DISTRIBUTED DATABASES : THE CASE OF MULTIPLE SERVERS

H. E. Payne

*Space Telescope Science Institute, 3700 San Martin Drive, Baltimore, MD
21218, U.S.A.*

Received July 17, 1996.

Abstract. Tremendous volumes of astronomical data are available at sites scattered about the World Wide Web (WWW). We have become familiar with the client/server model for accessing this information. If we view these services as a single distributed resource with multiple servers, the task of information retrieval takes on some new problems, such as inter-operability, reliability and speed. I discuss three approaches to the problem of searching multiple servers, which differ to the extent in which the retrieval engine is centralized.

Key words: databases

1. INFORMATION RETRIEVAL OVER THE INTERNET

An enormous volume and variety of astronomical information is now available in electronic form via the Internet. Data, software and scientific literature are available. World data centers are providing improved access to data. New instruments are creating large amounts of new data. The Hubble Space Telescope Archive, for example, currently holds about 2.4 TB of data, and is growing at a rate of 0.7 TB/year. The instruments to be installed next year will increase this rate to 2–4 TB/year. Software of interest to astronomers is available as source code and binaries. Updates and support are also available over the Internet. We can now read the ApJ Letters over the Internet, and will soon be able to read the main journal (provided our libraries sign up for site licenses!). In the near future the bulk of astronomical refereed publications will be electronic. A vast array of preprints and abstracts to the older literature is available and growing. Many more examples in a variety of fields were

described at a conference on Information Systems Interoperability (see <http://nssdc.gsfc.nasa.gov/sisic/sisic.html>).

The problem for the consumer of this information is basically the same as that of anyone else on the Internet: "How do I find the material of interest to me?" The data centers have done a good job of solving this problem for their own holdings, with solutions based on the client/server model. I would like to discuss the larger discovery problem of finding the right server in a model where "distributed databases" means databases on some number of *independent* servers separated by significant stretches of the Internet. The problem is to create a level of inter-operability between information servers. The two basic requirements are: (1) a solution cannot be imposed, but must be the result of cooperation and (2) the amount of effort required from the information providers must be minimal.

The Internet is currently dealing with the discovery problem via strategies like that followed by Digital's AltaVista search engine: index all of the text in all of the files found by following all of the links, and hope that searching by keyword will return files of interest. This is the method that Heck & Murtagh (1993) contrast with "Intelligent Information Retrieval" (IIR). I think of it as brute force information retrieval. Inter-operability is achieved by reducing everything to text. The metadata to which these Internet search engines are sensitive is basically limited to the mime type associated with each file. Hardly any use is made of this information except to filter out non-text files and to identify the title tag in HTML documents. Still, research has shown that the best match search results based on the brute force indexing of the entire text of documents are just as good as attempts to represent text with the limited vocabulary of a thesaurus (Belkin 1993). So we can hope for good results when applying this method for example to preprints.

The extension of this scheme to data presents a number of problems. It is possible to index the metadata associated with a particular dataset, as the NASA Global Change Master Directory (see <http://gcmd.gsfc.nasa.gov/z3950/distrib-search.html>) has done, but it generally makes no sense to index the data themselves. And word searching through the metadata is a dubious proposition – even if the description contains the word UV, that is a lot less assuring than checking some piece of metadata describing waveband and finding its value to be "UV."

Astronomical datasets and databases on the Internet are associated with metadata written to some hard-won standards. We have,

for example, the NASA/Science Office of Standards and Technology (NOST) standards. The FITS standard has been very successful as a medium for information archiving and exchange. Standards for dataset and database descriptions include the NSSDC's Common Data Format (CDF) and Standard Formatted Data Unit (SFDU) formats. Our field is prepared for inter-operability at a much higher level of sophistication, but only the first tentative steps have been taken. I would now like to discuss three systems taking these first steps. They differ in the degree to which their components are distributed.

2. DISTRIBUTED DATA – CENTRALIZED INDEX

The first example is one that I have actually worked on, along with Bob Hanisch and Archie Warnock: the Astronomical Software Directory Service (ASDS, see <http://asds.stsci.edu/asds/>), intended as an aid for astronomers looking for software to solve a particular problem. We decided that we did not want to archive, or warehouse, the software, but rather to maintain a centralized index to an inherently distributed collection, maintained by the software providers at their sites. The texts to be retrieved from our system consist of software documentation and descriptions, although we have given some thought to expanding it to comments taken from source code. Either way, our “texts” are just text. The representation is currently limited to HTML files available over the World Wide Web (WWW). The only requirement for participation in this service is a Web server and relevant HTML files.

