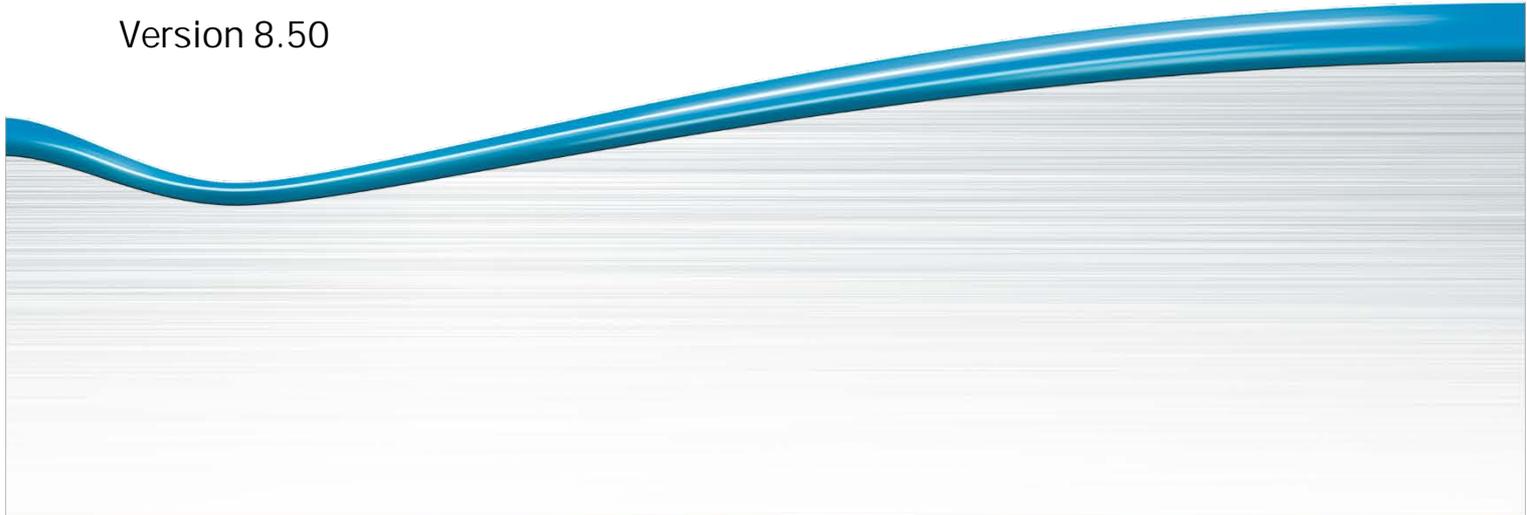


enaio[®]

Software Documentation
enaio[®] fulltext

Version 8.50



All software products as well as all related extension programs and additional functions are registered and/or in-use trademarks of OPTIMAL SYSTEMS GmbH, Berlin or its subsidiaries. They may only be used according to a valid licensing agreement. The software as well as related documentation are protected by German and international copyright law. Unauthorized duplication and sales is plagiarism and subject to criminal prosecution. All rights reserved, including reproduction, transmission, translation, and storage with/on all kinds of media. For all preconfigured test scenarios or demo presentations: All company and person names which occur in examples (screenshots) are fictional. Any resemblance to existing companies or persons is purely coincidental and unintentional.

Copyright 1992 – 2017 by OPTIMAL SYSTEMS GmbH
 Cicerostraße 26
 D-10709 Berlin

11.04.2017
Version 8.50

Contents

- Contents 3
- Introduction..... 4
 - Full Text Integration in enaio® 4
- enaio® fulltext 5
 - Installation 5
 - Installing the Components 5
 - Installation of Elasticsearch 5
 - Installation of Services 5
 - Configuration of Services 5
 - Service 'Index' 6
 - Service 'Search' 7
- Configuration of enaio® Components 9
 - enaio® Components – Introduction..... 9
 - Configuration in enaio® enterprise-manager 9
 - Object definition in enaio® editor 10
 - Applications as Facets of Hit Lists 11
 - Subsequent Indexing – the Automatic Action 'Full-Text Indexing' 12
 - The Automatic Action 'Full-Text Export on Object Level' 13
 - Query file 14
 - Logical Expressions 14
 - User Navigation in enaio® client 14
 - Search Expressions, Placeholders, and Combinations 16
 - Extended Search Modes 18

Introduction

Full Text Integration in enaio®

enaio® offers the possibility to index documents using indexing services and to search for them in enaio® client using the integrated full-text index.

