

# Fulcrum Ful/Text and SearchServer (PCDocs, Hummingbird, and OpenText)

© 2013 by Stephen E. Arnold, [www.arnoldit.com](http://www.arnoldit.com)

*Fulcrum positioned itself as a search system to leverage intellectual capital via content-based retrieval software. The 1983 system is sold today.*

*Fulcrum Technologies is now absorbed into OpenText. For most people, the identities of Ful/Text and SearchServer have been forgotten. From 1983 until the mid-1990s, Fulcrum was an Avis to the Verity Hertz approach to search. Fulcrum licensed its search system to original equipment manufacturers. Fulcrum expanded from search into marketing territory populated with buzzwords like “knowledge management,” “semantics,” “structured and unstructured information,” “federated search,” and similar jargon in the hopes of boosting revenues. Like Verity, Fulcrum dipped in and out of financial hot water. Like Verity, Fulcrum sold itself and then found that its buyer turned around and sold itself.*

*Fulcrum, based in Ottawa, landed some high profile clients, including the European Community. The company added to its technology by purchasing from America Online the WAISserver technology. Fulcrum marketed its WAIS-based SearchServer as a distributed search solution. Fulcrum asserted that its system delivered federated search results.*

*Fulcrum embraced a repository approach to content. Verity’s system left the source content on the server where the information was placed by a user. However, in other broad respects, Fulcrum’s business trajectory makes clear that enterprise search that offers basic key word and concept retrieval cannot sustain a commercial enterprise. The cost of staff, customer support, and system development are too onerous.*

*Fulcrum sold a share of its company to an Italian firm. Then the company embarked on a decade long journey of management shifts, product expansion, and ownership changes. Fulcrum’s journey anticipates most of the features touted by vendors in 2013 as “innovative.” Fulcrum also is a reminder that enterprise search is a tough business to grow with an on-premises license fee and consulting services financial model. You can still download Fulcrum code at <http://fulcrum.downvn.com/Linux-software-download/fulcrum>.*

*Author’s note:*


*This is a 2006 draft. It will not be updated.*

*Stephen E Arnold, December 23, 2013*

## Introduction

Fulcrum Technologies' journey from start up in 1983 to the search technology becoming part of OpenText in 2006 is not a recreational tour. The company's financial ups and downs, its surprising success in competing successfully against Verity, and the longevity of its technology are not widely discussed. The company's core technology is not the basic retrieval system that was used to search text on CD-ROMs. SearchServer reaches back to an Internet search technology developed by Thinking Machines, Brewster Kahle and others, and subsequently owned by America Online. Fulcrum Technologies' decades old system survives mostly unchanged in 2006. OpenText, the owner of the Fulcrum software, is likely to invest modestly in what is one of the oldest information retrieval systems available for licensing.

Table 1: Hummingbird Search Server: A Bird's Eye View

Product Thumbnail	
1 Search Brand	Ful/Text, Search Server, SearchSQL, SurfBoard (Z39.50 indexing), Fulcrum Knowledge Server and Workstation
2 OS Supported	IBM AIX, Microsoft, Solaris, Linux. (Strong Microsoft orientation), HP-UX
3 Est License Fee	Begins at \$5,000 per server and \$295 per seat. Maintenance is 15 percent per year. Custom price quote required.
4 Functions	Last in, first out indexing, automated metadata extraction, support for natural language queries of database (structured) content and standard office file types, support for languages such as Korean, and data discovery operations
5 Claimed Features	Hummingbird asserts that Search Server makes use of a distributed architecture to deliver high performance search.
6 Downsides	The system has been plagued with technical issues for years.
7 Similar To	ISYS Search Software, dtSearch
Product Close Up 	Hummingbird SearchServer is an advanced data retrieval solution for high volume, line-of-business Windows and UNIX information applications. Instead of basic search, Search Server provides "K" solutions; that is, knowledge-intensive applications with search. The user runs a query and the system displays results without the user having to know about collections, file types, or applications hosting the information. Hummingbird offered an original equipment option similar to Verity's. Hummingbird is integrating the Search Server technology into its other applications; for example, content management and document management. The system is alleged to be more responsive (fast) than Verity's.

### A Good Idea: Client-Server Search

Fulcrum Technologies was founded by four colleagues who lost their jobs at Ottawa-based NABU Manufacturing Ltd. The four principals embraced the idea of information retrieval delivered via the then-new client-server architecture, not the traditional mainframe. The software angle was to provide

---

"Everyone talks about the ROI of intellectual capital - this is where Hummingbird walks the talk. Hummingbird SearchServer greatly enhances the 'speed-to-mind' of an organization's information assets by delivering relevant information to employees, enabling them to make better decisions faster," said Andrew Pery, chief marketing officer, Hummingbird Ltd. "Hummingbird SearchServer is an invaluable asset to an organization's knowledge access strategy, whether it is utilized with Hummingbird Enterprise to perform searches across all organizational information sources or as a part of a customer or an OEM partner solution."—Hummingbird news release, in 2002 at <http://goo.gl/PXHHUf>

search and retrieval using client-server technology, not the mainframe approach used for IBM STAIRS and InfoData INQUIRE.

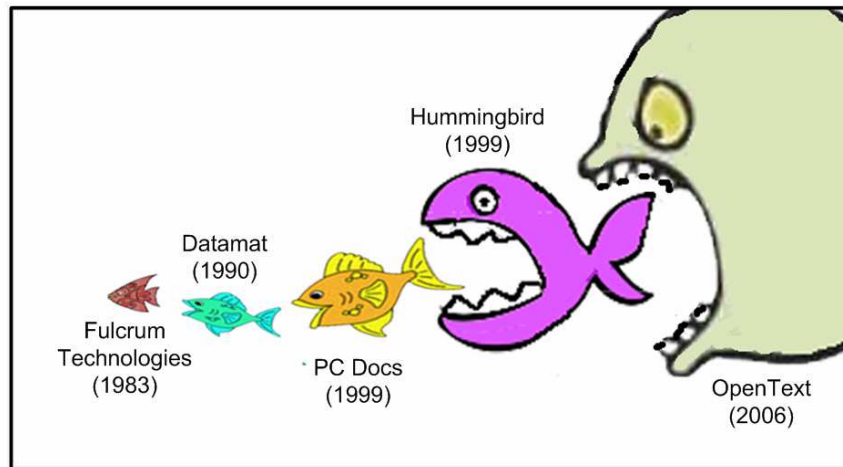
Long before the phrase "stealth mode" became part of the entrepreneur's jargon, Fulcrum maintained a low profile. The founders—Eric Goodwin, CEO, Peter Eddison, marketing, Peter Reid, CFO, and Ken Leese, Fulcrum's chief technologist set about assembling a search system. Once the FIND search system was in hand, the founders used their sales skills and contacts to start licensing the product. Fulcrum was less expensive to implement as a client-server system than a traditional mainframe search system. For five years, from 1983 to 1988, Fulcrum faced little direct competition. The company bet heavily on CD ROM retrieval, licensing to OEMs (original equipment manufacturers), and enterprise deals.

Fulcrum was one of the first search vendors to experience the thrill of victory and the agony of defeat. As competition in enterprise search increased, Fulcrum had to work overtime to prevent licensees from jumping to Verity's solution. Fulcrum invested in software that allowed its search system to index the contents of traditional databases. The company enhanced the search system's support for external dictionaries and implemented automated indexing and classification of content. Fulcrum, if the marketing collateral is accurate, was a pioneer in providing a search system that made it possible to search different types of content from a single interface. Fulcrum offered a search solution for Microsoft's popular Exchange email server and Word files. Fulcrum supported English, French, and other languages, including Japanese. The company was among the first to use the word "knowledge" to describe the problem enterprise search addressed. Today, these features and functions are items on a vendor's checklist of features. But in the years before Verity hit the market, Fulcrum was a *de facto* leader in search with tendrils of functionality reaching into document management, personalized information delivery sometimes called "selective dissemination of information," and expert identification. Using log data, Fulcrum's system could identify which person in an organization was knowledgeable about a specific topic.

