Weill Cornell
Medicine

Improving Drupal search experience with Apache Solr and Elasticsearch

Milos Pumpalovic – Web Front-end Developer
Gene Mohr – Web Back-end Developer

About Us

Milos Pumpalovic

- Front End Developer
- Drupal theming and module development
- mip2047@med.cornell.edu

Gene Mohr

- Back End Developer
- Full stack and Drupal module development
- bas2019@med.cornell.edu

What is search engine and why we use it?

Search engine in Drupal

What are the differences from Drupal Search API?

Why we use it?

What is Solr?

Solr is a super fast, open source, standalone search server built on Apache Lucene.

Some of the features are:

- Advanced Full-Text Search
- Designed for High Volume traffic
- Faceted Search and Filtering
- Scalable, flexible, extensible
- Rich Document Parsing – built-in ability to index PDF's, Word documents and more
- Multiple search indexes
- Query Suggestions, Spelling and More – advanced capability for auto-complete, spell checking, highlighting and more

Apache Solr Configuration

- Install module
- Create an index on Solr engine
- Create a view
- Attach facet filters

Install the Drupal modules

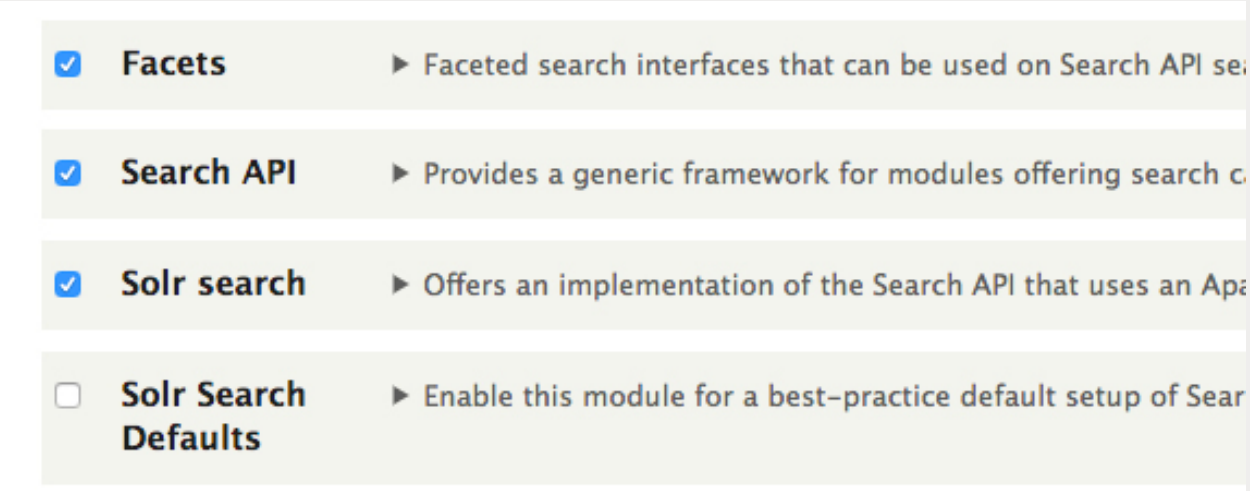
In Drupal 8, Composer is used to manage PHP libraries. The Search API module doesn’t include Solarium, so dependencies need to be set via Composer with the following commands:

```
$ composer config repositories.drupal composer https://packages.drupal.org/8
```

Download the module Search API Solr Search.

```
$ composer require drupal/search_api_solr
```

Install the following modules:



☑ **Facets** ▸ Faceted search interfaces that can be used on Search API se

☑ **Search API** ▸ Provides a generic framework for modules offering search c

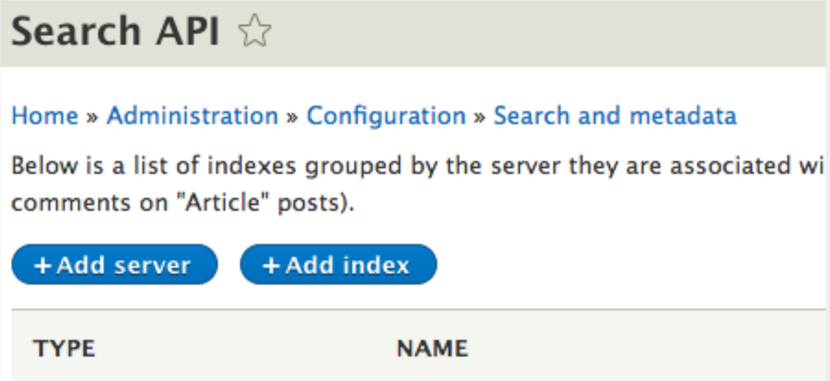
☑ **Solr search** ▸ Offers an implementation of the Search API that uses an Apa

□ **Solr Search Defaults** ▸ Enable this module for a best-practice default setup of Sear

Optional: install Solr Search Defaults module

Add Solr server

Go to Configuration -> Search and metadata -> Search API.
Click “Add Server” and configure the Server.
Server name example: Local Solr Server

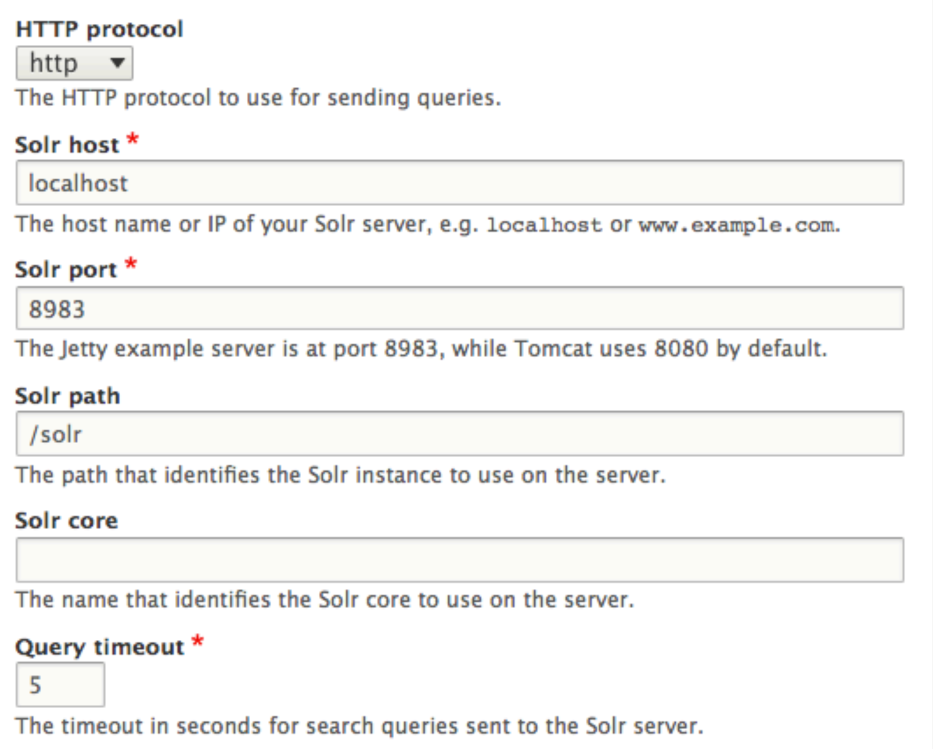


Search API ☆

Home » Administration » Configuration » Search and metadata

Below is a list of indexes grouped by the server they are associated wi
comments on "Article" posts).

