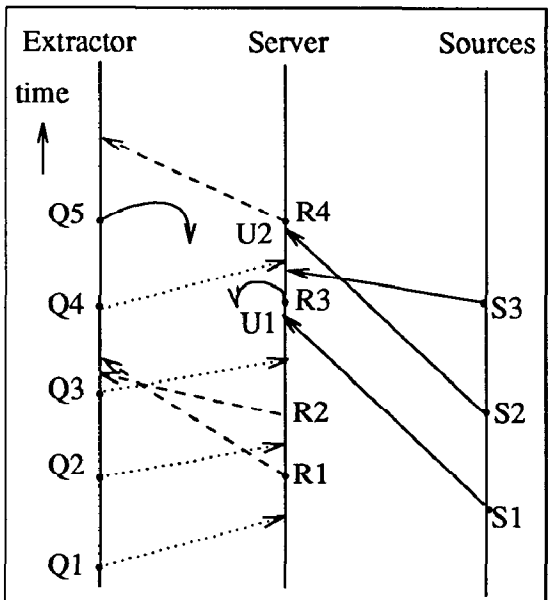

Figure 2.3 Space-Time Diagram

Because of the nature of computer networks, one cannot assume uniform response times or transmission rates. Notice that the result (R1) of the first query (Q1) arrives *after* the second and third queries (Q2, Q3) have been sent. It's transmission rate is so slow that it arrives after the second result R2 has arrived at the Extractor. Note also that query Q5 never arrives at the server; it is lost in the WAN. The same fate awaits the transmission of result R3.

In order to query effectively, we are forced to add some form of fault-tolerance to the scheduling algorithm. Let us assume that the probability of obtaining an answer to a query is ρ. This value represents the combination of the individual probabilities that the query arrives, that the result is generated, and that the result arrives. The effective flow of information from the Server to the Extractor is therefore smaller by a factor of ρ:

$$\Gamma = \rho \frac{\Lambda}{I} \qquad (2.2)$$

We are now tasked with defining ρ and Λ in some manner which will allow Γ to adjust to changes in failure and update rates. The algorithm should not allow very sudden changes in Γ, because a positive feedback cycle of large positive and negative changes in the query rate would make the entire analysis useless. An algorithm providing some kind of dampening or smoothing is therefore prefered.

For a given time t_i, ρ_i is defined as:

$$\rho_i = \begin{cases} 1 & \text{if response } R_i \text{ arrives for query } Q_i \\ 0 & \text{if response } R_i \text{ does not arrive for query } Q_i \end{cases} \qquad (2.3)$$

This result must of course be calculated after the fact. That is, ρ_i cannot be determined until a query is sent and either an answer arrives, or a timeout interval has expired. However, we need an average probability; we shall therefore define $\bar{P}$ to be the *average* probability of successful query - answer message pair transactions over some period of time. That is,

$$\bar{P} = \frac{1}{n}\sum_{i=1}^{n}\rho_i \qquad (2.4)$$

where n is some number of query transactions. One approach at this point would be to assume random behavior, maintain a running average, and re-calculate the average each time a transaction is completed. However, our preliminary statistics indicate that WAN failures may *not* be random. For this reason, we prefer a calculation which gives more precedence to the success or failure of recent transactions. We therefore propose the use of a so-called *moving* average which ignores transactions older than a specific age. We define the "window" of transactions w to be the number of transactions before the most recent transaction which should be considered. The windowed average probability of success $\bar{P}_w$ is then calculated as:

$$\bar{P}_w = \frac{1}{w}\sum_{i=n-w+1}^{n}\rho_i \qquad (2.5)$$

The *gain* of this adaptive probability calculation is defined by the window size w. A large w will cause the system to react more slowly to changes in the failure rate of the WAN. A smaller w makes the system more adaptive. One definition of the optimal choice for the window size can be derived by comparing predicted probabilities of events with the actual events for all possible values of w ranging over some time intervals $1 \rightarrow w, 2 \rightarrow w+1, \ldots, n-w+1 \rightarrow n$. The optimal value of w produces probabilities which on average predict the next event the best.

We define the following error-function G_w which computes the "goodness" of w by summing the differences between predicted and actual arrival probabilities over an interval of n events.

$$G_w = \left\{ \sum_{j=x}^{n} \left| \left[\frac{1}{w} \sum_{i=j-x+1}^{j} \rho_i \right] - \rho_{j+1} \right| \right\} \qquad (2.6)$$

The best w is the largest interval for which G_w is a minimum in the range of $1,2,...,n$.

