



## THE ISYS SEARCH SOFTWARE DEVELOPER'S KIT ISYS:SDK TECHNICAL OVERVIEW

---

© 2008, ISYS® Search Software, Inc.

### ISYS Search Software Worldwide

---

#### **The Americas**

ISYS Search Software Inc  
8765 East Orchard Road  
Suite702  
Englewood, CO 80111  
USA

Phone: +1 303 689 9998  
Email: [info-us@isys-search.com](mailto:info-us@isys-search.com)  
Fax: +1 303 689 9997

#### **Australia & Asia Pacific**

ISYS Search Software Pty Ltd  
Suite 102,10-12 Clarke St  
Crows Nest NSW 2065  
Australia

Phone: +61 2 9439 5800  
Email: [info-au@isys-search.com](mailto:info-au@isys-search.com)  
Fax: +61 2 9439 8569

#### **EMEA**

ISYS Search Software (UK) Ltd  
The Steam Mill  
Steam Mill Street  
Chester CH3 5AN  
United Kingdom

Phone: +44 1244 893 132  
Email: [info-uk@isys-search.com](mailto:info-uk@isys-search.com)  
Fax: +44 1244 893 322

## Introduction to ISYS Search Engine SDK

This document describes the capabilities of Version 8 of the ISYS Software Developers Kit (SDK). For specific programming details, please refer to the ISYS:sdk 8 Manual, which is supplied with the Kit.

The text indexing and retrieval engine that ISYS Search Software uses in its retail products is available to the custom applications programmer. The search engine embodies the complete functionality of the text indexing and search engine of ISYS, which has been in continuous development since 1988, and is hence mature and proven.

### Features

An ISYS index consists of references to as many as 64 million indexed documents. A document normally corresponds to a single word processor, spreadsheet or other text file, but may also correspond to a single instance of a database record or email message, or any other textual item of information. Multiple indexes can be chained together for simultaneous searching. The ISYS Search Engine offers the following key features:

- Compatibility with 200+ document formats, connectors, containers and more
- Powerful retrieval language, with rich operator set
- Extremely high performance indexing and retrieval, both in speed and index size
- Compatibility with C/C++, Visual Basic, ASP, Borland Delphi, Cold Fusion, Java and .NET.
- Small, easy-to-use Basic Retrieval API set
- Advanced API calls to provide additional functionality
- “External Access Module” API for access to additional file formats, including proprietary databases, or custom security schemes
- Access to all underlying API features used in the retail versions of ISYS
- COM, DLL, Java or .NET interfaces

The ISYS Search Engine is available for Windows 2000 (SP4), 2003 Server, XP, Vista and Linux.

The ISYS Search Engine emphasizes speed and scalability across volume, while at the same time being easy for the application developer to integrate. Most OEM applications using ISYS are able to achieve everything they want with the use of only a dozen or so ISYS calls, therefore the time spent mastering the ISYS API set does not have to be substantial.

### Who uses the ISYS Search Engine?

ISYS has been successfully integrated with applications such as:

