An Introduction to Heritrix

overview of the Heritrix crawler, *circa* version 1.0.0 (August, 2004). It outlines the original use-cases, , the general architecture, current capabilities and current limitations. It also describes how the crawler is currently used at the Internet Archive and future plans

The Heritrix Project

strategies tested, 2nd Q 2003

-Core crawler without user-interface created, to verify architecture and test coverage compared to HTTrack [HTTRACK] and Mercator [MERCATOR] crawlers, 3rd Q 2003

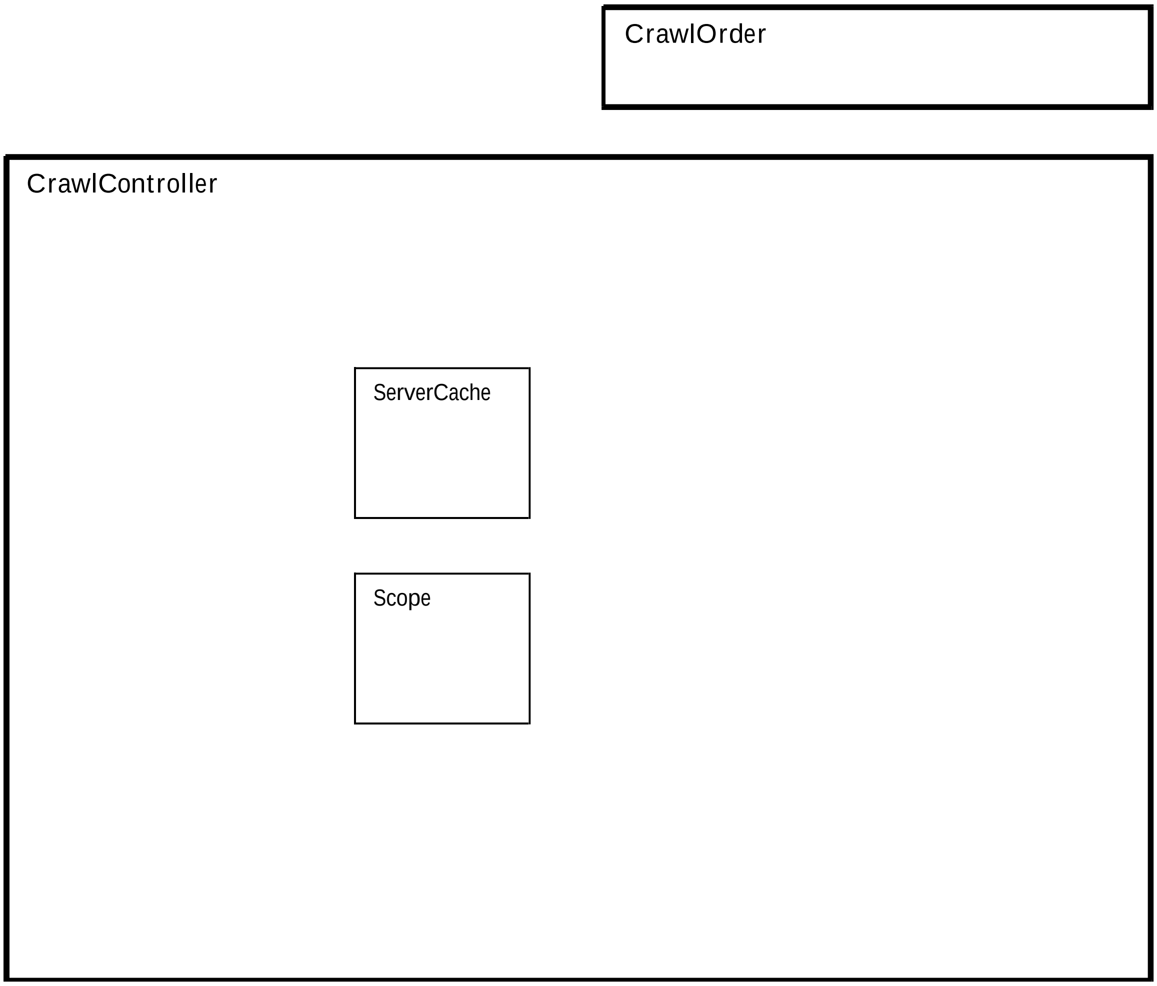
-Nordic Web Archive [NWA] programmers join project in San Francisco, 4th Q 2003 – 1st

The Frontier tracks which URIs are scheduled to be collected, and those that have already been collected. It is responsible for selecting the next URI to be tried (in step 1 above), and prevents the redundant rescheduling of already-scheduled URIs (in step 4 above).

