Configuration and administration

ELO iSearch

Table of contents

ELO iSearch guide

What does ELO iSearch offer?

The ELO ECM Suite provides an efficient solution for retrieving information and locating relevant data, even with large quantities. The most important matches are found quickly and conveniently.

Various search functions ensure that you can access the documents you need as well as information from any integrated third-party applications, such as ERP, CRM, or e-mail systems. ELO iSearch helps you find information faster by automatically completing search terms as you type, correcting typing errors, and searching for similar or related terms.

In addition, when you search for `contract`, ELO iSearch will also suggest synonyms such as `agreement`. Even if the search does not return a unique search term, ELO iSearch gets you the results you want, suggesting terms that are related to the original query. Intelligent dictionaries break compound words like `insurance policy` down into their basic forms. The words can also be converted back to their root form so you no longer have to worry about plurals, inflected forms, or other compound words when entering search terms – the search engine does this in the background.

If the user enters `policiy` instead of "policy", ELO iSearch displays the message: `Do you mean policy?` These intelligent assistance systems pick up on typing errors, improving the search results or in some cases ensuring that there are any hits at all.

Search queries using structured metadata search tools are often very precise and very specific. Searching for an invoice, contract, or quote number in the whole repository produces fast and reliable results.

Another challenge is to save frequently used, recurring, or complex search queries with complicated filter options and being able to apply them quickly and conveniently as needed. The tiles introduced with ELO ECM Suite 10 for saving and managing search favorites provide the perfect solution to this problem.

ELO iSearch in practice

Basic principles and the dual strategy

ELO iSearch

ELO iSearch enables full text searches, that is, of the text within a document as well as additional information such as metadata. It provides various options to extend the search to different areas. ELO iSearch analyzes and processes full text using search rules and patterns (algorithms).

A special search index is automatically created and updated in parallel to the SQL database. This ensures fast response times, even for complicated search queries. However, this also results in a minor delay before new documents can be found using ELO iSearch. Normally, this delay should be less than one minute.

The following sections apply to the use of ELO iSearch in the ELO Java Client.

Metadata search

The metadata search is a second search option for special requirements. It searches for the values exactly as they are entered right in the SQL database, meaning no delays.

Compared with ELO iSearch, it may deliver different results for the same search input. Typically, ELO iSearch will return more hits than the metadata search since language-specific search rules and patterns also finds plural forms of a search term, for example.

Generally, ELO iSearch takes the more modern, flexible, performant approach and is the preferred search variant in ELO.

Full text search

Full text refers to all the text content within a document. On an invoice, this would be not only the name and address of the recipient, but also the name and address of the issuer, the invoice items, amounts, and any payment instructions.

Linguistics

Linguistics focuses on the meaning and rules of human language. Based on such rules, ELO iSearch applies automated methods that prepare texts in such a way to allow them to be found as easily and conveniently as possible. As a result, documents with relevant contents can be found using simple search terms. Therefore, it is not necessary to know exactly how the word is written within the document. This is especially practical when a word has several different spelling variants. In the following, this will be referred to as language and search rules in the context of search patterns.

Breakdown of compound words

Compounds are a union of two or more words. Compounds, such as `insurance policy`, function as a single unit of meaning. They can be broken down into their components, in this case `insurance` and `policy`.

Root forms

When processing the query, the singular is derived from the plural form, or the infinitive of declined verbs is found.

Example 1

For example, when searching for `invoices`, `invoice` is also queried. However, it is not possible to form a plural from the singular of a word. The table below illustrates this and will help you to select and enter your search term.

> **Please note**
>
> Compound words are broken down and root forms are created during indexing. To use this feature, you have to enter the root form as the search term. If you use an inflected form as the search term, this exact form will be used for the search. The following table contains some examples.

Full text	Search term	Results
Invoices	Invoices	Yes
Invoices	Invoice	Yes
Invoice	Invoices	No
Incoming invoices	Incoming invoices	Yes
Incoming invoices	Invoices	No
Incoming invoices	Invoice	Yes
Invoice	Invoice	Yes

Languages

You need to keep in mind that all search rules depend on the language used. In English, a plural is usually formed by adding an 's' (for example, `book` and `books`), but in German there are several variants and also many irregular forms (one `Buch` but several `Bücher`).

ELO iSearch also applies different rules depending on the language. As a result, not all search rules and patterns are available to the same extent in each language. Most language rule and pattern optimizations apply to the German language. English is also supported.

To automatically process documents, the language of the full text is initially determined based on the words they contain. These words are used to process the text with search rules and patterns. This can become problematic in documents that contain multiple languages.

The metadata fields are always processed with the set/selected repository language.

Multi-word searches

If you enter multiple words into the search field, the search looks for documents containing all of these words. This means each additional word you enter narrows down the search, ideally helping you to find just the right document.

To search for multiple alternative terms, separate them using a comma, for example `business trip, customer appointment`. In this case, only one of the search terms has to appear in the document.

> **Please note**
>
> For the multi-word search to work, all searched terms must be contained in one area (see Search area/fields searched). Searching for `seating plan` in all search areas finds documents containing the words `seating` and `plan` either both in the full text, both in the metadata, or both in the feed. You would not find any documents containing the words in separate areas, i.e. one word in the full text and the other in the metadata.

Using wildcards in the search term

Index-based searches are based on a breakdown into tokens at the point of indexing and on a token comparison at the time of the search. As a result, this search is less suitable for using wildcards (e.g. *), as a character-based expansion has to be performed at the time of the search.

A search term with * cannot be broken down into its components via linguistic means and the search is not generally precise. Token expansion at the time of the search also leads to a significant reduction in search performance.

Generally, we recommend only using wildcards in the ELO iSearch as an exception if you need to find any results at all quickly. It's better to search by combining components (AND-linked, separated with spaces, or OR-linked, separated with commas). Also refer to the examples in the section Root forms.

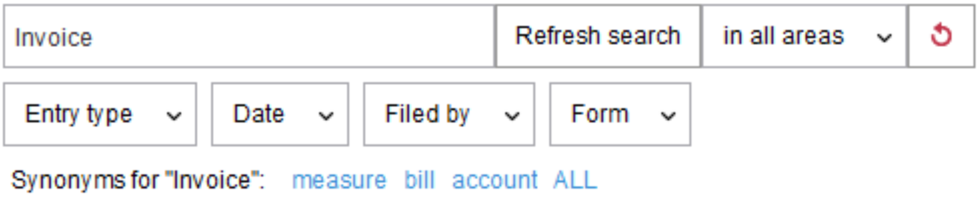
Phrase search

It is possible to search the document for not only individual words, but also for entire sentences by placing related words in quotation marks. The full text must contain these words in the exact order.

Searching for `John Smith` finds this exact name. Without the quotation marks, documents containing the words that are not right next to one another, for example, `John Cramer accompanied Helen Smith` would also be found.

Thesaurus

ELO iSearch has a built-in thesaurus that lists synonyms of the entered search term. They appear as soon as a word has been entered followed by a space.



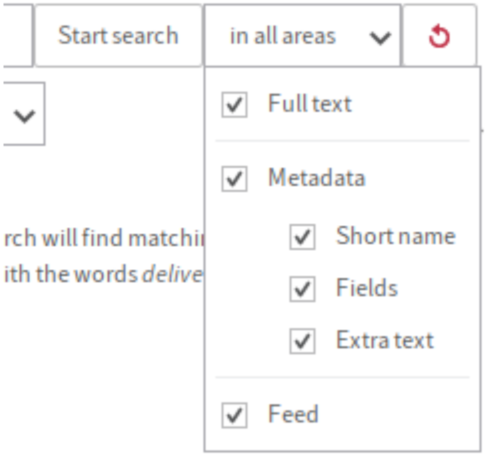
Invoice

Synonyms for "Invoice": measure bill account ALL

Select a synonym to apply it to the search field as an alternative. Companies can customize their thesaurus. It is also possible to remove suggestions. Refer to the Integrating custom thesauruses section.

Search area/fields searched

The options for restricting or extending the search to specific areas are located next to the search term field and the *Start search* button. This allows you to choose the area or areas you want to search for the words in. You can choose between the areas *Full text*, *Metadata*, and *Feed*. The *Metadata* area is split up into *Short name*, *Fields*, and *Extra text*. The default setting searches in all areas.



☑ Full text

☑ Metadata

- ☑ Short name
- ☑ Fields
- ☑ Extra text

☑ Feed

> **Information**
>
> As described above for the full text search, all fields are searched using the same search rules and patterns. This means root forms are identified and compound words are broken down first.

Full text

With digital documents, the full text is always the text exactly as it is in the document. Full text searches can therefore find all the contents of a document. Unlike the metadata search, the full text search is ideal for finding any term within the document, and not just defined keywords or meta information.

Scanned documents have to be converted into text form using OCR first. In some cases, letters or entire words may not be recognized correctly, which can prevent the document from being found.

Text blocks and columns may cause sentences to be mixed up, meaning a phrase search will not find the sentence.

Short name

Documents and folders are shown in ELO with their short name. For this reason, the short name should be the most accurate description of the content as possible.

Fields

When the *Fields* setting is enabled, all configured fields in all metadata forms are included in the search. For example, searching for `Jones` will return different results:

- `Jones Materials Ltd.` in the *Supplier* field
- `Susan Jones` in the *Customer* field
- `42 Jones Street` in the *Address* field

Extra text

The extra text tab in a metadata form allows you to add further information on a document or folder in ELO. Unlike a normal field, it provides space for longer texts and can be searched.

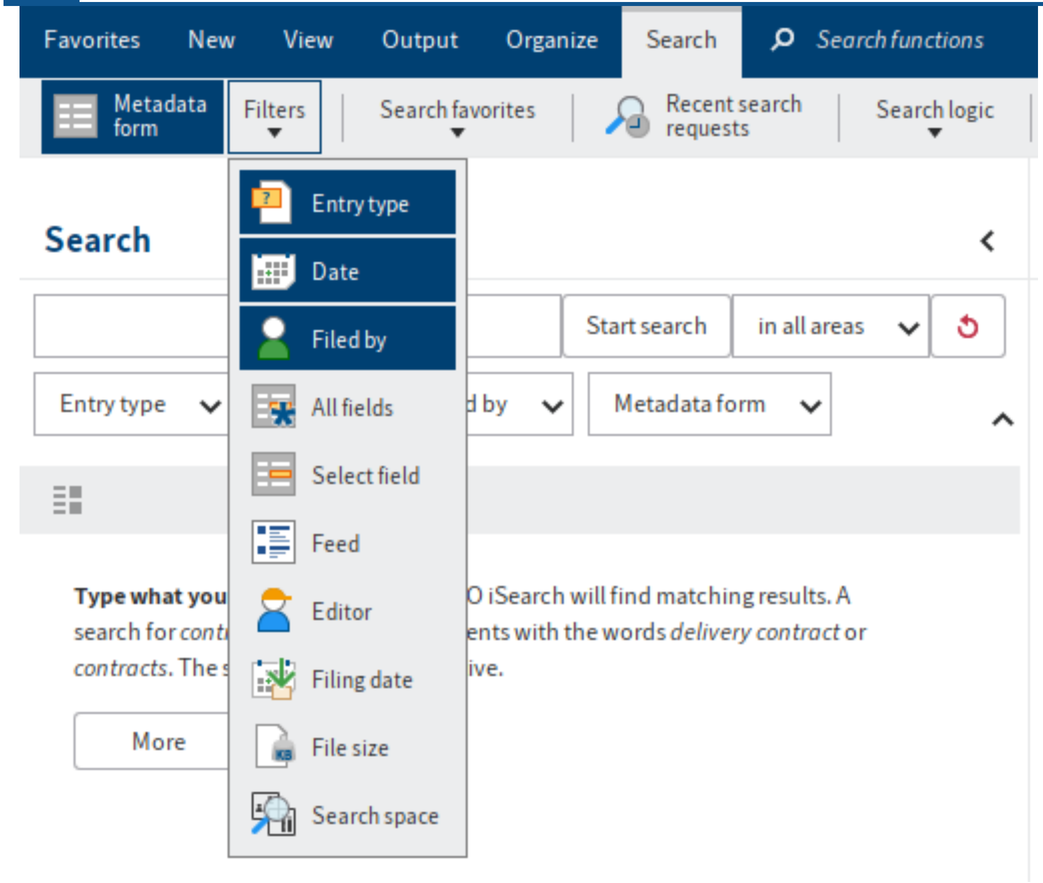
Feed

When *Feed* is selected, all posts and comments in the feed are searched in text form.

> **Information**
>
> The feed also contains the version numbers and version comments for both the current working version of a document and older versions.

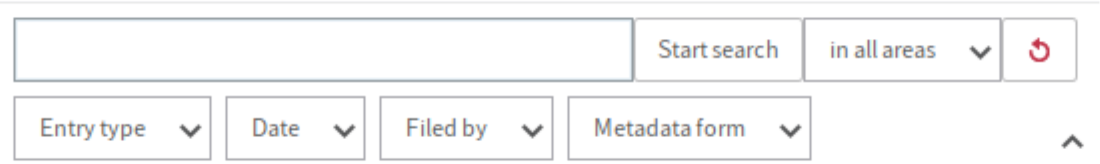
Filters

In many cases, it makes sense to incorporate structured information into the text-only search in order to narrow down the search range. ELO iSearch provides filters for this. Users can enable these filters on the ribbon, which then show up as additional buttons below the input field.



Type what you ...O iSearch will find matching results. A search for *conti*...ents with the words *delivery contract* or *contracts*. The s...ive.

A range of filters are shown below the search field by default. These are the most frequently used filters. Refer to the section on search favorites to find out how to set them universally, specific to groups, and individually.

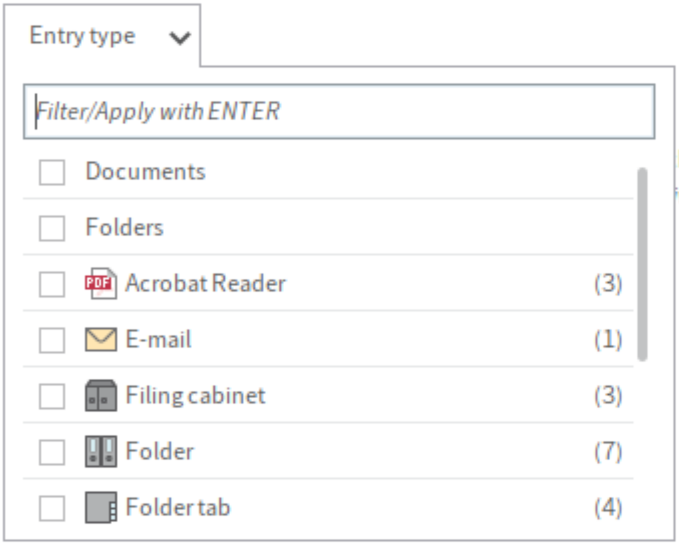


Selecting the button opens the respective filter and allows you to set special search specifications. It is almost always possible to make an entry via mouse or keyboard.

Just like the search, the filters take into account permissions set in ELO. Only the filter values for the entries the user is permitted to see are shown.

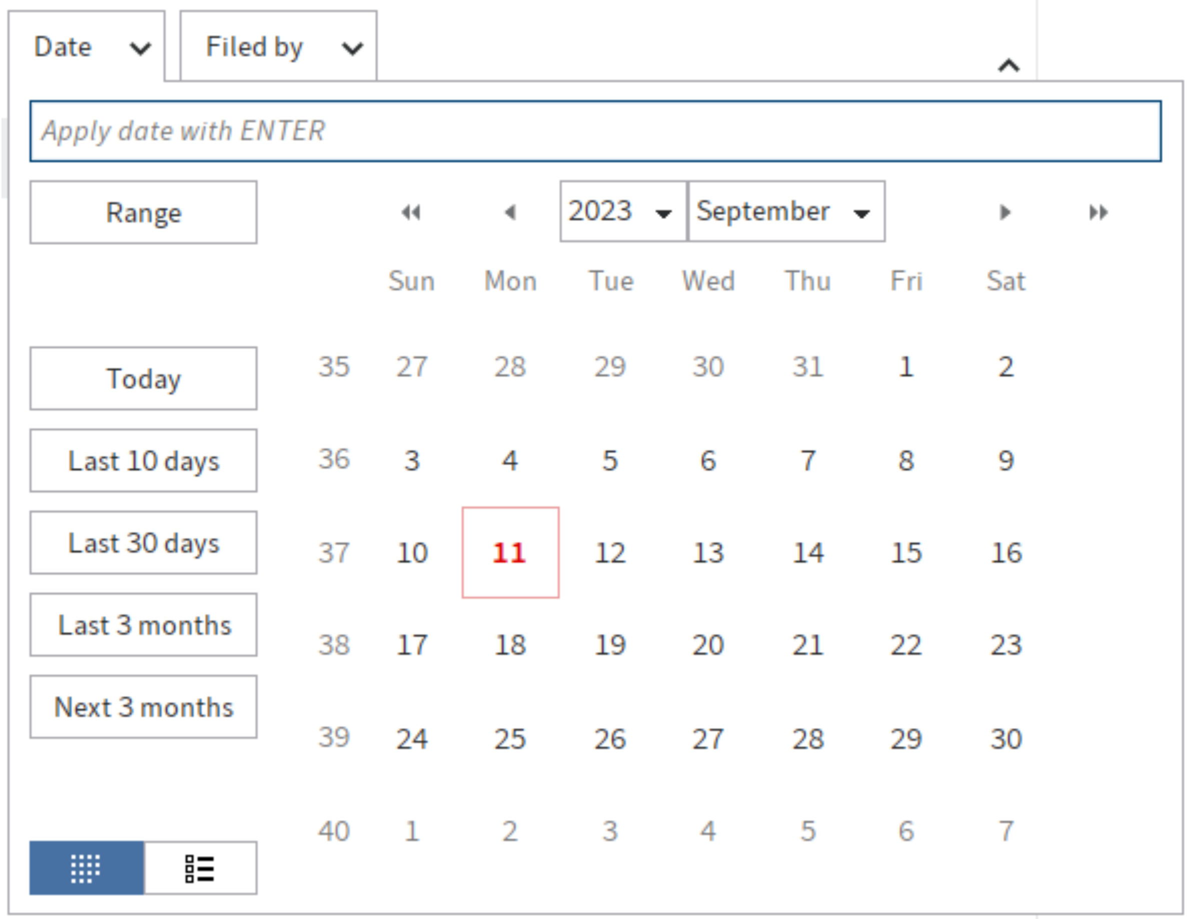
Entry type

As with the metadata form filter, the entry type filter shows a list of possible values. It is used in the same way, with search-as-you-type, autocomplete, or selection from the list. Unique to the entry type filter, *Documents* and *Folders* are always shown at the top, and combine all the corresponding entry types.

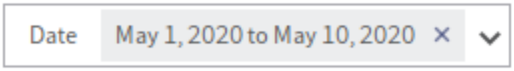


Date/filing date

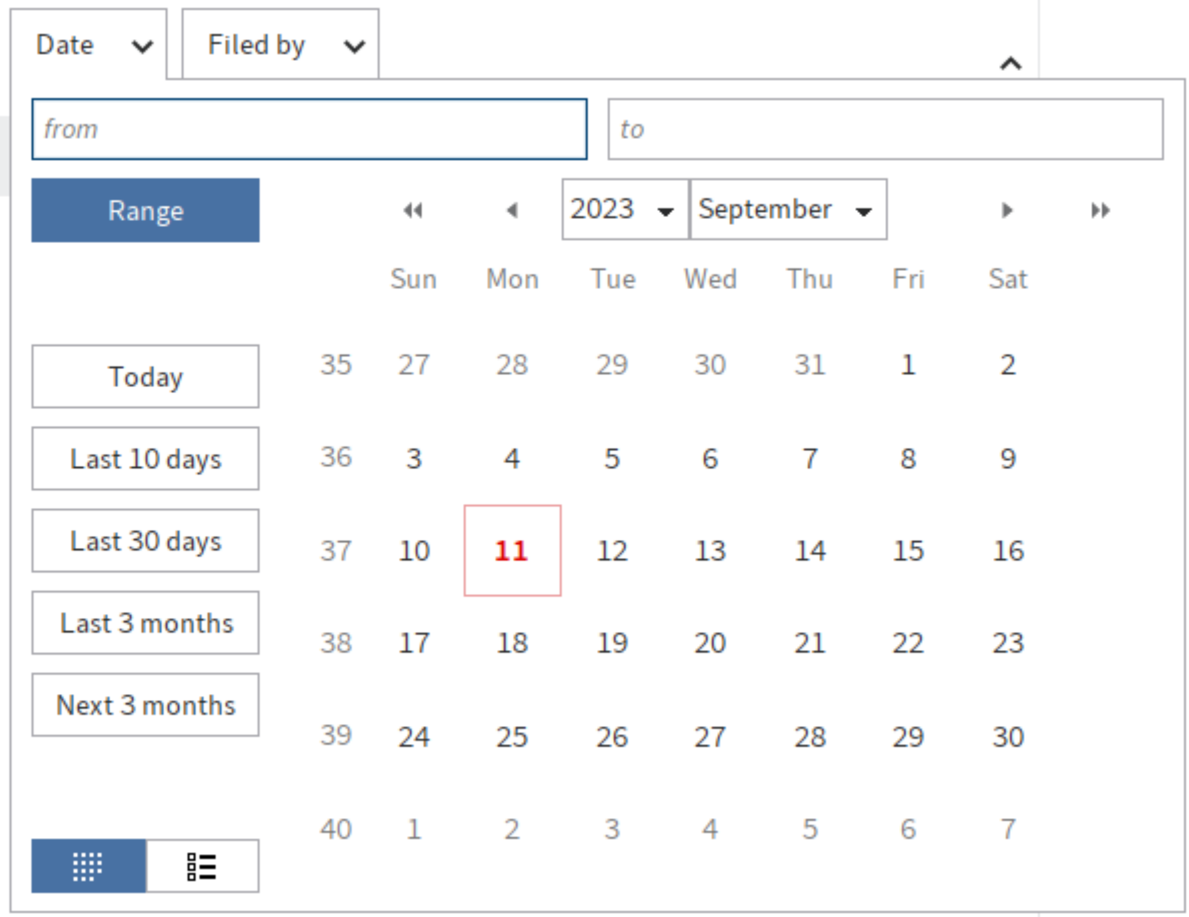
The filter for dates in the metadata offers a range of options for searching for a date. Searches always apply to an entire day, i.e. from `00:00:00` to `23:59:59`.



At the top of the figure, you will see an input field where you can enter an exact date. The ELO Java Client uses the date formats set in the configuration here. You can enter a time range using three periods to separate the dates, e.g. `May 1, 2020...May 10, 2020`. Complete your entry by pressing the ENTER key. The filter closes, and the time range is shown in a small gray box.

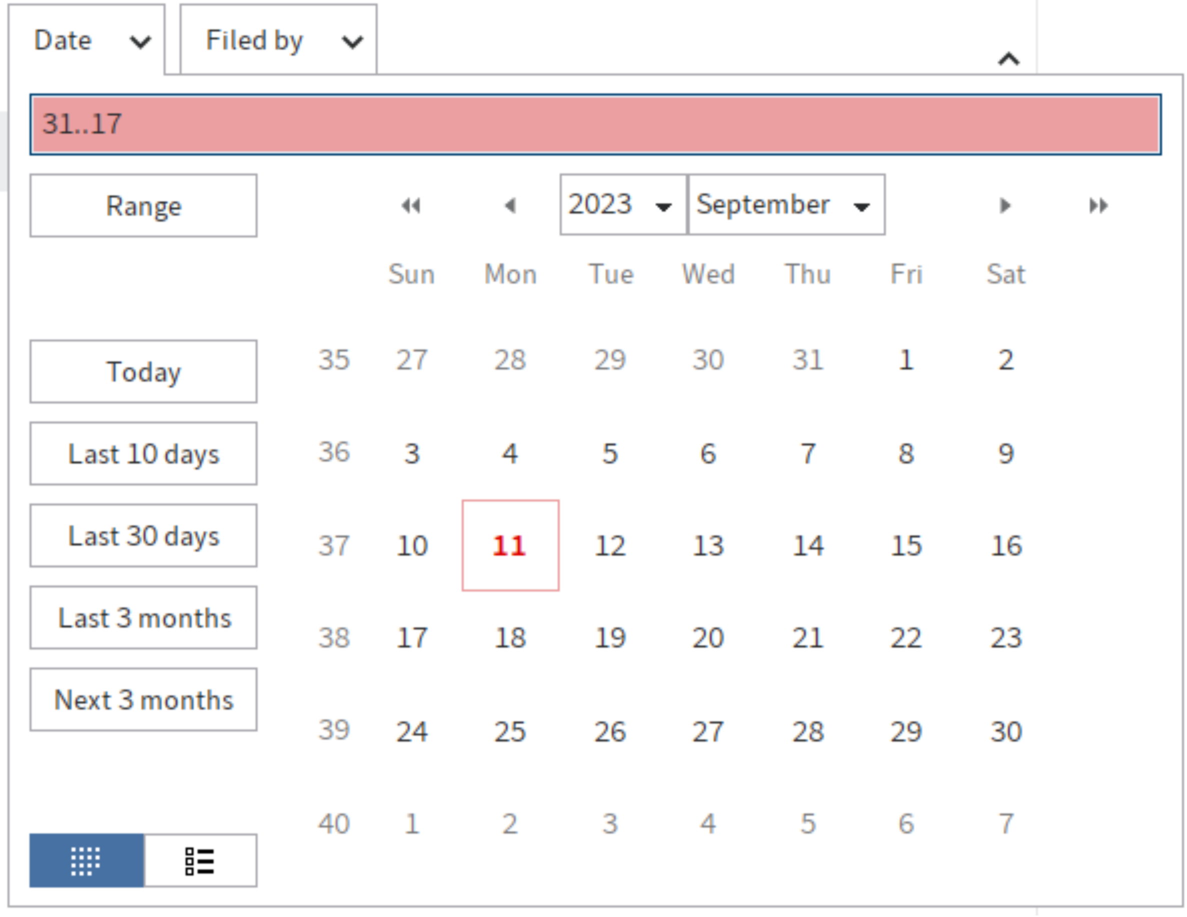


A more convenient way to enter a time range is using the *Range* mode with two input fields. After entering a `from` value, the focus automatically jumps to the `to` field, and you can complete both fields by selecting a date in the calendar.



	Sun	Mon	Tue	Wed	Thu	Fri	Sat
35	27	28	29	30	31	1	2
36	3	4	5	6	7	8	9
37	10	11	12	13	14	15	16
38	17	18	19	20	21	22	23
39	24	25	26	27	28	29	30
40	1	2	3	4	5	6	7

If the entry is not recognized as a date, the filter remains open when ENTER is pressed and the input field turns red to indicate an error.

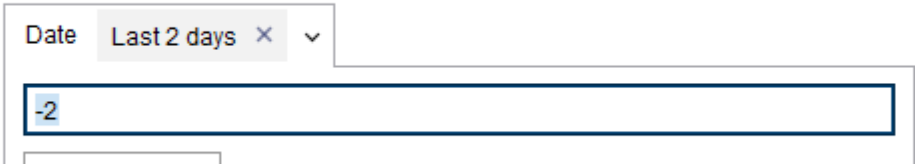


31..17

2023 September

	Sun	Mon	Tue	Wed	Thu	Fri	Sat
35	27	28	29	30	31	1	2
36	3	4	5	6	7	8	9
37	10	11	12	13	14	15	16
38	17	18	19	20	21	22	23
39	24	25	26	27	28	29	30
40	1	2	3	4	5	6	7

Relative time ranges can also be entered. Use plus and minus signs to set a time range in days, starting with the current date. Entering `-2` searches everything from two days before at midnight until 11:59 p.m. of the current day. Adding `m` allows you to use months. `+1m` searches everything from the current day at midnight to one month in the future (28, 29, 30, or 31 days depending on the month and year).