TYPE	NAME



HTTP protocol
http ▾
The HTTP protocol to use for sending queries.

Solr host *
localhost
The host name or IP of your Solr server, e.g. `localhost` or `www.example.com`.

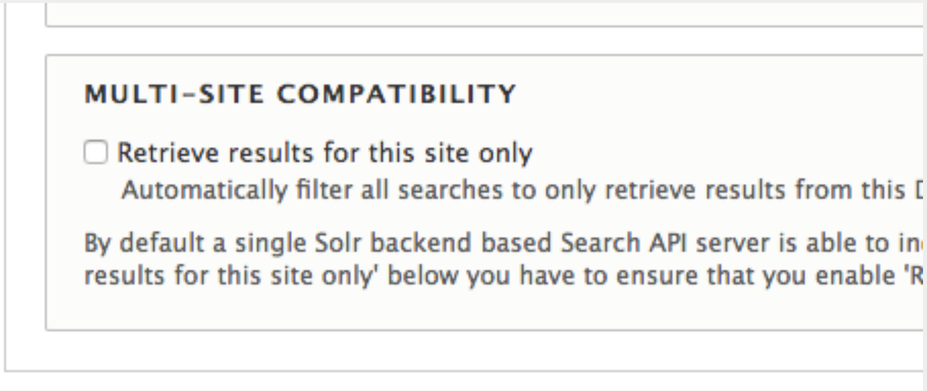
Solr port *
8983
The Jetty example server is at port 8983, while Tomcat uses 8080 by default.

Solr path
/solr
The path that identifies the Solr instance to use on the server.

Solr core

The name that identifies the Solr core to use on the server.

Query timeout *
5
The timeout in seconds for search queries sent to the Solr server.



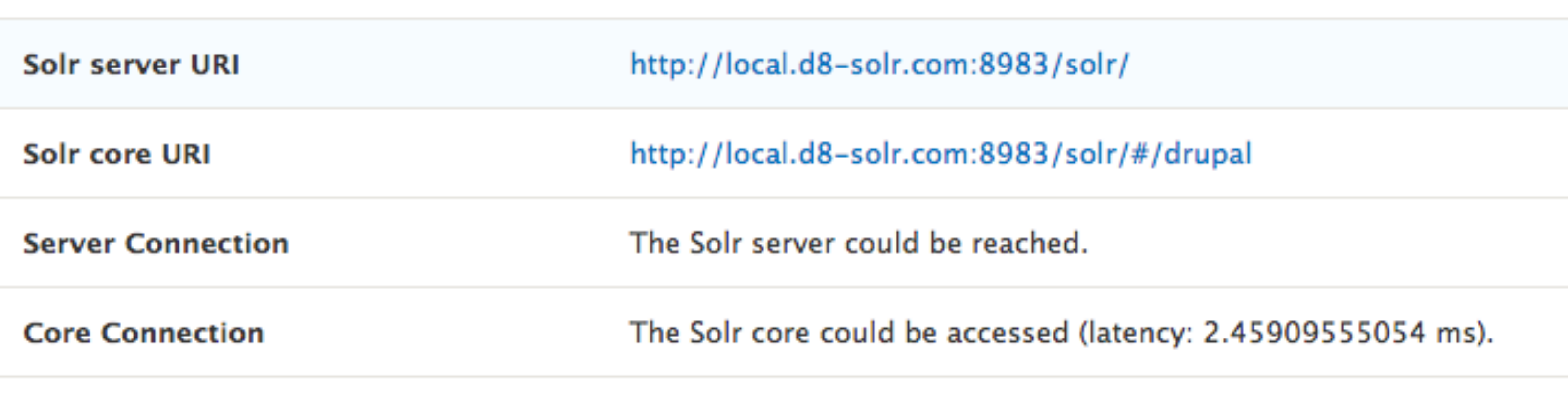
MULTI-SITE COMPATIBILITY

☐ Retrieve results for this site only
Automatically filter all searches to only retrieve results from this D

By default a single Solr backend based Search API server is able to in
results for this site only' below you have to ensure that you enable 'R

Confirm Solr server is connected

At this stage, the server's status page should show message:
"The Solr server could be reached" and "The Solr core could be accessed"

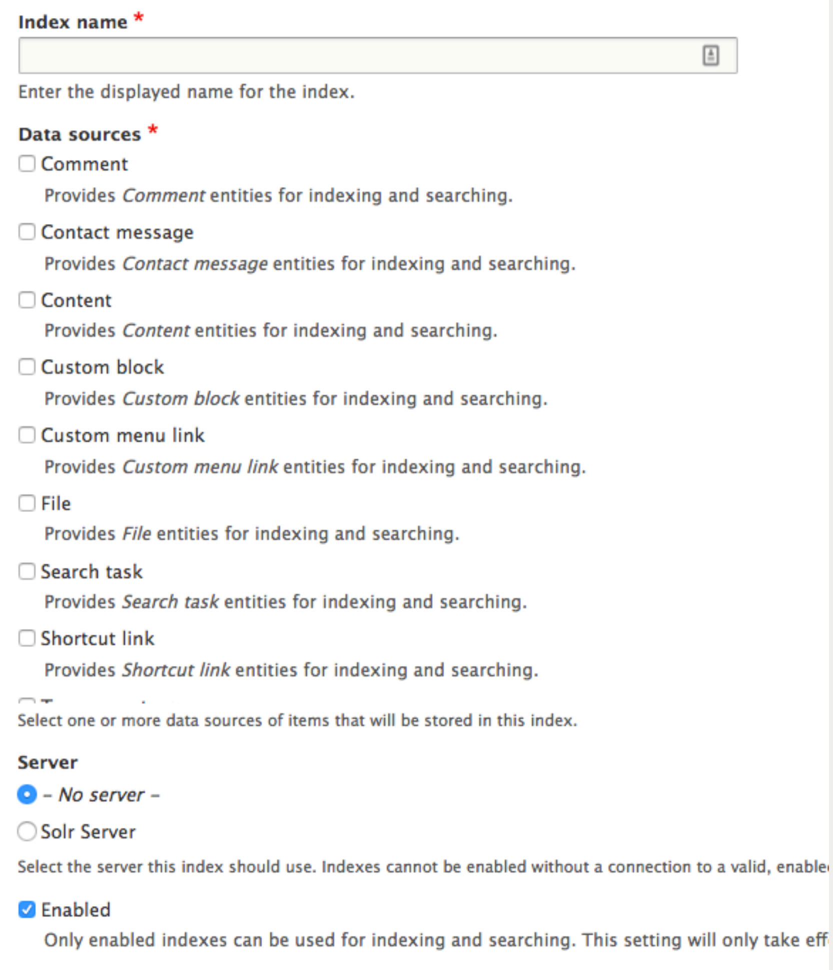


Solr server URI	http://local.d8-solr.com:8983/solr/
Solr core URI	http://local.d8-solr.com:8983/solr/#/drupal
Server Connection	The Solr server could be reached.
Core Connection	The Solr core could be accessed (latency: 2.45909555054 ms).

If Solr Search Defaults module was installed, it can be uninstalled as it's no longer required.

