Textkernel Search!

User Guide

Version 3.30.2

textkernel

Disclaimer

Textkernel BV provides this publication “AS IS” without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability or fitness for a particular purpose.

The publication could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. Textkernel BV may make improvements and/or changes in Search!, Sourcebox, Textractor and any other products described in this publication at any time.

Ownership

The intellectual property of Search!, Sourcebox, Textractor, and all other Textkernel products mentioned here belong to Textkernel BV, located in the Netherlands.

Usage

The Search! software may have been licensed to third parties for use/integration in a particular project, a particular time frame, and for a particular purpose. It is therefore not permitted to use it for any other purpose than described in the associated license agreement. Unauthorized use of this software, reverse compilation, making illegal copies or the reverse engineering of the software and algorithms is STRICTLY PROHIBITED.

See the appropriate license agreement for further details regarding usage.

Confidential Information

 This set of documentation contains confidential information of Textkernel BV, and is protected by copyright of Textkernel BV. It is expressly prohibited to wholly or partly copy this documentation and/or to give third parties, or unauthorized persons within your organization access to this documentation, except with prior written consent of Textkernel BV.

Licensing

For more information on licensing Textractor (or components thereof), please contact Textkernel BV, The Netherlands.

website
http://www.textkernel.nl
e-mail
info@textkernel.nl

Feedback

To constantly improve our software and documentation we very much appreciate your comments on any aspect of our work. Please send feedback to **service@textkernel.nl**

Table of Contents

1 Introduction and Overview

Textkernel Search! is a search engine designed for semantic search through CV's and other semi-structured documents. This document guides the user through the screens and describes all functionality of the system alongside.

Search! requires a web browser with JavaScript support. The following browsers are supported:

- Internet Explorer 8.0 and higher
- Firefox 3.6 and higher
- Chrome
- Safari 4 and higher

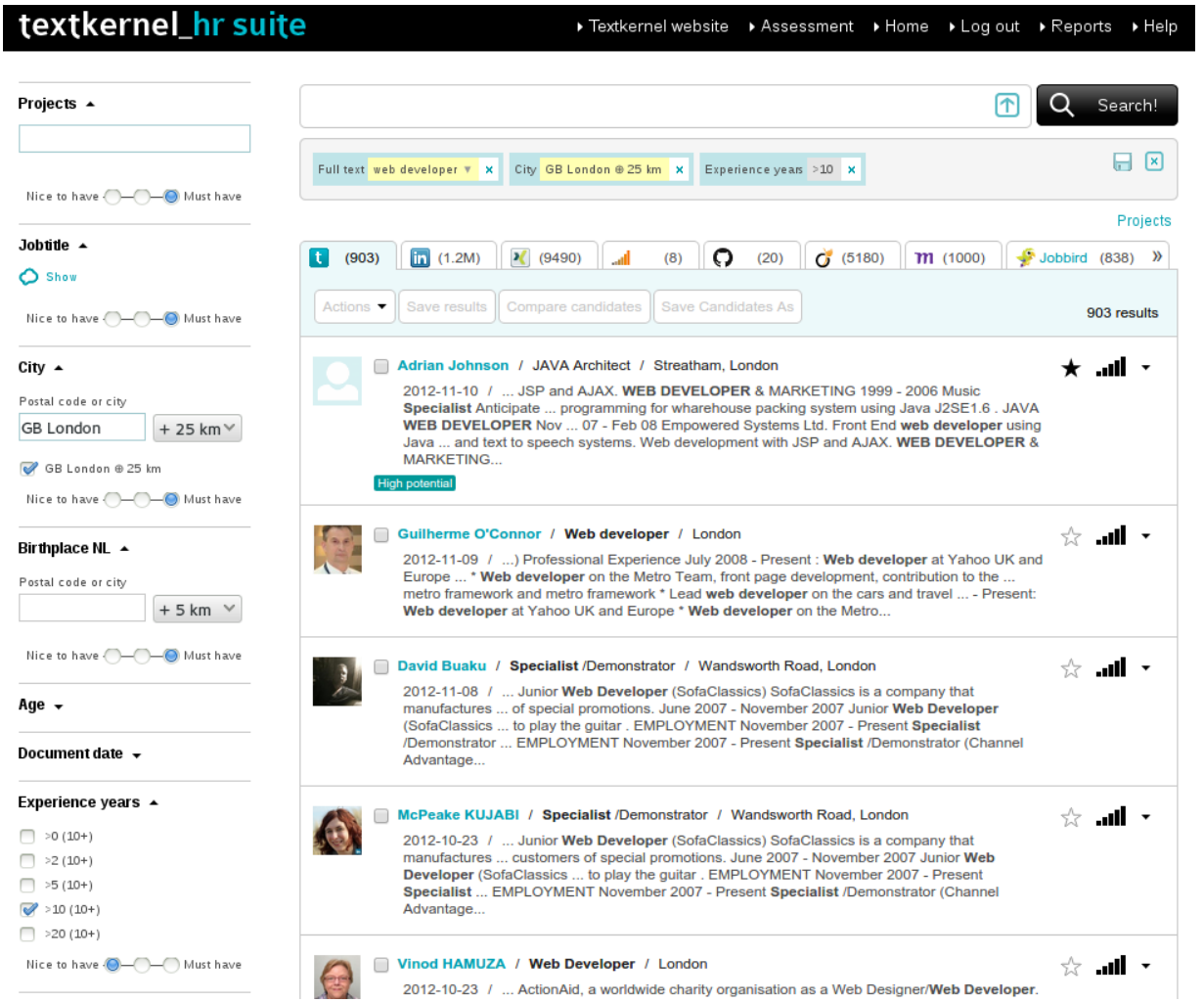


Figure 1. Screen Overview: The main Search! screen and its components.

1.1 Screen components

The Search! screen consists of 3 main components:

- the *Search Box:* in the upper right part of the screen allows to enter keyword queries. It also shows a summary of interpreted query parts as bread-crumbs and allows to easily manipulate the query. Section 2, **the Search Box**, describes all function in detail.
- the *Result Tabs:* in the middle of the screen show the query results of all configured sources. Section 3, **Result Tabs**, describes the different views on Search! results.
- the *Facet Bar:* on the left side of the screen allows to fine tune the query by facet selections, and provides an overview of the number of results for each defined category. Section 4, **the Facet Bar**, explains all features of the Facet Bar.

Furthermore, there are a number of screen components which can be opened:

- *Tag-clouds:* show frequent terms per field which can be selected to be added to the query. Section 5, **Tag-clouds**, describes tag-clouds and user interaction.
- *Saving dialogues:* allow the user to save searches, results and organize them in projects. Section 6, **Saving Projects, Searches and Results**, describes all actions in detail.
- *Comparison view:* shows selected results side by side and highlights the differences between the results. Section 7, **Comparison View**, describes how to navigate the compare view.

Finally Section 8, **the Query Language for Advanced Users**, covers all advanced features of the Search! query language.

2 The Search Box

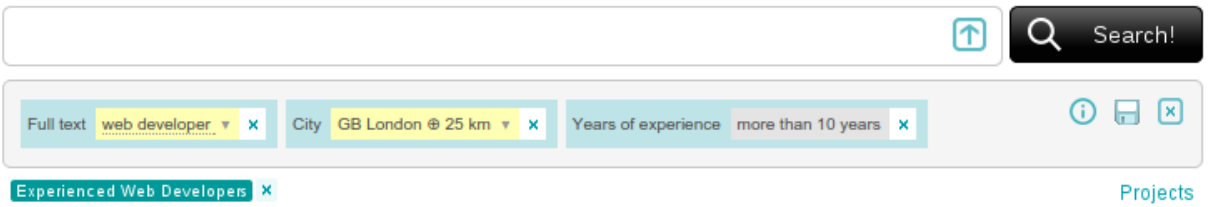


Figure 2. The Search Box: Allows to enter keyword queries and to manipulate them.

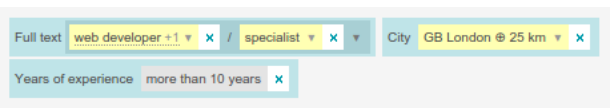
2.1 Interactions

	Description
Search!	start a new search with the keywords provided in the text field next to the button.
	(optional) start a new match search by uploading a document. Alternatively, Match! supports drag and drop of files from the desktop.
	(optional) shows the **query info** panel, seen only if the user has appropriate access rights
	(optional) shows the **save search** dialog, seen only if saving searches is enabled
	removes all query parts in the Search Box.
Projects	(optional) shows a **project management** dialog, seen only if projects management is enabled

2.2 Query Language

The text field in the Search Box allows to enter keywords queries. Next to simple words that are matched against the full content of the indexed documents, it is possible to express complex queries on fields, containing phrases, or proximity conditions. See **Section 5, The Query Language**, for more information.

2.3 Query Parts



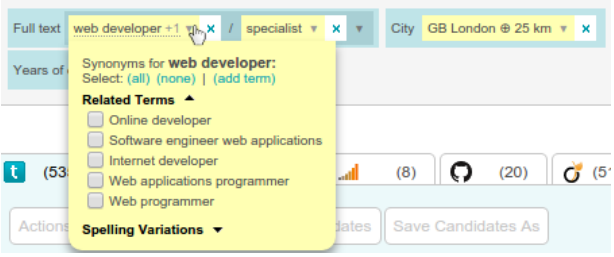
Interpreted query parts are shown as bread crumbs. The color of a bread crumb indicates whether a query part is "required" (yellow) or just "nice to have" (grey). If the user clicks on

the blue cross in a breadcrumb, the breadcrumb is removed from the query and the result view is automatically updated.