Both content and index data of documents, registers and folders can be indexed.

Full text integration is required for operating the enaio® platform and an efficient interaction of the individual enaio® components.

For the full text integration, Elasticsearch with Lucene will be integrated.

enaio® fulltext

Installation

Installing the Components

enaio® fulltext consists of Elasticsearch and the micro services 'index' and 'search.' The services will be integrated into the enaio® service-manager.

Information on complex installation with multiple full-text services can be obtained from Support or Consulting of OPTIMAL SYSTEMS GmbH.

Installation of Elasticsearch

Elasticsearch is installed via `elasticsearch_setup.exe` from the directory `\Microservices\Elasticsearch` of the installation data.

The installation directory, the HTTP port for Elasticsearch, and the index directory for the index database is specified.

The index directory must not be located below the installation directory and must provide sufficient place for the comprehensive index database. For questions about dimensioning, please contact Support or Consulting at OPTIMAL SYSTEMS GmbH.

Elasticsearch is automatically started as service.

Installation of Services

The 'index' and 'search' services will be installed via `os_service-manager_setup.exe` from the directory `\Microservices\OS_ServiceManager` of the installation data.

You select the 'index' and 'search' services as services from enaio® service-manager. enaio® service-manager with the services 'admin,' 'discovery,' 'messaging' is a necessary part of enaio®. The services are automatically started.

Configuration of Services

The 'index' and 'search' services are configured via enaio® service-admin, which you access in the browser via `http://<ip>:<port>`. The standard port of enaio® services-admin is 7273.

For the services, you open the corresponding configuration page from the drop-down list.

In which configuration file the parameter is stored is specified for each of the parameters. 'index' use the same configuration files as 'search.' In standard installations, the configuration can only be made via 'search.'

The entries for the technical user and the connection data to enaio® are specified by the enaio® services-manager during installation.

If the enaio® database is set to 'collation sensitive,' then for the clause evaluation in the configuration file `search-prod.yml` from the directory `..\services\service-manager\config\`, the parameter `enaio.dms.collation.casesensitive` must be set to `true`.

Service 'Index'

Parameter:

Technical user	User name of the technical user for all services The technical user for the services usually requires access rights for all folder, register, and document types, as well as the system roles 'Server: Switch job context' and 'Server: Run Ado jobs.'
Password	Password of the technical user
enaio® server	IP and port of enaio® server with connection probability. Multiple enaio® servers are specified separately by a '#'.
Elasticsearch - server	IP of Elasticsearch Standard: localhost
Elasticsearch - port	Port of Elasticsearch Default: 9300
Elasticsearch - cluster	Cluster name of Elasticsearch Standard: elasticsearch
Schema trustworthy IPs (Trusted IP Pattern)	Schema trustworthy IPs . * allows every access (default) enaio® gateway, enaio® server, and enaio® service manager always require access, further components possibly, which work with enaio® fulltext. For a list of addresses, IP addresses must always be placed in brackets, the separator between addressees is 'Pipe,' points in IP addresses must be masked with '/', but not double points in Ipv6

addresses.

If you save changes, the service will be restarted.

Service 'Search'

Parameter:

Technical user	User name of the technical user for all services The technical user for the services usually requires access rights for all folder, register, and document types, as well as the system roles 'Server: Switch job context' and 'Server: Run Ado jobs.'
Password	Password of the technical user
enaio® server	IP and port of enaio® server with connection probability. Multiple enaio® servers are specified separately by a '#'.
Elasticsearch - server	IP of Elasticsearch Standard: localhost
Elasticsearch - port	Port of Elasticsearch Default: 9300
Elasticsearch - cluster	Cluster name of Elasticsearch Standard: elasticsearch
Schema trustworthy IPs (Trusted IP Pattern)	Schema trustworthy IPs . * allows every access (default) enaio® gateway, enaio® server, and enaio® service manager always require access, further components possibly, which work with enaio® fulltext. For a list of addresses, IP addresses must always be placed in brackets, the separator between addressees is 'Pipe,' points in IP addresses must be masked with '/', but not double points in Ipv6 addresses.
Maximum hit number for activating spell check (Suggest minhits)	The spell check is activated if hit number is less than or equals the value indicated here. The spell check is a feature which incorporates other terms with regard to typing errors. Default: 5