Add search index

Go to Configuration -> Search and metadata -> Search API.
Click "Add Index"



Index name *

Enter the displayed name for the index.

Data sources *

□ Comment
Provides *Comment* entities for indexing and searching.

□ Contact message
Provides *Contact message* entities for indexing and searching.

□ Content
Provides *Content* entities for indexing and searching.

□ Custom block
Provides *Custom block* entities for indexing and searching.

□ Custom menu link
Provides *Custom menu link* entities for indexing and searching.

□ File
Provides *File* entities for indexing and searching.

□ Search task
Provides *Search task* entities for indexing and searching.

□ Shortcut link
Provides *Shortcut link* entities for indexing and searching.

Select one or more data sources of items that will be stored in this index.

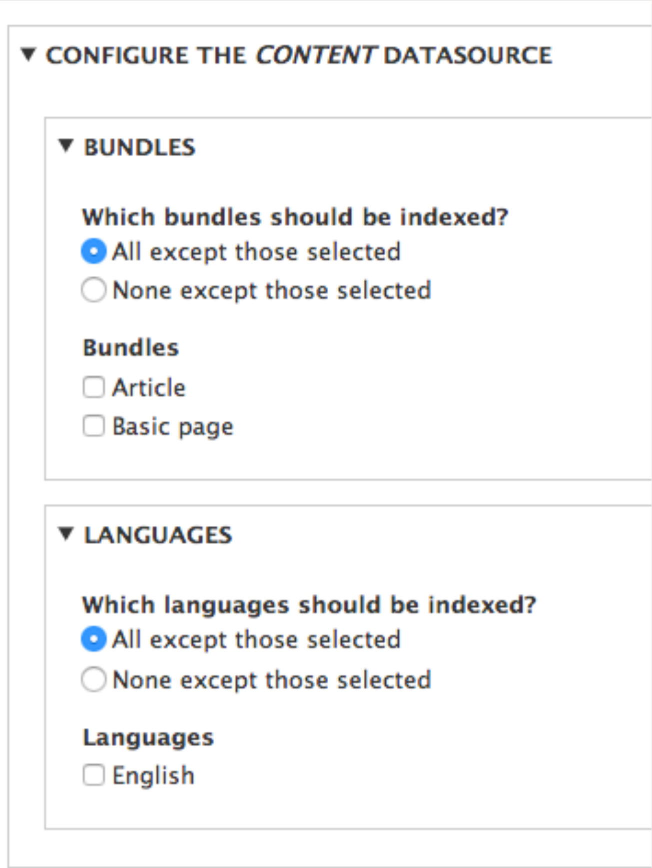
Server

● – *No server* –
○ Solr Server

Select the server this index should use. Indexes cannot be enabled without a connection to a valid, enable

☑ Enabled
Only enabled indexes can be used for indexing and searching. This setting will only take eff

Selecting the "Content" data source, options are presented to select which bundles are to be indexed



▼ CONFIGURE THE *CONTENT* DATASOURCE

▼ BUNDLES

Which bundles should be indexed?
● All except those selected
○ None except those selected

Bundles
□ Article
□ Basic page

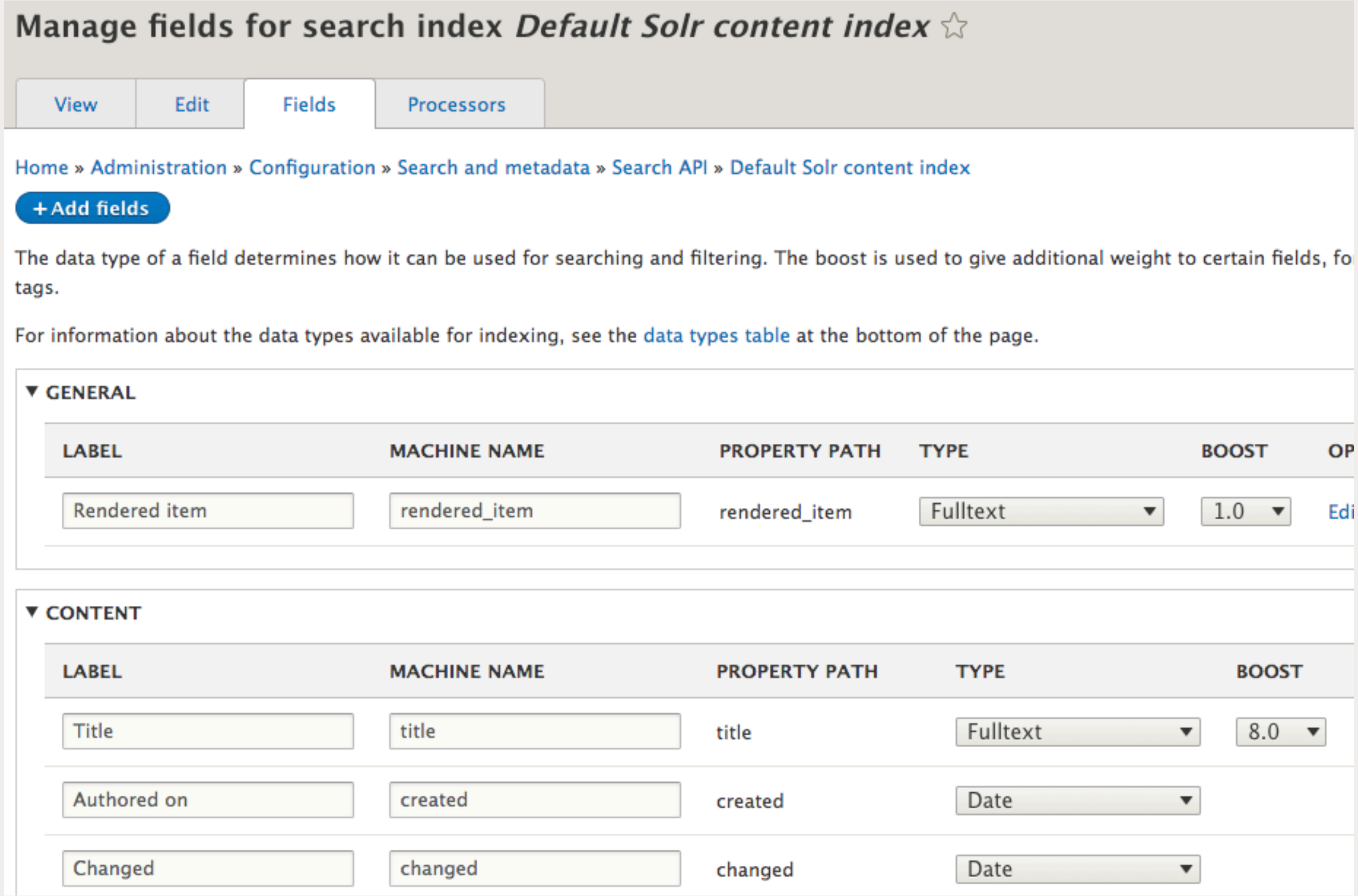
▼ LANGUAGES

Which languages should be indexed?
● All except those selected
○ None except those selected

Languages
□ English

Add fields to index

Before search can be performed, select all the fields that should be available to search. That is configured in the “Fields” tab.