If WAN failure probabilities are random as most computer network literature assumes, the optimum window choice size would be n. Re-calculating and varying w as new failure statistics arrive allows our algorithm to be more adaptive to short-term trends in WAN failure rates. This analysis is one method by which we attempt to optimize Γ.

The following plots illustrate graphically the advantage of using an adaptive scheduling algorithm, as opposed to one based on a simple average. These plots are actual data from approximately 500 WAN queries which were sent over a period of two months by the SEI WAN extraction program.

Actual ρ and $\bar{P}$

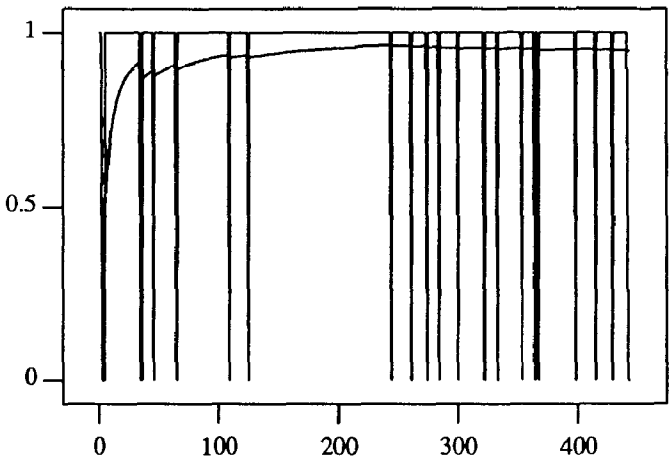

Figure 2.4a

Window Average ρ_w

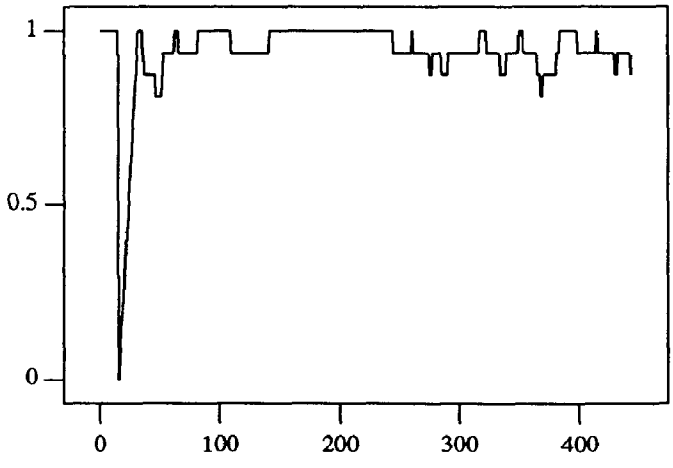

Figure 2.4b

2.5. Predicting Information Item Arrival Rate

An approximation of Λ is calculated indirectly by counting the new information items which are known to have arrived at the server during a given time interval. After a query is sent, if an answer arrives back, it will contain 0 or more information items. If we call the arrival time of answer i t_i, and the number of items contained in answer i I_i then Λ is simply:

$$\Lambda = \frac{I_i}{t_i - t_{i-1}} \qquad (2.7)$$

Given some time_interval, we compute the arrival rate of information items, Λ and define it as:

$$\bar{\Lambda} = \frac{1}{n} \sum_{i=1}^{n} \frac{I_i}{t_i - t_{i-1}} \qquad (2.8)$$

One might again be tempted to weigh the average according to more recent arrival rates, and perform a similar analysis for Λ as was done for w. In this case, however, the data are not binary probabilities, and we can glean more useful parameters by analyzing the *trend* of each successive Λ over time, instead of the average. Consider n successive values of Λ over some time interval $t_1 \le \tau \le t_n$. Each Λ_i versus t_i in the interval τ can be plotted as points in a curve. We can apply the generally accepted least-squares method of curve-fitting, which minimizes the square of the absolute deviation of each point from a calculated curve.

It is quite simple to fit many different kinds of curves to the data, including lines, polynomials, and exponentials. Our preliminary data indicate that the arrival rate Λ is best modeled by an exponential curve of the form:

$$\Lambda(t) = e^{a+bt} \qquad (2.9)$$

Obviously WAN information item arrival cannot maintain this exponential increase, but for the moment, we can best model the phenomenon by an exponential curve. If the rate of arriving information items changes then the model for Λ must be reconsidered.

The extraction part of our information server incorporates the analysis of both Γ and G_w into its query scheduling algorithm in order to maintain the best information flow with the fewest possible queries.

3. WAN Information Items

3.1. News Information

By far the majority of information items actively transmitted over WANs fall into the category of distributed bulletin board messages commonly called news items. These news items are usually viewed via one of a variety of news-reading programs, described in detail in [1].