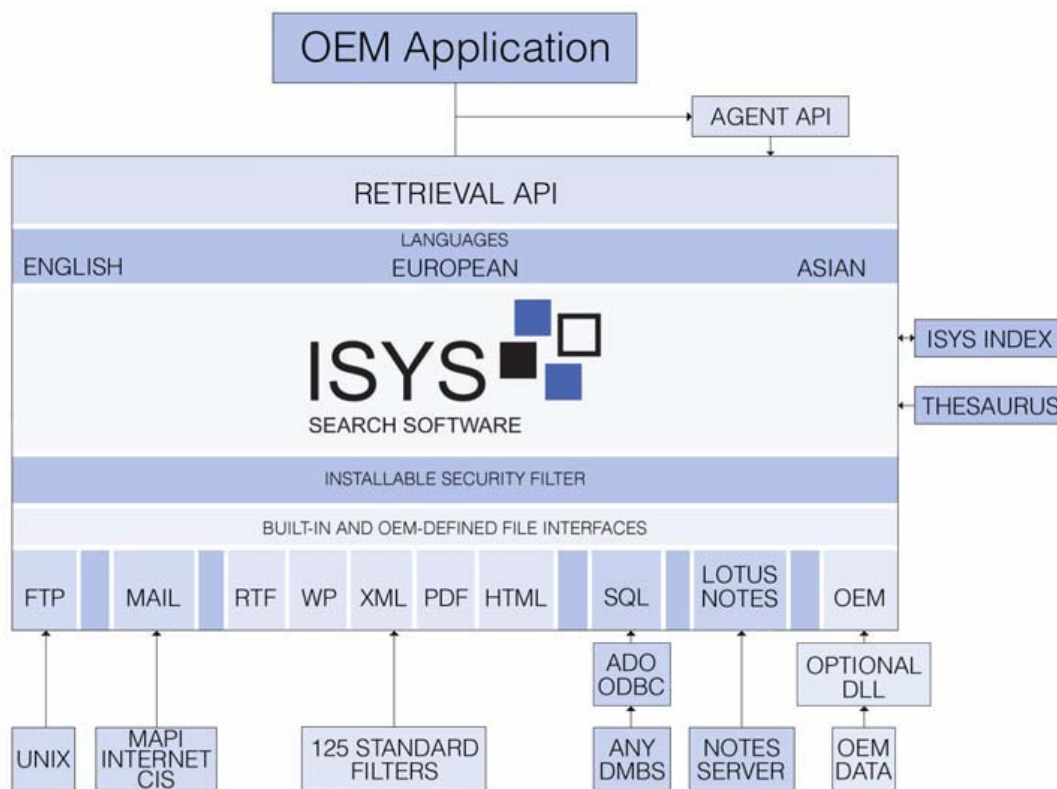
Corporate Portals	Searching company documents, serving legacy information, interfacing to office systems
Help desk systems	searching technical notes, problems, descriptions and resolutions
Litigation Support Systems	searching transcripts, case notes and brief banks
Human Resources systems	searching resumes, interview notes, review information
Television support systems	indexing and searching news as it happens
Document management systems	providing full-text search of managed documents
Legal indexes	trademarks and patents (130 gigabytes, 250 million records, 10 million documents)
Library systems	indexing structured records for unstructured access
Custom web servers	building your own custom server, but still providing advanced search
In-house applications	responding to users demands for cross-table, cross-database, cross-media searching
<hr/>	
Media Monitoring	real-time scanning of media items against lists of client requirements to generate real-time alerts
E-commerce	providing full-text search capability across large product databases

Any application that has a need to find information rapidly without dependence on structure can use the ISYS Search Engine to provide instant access to literally any information.

No matter whether the information resides in traditional third-party files such as word processors, SQL indexes, proprietary format files, or a proprietary information space with no direct physical representation, the ISYS Search Engine can index and retrieve it.

### Product Architecture

The ISYS Search Engine is a true “drop-in” text retrieval software component. It is a highly optimized set of WIN32 DLLs and Linux shared objects that consistently win comparative reviews for both functionality and performance. It is also the same engine and API that ISYS Search Software uses to create and maintain its ISYS range of retail products.



The program you write talks to the ISYS API. The DLL reads documents; it reads and writes ISYS indexes

## Document Formats

ISYS does not require you to translate your files into a proprietary format before indexing them. Instead, it reads files in their native formats, allowing you to use your favorite word processor, e-mail product or database.

The standard ISYS engine can recognize more than 200 different formats, connectors and containers. With the External Access Module feature, ISYS can even provide full functionality on your proprietary data files through a simple seven-entry-point interface DLL. ISYS can access any ODBC-compliant database server, as well as ADO and BDE, and provides extremely high performance XML support.

## COM or DLL interface

The primary means of access to the ISYS Search Engine is via DLL function calls, which enables the engine to be called from any language which supports calling WIN32 DLLs or Linux shared objects. The DLL call interface provides access to all engine features, and provides the greatest degree of control.

For use from scripting languages such as ASP or Cold Fusion, or where multi-threading is an issue, or just when it is convenient, you may also make use of the COM/Java interface to the ISYS engine. The COM interface provides access to basic retrieval functionality. It is not as flexible or as detailed as the full DLL call interface, but provides ready access to the core retrieval API.

## .NET or Java interface

The ISYS Search Engine is also callable from both .NET and Java programs, with sample code provided in Java, VB.Net and C#.

## Interface

The ISYS Search API consists of function calls arranged into several logical groups:

- Basic Retrieval
- Advanced Retrieval
- General Utilities
- High-Level Indexing

- Low-Level Indexing
- External Access Modules
- Result-List Manipulators
- Annotations
- Concept Trees
- Synonym Rings
- Named Sections
- Intelligent Agents

The SDK makes it easy to access these calls: simply include a standard header file in your project. The header file provides all the required constants, data structures, and function definitions your application needs to control the ISYS Search Engine. The Basic Retrieval calls provide the minimum functionality required to set up a full text retrieval system:

Initialize ISYS:	Initiates communication between your front-end program and the ISYS Search Engine.
Open an Index:	Instructs ISYS to open the files associated with an ISYS index.
Get Index Information:	Obtains information about the currently open index.
Perform a Query:	Accepts a query string coded in the ISYS Search language and returns the number of words and documents found. Your application may terminate a query while it is progressing.
Natural Language Query:	Accepts a query in plain everyday English and returns the number of words and documents found.
Sort Documents:	By default, documents are sorted in the order they were indexed. This call changes the sort sequence to be by date, size, number of hits, full path name, file name, or relevance.
Get Document Information:	Given a document reference number (its position in the list returned by performing a query), this call returns useful information

	about the document such as document name, number of words in document, and number of hits in the document.
Open Document:	Given a document reference number, this call opens a document.
Retrieve Document Line:	Given a document number and a line number, this call instantly retrieves that line from the file. Because ISYS uses random access to read document files, this call is extremely fast, even on complex file formats such as Word for Windows “fast saved” documents which are stored non-linearly in a complex structure.
Find Next/Previous Hit:	Given a document reference number and a line number, these calls return the number of the next or previous line that contains a found word. With these calls, your application can navigate directly from one hit to the next.
Close Document:	Closes a document.
Terminate ISYS:	Closes the connection between your application and the ISYS Search Engine.

---

Because all access to text is through the ISYS Search API, your application does not need to be aware of the original data formats.

Likewise, ISYS takes care of parsing documents into words. The emphasis is on ease of use by the application programmer, and hence the API is high level and easily callable from high level programming environments. A sample sequence of API calls might be:

#### Open Database

Perform Find “CAT/10/DOG” // Find “cat” within 10 words of “dog”

Get Details of Document 1

Get Details of Document 2

Get Details of Document 3

Open Document 2

Get Line Number of Hit After Line 0

Get Text of Line 347  
Get Text of Line 348  
Close Document 2  
Open Document 1  
Get Text of Line 1  
Get Text of Line N  
Perform Find "JAVA IN LANGUAGES"

Your application may access documents in any order. It may open many documents at once and may access lines within each document in any order. The ISYS Search Engine embeds information in its indexes that allow it to seek instantly to any point in any open file. To access a line in the middle of a 100MB file, ISYS seeks directly to that line; it does not have to read from the top of the file. This functionality is provided even for complex file formats such as Microsoft Word's "fast-saved" documents and external data sources (including large XML data sources). It ensures near-instantaneous performance, even while browsing extremely large documents.

The ISYS high level indexing engine keeps track of file names and file modification dates, simplifying the process of updating your indexes. The ISYS API also provides low-level access to indexing functions, allowing you to control and monitor those functions on a file-by-file basis.

### Search and Retrieval Language

Queries are passed to the API coded in text and expressed in the ISYS search language.

A query may consist of a single word or phrase

**Windows** *or* **Microsoft Windows**

or a series of words or phrases combined with operators:

**Microsoft Windows OR PenPoint**

ISYS supports the following Boolean operators:

AND

Boolean AND: both words or phrases must appear in the same document for the document to be selected.

OR	Boolean OR: locates documents which contain any one of the entered words or phrases.
NOT	Boolean negation: locates documents which contain the first word or phrase, but not the second.
XOR	Boolean XOR: locates documents which contain the first word or phrase or the second word or phrase, but not both.

---

ISYS also supports the following proximity operators:

/n,m/	Must appear from n words before to m words after, within the same paragraph.
//	Must appear in the same paragraph.
\n,m\ 	Must appear from n paragraphs before to m paragraphs after.
\\	Must appear in one paragraph either side.
...	Second term must appear after first term.
EXCEPT	Paragraph-level exclusion operator.

---

ISYS supports “named section” and “labeled paragraph” searching via the operators:

LABEL (Unary operator)	Term must appear at the start of a paragraph.
IN	First term must appear in a paragraph (or section) labeled by second term.

---

Other operators:

TO	Specifies an alphabetic, date, or numeric range search.
( )	Operator precedence.

---

AFTER	Search for documents after a specific date.
BEFORE	Search for documents before a specific date.
GE, LE	Numeric range searching.

---

**Wild Cards:** A wild-card symbol may appear once, anywhere in the word, much like the DOS wildcard character, (e.g. xyz\*abc), or may occur at each end of a word, for example \*date\*.

**Conflation:** Words may also be suffixed or prefixed by the Conflation operator which causes all tense-forms of the word to be retrieved. For example “worked~” would also retrieve “work”, “working”, “workers”, but not “workstation”. Conflation may also be used at the start of a word. Conflation may be selected in either broad or narrow modes.

