

Finding your data faster by combining content and metadata searches

Jörn Bilow, Entimo AG, Berlin, Germany

Abstract

While the traditional approach of searching directly in the database works well for small repositories, it has performance limitations with growing repository size. Storing repository information in an index would make searching for keywords or phrases in content very fast. In order to provide this functionality, a separate entimICE server component was created – the Indexing Server, which will be demonstrated in the session. The Indexing Server is based on cutting-edge search technologies that are integrated in the solution (e.g. Lucene, TIKa, Akka) and is fully integrated with the entimICE access rights mechanisms. A speciality of the entimICE Indexing Server is that the information which is to be indexed represents various content types supported by the entimICE application – documents (e.g. PDF, Word), SAS® datasets, data warehouse tables, CSV files and even files which reside outside of the entimICE repository on remote servers. The index also includes all metadata from the entimICE metadata repository, and thus allows content and metadata searches to be combined. Search queries can be defined in a simple search user interface or written in the Lucene native query language or based on a domain-specific search language (DSL). entimICE allows custom search grammars to be produced, thus making the creation of search queries very convenient.

1 Introduction

With the introduction of the entimICE Indexing Server, new and powerful functionality is available to entimICE users. Search and retrieval of information from the entimICE repository is now possible for any kind of content, leading to massive performance gains. These results are achieved by making use of common search machine technologies as the foundation of the entimICE Indexing Server. The Indexing Server is based on rock solid and widely used open source technologies such as Lucene or Tika that are integrated in many enterprise search solutions.

2 Indexing Server

The purpose of the new Indexing Server is to provide access to any information in entimICE as fast as possible. This includes not only object metadata in the repository (e.g. names, properties, object metadata or status, comments and tags), but also attached content, such as text in PDF or Office files as well as tabular content in SAS and CSV files or in tables in the clinical repository.

In order to query for the indexed information in a convenient way, entimICE offers an enhanced Objects Search view whenever the Indexing Server is available and running. Besides the familiar search criteria of the classic search view, a query text field allows search terms to be entered in a convenient domain-specific language (DSL). See chapter "Object Search View" for more details on the user interface. It is now possible to query for content in files and in the database. Search queries can be expressed in the query language. See chapter "Query Language" for more details on the query language.

2.1 Base Technology

The Indexing Server is based on rock solid and widely used open source technologies that are integrated in many enterprise search solutions. By using these technologies inside entimICE, it is possible to respect entimICE access rights (e.g. search results will never include objects not visible to the requestor in the repository tree) and to include information not accessible from outside.

The following main libraries are used in the Indexing Server:

- **Apache Lucene:** The heart of indexing, which allows indexing information to be written, and fast access with a search language and a flexible API
- **Apache Tika:** The module supports the extraction of textual content from dozens of popular file formats
- **Akka:** A mature actor and concurrency library written in Scala which helps to ease the parallel and asynchronous implementation of server code
- **Xtext:** An Eclipse-based set of tools to implement domain-specific languages (DSL) with an integrated editor that offers syntax highlighting and smart phrase selection support. It is used to provide a domain-specific search language.

The Indexing Server uses the concept of actors. Actors work in parallel and therefore utilize the resources of the underlying hardware in a more efficient way.

2.2 Index Information

Extracted information is written to the index initially and is updated automatically whenever an entimICE object or its attached content is changed.

The index is partitioned into so-called domains. In the standard configuration two domains are provided:

- The main domain for all information from the entimICE repository
- The import domain for all information from the monitored remote import directories

By configuration, more domains can be added.

Below are some examples of the information included in the index:

- Object attributes (e.g. name, comment, creation date)
- Object lifecycle information (e.g. object status)
- Object version attributes (e.g. last modification date or last change by user)
- Content of clinical data elements and structural metadata (e.g. SDTM, ADaM)
- File content (e.g. text, PDF, MS Office and numerous other popular file types)
- Content of programs, scripts, log files, listing files
- Program execution traces
- Content of datasets and tables (e.g. ClinRep tables, SAS datasets, CSV datasets)
- Complex mappings

2.3 General Query Syntax

A search query is a combination of one or more search phrases. Each search phrase starts with a unique keyword followed by a specific phrase. The generic structure of a search phrase can be described as: <where> <what> with the keyword specifying "where" and the phrase specifying "what". Phrases can be combined with "AND" or "OR" operators and grouped with parenthesis.

