

# 80-20 One Search

*Technology White Paper*





# 80-20 One Search

## *Technology White Paper*

4.....	Executive Summary
4.....	Search 100% of Your Information
4.....	Email: The One Search Difference
4.....	80-20 One Search
5.....	80-20 Discovery
5.....	80-20 Retriever
5.....	80-20 Meta Search
5.....	One Search Integration
6.....	User Interface
7.....	Architecture
7.....	Local Information
7.....	Enterprise Information
8.....	Security
8.....	Performance
9.....	Scalability
10.....	Enterprise Deployment
10.....	Underlying Technologies
10.....	Search Approach
10.....	Full Text Search
10.....	Conceptual Search
12.....	Dynamic Taxonomies
13.....	Connectivity, Compatibility and Platforms
13.....	Conclusion
14.....	Legal Notices

# Executive Summary

The 80-20 One Search suite allows users to find any information in an organization regardless of where it is located. The One Search suite provides powerful, easy to use search that covers the desktop, the enterprise and legacy systems. Combined with advanced technology that automatically and dynamically categorizes information, One Search dramatically reduces the cost and time associated with search.

## Search 100% of Your Information

As any enterprise search vendor will concede, indexing email at an enterprise level is a huge infrastructural challenge. Over 60% of an organization's unstructured information lies within email. It is highly dynamic and requires constant indexing. As a compute intensive process, solutions that employs centralized hardware are cost prohibitive. This reason alone explains why Microsoft Exchange administrators invariably disable server side indexing - it doesn't scale.

## Email: The One Search Difference

80-20 has taken a novel approach to the above problem. With significant and underutilized computing power available at the desktop, One Search employs unique distributed client/server architecture. This spreads out the massive computing task of organization-wide indexing and search, providing a highly scalable solution without unwieldy infrastructure costs.

In addition to solving the problem of server load, housing the email indexes locally has the additional benefits of security and mobility. Email indexes are unlikely to be of use to anyone else except their owner, and storing them locally guarantees the privacy of the index information. It also means that the client component of One Search is fully accessible to users on the road. Neither of these benefits can be delivered by a server centric solution.

This information below presents an overview of major product components in the One Search suite, a review of the architectural themes and details of the underlying technologies on which the suite of products is based.

## 80-20 One Search

The One Search suite is comprised of three products, Retriever, Discovery and Meta Search, which together provide ubiquitous information retrieval across the desktop, the enterprise and foreign data repositories.

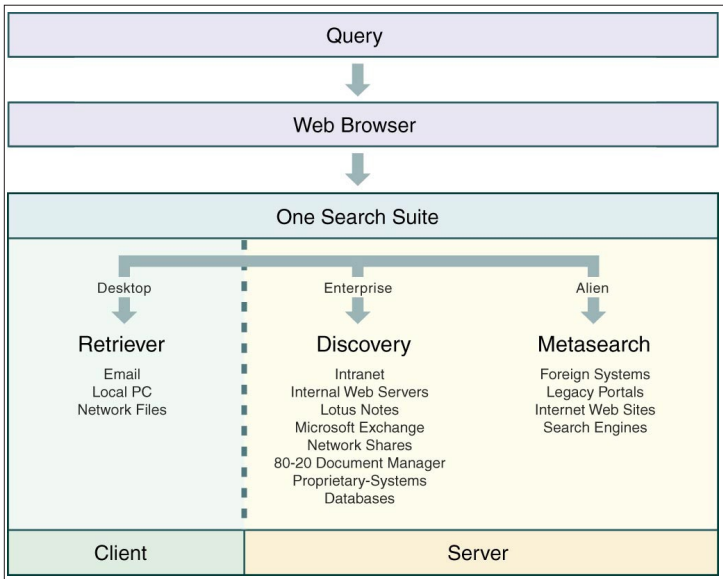


Figure 1 - One Search Suite of Products

Discovery, Retriever and Meta Search are targeted at managing unstructured data repositories. These repositories vary in their physical location, type of information they contain and the degree of control the enterprise can exercise on the information.

## 80-20 Discovery

Discovery is the foundation for our enterprise search services. The search engine hooks into standard repositories such as Intranets, Web servers and network file shares, as well as proprietary repositories such as Microsoft Exchange, Lotus Notes and 80-20 Document Manager. Discovery also provides an extensible framework for custom development and integration.

## 80-20 Retriever

Retriever delivers information retrieval at the desktop. Residing locally, Retriever provides full text indexing functionality on personal email and files stored on local hard drives. Retriever is integrated with Microsoft Outlook, indexes POP or Microsoft Exchange based email, email attachments and network file shares.