Example ISYS query: ((Disk or Dasd or Floppy) \\ Portable~) AND computer\*

**Dates:** ISYS includes optional intelligent date handling which can find dates regardless of the format in which they are expressed in the documents or the query. For example, Mar-20-96, 20-03-96, March 20 1996, or even the 20th of March, 1996.

**Numbers:** ISYS optionally intelligently recognizes and indexes numeric quantities, regardless of how they are expressed. For example, the phrase “two hundred and nine thousand one hundred and one” would be decoded into a numeric value and indexed as such. Likewise numeric quantities expressed in terms such as “10 million” or “1 000”.

**CD-ROMs:** ISYS indexes can be specially optimized to take advantage of the performance characteristics of CD-ROMs, resulting in queries executing two to three times faster on this medium.

## Natural Language Query

The retail versions of ISYS are renowned for their unparalleled ease-of-use.

A mainstay of this reputation is the Natural Language Query feature, which allows users to simply type a short statement or question in everyday language that describes their needs, for example:

How do I do a credit check?

Do I need a license to fish for Salmon?

How do I configure the printer?

ISYS applies advanced processing concepts to satisfy the users' needs based on its knowledge of the data it has indexed.

The Natural Language Query feature is available for use in your application through a single function call.

## Synonyms and Thesaurus

The ISYS Search Engine provides for user or OEM-defined synonyms, as well as featuring a built-in thesaurus which can be automatically included in searches.

## Intelligent Search Agent

The ISYS:sdk includes an API to ISYS' Intelligent Search Agent functionality, which lets your applications provide user-level tracking of what new information has been found, and what has already been seen.

## Multiple Languages

ISYS supports searching all major single-byte languages, including French, Spanish, Italian, German, Thai, Portuguese, and also supports Korean, Japanese, Simplified and Traditional Chinese. Asian language material in both MBCS and Unicode is supported, and normalized to MBCS or Unicode. ISYS understands the ideographic and non-ideographic nature of the Asian languages.

ISYS can combine multiple languages in a single query. For example, you can issue a Boolean query which combines French and English, or English and Chinese.

## Fuzzy Searching

ISYS recognizes that data is often sourced from scanned and OCR'd material. While OCR software improves all the time, it is far from perfect, and the cost of manually inspecting and correcting OCR errors can be very time consuming. By enabling the ISYS “Fuzzy precompensation for OCR and typographic errors” feature, ISYS can automatically adjust for many OCR scanning or typographical errors without operator intervention.

For example, in a document mainly about “ducks”, if the word “cluck” suddenly appeared, ISYS may deduce the “d” had been incorrectly OCR'd as a “cl”. If you searched on the word “duck” ISYS would also hit on the word “cluck”. However, “cluck” would still be found if you specifically searched for it. In other words, ISYS is not so presumptuous as to correct seeming errors, only to compensate for them. In a document mainly about chickens, the reverse may be true.

In a document about both ducks and chickens, ISYS would deem it too close to call. ISYS uses advanced heuristic processing to achieve fuzzy precompensation. Rather than basing this on a static dictionary, the dictionary it uses is the index itself, hence ISYS is adaptive and will function correctly even on proper nouns.

## Document Viewing

The standard ISYS Search Engine provides no document “viewing window” user interface component. Rather, it returns the text from the document along with special markers for found words and basic formatting. The ISYS Search Engine provides the tools required to obtain text from and to navigate through the document. Your application is responsible for setting up windows and scroll bars to display the information to the user. It does not, however, have to include filters to read the various word processor document formats — all access to the text is via the ISYS engine.

In some applications it may not be desirable to display the found text to the user at all. For example, when indexing text in a database, the results may be more meaningfully displayed through an application-specific database query screen than through full-text browsing. The ISYS Search Engine leaves these issues entirely under your control.

## WYSIWYG Document Browsing

ISYS:sdk 8 includes an optional WYSIWYG document browser. This browser may be subject to additional licensing considerations, but provides fully formatted browsing of documents, including tables, embedded graphics, compound

documents, and other detailed formatting. The degree of formatting supported is that provided by ISYS:desktop 8 when “WYSIWYG” indexing rules are in effect.

Naturally, WYSIWYG document browsing carries a performance overhead compared to standard browsing. Standard browsing is strongly recommended for any performance-critical applications, or where huge, monolithic document files are involved.

Whereas standard document browsing is achieved by the application developer entirely developing their own UI and calling the ISYS `Get_Document_Line()` function to retrieve portions of text, the WYSIWYG browser creates a new Windows window and passes back its handle (HWND). This window comes complete with scrollbars and becomes an active player in your UI with little or no further action on your part.