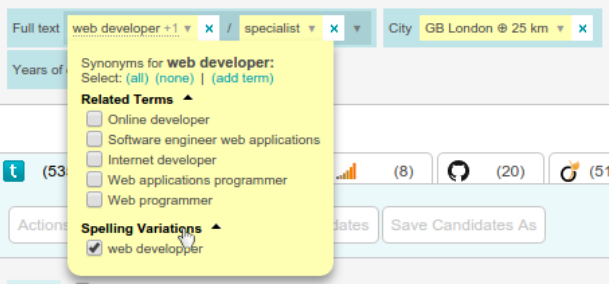
Query parts can also contain additional information, such as related terms or spelling variations. If a bread crumb contains more information, a small triangle is shown. Search! might be configured to automatically include some of this information while searching, in which case a plus sign is shown, followed by a number of automatically included terms.

Placing the mouse cursor over a bread crumb which has additional information selected shows a quick overview of the selected terms:

Clicking a bread crumb which has additional information shows a widget which allows selection of additional terms to expand the query. Selecting any of the terms adds it to the query and the result view is automatically updated. It is possible to automatically select or deselect all the additional terms by clicking "(all)" or "(none)" respectively.

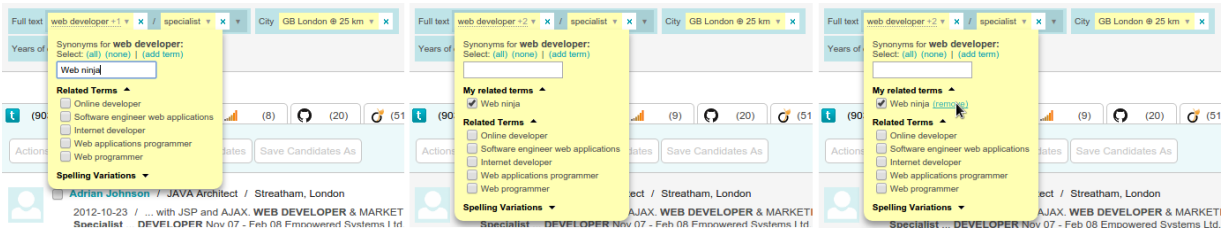


Terms in this widget are grouped by section. Every section can be configured to be automatically collapsed. If a bread crumb shows a count of selected terms (e.g. "+1") and no selections are initially seen in the widget, click a header of any of the collapsed sections to see the selected terms.

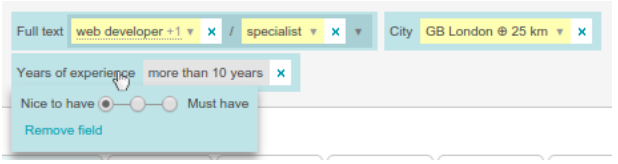


Depending on configuration, all synonym selections are remembered so that any new search with the same original term (breadcrumb) will then use the previous selected set of synonyms.

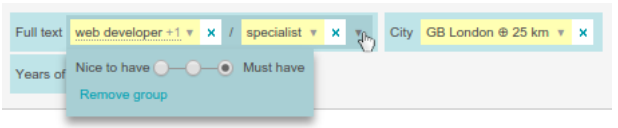
Also depending on configuration, it is possible to enrich the query with custom terms by clicking "Add term" and typing in the desired keyword or phrase. To remove a user-added related term, just hover over it and click "Remove".



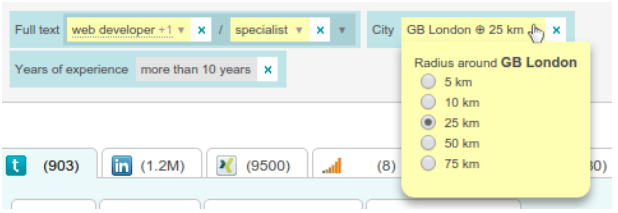
Clicking a field name (left of the breadcrumbs) shows a widget which allows changing the condition of all non-banned breadcrumbs in that field; alongside the link to remove the field from the query. For more details on conditions, please see the **Facet Selections** chapter.



Clicking the small triangle next to an OR group shows a similar widget which allows changing the condition of that OR group. alongside the link to remove the group. For more details on conditions, please see the **Facet Selections** chapter.



Clicking a bread crumb that shows a location and radius renders a widget that allows radius to be changed.



2.4 Auto-suggestion

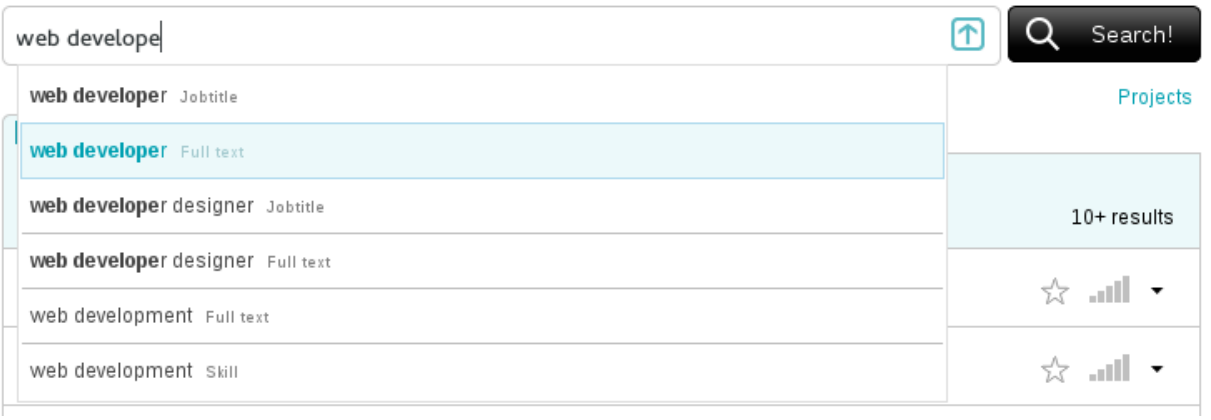


Figure 3. Auto-suggestion showing both completions and field suggestions

Auto-suggestion can be configured for any field that allows direct text input. Suggestions are offered as the user types and include both phrase completions, minor spelling corrections and field recognition.
In case of faceted fields (location, text or tagcloud) the suggestions are offered only for the terms related to that field. For the full-text search field, suggested terms are also followed with the relevant field suggestion, allowing for quick and easy fielded search.
To select a suggestion, either navigate to it with cursor keys on your keyboard and press the Return key or click on the suggestion.

2.5 Query information

Figure 4. Query info panel

When current user has appropriate access rights, the Query Info icon is shown which, when clicked, shows the exact query that is sent to the underlying search engine. This panel works for external and internal searchers alike and can be hidden either by clicking the Query Info icon again or the blue cross inside the panel.