Manage fields for search index *Default Solr content index* ☆

Home » Administration » Configuration » Search and metadata » Search API » Default Solr content index

The data type of a field determines how it can be used for searching and filtering. The boost is used to give additional weight to certain fields, fo tags.

For information about the data types available for indexing, see the data types table at the bottom of the page.

▼ GENERAL

LABEL	MACHINE NAME	PROPERTY PATH	TYPE	BOOST	OP
Rendered item	rendered_item	rendered_item	Fulltext ▾	1.0 ▾	Edi

▼ CONTENT

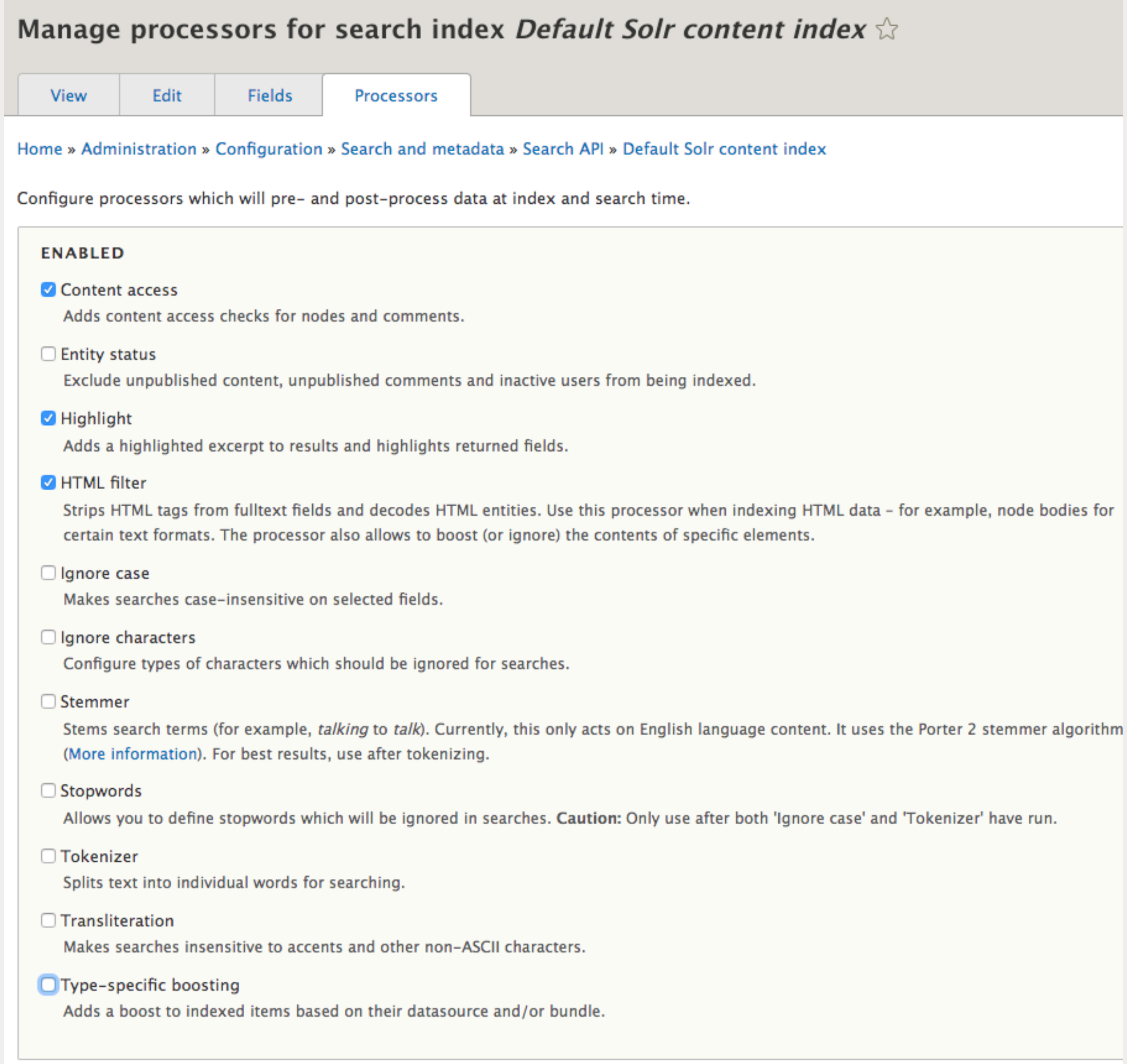
LABEL	MACHINE NAME	PROPERTY PATH	TYPE	BOOST
Title	title	title	Fulltext ▾	8.0 ▾
Authored on	created	created	Date ▾	
Changed	changed	changed	Date ▾	

Add processors to the index

Last step is to add additional 'processors'.

This includes items such as:

- Content access
- Ignore case (case-insensitive search)
- Tokenizer (split into individual words)



Manage processors for search index *Default Solr content index* ☆

Home » Administration » Configuration » Search and metadata » Search API » Default Solr content index

Configure processors which will pre- and post-process data at index and search time.

ENABLED

☑ Content access
Adds content access checks for nodes and comments.

□ Entity status
Exclude unpublished content, unpublished comments and inactive users from being indexed.

☑ Highlight
Adds a highlighted excerpt to results and highlights returned fields.

☑ HTML filter
Strips HTML tags from fulltext fields and decodes HTML entities. Use this processor when indexing HTML data – for example, node bodies for certain text formats. The processor also allows to boost (or ignore) the contents of specific elements.

□ Ignore case
Makes searches case-insensitive on selected fields.

□ Ignore characters
Configure types of characters which should be ignored for searches.

□ Stemmer
Stems search terms (for example, *talking* to *talk*). Currently, this only acts on English language content. It uses the Porter 2 stemmer algorithm (More information). For best results, use after tokenizing.

□ Stopwords
Allows you to define stopwords which will be ignored in searches. **Caution:** Only use after both 'Ignore case' and 'Tokenizer' have run.

□ Tokenizer
Splits text into individual words for searching.