## Indexing API

ISYS document indexing may be performed at three levels. At the highest level, a configuration file is created that contains a rule-base of how documents located on various volumes should be treated. The configuration file may either be created through a series of API calls or, more simply, preconfigured using an ASCII editor or otherwise generated by your application. It is not necessary to use the configuration API to create the configuration file, although you may do so if you choose.

The index is automatically brought up to date by an Update process, whereby new documents are indexed, altered documents re-indexed, and references to deleted documents are removed.

The ISYS Search Engine automatically scans the disk directories and determines which documents have been created, which have been changed and which have been deleted. Call-backs advise of indexing progress.

The second lower level indexing mechanism is known as the “low level indexing API”, and bestows complete control of the indexing and deindexing process with the application. The OEM application directs the ISYS engine to index and de-index specific files in the active voice. The host program provides the “file name” of the file to be indexed. The file name need not be an actual disk-based filename, but can be considered a 255-byte access key that uniquely identifies the document. The host program is returned a 32-bit handle by which the index knows the document. The application becomes completely responsible for deciding which files get indexed and when. The application also decides when files become de-indexed. A special form of de-indexing is also available which provides faster performance if the original text of the indexed document is still available, as is often the case with document management systems, for example.

The third method of indexing is “transactional indexing”, whereby the application constructs a transaction file containing various statements of fact, for example, “this document still exists”, “I know this document no longer exist”, and “here is a document, its identifier and its content”. The ISYS Engine reads the transaction file and updates its index according to the statements of fact. This enables applications to update an ISYS index without necessarily having a complete view of the document set in its entirety at any one time.

The simplest method, however, when the documents correspond to operating system files or objects enumerated by an External Access Module, is a single line of code to invoke the ISYS “Update” process, which automatically and completely reconciles the content of the index with reality.

### **External Access Module API**

The External Access Module feature enables OEM developers to implement a DLL through which ISYS can access text information that resides in “foreign” data sources which ISYS can not directly read.

This can be used, for example, to let ISYS index text residing in proprietary data files within your application or remote data. The External Access Module feature means ISYS can be used as a true plug-in engine, with API’s at both the top (retrieval) and the bottom (text access) level.

The DLL is provided by the OEM customer and is coded according to ISYS specifications. The External Access Module can be used to index text either through the first level indexing API (rule-based) or the low level indexing API (OEM-directed). Because application programs are completely unaware of whether the text they are accessing came via a built-in interface or an external interface, external text sources may be used either through OEM applications or through the standard retail ISYS user interface. An overview of the API that the DLL must provide is:

- **Open Document.** Given a 255-byte unique document identifier, open the document and make it ready to be read. The document identifier does not have to correspond to a disk-based file name, but may do so if you wish.
- **Read a character.** Returns a single byte from the file with special character codes defined for soft space, tab, indent, soft return and hard return. Also returns a 96-bit “address” that represents the location of this character byte within the file, in whatever terms are meaningful to the OEM DLL.

- **Seek.** Given an 96-bit “address” previously generated by the DLL, seek within the text file so that the next character read is the one that was originally read and return the 96-bit address quoted in the seek. The DLL must not store its own context-dependent information, except within the 96 bits, and can expect to be called to seek randomly to any character within the file.
- **Get Time Stamp.** Returns a 32-bit quantity that indicates the change generation of the document. This may either be a 32-bit time stamp or a simple incrementing generation number. ISYS uses this information to determine what documents need reindexing.

If rule-based indexing is being used, the External Access Module DLL should also define and export:

- **Scan\_First\_Directory/Scan\_Next\_Directory.** Called by the ISYS Search Engine when rule-based indexing is being performed to determine what documents exist. The DLL returns the “name” (255-byte identifier) of all the documents, one by one. When no more documents exist, the DLL returns a ‘false’ return code. The DLL passes back to ISYS the names and time stamps (or generation numbers) of the documents. ISYS uses this information to determine what documents should be indexed, reindexed or deindexed during rule-based indexing runs.

Developing an access module is usually a trivial matter and often completed in a single afternoon if the developer already has routines to read the file formats concerned, as is usually the case.

### **Sample Code and Support**

Sample applications with source code are provided in Visual Basic, Delphi, C, ASP, Cold Fusion and .Net. The applications completely demonstrate the calls of the Basic Retrieval group. The Visual Basic DECLARE statements may be used with any other similar version of Basic, for example Access Basic. ISYS Search Software provides high quality, developer-to-developer level technical support for all SDK customers.