3 Result Tabs

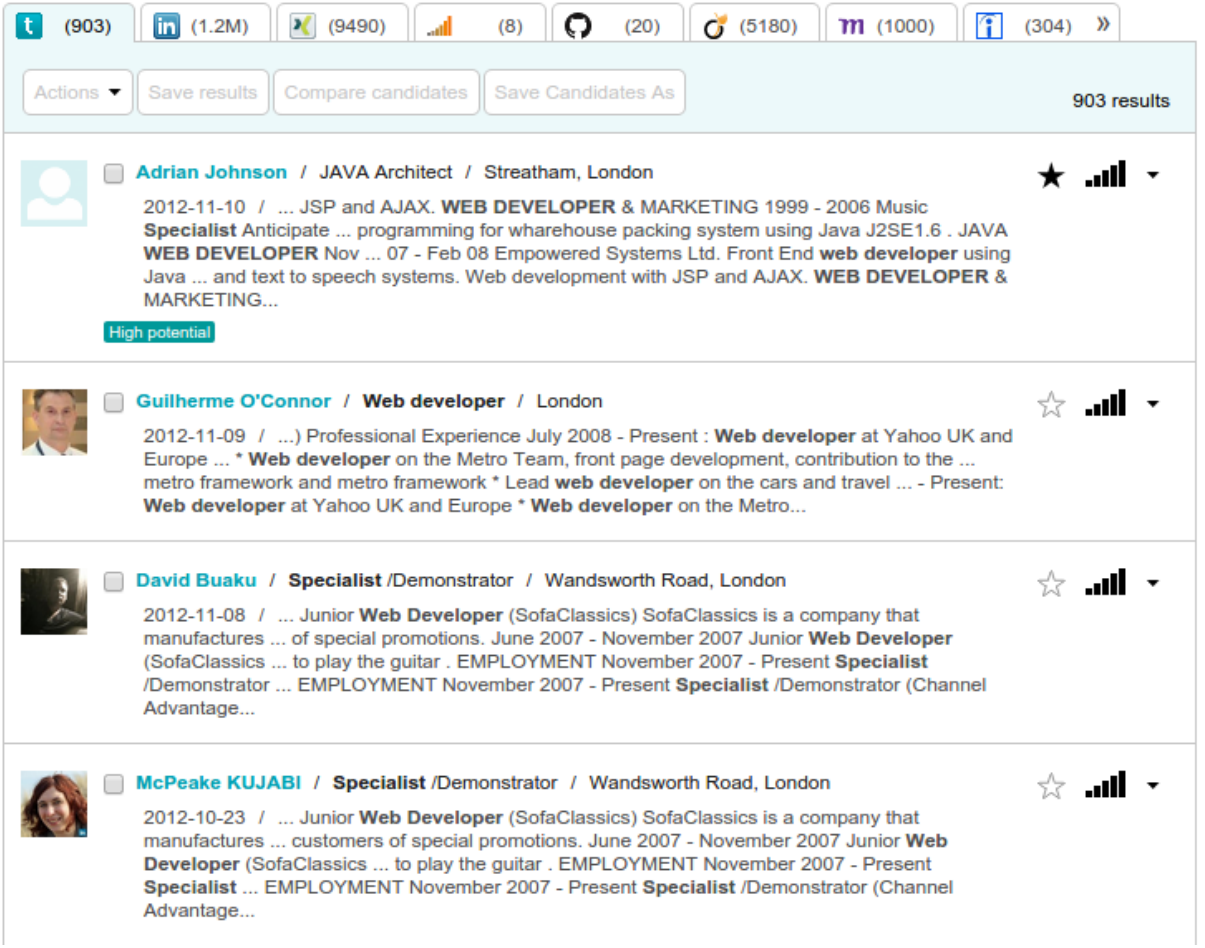


Figure 5. The Result Tabs: Showing search results from all configured information sources.

In the result tab the query results are presented to the user. For every configured information source the results of the query are presented on a different tab. Clicking a certain tab shows the results from that data source. The number next to the name of the tab shows the number of results found in the respective data source.

In case there are more tabs than the available space to render all of them, a small arrow will be shown which, when clicked, will show all of the remaining tabs in a drop-down.

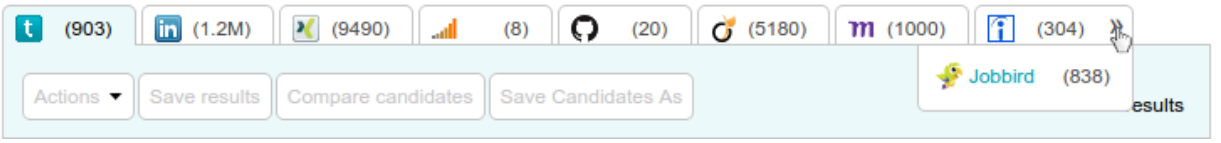


Figure 6. Too many tabs

Choosing a tab from the dropdown will replace the last tab with the selected one.

Figure 7.

All tabs are structured similarly:

- in the top right corner the total number of search results is shown,
- an action menu on the top left corner allows customized actions to be performed on selected results.
- all results are presented below, top to bottom, sorted by relevance,

3.1 Result item in the List View

The result item box contains a short description of the result, containing the metadata fields configured for the list view. If the query was not empty, the list view also presents a short snippet text for each result item with highlighted keywords to show where the query keywords are matched in the document.

	Description
	open the **expanded view** of the item containing additional information and possible actions that can be performed on a result item.
	The score visualisation indicates how many query criteria are matched by the result item. If all bars are filled, the result matches all criteria. If 4 out of 5 bars are filled it matches around 80% of the criteria. The score visualisation is based on estimates, hence it should be regarded as an indication only. Hovering this visualisation with a mouse cursor displays an informational widget showing which query parts have been matched for this query.
	Indicator whether the current result is saved in the active project (empty star: not saved, full star: saved). Clicking the star also saves or removes the result from the active project, depending on the current state.

3.2 Expanded View

- The expanded view displays more metadata of a result item. This view should help to roughly judge the relevance of an item.
- Matching search criteria are highlighted.
- Depending on the configuration different actions will be available per result item, such as openening the original document in a new browser tab.

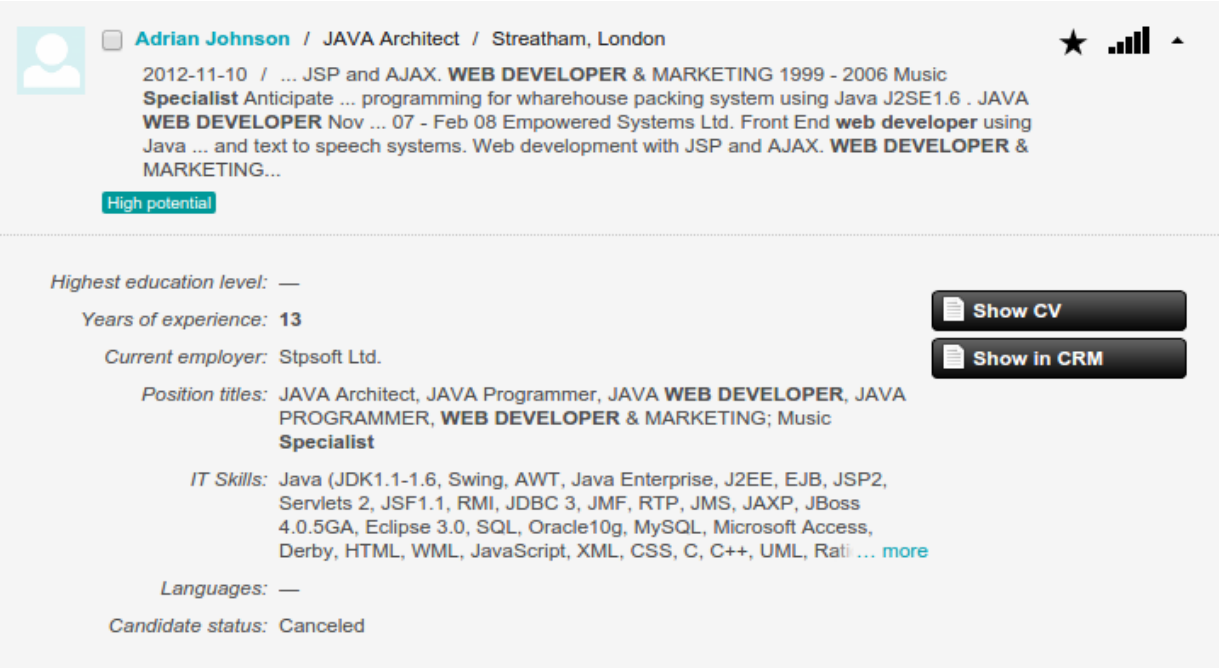
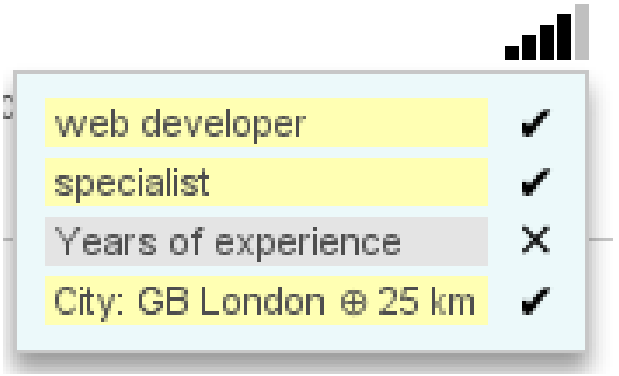


Figure 8. Expanded view of a result item



Query part scores widget, showing three matching query parts and one that does not match.

3.3 Tagging

Search! allows tagging of results. Tags are labels displayed on the search results and saved results. When enabled, any document found in Search! can be tagged to mark it for a later purpose. In Search! tags are projects. Tags are shared with other users by **sharing projects**.