## 80-20 Meta Search

Meta Search provides a gateway to legacy information systems, consolidating search results from alien search technologies for direct presentation to the user. As a broker of search services, Meta Search enhances the information provided by legacy systems by generating dynamic taxonomies from search results. These present a structure to the user that allows them to quickly navigate through potentially large amounts of information from disparate sources.

# One Search Integration

The services provided by Retriever, Discovery and Meta Search are combined in the One Search suite to provide integrated search capabilities across all information sources. All components of the suite are web compliant and fully customizable by the user. One Search can be deployed in its entirety across the enterprise or as select components on a user by user basis. The One Search suite is compatible with several industry leading portals and is easily modified from the standard portal, shown below:

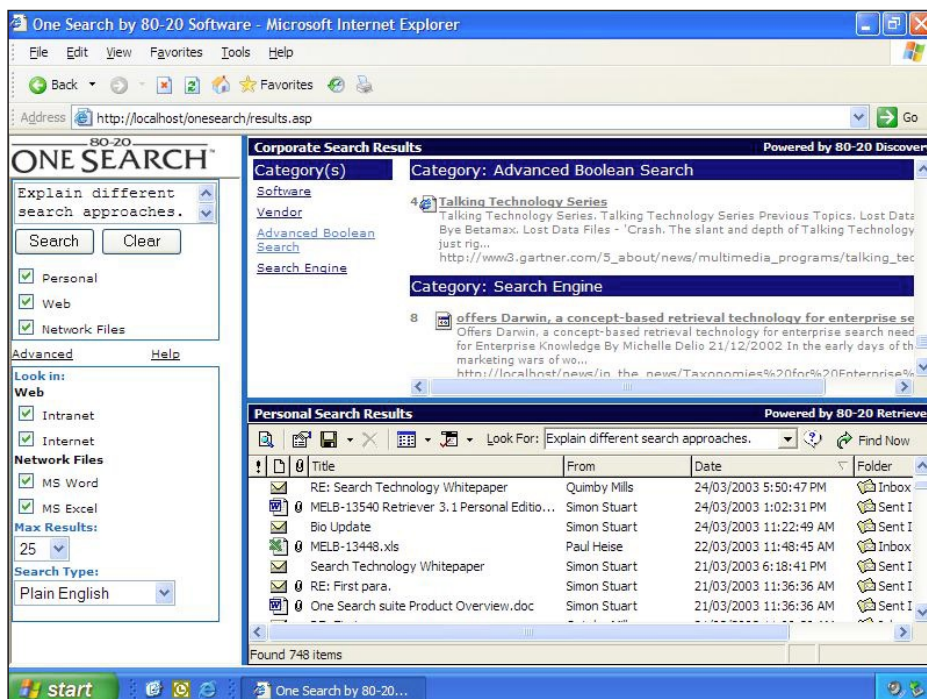


Figure 2- One Search Portal

# User Interfaces

Enterprise search is accessible through either the web browser or, for custom applications, via XML. The thin client approach supports generic web browsers with HTML 2.0 support, as well as advanced browsers such as Microsoft Internet Explorer 6.0. Auto detection of browser capabilities lets Discovery provide the user with features such as in-place viewing of documents in their native format, as illustrated below with a Microsoft Word document:

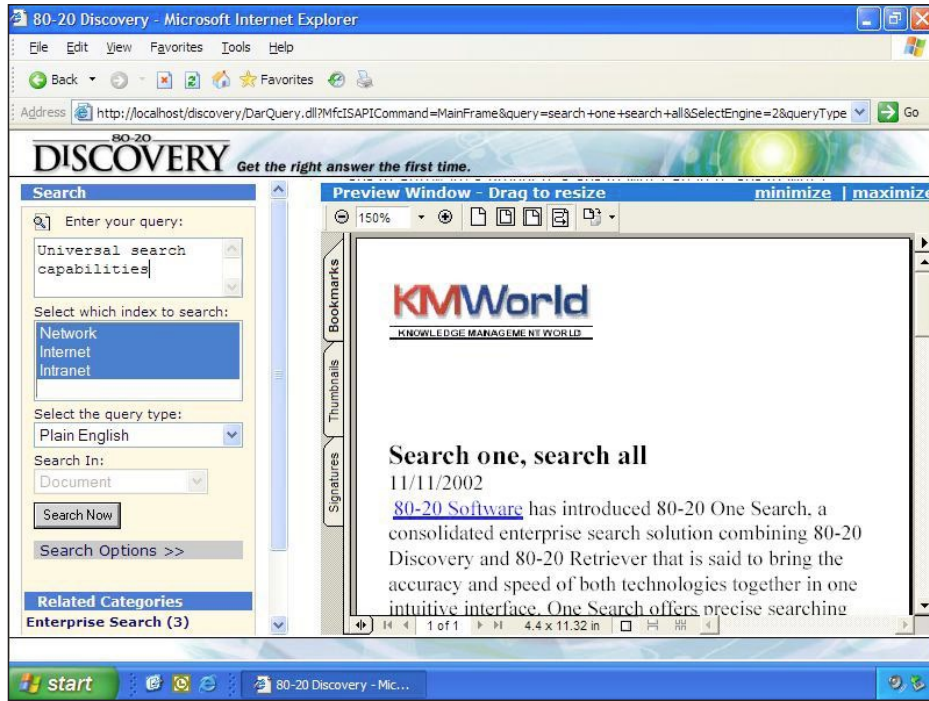


Figure 3 - In-Place Document Viewing with Discovery

Desktop search is also accessible through a thin client and is tightly integrated into Microsoft Outlook as a COM add-in. Like Discovery, Retriever allows users to view documents and attachments natively in place, using Retriever's preview window (see below).

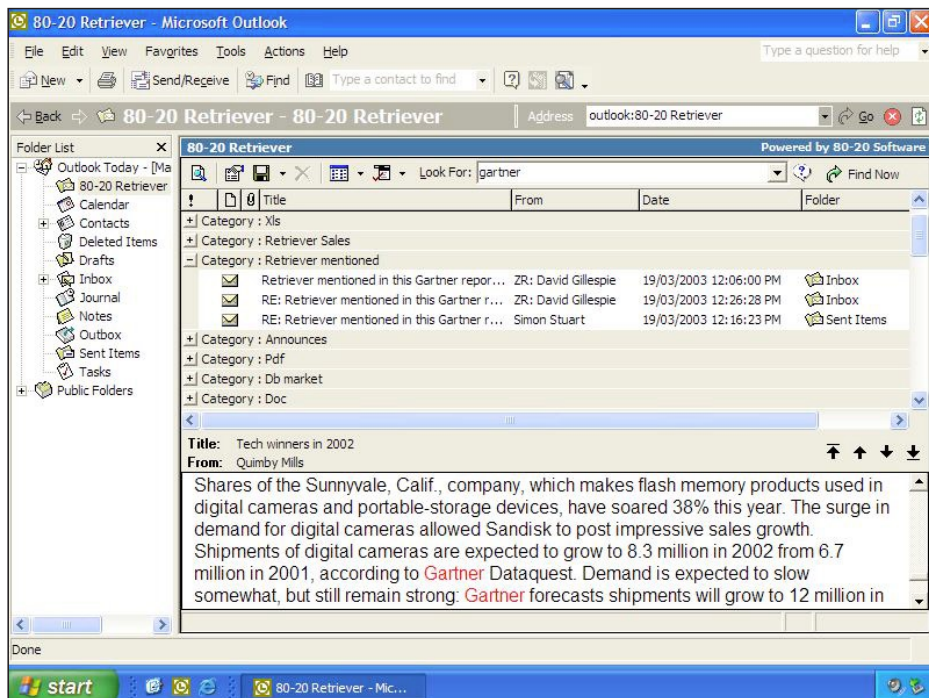


Figure 4 - 80-20 Retriever within Microsoft Outlook

For Portal applications such as IBM WebSphere and Plumtree, Retriever integrates seamlessly into the browser as a drop-in ActiveX control with a standard COM interface for executing queries.

## Architecture

The One Search suite employs a distributed client/server architecture. Desktop search is distributed to the client for performance and mobility, whereas enterprise search remains centralized at the server. A simplified illustration of this architecture is presented below:

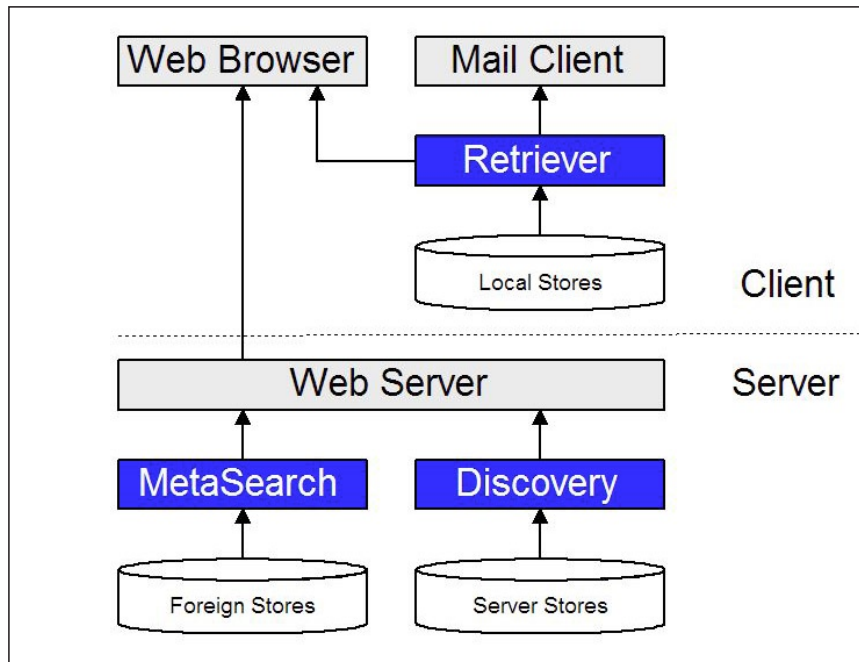


Figure 5 - One Search Distributed Client / Server Architecture

The main design goal of the One Search suite is to provide a single point of access for information retrieval across the desktop, throughout the enterprise and throughout alien systems. Practical considerations of such a goal include mobility, scalability and security, and the degree to which these need to be supported is tied up in the proximity of information to the user.

### Local Information

Desktop information, including email, is volatile, frequently accessed and baggy. The One Search suite maintains desktop indexes locally, addressing both security and mobility by guaranteeing 100% availability and physically preventing unauthorized access. Distributing the computational load of indexing an organization's email to each PC utilizes the idle processing power of desktop computers. Consequently, this alleviates the need for a high cost server infrastructure and scales linearly with the number of users in an enterprise without additional cost.

### Enterprise Information

In the enterprise, the One Search suite addresses integration and compatibility in addition to security and scalability. 80-20 Discovery indexes web servers, network file shares, Microsoft Exchange public folders, Lotus Notes and 80-20's document management system as part of an open connector architecture. A connector API also allows users or third party providers to interface their proprietary systems with Discovery. The presence of new Discovery connectors is detected dynamically, alleviating the need to install additional software or interrupt server operation.

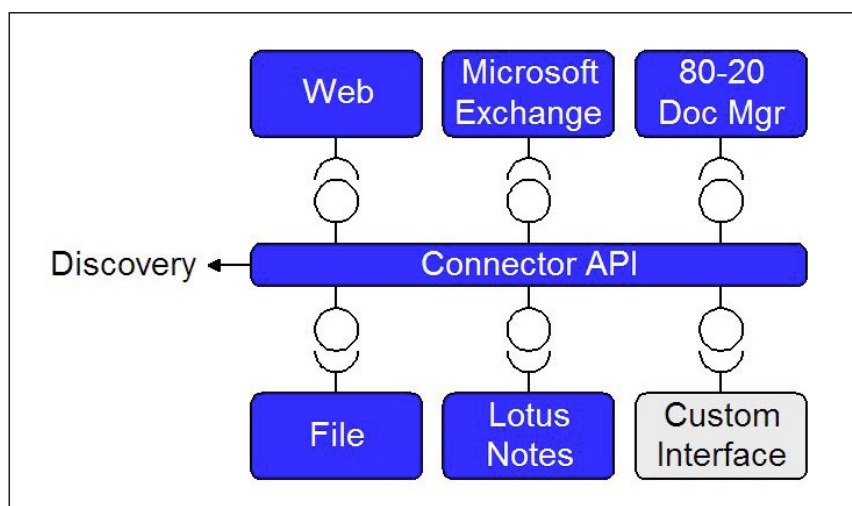


Figure 6 - Discovery Connector Interface

Discovery provides document level security on all its connectors including its custom interface, ensuring access to protected documents is mirrored in Discovery as it is in the source repository.

## Security

Discovery provides native support for Microsoft Windows authentication. It supports Active Directory, NT Domains and NT Workgroups, using the highest level of authentication present at the server.

Document level security is implemented using NT ACL's where support is implemented by the interface provider (e.g. File System, Exchange, 80-20 Document Manager, etc.). Document level security is implemented in the Lotus Notes connector by way of a Notes database that maps NT IDs to Notes security objects, including users and groups. Nesting of groups and other Notes security objects are supported.