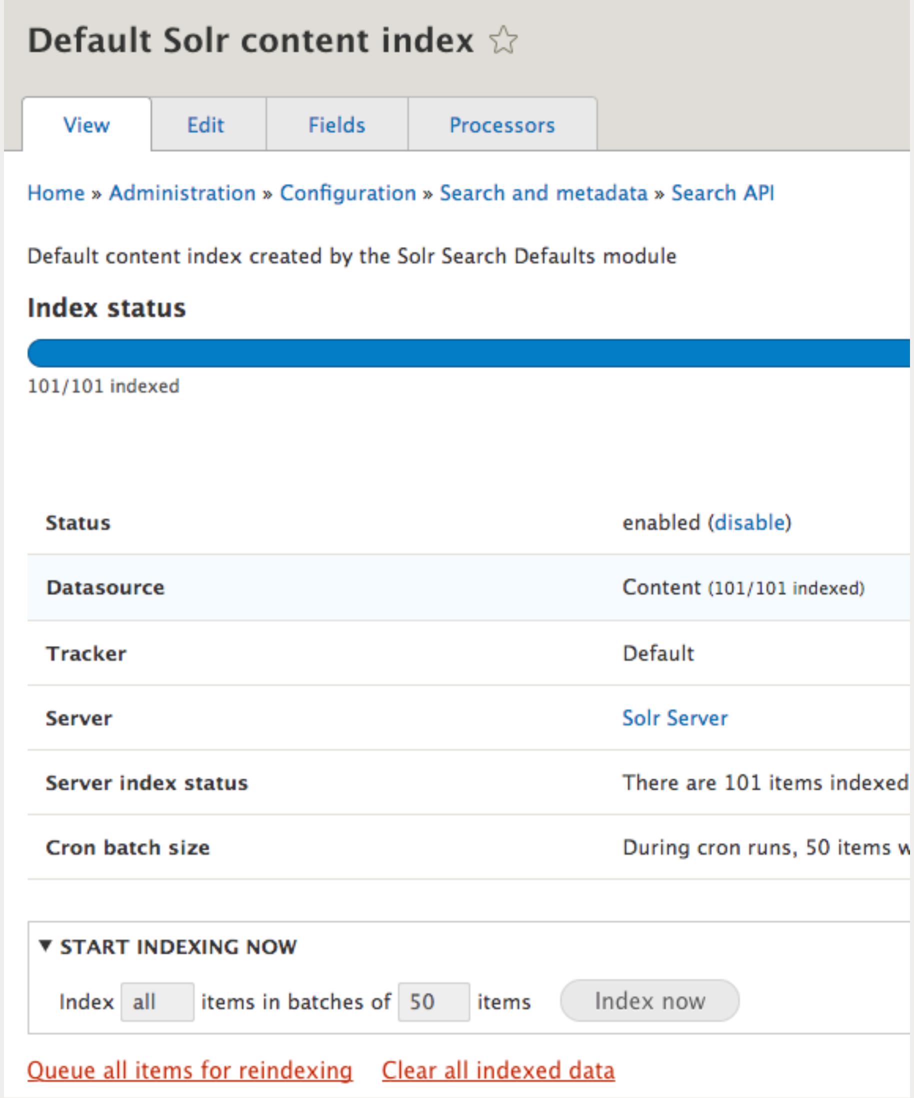
□ Transliteration
Makes searches insensitive to accents and other non-ASCII characters.

□ Type-specific boosting
Adds a boost to indexed items based on their datasource and/or bundle.

Verify search index is working

Once fields and processors are setup, going back to the ”View” tab, will show the status of the index, and at this point, the content is ready to be indexed if not already set to index immediately when the index is created.

Indexing of content is done via cron and any new content will get indexed then.



Default Solr content index ☆

Home » Administration » Configuration » Search and metadata » Search API

Default content index created by the Solr Search Defaults module

Index status

101/101 indexed

Status	enabled (disable)
Datasource	Content (101/101 indexed)
Tracker	Default
Server	Solr Server
Server index status	There are 101 items indexed
Cron batch size	During cron runs, 50 items w

▼ START INDEXING NOW

Index all items in batches of 50 items

Queue all items for reindexing   Clear all indexed data

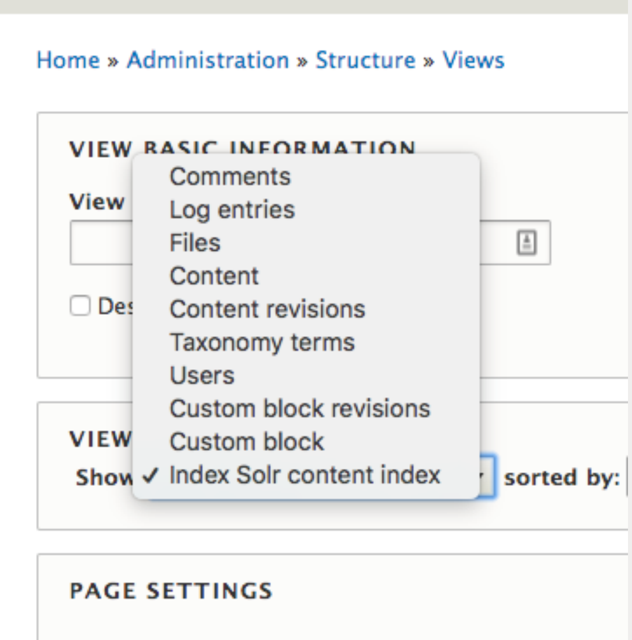
Create a Faceted Solr Search page

Final step is to create a search page. This is done by creating a View and create a Page.

General settings to configure:
View Settings: Show > Index (created earlier)
Page Settings:
- Page Title
- Path: /search/content
- Display Format > Unformatted list of “Rendered Entity”

The click “Save and edit” to configure the view.

1. Under Format > Show, click “Settings” for Rendered Entity and change ”view mode” to Teaser
2. Under Filter Criteria, add “Fulltext search” field and expose the field for filtering

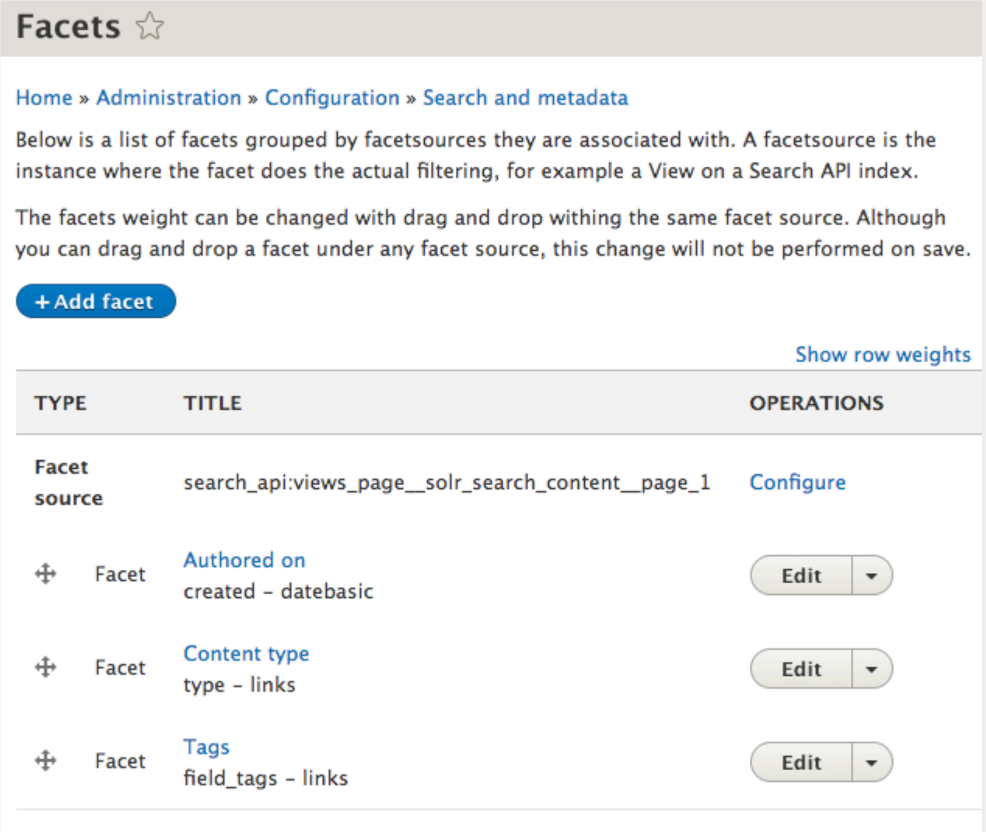


Further solr configuration can be done in the Solr schema, such as:

- Searching parts of works, using Solr’s EdgeNGramFilterFactory, for example, ”info” will match ”information”
- Lower Case Tokenizer – remove whitespace and non-letters and convert all letters to lowercase

Add search facets

With the search page setup now, we want to add facets to let users filter down content.
Navigate to Configuration > Search and metadata > Facets then click “Add facet”



Facets ☆

Home » Administration » Configuration » Search and metadata

Below is a list of facets grouped by facetsources they are associated with. A facetsource is the instance where the facet does the actual filtering, for example a View on a Search API index.