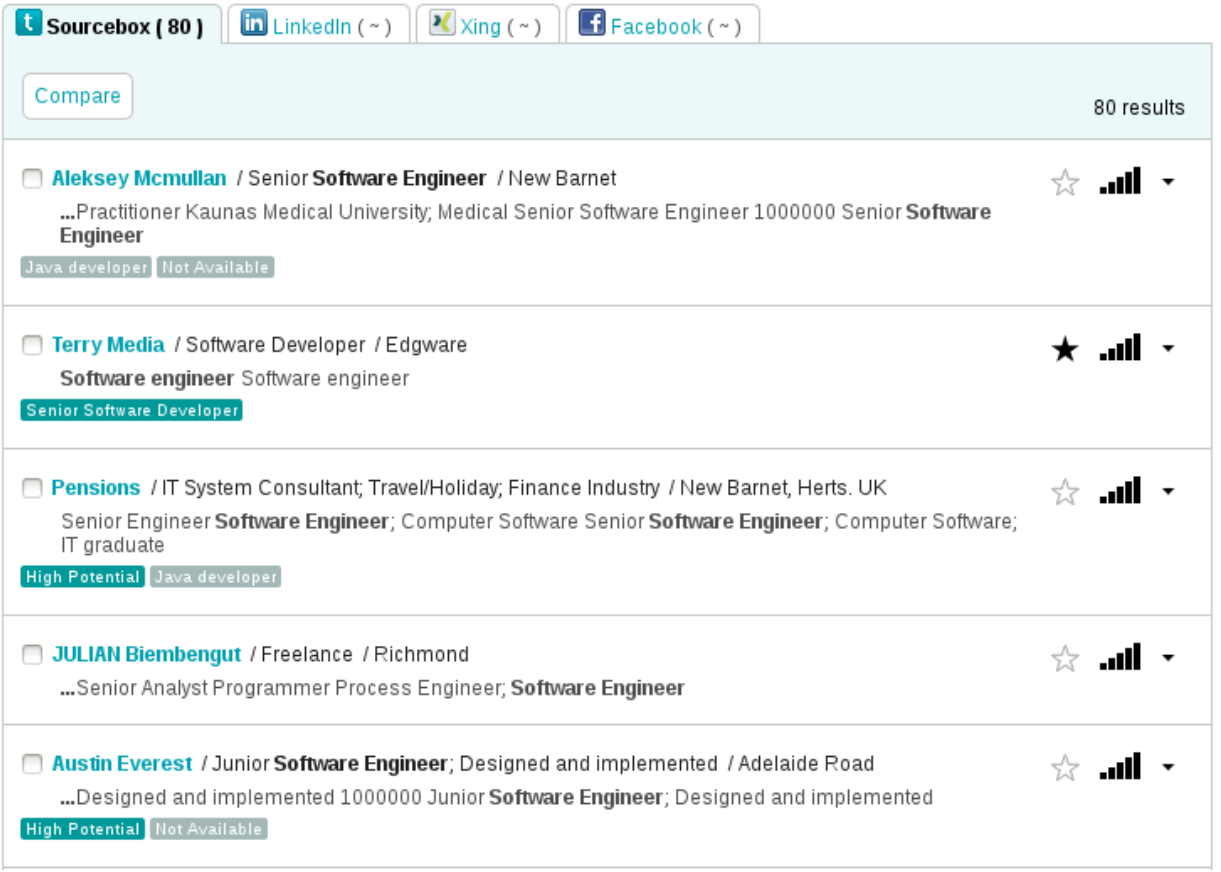


Figure 9. Tagged search results

When tagging is enabled, tags are displayed on the retrieved results, like in the figure above. The tags indicate to which project a results is saved to. Own projects are distinguished from shared projects by color.

To create a new tag, a project needs to be created with the desired name. Then saving a result will add the tag (see saving results in the **saving section**).

Clicking a tag brings up the saved results view for that project.

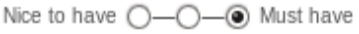
Also, the **projects facet** can be used to query by tags.

4 The Facet Bar

The Facet Bar provides a useful overview of the found search results. The results are categorized and the numbers next to the facet items show how many results belong to each category. The Facet Bar also allows to fine-tune the search results. Clicking on a certain facet item can restrict or rerank the results.

4.1 Facet Sections

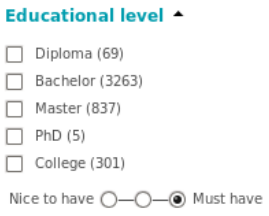
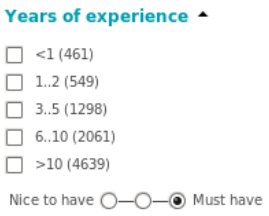
The Facet Bar is divided in different sections, each belonging to a different facet dimension. Each facet section has a name, e.g. "Education Level", describing the categorization dimension. The triangular symbol next to the facet name allows to fold or unfold the facet section for a better overview in the Facet bar. Furthermore, each facet section comes with radio buttons at the bottom to change the condition of its selections.

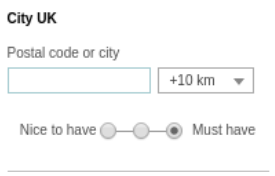


Currently Search! knows the following three conditions:

- "*nice to have*": items satisfying the selections get a higher ranking,
- "*should have*": items satisfying the selections get a considerably higher ranking,
- "*must have*": only items satisfying the selections are retrieved.

4.2 Different Facet Types

Facet Type	Description
	The fixed facet shows a constant number of categories and in paranthesis the number of result documents belonging to each category. Selecting a facet item will issue a new query where the new selection is taken into account.
	Facet categories can also be defined by ranges, either numeric values or dates. Selections on ranges have the same effect as selecting facet items in a fixed facet.

Facet Type	Description
	A location facet comes with a text field and a distance selector. Any input must be finished by pressing the RETURN key in the text field and then selecting from available distances. Alternatively, it is possible to select from available distances first and after that fill in location text field while it is under focus. Search! tries then to disambiguate the entered location description. If the disambiguation is successful, a single selection is shown below the text field. If the disambiguation cannot decide which location was meant, the facet displays all possible location selections, and the user can deselect unwanted locations. Changes in the distance selector will immediately update the search results.
	Facet sections (such as job titles) having too many different values cannot be displayed as a fixed facet. Alternatively Search! can present them in so-called **tag-clouds**. The tag-cloud facet enables to open and close its corresponding tag-cloud. All selections in the tag-cloud are displayed in the tag-cloud facet. Deselected items will disappear.

5 Tag-clouds

Figure 10. Tag-cloud: displays related terms and expressions.

Tag-clouds show characteristic related terms and expressions from the retrieved documents. A tag-cloud is related to a specific field of the document and shows only the keywords belonging to that specific field.

Tag-clouds provide suggestions to refine the query. By Selecting a tag-cloud keyword a new query is issued taking into account the new query refinement.

5.1 Interactions

	Description
	A tag-cloud gets closed by clicking the close button in the upper left corner.
	Tag-clouds also provide a text field to enter keywords not shown in the tag-cloud. This enables the user to express field-specific query constraints without using the query language. All input entered in the text field must be finished by pressing the RETURN key. If auto-suggestion is enabled, the text field will offer search term suggestions relevant to this field. The (+/-) switch before the text field input specifies whether the newly added term is a required or a banned part. This switch also influences the clicked terms from the tag-cloud. Depending on configuration, the text field input might be on the facet instead inside the tag-cloud, in which case only the (+/-) switch will be shown inside the tag-cloud.

6 Saving Projects, Searches and Results

Search! can save searches and results. All saving options can be configured to be on or off, so depending on the environment some features may be enabled while others are disabled. Saving uses the database to remember saved items.

It is also possible to remember searches without saving them to the database by using browser history and bookmarks:

6.1 Browser History

Search! is fully integrated with browser's Back and Forward buttons and as such allows navigation through user's searching history.
Searches can be saved to bookmarks, allowing opening favourite searches from within the user's bookmarks. A limitation of using the browser to save searches is that they can not be easily shared or accessed from different computers.

6.2 Saving Searches

To save a search, first create a query and then click the "Save search" button. The following modal screen will appear:

Figure 11. Save search dialog

Within this screen, a name for this search can be entered and a project for this search can be chosen. If "New project" has been selected, a new project will be created with the same name as the search. To finalize the saving, either click "Save" button or press the Return key on your keyboard.
It is possible to disable project management for a Search! installation, in which case the Project drop-down menu will not appear.

6.3 Saving Results

To save a single result, click the empty star symbol on the right hand side of the result. To save multiple results at the same time, tick the checkboxes next to desired results and then

click "Save" button. The star will turn full as a confirmation of a successful save. If there is no active project at the moment of saving, a pop-up window will appear within which a desired project can be chosen.
To remove a previously saved result, the procedure is the same - either click the full star symbol or select multiple items and then click "Remove" button.

Figure 12. Saving multiple results

6.4 Project management

Projects are collaborative activities aimed at fulfilling a certain aim, such as finding a candidate for a certain vacancy. Projects as such can contain multiple Saved Searches and multiple Saved Results (contained in a single result list). Depending on the configuration, any of these aspects can be either enabled or disabled.

If enabled, clicking the projects button will show the following modal screen:

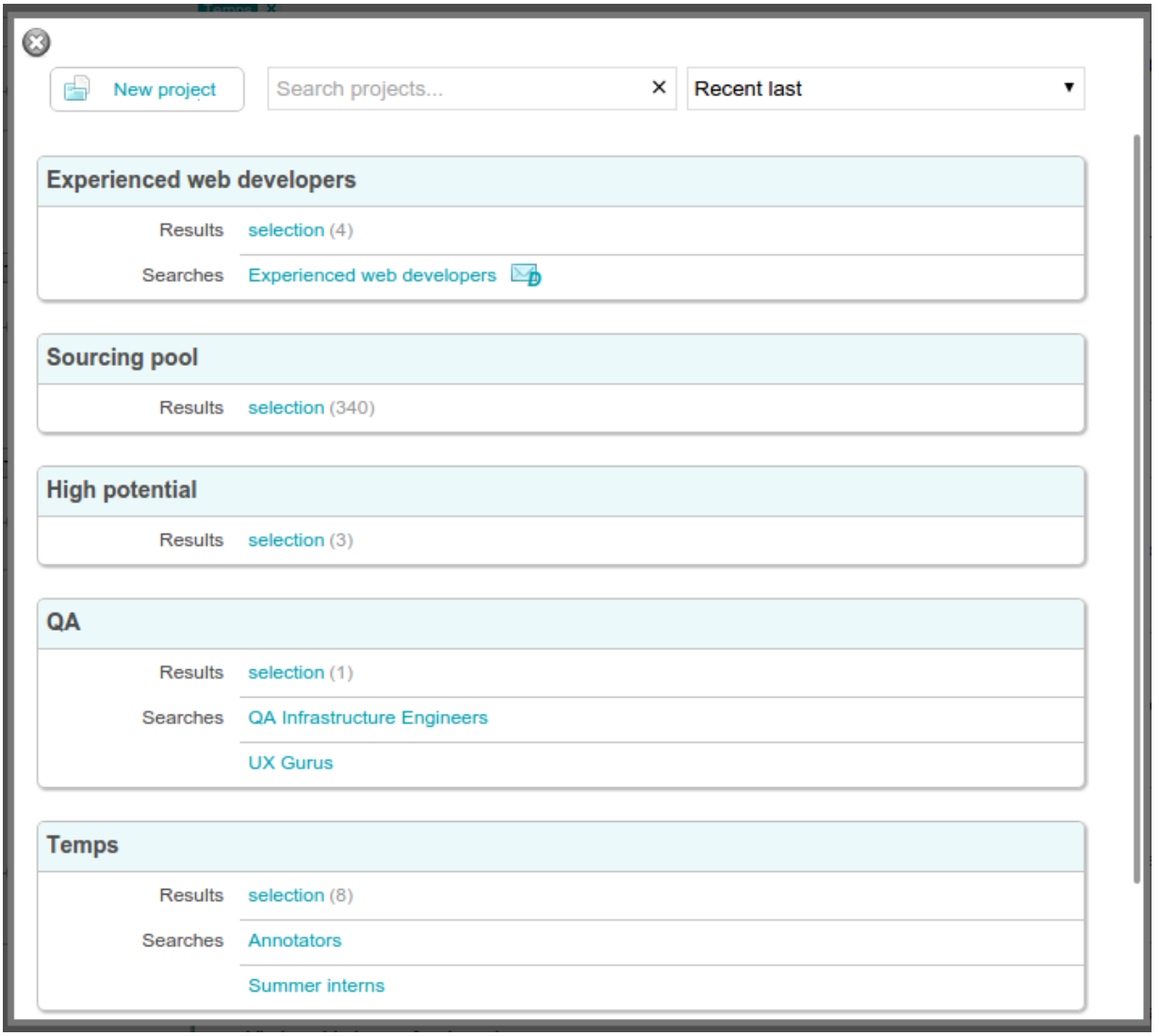
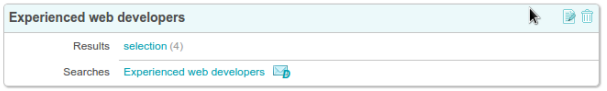
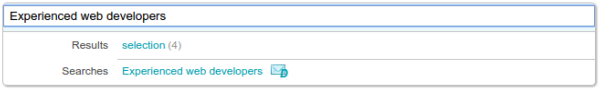


Figure 13. Projects modal

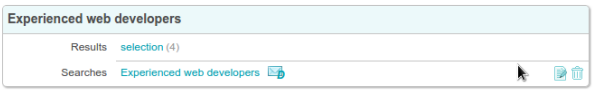
To restore a saved search state, click a name in the "Searches" list.
To view a list of saved results for this project, click the "selection" link.
To just set a project as currently active, click its name.
An envelope icon is shown next to the saved searches which have e-mail alerting configured.
Hovering the project name shows manipulation controls on the right hand side of the header:



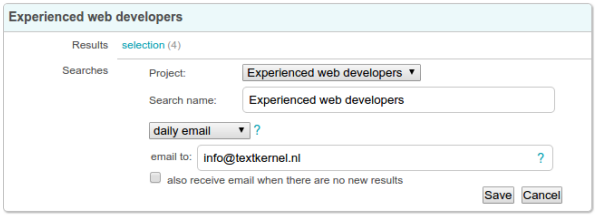
To delete a project, click the bin icon on the right hand side of the header.
To rename a project, click the edit icon. The project name will be replaced by a text box in which a new name can be entered.



To confirm the name change, press Return key on your keyboard or just click outside the text box.
Similarly, hovering a saved search row shows manipulation controls on the right hand side of the list:



To delete a saved search, click the bin icon on the right hand side of the list.
To edit a saved search, first click the edit icon. Following edit fields will appear:



If the current Search! installation has email alerts enabled, additional fields will be shown for scheduling an email alert. These fields allow selection of email frequency (never, daily, weekly, monthly), desired recepients, as well as the option to send an alert if there are no new matches for the query in the selected period.

Desired recipients can be entered manually by entering multiple, comma-separated addresses; or, when Search! is configured with both **project sharing** as well as an email-address service enabled, by using a drop-down menu to select the persons or roles that will receive the email alert.
Daily e-mails are sent every morning at 06:00 and include new results that have been indexed within the last day. Weekly e-mails are sent every monday morning at 06:00 and include results that have been indexed within the last week. Montly e-mails are sent at 06:00 of every first day of the month, and they include new results that have been indexed within the last month.
After modifying the data, either click the Save button, or press the Return key on your keyboard.

In case you have many projects and need a better way to grasp them, it is possible to sort them either by name (A-Z or Z-A) or by creation date (newest first or oldest first)

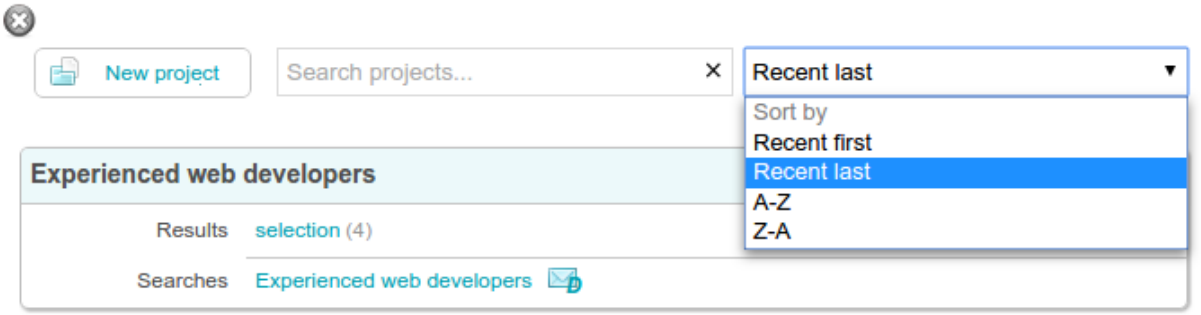


Figure 14. Sorting projects

It is also possible to filter the projects list. Filtering is done on partial match on the name (e.g. "man" will match both "manager" and "Germany"). Both filtering and sorting can be used at the same time.

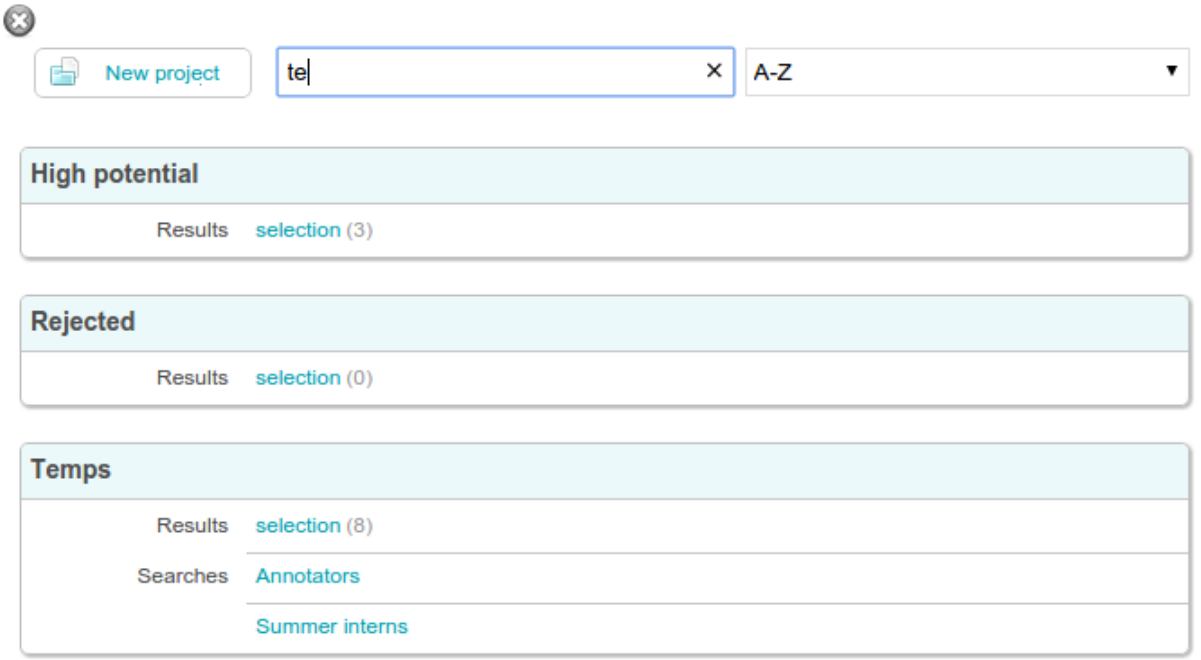


Figure 15. Searching in projects (filtering)

6.4.1 Searching by project

Figure 16. Project facet

It is possible to search by project with the projects facet. Like other search criteria, projects can be queried as "nice to have", "should have", "must have" or "banned" mode. This way results can be filtered by project, or a preference for certain projects can be expressed. Auto-completion will help finding existing project names.

6.5 Searches management

Figure 17.

In case project management has been disabled for this Search! installation, yet saving searches is enabled, a reduced version of the projects management modal can be found under the "Searches" button.