In the first prototype of ASDS, the “text surrogate” was a searchable index that was partly distributed and partly centralized. There are two basic approaches to indexing a collection of HTML documents: (1) using a Web crawler, starting at some point in your local Web, to generate a list of URLs to be retrieved and indexed, filtering out URLs outside your organization, or (2) using a directory search, starting at some root directory, to generate a list of files to be opened and indexed, filtering out file names, perhaps by file name extension. We chose CNIDR's Isite search engine, which falls into the second category. Isite has been intended as a replacement for CNIDR's freeWAIS. We chose it primarily for its ability to do fielded searches, along with some other attractive features like Boolean operations, right truncation (using “galax*” to search for both “galaxy” and

“galaxies”) and something called “virtual databases,” which allows you to search multiple databases as if they were a single database.

We modified Isite so that retrieving a URL was essentially equivalent to opening a file. That turns out to have been a big mistake. Search engines that expect their texts to be local do not bother storing information that can easily be recovered at search time by opening the file again. However, there is a very large performance penalty for going out onto the Internet at search time. In addition to the overheads and bandwidth limitations that simply slow things down, servers and name servers can go down making parts of the database completely unavailable in an unpredictable way. Even worse, the search algorithm is such that searches can fail completely if the documents changed between the times that the index was constructed and the query is performed. More than anything else, this fragility to changes in the document collection is responsible for the lack of success of the first ASDS prototype.

The NASA Global Change Master Directory uses the same modified Isite software to index dataset descriptions written in a format called Directory Interchange Format (DIF). The DIF files are spread across two machines, and the index is built on one of them, where the files can be opened directly. DIFs on the other machine are opened as URLs. While there is still some overhead associated with opening each URL, the long delays that can occur over significant hops across the Internet are avoided, and the overall performance is satisfactory. And since both machines are local in the sense that the index maintainers are the document maintainers, the index cannot get out of sync with the documents. The conclusions of the comparison with ASDS would include the unsurprising result that significant Internet traffic at search time is a serious mistake. If the index is centralized, it needs to contain enough information to support a full search and hence to be robust against changes in the original documents. This means that false hits and missed hits will always be a possibility. Reliable searching will require frequent re-indexing, or giving the remote document maintainers the possibility of initiating a re-indexing.

A new release of the Isite software is planned, which would be made suitable for our application by avoiding all Internet traffic at search time. I was hoping that we would have that release by this meeting, but that is not the case. When you visit our site, feel free to browse, but don't expect searches to work.

3. DISTRIBUTED INDEXES – CENTRALIZED QUERY INTERPRETER

My second example is the One-Shot World-Wide Preprint Search (<http://www.ictp.trieste.it/indexes/preprints.html>) recently introduced by the International Centre for Theoretical Physics (ICTP), linking major preprint services. I would like to thank Dr. Enrique Canessa for sending an early draft of a paper describing this service. Large, centralized, searchable electronic preprint services, like those created by Paul Ginsparg have been a tremendous aid in keeping up with the flood of preprints. Because they are conspicuous, these services attract many authors wanting to publicize their work. And because many authors are using them, they therefore become more useful to researchers at large. The difficulty that arises is that there are now a number preprint services – although this is currently more of a problem in the high-energy physics field than in astronomy. It is easy to become saturated with e-mail notification messages, and it becomes more tedious to search for preprints of interest since different services may have different user interfaces. The “One-Shot” project was designed to solve the search problem by accepting a query from a user, tailoring it to the separate requirements of a number of preprint services, submitting the customized queries and gathering the results to present to the user. The idea is to multiply the users’ efforts and to reduce the possibility of overlooking a useful archive.

Layered on top of the existing services, it distributes a query sequentially to the participating collections, collects the results and presents them to the user. In this model, the texts to be retrieved are represented by completely independent databases. The only requirement for participating in this system is that the databases have Web interfaces and a search engine. The difference from the previous model is that the search engine is now distributed. The internals of each database, like the search engine used, are unimportant. The user enters a search string query, the year(s) to be searched and the services to be queried. Advanced features available from the individual collections remain largely hidden from the user, in the interest of interoperability. A number of artificial limits on the number of databases to be searched and the breadth of each search (in time) have been imposed to prevent searches from taking too much time. At the moment, the ICTP has only a rather slow connection to the Internet, imposing a double penalty on the user: when the lookup re-

sults are collected in Trieste and again when they are shipped out to the user. The results are presented to the user as they are gathered, essentially unchanged.