When Verity entered the market, Fulcrum was focused on a centralized approach to search and content processing. Verity introduced a distributed architecture. The idea was that it was more efficient to use resources on the network near the content than it was to create a centralized data warehouse that effectively doubled the cost of storage and required a massive infrastructure.

Fulcrum responded with its own distributed architecture dubbed SearchServer. Fulcrum was "inspired" by Brewster Kahle's Wide Area Information Server approach. Fulcrum, despite the financial pressures it faced, bought the WAISserver technology that by 1996 was part of the America Online company. SearchServer, which dates from the late 1980s, is still in use by OpenText Corp., the company that took ownership of Fulcrum's software in

2006. Fulcrum Ful/Text and Fulcrum SearchServer may be the oldest search technology still in the commercial channel. As many search vendors' technology were removed from service, Fulcrum's search software soldiers on. This is a tribute to the quality of the Fulcrum engineering and to the willingness of companies that have owned Fulcrum over the last 20 to 25 years willingness to keep selling a somewhat dated, almost antique search and retrieval systems.



Fulcrum's technology was sold to Datamat, an Italian software integration company in 1990. Then in 1999, when Fulcrum had fallen into some financial difficulties, PC Docs (another Canadian document management company) bought Fulcrum in 1999. Later that year, Hummingbird (a Toronto-based firm with which Fulcrum had a relationship) purchased PC Docs in 1999. Then in 2006, OpenText, which had made an attempt to purchase Fulcrum in 1999) purchased Hummingbird. Today OpenText markets the Fulcrum technology. One can argue that Fulcrum is the search system which has one of the longest continuous records for enterprise deployment.

But Fulcrum made some interesting business decisions as it moved through its quarter century trajectory. First, the company focused on CD ROMs. When that technology collapsed, the Internet had become "the next big thing." Fulcrum had to scramble to create an Internet-centric product. That delay contributed to the company's volatile financial performance.

Second, the company was strong in direct sales, but the company was not a sophisticated marketer. The Fulcrum brand had cachet where Fulcrum made inroads based on relationships and referrals. Otherwise, the company had a very low profile. Once the firm went public in 1993, Fulcrum had to communicate with investors. By 1995, when the firm's stock was listed on the

Toronto Exchange, Fulcrum was taking a more consistent approach to establishing a brand and creating buzz around specific products like Ful/Text and SearchServer. Compared to the buzz Verity had established, Fulcrum was perceived as a secondary player, not the leader the firm actually was.

The company made an effort to demonstrate the precision and recall capabilities of its software by participating in the TREC project, sponsored by the US government. The problem was that the information about a specific vendor's search system relative to other companies participating in annual search "showdowns" was not good fodder for marketers. The TREC results were for research purposes only. Fulcrum performed as well as other sophisticated systems; that is, hitting about 80 percent for precision and recall on various TREC test collections. Once the sale and resale of the company began in 1999, the identities of the Fulcrum products was continuously diluted.

Fulcrum's search system is a component in larger enterprise solutions. These are positioned as knowledge management, content management, portal systems, and other jargon. Underneath the layers of third-party software, systems developed in 1983 continue to deliver search and retrieval to users who license "solutions" from the current Fulcrum owner, OpenText. OpenText owns BRS (bibliographic retrieval services) search, the original OpenText SGML search system developed by Tim Bray and his colleagues, Nstein (a content processing and indexing system acquired by OpenText), and BASIS (a report-oriented search system developed by Information Dimensions Inc. also purchased by OpenText). With so many different and incompatible search systems to support, it is unlikely that OpenText can make significant, sustained investments in Fulcrum's search and content processing systems. The OpenText solution that one can license today may be running software that is mostly unchanged since the early 1980s.

## **Selected Executives and Board Members**

- As the company moved through its different owners, not surprisingly, top management changed. The original founders were:
- Eric Goodwin, chief executive officer
- Peter Eddison, vice president of marketing
- Peter Reid, chief financial officer
- Ken Leese, president during Fulcrum's start up phase. Then he became the chief technology officer and product development, including Fulcrum FIND and Fulcrum Knowledge Network

An important hire for Fulcrum was Mike Laginski, marketing, sales, product development, and customer services. Laginski was formerly a Lotus Development Corporation executive. He became chief operating officer of Ful-

---

“There is enormous potential in leveraging PC Docs' document and knowledge management solutions, and its financial and case management systems with Hummingbird's business intelligence products. Our first aggressive move will be to commence integration of our technology platforms to deliver to the market the first true enterprise knowledge portal offering as quickly as possible.” — Ruby Osten, CEO, PC Docs at <http://goo.gl/PXHHUf>

crum. Fulcrum and IBM Cognos enjoyed a productive relationship partially because of Mr. Laginski.

The voice of Fulcrum's technology after the 1999 purchase of the company by PC Docs was Stephen Tomlinson, a technologist.

Fred Sorkin, a principal in Hummingbird, joined the Fulcrum Board of Directors. In 1999, Hummingbird acquired PC Docs and the Fulcrum Technologies' assets. Similar threads unite Fulcrum and Datamat executives.

## **Selected Clients**

Fulcrum did not provide a list of its clients prior to its initial public offering. Based on references to Fulcrum's software, the following list represents some of the companies that either licensed Ful/Text or another Fulcrum product for enterprise use or entered into an OEM agreement to use the Fulcrum Ful/Text search technology in products identified with the OEM partner; for example, Allaire's ColdFusion application development platform. Verity and later Autonomy emulated Fulcrum's OEM approach.

- Adobe, which subsequently licensed the Lextek Onix system
- Agence France-Presse
- Charles Schwab
- CompuServe
- Digital Equipment Corp.
- European Commission
- European Space Agency
- Hewlett Packard
- IHS Environmental Information
- Microsoft (Once used to index Microsoft Network)
- Michigan State Police
- Nabisco
- Novell
- Olivetti
- Siemens
- Sun Microsystems
- Thomson Corporation (First Call)
- Unisys Corporation
- Wells Fargo.

Fulcrum entered into a number of partnerships. The idea is that partners have information about the search and content processing needs of their customers. The partner introduces the Fulcrum system to its clients. Both the partner and Fulcrum enjoy financial benefits from the introduction. Autonomy and Endeca implemented a similar approach possibly learning from Fulcrum's efforts.

- Astea International
- Clarify
- Fujitsu
- Platinum Technology
- ProAmerica
- Quintus
- Software Artistry.

## **Financial Performance**

Fulcrum was a privately-held firm from 1983 until 1993. The financial performance of the company between 1993 and its purchase by PC Docs in 1999 was erratic. The company dipped in and out of profitability in those six years. Once the company was purchased by PC Docs, the Fulcrum property was absorbed into the PC Docs financial report. With each acquisition, the revenue directly related to Fulcrum software has been difficult to determine.

Based on information shared at search conferences, the sale to PC Docs was necessary. Fulcrum owed money and lacked the resources to meet that commitment.