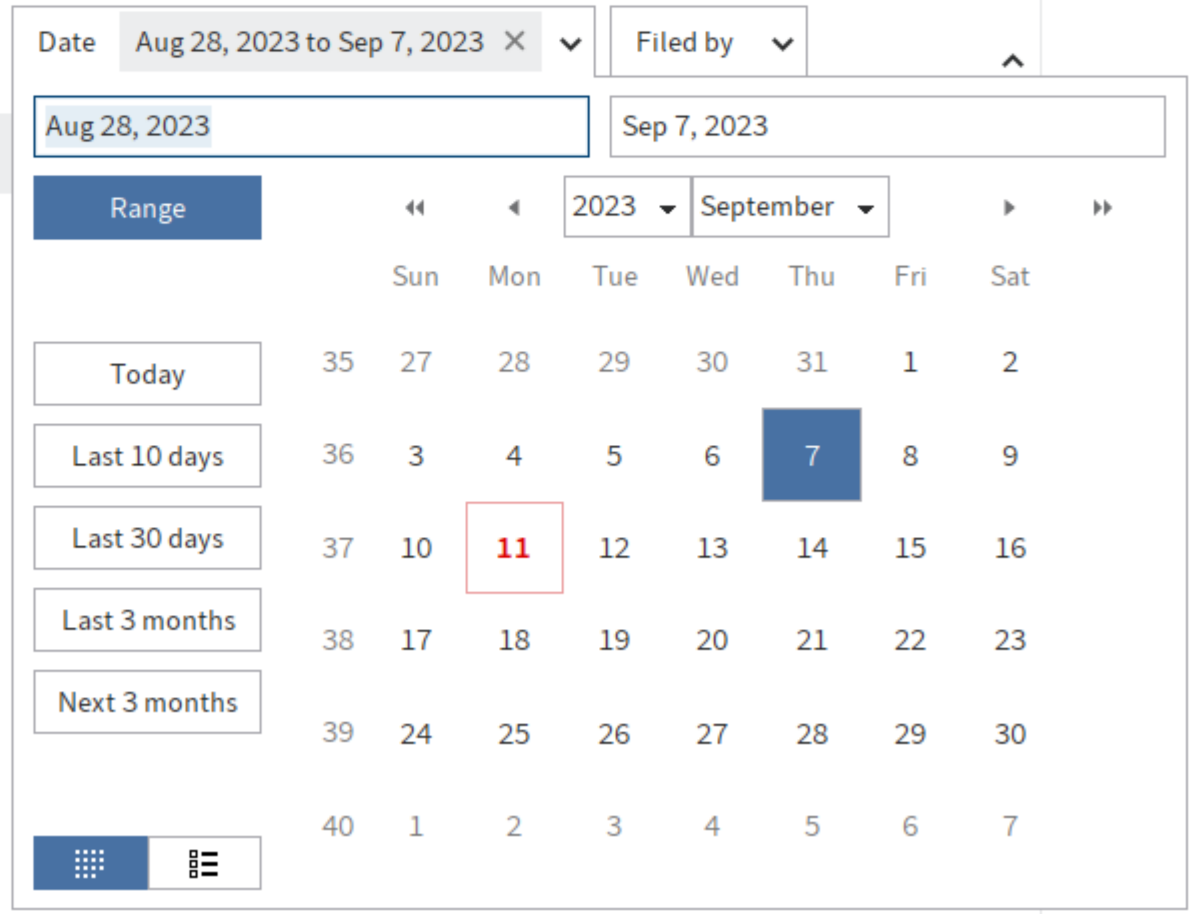


Last 2 days ×

-2

On the left, there are five buttons with typical time ranges. After being selected, the time range is entered to the input field in order to show the syntax used.

Dates can be selected using the calendar on the right. The current date is displayed in red. Navigate between months and years using the arrows above the calendar. You can select the required month or year using the respective pull-down menus for the month field and year field. Select an individual day with the mouse. Click on a second day to select a time range.



Date Aug 28, 2023 to Sep 7, 2023

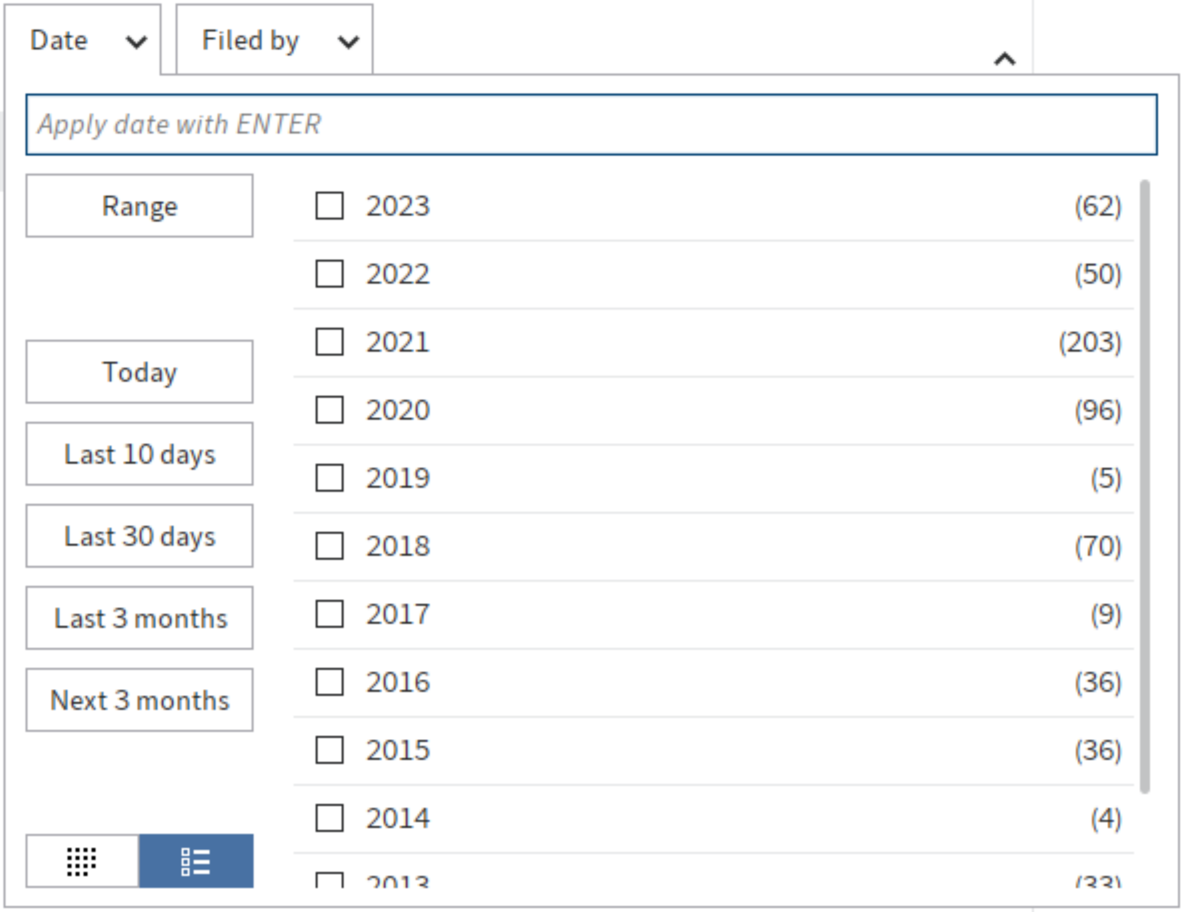
Aug 28, 2023

Sep 7, 2023

2023 September

	Sun	Mon	Tue	Wed	Thu	Fri	Sat
35	27	28	29	30	31	1	2
36	3	4	5	6	7	8	9
37	10	11	12	13	14	15	16
38	17	18	19	20	21	22	23
39	24	25	26	27	28	29	30
40	1	2	3	4	5	6	7

There are two more buttons at the bottom left that allow you to switch between the calendar and the year overview in the form of a list. The list form shows all available year values as well as the number of associated entries to the right in parentheses. You can select a single year (by selecting a list entry) or several years (by selecting the check boxes).

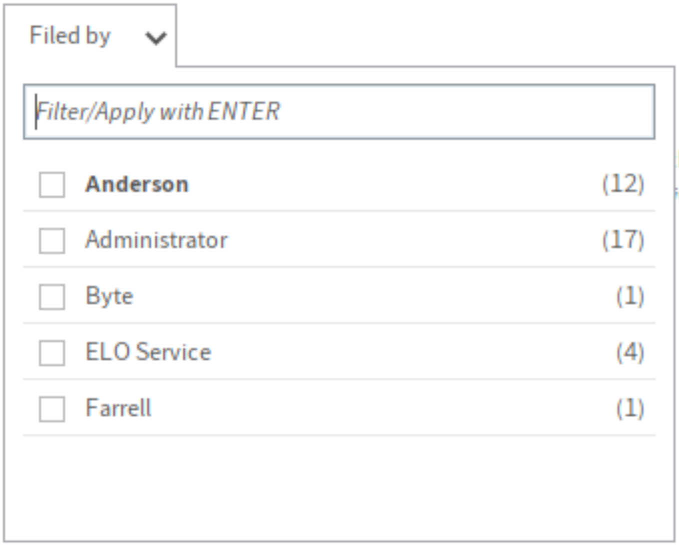


Apply date with ENTER

□ 2023 (62)
□ 2022 (50)
□ 2021 (203)
□ 2020 (96)
□ 2019 (5)
□ 2018 (70)
□ 2017 (9)
□ 2016 (36)
□ 2015 (36)
□ 2014 (4)
□ 2013 (33)

Filed by/Editor

The *Filed by* and *Editor* filters are similar to those for the metadata form and entry type. The user currently logged onto ELO is shown at the very top.



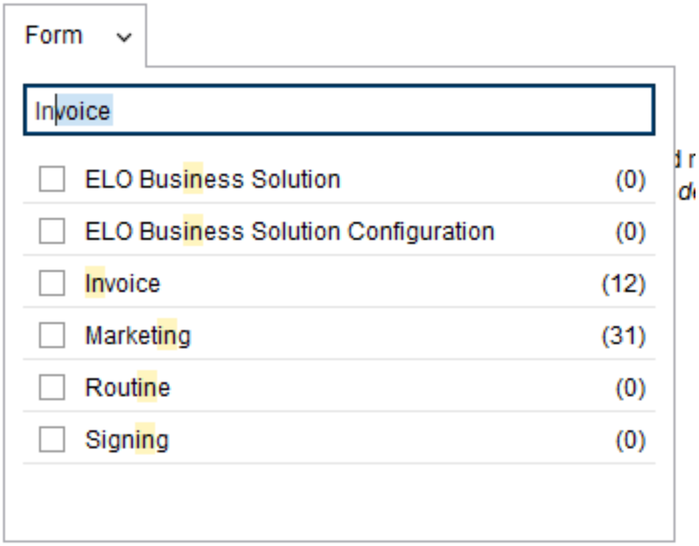
Filter/Apply with ENTER

□ **Anderson** (12)
□ Administrator (17)
□ Byte (1)
□ ELO Service (4)
□ Farrell (1)

Metadata form

The filter shows all metadata forms in use as well as the number of associated entries to the right in parentheses. The metadata forms are sorted by alphabetical order. You can select a single form or multiple forms using the check boxes.

When you enter text into the filter field, the *search-as-you-type* function automatically starts. Your current entry is instantly compared with possible values. This narrows down the list so that only metadata forms containing the entered text are shown. It does not matter where the text appears. In the following example, entering `In` finds both `Invoice` and `Marketing`. The matching text is then highlighted in light yellow.



Invoice

☐ ELO Business Solution (0)
☐ ELO Business Solution Configuration (0)
☐ Invoice (12)
☐ Marketing (31)
☐ Routine (0)
☐ Signing (0)

When typing, autocomplete also provides suggestions. Your entry is interpreted as the beginning of the metadata form name, so the matching form with the most entries is entered to the input field as extra text and selected. Press ENTER to accept and apply the autocomplete suggestion, or keep typing to search for another metadata form.

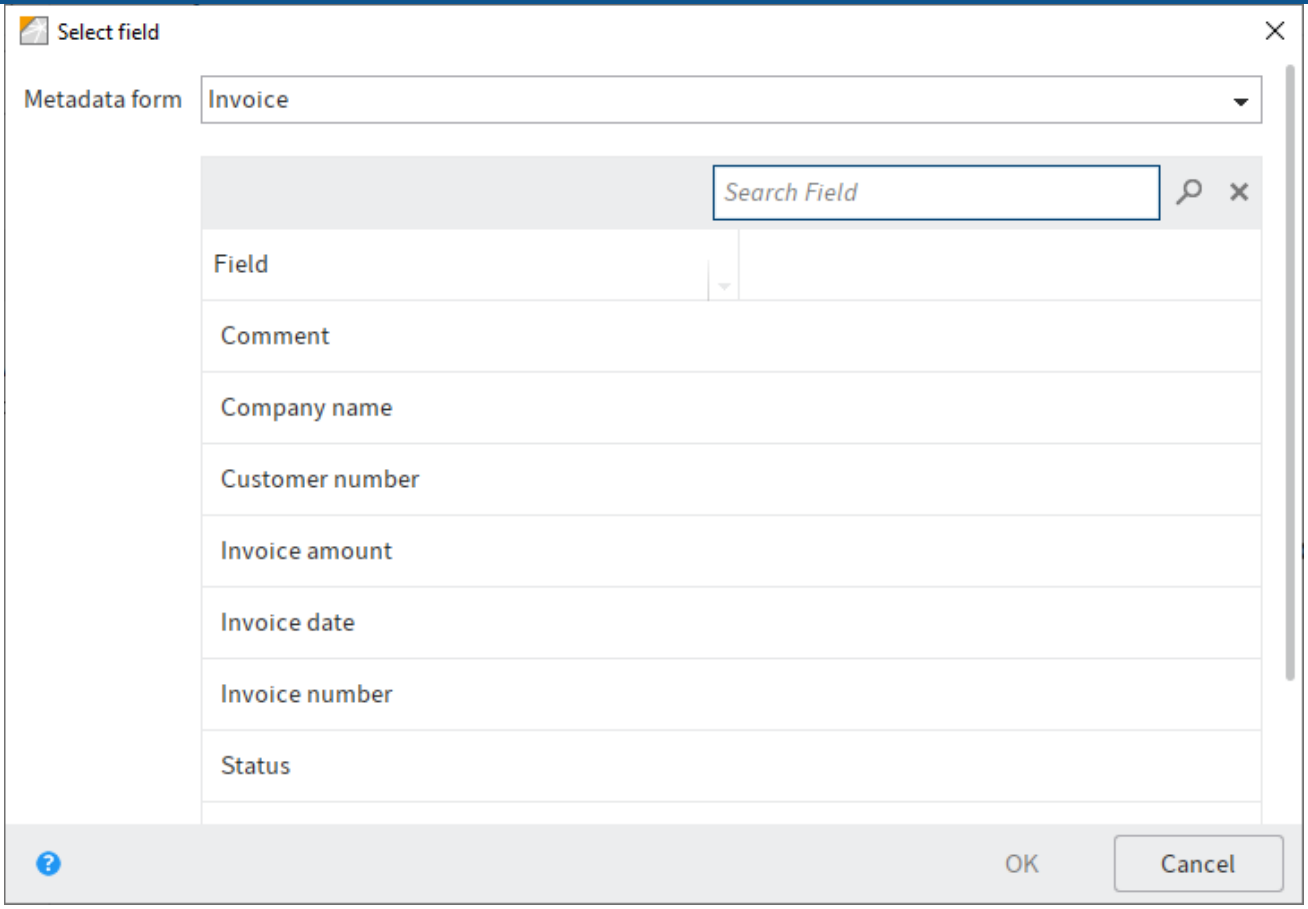
All fields

This filter searches all configured fields in all metadata forms. It uses the same linguistic rules as the iSearch input field and provides an easy way to differentiate between searches in the full text database (via input fields) and in the metadata (with this filter).

If you want to find related terms, numeric values, or dates, it is better to restrict the search to a specific field.

Select field

You can manually add filters for specific fields. In the dialog box, first select the metadata form, then the fields. If you have already narrowed down your search to a form using a form filter, this is pre-selected in the dialog box.



Metadata form: Invoice

Field

Comment

Company name

Customer number

Invoice amount

Invoice date

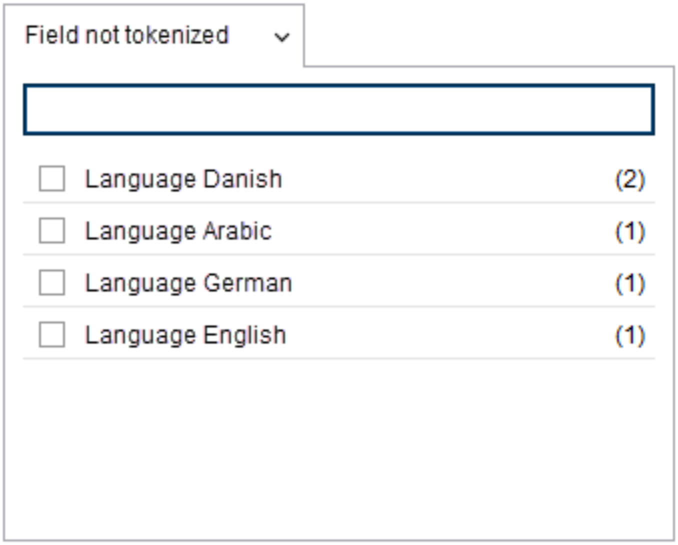
Invoice number

Status

> **Information**
>
> The fields can simply be selected in the metadata form. However, the field filter applies for all forms if the field template for the selected field is also used in other metadata forms. Restrict the search to a metadata form using a form filter.

The field filter is compatible with search rule and pattern methods or complete values. This is a field template configuration.

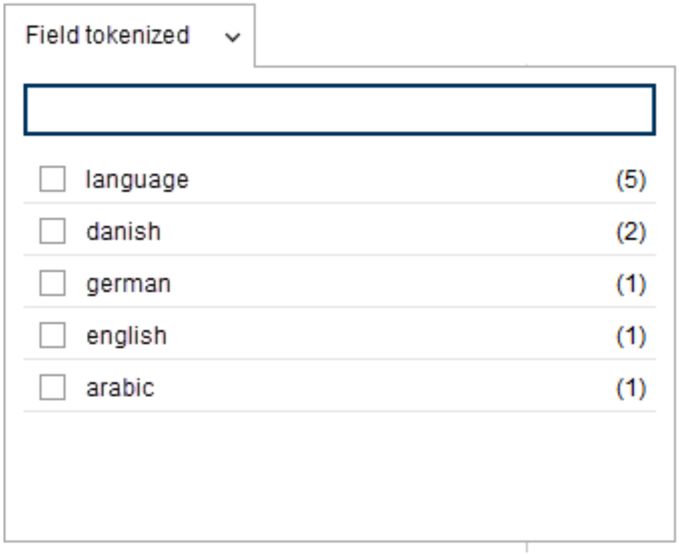
The *Disable tokenization* option (see Tokenization section) for a field template disables the search rules and patterns for this field. The exact values in the metadata appear in the filter list. For this field, the search also works with exact values. In this case, partial terms are not taken into account.



Field not tokenized

□ Language Danish	(2)
□ Language Arabic	(1)
□ Language German	(1)
□ Language English	(1)

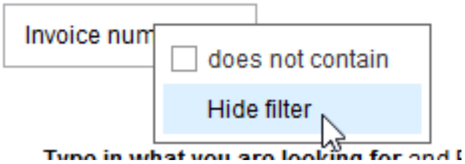
When search rules and patterns are enabled, the filter list only contains individual words – both the original words as well as their root forms. The search options in this filter correspond to those of the search input field.



Field tokenized

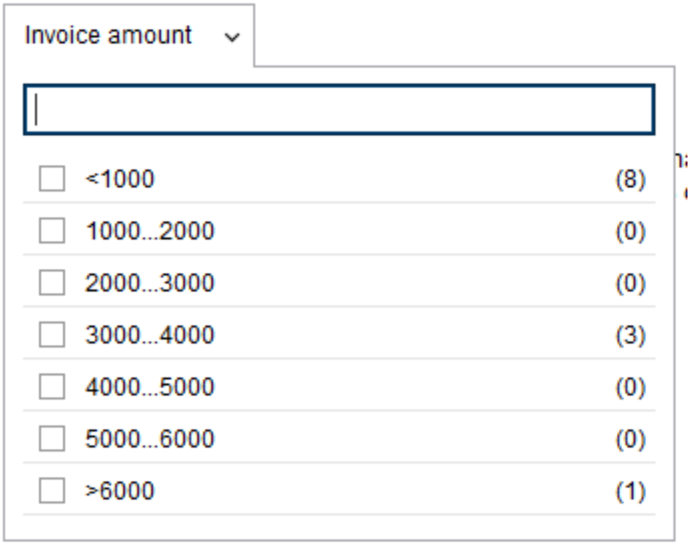
□ language	(5)
□ danish	(2)
□ german	(1)
□ english	(1)
□ arabic	(1)

You can also hide specific filters via the context menu.



Numeric fields

In the case of fields that are configured as numeric data, the filter list shows the value ranges as search suggestions.



Invoice amount

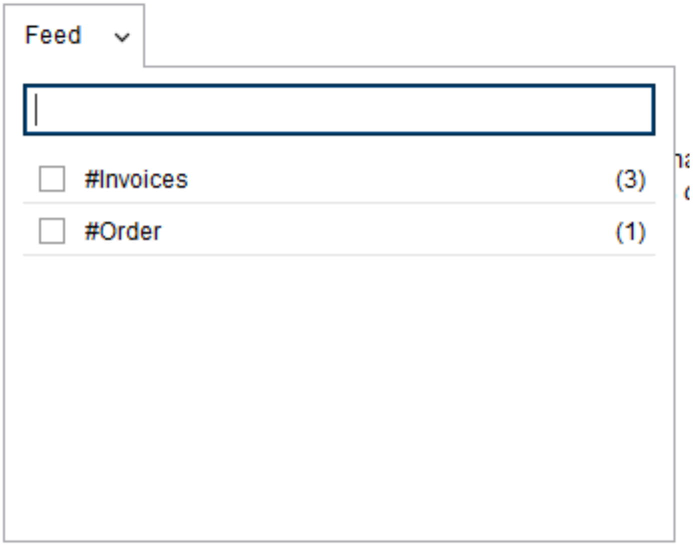
□	<1000	(8)
□	1000...2000	(0)
□	2000...3000	(0)
□	3000...4000	(3)
□	4000...5000	(0)
□	5000...6000	(0)
□	>6000	(1)

You can also enter ranges to the filter with three periods as separators, with `100...900` corresponding to a search from 100 to 900. You can also search using *less than… (<)* and *greater than…()* e.g. `>2500`. When you enter text and press the ENTER key, the filter input field turns red if your input is invalid.

Feed

The filter for the search in the feed allows you to search all feed content. This includes both posts and comments created manually as well as posts generated automatically, such as those related to a new document version. The feed also contains the version numbers and version comments for both the current working version of a document and older versions.

The feed filter list shows the most popular hashtags as search suggestions. As with all other entries, these are searched based on the standard linguistic rules in the feed text content.

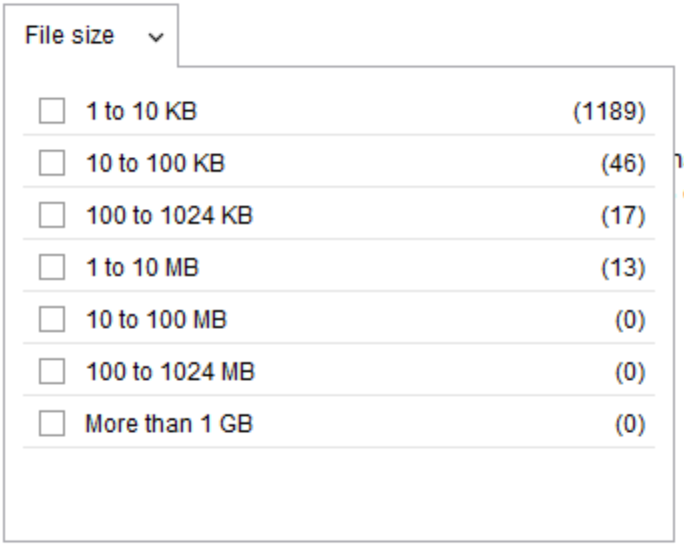


Feed

□	#Invoices	(3)
□	#Order	(1)

File size

The document file size filter is the only filter without an input field. You can select a size range from the available list.

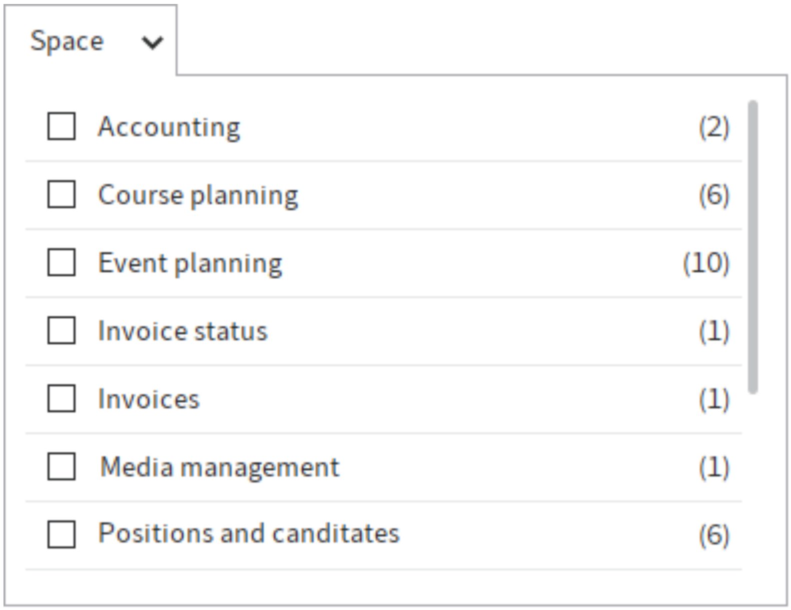


File size ⌄

□	1 to 10 KB	(1189)
□	10 to 100 KB	(46)
□	100 to 1024 KB	(17)
□	1 to 10 MB	(13)
□	10 to 100 MB	(0)
□	100 to 1024 MB	(0)
□	More than 1 GB	(0)

Search space

The ELO Spaces filter enables you to run a search in a selected space (Teamspace or Workspace).