6.6 Sharing projects

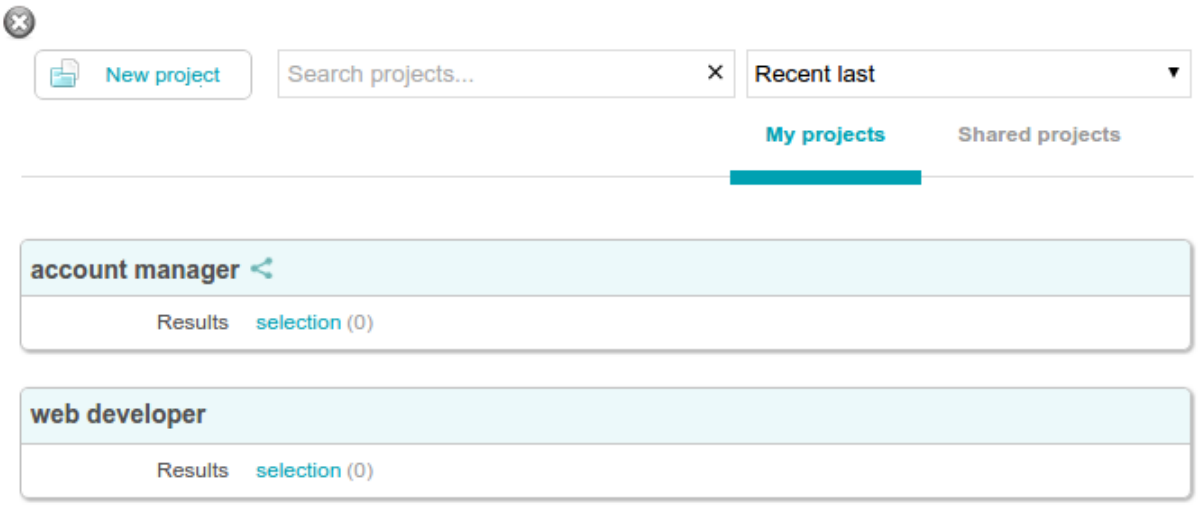


Figure 18.

If the current Search! installation is configured to allow sharing of documents among users, the project overview will look slightly different as shown above. It is then possible to switch between the list of owned projects, and the list of projects shared by others, by clicking on the respective tabs.

The project list of owned projects also indicates for each project whether it is shared with others or not. A sharing symbol appears on the left side of the project name if the project is share.

A similar sharing symbol appears in the projects editing options on the right side of the project name showing up on hover. Clicking the symbol opens a project sharing dialog which allows to set with whome the current project should be shared.

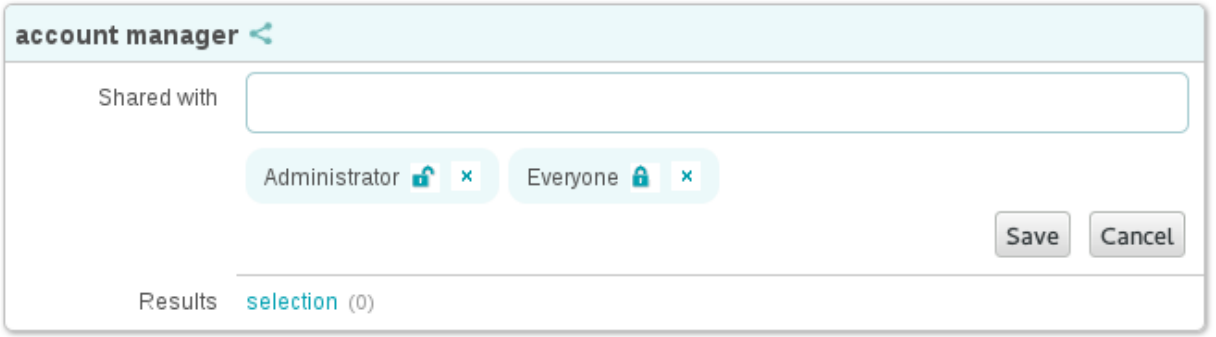


Figure 19.

The project sharing dialog comes with a text-field to enter the persons or roles that are allowed to see and optionally edit the current project. Auto-completion will help to see which other users or roles are eligble for sharing.

Once a person or role is selected, it shows up as a new share in a small box below summarizing the current sharing settings. Each single share can be removed here by clicking the closing symbol, and switched between read-only and read-write access.

Read-only access allows a user to see the project, its saved searches and results, without being able to edit them. Read-write access will also allow a user to add or remove saved searches and results. It is in any case not possible to rename or delete other people's projects.

All changes made in this dialog only become effective once the save button has been clicked.

7 Comparison view

Search! offers an easy comparison view of chosen search results, combining the results' metadata and the information about which parts of the query were matched in these results.

If the account is configured to have comparison view enabled, checkboxes will appear next to the search results and a "Compare candidates" (or equivalent) button will appear above the results list.

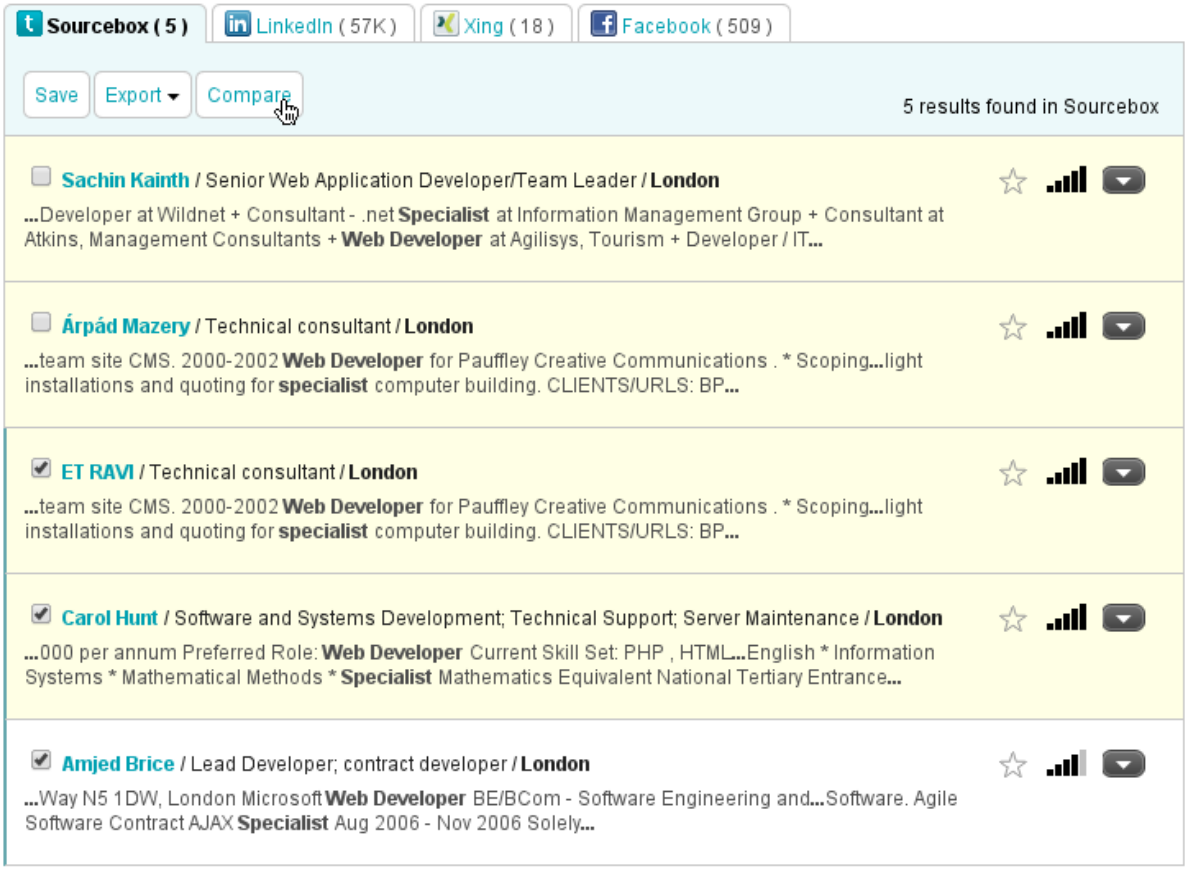
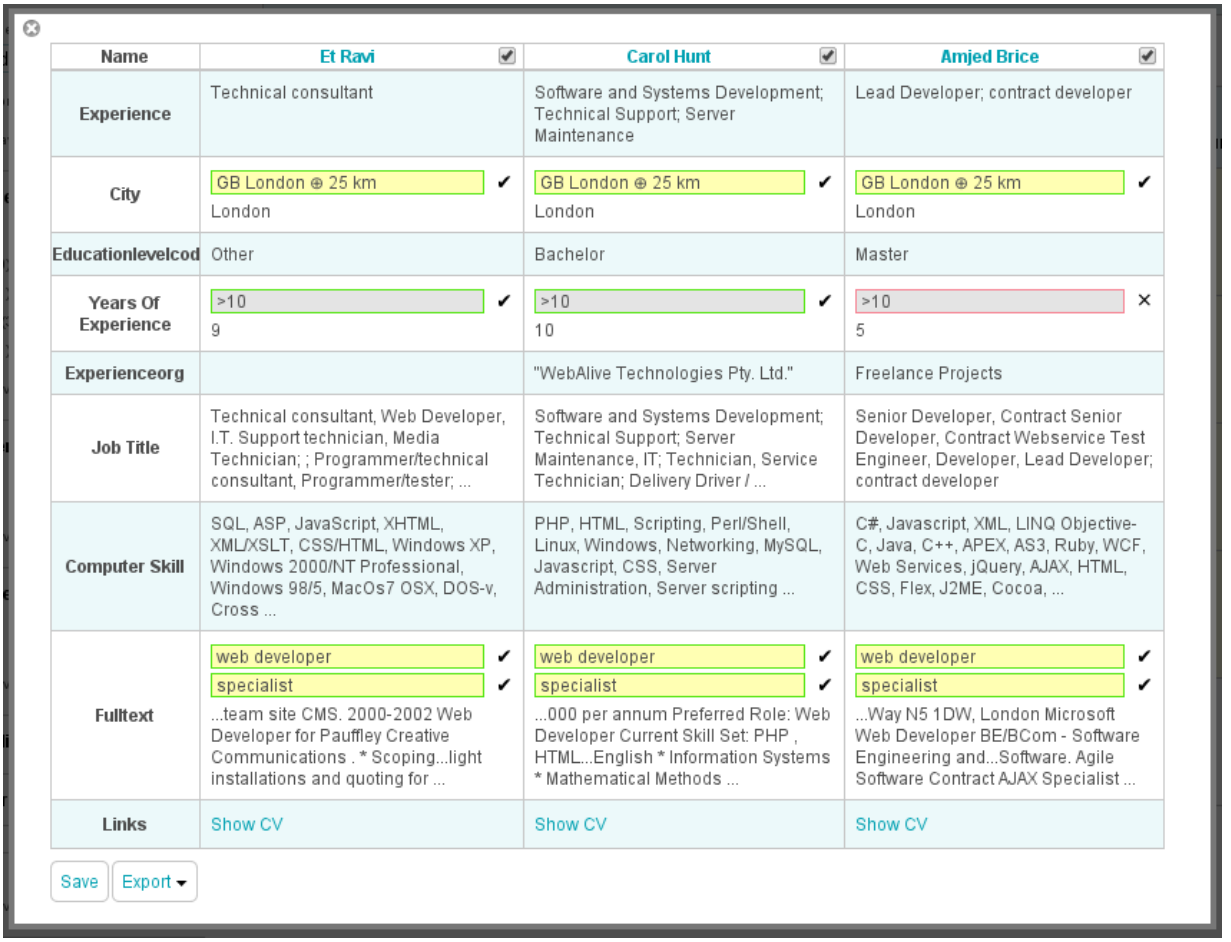