Financial data for the period from 1991 to 1995 are available from the US Securities & Exchange Commission. Fulcrum's revenue rose from \$6.3 million in 1991 to \$34.5 million in 1995, a robust 5X multiplier. At the same time, the company reported a net income increase from \$139,000 in 1991 to \$5.3 million in 1995. In this same time frame, Fulcrum was a beneficiary of Canadian tax credits and government grants. The company also suffered in the foreign exchange market. Research and development expenses in the five year interval skyrocketed from \$1.3 million in 1991 to \$6 million in 1995. The cost of funding search technology is a significant cost factor in the Fulcrum financials. Another cost spike is evident in the cost for sales and marketing. In 1991, Fulcrum spent \$4 million to close deals. In 1995, the company reported expenses of \$19 million. With about 250 full time equivalents, Fulcrum was generating about \$137,000 per FTE. Laying off employees and cutting costs loomed for Fulcrum without a buy out deal.

Fulcrum's search business required increasingly large investments to close deals and keep the technology humming. Fulcrum's cost patterns are a reminder to other search entrepreneurs that the sales process can be a

---

The enhancements to SearchServer [3.7] will allow us to strengthen our leadership position for rich knowledge management applications that require strong, underlying search software,” said Fulcrum president and COO Mike Laginski. Pricing for SearchServer Version 3.7 varies according to the number of users; customization services are available to tailor each installation to specific customer needs.— Mike Laginski, chief operating officer

lengthy one. The need for resources to maintain and enhance the technology indicate an ever increasing appetite.

PC Docs paid \$21 million in stock for Fulcrum in 1997. The year the deal closed, Fulcrum’s revenue dipped and the company was losing money. PC Docs appears to have snagged some high value technology in a bargain basement sale without having to use cash. Hummingbird purchased PC Docs for a stock deal estimated at \$155 million.<sup>1</sup> When Fulcrum ended up in the hands of OpenText in 2006, that deal was worth an estimated \$450 million.

The lesson is that the value of a search vendor comes from selling a company with customers and a revenue stream. The “value” of Fulcrum is an important signal for other search vendors who want to cash out of a business from which it is difficult to generate and maintain strong top line growth and a healthy margin.

Fulcrum is one of the first vendors to make explicit the management, technical and financial challenges associated with a business focused on information retrieval.

## The Product Line Up

Prior to the sale of Fulcrum to PC Docs, Fulcrum had a healthy portfolio of products. Each of these expanded the application of search-and-retrieval technology to functions.

### *Ful/Text Search*

In 1995 Fulcrum offered a “family” of products.

***Ful/Text Server.*** The Ful/Text Server is the core full text indexing and retrieval engine. In 1995, the company released version 6.1. In 2006, OpenText left the basic search-and-retrieval Ful/Text system mostly unchanged.

***Ful/Text Software Development Kit.*** The Ful/Text SDK is a command library consisting of more than 250 C language function calls. Software developers code features and functions they require the Ful/Text Server to perform.

***SearchServer.*** This is an SQL-based collection indexing and retrieval software. SearchServer bakes in the Ful/Text Search system. Developers familiar with Structured Query Language build search-based applications on the SearchServer. The content must be in a format supported by the SearchServer. Organizations without unstructured information have to transform the content for the SearchServer. In contrast to Ful/Text, SearchServer is designed to be more “developer friendly” and make it possible to support a distributed architecture similar to that used by Verity.

---

<sup>1</sup> See Nancy Weil, Network World Fusion, March 5, 1999 at <http://www.networkworld.com/news/1999/0305pcdocs.html>



Fulcrum anticipated iPhrase's and MarkLogic's approach to document management. In SearchServer, text documents are represented in a relational-like table. The table incorporates a full-text index.

SearchServer conforms to subsets of the Open Database Connectivity (ODBC) interface for C programming language applications and the Java Database Connectivity (JDBC) interface for Java applications. Almost 200 document formats are supported, such as Word, WordPerfect, Excel, PowerPoint, PDF and HTML. SearchServer works in Unicode internally and supports most of the world's major character sets and languages. SearchServer supports more than a dozen languages.

The company said in 1995:

Since its introduction, SearchServer has been licensed to over 100 corporate customers including AT&T Global Information Solutions Inc., National Semiconductor Corporation, Florida Power Corporation, CompuServe Incorporated, the European Economic Community and Banca di Roma.<sup>2</sup>

Search Server is based on WAISserver. The company's high-profile system is derivative. The innovation approach pivots on acquiring technology and then marketing that technology more effectively.

**SearchServer Software Developer's Kit.** The SDK is available for C programmers developing SearchServer-based applications on non-Windows platforms. This SDK provides access to SearchServer functionality directly through the SearchServer Application Program Interface (API). Fulcrum asserts that its approach is standards-based.

**Search Builder.** The Search Builder software provides graphical development tools. Developers can integrate SearchServer in custom application solutions for the Windows desktop. Programmers can use Powersoft's PowerBuilder and Microsoft Visual Basic and Visual C++.

**Fulcrum Surfboard.** Surfboard is the connector to hook Fulcrum technology to the Internet. The software consists of a Web server, a gateway, and the Surfboard server. The system supports document databases. The system is a bit of a Rube Goldberg machine. The Web server forwards search requests to the Surfboard gateway. Then the HTTP requests are converted to Z39.50 requests. The gateway then translates the search results back into HTTP responses. In a sense, Surfboard is the Ful/Text search engine rigged for use on Internet servers. Fulcrum states:

Surfboard makes it easy to bring robust and scalable text retrieval to Internet applications because it is fully compatible with all major Internet and World Wide Web standards for document formats like HTML, Internet NetNews, E-mail, and ASCII text, and for Internet browsers including those compatible with HTTP such as Netscape

---

<sup>2</sup>. Fulcrum Technologies Inc., 1995 Form 10-K Annual Report.

Navigator and NCSA Mosaic, WAIS, Z39.50 and Gopher.” Surfboard also supports clients from on-line services like America On-Line, Delphi, and CompuServe.<sup>3</sup>

Easy is a relative concept. Surfboard is Fulcrum’s response to the explosion of interest in Internet technology in general and Internet-centric search and retrieval.

## SearchServer Up Close

---

Fulcrum markets aggressively to large companies, government agencies, and influential enterprise hardware and software partners; for example, Hewlett Packard and IBM. The reason is that SearchServer is a complex constellation of software. A licensee must have the resources to license, deploy, provision, maintain, and optimize SearchServer. By 2006, SearchServer abandoned its initial principle of a small, easy-to-deploy system.

### Innovations?

Fulcrum’s shift from relative simplicity to notable complexity is important. The company responded to its need to generate revenue by layering more and more features and functions into its Ful/Text system. By grafting WAIS-server-like architecture to the Ful/Text system, Fulcrum blazed a trail that would be followed by such vendors as Entopia and Fast Search & Transfer.