Custom interface support for document level security is implemented by the connector provider via an authorization function. When documents are indexed, the connector provider returns document security information in addition to the document itself. The security information in Discovery is stored as a variable length BLOB (Binary Large Object). When the document is later identified during the search phase, the BLOB is passed back to the connector provider along with the user credentials. The provider then validates the credentials against the BLOB and indicates whether authorization is granted. If authorization fails, Discovery continues to search the index for further relevant results.

This security agnostic approach to implementing document level security allows Discovery to authenticate users and authorize document access by signing on through login or through the browser. This facilitates transparent information access to users from any information source.

## Performance

The approach used in implementing document level security has a marked effect on query time performance. The three most common options in implementing a security structure are: pre-fetching document permissions, post-processing, or a hybrid involving caching permissions. Discovery is oriented towards pre-fetching document permissions, although Discovery's connector interface supports post-processing if the connector provider implements this functionality. The benefit of post-processing is in the guarantee that a change in document permissions are reflected immediately in search results. This comes at the cost of run time performance, which significantly limits scalability. The system that provides each indexed document must be available at the time a query is executed to authorize access to the document. Permission caching attempts to alleviate this problem but also suffers from scalability limitations as cache latency negates the advantage that post-processing document permissions delivers.

Discovery's strategy of pre-processing document permissions provides the most scalable solution where security is essential. Eliminating the need to return to the document repository at query time provides a significant performance increase (in addition to other benefits like reduced server load). In addition, Discovery's efficient indexing algorithm allows queries to be executed as an index is being built. Discovery indexes can be set to update only modified documents reducing the turnaround time for index updates. When combined with comprehensive scheduling features that allow periodic index updates to be run, maintaining current indexes that scale to millions of documents in a secure environment is a trivial undertaking.

Discovery's extensibility in connecting to proprietary repositories supports the pre-processing of permissions through a simple interface. Passing a document's permissions BLOB and the user credentials to the connector allows the implementer to determine the most appropriate method for authorizing document access. This is illustrated in the Exchange Public Folder connector which gives the administrator both the option of dynamically authorizing access at query time or pre-processing permissions. This provides maximum flexibility to the administrator depending on the needs of the application.

## Scalability

The One Search distributed client/server architecture lends itself naturally to large scale deployments. With email constituting over 60% of enterprise data, Retriever scales linearly as desktops are added to the enterprise.

For enterprise information, Discovery servers can be installed and running in minutes, centrally deployed by Discovery's remote administration console. Console authentication is provided by the operating system, with encryption applied to all transmitted credentials. Discovery also broadcasts read only, XML based status updates on indexing and index operation, which can be monitored by the console visually. These can also be collected by reporting applications or used to generate administrator notifications such as cell phone alerts.

Discovery servers can be clustered together into server farms to manage data in the terabyte range, with queries distributed by 80-20 Meta Search. For high query loads, redundant arrays of Discovery servers can be arranged and combined with separate staging and production areas. Discovery's advanced event scheduling features enable periodic index updates and rebuilding to assist with staged environments.