Maximum score value activating spell check (Suggest minscore)	Spell check is executed if the maximum score value of hits is between 0 and the value indicated here. If the maximum score value of hits is higher than the value entered here, the spell check is not activated assuming that there are not typing errors. Default: 0.6
Context size for preview generation (Preview context words)	Indicate the number of words before and after the search term found in the document. Default: 100
Number of text sections for preview generation (Preview summary fragments)	Specify the number of text sections for preview generation in hit lists. Default: 1

If you save changes, the service will be restarted.

Configuration of enaio® Components

enaio® Components – Introduction

Full text integration for object types is configured with enaio® enterprise-manager and enaio® editor.

The full text integration of image document types additionally requires the configuration of the OCR component Abbyy FineReader. The configuration is documented in the Administration manual.

The automatic action 'Full-text export on object level' enables you to index objects of object types with the property 'Full-text indexing – not automated.'

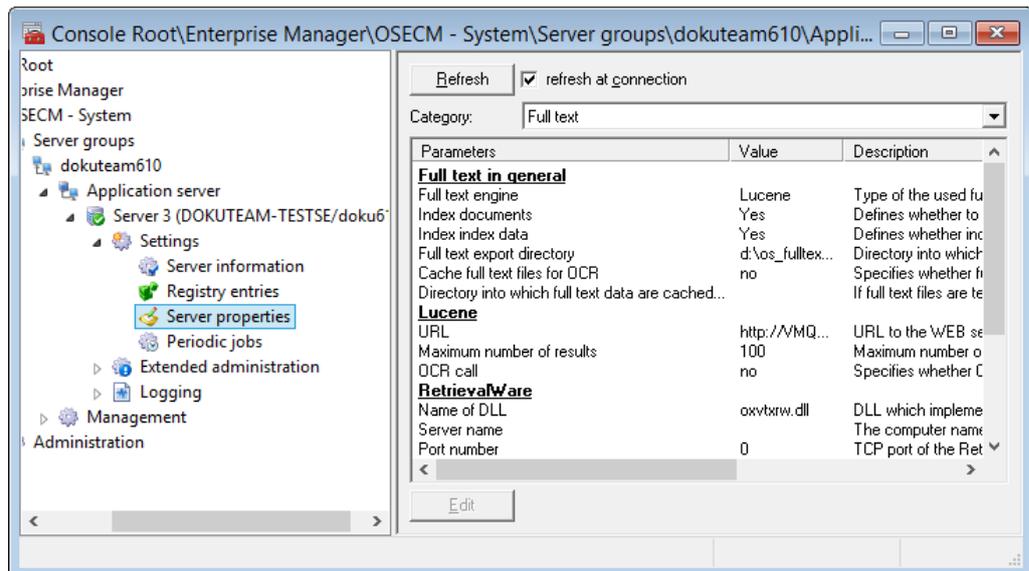
An additional search field is added to index data forms of fully indexed DMS objects in enaio® client that allows users to start full text searches.

Documents encrypted by enaio® client cannot be indexed.

Configuration in enaio® enterprise-manager

When administering enaio® server with enaio® enterprise-manager, via the server property 'Category: Full text' you specify Lucene as the type of the full-text database.

Furthermore decide whether Documents, the Index data, or both will be indexed. Specify the full-text export directory, specify whether full text files for OCR are temporarily saved in order to prevent conflicts caused by multiple file access. To do so, also specify a directory.



Afterwards, specify the following settings.

§ **URL**

Web service URL of the full-text engine.