The Processor Chains include modular *Processors* that perform specific, ordered actions on each URI in turn. These include fetching the URI (as in step 2 above), analyzing the returned results (as in step 3 above), and passing discovered URIs back to the Frontier (as in step 4 above).

Figure 1 shows these major components of the crawler, as well as other supporting components, with major relationships highlighted.

Figure 1: Major Components of Heritrix in a Representative Configuration



The *ServerCache* holds persistent data about servers that can be shared across CrawlURIs and time. It contains any number of *CrawlServer* entities, collecting information such as IP addresses, robots exclusion policies, historical responsiveness, and per-host crawl statistics.

The overall functionality of a crawler with respect to a scheduled URI is largely specified by the series of Processors configured to run. Each Processor in turn performs its tasks, marks up the CrawlURI state, and returns. The tasks performed will often vary conditionally based on URI type, history, or retrieved content. Certain CrawlURI state also affects whether and which further processing occurs. (For example, earlier Processors may cause later processing to be skipped.)

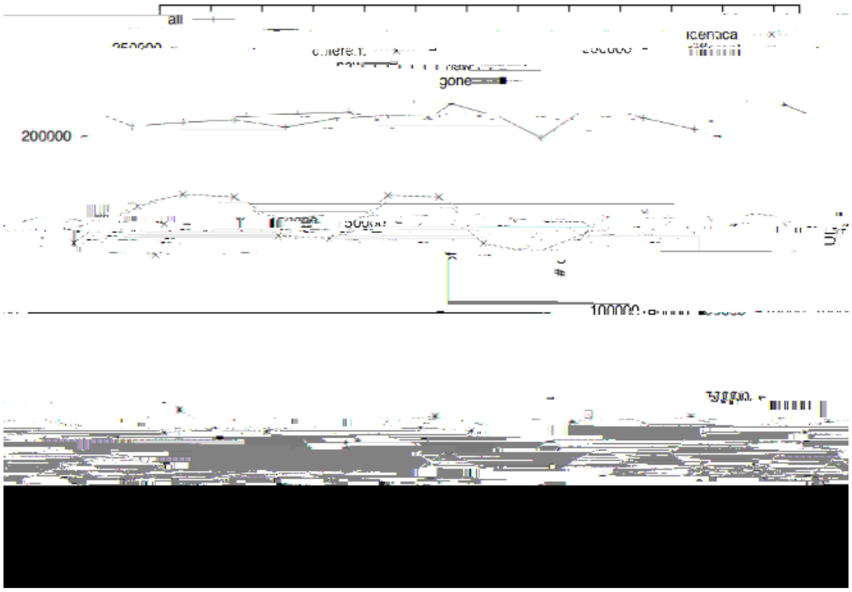
Processors are collected into five chains:

Processors in the *Prefetch Chain*

	FetchHTTP	Performs HTTP retrievals, for URIs of the “*http:*” *https:*” schemes.

instances whether all instances are run on a single machine or spread across multiple machines.

[redacted]



Many of these tests and quality assurance procedures are still in development as we continue to learn more about quality archival crawling.

Future Plans

Current plans for the future development of Heritrix, post 1.0.0, fall into three general categories: improving its scale of operation; adding advanced scheduling options; and incremental modular extension.

Separately, there are a number of incremental feature improvements planned to expand the base crawler capabilities and the range of optional components available. The Internet Archive plans to implement:

Support for FTP fetching

Improved recoveryfor 88 774eptIm set checkpoints

Automatic detection and adaptation to many “crawler traps” and challenging website design practices (such as URI-line session-IDs)

Additional operator options for specifying 88 77 scopes, dealing with in-88 77