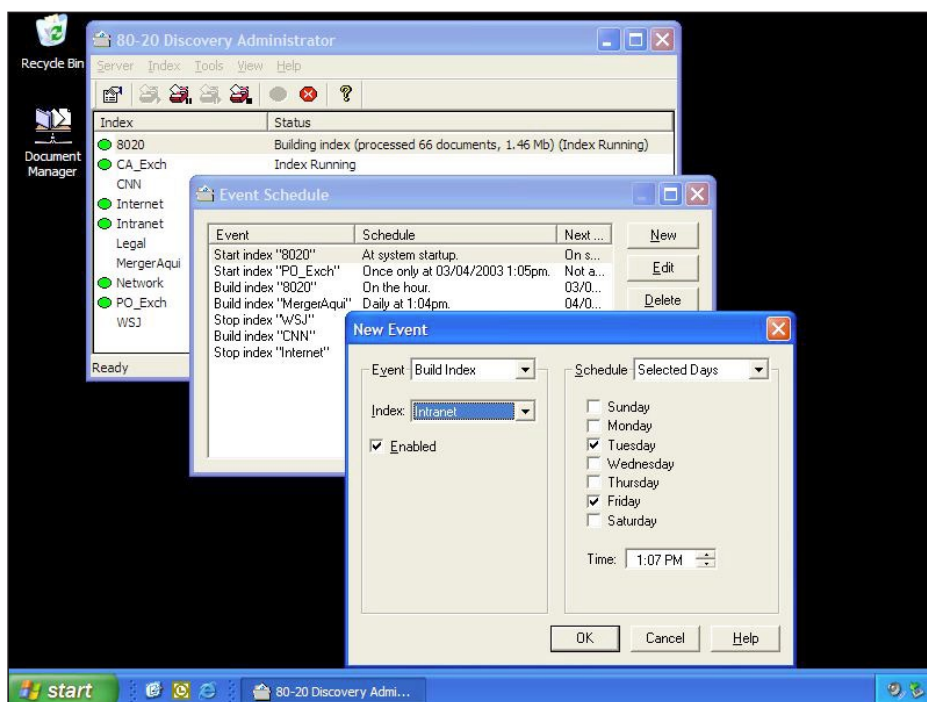


Figure 7 - Discovery Event Scheduling

For customers with heavily distributed data, global search requirements and low speed interconnectors, Meta Search fields and lists query results from remote servers. Results are consolidated and categorized locally in a manner completely transparent to the end user.

One Search technologies also employ well known techniques such as thread pooling and caching to managing high loads in both an indexing and querying environment where millions of documents are the norm. Combined with the extremely high performance its neural network, the powerful distributed client/server architecture allows One Search to scale to meet the needs of any enterprise search requirement.

## Enterprise Deployment

Retriever allows rapid deployment in an enterprise environment by supporting technologies designed to assist administrative staff, such as group deployment features of Windows Installer. Where Exchange based mail is implemented, Retriever can optionally utilize offline stores to completely eliminate mail server interaction, serving to reducing index build time and server load. For verification, all installable Windows components are digitally signed.

## Underlying Technologies

One Search suite combines powerful search technology with an extensive range of repository connectors and advanced corporate deployment features. These are described in detail below.

## Search Approach

At the core of the One Search suite lies its information retrieval technology. One Search is based on 80-20's search platform, codenamed 'Darwin', giving users a choice of both a conventional keyword engine as well as an advanced conceptual engine to perform specific and fuzzy searches respectively. Discovery, Retriever and Meta Search are based on the Darwin platform, which is fully accessible via an open application programming interface (API). Darwin is available for OEM applications.

## Full Text Search

Both Discovery and Retriever provide full text search capabilities within the realm of their keyword capability. Users can search for individual terms, phrases, use Boolean operators, limit the searches to specific domains within the corpus or a combination of the above. Keyword searches enable the user to quickly locate documents based on specific criteria and to perform exhaustive searches of large document sets to confidently identify all relevant information.

## Conceptual Search

Discovery provides additional capabilities over keyword search with its advanced concept search. Concept search goes a step further than most search technologies to identify a collection of key concepts that represent a document, then draw a complex series of associations to documents also containing similar concepts. Discovery iterates through this process as documents are added, building a network of relationships between documents that are drawn upon in later stages. The assessment of document similarity is a key feature of concept searching because it allows Discovery to return relevant documents that may not contain any terms in the original query. This approach lets users locate important information without needing to know the exact terms that appear in the document. This also eliminates the need to keep reformulating queries in the hope that all the relevant information has been found.

Three key technologies make up the concept engine: Natural Language Processing, Feature Extraction and a Self Generating Neural Network. The following sections present these technologies in more detail:

## Natural Language Processing

Discovery caters for both search power users and novices. Power users can take advantage of advanced search options provided with the keyword engine, whereas general users are more likely to enter a query just as they would ask a question a person. The Natural Language Processor (NLP) is specifically designed for the latter. The NLP translates simple or complex human readable questions into a collection of weighted terms and phrases, each identifiable with a formal part of speech (e.g. noun, adverb, preposition). This patented translation phase aims to identify context in a query assisting in determining the importance of key terms, and is a precursor to searching the index.

The NLP module contains a rule based engine that is prepared with a set of language syntax patterns formulated by linguistic experts. The key to the power of the NLP engine is not only its ability to apply language rules to a query, but also to account for the loose adherence people have to the rules of language, making suitable adjustments to the information sent to the concept engine. A simplification of the linguistic rules and context assessments made by the NLP is illustrated in the following example:

*Discovery is set up to provide shopping mall visitors with an electronic information desk. The visitor types in 'Where can I buy a cup of coffee?' The phrase 'cup of coffee' consists of the sequence, 'noun-preposition-noun.' However, in this context, 'cup' is a unit noun. Familiar with the sequence and the possible parts of speech that can be applied to each word, Discovery subsequently discounts the noun 'cup'. Consequently, the search engine identifies 'Coffee Shops' as direct hits and eliminates retail outlets describing cups or other synonyms that the search engine may otherwise try. In the absence of NLP a keyword engine would be unable to make this distinction.*

One Search also adds further value to the NLP by allowing broad based customization for domain specific business terms in the NLP's dictionaries.