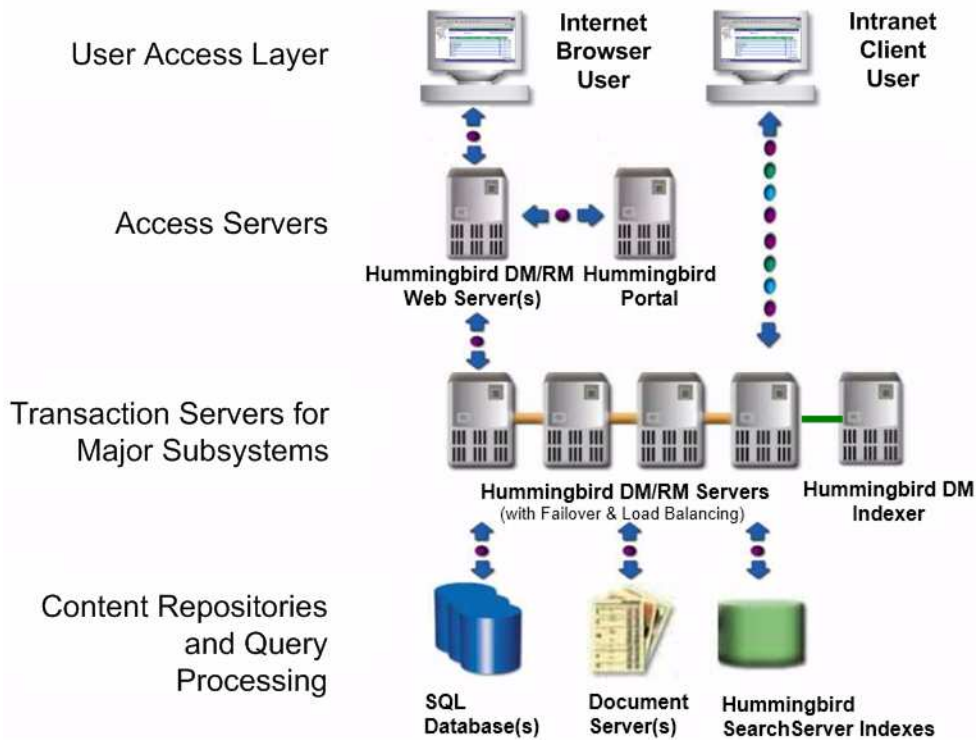
#### WAIS: A Snapshot

The Wide Area Information Servers (WAIS) is a hierarchical system of searchable databases distributed across servers on the Internet. The late 1980s’ design permits distributed management of information. The idea for WAIS was initially a Thinking Machines’ project. But other firms’ engineers contributed. The system implemented the ANSI standard Z39.50 to search indexes on remote computers via TCP/IP. WAIS Inc. was set up in the early 1990s as a commercial entity. The US government funded CNIDR, an acronym for the Clearinghouse for Networked Information Discovery and REtrieval. CNIDR created an open source version of WAIS. WAIS Inc. was acquired in May 1995 by America Online. AOL sold WAISserver to Fulcrum in 1996. The WAIS project provided impetus to structured data access via the Internet. One drawback of WAIS is that the information to be indexed has to be local to the machine running WAISserver.

---

<sup>3</sup>. Fulcrum Technologies Inc., 1995 Form 10-K Annual Report product discussion section. Digital filing is not paginated.

Search, Fulcrum executives decided, was not by itself a big revenue producer.



SearchServer sits in the functional center of the Hummingbird FTR (full text retrieval) system for content management. The system requires multiple layers. Each layer requires hardware and software. The basic search and retrieval functions have ballooned in complexity. Tuning a SearchServer is a complex and often time consuming task. Managing a SearchServer required a separate development environment that adds to the cost of the SearchServer-based system.

Fulcrum was an early adopter of search application programming interfaces. A developer could integrate search and retrieval into other enterprise applications or build a completely new information-centric application. Today a search API is an item on a checklist. In the mid-1980s, the idea, when applied to search, further differentiated Fulcrum's system from the mainframe systems then available.

Ful/Text and SearchServer solved an information access problem for large organizations. Fulcrum's approach integrates access to text (unstructured information) and structured data (row-and-column data). The system could allow an enterprise user to run a single query and see relevant information from different sources in the organization.

Peter Eddison told the Seybold Group in 1996:

---

**“It’s easy now to see that Fulcrum made a crucial wrong turn in the mid-1990s, when it failed to anticipate the impact of the Internet. The company then tried to adapt its search engine technology to the mass market when the software was already becoming a commodity. “We should have made search engines better,” said a former senior designer, “not just more accessible.”—<http://groups.yahoo.com/group/Ottawa-HiTech/message/766> in 2003**

Fulcrum invented the idea of an API for text retrieval. At the beginning of the '90s the notion of 'fitting in' seemed to us to mean integrating into corporate databases as well as indexing office documents. The full-text retrieval market has been much smaller than the structured database management environment, and it seemed logical to develop a text retrieval product that would integrate well with database applications. We initially attempted database compatibility with a variation of the structured full-text query language (SFQL) proposed by the aircraft manufacturers and airline companies for CD-ROM, but by late 1991 it became clear that with Microsoft behind it, what would eventually surface as ODBC was the database integration standard to go with. Our SearchServer engine was developed with an architecture compliant with ODBC (Microsoft's Open Database Connectivity) and SQL (Structured Query Language) Access Group.”<sup>4</sup>

SearchServer is important because it collects a number of interesting functions within one system that is made up of content transformation operations, content processing and indexing functions, and information retrieval via key words or Boolean queries. Search vendors entering the market after Fulcrum as a brand had dissipated are often surprised to learn that the Ottawa-based company is one of the first enterprise vendors to:

- Provide a system that could enable an enterprise-wise view of information from a single point of access via key word search
- Support content in traditional relational databases so that the transformed content in a SearchServer repository was the single place a user could access to retrieve source documents and data
- Integrate content from diverse sources so the user does not have to launch native applications to view content or run standalone queries on different systems to locate scattered information
- Make what is called “knowledge” available to users of an Intranet in a single facility or in geographically dispersed facilities
- Support fine-grained security for data and documents
- Integrate with Microsoft systems, including Microsoft BackOffice and Exchange Server
- Offer automatic indexing and tagging content with categories (facets)
- Permit cross-language document retrieval via multilingual dictionaries that translate a user’s query
- Make search-based applications part of information retrieval.

<sup>4</sup>. Bernard Banet, “Fulcrum’s SearchServer Family: Standing Out by Fitting In, *Seybold Report on Desktop Publishing*, May 20, 1996.

As prescient as Fulcrum's technologists are, the company was slow to grasp the importance of Internet technology. In late 1994, Fulcrum offered software that included a spider (Internet content acquisition component subsystem) and browser-based search and the display of HTML content.

## **Selected Features of SearchServer**

By 1999, Fulcrum's Search Server had a number of features that other vendors were adding to their systems; for example:

- Search server supported free text and natural language queries. Full support for Boolean was available. Search Server offered a "fuzzy and" that could relax a query or correct a user's spelling in order to return possible relevant hits otherwise excluded from a strict Boolean result list.
- Statistical algorithms for relevance ranking, including inverse document frequency to assign higher weight to rare or specialized terms typically found in scientific and technical documents. The user can change the reorder of the returned documents for quick results list browsing.
- Lemmatization (word stemming), a feature lacking from Google until the mid-2000s
- Use of a thesaurus or external dictionary to handle synonyms or other special word uses. Search Server would index the words and phrases in a document and also assign terms in the controlled term list
- Support for "more like this." When activated, it locates, ranks, and lists all documents with similar content in the selected sources.
- Multilingual Document Searching, featuring advanced linguistic support for all main European languages, plus Chinese, Japanese, and Korean. Search Server supported Unicode years before other vendors. Search Server has permitted search of content in Russian and Arabic for decades.
- Database Searching, allowing the content of relational database management systems to be searchable on the content of any column in a database row.
- Last in First out: Allows the most recently indexed documents to be at the top of the result list set without any sort requirement.
- Specialized views. Non-HTML documents can optionally be rendered in high-quality HTML, including graphic images in many formats.

Search Server is a proprietary software product.

---

**“Our focus is on the corporate Intranet. In the more structured world of intranets, web crawlers are not the way to go. We are trying to differentiate ourselves by ease of use,” Mr. Eddison states. “Our Intuitive Searching allows the user to grab an interesting paragraph, and say: “Find me more like this. We firmly believe that the user is the smartest part of the process, and we provide tools to help the user find information, wherever it is.”—Tony McKinley “Intranet Content at AIIM, no date.**

## Search Server Functionality

SearchServer uses server-based indexes, not a content repository like iPhrase or TeraText. When users search for information, it's the index that is searched, not the information sources.