The facets weight can be changed with drag and drop withing the same facet source. Although you can drag and drop a facet under any facet source, this change will not be performed on save.

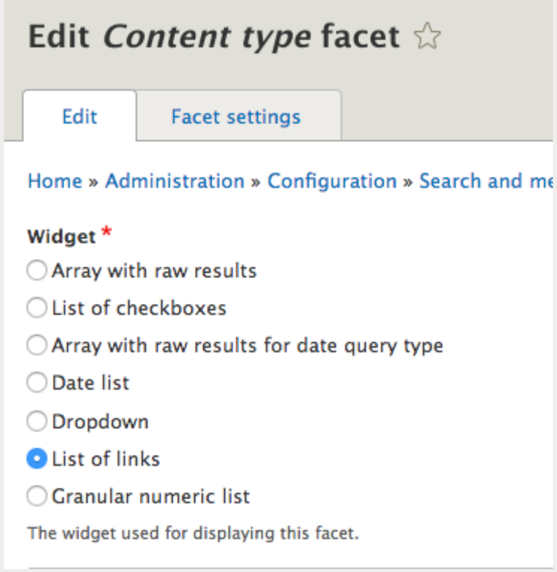
+ Add facet

Show row weights

TYPE	TITLE	OPERATIONS
Facet source	search_api:views_page__solr_search_content__page_1	Configure
✥ Facet	Authored on created – datebasic	Edit ▾
✥ Facet	Content type type – links	Edit ▾
✥ Facet	Tags field_tags – links	Edit ▾

Facets have a number of settings to configure:

- Widget
- Show the amount of results
- Sorting (by count, display value…)
- Operator (OR and AND - AND filters are exclusive and narrow the result set. OR filters are inclusive and widen the result set.)
- …



Edit *Content type* facet ☆

Edit | Facet settings

Home » Administration » Configuration » Search and me

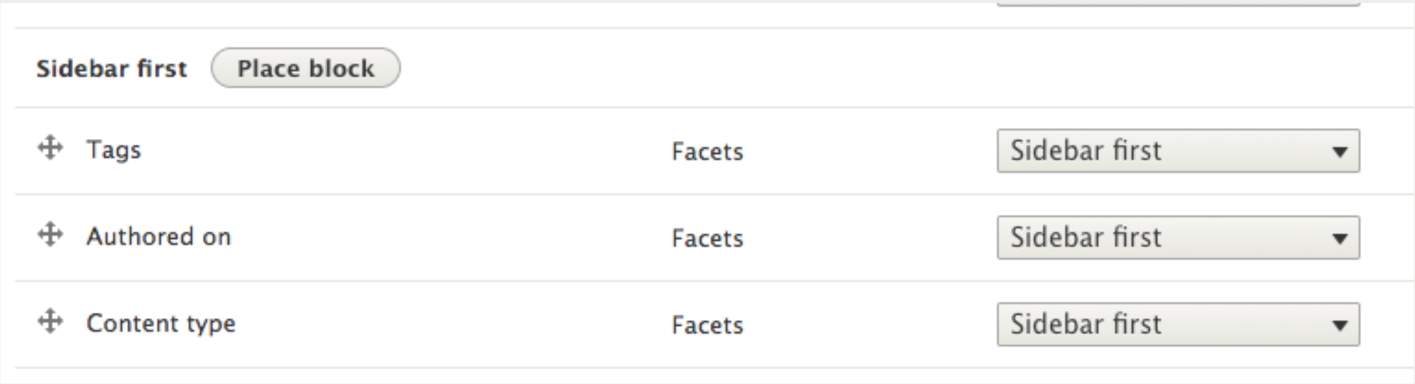
Widget *

○ Array with raw results
○ List of checkboxes
○ Array with raw results for date query type
○ Date list
○ Dropdown
● List of links
○ Granular numeric list

The widget used for displaying this facet.

