The Harvest Information Discovery and Access System

'Harvest is an integrated set of tools to gather, extract, organize, search, cache, and replicate relevant information across the Internet. With modest effort users can tailor Harvest to digest information in many different formats, and offer custom search services on the Internet. Moreover, Harvest makes very efficient use of network traffic, remote servers, and disk space.'

'Harvest provides a very efficient means of gathering and distributing indexing information; supports the easy construction of many different types of indexes customized to suit the peculiarities of each information collection; and provides caching and replication support to alleviate bottlenecks. '

The Harvest project is motivated by the facts that

- WWW information indexers make little attempt to share overlapping information with other indexers
- most information indexing systems offer little support for customizing how different information formats and handled
- network and server bottlenecks occur with popular data sources and indexes

It is essentially an architecture to provide sites with tools to gather, index and query information collections. Indexes, rather than data objects (documents etc) are replicated across distributed sites and update algorithms distribute updates from index to index. Sites can utilise some or all of the components of the architecture.

Architecture components (see technical discussion)

gatherers	avoid retrieval of individual objects site-resident software does indexing (Provider) gatherer retrieves indexing information from Providers Essence information extraction system
brokers	provide indexed query interface to information retrieve info from gatherers or other brokers incrementally update indexes
index/search systems	a broker-indexer interface accomodates a variety of search engines currently support WAIS, Glimpse and Nebula (Glimpse distributed with system)
replicators	Flood-d sends objects from one member of a broker group to others along a graph Mirror-d sends state information to catch missed updates
object caches	very fast cache (2x CERN object cache for hits, 100x for misses)
Harvest object system	users can associate method code with data which is downloaded and executed

Gatherers and Brokers communicate using an attribute-value stream protocol called the Summary Object Interchange Format (SOIF), and can be arranged in various ways

- A Gatherer can access an information Provider from across the network using the native FTP, Gopher, or HTTP protocols.

- A Gatherer can be run at the Provider site, saving a great deal of server load and network traffic.
- A Broker can collect information from many Gatherers, to build an index of widely distributed information.
- A Gatherer can feed information to many Brokers, saving repeated gathering costs.
- Brokers can retrieve information from other Brokers, in effect cascading indexed views from one another, using the Broker's query interface to filter/refine the information from one Broker to the next.

Information extraction

The Essence information extraction system is customizable to deal with different information formats. Basically it is able to deal with encodings such as *tar*, and extracts information depending on file file types. For example it can be customised to extract author and title lines from LaTeX documents.

Comments

Harvest seems very similar in notion to NCSTRL with similar pros/cons (see review)

Sites can provide their own indexes, search engines and interfaces to the indexes (A list of brokers/gatherers/caches)

The architecture is scalable

Distributed indexes are not mirrors. Replications occur to incrementally update indexes

Query interfaces differ from site to site.

Interfaces to the *same* index can be different (eg CS report duplicate sites)

One broker supposedly indexes 32000 CS tech reports from 280 sites dealing with multiple formats.

There is a digital library broker (although it seems inactive at the time of writing).

The cache is very fast

My initial impression was that there would be some joining of indexes or some dynamic distributed index querying so that I could go to any Harvest broker and reap information from remote indexes. This isn’t the case. The indexes seem pretty much stand-alone.

The Glimpse search tool supports some regular expressions, querying by attribute (eg title, full-text), phrase searching, boolean queries, case sensitivity on/off, word boundary matching, number of errors allowed, and different result set presentations (verbose, matched lines, object descriptions). The CS Report Broker provides predefined search methods (near, accrue, and, or, phrase, sentence, paragraph, stem, thesaurus) available from a menu, applied to a list of keywords.

Steve Jones July 1996