### *Search*

Ful/Text is a full-text indexing and retrieval system. It can be used for both unstructured information and for structured data. The structured data requires use of routines developed for the original Fulcrum SearchSQL module. The search component offers a library of more than 100 callable routines. A licensee can use these to create a content management system or other search-enabled application.

Fulcrum (Fulcrum 2000), which permits fuzzy Boolean searches, a relaxation technique primarily used for the Boolean operator AND, and having nothing to do with phonetic or orthographic variants.

SearchServer client applications are customized graphical screens set up for specific employees to use. These screens allow the licensee to offer advanced but easy-to-use techniques such as similarity searching (more like this), item summarization, fuzzy searching, and multilingual stemming. A search can be based on a word, a phrase, or the full text of the item. Alternatively, users can choose to search by structured properties, either standard or custom.

- PDF support: Fulcrum SearchServer currently has a text reader that uses Adobe's Acrobat toolkit to permit indexing of PDF documents on all supported UNIX and Windows platforms.
- XML support: SearchServer includes structured document support with the XML text reader. Users may restrict queries using XML document structure—for example, searching for terms that occur only within a specific element of the DTD. The XPATH query standard is being followed in conjunction with standard SearchServer search features such as Boolean, weighting of terms, and relevance ranking. Thus it is possible to rank rankings according to whether terms appear in a specific heading or in the body of the copy. This weighting technique is available for all searches, not just XML.
- Exchange Server: It is possible with the Enterprise KM to index and search mail and attachments. Native security is observed.
- Lotus Notes: It is possible to index information in Lotus Notes databases with the KM option of Hummingbird Enterprise.
- Data transformation: Hummingbird offers ETL (Extracting, Transforming [or Transporting] and Loading) functions to move information to and from a data warehouse.

SearchServer offers the capability of storing queries to run at scheduled times or when particular events (triggers) occur. As early as the mid-1990s, some of SearchServer's customers—for example L'Exposse and Agence France-Presse—had developed agents using Fulcrum's toolkits. Agents are now a standard pre-packaged feature with Hummingbird KM.

### ***Database Integration***

In many SearchServer applications, text retrieval capabilities are integrated with a relational database system (for example, call centers or problem tracking). SearchServer can access text objects stored within a relational database for indexing and retrieval. Such an application can take full advantage of these complementary technologies, allowing the full power of the SearchServer engine to be combined with the security and transactional stability of a relational database system.

SearchServer includes an ODBC database text reader that can be used to access information from within any ODBC-enabled database on Microsoft Windows, UNIX, and mainframe (OS/390) machines. Developers can add support for other databases on other platforms through the development of an appropriate database text reader.

SearchServer's database text indexing is done directly from the source database, without any export or storage to intermediate formats. Text can also be indexed incrementally when data changes, making maintenance of indexes nominally faster than with approaches that must export to formats such as HTML before indexing.

### ***Knowledge bases***

Ful/Text uses controlled term lists and dictionaries to perform query expansion and generate "See Also" or categories of related content. The use of knowledge bases means that the indexing system requires attention from a professional indexing professional. The terms are automatically assigned; however, the knowledge bases have to be maintained in order to take full advantage of controlled term indexing.

### ***Zones***

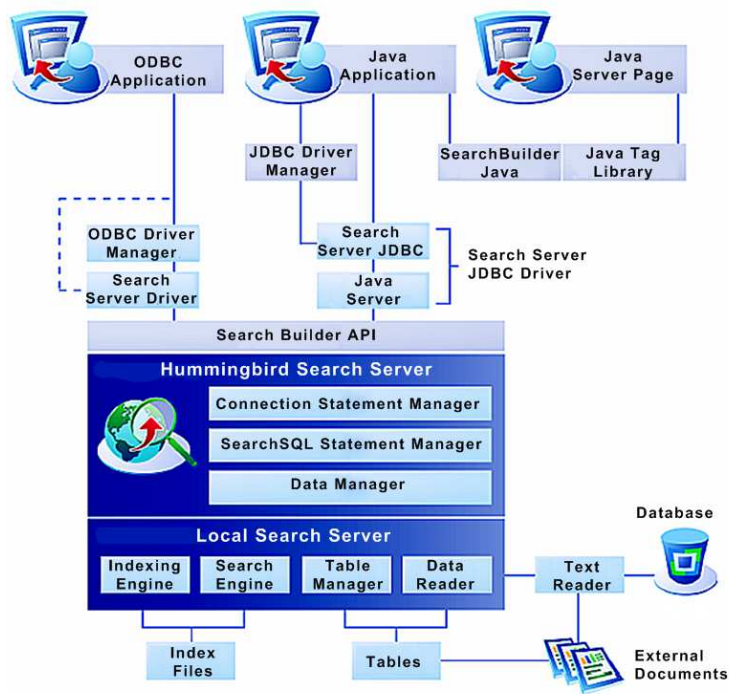
SearchServer's content processing identified "zones" within a document. The system assigns a "zone tag" to each document. The result is that each range of text is associated with a zone. The zones are included in the document index. A person searching for information can use the zone tags to limit the query to a particular region of a document. For example, a zone tag can identify a section heading. The search returns only documents in which the user's query terms appear in major headings within a document. Queries can combine a zone tag with words and phrases. The idea is to allow the user to pinpoint specific sections of long documents. These sections will eliminate

the need for the user to browse a long file looking for relevant sections. Fulcrum's hit highlighting allowed a user to examine a hit quickly.

## SearchServer Details

Stripped to its essentials, SearchServer acquires content, indexes it, and permits a user to locate documents matching a query. Fulcrum uses a client-server architecture. The server component is available for Unix and Linux. When OS/2 was available, Fulcrum supported that operating system. Fulcrum also supports Apple Macintosh PowerPCs. Most Fulcrum clients access the content via Microsoft Windows.

The core of SearchServer consists of a number of subsystems. The diagram below comes from Hummingbird marketing collateral. The main components of the system include a number of complex operations.



The SearchServer architecture accommodates structured and unstructured data. The approach invites dedicated servers for document processing, query processing, and system administration.

By 1990, enterprise search vendors were emphasizing that information retrieval was a mission critical function. Without access to information,



“knowledge” would be lost. In 1995, Fulcrum started describing its products as “K” systems. The “K” was shorthand for “knowledge.”

## **Content Processing**

A SearchServer administrator has to set up a collection of document files. Each collection may contain unstructured information such as Word files or structured information in the form of database tables from traditional databases like Microsoft SQL Server or Oracle. If content is in a form that SearchServer can process with its native filters, the administrator can move to the next step. Each collection had one or more “dictionaries” that provided access and term mappings. If these “mappings” are in hand, the indexing and repository builds begin. If the “mappings” are not available, subject matter experts have to create the controlled term lists, taxonomy/classification schema and any specialized dictionaries (for example, named entities).

Different content types have different formats. Fulcrum provided software connectors and adaptors that rendered the document in the format used by the source application. If a licensee wanted to process and display a document in a format not supported by Fulcrum, the application programming interface allowed a licensee to create a purpose-built display filter.

Fulcrum’s system indexes the documents in the collections. The index and the collection reside on-premises in one or more servers and storage devices. Proximity of the content processing and query system is one method Fulcrum used to argue that Ful/Text was more responsive (faster) than Verity Topic and later K2.

## **Query Processing**

Users can create queries using free text or Boolean AND, OR, and NOT operators. Wild cards are permitted. The system supports parametric queries. A user can retrieve results from a database table without having to formulate an SQL (structured query language) query. The system allows numeric range, date-defined, and proximity queries.