In a WAN news system, the full-text of an information item is available immediately; broad categories are presented, and within these categories, articles are presented in chronological order. Per-category stop-lists or normal key-word searching may also be used, but the main purpose of news readers is "browsing" through recent articles, not searching for specific information.

With one key stroke, a person reading a news item can post a follow-up article directly to the author or to the *entire* network. This power causes problems because readers' ideas of what constitutes an appropriate message often clash. A large percentage of information is of limited use.

The topics covered by news information are diverse. Almost all areas of computer science and computer hardware have categories devoted to their discussion. These topics have the highest volume. Other major scientific topics which are well represented include several areas of mathematics, biology, astronomy, chemistry, physics, psychology, linguistics, and communications. Additionally, several recreational activities have news categories including music, games, books, film, and writing.

3.2. Bibliographical References

Among the more useful services provided by WANs is the timely, periodic posting of bibliographical references extracted from current printed journals. These postings adhere to a standard [4] format, enabling any node on the network to archive and retrieve them.

We have developed methods which extract and store these bibliographical references automatically, based on the indexing provided either by the journals or other sources. We have also developed text filters for updating and translating bibliographical references between storage formats.

3.3. Source Code and Executable Programs

Another useful information service provided by many WANs is the dissemination of source code and executable programs for a variety of computers. Altruistic programmers often send their programs over WANs to help other programmers with their problems. This software is then archived at many sites, and further reproduction or re-transmission is encouraged.

4. A WAN Information Server

4.1. Overview

A broad overview of the software we have thusfar implemented appears in figure 4.1; it illustrates the interactions and layers of abstraction in the system. Transport moves data between programs and machines on the WAN. News and E-mail use the WAN transport to communicate information to users and programs. The Archiver and Database Server provide both local and remote database operations. SEI is the Selective Extractor of Information, whose query scheduling models were discussed in section 2. The Indexor indexes text. Finally, the Information Server provides an SDI service using all of the underlying components.

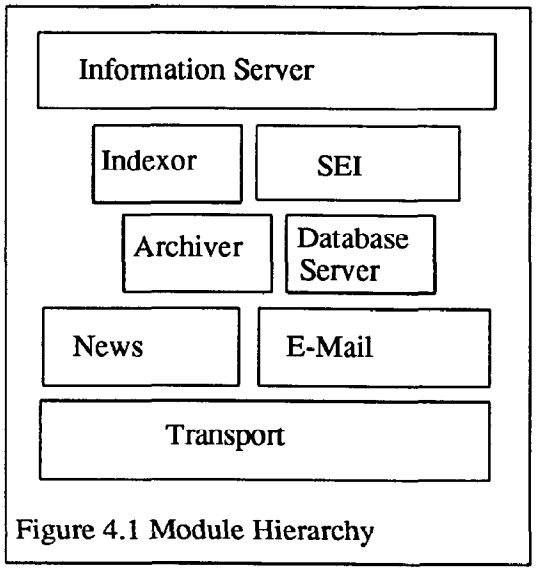

Figure 4.1 Module Hierarchy

Because of their communications medium, the modules are well-separated, autonomous entities; this autonomy allows the component programs to be on separate machines anywhere in a WAN. In practice, most of these modules run on machines in a local area network, but the software notices no logical difference between modules on the same machine and programs separated by large geographical distances. Module autonomy has the additional beneficial side effect of adding robustness to the system.

A subscriber's view of the Information Server is his electronic mail user agent. Subscribers send and receive E-mail to and from the program in exactly the same manner as they do with people.

4.2. Automatic Selection and Archival

The SEI module is a so-called "daemon" program which "wakes up" and runs periodically; it examines network news data which have arrived since the last time it was run, in order to decide if any of these newly-arrived items should be stored for use by other modules. Network News items are normally deleted automatically after a few days or weeks because so many new items arrive so quickly. SEI maintains a database of news-category-specific selection criteria, as well as some state and book keeping data for statistics and logs. Each network news article is examined in the order in which it arrives on SEI's system. When an article matches the selection criterion for its group, the program sends the entire article via electronic mail to the archival module or performs another category-specific action; for example, the article may be mailed directly to an individual, a different, remote archival system, or a completely separate program may be run with the article as input. When the program has finished processing all new news data, it schedules itself to run again at the next interval.

The archival program is invoked asynchronously by the electronic mail system; it searches for indexing fields in the received item, updates log files and indexes, and finally names, and stores the information item. The archiving program is quite robust; it ignores duplicate entries, and logs errors. Separating the extraction and archival process via electronic mail has many advantages, not least of which is that users can manually send their own data via E-mail to be archived from any local or remote machine.