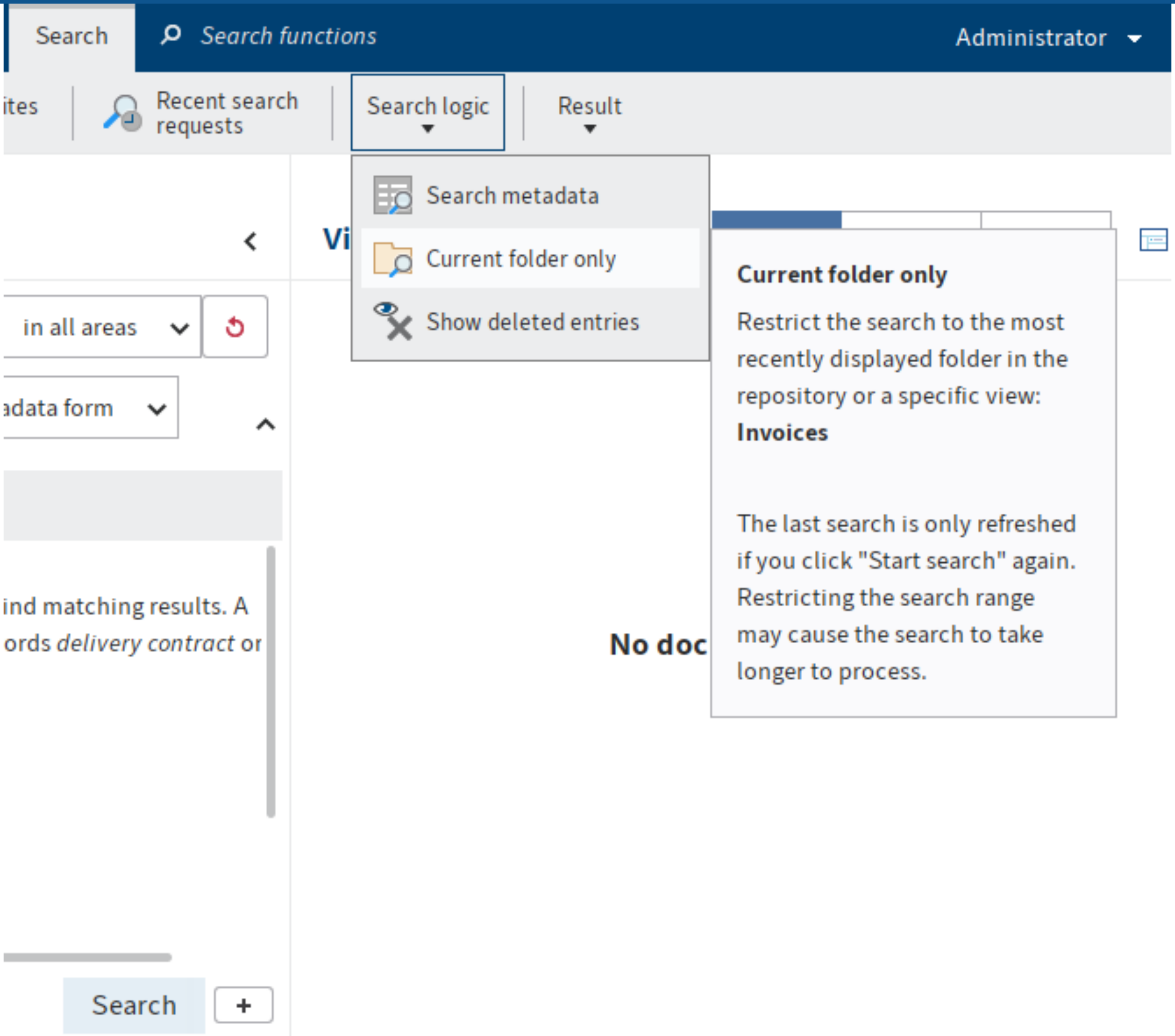


Space ⌄

□	Accounting	(2)
□	Course planning	(6)
□	Event planning	(10)
□	Invoice status	(1)
□	Invoices	(1)
□	Media management	(1)
□	Positions and canditates	(6)

Search in the current folder

The *Current folder only* search option is located on the ribbon and narrows the ELO iSearch down to the most recently used folder. The tooltip shows the current folder.



Current folder only

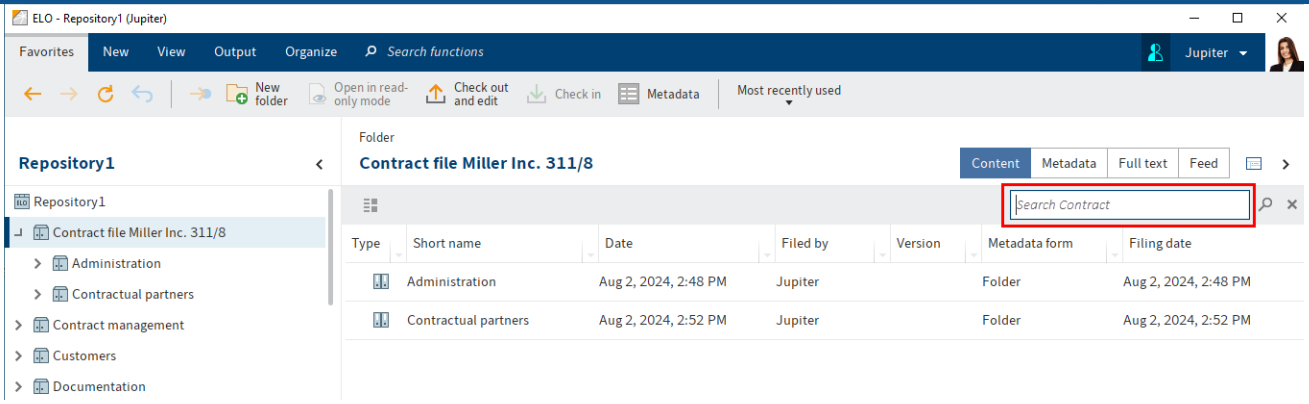
Restrict the search to the most recently displayed folder in the repository or a specific view: **Invoices**

The last search is only refreshed if you click "Start search" again. Restricting the search range may cause the search to take longer to process.

Since this setting re-filters the results from the search index, it also slows down the search significantly. You should only use this setting if you have applied other search criteria and are certain there will be a relatively small amount of results. Search filters are ideal for this purpose.

Search via the relationship to regions/business objects

For business objects, you start a search within the region of a business object via the search field in the viewer pane. The system searches for the entered short name, restricted to the region of the business object, including the child folders belonging to the region. At the technical level, this search is a filter by region ID.



Type	Short name	Date	Filed by	Version	Metadata form	Filing date
	Administration	Aug 2, 2024, 2:48 PM	Jupiter		Folder	Aug 2, 2024, 2:48 PM
	Contractual partners	Aug 2, 2024, 2:52 PM	Jupiter		Folder	Aug 2, 2024, 2:52 PM

For fields passed on through regions, objects located in the same region can also be found using the *Search* work area (you will find more information under ELO packages > Metadata > 'Usage' tab > Search and ELO Server - Installation and operation > New features > ELO 23.3 changes > ELO Indexserver and ELO iSearch > Customizations to support regions > Text search in ELO iSearch via the fields passed on through regions).

Please note

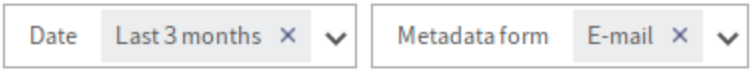
Passing on fields only applies to gen. 2 elements, so only these elements are found and not gen. 1 elements.

Save search

In ELO iSearch, you can save your search entries. Use the *Save search* function (*Ribbon > Search*).

This saves the entire ELO iSearch entry, including the selected filters and filter values. The *Current folder only* setting (on the ribbon) is not saved, as this has an effect on search performance and should therefore only be used in exceptional cases.

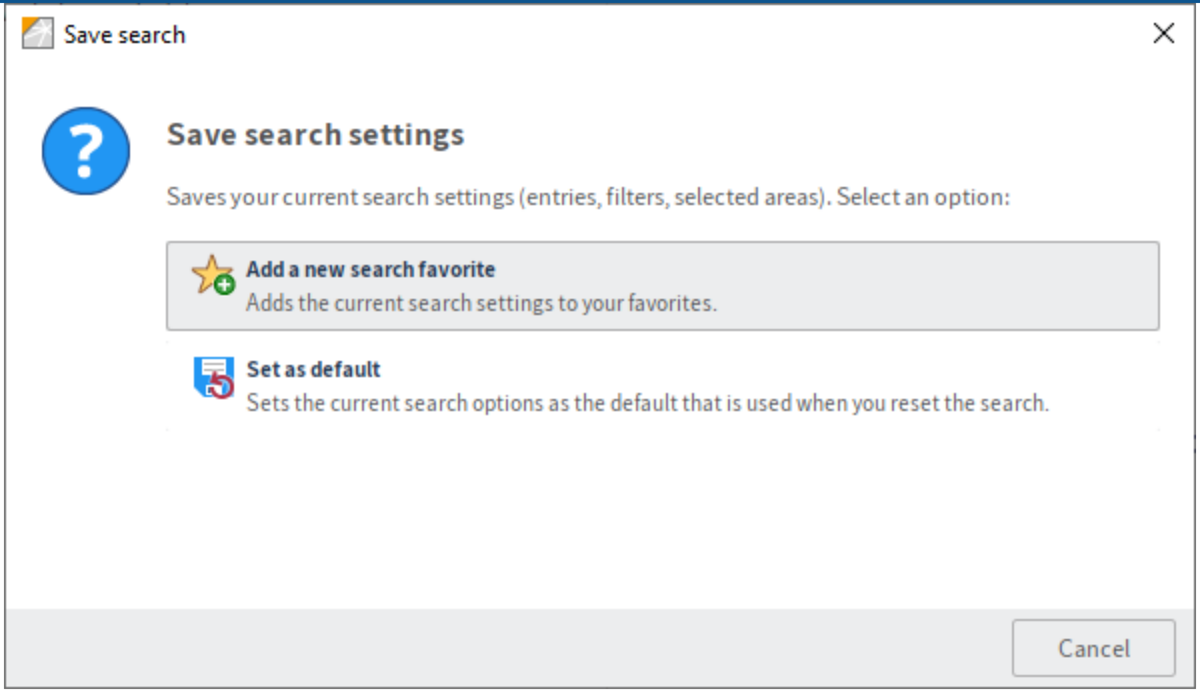
For example, you can create a search favorite for a search for current e-mails. Then, all you have to do is enter your search term. The metadata form and time range will be set automatically.



Date: Last 3 months ✕

Metadata form: E-mail ✕

When saving the search, you have two options: to create a new search favorite or to change the default filter.



Save search

Save search settings

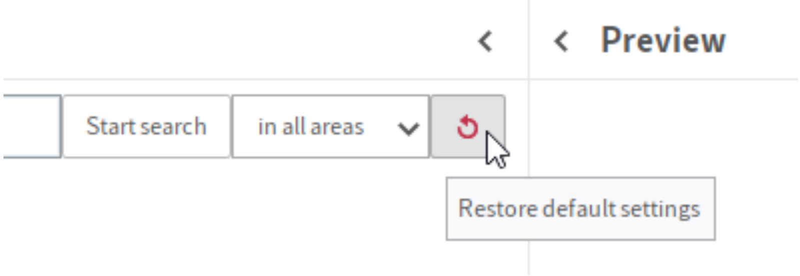
Saves your current search settings (entries, filters, selected areas). Select an option:

Add a new search favorite
Adds the current search settings to your favorites.

Set as default
Sets the current search options as the default that is used when you reset the search.

Default filters

You can customize the default filters described in the Filters chapter to meet your requirements with the *Save search* function. Your personal settings are applied when you start the client and can be reset at any time by selecting the *Restore default settings* button.

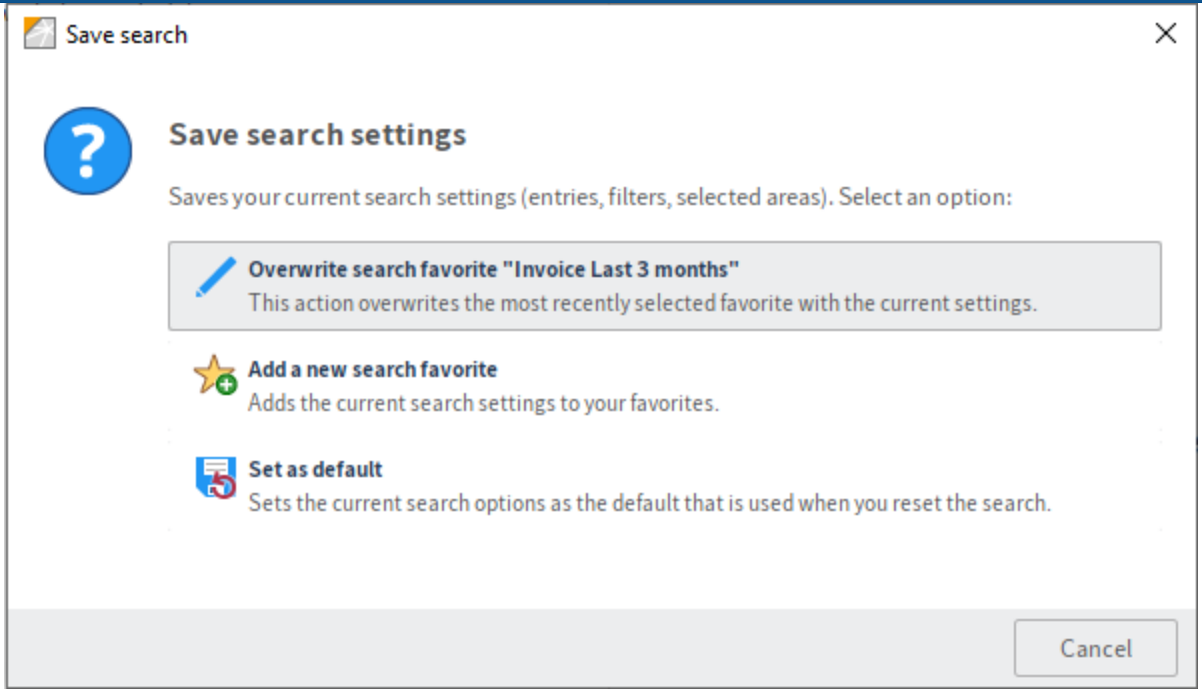


If you make a change to a search filter you are using, you first restore the saved search filter. In this case, select *Restore default settings* again to restore the default filters.

Search favorites

Search favorites make it easy to perform different recurring searches. Favorites can be displayed on the ribbon. After a restart, they will also appear as tiles in your *My ELO* workspace in the ELO Java Client and ELO Web Client. When creating a search favorite, you are requested to assign it a unique name.

If you are using a search favorite and then save your search, the dialog box gives you the option to change your current search favorite:



Save search

Save search settings

Saves your current search settings (entries, filters, selected areas). Select an option:

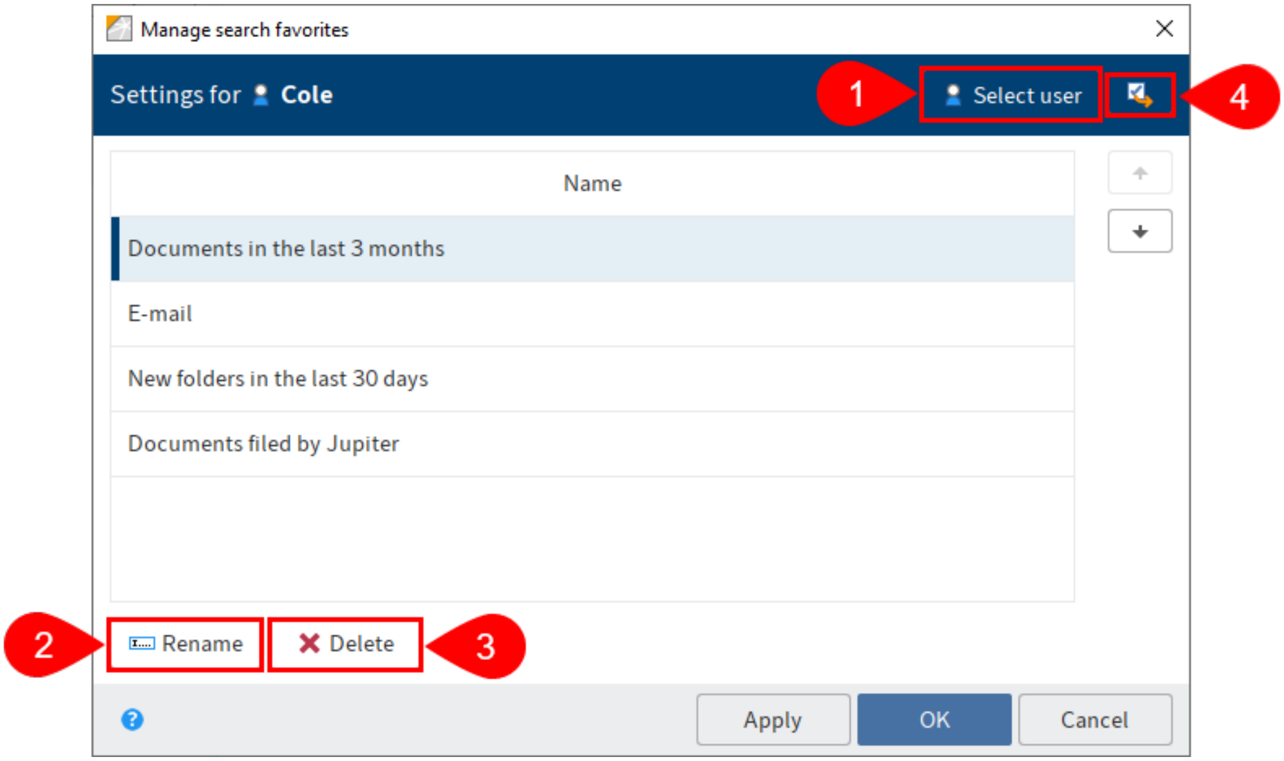
Overwrite search favorite "Invoice Last 3 months"
This action overwrites the most recently selected favorite with the current settings.

Add a new search favorite
Adds the current search settings to your favorites.

Set as default
Sets the current search options as the default that is used when you reset the search.

Managing search filters

The administrator can define global search filters for other ELO accounts. On the *Search favorites* tab, navigate to the *Manage search favorites* function.



Manage search favorites

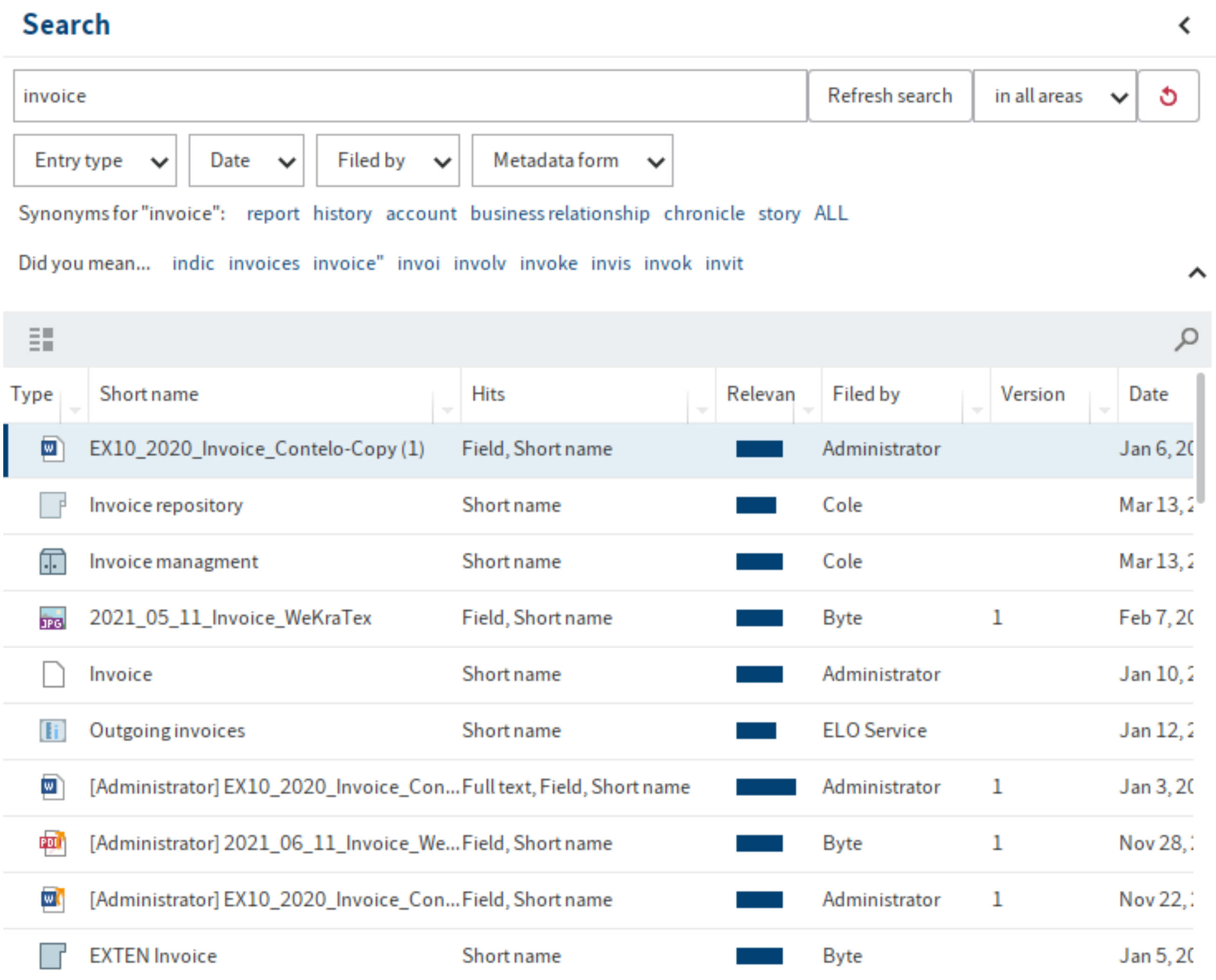
Settings for Cole

Name
Documents in the last 3 months
E-mail
New folders in the last 30 days
Documents filed by Jupiter

Under *Select user* (1), you can select whose favorites you want to manage. You can rename (2) or delete (3) search favorites for users or option groups. Select *Transfer selected settings to other* *users* (4) to assign search favorites you have selected to other users, option groups, and globally. The default filter settings are shown in the dialog box if saved for the selected user.

Results list

The result of your search is shown in a table (results list) with entries matching your search query. Entries are sorted in descending order by their filing date, i.e. the newest documents are shown at the top of the results list. You can change how the results list is sorted at any time based on your needs and the values and information relevant to the search result.



Synonyms for "invoice": report history account business relationship chronicle story ALL

Did you mean... indic invoices invoice" invoi involv invoke invis invok invit

Type	Short name	Hits	Relevan	Filed by	Version	Date
W	EX10_2020_Invoice_Contelo-Copy (1)	Field, Short name	■	Administrator		Jan 6, 20
	Invoice repository	Short name	■	Cole		Mar 13, 2
	Invoice managment	Short name	■	Cole		Mar 13, 2
JPG	2021_05_11_Invoice_WeKraTex	Field, Short name	■	Byte	1	Feb 7, 20
	Invoice	Short name	■	Administrator		Jan 10, 2
	Outgoing invoices	Short name	■	ELO Service		Jan 12, 2
W	[Administrator] EX10_2020_Invoice_Con...	Full text, Field, Short name	■	Administrator	1	Jan 3, 20
PDF	[Administrator] 2021_06_11_Invoice_We...	Field, Short name	■	Byte	1	Nov 28,
W	[Administrator] EX10_2020_Invoice_Con...	Field, Short name	■	Administrator	1	Nov 22,
	EXTEN Invoice	Short name	■	Byte		Jan 5, 20

> **Information**
>
> When displaying the matches found in the document in the table view (see corresponding column), all parts of the document matching the search entry are displayed, even if a specific match was excluded from the search area (see Search area/fields searched section). The document in question was only found via the specified search area, but it can be helpful to know of other text matches occurring in the document.

Information

Hits are only shown in the results list if a text search was performed via the search input field (also possible in combination with filters). If the search input field is empty or contains the wildcard character *, because you are only filtering via fields for example, no hits are shown.

Number of results and pages

To speed up the search, a parameter is available that limits the maximum number of results to be shown. This parameter is configured on the server and can be adjusted based on the requirements. The relevant permissions are required for this.

If this number/parameter value is exceeded, older documents are not shown in the search result. By using a date filter, you can access these documents specifically or exclude them from the search.

The ELO Java Client shows the results divided into individual results pages. This is necessary since the results are prepared linearly based on the number of hits. Use the left and right arrows to navigate between pages.

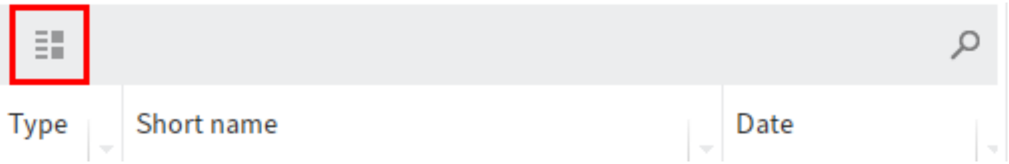
Results 1 to 100 of 568

Please note

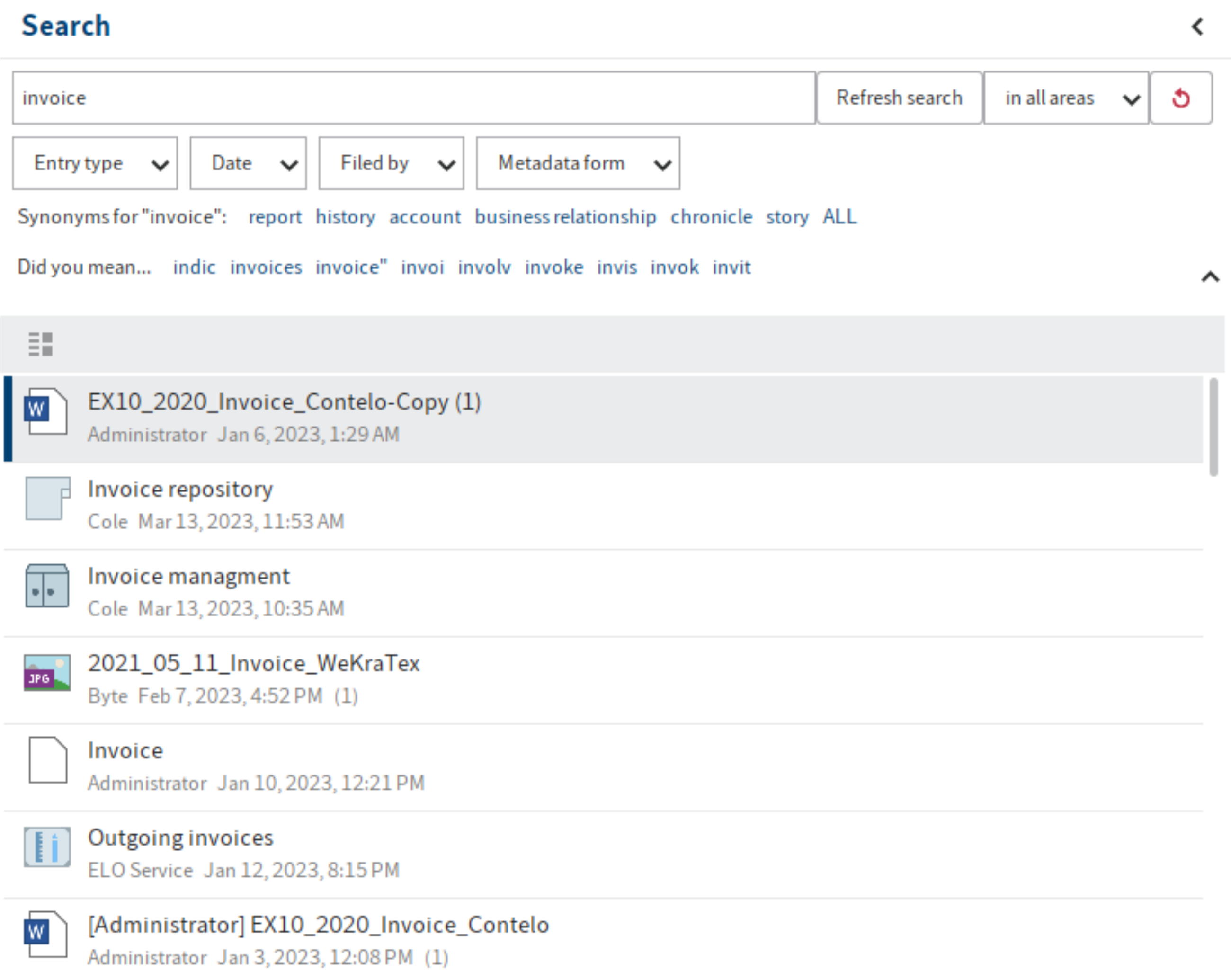
The total number shown here is only an estimate. Permissions in ELO may change over time, such as when new documents are filed to ELO. For this reason, the Indexserver filters all entries again, meaning fewer or more results may be shown. The client adjusts the value displayed as soon as the search is completed and the entries have been filtered again. The same happens when searching in the current folder. These values may differ significantly since the estimate always refers to the entire repository.

List, tile, and tree views

Besides the table view, the ELO Java Client also offers a list, tile, or tree view to display the search results. You will find the display options on the gray toolbar.