Figure 20. Account with the comparison view enabled

Selecting the desired results and clicking "Compare candidates" opens the comparison view. This view offers a tabular representation of both the selected results' metadata and matching query parts.



Name	Et Ravi	Carol Hunt	Amjed Brice
Experience	Technical consultant	Software and Systems Development; Technical Support; Server Maintenance	Lead Developer; contract developer
City	GB London ⊕ 25 km ✔ London	GB London ⊕ 25 km ✔ London	GB London ⊕ 25 km ✔ London
Educationlevelcod	Other	Bachelor	Master
Years Of Experience	>10 ✔ 9	>10 ✔ 10	>10 ✕ 5
Experienceorg		"WebAlive Technologies Pty. Ltd."	Freelance Projects
Job Title	Technical consultant, Web Developer, I.T. Support technician, Media Technician; ; Programmer/technical consultant, Programmer/tester; ...	Software and Systems Development; Technical Support; Server Maintenance, IT; Technician, Service Technician; Delivery Driver / ...	Senior Developer, Contract Senior Developer, Contract Webservice Test Engineer, Developer, Lead Developer; contract developer
Computer Skill	SQL, ASP, JavaScript, XHTML, XML/XSLT, CSS/HTML, Windows XP, Windows 2000/NT Professional, Windows 98/5, MacOs7 OSX, DOS-v, Cross ...	PHP, HTML, Scripting, Perl/Shell, Linux, Windows, Networking, MySQL, Javascript, CSS, Server Administration, Server scripting ...	C#, Javascript, XML, LINQ Objective-C, Java, C++, APEX, AS3, Ruby, WCF, Web Services, jQuery, AJAX, HTML, CSS, Flex, J2ME, Cocoa, ...
Fulltext	web developer ✔ specialist ✔ ...team site CMS. 2000-2002 Web Developer for Pauffley Creative Communications . * Scoping...light installations and quoting for ...	web developer ✔ specialist ✔ ...000 per annum Preferred Role: Web Developer Current Skill Set: PHP , HTML...English * Information Systems * Mathematical Methods ...	web developer ✔ specialist ✔ ...Way N5 1DW, London Microsoft Web Developer BE/BCom - Software Engineering and...Software. Agile Software Contract AJAX Specialist ...
Links	Show CV	Show CV	Show CV

Save | Export

Figure 21. Comparison view

For comparison convenience, it is possible to re-order the columns by dragging the column header. A vertical guideline shows where the column will be positioned when dropped.

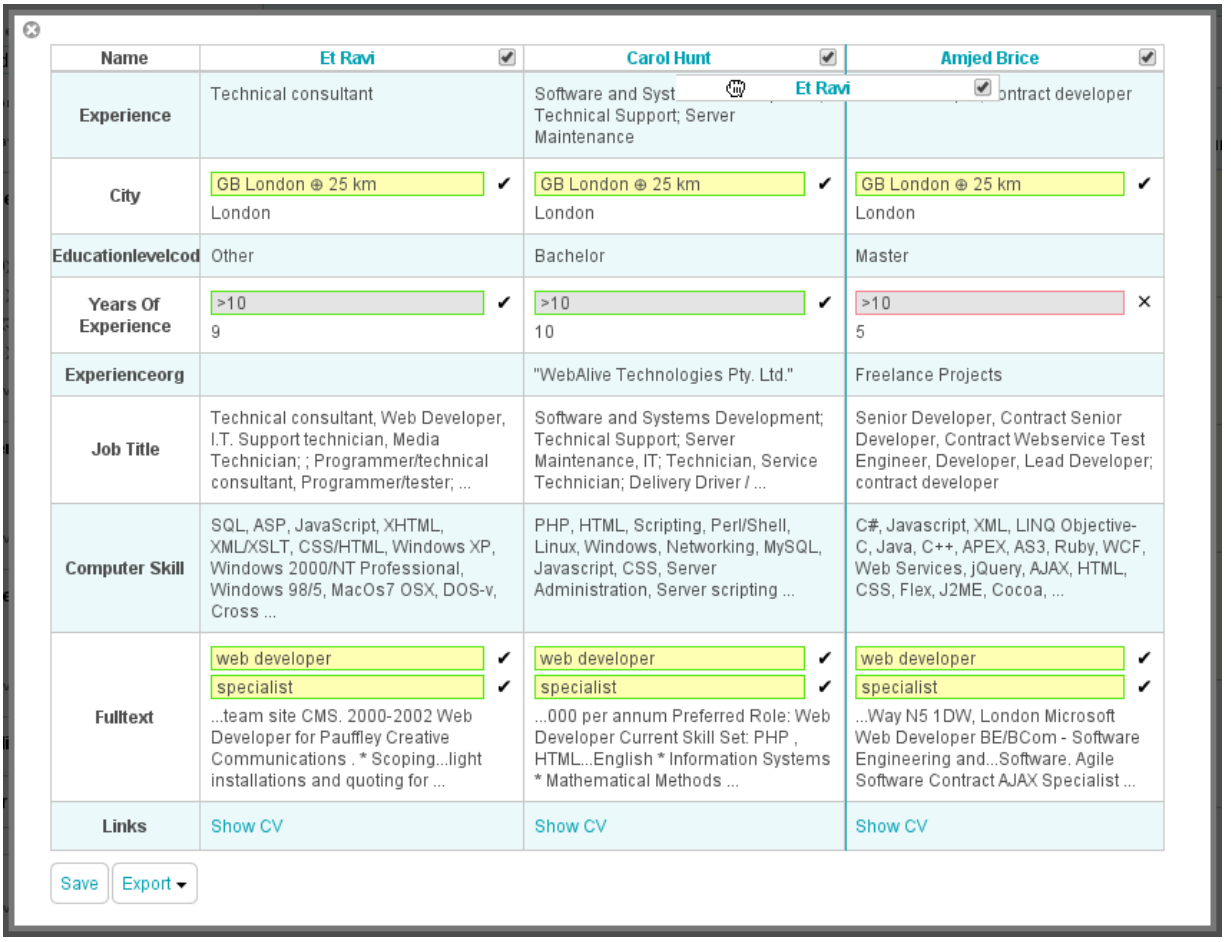


Figure 22. Column re-ordering

Some query parts and metadata fields can be too long to fit into their containers. To achieve a consistent look, they are truncated and followed by an ellipsis. Hovering a truncated query part or a metadata field shows a tooltip containing the full, non-truncated version of the text.