The archived data are periodically compressed and moved to magnetic cassettes. Additional index information may also be added manually for those items which lack necessary indexing fields.

4.3. WAN Database Query

The automatic WAN database query part of SEI periodically queries several databases at remote sites, and maintains information about them. It tracks their indices, contents, and status. Because the remote database systems have widely varying command languages and query message requirements, this program is highly parameterized and data-driven. For each database tracked by this system, the message format, command syntax, E-mail address, query, and other special requirements are stored. The system is designed to allow for the flexible change, manipulation, and additions to these data, because of the dynamic nature of database servers on the WAN.

The following pseudo-code describes the general algorithm:

```
Get list of servers;
FOR each server DO
  Interval := Now - LastQuery;
  Calculate QueryInterval for this server;
  IF ( Interval >= QueryInterval ) THEN
    Look up address, command information in database;
    Form query;
    Send query;
    LastQuery := Now;
    Update accounting, log information;
  END IF
END FOR;
Reschedule ourself to run at next interval.
```

A separate process tracks the remote-site response times, and re-sends messages which have timed-out or were lost. Although E-mail is very convenient, it is not yet so reliable that we can depend on it completely.

When responses arrive back from the remote systems, they are processed by a message reception handler. Indexes of remote databases are compared with previous indexes. The differences are then analyzed for prospective new information which may then be extracted from the remote database.

4.4. SDI

We expect an SDI subscription service to be completed by the time this paper is published. This system maintains a list of users (or programs) and profiles of their interests. It then matches the in-coming local or remote database information against the interest profiles of each subscriber, and periodically sends E-mail to each subscriber with new information specific to his interests. Subscribers can specify short-term, long-term, and time-varying interests in their profiles via electronic mail, and the SDI system adjusts the profile data and match criteria according to user feedback.

User profiles in this system are collections of example messages which the user determines to be appropriate. Since these messages contain more than the few terms typical in most IR queries, more sophisticated methods of statistical text comparison become possible.

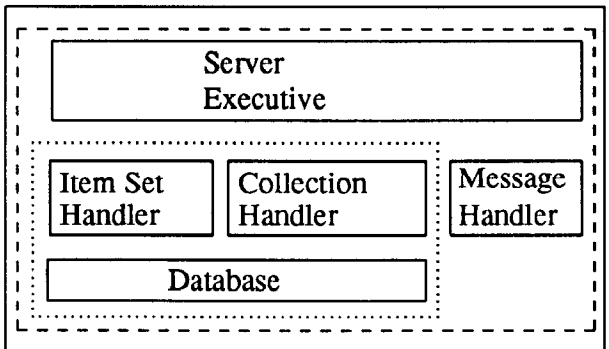

Figure 4.2 Information Server

The major SDI system components appear in figure 4.2. The Item Set Handler determines appropriate subsets of items in the collection for categorizing items of similar meaning by calculating similarities. The collection handler invokes the Indexor to index individual information items, including all retrieved information and user profiles. An N-gram based indexing module [9] was extended and adapted for use in this system. In particular, the text reduction components (stop list and stemming) had to be adapted to the WAN information because of the uncontrolled vocabulary and large noise content of WAN information. Terms in the WAN items with a negative discrimination are eliminated, which greatly reduces the text volume.

4.5. Some Statistics

The system currently operates on 55 MBytes of data; 6 MBytes of new data arrive daily, and 6 MBytes of old data are deleted. Twice daily, all 55 MBytes are "check-pointed," and the newly arrived items are indexed. The system processes text at a megabyte per minute and runs on a 5-MIPS Unix file server. It is in an "alpha" test stage with only 5 test subscribers.

We have verified that word frequencies in WAN data obey Zipf's law, despite the uncontrolled vocabulary and large noise content. The trigrams, and tetragrams from WAN data also obey Zipf's law. However, 93,538 distinct Tetragrams (20% of all possible), 17,202 trigrams (98% of all possible), and all 676 bigrams appear in WAN text as opposed to 3% of all tetragrams, 26% of all trigrams, and 90% of all bigrams in the INSPEC test collection [9].

The SEI system queries 14 separate database servers at an average of 4 queries per server per week. Returning messages arrive at approximately 50 per week and are processed as soon as they arrive, unless the Indexor is running, in which case they are queued for later processing. More answer messages arrive than the number of queries because of diagnostic E-mail messages. An average query result is 30 Kbytes. Approximately 96% of all queries are successful. The average query receives a reply within 83 minutes, but the variation is large (minimum 109 seconds, maximum 8.3 days).