The list view is especially compact, making it ideal for providing a clear overview of the number of results and leaving a great amount of space to preview documents on the user's screen.



invoice

Synonyms for "invoice": report history account business relationship chronicle story ALL

Did you mean... indic invoices invoice" invoi involv invoke invis invok invit

EX10_2020_Invoice_Contelo-Copy (1)
Administrator Jan 6, 2023, 1:29 AM

Invoice repository
Cole Mar 13, 2023, 11:53 AM

Invoice managment
Cole Mar 13, 2023, 10:35 AM

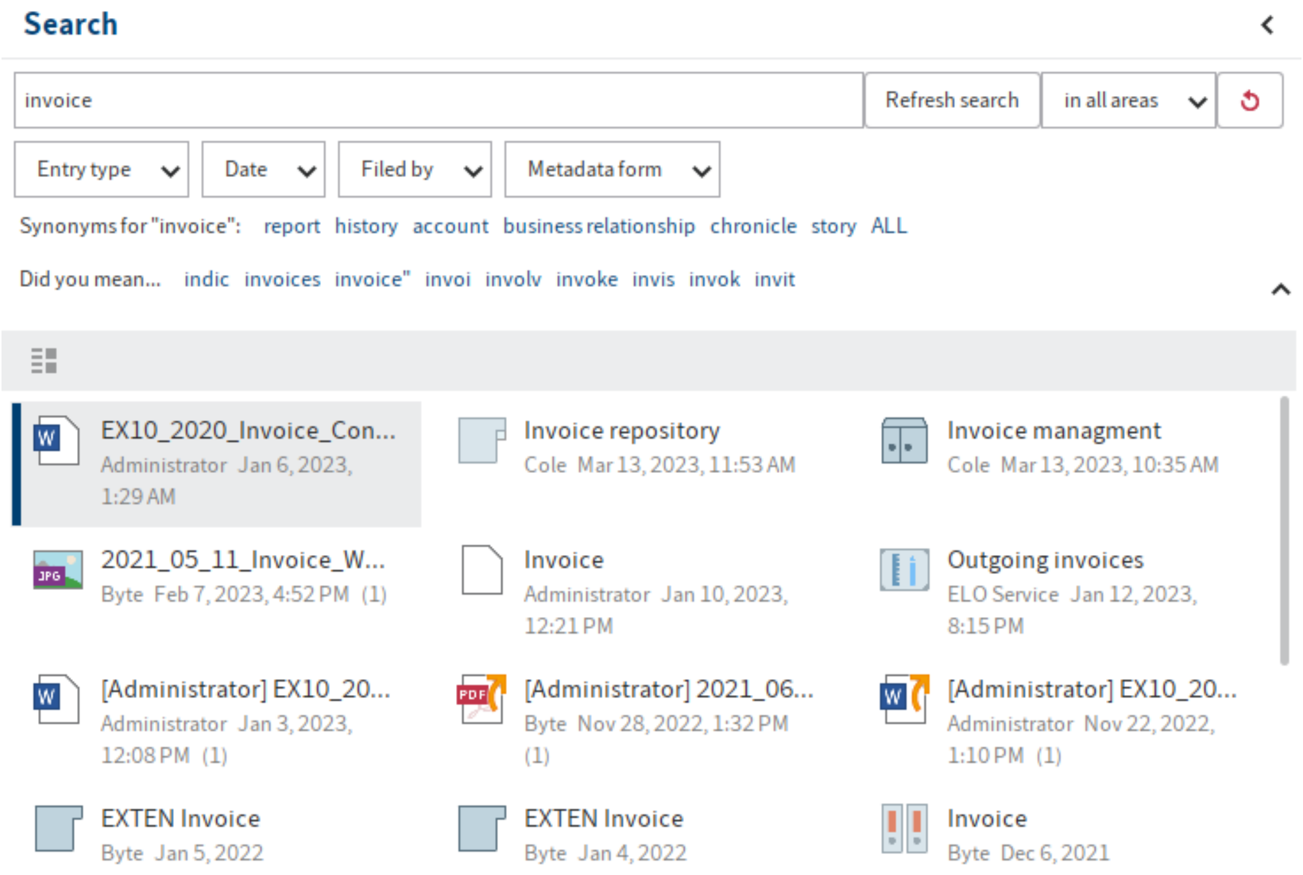
2021_05_11_Invoice_WeKraTex
Byte Feb 7, 2023, 4:52 PM (1)

Invoice
Administrator Jan 10, 2023, 12:21 PM

Outgoing invoices
ELO Service Jan 12, 2023, 8:15 PM

[Administrator] EX10_2020_Invoice_Contelo
Administrator Jan 3, 2023, 12:08 PM (1)

The tile view is especially suitable for thumbnails. Please remember that these are generated by the client, so all documents have to be downloaded from the server first. This view is suitable for manageable amounts of results on a local network.



invoice

Synonyms for "invoice": report history account business relationship chronicle story ALL

Did you mean... indic invoices invoice" invoi involv invoke invis invok invit

EX10_2020_Invoice_Con... Administrator Jan 6, 2023, 1:29 AM

Invoice repository Cole Mar 13, 2023, 11:53 AM

Invoice managment Cole Mar 13, 2023, 10:35 AM

2021_05_11_Invoice_W... Byte Feb 7, 2023, 4:52 PM (1)

Invoice Administrator Jan 10, 2023, 12:21 PM

Outgoing invoices ELO Service Jan 12, 2023, 8:15 PM

[Administrator] EX10_20... Administrator Jan 3, 2023, 12:08 PM (1)

[Administrator] 2021_06... Byte Nov 28, 2022, 1:32 PM (1)

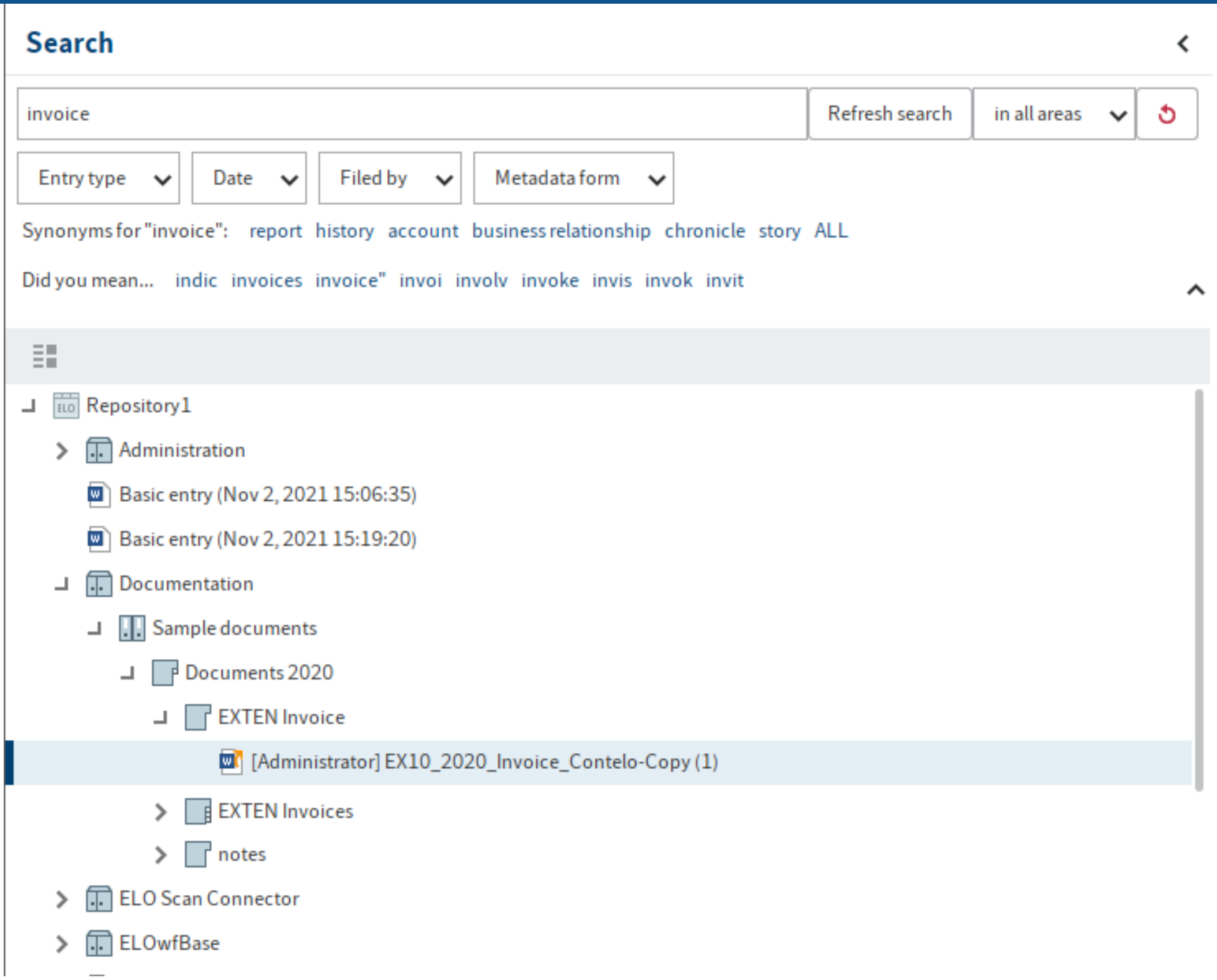
[Administrator] EX10_20... Administrator Nov 22, 2022, 1:10 PM (1)

EXTEN Invoice Byte Jan 5, 2022

EXTEN Invoice Byte Jan 4, 2022

Invoice Byte Dec 6, 2021

The tree view allows you to arrange the results in a tree structure based on different criteria. One option is to display the result according to the structure in ELO. You can also define your own structures using the fields of the entries. For the *Invoice* metadata form, for example, a structure by year and company name would be possible as long as the metadata form contained the required values as fields.



invoice

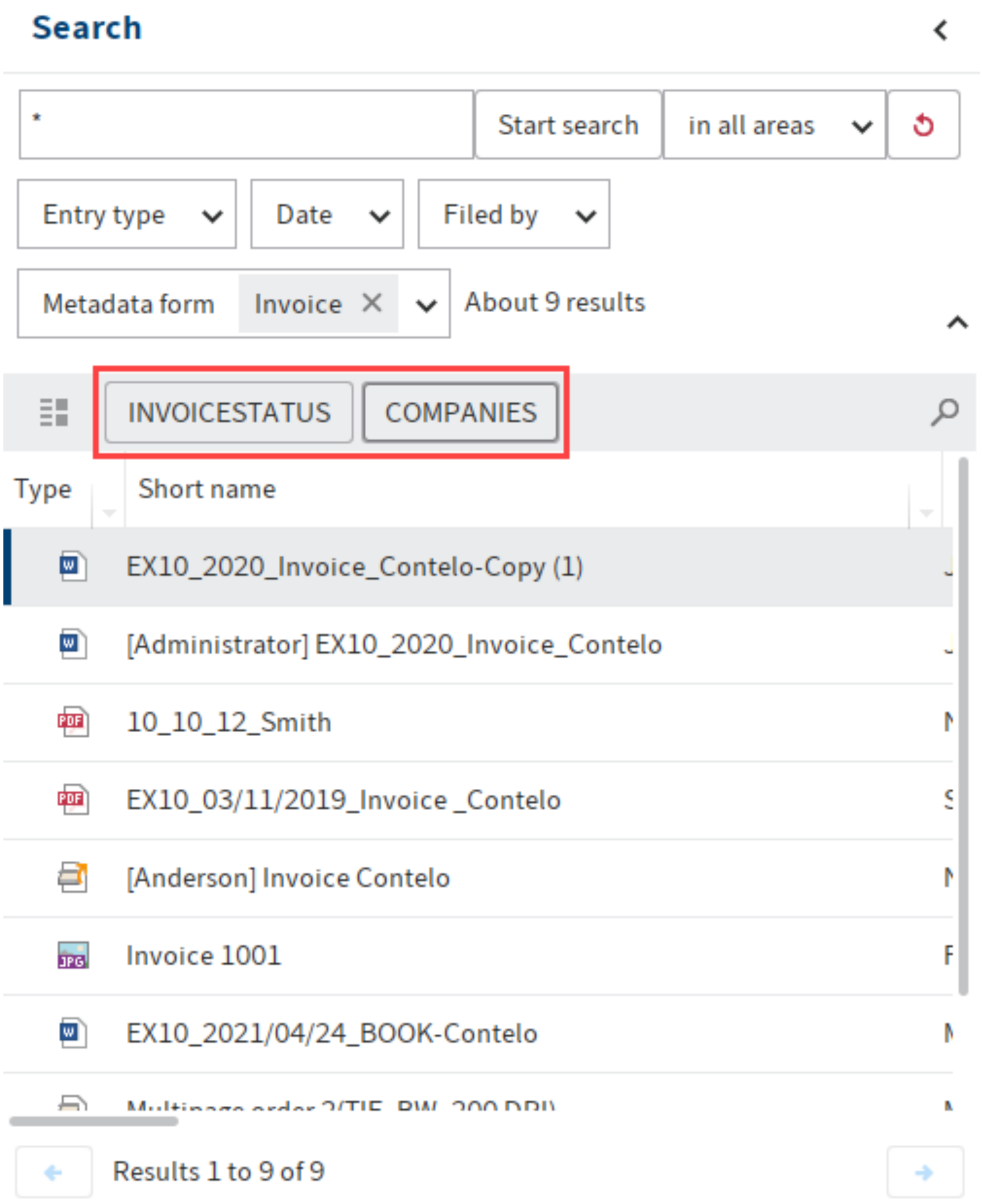
Synonyms for "invoice": report history account business relationship chronicle story ALL

Did you mean... indic invoices invoice" invoi involv invoke invis invok invit

- Repository1
  - Administration
  - Basic entry (Nov 2, 2021 15:06:35)
  - Basic entry (Nov 2, 2021 15:19:20)
  - Documentation
    - Sample documents
      - Documents 2020
        - EXTEN Invoice
          - [Administrator] EX10_2020_Invoice_Contelo-Copy (1)
        - EXTEN Invoices
        - notes
  - ELO Scan Connector
  - ELOwfBase

Dashboards

In the *Search* work area, buttons are available that allow you to launch dashboards after conducting a search with a form filter. These buttons are only displayed if dashboards have been stored for the corresponding metadata form.



Search: *

Metadata form: Invoice

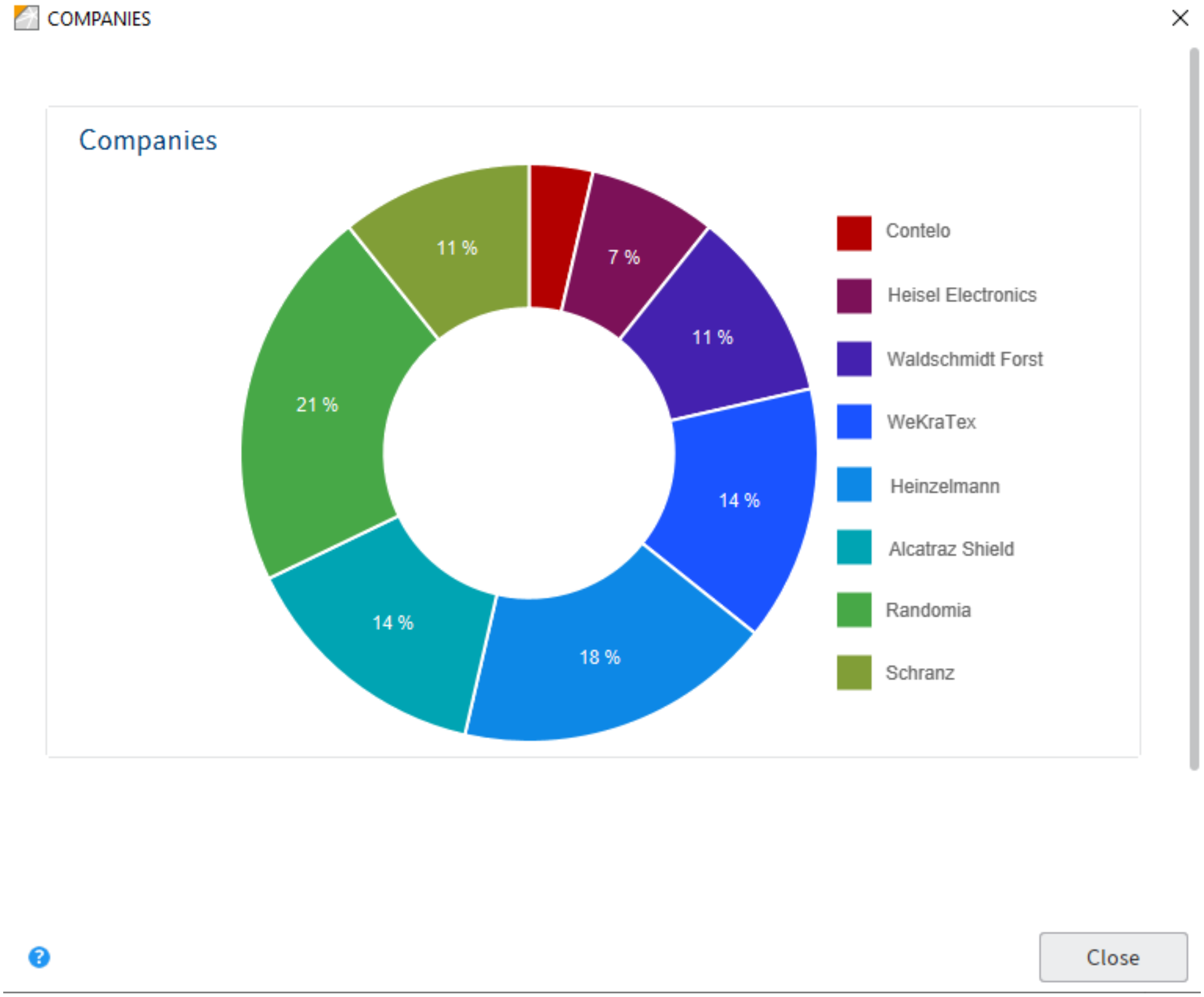
About 9 results

Type	Short name
W	EX10_2020_Invoice_Contelo-Copy (1)
W	[Administrator] EX10_2020_Invoice_Contelo
PDF	10_10_12_Smith
PDF	EX10_03/11/2019_Invoice _Contelo
	[Anderson] Invoice Contelo
JPG	Invoice 1001
W	EX10_2021/04/24_BOOK-Contelo
	Multipage order 2(TIF, BW, 200 DPI)

Results 1 to 9 of 9

You can store the dashboards in the ELO Administration Console.

The dashboards allow you to visualize metadata of entries filed with the same metadata form in different ways. You can use bar charts, line charts, and doughnut charts.



If you select a dashboard button, a new dialog box opens. You can move the mouse over the dashboards to view details about them.

For information on how to create dashboards, refer to the *ELO packages* documentation, *Metadata > Metadata forms > 'Content' tab > Create dashboards*.

Technical description

Architecture

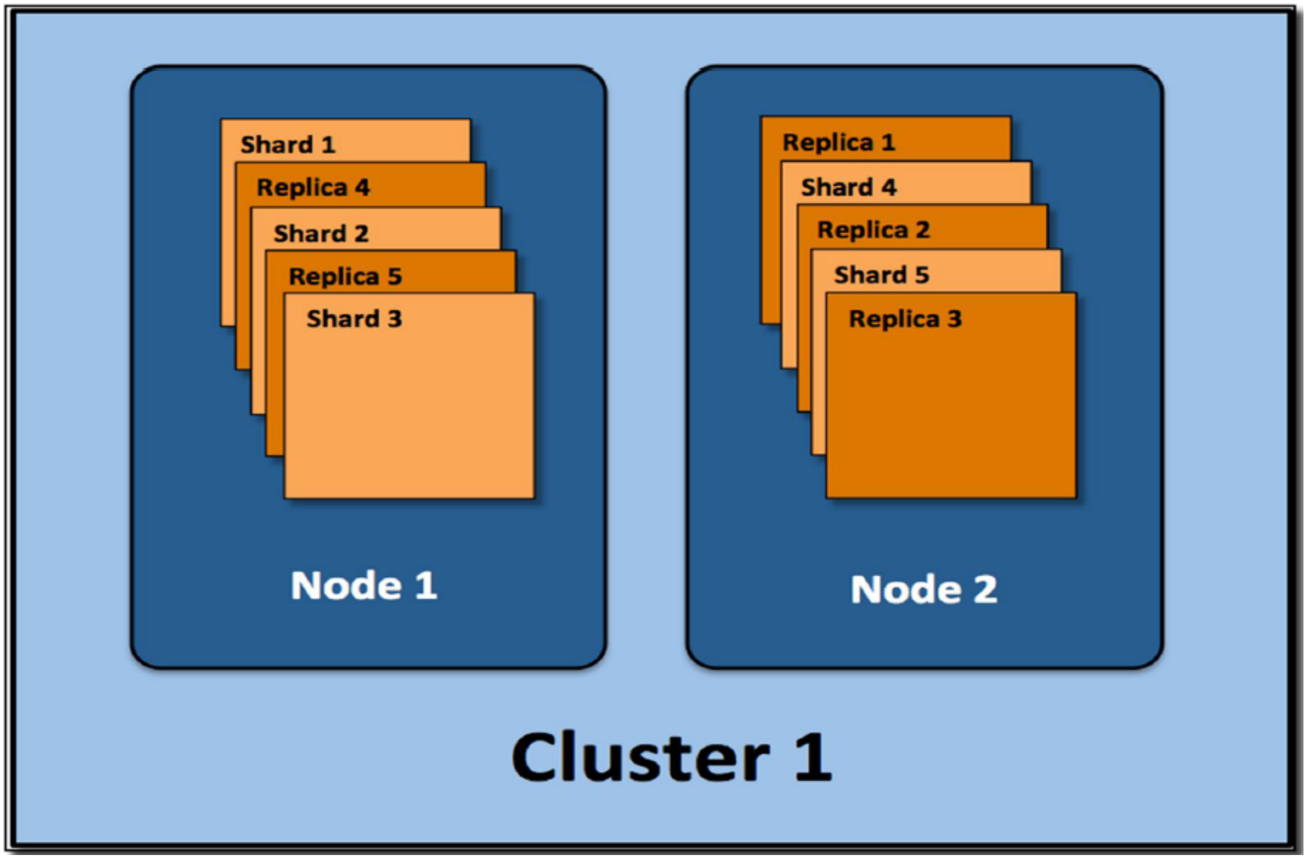
Elasticsearch

Elasticsearch is a distributed, highly scalable open source search engine based on Apache Lucene that allows users to save large data volumes and perform extremely fast searches. Communication takes place via a RESTful web interface.

There is a great amount of reference material available online, including basic documentation and methods. This documentation only addresses the details relevant for use in the ELO system environment. For more information, we recommend the following starting point for further research: https://www.elastic.co/guide/en/elasticsearch/reference/current/index.html.

Lucene

Elasticsearch uses Apache Lucene as its *core library*. With Elasticsearch, each index can be divided into several parts (referred to as *shards*). One shard corresponds to one Lucene index, which consists of one folder containing the associated index files.



The shards can be distributed to multiple servers (*nodes*). You can also create *replicas*. These form a *cluster*.

Language support

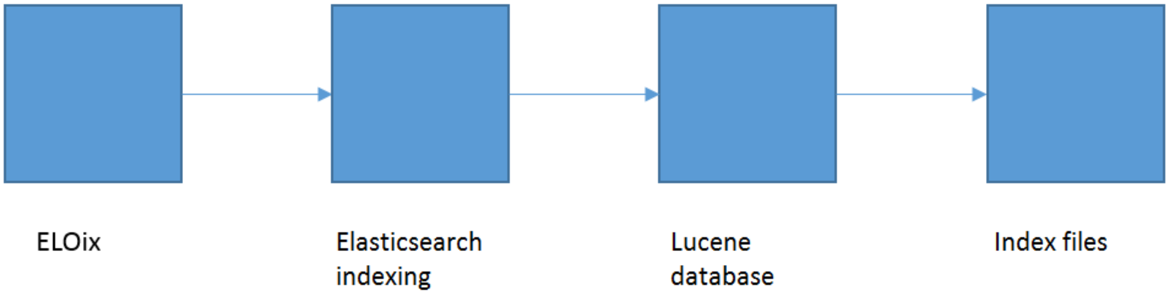
The Elasticsearch uses its own language analyzer during indexing and the search process. This optimizes language-specific search rules and patterns. For ELO specifically, please note that the repository language (selected during installation) has a major influence on the search.

The following languages are supported:

German, English, French, Spanish, Italian, Portuguese, Danish, Swedish, Polish, Dutch, Czech, Hungarian, Romanian, Turkish, Bulgarian, Finnish, Greek, Norwegian, and Russian.

Processing with ELO

The following flowchart illustrates processing with ELO:



The ELO Indexserver transfers the data to Elasticsearch, which uses its own language analyzer during indexing and the search process. The index data is saved in Lucene indices.

Searching indexes requires access to:

- The index files
- A consistence Lucene database
- A functional Elasticsearch language analyzer
- A running ELO Indexserver (ELOix).

Security

Since ELO ECM Suite 10.2, communication between the ELO Indexserver (ELOix) and Elasticsearch is encrypted (SSL). The *Search Guard plug-in* is used for this.

Installation and upgrade

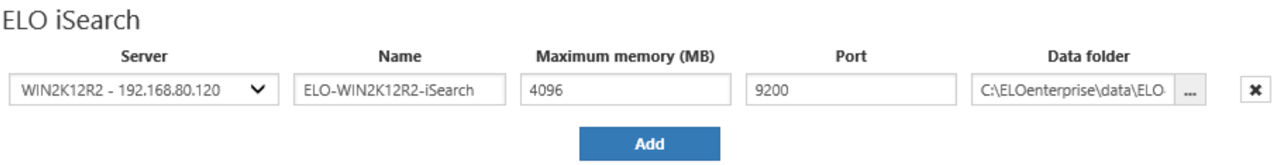
ELO Server Setup

Installations or upgrades are done in the ELO Server Setup, which generates the configuration if it doesn't exist and installs the Elasticsearch program and the service, as well as a data directory where the index files are saved. The following options are available in the ELO Server Setup:

-

Name of service

- Memory value
- Port
- Data directory



ELO iSearch

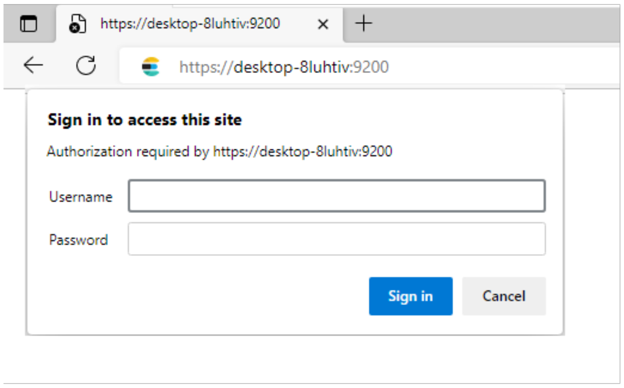
Server	Name	Maximum memory (MB)	Port	Data folder
WIN2K12R2 - 192.168.80.120	ELO-WIN2K12R2-iSearch	4096	9200	C:\ELOenterprise\data\ELO

- With ELOenterprise, additional Elasticsearch servers (see also License terms when distributing server processes).

> **Please note**
>
> As of ELO 21.2, the ELO iSearch port (default: 9200) is secured with TLS/SSL.
>
> You may get a certificate warning the first time you open it in a browser.
>
> Depending on the browser and operating system, you must import the certificate into the certificate store of the browser or operating system. For example, if you are using Microsoft Edge or Google Chrome on Microsoft Windows, load the *keystore.jks* file from `<ELO>\config\elastic\<ELO iSearch server>\certificates` into the Windows certificate store. If you are using Mozilla Firefox, add the certificate to the Firefox certificate store by ignoring the certificate warning message.

Checks after installation

After installation, you can call `https://<server>:<port>` (default port: 9200) with an administrative Tomcat account to perform some checks. If you use any browser, a JSON file is output, which you can open in an editor.