## Feature Extraction

The process of identifying concepts within a document differs from the keyword approach both in the way that terms are identified and the value applied to them.

The feature extractor process passes a 'sliding window' over the document, looking for fixed length patterns of characters. These patterns are known as n-grams, which distinguishes them from words as used in a keyword engine. This n-gram approach provides the benefit of identifying patterns in multi word phrases such as 'General Electric' without succumbing to the shortcomings of a language dependant process known as word stemming.

*Stemming is an approach frequently used by keyword engines to find documents containing all morphological variants of a word, e.g. 'Electric' and 'Electricity'. Rule based implementations occasionally produces maverick results, e.g. mistaking 'Organ' and 'Organization' as well as 'Execute' and 'Executive', so that a search for 'Executive organisation' will return documents that contain 'Organ' and 'Execute'.*

*In addition, stemming makes no allowances for context differences resulting from word order. E.g. 'Computer network' is not distinguished from 'Network computer'. N-gram analysis distinguishes between these two terms due to their sequence.*

The concept engine assigns weights to patterns found in a document, both with respect to each other and also in context to the frequency with which they occur in the overall document corpus. When a collection of patterns establishes them as relatively novel, overlaps in the novel patterns are examined for contained words and phrases that subsequently become concepts for that document. This weighting process continues to be updated as the corpus changes, so that the relevancy of a document is always maintained in the context of all the other documents. The novel concepts identified within a document are then submitted to Discovery's neural network for storage, persistence and retrieval.

In the process of identifying document concepts, the feature extractor also generates a document abstract containing sections of the document that best represent the key terms contained within.

## Neural Networks

The principal goal of the neural network is to provide a mechanism for retrieving related information. The neural network accepts documents from the feature extractor and arranges them in groups of similar documents. During the retrieval phase, navigation through the network is fast and efficient, providing access to the 'winning' document as well as related documents in a close proximity.

The advantage of using a neural network over a common inverted index is in its ability to provide proximity between search terms. The neural network can locate documents with similar concepts to the best document, even when the similar documents do not contain search terms in the user's query. When a search request enters the neural network, it passes through a sequence of 'nodes' to the optimal 'leaf' node and then the network considers other nodes in close proximity. This approach is not possible with a keyword index as two arbitrary terms have no relationship to each other.

This sequence is extremely fast and thus well suited for large datasets consisting of millions of documents. In addition, once the 'winning' document is found, the engine need only look in the immediate proximity to find the next most relevant document. This makes finding the top 10, 50 or 100 documents only one or two nodes away and thus, very fast.

## Dynamic Taxonomies

One Search technology dynamically classifies documents as searches are performed, generating a taxonomy on the fly that groups together search results with similar concepts. An example of this dynamic taxonomy generation, available in all One Search products, is seen here in the 80-20 Web Site in response to a query on 'Press Releases':