§ **Maximum number of results**

The maximum number of results that can be returned by a full text query. The default value (1000) is high and can be reduced.

§ **OCR call**

Specifies whether OCR processing is carried out before image documents are transferred to the full text engine.

§ **Full-text auto-complete server**

Address or name of the server which runs the auto-complete feature for the full text service. In most cases the address or name is identical to the address of the full-text server.

§ **Full-text auto-complete port**

Port of the service.

Object definition in enaio® editor

Indicate folder, register, and document types to be fully indexed in the object definition in enaio® editor.

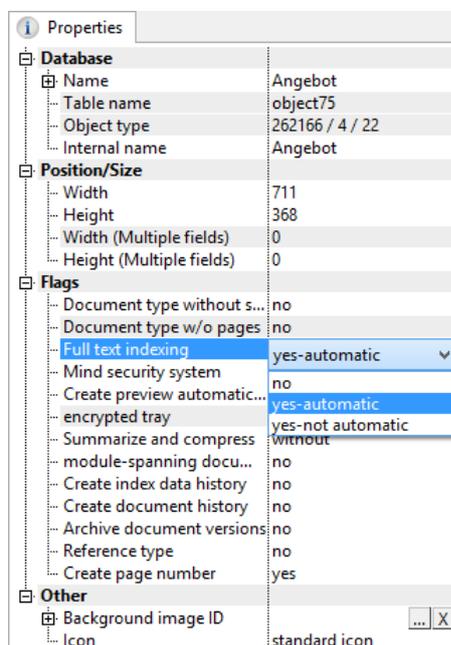
The properties dialog of an object type allows you to determine whether or not to full text index objects of the type. All dialog elements of the objects receive the property 'Full text export'. You can turn on and off this property for each element individually.

All objects of an object type being set to 'Full text indexing - automatic' will automatically be fully indexed.

When setting an object type to 'Full text indexing-not automated' the automatic action 'Full text export on object level' will index with full text all objects of this type.

Reference documents will also be indexed with full text.

Index data changes of cross-type reference documents will be updated automatically, but content modifications will not.



Document types will not be indexed if the property 'Encrypted filing' has been activated. Documents can be indexed if they are encrypted by enaio® server.

When activating the property **Full text indexing** for a document type, database adjustment is not required.

Applications as Facets of Hit Lists

Full text hit lists also contain an area with facets – dynamic filters that can be used to restrict the hit lists based on properties.

In the facets area, the following hit properties are given:

- § Object type
- § Application
- § Edited
- § Editor
- § Created
- § Creator

For each property, the number of hits with that property is given.

Hits are assigned to applications using groups of MIME types. These data are stored in database tables and can be customized for specific projects.

Via their user-specific settings, users can enable/disable basic facets and enable list catalogs for every object type as facets. These facets are shown in a full-text hit list of the object type.

Subsequent Indexing – the Automatic Action 'Full-Text Indexing'

Documents already created in enaio® can be fully indexed afterwards. For this purpose, the automatic action 'Full text indexing' provided by the library `axacidx.dll` is used in enaio® administrator.

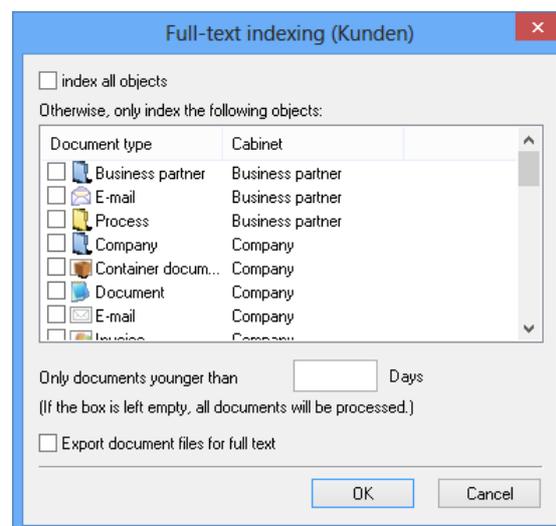
When configuring the automatic action, select the document types to be full text indexed afterwards and define whether to not only export index data but also the document files.