Add search facets

Last step, add the newly created Facet blocks on the Block Layout page



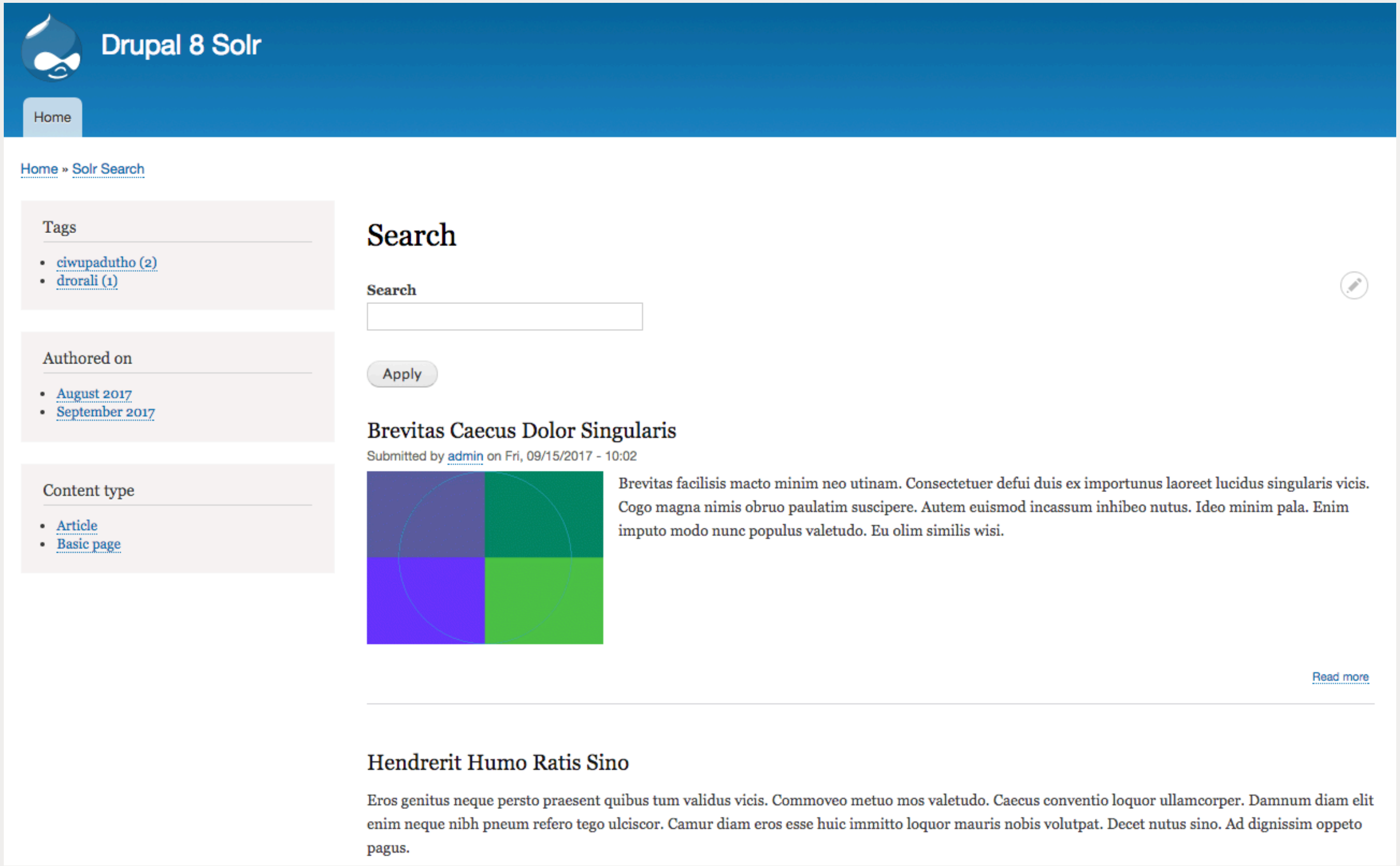
Sidebar first Place block

Tags	Facets	Sidebar first ▾
Authored on	Facets	Sidebar first ▾
Content type	Facets	Sidebar first ▾

Note: Facet blocks don't need to have the page visibility set, it will automatically detect if the page a search page, otherwise, they will not be displayed

Search Page

Finally, view the search page created



Home » Solr Search

Tags

- ciwupadutho (2)
- drorali (1)

Authored on

- August 2017
- September 2017

Content type

- Article
- Basic page

Search

Search

Brevitas Caecus Dolor Singularis

Submitted by admin on Fri, 09/15/2017 - 10:02

Brevitas facilisis macto minim neo utinam. Consectetuer defui duis ex importunus laoreet lucidus singularis vicis. Cogo magna nimis obruo paulatim suscipere. Autem euismod incassum inhibeo nutus. Ideo minim pala. Enim imputo modo nunc populus valetudo. Eu olim similis wisi.

Read more

Hendrerit Humo Ratis Sino

Eros genitus neque persto praesent quibus tum validus vicis. Commoveo metuo mos valetudo. Caecus conventio loquor ullamcorper. Damnum diam elit enim neque nibh pneum refero tego ulciscor. Camur diam eros esse huic immitto loquor mauris nobis volutpat. Decet nutus sino. Ad dignissim oppeto pagus.

What is Elasticsearch?

- Also capable of providing advanced full-text search
- Easier to install
- REST based APIs, a simple HTTP interface, and uses schema-free JSON
- Distributed Document Store - easy to use JSON document-oriented storage platform
- Logging data and analysis
- Visualization and real-time data monitoring when couple of Kibana

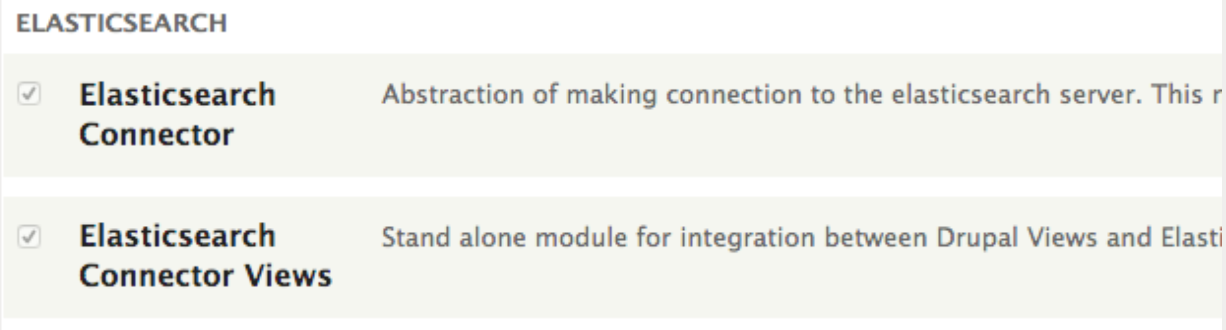
Elasticsearch Configuration

- Install module
- Create an index on Elasticsearch engine
- Create a view
- Attach facet filters

Install the Drupal modules

The Search API module needs Elasticsearch PHP library which provides the abstract layer of Elasticsearch Connector module in Drupal. This can be installed through composer.

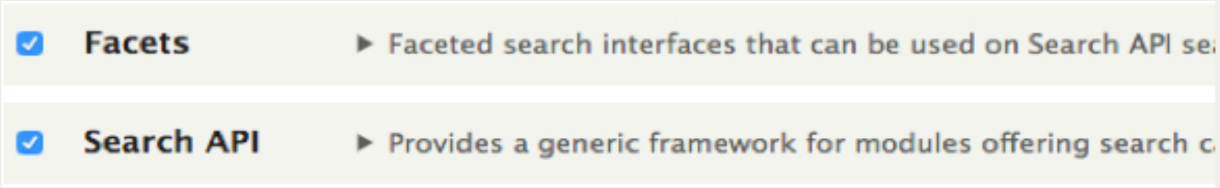
```
$ composer require nodespark/des-connector:5.x-dev
$ composer update
```



ELASTICSEARCH

☑ **Elasticsearch Connector** Abstraction of making connection to the elasticsearch server. This r

☑ **Elasticsearch Connector Views** Stand alone module for integration between Drupal Views and Elasti



☑ **Facets** ▸ Faceted search interfaces that can be used on Search API se

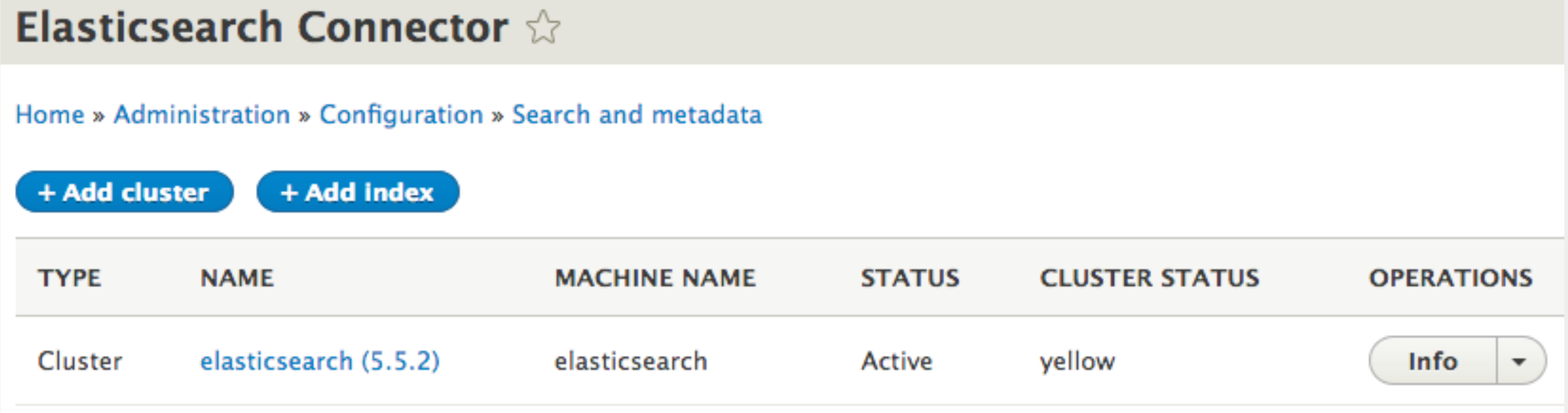
☑ **Search API** ▸ Provides a generic framework for modules offering search c

NOTE: Elasticsearch doesn’t have Solr Search Defaults-like module. Setting up and creating a view page has to be done manually.