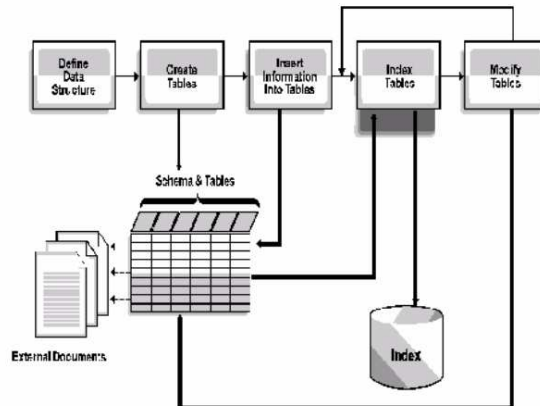
An administrator can configure content to be boosted; that is, appear higher in a results list. A user can also assign “weight” to signal the importance of a word or phrase in a query.

Fulcrum displays a results list to the user. The user can then examine a “hit” in the results list. If that document is germane to the user’s query, the user can use the retrieved document as a new search. Fulcrum’s system worked with less latency when the user highlighted a brief passage of a longer document. Without user highlighting, the Fulcrum system would parse the source document and then query the index for similar documents. A long document used as a query would often return a null set because the only document matching the query is the one the index matches. Fulcrum’s “More Like

This” function is often frustrating for some users who do not understand the nuances of widening and narrowing queries.

## Text Reader: Viewing Documents without Launching an External Application

TextReader is Fulcrum’s name for a conversion program. Fulcrum’s content processing system requires files in a format Fulcrum can parse, index, and zone. TextReader consists of different adaptors; for example, the Fulcrum Technologies Multi-Format widget. The idea is that Fulcrum can process content directly without intermediate file transformation. Fulcrum supports most major document formats as well as Oracle, SQL Server, and Sybase data tables. The Fulcrum API allows a developer to create a customized TextReader script. Support for image and rich media is limited.



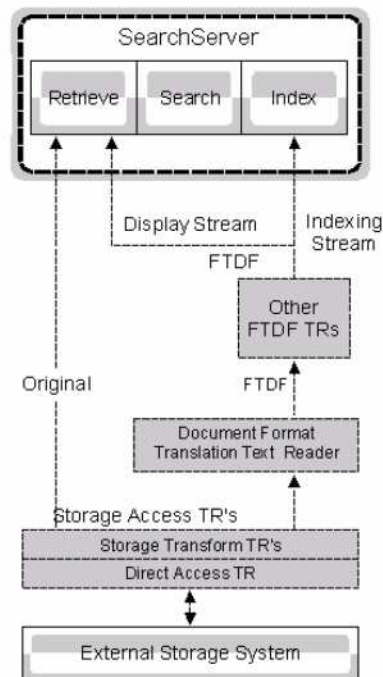
The system requires that schema be developed and content generated so that the system can produce and index and display a meaningful result to the user. (Source: Software Artistry, Text Retrieval Guide, no date.)

## Indexing Options

SearchServer supports a number of indexing options. These include:

- Advanced Metatags and Metadata Support. Additional user-defined Metatags from HTML and XML documents, and from OLE attributes of documents can be indexed into separate SearchServer columns. This metadata can be configured to include or exclude specific tags as defined in a property map file. In addition, document metadata can be extracted into searchable columns at indexing time.
- Literal and Value Indexing Options for Numeric Data and Dates. Numeric data may be searched for an exact match or for a range within specified integer limits. Also, any number of date columns can be defined in a table. These date columns can be searched for an exact match or for a desired range of dates.

- Customizable Index Schema. Developers can configure SearchServer to suit document or database configuration requirements by having index tables created with custom columns with specific data types. Additional columns of different sizes and data types can be used to store tagged information extracted from documents so that the original document does not have to be opened at search time to retrieve it. This allows for the design of extremely high- performance search applications.
- Flexible Word Separator Configuration. Some languages or applications require special treatment when looking for separators. SearchServer can be configured to treat specific types of characters as word separators, or as numerical separators.



The Fulcrum approach requires a low latency system due to the number of interactions among the system's components.

Implementing these options requires investment in editorial activities, as well as facility with the Search Builder SDK. SearchServer does not perform most of these functions automatically. Autonomy, in contrast, emphasizes the automated nature of its pre-processing functions.

The product loads the data and document references into a table; then the next step is to index the table. (Just to be clear: the repository is a place to store the information object. The object can reside on the original machine or in the repository. The index is a separate component. Pointers allow the user to retrieve the original object from its location on the network or in the system.) Administrators can have direct access to the tables, but must learn Hummingbird's quasi-SQL language.

The index files created by this process enable searches to be run against the information contents of the table. During indexing, each document referenced in the table is read into the engine, all of the words in the documents are recognized, and their locations recorded. The only words excluded from the index are those listed in the stop-word list and those found in any portion of a document designated as "non-indexed." The indexing overhead typically ranges from 20 to 40 percent of the size of the original text, depending on the nature of the data.



Hummingbird freshened the Fulcrum interfaces. Displayed is the graphical administration system interface. Early releases of Fulcrum required administrators to edit configuration files. Source: Hummingbird DM Administration Guide, 2004

Special indexing facilities are provided for retrieval of numeric terms. Entire columns or portions of columns may be designated to be "value-indexed." Numeric strings found in these portions are then specially indexed as decimal numbers and can be searched more efficiently. Unlike normally indexed terms, numeric terms that are value indexed can be searched using numeric range operators ("greater than" and "less than"). Term highlighting is unsupported for value-indexed terms.

**“This acquisition further demonstrates our leadership in delivering Internet and Intranet solutions for the enterprise. From its inception, WAIS has been highly regarded as a proponent of open standards, like the ISO Z39.50 standard, the accepted protocol for wide area searching on the Internet. This protocol is ideal for providing distributed, scalable searching solutions for large Intranets. It is a standard that we are committed to bringing fully to the NT platform from the UNIX world, so that corporate administrators can benefit from using the standard across mixed networks.”**—Eric Goodwin, President at <http://goo.gl/fdnGq7>

Tables referencing dynamic, or changing text, must be re-indexed from time to time to incorporate information about new and modified documents and to remove information pertaining to documents that have been deleted. Only documents that have been added, modified or deleted are re-indexed.

SearchServer supports two modes of indexing:

- Immediate indexing, which is the default indexing method, provides instant re-indexing when table content is changed. A differential index is used to accumulate changes to the index information as they happen. Eventually, a periodic indexing process is run to merge the differential index with the primary index.
- Periodic indexing is a “batch” mode method of indexing. Periodic indexing can be scheduled to run automatically through the operating system administration facilities, or can be initiated at any time.

Like Verity and other competitors, SearchServer supports indexing and searching operations independently. Searching is fully supported while an indexing process is underway, and vice versa. This architecture enables SearchServer to support sizable, multi-user environments.

Any unsuccessfully indexed document is marked for subsequent identification. The SearchServer administrator can determine which documents were unsuccessfully indexed and why. A message is generated stating that “Indexing – locating unsuccessfully indexed items” and “Documents – locating unsuccessfully indexed.”

SearchServer also indexes the catalog data for each catalog entry regardless of whether or not the external text has been successfully indexed. This enables the SearchServer administrator to search and open unsuccessfully indexed documents, examine them, and take corrective action. (Other systems often discard unreadable documents.)

### *Development Tools*

Fulcrum was one of the first search vendors to provide software development kits to allow customers and partners to customize certain functions for a particular retrieval problem. Hummingbird has continued this tradition. Fulcrum’s SDK allowed licensees to customize the format and screen appearance for the submission of search queries; to create the user interfaces for intuitive and Boolean searching; and to display ranked result lists, view documents with search “hits” highlighted or print documents retrieved by the search module.