You can optionally limit the number of documents to those that are younger than the indicated time (in days).

Object types with the property 'Full text indexing – yes' are displayed. The property 'Full-text index' of dialog elements is taken into account. Reference documents will also be fully indexed.

The action indexes a maximum of 50,000 documents.

The user of the action requires access rights to all full text object types and to all folders and registers in which full text documents are located, also the system role 'Server: Run Ado jobs.'



Details about the automatic actions can be found in the Administration handbook.

The Automatic Action 'Full-Text Export on Object Level'

The automatic action 'Full text export on object level' (`axacvexp.dll`) allows you to index data of object types with the property 'Full text indexing – not automated' and the property 'Full text indexing – automated'.

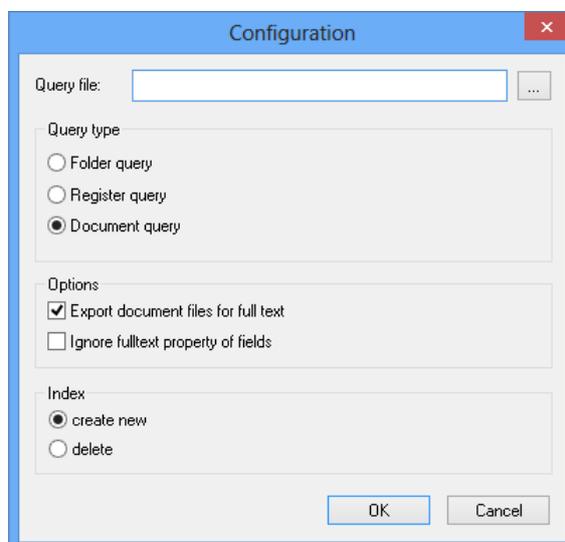
Data of object types with the property 'Full-text indexing – not automated' can only be indexed using this automatic action.

The action indexes a maximum of 50,000 documents.

The user of the action requires access rights to all full text object types and to all folders and registers in which full text documents are located, also the system role 'Server: Run Ado jobs.'

This action allows you to delete data of already fully indexed objects from the full text index.

Create a query file for this action with which you indicate the objects to be fully indexed. The query file may contain logical expressions enabling you to limit the object selection to those objects fulfilling these conditions.



Indicate the **Query file** in the configuration dialog and determine the **Query type**, i.e. folder query, register query, or document query.

The **Options** area allows you to decide whether to only index index data or document files as well.

In addition, you can define the property 'Full text index' to be ignored in order to index all index data.

The **Index** area offers to add query object data to the index (**Create new**) or to delete them from the index.

The **Ignore full text property of fields** option is not taken into consideration, if the Content-Processing-Bus is used. The **Delete index** option is also ignored, i.e. the action with this option is inoperable.

Query file

You can create the query file with any arbitrary text editor. It has the following structure:

[ANFRAGE]	The file begins with the section name 'Query'.
SCHRANK=folder type name	Enter the name of the folder type that the objects originate from into the first line.
REGISTER=register name	Additionally enter the register type name the objects originate from for the query type 'Register query'.
DOKUMENT=document type name	Additionally enter the document type name the objects originate from for the query type 'Document query'.
KLAUSEL1=Objekt@Feld=Wert	Optional logical expressions allow you to limit the selection to those objects that fulfill these conditions.
...	
KLAUSELn=Objekt@Feld=Wert	Logical expressions must be numbered consecutively.

You must use internal names and enclose them in percent signs.

Logical Expressions

Optional logic expressions allow you to limit the selection to those objects that are indexed with the indicated value in the indicated field.