Search queries can be written based on a domain-specific search language (DSL). XText offers a custom editor for grammars. Instead of using the Lucene language, users can formulate searches in a language that is more natural to their domain, such as CDISC or metadata searches. The query language offers a broad range of keywords that can be used

when constructing search queries. Qualified keys, wildcards, date and time values and conditions are supported. Searches can include node information regarding the directory tree in the repository, information about users, versions, object history, linking and various other metadata. CDISC metadata attributes can also be specified for a search. Detailed information about the DSL is available from Entimo on request. By having the option to use the native Lucene language, the DSL adds convenience without impairing the flexibility.

2.4 Object Search View

EntimICE offers users an Object Search view that has two operating modes: search with or without the Indexing Server. The modes are easily switchable with a check box.

When the Indexing Server is available, a special query field for the domain-specific search language is shown in the main part of the search dialog.

In many cases, the user interface will be sufficient to define search options. In the case of content searches or very specific searches, the DSL offers a convenient way to define search queries.

A classic search can now be seamlessly combined with a content search. With phrases such as "text submission FDA" all content can be searched for the words "submission" or "FDA". Many other phrases are possible, such as "cdisc #domain = ae" to search for CDISC-related metadata, or "status production" to search for objects with a given status.

Phrases can be combined using "AND" and "OR" operators, and complete DSL phrases will be combined with the search criteria selected or filled out in the search user interface. This allows for convenient, user interface-supported searches, or for more specific, DSL-oriented searches, or a combination of both. The DSL editor offers syntax highlighting and supports the selection of phrase parts (by using the control-space key shortcut). If there is a syntax error in the DSL phrase, the search returns an error.

2.5 Remote File Monitoring

To avoid useless overloading of the entimICE core system with huge amounts of legacy data or Big Data stores, the Indexing Server is able to point to external directories. The directories are accessible by entimICE (e.g. legacy studies required for pooling).

The Indexing Server can observe such directories using a special file monitor. All files in sub-directories are initially indexed and the directory tree monitors changes and also keeps track of the tree history. Any change in the directory tree immediately triggers a request to the file monitoring actor, which communicates with the content extraction actors.

3 Search

3.1 Custom Content Extractors

The Indexing Server has two kinds of content extractors – metadata and content extractors. As node metadata and content may contain special coded properties which are relevant for the index, but cannot be translated by a generic extractor (e.g. an ODM parser that extracts only relevant information from this special XML format), entimICE has an option to add custom extractors for specific content types.

3.2 Search Functions

A highlight of the entimICE Indexing Server is its ability to define search functions in the configuration. When creating search queries, users can use this function, which will be replaced by the full search query at processing time. This approach makes specific searches easier to write and remember.

3.3 Sample Queries

The following search examples illustrate the capabilities of the entimICE Indexing Server.

Find objects:

- with a certain status ("approved")
- in a certain repository area (e.g. "project_X")
- within this area in a certain folder type (study's prod area)
- in a specific folder ("program_tlg")
- of a certain object type (e.g. "SAS Program")

Find objects that have been updated by a specific "userID" in the last three days (Note: "updated" means status change).

Find objects having a specific object metadata value:

- either directly assigned or inherited from parents
- currently, or in history

Find structure metadata objects belonging to "SDTM 3.2" (which might be defined in structure metadata or in a study folder above)

Find programs:

- that read the CSV file (e.g. "root/A/B/C/D.csv")
- in a certain file version (e.g. "14.2")
- within a certain time span

4 Conclusion

The amount and complexity of clinical data will most likely increase in the future. Finding the right information in a reproducible, easy and effective manner will be an ever more challenging task. By indexing the content as well as the metadata information of the clinical repository, the entimICE Indexing Server adds value for the clinical scientist. The domain-specific language enables users to build searches using familiar terms instead of a more complex proprietary syntax.

The central benefit of the entimICE Indexing Server is its awareness of access rights for all search results. By definition, users can only find those search results for which they have been granted access rights.

Contact Information

Your comments and questions are valued and encouraged. Contact the author at:

Jörn Bilow
Entimo AG | Stralauer Platz 33-34 | 10243 Berlin | Germany
Email: bil [at] entimo.de
Work Phone: +49 30 520 024 100
Web: www.entimo.com

Brand and product names are trademarks of their respective companies.