“SearchBuilder” is the Hummingbird’s SDK for building UNIX and Microsoft Windows-based knowledge applications. SearchBuilder can be used for cross-platform development with applications developed in C, C++, Microsoft Visual Basic, and Java. The SearchBuilder toolkit comes complete

with sample application code for prototyping and coding, debugging facilities, and searchable online documentation.

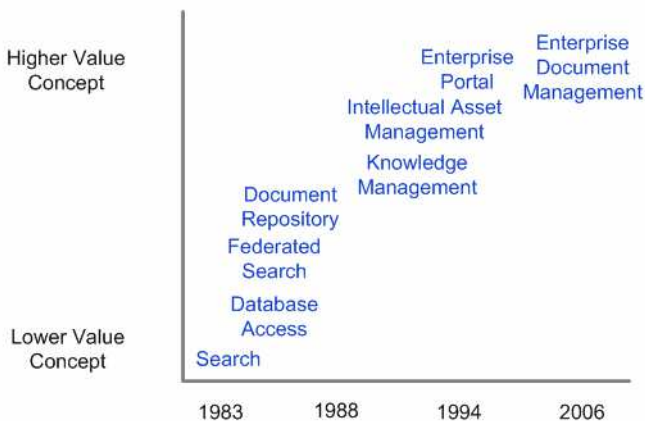
### *Security*

Hummingbird supports a single sign-on (access to all applications and information on a per-user security profile via one password), user authentication (based on existing security profile. These include LDAP, NDS, ADS, NTLM and other industry-standard security models), data encryption, and other security functions that protect the integrity of applications and information.

## **Repositioning: From Search to Knowledge**

---

I find Fulcrum from 1983 to 2006 to be like a cell culture in a Petri dish. The company changed from search (a relatively fresh concept in 1983) to database search, then to federated search, and finally into the esoteric realm of “knowledge” and “intellectual capital.” These are glittering concepts that are quite difficult to define. Search, on the other hand, means that a user keys in a word or phrase and gets a hit pointing to a document with the word or phrase in it.



When the Web revolutionized content, Fulcrum was pursuing “portals” that put information at users’ fingertips and made an organization’s knowledge available to system users. As regulations proliferated, Fulcrum and its new owners embraced risk reduction via enterprise document management. With each shift, the company wanted to close deals and maintain revenue momentum. The Fulcrum approach was imitated across the decades by vendors that learned search was a tough sell.

Based on the information available to the ArnoldIT study team, Fulcrum invested more effort in mapping the core technology to jargon than to filing patents.<sup>5</sup> Fulcrum, despite its innovations, was not patenting its systems and methods. Queries of the patent literature return more references to Fulcrum's search system as a way to implement another's inventor's method.

Fulcrum's senior management put emphasis on chasing concepts that elevated search from a "nice to have" service to a "must have enterprise solution." One lesson from Fulcrum's positioning is that search as a concept is not a high-value software. Shocking as this idea seems, Fulcrum's mixed financial performance, its early sales of a majority of the company to an Italian company and then the final sale to PC Docs indicate that jargon alone cannot sustain an information retrieval company.

Another lesson is that Fulcrum was unable to generate sufficient revenue to stave off financial crunches. As a result, the company became a target for undervalued acquisition.

## **SearchServer before Hummingbird**

Search Server is losing its unique identity in the enterprise search market. Open Text owns a number of search and retrieval systems. Most of these are frozen so that the technical investments are minimized. OpenText seeks to harvest revenue and profit from its search system, not invest in developing new, enhanced methods for enterprise search.

SearchServer asserts that it has "open architecture", which means Web services. Because of this design, "a Hummingbird SearchServer application can grow as organizational requirements change. The burden for enhancing Search Server falls upon the licensee unless the licensee has the resources to hire OpenText engineers to implement a customized solution.

Is Search Server scalable to multi-terabyte databases and sufficiently well-architected to support thousands of simultaneous queries? The answer is, "If the licensee has the money, Search Server can be adapted to almost any enterprise search requirement." The better question is should an organization license 25 year old technology or look for more up-to-date systems?

The company has successfully shifted from its Fulcrum architecture to a Web services approach. Licensees will find that hooking SearchServer into other enterprise applications is easier. Furthermore, the new structure makes it easier to use SearchServer with portal, document management, data integration, search, business intelligence, and analytic applications.

Hummingbird has introduced what it calls HCS or Hummingbird Core Services. These are services such as logging, authentication, data access, licensing, scheduling, and entitlements. Each is made available within the

---

<sup>5</sup>. Emails requesting an opportunity to interview the founders were ignored.

---

**“The first is to recognize that managerialism when applied to complex systems will not guarantee success. In complex “chaos” systems such as large ICT projects, more precision and closer management won’t/can’t lead to a controlled outcome.”**—Public Record Office Victoria at <http://prov.vic.gov.au/blog-only/why-ict-projects-fail>

Hummingbird Core Data Store. Access to these services will be via an API called Hummingbird Repository Retrieval Protocol (HRRP), based on HTTP. Releases moving forward will incorporate these features. The basic Hummingbird license includes these core services.

The goal is to make each of Hummingbird’s separate enterprise solutions increasingly seamless. This could help to eliminate the hard boundaries that currently exist between product offerings within ECM suites like Hummingbird’s, and their separate technology areas. Document management, collaboration, portal, network connectivity, and search can all be somewhat partitioned. Hummingbird is positioning its enterprise software writ large – and not just its search solution – as a way to access both structured and unstructured data. A number of search-focused vendors also offer products that can bridge these data gaps – some at lower price points and with products that require less human support.

## **SearchServer after Hummingbird**

As Fulcrum moved forward, the company packaged its software as “the Fulcrum Knowledge Network.” Search lacked the cachet other business jargon connoted. The concept of search was not devalued by potential customers. Search did not impart an aura of importance. Fulcrum was one of the first—possibly the first vendor of enterprise search—to move from buzzword to buzzword in the hopes of finding a way to sell search more easily.

The knowledge concept, as Fulcrum presented it in a news release, was “a software bundle that provided a licensee with an integrated suite that intelligently collects the information users want, from any location, and presents it in a way so that users can take immediate action.” Fulcrum said that the result is a more complete and precise information set for the users, thus paving the way for the creation of actionable knowledge. After 20 years, Fulcrum’s marketing pitch is usable today. I do not think other search vendors copied Fulcrum’s sales pitch. Working independently, other companies in the search business found themselves unconsciously retracing Fulcrum’s journey.

In the course of building on the value of its Fulcrum software, Hummingbird rolled out a “Desktop Search” client. This client allows users to search their email and local files when online or offline. Hummingbird emphasizes that a user can search and access both local and enterprise information from a single interface. Fulcrum offered this functionality for many years. Hummingbird is putting old wine in new bottles. The new client works within Microsoft Outlook. A query returns results from Outlook files locally and on an Exchange server. Fulcrum’s system was able to index Exchange content prior to its acquisition by Hummingbird.

SearchServer receives a new Java interface. Hummingbirds makes changes to the way controlled terms lists are used by SearchServer. The shifts keep



SearchServer compliant with the Z39.50 standard. Other tweaks to SearchServer include automatic language detection and putting multiple languages in a single content repository. Support for compound words in Dutch, Finnish, German, and Korean are added. Hummingbird includes connectors so content in compressed format and XML can be indexed. Hummingbird, however, focuses on usability, a code word for interface changes. The Fulcrum system required system administrators to edit configuration files. Hummingbird adds graphical interfaces. Hummingbird, however, is able to improve index and query processing slightly. The Fulcrum architecture, due to its design, requires significant resources to minimize latency resulting from the numerous servers and processes that are interdependent.