Example:

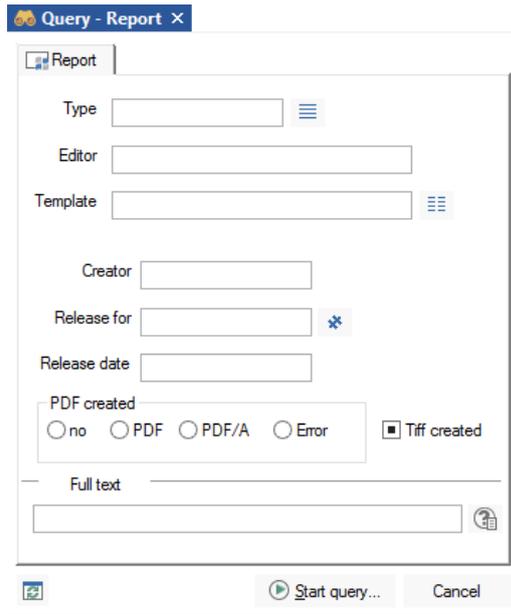
```
Klausel1=Kunde@Status=abgeschlossen
```

Only those documents of the specified document type are indexed by the document query which, in the index data of the object type 'customer,' such as a folder, are indexed for example with the value 'completed' in the 'Status' field.

User Navigation in enaio® client

The index data form of archive objects for which full text indexing has been configured contains an additional area for full text searches.

This search area is only displayed after the module 'VTX' was licensed for the workstation.



In the **Full text** area, you can enter the terms to be searched for in the full text index. You can combine terms logically and use placeholders.

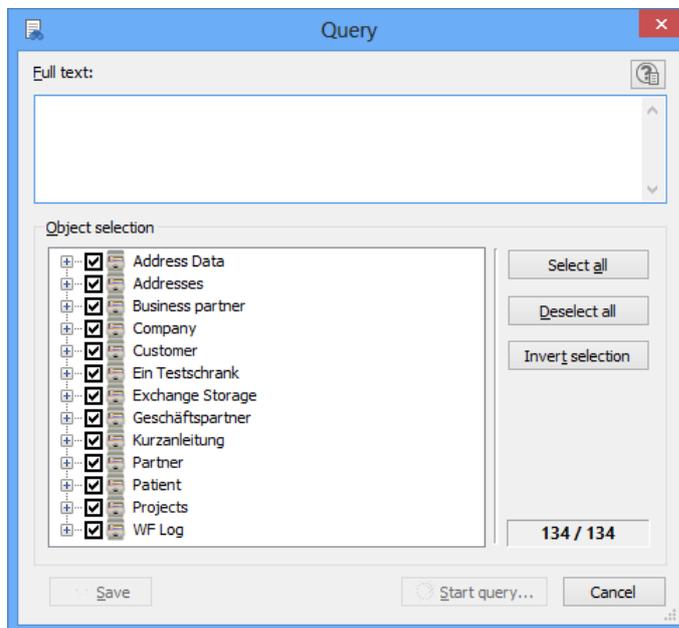
For details click on the help button in the **Full-text** area (see 'Search Expressions, Placeholders, and Combinations').



Searches using index data and searches using full text are combined with the logical AND.

Depending on the configuration in enaio® enterprise-manager, the full text index contains either all terms out of the content of documents, or all words from the index data of folders, registers, and documents, or both.

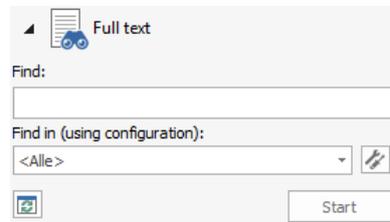
Double-click the ribbon tab **START** to open a full text search form that searches the full text index only.



Enter terms, select object types in the 'Object selection' area and confirm by clicking **Start query**.

The hit list offers all DMS objects of the selected type with entered terms being contained in their full text index. The hit list does not show whether entered terms are contained in the full text index of the index data or in the full text index of the document.

When executing a full text query from the navigation, the user can configure object selections.



If a folder or register is selected, a full text search can be run from the ribbon through the content of the folder or the register.