This system was designed with sensitivity to the requirement of working over the Internet. Each system is “ping’ed” before being queried, to ensure that the combined network/host load is acceptable, for example. Searches are quite a bit slower than the search of a single service, but the robustness against the vagaries of traffic across the Internet is one of the strengths of this service. Since there are no centralized data, there is no synchronization problem. The individual services presumably keep their indexes up to date, so that if the services are available when accessed by the One-Shot service, then the correct results will be obtained. This is an advantage over the centralized index. On the other hand, the centralized index applies the same search algorithm to the entire collection, so that it is possible to find the best matching document in the collection, rather than a list of the best matches on each server.

Another drawback to this approach, likely to be exacerbated when searching for data sets instead of text, is that the search is reduced to a simple common-denominator search. In this case, use of the advanced capabilities of the search engines, like word grouping and Boolean operators, is severely restricted. In the case of data sets, the least common denominator will be much more sophisticated. It is going to be much more difficult to craft scripts to translate the generic query entered by the user into the service-specific queries to be sent out, at least as long as the individual services are using different interfaces and search engines. One might hope for a situation in which each service were described by metadata in a common format so that the central query exploder could mechanically translate the queries as required for each participating service.

4. DISTRIBUTED QUERY INTERPRETER – CENTRALIZED QUERY EXPLODER

The last system I would like to discuss is only in the planning stage. It is the AstroBrowse project, headed by Steve Murray. The basic idea is to do for data providers what the One-Shot Preprint search does for preprints. It is designed to attack the data location problem. (The difficult problem of the heterogeneity of the available data products is not addressed.) The scheme is to replicate a query given in some generic format, send it to participating servers, which

then translate the query into their own local format for execution. The difference from the previous scheme is that the query translation is done at each site rather than at the central location.

AstroBrowse has been the subject of two meetings of representatives of a number of data centers. Reports are available on the Web. The participants were interested in a system in which they could participate with relatively little effort, which would not detract from the specialized user interfaces giving full access to their services' features and which was tolerant of the differences between the services. They wanted to avoid a centralized system that would compel them to make changes in their services before they could participate. The major effort associated with the AstroBrowse project has been getting major data providers to understand each others services and to agree on the common-denominator query. The amount of effort devoted to cooperation between the parties is likely to dwarf the effort devoted to technical problems.

Table 1. AstroBrowse query components.

NAME	object name specification	DATA TYPE	image spectrum time series flux measurement visibility data
POSITION	(α, δ, r) or $(\alpha_{min}, \alpha_{max})$ $(\delta_{min}, \delta_{max})$. Positions should be available through name resolution services (NED, SIMBAD).	BANDPASS	gamma-ray x-ray UV optical IR mm/sub-mm radio
DATA CLASS	pointed observation catalog survey reference data simulation		
OBSERVATORY	mission, facility, or observatory	TIME	date or date range

The basic query they came up with at the first of their two meetings is shown in Table 1. The most common feature of the participating services was the ability to search for data based on source

name or coordinates. The other items here were considered essential to prevent the user from being inundated by irrelevant information and to protect the services against poorly posed queries from users. Participants envisioned one level of query where each service would return descriptions of their holdings, which would include some kind of description along with information for at least the information given in Table 1. A full query would return information about all of the data sets matching the given criteria. The information might be as rudimentary as a count of the number of matching data objects, or might include object descriptions with URLs taking users back to the service involved. The system for figuring out what services are participating might require some centralized registration scheme, although this could evolve into a fully distributed system resembling the Internet name resolver technology.

5. SUMMARY

At the moment, attempts to supply inter-operability to independent astronomical data services on the Internet are just getting underway. Much work remains to be done just to achieve the retrieval of astronomically interesting data. We have not reached the point of applying the ideas of Intelligent Information retrieval, but it is not too soon to start thinking about it. The obvious need is for an intelligent agent for deciding which services to query, based on the information needs of the user.

Developers should avoid re-inventing the wheel. The Z39.50 protocol is available for exchanging queries, the HTTP protocol is the prohibitive favorite for returning query results, and formats are available for database descriptions – too many, perhaps.

And finally, whenever a service involves independent sites, it is impossible to overestimate the importance of cooperation between the sites.

REFERENCES

- Belkin N.J. 1993, in *Intelligent Information Retrieval: The Case of Astronomy and Related Space Sciences*, eds. A. Heck & F. Murtagh, Kluwer Academic Publishers, Dordrecht, p. 9
- Heck A., Murtagh F. 1993, in *Intelligent Information Retrieval: The Case of Astronomy and Related Space Sciences*, eds. A. Heck & F. Murtagh, Kluwer Academic Publishers, Dordrecht, p. 3