The elimination of non-letters produces a 25% reduction. Elimination of one or two letter words not in the anti-stop list gives a 1% reduction. Elimination of words in a stop list of the 150 most frequent english words yields a 16% reduction. A modified Porter stemming algorithm [9] further reduces volume by 22%. Finally, the elimination of terms with a poor discrimination gives a 15% reduction, for a total reduction of 79%. Each of these reduction steps except discrimination executes faster than disk-access speed, allowing efficient "streaming" and "piping" of data.

5. Conclusions

5.1. Summary

How do we stay informed? We read newspapers, listen to radio, watch television, subscribe to magazines, talk to our friends and colleagues. Researchers attend conferences, read papers, and use libraries. Most of these information media are well-established and mature. Telephone and video often mediate personal communications, but conferences and research journals still dominate formal scientific idea interchange. Some relatively new forms of communications, namely computer-mediated communications systems offer some advantages to traditional media and are becoming widely available with the explosion of computer usage. We have developed some components of an SDI system which take advantage of wide-area network communications capabilities, namely news, electronic mail, and remote databases.

This research has so far centered upon the chaotic information flowing on the hundred thousand machines of the uucp, arpa, csnet, and bitnet research networks and network news systems. We have developed automatic archival software which selectively filters and archives news as it flows by. Additionally, we have developed models for the automatic, repeated query of WAN database servers and a software system which demonstrates the viability of our analysis.

The wide-area network based SDI system provides a subscription service via electronic mail. It gives its subscribers access to different types and sources of information. Traditional computer information retrieval systems require users actively to search for new data. Our system involves a passive user who is periodically informed when new data matching his interest profile becomes available. The system differs from classical SDI or clipping services in the speed, medium, and information it provides.

5.2. Future Direction

The user-interface is one of the most important aspects of any computer system. The wide-area network IR sub-systems presented here lack a modern, graphical user-interface. One direction of our future work will therefore be to add a convenient user-interface to the system.

Our selection and matching criteria are based on syntactic analysis and statistical text analysis for automatic indexing. We shall investigate whether more semantics can be added by using information structures for specific domains of discourse.

Finally, we must develop models along the lines indicated in section 2, and anticipate trends in wide-area network information retrieval. For example, subscribers' interests wander, and we hope to develop methods to change out-dated topics in their profiles automatically.

References

1. Anderson, Bart, Bryan Costales, and Harry Henderson, *Unix Communications*, Howard W. Sams and Company The Waite Group, Indianapolis, Indiana, 1987.

2. Dongarra, Jack J. and Eric Grosse, "Distribution of mathematical software via electronic mail," *Communications of the ACM*, vol. 30, no. 5, pp. 403-407, May 1987.

3. Kantor, Brian and Phil Lapsley, "Network News Transfer Protocol: A Proposed standard for the stream-based transmission of news," *Request for comments (RFC) 977*, Defense advanced research projects agency DARPA, February 1986.

4. Lesk, Michael, "refer - A bibliography system," in *Using nroff and troff on the Sun Workstation*, Sun Microsystems Inc., February 1988.

5. Quarterman, John S and Josiah C Hoskins, "Notable computer networks," *Communications of the ACM*, vol. 29, no. 10, pp. 932-971, October 1986.

6. Salton, G and M McGill, *Introduction to Modern Information Retrieval*, McGraw Hill, New York, 1983.

7. Smith, J M, P A Bernstein, U Dayal, N Goodman, T Landers, and K W Lin, "Multibase - Integrating Heterogeneous Distributed Database Systems," *AFIPS Conference Proceedings, National Computer Conference, Chicago*, vol. 50, AFIPS, Arlington, VA, 1981.

8. Tannenbaum, Andrew, *Computer networks*, Addison Weseley, August 1988.

9. Teufel, Bernd, "Informationsspuren zum numerischen und graphischen Vergleich von reduzierten natuerlichsprachlichen Texten," D. Sc. Theses ETH No. 8782, Zuerich, 1989.

10. Van-de-Riet, R P and W Litwin, "A General Framework for the Architecture of Distributed Databases," *Proceedings of the Second International Seminar of Distributed Data Sharing Systems*, North Holland, Amsterdam, Netherlands, 1982.

11. Wyle, Mitchell F and Edward Fox, "Annotated bibliography relating to automatic indexing in information retrieval," *SIGIR Forum ACM Special Interest Group on Information Retrieval*, vol. 21, 1-2, Fall Winter 1986/87.

12. Wyle, Mitchell F, "File Directory Transmission via Electronic Mail," *SIGUCCS ACM Special Interest Group of University Computing Center Services*, June 1988.