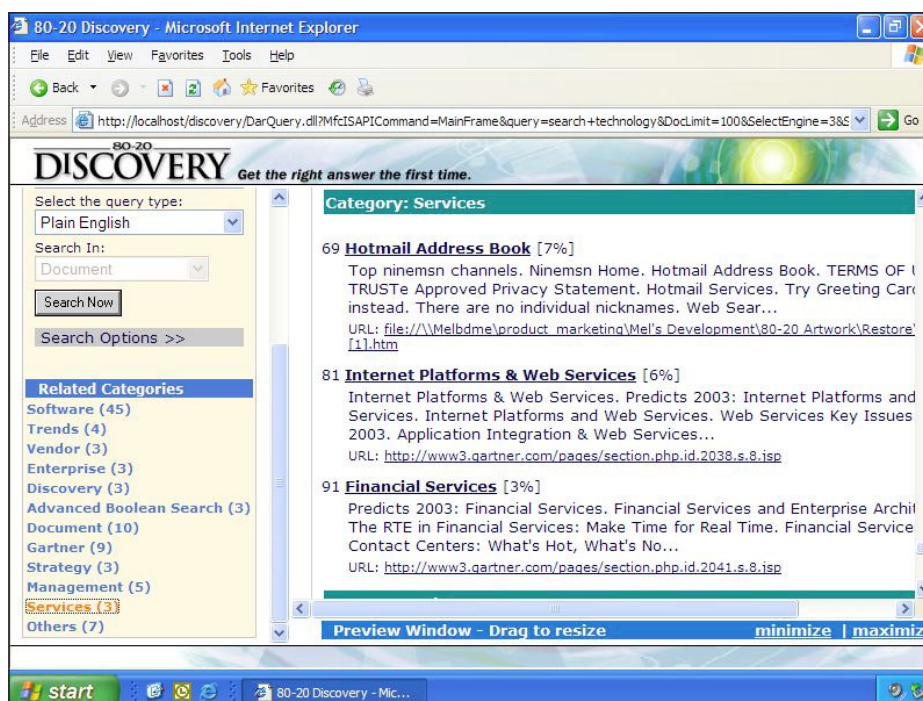


Figure 8 - Automatic Classification of Search Results

The creation, maintenance and enforcement of manual taxonomies in organizations is costly, resource consuming and difficult to enforce. Manual taxonomies age with time and in any event, individuals are highly unlikely to tag their personal email. The One Search approach is to examine the document for its underlying concepts in context of both the query and the other returned

documents. In this way, taxonomies are generated dynamically and the concept chosen for a document will change depending on the query and documents considered to be similar.

The information used to generate dynamic taxonomies is first obtained during the feature extraction stage. As feature extraction can be intensive, pre-processing optimizes performance at query time. In addition to generating a human readable abstract, the feature extractor also stores a more detailed summary of a document's concepts for the specific purpose of automatic categorization.

During the query phase, the search engine returns a collection of the best search results. The component responsible for taxonomy generation, the clusterer, requests the document summary generated by the feature extractor. Using this as a guide, each document is plotted on a concept map and documents containing common concepts are grouped together to form a cluster. The best term for the cluster is identified, and the collection of cluster names is presented to the user to allow them to quickly navigate to related search results.

Through the clustering process, One Search negates the need for a costly manual taxonomy by ensuring that the taxonomy presented is drawn from the actual content of the documents rather than a fixed structure that needs to be constantly maintained.

## **Connectivity, Compatibility and Platforms**

Retriever has the capacity to index both POP and Exchange based mail, PST and archived mail stores as well as local file systems and network file shares.

Standard Discovery connectors allow indexing of Web servers (Both HTTP and HTTPS using SSL), network file shares, Microsoft Exchange 5.5 and 2000 Public Folders, 80-20 Document Manager, Lotus Notes and proprietary systems via the open connector architecture. Full document level security is supported on all connectors. Exchange Public Folder access can be provided through both Microsoft Outlook Web Access (OWA) or, as a secure option, through 80-20's digitally signed ActiveX control in combination with Microsoft Outlook.

The Discovery web crawler supports extensive customization options for site selection and navigation. This includes explicit site traversal limits both across and down into linked sites, as well as the ability to specify complex regular expressions for address resolution. Discovery is fully compatible with web proxies.

One Search products are capable of processing all common document types including Microsoft Office, PDF, text and all HTML variants. Discovery supports any document type that can be filtered using Microsoft IFilters.

One Search supports commonly used and industry standard interfaces such as: C++ for Discovery's custom connector and the Darwin SDK, COM for Retriever queries, and XML for Discovery queries, search results and server monitoring.

One Search products are supported on the Windows NT, 2000 and XP platforms. One Search has no prerequisite software, installs standalone and is side-by-side compatible with Microsoft server products such as Exchange and SQL Server.

## **Conclusion**

This document presents an overview of the major product components in the 80-20 One Search suite, a review of the architectural themes the suite addresses and details of the underlying technologies on which the suite's products are based.

For more information please visit <http://www.80-20.com> or email us at [sales@80-20.com](mailto:sales@80-20.com)

## Legal Notices

The information contained in this document represents the current view of 80-20 Software on the issues discussed as of the date of publication. Because 80-20 must respond to changing market conditions, it should not be interpreted to be a commitment on the part of 80-20, and 80-20 cannot guarantee the accuracy of any information presented after the date of publication.

This White Paper is for informational purposes only. 80-20 MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AS TO THE INFORMATION IN THIS DOCUMENT.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of 80-20 Software.

80-20 has patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from 80-20, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.



### **Worldwide Headquarters**

80-20 Software  
10 Queens Road - Level 8  
Melbourne, 3004  
Victoria, AUSTRALIA  
Phone: +61 3 9866 8755  
Fax: +61 3 9866 8766

### **U.S. Headquarters**

80-20 Software  
2055 112th Ave, Suite 120  
Bellevue, WA 98004  
USA  
Phone: +1 426 739 6767  
Fax: +1 425 739 9228

**Sales and Product Information**  
sales@80-20.com

**Partnership**  
partners@80-20.com

**Technical Support for Existing Customers**  
support@80-20.com