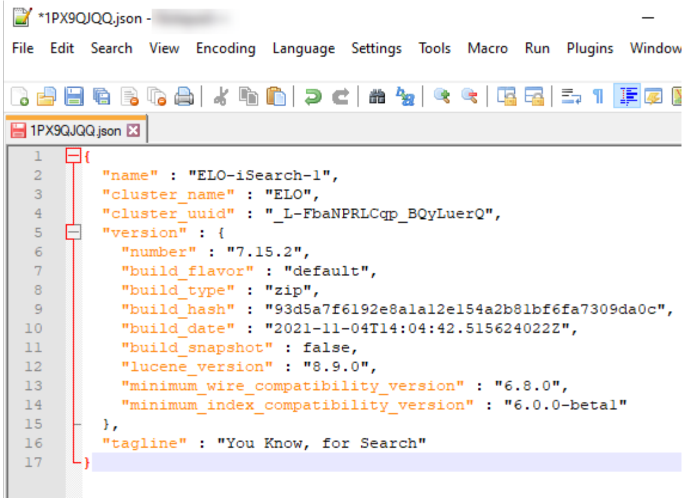


Sign in to access this site

Authorization required by https://desktop-8luhtiv:9200

Username

Password

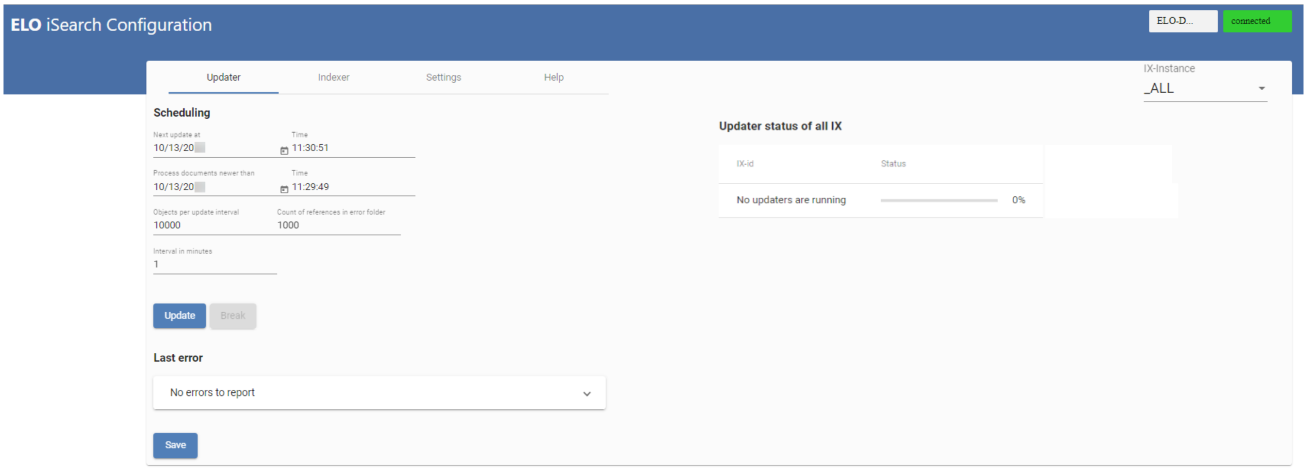


```
{
  "name" : "ELO-iSearch-1",
  "cluster_name" : "ELO",
  "cluster_uuid" : "_L-FbaNPRLCqp_BQyLuerQ",
  "version" : {
    "number" : "7.15.2",
    "build_flavor" : "default",
    "build_type" : "zip",
    "build_hash" : "93d5a7f6192e8a1a12e154a2b81bf6fa7309da0c",
    "build_date" : "2021-11-04T14:04:42.515624022Z",
    "build_snapshot" : false,
    "lucene_version" : "8.9.0",
    "minimum_wire_compatibility_version" : "6.8.0",
    "minimum_index_compatibility_version" : "6.0.0-beta1"
  },
  "tagline" : "You Know, for Search"
}
```

This allows you to view important basic data such as the version and name.

You can perform additional checks in the ELO iSearch configuration. You can reach the configuration via the Indexserver configuration page or the following link (log on with an administrative Tomcat account):

```
https://<server>:<ixport>ix-<REPOSITORY>/manager/esconfig/#/iSearchConfig
```



IX-Instance: _ALL

Scheduling

Next update at: 10/13/20 Time: 11:30:51

Process documents newer than: 10/13/20 Time: 11:29:49

Objects per update interval: 10000

Count of references in error folder: 1000

Interval in minutes: 1

Last error: No errors to report

Updater status of all IX

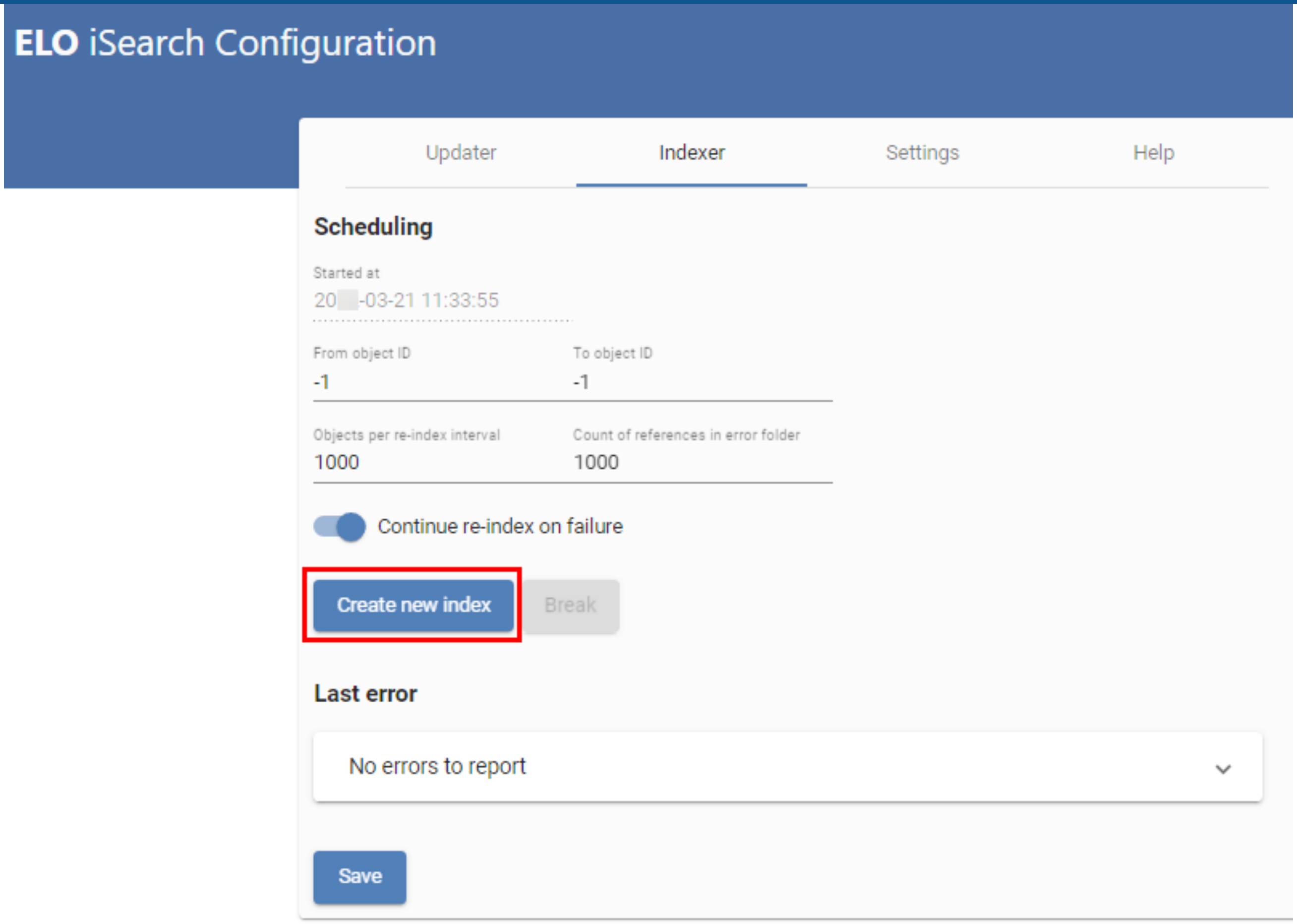
IX-id	Status
No updaters are running	0%

Reindexing after upgrading from older ELO versions

Updating from an earlier ELO version (up to ELO version 23.5) to ELO version 23.6 or higher requires you to reindex the index data. The ELO Server Setup removes the previous Elasticsearch indexes as they are no longer compatible with the current Elasticsearch version.

During reindexing, first the existing index is deleted and then rebuilt. All fields are saved in both *tokenized* and *not tokenized* (as a phrase) forms meaning changes to this field setting do not require a reindex. In addition, the full text content is read out from the *FT*.txt* files generated in the ELO full text process and used for indexing.

You can initiate a reindex on the *Indexer* tab of the ELO iSearch configuration.



ELO iSearch Configuration

Scheduling

Started at
20[redacted]-03-21 11:33:55

From object ID	To object ID
-1	-1

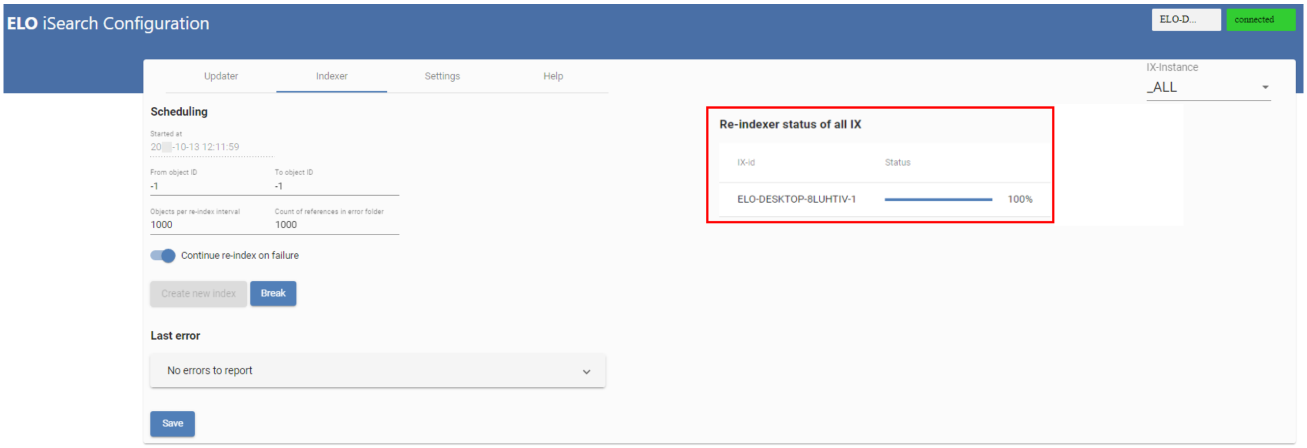
Objects per re-index interval	Count of references in error folder
1000	1000

Continue re-index on failure

Last error

No errors to report

The progress is shown after you refresh the page.



ELO iSearch Configuration

IX-Instance
_ALL

Scheduling

Started at
20[redacted]-10-13 12:11:59

From object ID	To object ID
-1	-1

Objects per re-index interval	Count of references in error folder
1000	1000

Continue re-index on failure

Last error

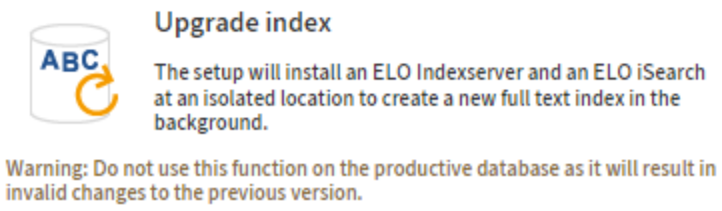
No errors to report

Re-indexer status of all IX

IX-id	Status
ELO-DESKTOP-8LUHTIV-1	100%

Background reindex

For upgrades from ELO version 9.3 and higher, the ELO Server Setup *Upgrade Index* function installs an isolated ELO Indexserver that builds the Elasticsearch index in the background. This makes sense for large systems in particular, as reindexing can take several days. You can use this function when updating to a more current ELO version or when making changes to the iSearch settings.



Upgrade index

The setup will install an ELO Indexserver and an ELO iSearch at an isolated location to create a new full text index in the background.

Warning: Do not use this function on the productive database as it will result in invalid changes to the previous version.

You can import this index when upgrading ELO so that the full scope of search functions is available immediately after the upgrade.

Refer to the Background reindex section for details of this process.

Sizing and performance

A performant search is crucial for users to accept the ELO system. For this reason, the Elasticsearch must be optimized regularly (e.g. every six months, depending on the additional document and data volume) along with database performance. You should check whether this is necessary at the user level (by performing complex searches) but also at the server level (by occasionally analyzing the log file, see the section Log file).

General information on sizing

Elasticsearch performance depends on the following factors:

- RAM both for the Elasticsearch JVM (Java Virtual Machine), additional Elasticsearch memory for caches, which is managed by the operating system, and for the operating system in general
- Number of processors/cores
- Memory technology/memory access times

The following general statements on sizing apply:

- RAM (see also the section RAM sizing):
  - For small environments, the standard 4 GB + 4 GB of additional RAM suffice.
  - You should allow approx. 24 GB (12 GB + 12 GB) for medium-sized environments (approx. 10 million documents).
  - As there is a limit of around 30 GB per JVM for large environments, server clustering is recommended.
- Processors/cores:
  - Each search requires a thread, i.e. a higher number of processors/cores achieves improved parallelization.
- Hard drives:
  - Fast hard drives (SSD) significantly increase performance. We urgently recommend using SSD storage to run Elasticsearch.

Even for medium-sized environments, we recommend using ELOenterprise (see also License terms when distributing server processes).

RAM sizing

With Elasticsearch, memory usage is not only related to the number and size of the documents. How the user works also plays a major role.

Generally, when it comes to memory usage, it is necessary to distinguish between the Java heap for the ELO iSearch service and the additionally required RAM for Elasticsearch. There is a clear recommendation to keep at least as much free memory available as is configured for the Java heap.

You configure a fixed Java heap in the ELO iSearch service, for example 10 GB, or initially 4 GB for a new ELO installation. There should then be as much free memory that isn't utilized by other applications or the operating system itself available to the operating system as a file cache system for Elasticsearch. This memory is used by the operating system as a file system cache and therefore by Elasticsearch. The main contents it contains are index data (fixed and temporary) and some types of caches for search queries.

By contrast, the Java heap contains the Elastic application itself, including a basic amount of memory per running shard (Lucene index) as well as the query cache and the field data cache.

For the initial size of the Java heap, you could use the following guideline value, which has to be adjusted later on depending on actual usage: basic amount 4 GB, plus as much as is currently needed (1 GB for 1 million ELO documents is a good starting value). Everything else is much more valuable as a file system cache. As an empirical value, the correct configuration of the Java heap is 8 to 16 GB for approx. 10 million documents. The exact value depends on actual usage and the type of index fields. If users in the ELO client frequently use filters for index fields that are marked as *tokenized* and context terms (i.e. filter values that can be selected for this field) can be shown and the index for this metadata form contains millions of documents, all potentially available field values are loaded to the Java heap and end up in the field data cache. This can cause the size of the cache and thus the required Java heap to grow significantly. This is also the case when sorting by this kind of field. If only aggregating to *notTokenized* fields or no filter suggestions are requested by users, this doesn't have an impact on the field data cache. Instead, the field contents land in temporary virtual files, which should be able to be found in the operating system file system cache mentioned above. This makes it difficult to predict how high the Java heap should be configured.

> **Please note**
>
> A separate Elasticsearch index is built for each metadata form. This requires a basic Java heap of approx. 50 MB (for a 1-node system without replicas). A large number of metadata forms therefore requires a large Java heap and thus a large amount of RAM for the ELO iSearch.
>
> The recommendation is therefore: When configuring metadata forms, the number of forms used should not reach double-digits whenever possible. This applies for first generation and second generation metadata forms.

For the overall sizing of the Java heap and the remaining RAM as a file system cache, we recommend first determining the memory usage of all other applications on a system that are still running on the same computer or the VM, including the operating system. For example, an ELO Indexserver, ELO Textreader, and ELO OCR instance are still running there whose consumption isn't determined solely based on the configured memory, but that also requires a large file system cache. You can find the total amount for each application in your operating system's Task Manager application. Add the operating system usage to this. The total system RAM is made up of this value, the Java heap configured for Elasticsearch, and at least as much free memory as for the Elastic file system cache.

Example:

For example, the ELOix/ELOtr/ELOocr/operating system require 12 GB. ELO iSearch is configured for an 8 GB Java heap and requires another 8 GB for the file system cache on top of that. The total amount is therefore 12 GB + 8 GB + 8 GB = 28 GB. This would leave room for 4 GB in a 32 GB system. This isn't very much when the repository is growing.

In larger systems, we recommend running ELO iSearch on a separate VM or computer to prevent it from competing with other applications over the file system cache and forcing the operating system to outsource.

As a general rule: only as much Java heap as is needed, but as much file system cache as possible.

> **Information**
>
> We recommend setting the Java heap for ELO iSearch in the ELO Server Setup.

Optimizing the indexing process

You can only optimize the indexing process to a certain extent. The number of documents indexed at the same time is determined by the ELO Indexserver (ELOix). Should you have to reindex large volumes of documents, we recommend providing a separate Indexserver for this task (see also Background reindex).

Optimizing search performance

Performance factors were already covered in the section General information on sizing. In large environments, increasing the number of shards per index can have a positive effect on the search speed if they are distributed to multiple nodes. Please note that changing the number of shards requires reindexing. Refer to the section Setting the number of shards per index for more information on configuring the number of shards.

Duration of a reindex

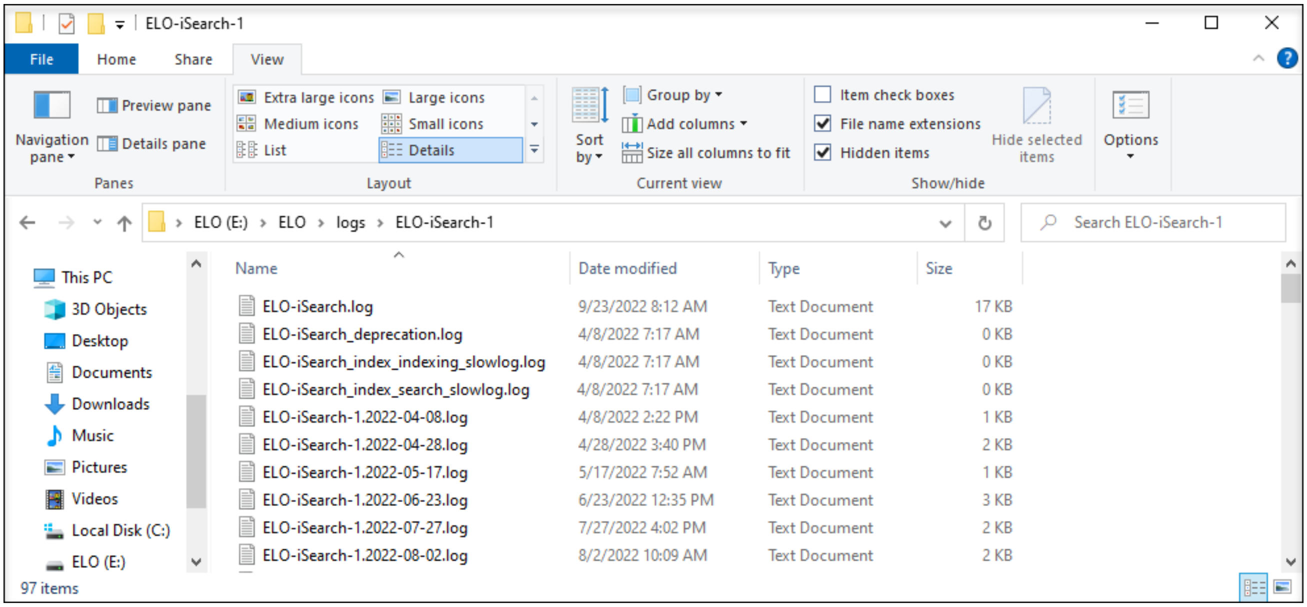
Based on experience, a reindex for 1 million documents will take around 20 hours in a low-performance environment that has not been tuned, or around 5 hours on performant hardware.

Operation

There are two important aspects to consider when operating Elasticsearch:

- Log files and their content
- Creating evaluations

Log file



☐ Item check boxes
☑ File name extensions
☑ Hidden items

Name	Date modified	Type	Size
ELO-iSearch.log	9/23/2022 8:12 AM	Text Document	17 KB
ELO-iSearch_deprecation.log	4/8/2022 7:17 AM	Text Document	0 KB
ELO-iSearch_index_indexing_slowlog.log	4/8/2022 7:17 AM	Text Document	0 KB
ELO-iSearch_index_search_slowlog.log	4/8/2022 7:17 AM	Text Document	0 KB
ELO-iSearch-1.2022-04-08.log	4/8/2022 2:22 PM	Text Document	1 KB
ELO-iSearch-1.2022-04-28.log	4/28/2022 3:40 PM	Text Document	2 KB
ELO-iSearch-1.2022-05-17.log	5/17/2022 7:52 AM	Text Document	1 KB
ELO-iSearch-1.2022-06-23.log	6/23/2022 12:35 PM	Text Document	3 KB
ELO-iSearch-1.2022-07-27.log	7/27/2022 4:02 PM	Text Document	2 KB
ELO-iSearch-1.2022-08-02.log	8/2/2022 10:09 AM	Text Document	2 KB

97 items

By default, the Elasticsearch log files are stored in the ELO installation directory within the *logs* folder.

Here, the *ELO.log* file is relevant (or `ELO-<date>.log` for older log files).

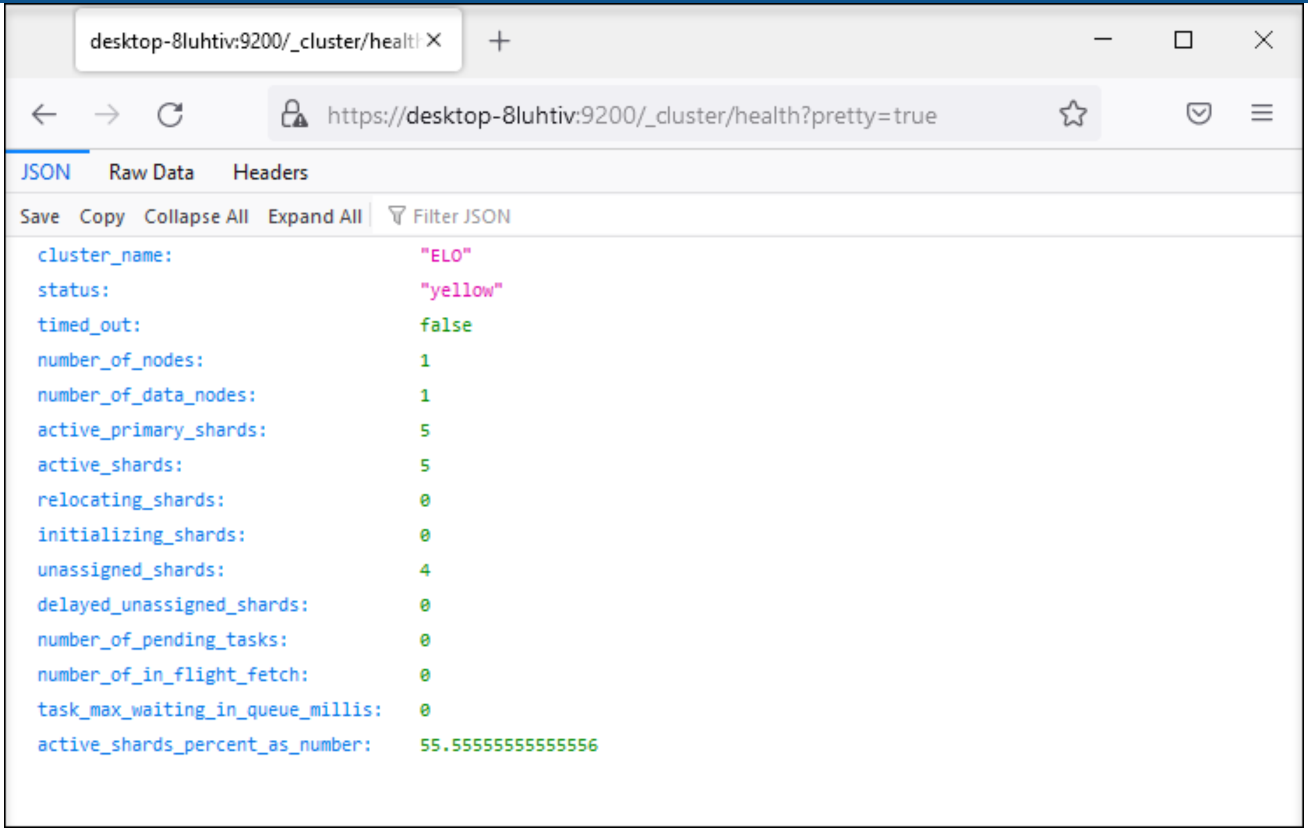
In individual cases, it may make sense to look at the Indexserver log file (`ix-<repository>.log`). For ELO 11 and ELO 12, you can search for the term `queryTerm` in the log file, for example. The following example shows the output of a full text search for the term `test`:

```
22:51:34,723 eloix-find-40 eloix-find-40 INFO (ElasticClient.java:183) - find(searchId=[(09A4500
```

Evaluations

You can query various Elasticsearch statuses via the browser. You can then use these for monitoring solutions (e.g. Nagios) if desired. You require an administrative Tomcat account to log on. A query outputs a JSON file like the one shown in the following screenshot, which you can show in Mozilla Firefox.