Search Expressions, Placeholders, and Combinations

The following search options are available using the full text search:

Search expression	<p>Search expressions may contain numerical and alphabetical characters, including the German umlauts.</p> <p>Search terms are not case-sensitive. Search expressions are normalized, e.g. umlauts are resolved. The system always searches for the basic form and for parts of terms.</p> <p>Example: A search for 'dreaming' also shows 'dream,' 'dreamed,' and 'dreamcatcher' as hits.</p> <p>Searching for the basic form does not however find the derived forms: 'Dream' does not find 'dreaming,' but it does find 'dreamcatcher.'</p>
Search expression and placeholders	<p>Search expressions can be combined with the following placeholders:</p> <p>'*' stands for an arbitrary number of arbitrary characters.</p> <p>'?' stands for exactly one arbitrary character.</p> <p>Placeholders can be used at the start, at the end, and within a search term, and it is also possible to combine them.</p> <p>Searching with placeholders at the start takes a lot longer.</p> <p>When issuing full text queries, the auto asterisk settings which have been predefined for enaio® client will be ignored. Thus, the placeholder '*' will not be added automatically.</p>
Operators	<p>Search for documents which contain one of multiple search expressions:</p>

<p>Multiple search expressions are linked with the logical OR, if they are separated by the Boolean operator 'OR.' There must be at least one space before and after the operator.</p> <p>If expressions are not separated by an operator, these expressions are implicitly combined using the logical OR.</p> <p>Example: order editor Maier</p> <p>Documents in which at least one of the three words is found will be shown.</p> <p>Search for documents that contain all search expressions:</p> <p>Multiple search expressions can be connected with the logical AND if they are separated either by the Boolean operator 'AND' or the characters '&&.' There must be at least one space before and after the operator.</p> <p>Example: order AND editor && Maier</p> <p>Documents in which all three words are found will be displayed.</p> <p>Search for documents that do not contain the search expression:</p> <p>The Boolean operator 'NOT' can be placed before a search expression. In this case, documents are displayed that do not contain the expression.</p> <p>Example: editor AND NOT Maier</p> <p>Documents containing 'creator' and not 'smith' will be displayed.</p> <p>Search for documents that contain the exact same sequence of search expressions:</p> <p>It will be searched for the exact sequence of terms between quotation marks.</p> <p>Example: "Klaus Maier"</p> <p>Documents where 'peter' is exactly in front of 'smith' are displayed.</p> <p>Without quotation marks, documents would be displayed that contain 'peter' or 'smith'.</p> <p>Number of words between search expressions</p> <p>The sequence of two search terms in quotation marks can be extended by defining the maximum number of words within the sequence.</p> <p>Example: "Klaus Maier"~1</p> <p>Documents containing 'Klaus Peter Maier' will also be shown, but not documents containing 'Klaus Peter Maria Maier.'</p> <p>Search expressions within a clause</p> <p>Search expressions within a clause are searched for as follows:</p>
--

	<p>NEAR/S(Klaus Project Manager)</p> <p>Within the brackets multiple search expressions can be specified and combined. Here, AND is used to link, if nothing is specified.</p> <p>Combinations</p> <p>Use different operators to combine multiple search expressions.</p> <p>Example: Schulze AND Maier OR Müller</p> <p>However, you must use parentheses to specify the order of operation.</p> <p>Example: Schulze AND (Maier OR Müller)</p>
--	--

You can find further details about special search functions here:

http://lucene.apache.org/core/4_8_0/queryparser/org/apache/lucene/queryparser/classic/package-summary.html#package_description

Searches using index data and **FULL TEXT** searches are combined with the logical **AND**. The hit list thus contains the documents which fulfill both the search criteria over the full text index and over the index data.

For the full text search for a date, which is part of the index data, you must enter the date in the following format: DD.MM.YYYY

Extended Search Modes

The standard search extends the search across the basic form of search expressions and to parts of terms.

These extensions can be switched off by specifying a search mode:

MODE/B	<p>No extension to parts of terms.</p> <p>Example: MODE/B&dreaming</p> <p>'Dream' is found, but not 'dreamcatcher.'</p>
MODE/D	<p>Exact search, no extension to parts of terms and no extension beyond the basic form.</p> <p>Example: MODE/D&dreaming</p> <p>Neither 'dream' nor 'dreamcatcher' are found but 'dreaming' is found.</p>

The mode is followed by '&' as separator before the search expression. Multiple search expressions following the mode without '&' in brackets.

Example: MODE/D(dream AND sleep)

For combinations of search expressions, different modes can be specified for the search expressions.

Example: MODE/D&dream AND MODE/B&sleep

Note that the modes are written in block capitals.