Hummingbird's other upgrades to the early 1980s SearchServer include:

- Improvements to the Fulcrum document summarization system. Instead of forcing a user to browse a long document, Hummingbird's SearchServer allows users to preview large documents with an automatically-generated one-page summary of a hit in a results list. If a search term appears in the summary, that term is highlighted.
- Better Support for XML document search. SearchServer maintains information on the document's elements and their relationships. A user can for the first time restrict the query to specific tags within a document giving the Fulcrum "zones" tag more utility.
- Natural Language Processing (NLP). Hummingbird seizes on NLP as a way to allow users to search without using Boolean syntax. Hummingbird emphasizes that linguistic analysis in conjunction with the concepts extracted from the index improve the search system. SearchServer retains its support for controlled term lists and licensee-provided classification schemes.
- On-the-Fly Clustering. Hummingbird, like Inxight (Xerox PARC), Northern Light, and Vivisimo group query results into broad categories.

Hummingbird retains the repository approach to content. The approach reduces to some degree the network congestion created by Autonomy's and Verity's distributed broker architecture. But repositories require storing processed content and the resulting indexes. The approach adds to administrative and storage costs to the Hummingbird search system. SearchServer acquires the new and changed content over the licensee's network. As a result, the licensee's network can slow if content acquisition is inexpertly scheduled.

## ArnoldIT Opinion

---

Fulcrum's search-and-retrieval system dates from 1983. An organization licensing the OpenText's Hummingbird search system is tapping into legacy technology. The jargon about NLP, distributed processing, multi-language support is less important than the fact that Fulcrum technology is embedded in other systems. Enhancements are made, but significant changes to the 20-year-old system are not evident. OpenText has an opportunity to demonstrate that it can acquire technology, invest in it, and build upon the foundation the acquired system provides. However, OpenText has to support and enhance a number of unrelated search systems. OpenText will have to generate significant revenue to make it possible for the company to make substantial, meaningful improvements to technology that could be considered at the end of its useful life.

### **Possible Drawbacks**

SearchServer can be a usable enterprise search solution. However, in comparison with products available from other vendors, Fulcrum's technology is falling behind in performance, ease of use, and advanced features. Frankly the system is showing its age. Graphical administrative interfaces give the system a fresher, more youthful look. But beneath the make up, the technology is anchored in early 1980s methods. In contrast, Endeca or Fast Search & Transfer, to name two examples, promise licensees newer technology. Search, for OpenText, is an opportunity to sell large, complex systems and services. Search is not the main business of OpenText, the current owner of the Fulcrum technology.

No enterprise search system is without flaws. However, Fulcrum requires considerable manual work prior to installation, during operation, and at upgrade time. To cite one example, consider indexing content. Before indexing, the licensee has to perform a number of separate processes. Once those are complete, then content can be indexed. Certain tasks require transformation of content. Other processes require the manual preparation of dictionaries and knowledge bases.

When customization is needed, programmers will have to set up, maintain, and customize the system. Graphical aids and easily edited templates are limited. Unlike some enterprise solutions from other vendors, Fulcrum's does not "snap in" to other enterprise applications, a claim Google's Search Appliance makes. Fulcrum's system has lacked for decades tight integration with enterprise applications. Compared to Endeca, the system offers fewer and more limited analytic tools to monitor user actions. A licensee can, of course, write code to implement analytics, but this adds to the cost of the system.

Incremental indexing can be slow unless appropriate infrastructure in the form of fast processors, sufficient random access memory, and high-speed storage are available.

Table 2: Fulcrum Search Technology Checklist

Attribute	Verity Asserts	ArnoldIT Comment
1 Platform	HP-UX, IBM AIX, Solaris, Linux, Microsoft	
2 Keyword search	Yes	The system requires dictionaries and controlled term lists. Automatic indexing uses these knowledge bases.
3 Text mining	No	The application programming interface can be used to hook in third party applications
4 Automated indexing	Yes	The system uses controlled term lists. These require a subject matter expert to maintain them.
5 Personalization	Yes	
6 Workflow	No	The application programming interface can be used to hook in third party applications
7 Interface	Graphical and command line	Scripts and original code are required to perform some tasks
8 Hosted service	No	
9 Administrative interface and tools	Primarily command line. Some graphical interfaces are now available.	
10 Application programming interface	Yes	Dozens of "calls" or "hooks" are available to the licensee
11 Professional services	Yes	
12 Security	Yes	The application programming interface can be used to customize security features
13 Connectors	Standard office file types, SGML, and Codd type databases like IBM Informix, Oracle and SQL Server	The API allows other file types to be supported
14 Support for structured data	Yes	
15 Relevance ranking	Yes	Relevance can be tuned via the configuration options
16 Video	No	The system can index metadata about images and videos if available
17 Federated search	Yes	
18 Fielded search	Yes	
19 Content crawler	Yes	Crawling "Web style" was added as a feature comparatively late in Fulcrum's development
20 Price	As low as a few hundred dollars to six figures or more	Pricing was below that of Verity.

To recap, Fulcrum has a number of drawbacks. These include:

- Erratic financial performance
- A complex system that has a reputation to be difficult to configure, maintain, optimize, and upgrade
- Performance issues with content and query processing
- The design “feature” that associates documents in the repository with a table or series of tables.

## **Anticipated Benefits**

Fulcrum and its subsequent owners’ marketing collateral explain that Ful/Text, SearchServer, and the other software products can deliver basic search and support search-enabled applications. These range from customer support systems to managing content in a document repository to meet regulatory requirements.

Fulcrum’s technology is becoming a component of other enterprise applications. Search is a utility, and it is baked in or bundled with a software package. An organization does not have to make a separate decision about search.

At the time of the sale of Hummingbird to OpenText, Fulcrum’s search system is included with Hummingbird’s enterprise applications. The system provides a range of features and is tightly integrated with certain Hummingbird systems. The approach eliminates some of the set up issues associated with installing a standalone version of Fulcrum.

Hummingbird has demonstrated that it is willing to make some enhancements to the Fulcrum system. However, most of these are cosmetic or designed to keep the system somewhat competitive with other vendors’ offerings.

Like other legacy search systems, Fulcrum’s technology requires significant computing resources.

Other benefits of the Hummingbird’s approach include:

- A large customer base that provides a user group that can be tapped for information.
- Extensive customization is possible via the software development kit and the numerous calls available in the application programming interface. For a skilled developer, Fulcrum can be endlessly customized and extended.
- Hummingbird offers professional services. A licensee can get for-fee support.

## Net Net

Fulcrum has been a harbinger and bellwether for the enterprise search industry. The company introduced or claimed it offered functions designed to make search a must-have application. The company's marketers moved through concepts with bewildering speed in an effort to find the combination to a revenue treasure chest. Many vendors entering the market after Fulcrum had to sell out to avoid financial difficulty found themselves following a trail blazed by Fulcrum.

Fulcrum technology is now owned by OpenText, and there is insufficient information to say with confidence that Fulcrum will be brought up to parity with competitive products. Many of these suffer from the same limitations as Fulcrum, so the enterprise search sector is likely to remain volatile for the foreseeable future.

If an organization requires a document management system that includes search, the Hummingbird systems warrant investigation. A looming question is, "Why hasn't SearchServer become the dominant search technology?"

The short answer is, "Priorities." Search is a utility function for other, higher value software and systems. Not surprisingly, investment in 1980s technology has been greeted without much enthusiasm by Fulcrum's numerous owners.

What is truly remarkable is that a search system that is decades old is still available today in a form that is essentially unchanged. Remarkable.

Stephen E Arnold

Minor edits to a rough draft on December 10, 2013