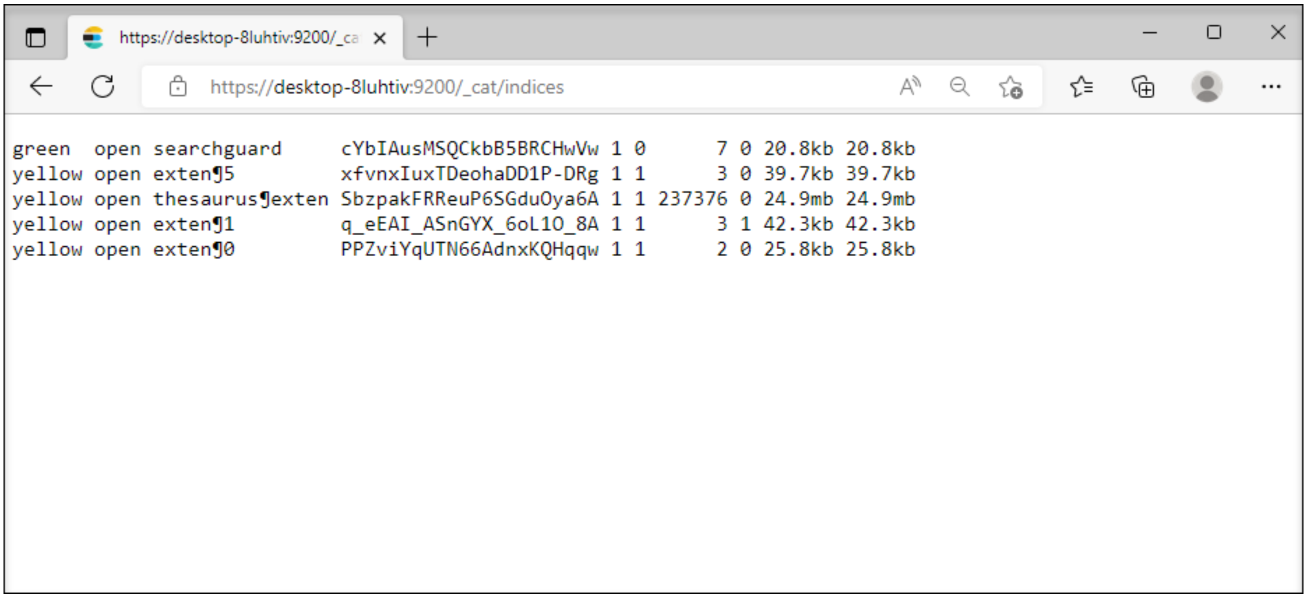
The URL for the *health check* is: `https://<server>:9200/_cluster/health?pretty=true`


desktop-8luhtiv:9200/_cluster/healt
https://desktop-8luhtiv:9200/_cluster/health?pretty=true
JSON
Raw Data
Headers
Save
Copy
Collapse All
Expand All
Filter JSON

```
cluster_name:                     "ELO"
status:                           "yellow"
timed_out:                        false
number_of_nodes:                  1
number_of_data_nodes:             1
active_primary_shards:            5
active_shards:                    5
relocating_shards:                0
initializing_shards:              0
unassigned_shards:                4
delayed_unassigned_shards:        0
number_of_pending_tasks:          0
number_of_in_flight_fetch:        0
task_max_waiting_in_queue_millis: 0
active_shards_percent_as_number:  55.55555555555556
```

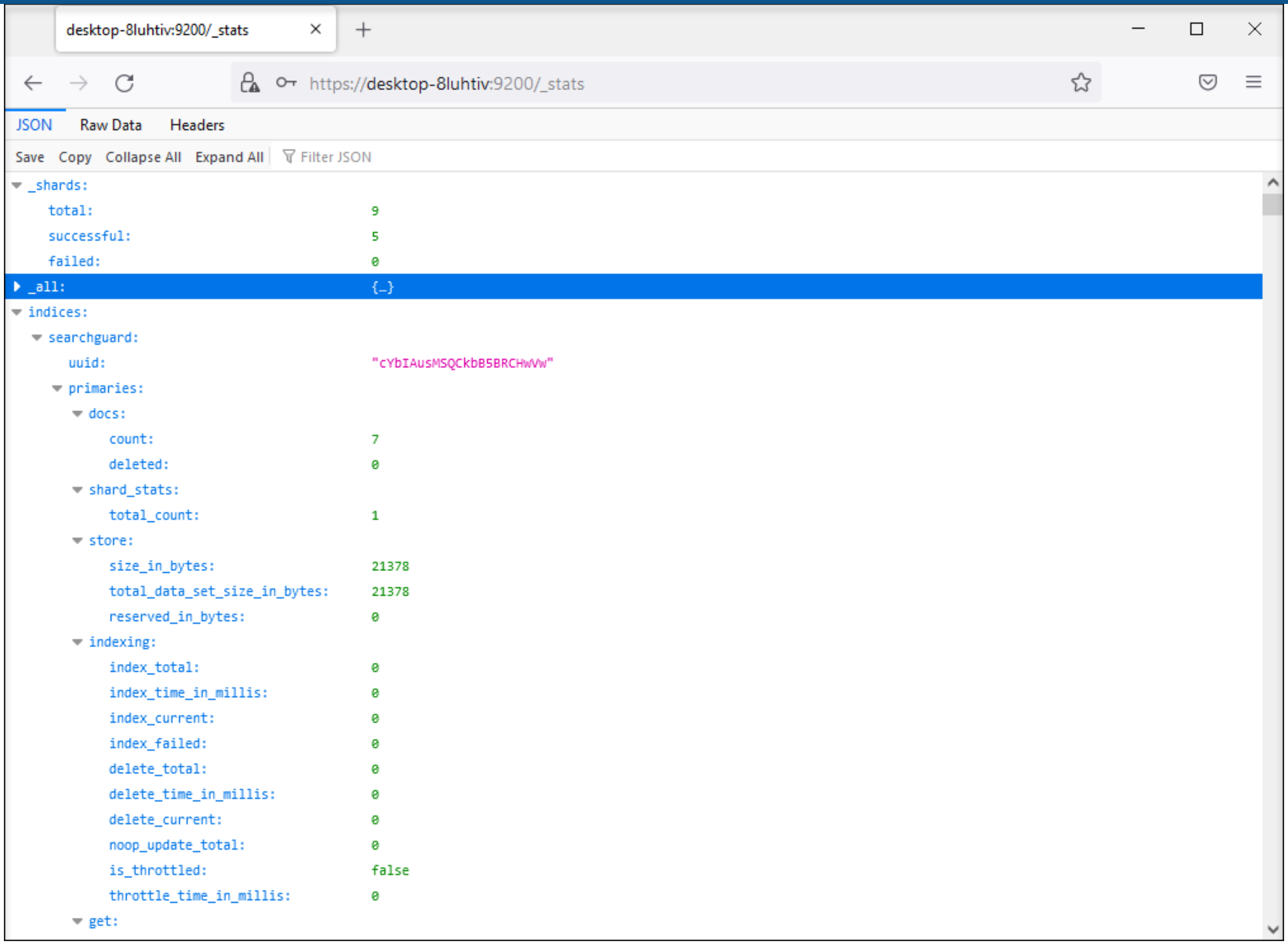
The status `yellow` means that the index is functional but the replica shards are not all available. In a single server environment using (1) as the default setting for replicas, the `yellow` status is normal.

The URL for the index directory name is: `https://<server>:<port>/_cat/indices`


https://desktop-8luhtiv:9200/_ca
https://desktop-8luhtiv:9200/_cat/indices

```
green  open searchguard      cYbIAusMSQCkbB5BRCHwVw 1 0      7 0 20.8kb 20.8kb
yellow open exten¶5          xfvnxIuxTDeohaDD1P-DRg 1 1      3 0 39.7kb 39.7kb
yellow open thesaurus¶exten SbzpakFRReuP6SGduOya6A 1 1 237376 0 24.9mb 24.9mb
yellow open exten¶1          q_eEAI_ASnGYX_6oL1O_8A 1 1      3 1 42.3kb 42.3kb
yellow open exten¶0          PPZviYqUTN66AdnxKQHqqw 1 1      2 0 25.8kb 25.8kb
```

One index exists for each metadata form, and can be recognized based on its ID.


desktop-8luhtiv:9200/_stats
https://desktop-8luhtiv:9200/_stats
JSON
Raw Data
Headers
Save
Copy
Collapse All
Expand All
Filter JSON

```
▼ _shards:
    total:                          9
    successful:                     5
    failed:                         0
▶ _all:                             {…}
▼ indices:
  ▼ searchguard:
      uuid:                         "cYbIAusMSQCkbB5BRCHwVw"
    ▼ primaries:
      ▼ docs:
          count:                    7
          deleted:                  0
      ▼ shard_stats:
          total_count:              1
      ▼ store:
          size_in_bytes:            21378
          total_data_set_size_in_bytes: 21378
          reserved_in_bytes:        0
      ▼ indexing:
          index_total:              0
          index_time_in_millis:     0
          index_current:            0
          index_failed:             0
          delete_total:             0
          delete_time_in_millis:    0
          delete_current:           0
          noop_update_total:        0
          is_throttled:             false
          throttle_time_in_millis:  0
      ▼ get:
```

The URL for statistical data is: `https://\<server\>:9200/_stats`.

This allows you to check whether a new document has been added to the index. After a new document has been added, the value in the first line of the query after `count` is increased by 1.

Additional queries are possible in the same fashion. These are described in the Elasticsearch online documentation.

Backup options and strategies

The iSearch data can be restored fully from the database via the Indexserver. The decision whether to create a backup of the Elasticsearch index depends on the following factors:

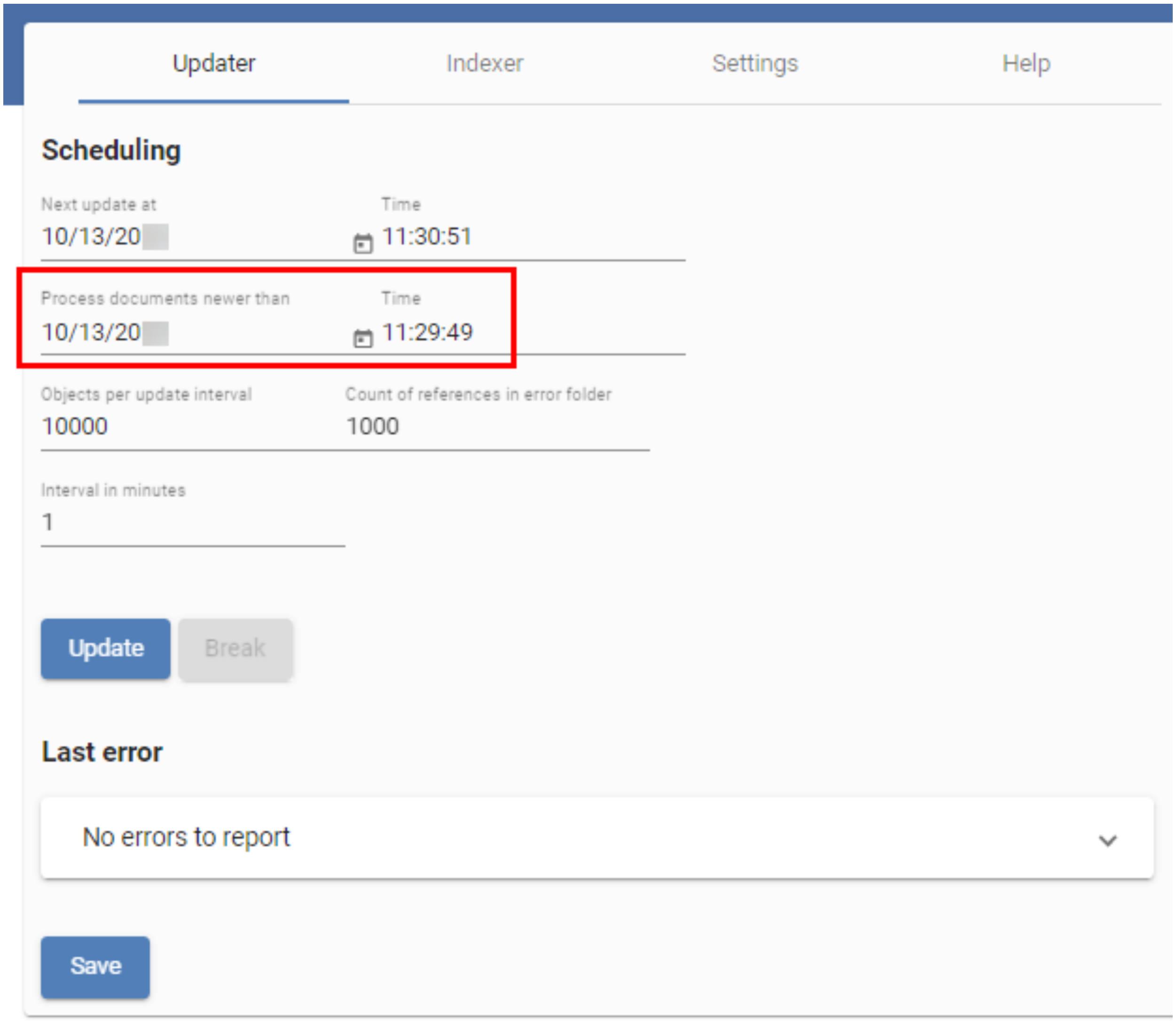
- How much downtime is acceptable for users?

Keeping downtime to a minimum requires a backup strategy that allows a backed up iSearch index to quickly be imported to a live system.

- How long will it take to restore the iSearch index?

The duration of the initial reindex can be used as an indicator of how long subsequent reindexes will take. You have to decide whether this downtime period is within reason for users.

There are two options for creating a backup. For both solutions, the indices between the different Indexserver versions may not necessarily be compatible. After importing the backup, you will have to reindex the documents changed since the backup was created. This can be done using the Updater.



Scheduling

Next update at: 10/13/20[redacted] Time: 11:30:51

Process documents newer than: 10/13/20[redacted] Time: 11:29:49

Objects per update interval: 10000

Count of references in error folder: 1000

Interval in minutes: 1

Last error

No errors to report

The value for *Process documents newer than* is set to the time the backup was created. It makes sense to create a backup once a day or week to keep the process for indexing changed documents to a minimum.

Please note that the index named *searchguard* is necessary to ensure SSL encryption by means of the *Search Guard plug-in*. If this index does not exist, you the Indexserver will not be able to connect to Elasticsearch. If you have accidentally deleted the *searchguard* index, you can restore it by running the server setup again. The *searchguard index* also depends on the selected administrative Tomcat account, meaning you cannot just copy the index if the account or passwords differ, and the index must be created by the server setup.

Copying an existing index

You can copy the existing index from the file system and re-import it at a later point in time. Copy the entire folder located at *<installation directory>/data/<...><server name>/index* as a backup. If the backup is required, replace the content of the folder named above with the backup.

Before copying, you should stop the iSearch service to prevent access errors.

Creating snapshots with Elasticsearch

Elasticsearch offers the option to create snapshots of existing indexes. If snapshots are generated regularly, a complete copy is not created every time. Instead, only the changed files are saved to reduce the amount of memory required.

You will find a guide here:

https://www.elastic.co/guide/en/elasticsearch/reference/7.15/snapshot-restore.html

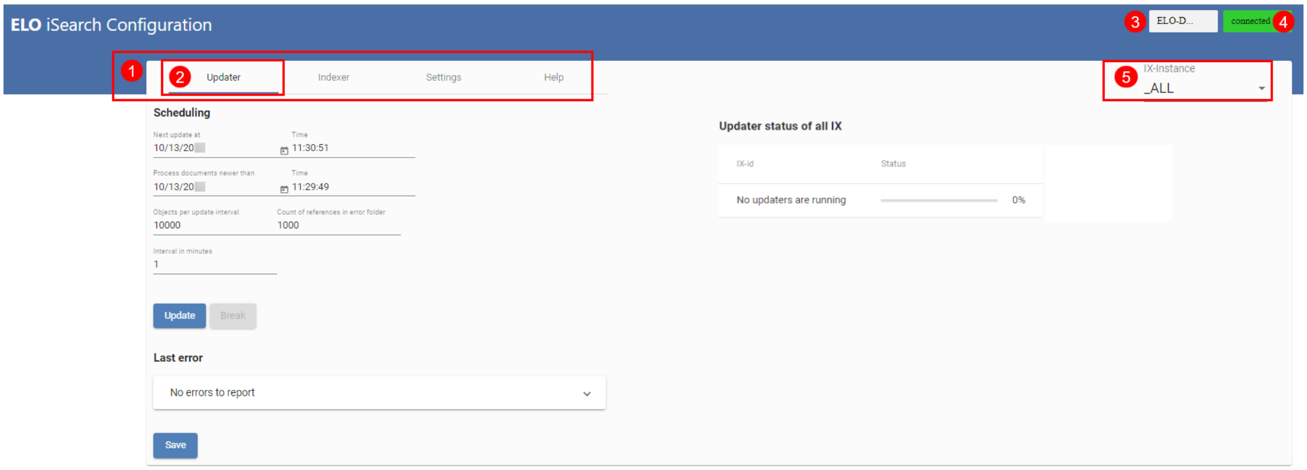
> **Please note**
>
> To recover a snapshot, the following line has to be inserted in the Elasticsearch configuration file (`<installation directory>/config/elastic/<server name>/elasticsearch.yml`): `searchguard.enable_snapshot_restore_privilege: true`

Tips and tricks

ELO iSearch configuration

You can open the *ELO iSearch Configuration* page, available starting with ELO 11, via the Indexserver configuration page or at the following link:

```
https://<server>:<Port>:/ix-<Repository>/manager/esconfig/#/isearchConfig
```



IX-Instance: _ALL

Scheduling

Next update at: 10/13/20[redacted] Time: 11:30:51

Process documents newer than: 10/13/20[redacted] Time: 11:29:49

Objects per update interval: 10000

Count of references in error folder: 1000

Interval in minutes: 1

Last error

No errors to report

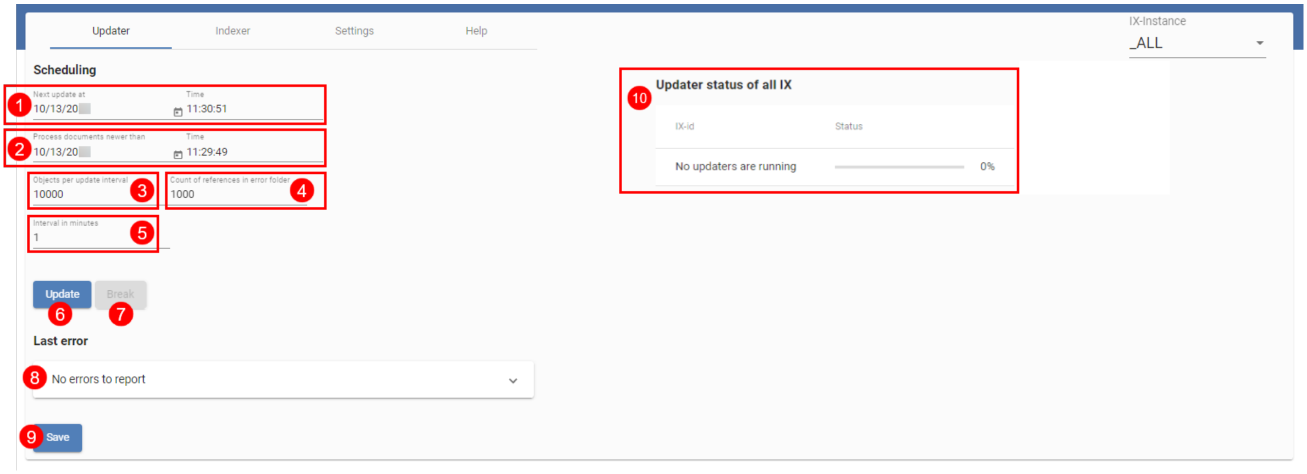
Updater status of all IX

IX-id	Status	
No updaters are running		0%

The above figure shows the ELO iSearch configuration. In the ELO iSearch configuration, you can access the different functions of the iSearch (1); the active tab is underlined in blue (2). At the top right (3) you will see the name of the ELOix instance calling the ELO iSearch configuration, with the connection status to ELO iSearch (4) either *connected* or *disconnected*.

Since ELO 11, the settings for ELO iSearch are saved in the eloixopt table specific to the instance or for all instances (*_ALL*), as with the Indexserver options. In the table, all the ELO iSearch settings have the prefix *esearch*. In the ELO iSearch configuration, you can view and edit the settings for all instances (*_ALL*) or specific instances (5).

Updater



IX-Instance: _ALL

Scheduling

Next update at: 10/13/20[redacted] Time: 11:30:51

Process documents newer than: 10/13/20[redacted] Time: 11:29:49

Objects per update interval: 10000

Count of references in error folder: 1000

Interval in minutes: 1

Last error

No errors to report

Updater status of all IX

IX-id	Status
No updaters are running	0%

The above figure shows an overview of the Updater. This is required to regularly update ELO iSearch. It also ensures new and changed objects are sent to ELO iSearch. In the top line (1), you will see when the Updater is scheduled to start again, and the next line (2) indicates from which date on documents are processed. You can set both values. In the default configuration, the Updater runs every minute (5) and processes a maximum of 1000 objects (3) in the smallest time unit (1 minute). Please note that the number of selected objects is cached in the ELOix memory, so you should only change this value with caution.

If there are issues indexing a document, a reference to the document is created at the following path in ELO:

```
Administration//Fulltext Configuration//Update Errors
```

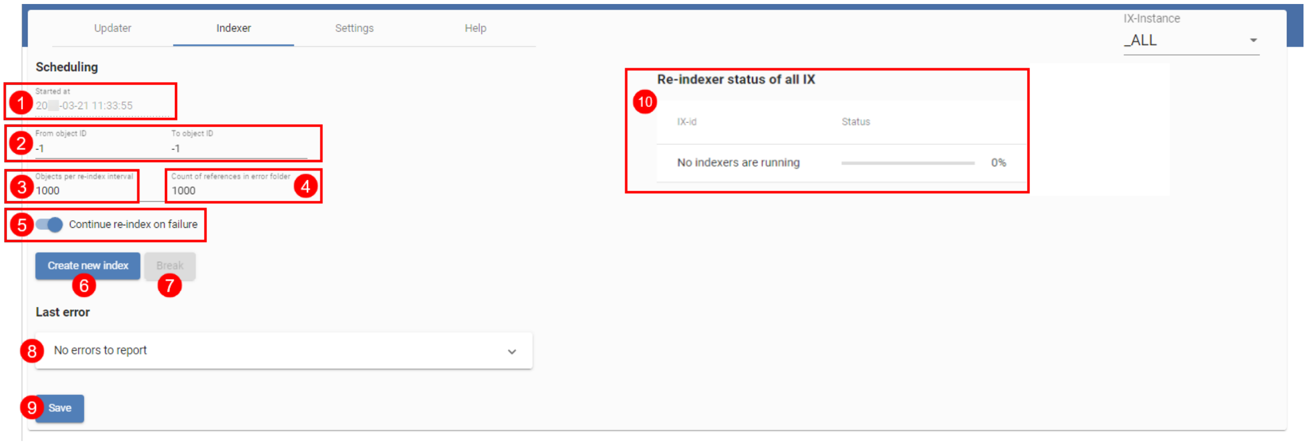
This allows you to easily monitor errors and make sure that documents are sent to ELO iSearch again, for example by opening the *Metadata* dialog box. You can set the maximum number of documents in this folder in (4). If you set `0`, no references will be created.

Save any changes with the *Save* button (9).

Updater status of all IX (10) provides an overview of all running update processes. You can also manually start the update process by clicking *Update* (6) or pause by selecting *Break* (7). Note that starting and stopping always applies to the current ELOix instance.

The most recent error in the update process is shown in *Last Error* (8).

Indexer



IX-Instance: _ALL

Scheduling

Started at: 20[redacted]-03-21 11:33:55

From object ID: -1

To object ID: -1

Objects per re-index interval: 1000

Count of references in error folder: 1000

Continue re-index on failure

Last error

No errors to report

Re-indexer status of all IX

IX-id	Status	
No indexers are running		0%

The Indexer (image above) initially provides ELO iSearch with objects from ELO. All existing objects are sent to ELO iSearch, beginning with objects with the highest ID, indexed in descending order. As a result, new objects can be found first, provided the index is used in the productive environment right away.

The top line (1) indicates the start time of the last reindex process. In (2) and (3), you can choose the object IDs to be indexed. If you would like to perform a complete reindex, set both values to `-1`. The existing index is deleted as a result. If you would like to add only specific objects to an existing index, you can do this by means of IDs.

The settings for the objects per index interval (3) and references in ELO (4) correspond to those of the Updater. The references are filed to the folder at the following path:

```
Administration//Fulltext Configuration//Reindex Errors
```

This value is the same for both the Updater and the Indexer.

The *Continue Reindex on Failure* setting (5) indicates whether to cancel or continue reindexing in case of an error. If you would like to continue, you should regularly check the log outputs.

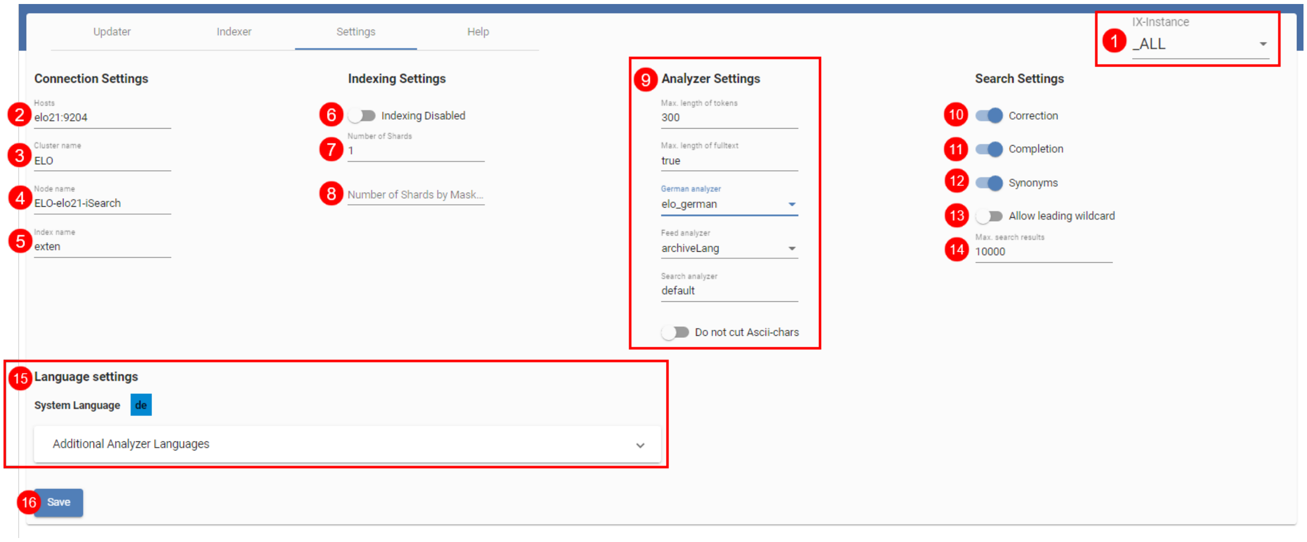
The most recent error in the reindex process is shown in *Last Error* (8).

Select *Save* (9) to save the settings.

You can start a reindex (if both object IDs are set to `-1`), or continue (6) or cancel it (7). As with the Updater, remember that this always applies to the ELOix instance the configuration page was called with.

(10) provides an overview of all active reindex processes.

Settings



IX-Instance (1): _ALL

Connection Settings

Hosts (2): elo21:9204
Cluster name (3): ELO
Node name (4): ELO-elo21-iSearch
Index name (5): exten

Indexing Settings

(6) Indexing Disabled
Number of Shards (7): 1
(8) Number of Shards by Mask...

Analyzer Settings (9)

Max. length of tokens: 300
Max. length of fulltext: true
German analyzer: elo_german
Feed analyzer: archiveLang
Search analyzer: default
Do not cut Ascii-chars

Search Settings

(10) Correction
(11) Completion
(12) Synonyms
(13) Allow leading wildcard
Max. search results (14): 10000

Language settings (15)

System Language: de
Additional Analyzer Languages

(16) Save

The above image contains an overview of additional setting options for ELO iSearch. In *IX instance* (1), select whether the settings should be applied to all instances or to a specific instance (see also the next figure *ELO iSearch configuration, 'Setting' tab, instance-specific settings*).

Details on the connection between ELOix and Elasticsearch are specified in (2), (3), and (4). The name of the Elasticsearch index (5) corresponds to the name of the repository in lower case letters by default.

You can also specify whether an ELOix should be used to send objects to ELO iSearch at all (6). If this setting is enabled, the full text search via ELO iSearch is still enabled.

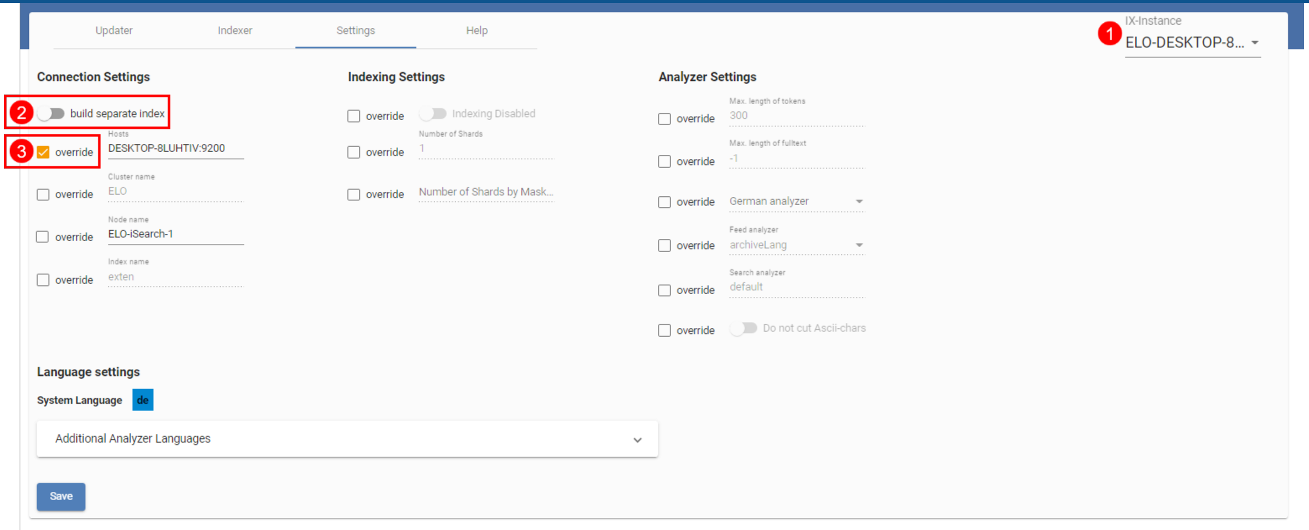
The number of shards and shards per form is set under (7) and (8). Th chapter Setting the number of shards per index describes these settings in more detail.