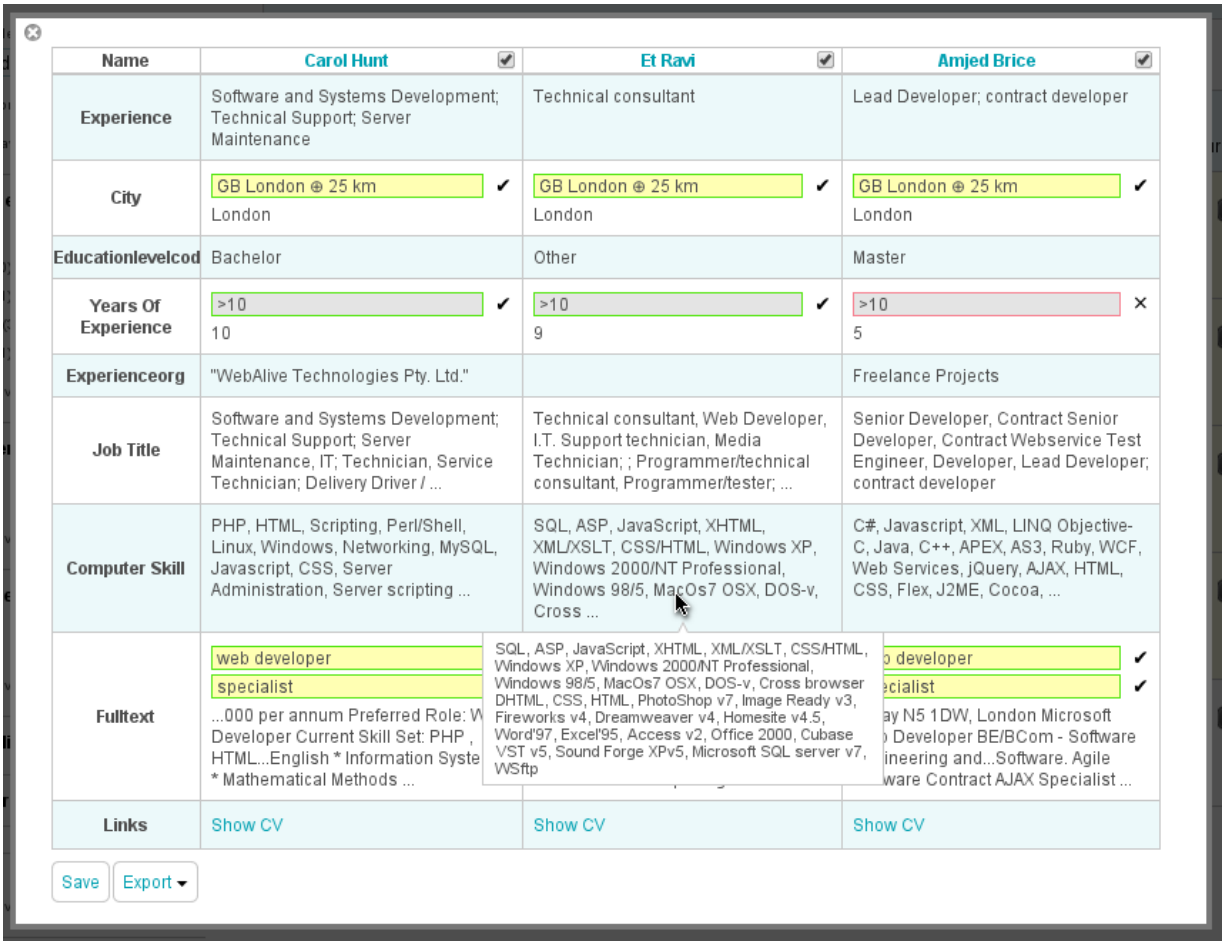


Figure 23. Query part tooltip

8 Query Language for Advanced Users

8.1 Keywords

```
Java developer Amsterdam
```

Keywords are *not* case sensitive, that is "Java", "JAVA" and "java" are the same.

Adding keywords to a query further limits the result set. In the example above, adding Amsterdam will return results containing Amsterdam in addition to Java and Developer. An *empty query*, that is without any keywords, returns all.

Keywords containing special characters must be quoted as a phrase, such as "page#3". However, Search! recognizes most common keywords like C++ that do not need quotes.

8.2 Phrases

```
"Java developer" Amsterdam
```

Phrases enclosed by quotes are used to match a sequence of words. Special characters are ignored inside a phrase.

8.3 Proximity

```
[Java developer] Amsterdam
[developer C++ Java]
```

Query terms within [] match if they occur in the document in any order, possibly with one or two words in between. This is called proximity matching. Proximity matching is more flexible than phrases, e.g. the above query [Java developer] also matches documents containing "java software developer" and "java enterprise software developer". Proximity queries can only contain simple keywords, but not phrases. Quoting inside proximity queries will therefore be ignored.

8.4 Wildcard

```
develop*
```

Search! allows to perform wildcard queries. If a query term ends with a trailing * it gets expanded to the most common completions. Documents containing any of the found completions

will match. For example, the above query develop* will match on developer, development, developed, etc. A few restrictions apply:

- the * symbol can only be used at the end of a query term
- the * symbol needs to be preceded by at least 2 characters
- wildcard terms cannot be part of phrases of proximity expressions

8.5 Weighting

```
developer [software engineer] #1.5 "software architect" #0.8
```

Terms are weighted within a query by adding a number weight behind the term. If no weight is specified, a term receives a default weight of #1.0. The weight influences the ranking of documents of the weighted query expression in relation to the other expressions of the same field. Weights can also be assigned to more complex query expressions, such as phrases, proximity conditions or range queries. In the example above [software engineer] contributes 1.5 times as much to the result score as normal and "software architect" 0.8 times as much. Weights only influence the relative importance among query parts belonging to the same field. Example:

```
developer experience:[software engineer] #1.5 "software architect" #0.8
```

The weight of 1.5 given to [software engineer] will be ignored in this case. Since [software engineer] is the only query expression on the experience field, there is no other query expression where the weight would relate to. Also, weights inside an OR-combination are ignored.

8.6 Fields

```
Java developer city:Amsterdam
experience:[Java developer] Amsterdam
```

If a term, phrase or proximity expression should be matched on a certain metadata field or section of a document, the query expression needs to be preceded by the field name and a colon. The above query searches for the word Amsterdam only in the city field of the document, respectively for java developer only in the experience section.

8.7 Numeric / Date Range Conditions

```
Java developer experience:>2
Java developer experience:5..10
Java developer date:<2010-10-01
```

Numeric or date (range) conditions are expressed by "<",">","=" and "..". Dates must be of yyyy-mm-dd format. Numeric or date conditions are always assigned to a specified field.

8.8 Location Conditions

```
Java developer location:Amsterdam+20
Java developer location:1018+20
Java developer location:"New York"+25
location:"33.14 -12.25"+50
location:"33.14 -12.25"
```

Location (+radius) conditions are expressed by naming a location (city name, postal code, geo coordinates) followed by a radius in kilometers after the "+" symbol. If the radius is omitted only exact matches are returned.

8.9 No-Value Search

```
skills:--
```

It is possible to search explicitly for documents having no value indexed for a certain field. This is especially useful in OR-combination with other values to include documents having no value as well.

It is not possible to perform a no-value search on the full-text field.

8.10 Nice-to-have Expressions

```
"Java developer" %javascript
```

Nice-to-have search expressions are marked with a preceding %. Instead of limiting the result set to the searched term, they only modify the ranking of the search results. The example query above will return all Java developers, with or without javascript in their CV, but the ones that also contain javascript will be ranked higher in the results.

If a query consists solely of nice-to-have expressions, than the result set is the same as returned by the empty query: all documents are returned, ranked by the nice-to-have expressions.

8.11 Should-have Expressions

```
"Java developer" ^javascript
```

Should-have search expressions are marked with a preceding ^. Their behavior is identical to Nice-to-have Expressions except that they carry more weight when modifying the ranking.

8.12 Banned Expressions

```
Java developer -city:Amsterdam
```

Query expressions marked by a preceding - are banned expressions, they exclude all documents matching this expression from the result. It is sometimes easier to filter out documents containing a certain term than to list all positive examples.

8.13 OR Groups / Specifying Synonyms

```
compskills:(C C++) Amsterdam
([java developer] [software developer]) Amsterdam
```

Alternative search terms can be combined in an OR group by surrounding the expressions in brackets (). This is useful for specifying a list of synonyms or to allow multiple required facet items. In the example above, capturing both C and C++ in an OR-group ensures that all documents contain either C or C++ as a computer skills as well as containing Amsterdam. The second example shows that weights can be used within an OR-group. OR-groups may be used on the full text (second example) or on a specific field (first example), they may not be nested.

8.14 Field Groups

```
compskills:|(C C++) html|
compskills:|java %jquery|
```

Different query parts on the same field can be grouped using pipe characters (||). This is useful to prevent repetition of a field name, but other than that the two queries 'compskills:java compskills:jquery' and 'compskills:|java jquery|' are equivalent. Within the pipe characters a limited subset of the query language can be used: OR groups, conditions, terms, phrases, proximity expressions are allowed, however no field names can be used. All conditions within the field group overrule the group condition, except for the default condition which does not overrule the group condition. Query part weights are allowed within the field group but it is not possible to assign a weight to the field group itself.

8.15 Boolean Expressions

```
developer java OR .net
Perl AND C++ NOT Java
```

The query language has a limited support for commonly used Boolean expressions. The above examples are interpreted as expected, respectively searching for a document mentioning the keywords 'developer' and either 'java' or '.net', and a document mentioning the keywords 'Perl' and 'C++' but not the keyword 'Java'.

Boolean operators are only interpreted in simple full-text queries. If the query contains any other above mentioned query syntax, such as field names or condition operators, boolean operators won't be recognized.