The Analyzer settings (9) determine how text sent to Elastic search is processed. Normally, you can use the default settings.

Settings (10) to (14) apply to the search. You can set whether to return suggestions for corrections (10) or synonyms (12) for the search query and whether autocomplete (11) is active. Furthermore, you can enable searches with a wildcard character at the beginning of the term (13). This is disabled by default, as searches with a leading wildcard can cause high load on the server. You can also set the maximum search results (14) returned via the `findFirstSords()` and `findNextSords()` ELOix API calls.

You can edit the ELO iSearch language settings under *Language Settings* (15). The system or repository language is set by the server setup and cannot be changed. You can select additional languages depending on the available licenses.

Select *Save* (16) to save all settings. Please note that the settings (5), (7), (8), (9), and (15) require a reindex to apply to all data and not cause inconsistent data in ELO iSearch.



IX-Instance: ELO-DESKTOP-8...

Connection Settings

- build separate index
- ☑ override — Hosts: DESKTOP-8LUHTIV:9200
- □ override — Cluster name: ELO
- □ override — Node name: ELO-iSearch-1
- □ override — Index name: exten

Indexing Settings

- □ override — Indexing Disabled
- □ override — Number of Shards: 1
- □ override — Number of Shards by Mask...

Analyzer Settings

- □ override — Max. length of tokens: 300
- □ override — Max. length of fulltext: -1
- □ override — German analyzer
- □ override — Feed analyzer: archiveLang
- □ override — Search analyzer: default
- □ override — Do not cut Ascii-chars

Language settings

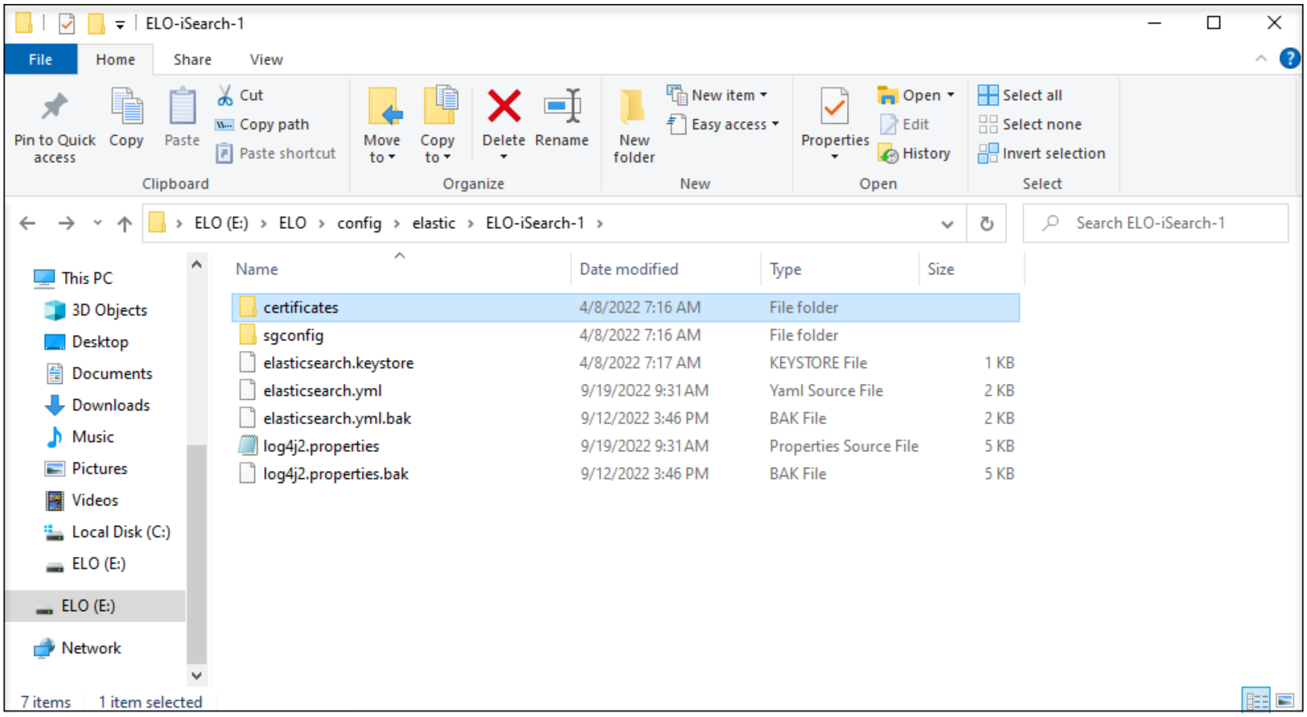
System Language: de

Additional Analyzer Languages

The figure shows the setting page for the instance ELO-DESKTOP-8LUHTIV (1). All settings where the override check box is enabled (3) overwrite the settings applied for all instances.

You can only apply the *build separate index* (2) setting for specific instances. If this setting is enabled, a separate index is built with this Indexserver if either the connection settings or the index name differs from those of the other instances. This is explained in more detail in the chapter Background reindex.

Elasticsearch service configuration



ELO (E:) › ELO › config › elastic › ELO-iSearch-1 ›

Name	Date modified	Type	Size
certificates	4/8/2022 7:16 AM	File folder	
sgconfig	4/8/2022 7:16 AM	File folder	
elasticsearch.keystore	4/8/2022 7:17 AM	KEYSTORE File	1 KB
elasticsearch.yml	9/19/2022 9:31 AM	Yaml Source File	2 KB
elasticsearch.yml.bak	9/12/2022 3:46 PM	BAK File	2 KB
log4j2.properties	9/19/2022 9:31 AM	Properties Source File	5 KB
log4j2.properties.bak	9/12/2022 3:46 PM	BAK File	5 KB

The Elasticsearch service is configured via multiple files and directories.

In detail, the files and directories contain the following:

-

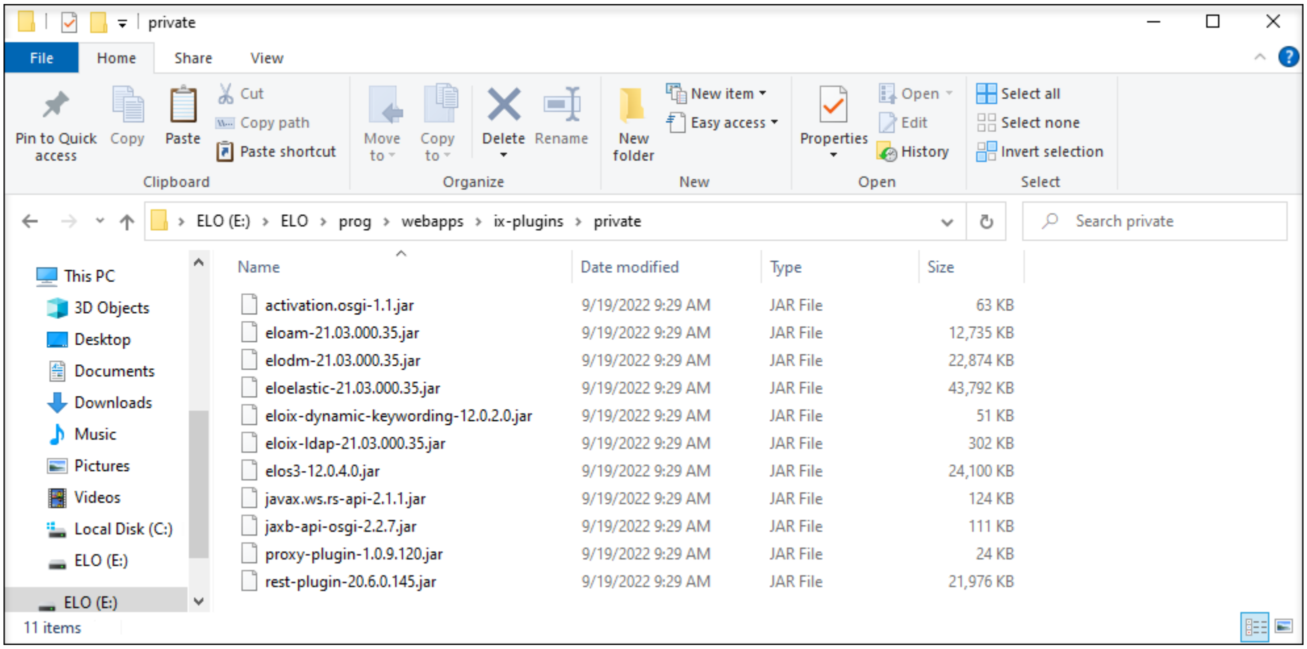
elasticsearch.yml: Central configuration file for Elasticsearch. Refer to the following website for more detailed information: https://www.elastic.co/guide/en/elasticsearch/reference/current/important-settings.html

- log4j.properties: logging settings
- sgconfig: directory containing the *Search Guard plug-in* configuration
- certificates: keystore and truststore for SSL communication with ELO Indexserver

SSL communication

Starting with ELO ECM Suite 10.2, communication between ELO Indexserver instances and Elasticsearch is encrypted. The ELO Server Setup creates *self-signed certificates* as well as corresponding keystores and truststores during installation. The keystore and truststore are created within the respective config directories (Indexserver and Elasticsearch) and can be opened with the standard Java tools (e.g. *keytool.exe*) to check the certificates. Enter the password for the administrative Tomcat account.

Indexserver plug-ins



Name	Date modified	Type	Size
activation.osgi-1.1.jar	9/19/2022 9:29 AM	JAR File	63 KB
eloam-21.03.000.35.jar	9/19/2022 9:29 AM	JAR File	12,735 KB
elodm-21.03.000.35.jar	9/19/2022 9:29 AM	JAR File	22,874 KB
eloelastic-21.03.000.35.jar	9/19/2022 9:29 AM	JAR File	43,792 KB
eloix-dynamic-keywording-12.0.2.0.jar	9/19/2022 9:29 AM	JAR File	51 KB
eloix-ldap-21.03.000.35.jar	9/19/2022 9:29 AM	JAR File	302 KB
elos3-12.0.4.0.jar	9/19/2022 9:29 AM	JAR File	24,100 KB
javax.ws.rs-api-2.1.1.jar	9/19/2022 9:29 AM	JAR File	124 KB
jaxb-api-osgi-2.2.7.jar	9/19/2022 9:29 AM	JAR File	111 KB
proxy-plugin-1.0.9.120.jar	9/19/2022 9:29 AM	JAR File	24 KB
rest-plugin-20.6.0.145.jar	9/19/2022 9:29 AM	JAR File	21,976 KB

Since ELO 11, ELO Indexserver can be extended using OSGi plug-ins. For example, the Elasticsearch interface is located in this type of plug-in. In the future, this distribution will enable access to a newer Elasticsearch index with an older version of the ELO Indexserver. One plug-in directory is configured for each Indexserver.

Setting the number of shards per index

Starting with ELO 11, one index is created per metadata form. By default, each index consists of one shard. Creating one index per metadata form alone boosts cluster performance, as the indices are distributed to several nodes.

It also makes sense to split large indices into several shards. The advantage of this is that search queries are run in parallel on all shards. However, the number of search queries run in parallel within a node does not exceed the number of available processors.

As a result, the number of shards is based on the data contained. Elasticsearch recommends keeping the shard size between 20GB and 40GB. Too small of shards results in excessive communication between the shards, as the data has to be synchronized between them. Metadata forms containing an excessive amount of sords with full text should be split into multiple shards whenever possible.

The number of shards per metadata form can be set on the ELO iSearch configuration page, but changing this number requires all data to be reindexed.

Setting the number of replicas per index

Replicas are like copies of an index. You can choose any value, including 0. One primary shard is created for each shard, as well as the specified number of replica shards in the cluster. Primary shards and replica shards are always created on different nodes to be able to continue providing data in case of a failure. If a node fails, the primary shards that were previously allocated to this node are allocated to a different node. The replica shards are distributed to other nodes to ensure the best possible system reliability and performance.

Another advantage of replica shards is parallel search queries: Elasticsearch distributes search queries to different primary shards and replica shards in order to balance the load of the search queries.

The health status of an index depends on what shard are available: If the primary shard is available, the status is *yellow*. If all replica shards are also available, the status is *green*.

The number of replica shards should be adjusted to the number of nodes in the cluster so that the shards can be distributed to several nodes in order to ensure system reliability. On the other hand, each replica requires just as much memory as the primary shard.

The default setting for the number of replicas is 1. In a cluster of more than one instance, each shard is replicated once, meaning it exists twice in total. Changing the number of replicas does not require reindexing.

To speed up reindexing in a cluster, you can set the number of replicas to `0` for the duration of reindexing and then increase it once reindexing is complete. If the number of replicas > 0, the data in both the primary shard and each replica shard is indexed at the same time. This means that analysis and conversion to Elasticsearch data takes place several times in parallel. If the number of replicas is set to `0`, the data is only processed in one node during reindexing. Once the number of replicas is increased, the indexed primary shards are copied to the other nodes, which is generally much faster. Of course, system reliability cannot be guaranteed during reindexing.

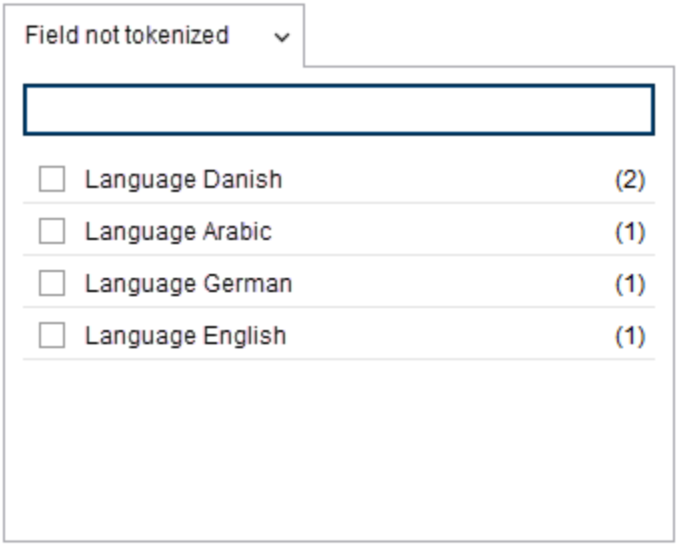
The setting applies to an index and can be changed to 2 as in the following metadata form with ID `42` in the *repository* repository, for example with the curl tool (`https://curl.haxx.se/`):

```
curl -X PUT "localhost:9200/repository¶42/_settings" -H 'Content-Type: application/json' -d'
{
    "index" : {
       "number_of_replicas" : 2
    }
}
'
```

If you'd like to change the setting for all indices, do not specify an index and use the URL the `localhost:9200/_settings`.

Tokenization

The *Disable tokenization* option for a field template disables the search rules and patterns for this field. The exact values in the metadata appear in the filter list. For this field, the search also works with exact values. In this case, partial terms are not taken into account.



□ Language Danish (2)
□ Language Arabic (1)
□ Language German (1)
□ Language English (1)

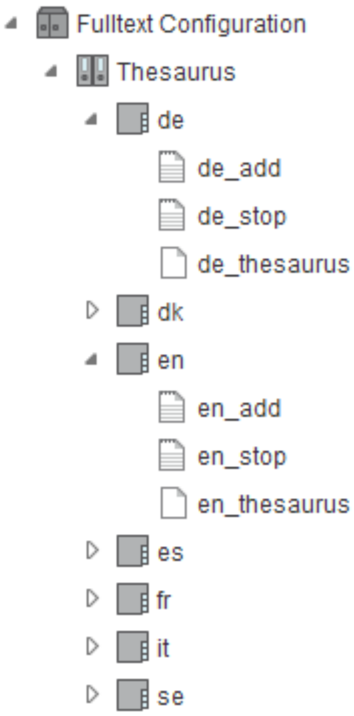
When search rules and patterns are enabled, the filter list only contains individual words – both the original words as well as their root forms. The search options in this filter correspond to those of the search input field.

Integrating custom thesauruses

You can customize the thesauruses that ELO iSearch uses for a repository. The necessary files are located at the following path in ELO:

`Administration//Fulltext Configuration//Thesaurus`

This path contains a country-coded child directory containing three documents for each language.



- Fulltext Configuration
  - Thesaurus
    - de
      - de_add
      - de_stop
      - de_thesaurus
    - dk
    - en
      - en_add
      - en_stop
      - en_thesaurus
    - es
    - fr
    - it
    - se

<Language>_thesaurus: The thesaurus supplied by ELO for this language. You shouldn't change this file so that it can be replaced with a newer version when performing an update.

<\Language>_add: In this document, you can enter terms to be added to the thesaurus as synonyms.

<Language>_stop: With this document, you can exclude specific terms from the thesaurus that do not fit your application.

> **Please note**
>
> After these documents are changed, the ELO Indexserver has to be restarted so that the thesaurus can be rebuilt based on the changed files.

Cluster operation

You can read about how to run ELO in a cluster in the documentation on ELO iSearch under ELO server – Installation and operation.

Recommendations in case of error messages

Error message containing 'pending translog recovery'

After an unplanned restart of the ELO iSearch service, the iSearch log may contain error messages with the text `pending translog recovery`.

Example:

```
[2023-06-29T13:22:51,662][WARN ][o.e.i.f.SyncedFlushService] [ELO-iSearch] failed to flush shard
java.lang.IllegalStateException: [contelo¶2][0] flushes are disabled - pending translog recovery
        at org.elasticsearch.index.engine.InternalEngine.ensureCanFlush(InternalEngine.java:2572
```

There may be a lot of these messages and they might appear when ELO iSearch is closed and didn't manage to transfer all transactions to the index before shutting down (e.g. if there wasn't enough RAM or the hard disk was full). On bootup, Elasticsearch then attempts to transfer the Translog file to the index. With a high volume of data, this can take an hour or possibly even several hours. The indexes are not available during this time.

Recommendation:

First, you have to wait until the entire Translog has been transferred to the index and these messages no longer appear. Next, you should check what the original problem was that caused the ELO iSearch service to shut down.

Avoid starting re-indexing via the ELO iSearch configuration page while these messages are appearing. Generally, the index is consistent again once Translog recovery is completed and so re-indexing is not necessary.

If the system is forced to shut down during Translog recovery, this may result in permanently corrupt index data that can no longer be fixed with Translog recovery. As a result, re-indexing must be performed or a fresh install of the ELO iSearch may even be necessary.

Power search

The ELO iSearch power search enables users to send complex search queries to ELO iSearch and use the iSearch syntax.

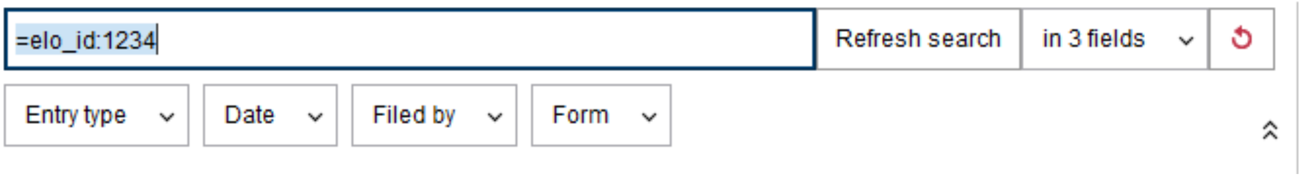
The power search function in ELO iSearch allows you to access the underlying Elasticsearch database. As a result, users can directly access fields that they cannot select using a filter as well as use the Elasticsearch syntax to run complex search queries.

The ability to access internal structures is also the downside of the power search given that the syntax is highly technical and requires knowledge of the internal data structures. It should also be noted that these internal structures can change in future versions of ELO, meaning that existing search requests may no longer work or no longer work properly.

> **Important**
>
> Information: You should only use the power search in exceptions, i.e. if a normal search query will not achieve the desired result, such as with certain administrative requests.

Using the power search

You can use the power search in the ELO Java Client as well as in the ELO Web Client. Starting with an equal character (=), enter the search query in the field provided as shown in the example below.



=elo_id:1234

The power search sends requests straight to the Elasticsearch via the ELO Indexserver. The ELO Indexserver carries out an authorization check to ensure that you only see the documents that you have permissions to even when running the power search.

The power search syntax corresponds to the syntax *QueryStringQuery*, as documented by Elasticsearch:

https://www.elastic.co/guide/en/elasticsearch/reference/5.6/query-dsl-query-string-query.html#query-string-syntax

It is especially important that you are able to escape special characters. This is necessary if the user wants the character to be interpreted as a normal letter in order to ignore any special meaning in the context in which they are being used, which would affect the search. The following characters have a special meaning in the syntax *QueryStringQuery* and must therefore be escaped if this meaning is to be ignored:

Characters	**Meaning**
+<word>	<word> must occur in the document
-<word>	<word> must not occur in the document (unless a space precedes it)
<word1> && <word2>	The document must contain both words
<word1> and <word2> separated by two pipe symbols	The document must contain at least one of the two words
> x < x	Greater/less than (works for numbers or words)
>= x <= x	Greater than or equal to/less than or equal to (works for numbers as well as words)
=	Has no significance on its own, at the beginning of the line: power search
{x TO y}	Range search (without x and y)
[x TO y]	Range search (including x and y)
<word>^	Significance of word
"<word> <word>"	Phrase search
<word>~	Fuzziness (fuzzy search)

Characters	Meaning
*	Wildcard, any number of characters
?	Wildcard, exact number of characters to replace
\	Escape character
/	No meaning
<field name>:<word>	Search <field name> for <word>

Field names

It is possible to query Elasticsearch fields directly provided that the user knows the name of the field. ELO does not provide a list of the field names, but administrators can request them via the Elasticsearch REST interface.

```
http://<host>:<port>/<repository name in lowercase>/_mapping?pretty
```

Elasticsearch provides more information on interpreting requests here: *https://www.elastic.co/guide/en/elasticsearch/reference/5.6/index.html*

> **Please note**
>
> ELO reserves the right to change the names of fields in between versions. It is therefore possible that a search query will work with one version, but not with the next.

Examples of requests

It often makes sense to find out whether a document can be found using the iSearch. You can do so using the following search query:

```
=elo_id:<object_id>
```

To perform the search query, you need the right *View all entries, ignore permissions*. If the search does not come up with the correct results, the document has not be added to the iSearch.

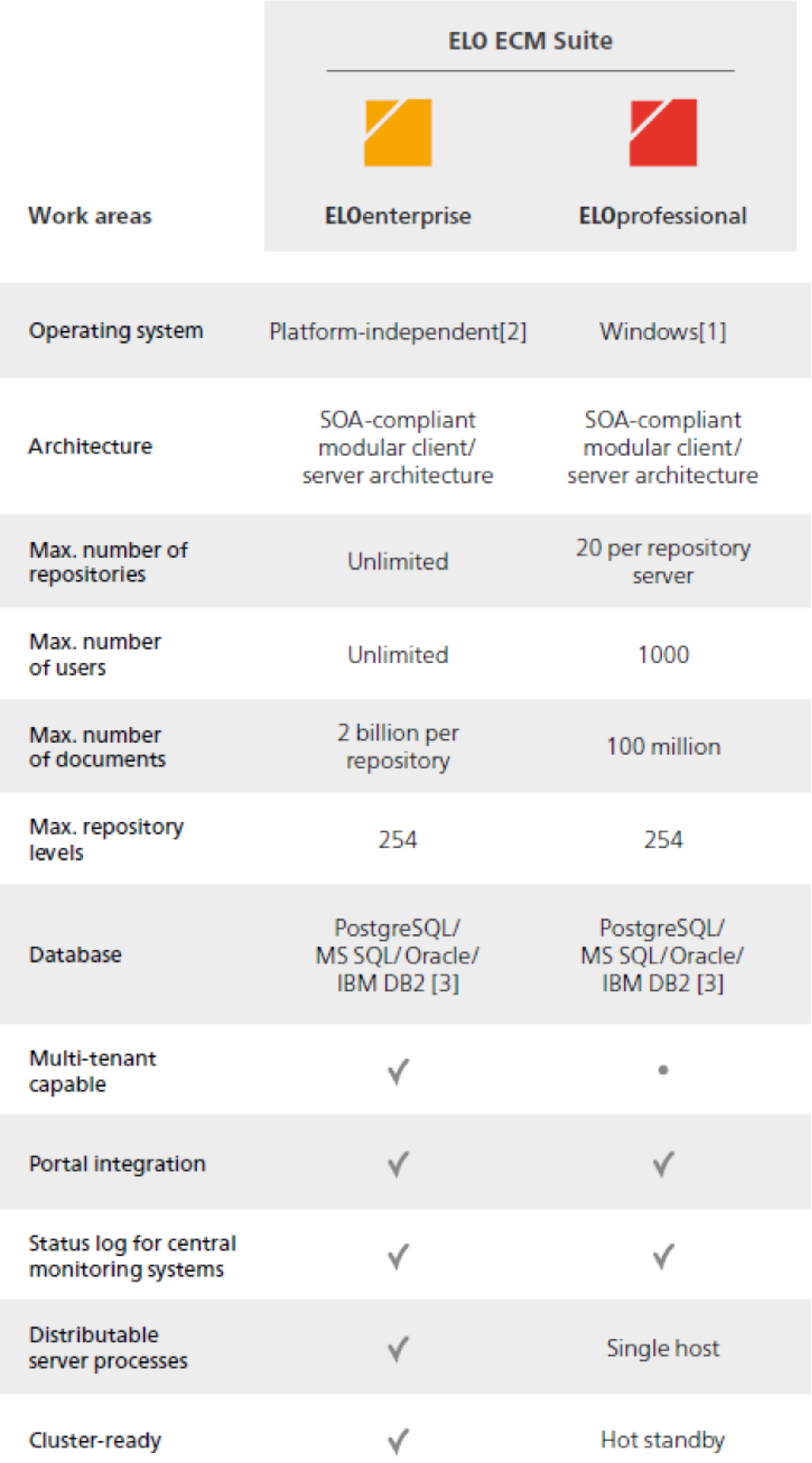
The search for `ELO[123` leads to a syntax error since the square bracket is a special character that can normally be used for a range search. You can still search for this character by issuing the following request in the power search:

```
=ELO\[123
```

License terms when distributing service processes

ELOprofessional systems are only cluster-ready to a certain extent. They only allow for hot standby operation of individual components. ELOenterprise systems, however, are fully cluster-ready.

With ELOprofessional, server processes may only be distributed within the same host, i.e. on a server system. In contrast, with ELOenterprise processes can be distributed across multiple hosts (VMs or physical servers).



	ELO ECM Suite	
Work areas	ELOenterprise	ELOprofessional
Operating system	Platform-independent[2]	Windows[1]
Architecture	SOA-compliant modular client/ server architecture	SOA-compliant modular client/ server architecture
Max. number of repositories	Unlimited	20 per repository server
Max. number of users	Unlimited	1000
Max. number of documents	2 billion per repository	100 million
Max. repository levels	254	254
Database	PostgreSQL/ MS SQL/ Oracle/ IBM DB2 [3]	PostgreSQL/ MS SQL/ Oracle/ IBM DB2 [3]
Multi-tenant capable	✓	•
Portal integration	✓	✓
Status log for central monitoring systems	✓	✓
Distributable server processes	✓	Single host
Cluster-ready	✓	Hot standby

With ELOprofessional, the ELO server components (e.g. ELO Automation Services (ELOas)) can be distributed across multiple Tomcat servers as long as they are located on one server system (VM or physical server). In contrast, with ELOenterprise, Tomcat servers can be installed on multiple server systems (VMs or physical servers).

Glossary

Term	Explanation
Escape	The escape character prevents the executing program from recognizing the next character as a functional character. Escaping is necessary if the user wants the character to be interpreted as a normal letter in order to ignore any special meaning in the context in which they are being used.
Garbage Collection (GC)	In software and information technology, garbage collection, abbreviated GC, refers to automatic memory management that minimizes the resources required by a computer program. At runtime, resources that are no longer required are identified automatically to free them up.(Source: https://en.wikipedia.org/wiki/Garbage_collection_(computer_science), last edited March 29, 2022)
Lucene	Apache Lucene is a high-performance, open-source program library especially suitable for full text search. Lucene can index any type of documents, including texts, numbers, date values, or GPS positions. Lucene is the underlying data storage structure for the Elasticsearch search and analytics engine.
NODE/ NODES	An Elasticsearch server corresponds to one node.
OCR	*Optical character recognition* OCR is the automatic recognition of text within images. Scanned documents are converted into text form using OCR.
OSGi (OSGi plug-ins)	The OSGi Alliance (previously *Open Services Gateway initiative*) specifies a hardware-independent dynamic software platform, that makes it easier to modularize and administrate (*service registry*) applications and services with a component model (*bundle/service*). The OSGi platform requires a Java Virtual Machine (JVM) to provide the OSGi framework.(Source: https://en.wikipedia.org/wiki/OSGi, last edited March 29, 2022)
REST	Representational state transfer (abbreviated REST or ReST) refers to a programming paradigm for distributed systems, in particular for web services. REST is an abstraction of the structure and behavior of the World Wide Web. It aims to create an architectural style that better represents the requirements of the modern web. REST primarily differs because it uses a uniform interface. REST is primarily focused on machine-to-machine communication.(Source: https://en.wikipedia.org/wiki/Representational_state_transfer, last edited March 29, 2022)
Replication	Replication refers to the repeated storage of the same data. In the Elasticsearch, you can set how many replicas of a shard you would like with this backup function. These are then intelligently distributed to cluster nodes. This distributes searches to several nodes. In case of node failure, data replications exist, keeping the search available without losing data.
Shard	The data of an Elasticsearch search index is split into shards.

Term	Explanation
Language analyzer	Elasticsearch is supplied with analyzers for the most common languages. These language analyzers perform four primary functions: they divide the text into single words, convert the tokens into lowercase letters, remove common stop words, and derive the root forms of the tokens.
Search index	Elasticsearch stores all data in a search index. This index is then split into types and documents with properties.
SWAP mode/ swapping	In IT, swapping within the storage hierarchy refers to moving data from a higher to a lower level or vice versa. Swapping is intended to combine the advantages of higher levels, that is higher speed, and lower levels, that is higher capacity and less expensive memory.(Source:https://de.wikipedia.org/wiki/Swapping, last updated March 29, 2022)
Token	When creating or updating a data set, the Elasticsearch breaks down the searchable text in the data set into individual terms, referred to as tokens. The tokens are then stored in the search index. A search query is also broken down into tokens. These are then searched for in the search index. A token does not always correspond to a grammatically correct word. Instead, it is a breakdown and simplification of words meaning that searching for *car* also finds documents containing only the word *cars*.
Tomcat	Apache Tomcat is an open source web server and web container that implements the specifications for Java servlets and Jakarta Server Pages (JSP), allowing web applications written in Java to be run as servlets or on a JSP basis.
YML	YML is the file extension for the simplified markup language YAML. YAML is used for data serialization and is based on XML as well as the data structures in the languages Perl, Python, and C.

Filing definition

Filing definition

When indexing a document in the ELO repository, the metadata, full text content, and information from feed posts associated with the document are added to the iSearch database.

During the indexing process, full text and feed content is analyzed in steps based on the language and the result is stored in the corresponding Elasticsearch index. In the case of metadata, text fields are analyzed and the contents of each field are stored unchanged (e.g. as keyword, date, etc.).

This makes it easier for users to find the information that is relevant for their specific use case. They can simultaneously search the full text and feed contents as well as the metadata.

Steps before indexing

The metadata of documents in the ELO repository are available in the SQL database and can be immediately accessed by the Indexserver. The full text information, on the other hand, first needs to be extracted from a document. This task is performed by the ELO Textreader module.

The Textreader fetches the document from the Indexserver, extract its full text contents and sends this back to the Indexserver, which stores it as a separate file in the ELO repository. The Textreader also uses OCR software to extract text contents if required (e.g. in image files).

Regular indexing process

The indexing process is performed by the Indexserver. The Indexserver checks at regular intervals (updater interval can be configured in the ELO iSearch configuration) which documents have been added, modified, or deleted since the last indexing run. These need to be updated in the iSearch indexes.

The Indexserver retrieves the metadata and feed information of the documents from the SQL database. Full text contents (if available) are retrieved from the repository with the ELO Textreader. Once processed, the information is forwarded to the iSearch, i.e. the Elasticsearch instance, which then indexes the field contents. In the process, the full text and feed content is analyzed in accordance with the language as described above.

> **Please note**
>
> You should consider carefully which documents to add to the iSearch database. Adding too much text information will reduce search engine performance. It can make sense to exclude certain types of documents (e.g. entire book, brochures, etc.).

Background reindex

Introduction

This document describes the *Upgrade Index* function in the ELO Server Setup, as well as its application.

> **Please note**
>
> The *Upgrade index* function in the ELO Server Setup must not be performed on the production database in ELO 12 or higher, as changes to the database during the installation process will cause issues in older versions of ELO.
>
> If you still want to use the function, you must duplicate the data of the production database in a separate database instance. You must specify this separate database instance when you run the *Upgrade index* function.
>
> These instructions do not cover the installation. A reindex of the full text database takes place before installation and should be completed before you begin installation.

Descriptions of functions

Before you update ELO iSearch to a new version, you need to reindex the full text database. To ensure the availability of the full text search during this process, the *Upgrade Index* function in the ELO Server Setup allows you to set up the full text database before updating the server.

An isolated system is used, which consists of an Apache Tomcat with ELO Indexserver and an ELO iSearch instance that can be used for reindexing, while the ELO server components continue running with their former versions. Once reindexing is completed, the ELO server can be updated with very little downtime.

Use case

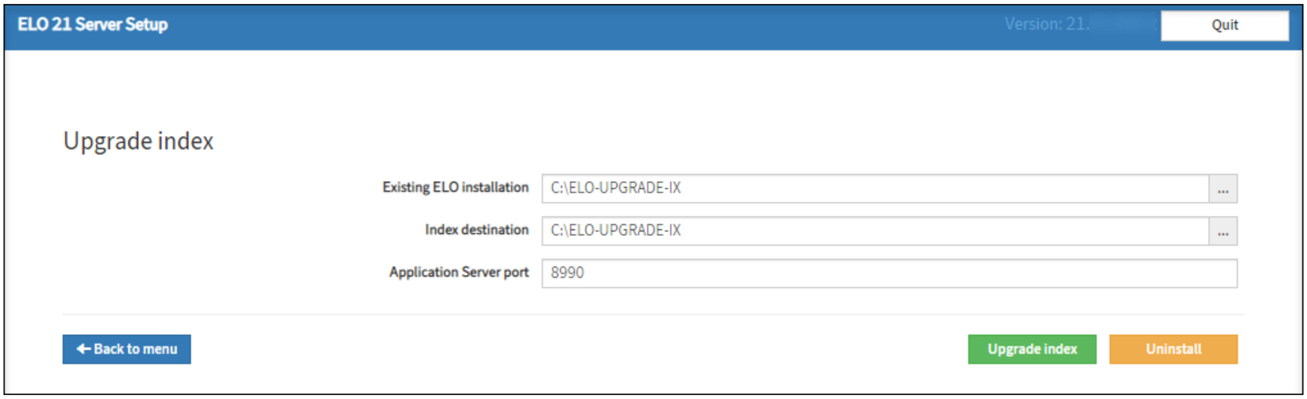
You should use the *Upgrade Index* function if the duration of the reindex process and the resulting limited availability of the full text search would exceed the tolerance limit.

A reindex of up to approx. 1 million documents (sum of all repositories) should be done over the weekend. Of course, this depends on the hardware used as well as the average document size.

Installation

To start reindexing, start the current version of the ELO Server Setup.

Click *Upgrade Index* on the ELO Server Setup start page.



Existing ELO installation: C:\ELO-UPGRADE-IX

Index destination: C:\ELO-UPGRADE-IX

Application Server port: 8990

Existing ELO installation: File system path to the existing ELO installation.

Index destination: File system path where you want to install the system for setting up the index. Once reindexing is completed, we recommend specifying a path different from the ELO installation path.

Application Server Port: Port for the Indexserver Tomcat. This port must not already be in use by another operating system service, including ELO services.

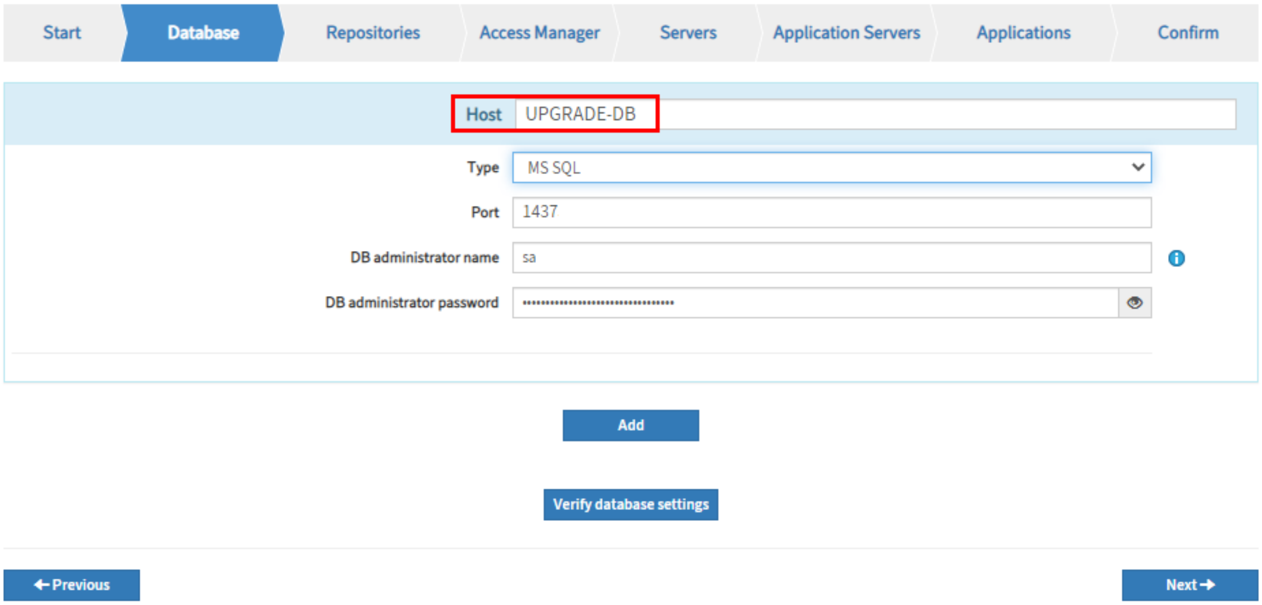
Upgrade Index: Configures the system for reindexing. First, you are forwarded to a summary, where you can then start the actual installation process.

Uninstall: Uninstalls the system installed during the *Upgrade Index* process. This must be done after reindexing. The services and programs installed for reindexing are removed. Data generated in the process, such as the full text database, is retained and must be subsequently removed manually.

Upgrade Index

Selecting the *Upgrade Index* button takes you to the installation overview page. The information summarized there only applies to relevant points of the reindexing process.

> **Please note**
>
> The *Upgrade Index* function in the ELO Server Setup must not be performed on the production database in ELO 12 or higher, as changes to the database during the installation process will cause issues in older versions of ELO (e.g. affects the substitution rules).



Host UPGRADE-DB

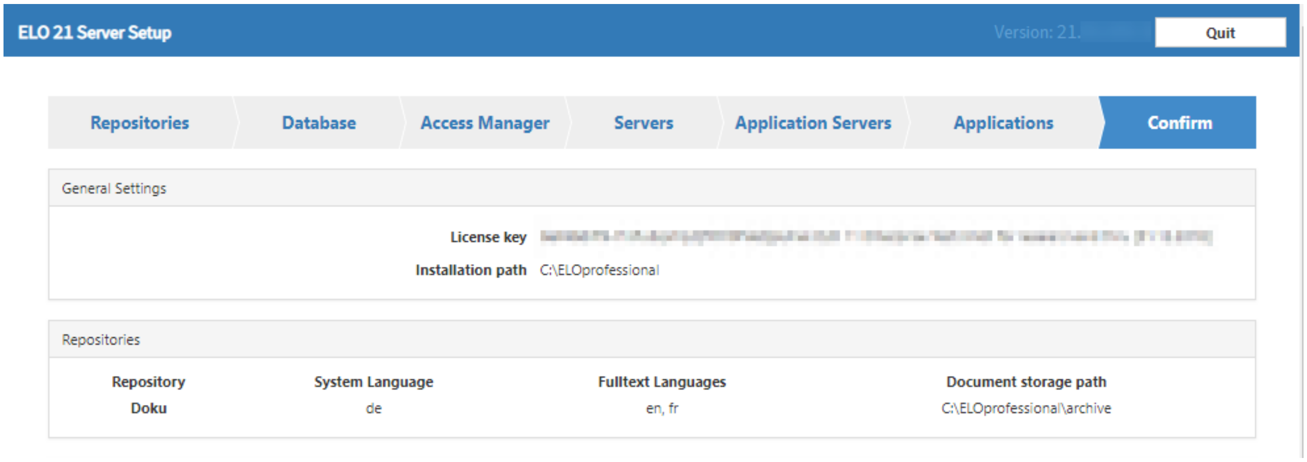
Type MS SQL

Port 1437

DB administrator name sa

DB administrator password ••••••••••••••••••••••••••••••••

If you still want to use the function, you must duplicate the data of the production database in a separate database instance. You must specify this separate database instance in the *Host* field on the *Database* tab.



General Settings

License key [redacted]

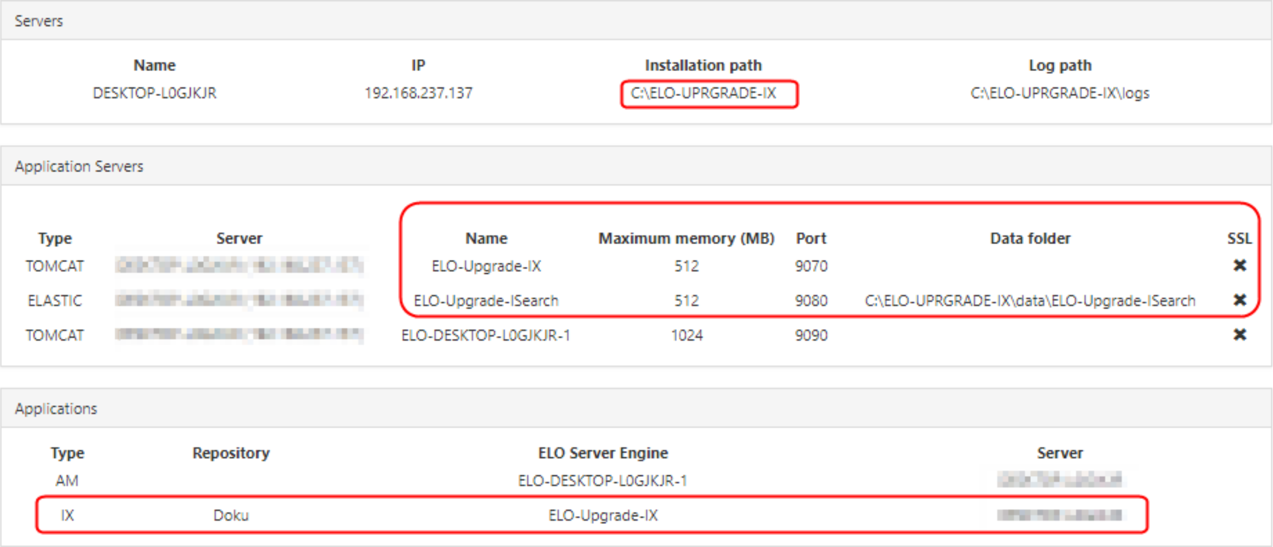
Installation path C:\ELOprofessional

Repositories

Repository	System Language	Fulltext Languages	Document storage path
Doku	de	en, fr	C:\ELOprofessional\archive

The installation path specified here and the document paths must correspond to the paths of the ELO installation.

> **Please note**
>
> You must use a current license key.



Servers

Name	IP	Installation path	Log path
DESKTOP-L0GJKJR	192.168.237.137	C:\ELO-UPRGRADE-IX	C:\ELO-UPRGRADE-IX\logs

Application Servers

Type	Server	Name	Maximum memory (MB)	Port	Data folder	SSL
TOMCAT	[redacted]	ELO-Upgrade-IX	512	9070		✖
ELASTIC	[redacted]	ELO-Upgrade-ISearch	512	9080	C:\ELO-UPRGRADE-IX\data\ELO-Upgrade-ISearch	✖
TOMCAT	[redacted]	ELO-DESKTOP-L0GJKJR-1	1024	9090		✖

Applications

Type	Repository	ELO Server Engine	Server
AM		ELO-DESKTOP-L0GJKJR-1	[redacted]
IX	Doku	ELO-Upgrade-IX	[redacted]

The paths (*Installation path*) under *Servers* must correspond to the path defined under *Index destination*. This also applies to *Data Folder* under *Application Servers*. This is where the ELO iSearch full text database is located after reindexing.

Select *Install* to start the installation of the upgrade system.

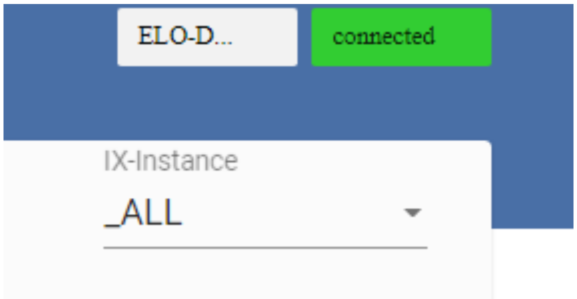
The ELO Server Setup installs the upgrade system consisting of an ELO Indexserver and an instance of ELO iSearch.

> **Information**
>
> No changes are made to the previous system.

After the installation is completed, select *Finish* to close the ELO Server Setup.

Reindexing in the upgrade system

When the installation of the upgrade system is completed, go to the *ELO Elasticsearch Configuration* page (from the *status page of the newly installed ELO Indexserver > Configure Options > iSearch config*).



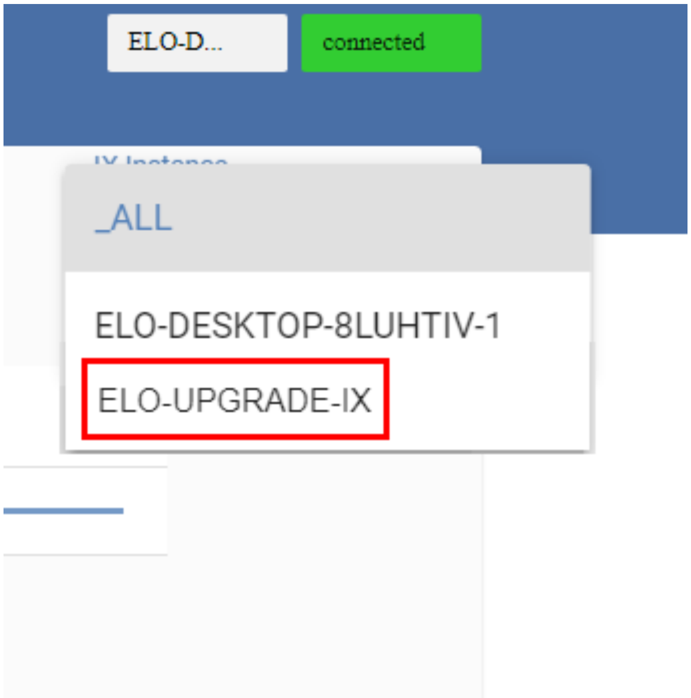
IX-Instance: _ALL

Select the instance *_ALL* from the *IX-Instance* drop-down menu if it is not already selected.

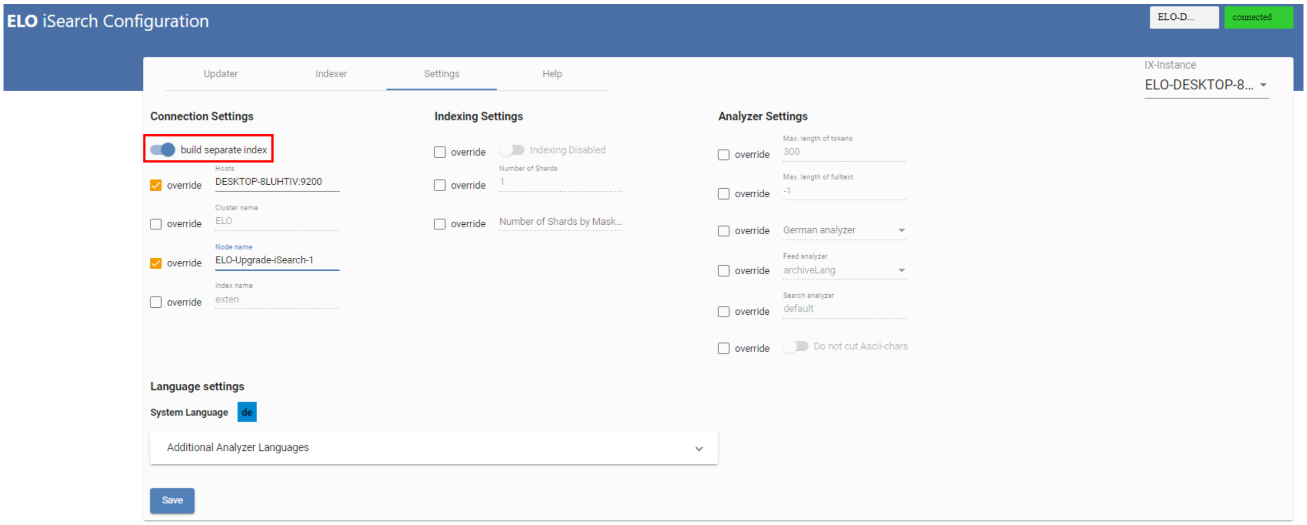
Check the settings for the instance *_ALL* on the *Settings* tab.

Make sure that the correct languages are selected in the *Language settings* area.

Click *Save* to save the settings.



Select the instance of the upgrade system from the *IX-Instance* drop-down menu.



IX-Instance: ELO-DESKTOP-8...

Connection Settings

build separate index (on)

☑ override Hosts: DESKTOP-8LUHTIV:9200

□ override Cluster name: ELO

☑ override Node name: ELO-Upgrade-iSearch-1

□ override Index name: exten

Indexing Settings

□ override Indexing Disabled

□ override Number of Shards: 1

□ override Number of Shards by Mask...

Analyzer Settings

□ override Max. length of tokens: 300

□ override Max. length of fulltext: -1

□ override German analyzer

□ override Feed analyzer: archiveLang

□ override Search analyzer: default

□ override Do not cut Ascii-chars

Language settings

System Language: de

Additional Analyzer Languages

Check the settings for the instance on the *Settings* tab.

The *Index name* field must contain the name of the repository (in lower case).

Enable the option *build separate index*.

Click *Save* to save the settings.

Check the settings on the *Indexer* tab.

You need to enter the value *-1* into the fields `From object id` and *To object id*.

Click *Save* to save the settings.

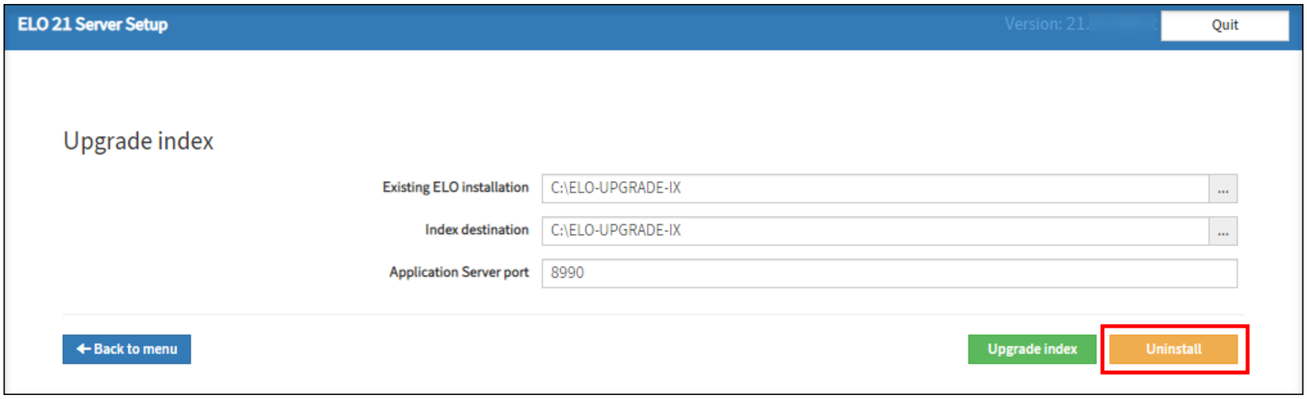
Select *Create new index* to start the reindexing process.

Uninstalling the upgrade system

Once the reindexing process is completed, you can uninstall the upgrade system. The index that was created is retained.

To proceed, run the ELO Server Setup again.

Select *Upgrade Index* on the ELO Server Setup start page.



Upgrade index

Existing ELO installation: C:\ELO-UPGRADE-IX

Index destination: C:\ELO-UPGRADE-IX

Application Server port: 8990

The Upgrade index screen opens. The path specifications should correspond to those of the previously created upgrade system.

Select *Uninstall*.

The upgrade system is uninstalled.

Once the uninstallation is completed, select *Finish* to close the ELO Server Setup.

Updating to a new ELO version

If the upgrade system has been successfully uninstalled, you can now update the ELO server.

To do so, run the ELO Server Setup again.

Update the ELO server in the usual way (for more information, see the *ELO Server* documentation) with the *Standard Configuration/Update* or *Advanced Configuration/Update* functions.

After the installation is completed, select *Finish* to close the ELO Server Setup.

Migrating data

Once you have successfully installed the new ELO version, you can transfer the index created by the upgrade system to your live system.

Stop the ELO iSearch service.

Navigate to the following folder in the file system:

```
<installation path of the upgrade system>/data/ELO-Upgrade-ISearch/
```

Copy the *index* folder.

Navigate to the following folder in the file system:

```
<installation path of the ELO server>/data/
```

Insert the copied folder and replace the existing *index* folder.

Start the ELO iSearch service.

Check the settings on the *ELO Elasticsearch Configuration* page.

Please make sure that the *build separate index* button is not enabled for any instance.

Test ELO iSearch in an ELO client.

If migration was successful, you can delete the remaining data from the upgrade system.

Upgrade Index on a separate server

Reindexing requires a great amount of CPU resources and can cause high load on the server. You can mitigate the effects on the live system by reducing the process priority and performing reindexing on a separate server.

The setup requires two configuration files (`%HOMEPATH%\.elosetup.conf` and `<ELO>\config\serversetup2\elosetup.conf`) to run the *Upgrade Index* function on another server. Create the `<ELO>\config\serversetup2\` folder. The path must match the live system's path. Other folders, such as `<ELO>\archive` or `<ELO>\data` are not required.

Next, the function can be run as described in Upgrade Index.

Upgrade Index when migrating from ELO 9

You can also get the full text database ready using this method when migrating from ELO 9 to ELO 10.1 or higher. Rolling out Indexserver 10 also updates the database. For this reason, version 9.16.110 of the Indexserver or higher is required for ELO 9. An update from ELO 8 to ELO 10.1 is not supported.

As described in Upgrade Index on a separate server, the setup requires two configuration files for the *Upgrade Index* function, but these do not exist in ELO 9 systems and need to be created first. These must be created first. The setup creates the two files during the upgrade process.

Important

> You should not complete the upgrade process. After entering the existing installation path and the license key, select *Configure Upgrade* or *Review Upgrade* after the system is analyzed to cancel the upgrade process. The upgrade must not be installed!

The two files are then created. Now, continue with Upgrade Index or Upgrade Index on a separate server.