Add Elasticsearch

Go to Configuration > Search and metadata > **Elasticsearch Connector**.

Click “Add Cluster” and configure the Server.



Elasticsearch Connector ☆

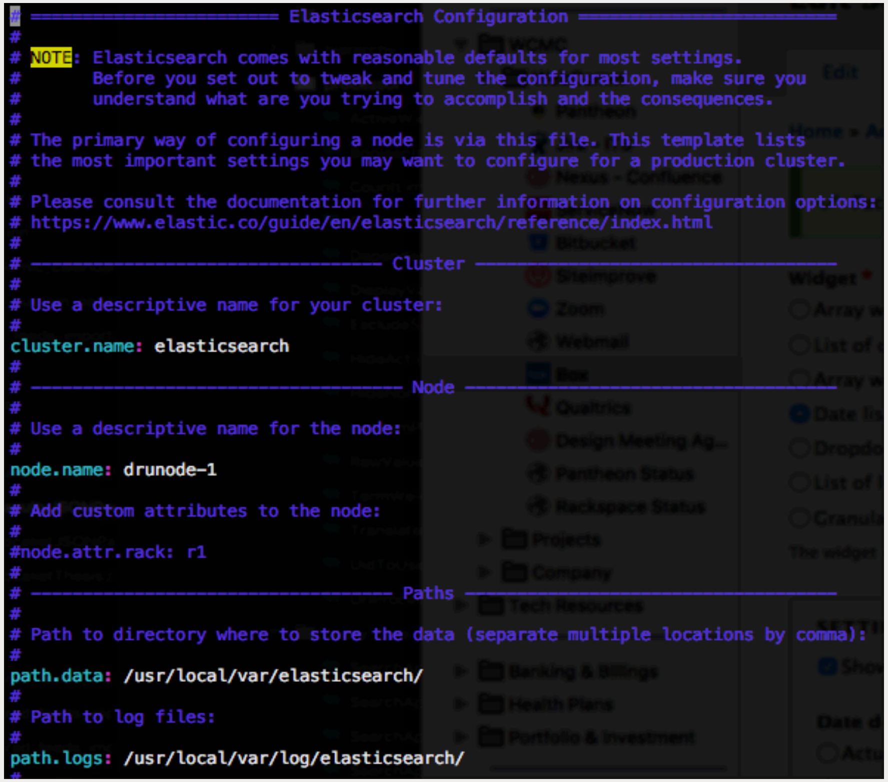
Home » Administration » Configuration » Search and metadata

TYPE	NAME	MACHINE NAME	STATUS	CLUSTER STATUS	OPERATIONS
Cluster	elasticsearch (5.5.2)	elasticsearch	Active	yellow	Info ▾

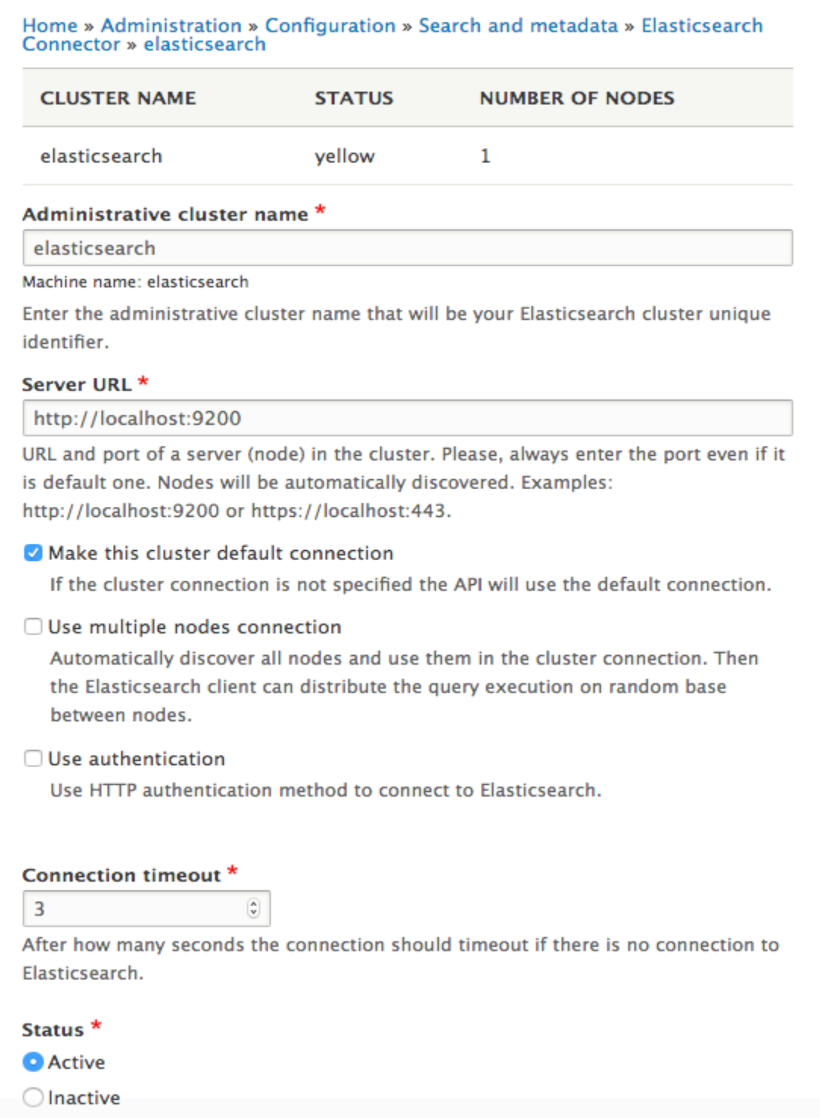
Add Elasticsearch (continue)

Go to Configuration > Search and metadata > **Elasticsearch Connector**. Click “Add Cluster” and configure the server.

As default, it is “elasticsearch.” If you want to edit the cluster/node information, edit elasticsearch.yml file.



```
# ======================== Elasticsearch Configuration =========================
#
# NOTE: Elasticsearch comes with reasonable defaults for most settings.
#       Before you set out to tweak and tune the configuration, make sure you
#       understand what are you trying to accomplish and the consequences.
#
# The primary way of configuring a node is via this file. This template lists
# the most important settings you may want to configure for a production cluster.
#
# Please consult the documentation for further information on configuration options:
# https://www.elastic.co/guide/en/elasticsearch/reference/index.html
#
# ---------------------------------- Cluster -----------------------------------
#
# Use a descriptive name for your cluster:
#
cluster.name: elasticsearch
#
# ------------------------------------ Node ------------------------------------
#
# Use a descriptive name for the node:
#
node.name: drunode-1
#
# Add custom attributes to the node:
#
#node.attr.rack: r1
#
# ----------------------------------- Paths ------------------------------------
#
# Path to directory where to store the data (separate multiple locations by comma):
#
path.data: /usr/local/var/elasticsearch/
#
# Path to log files:
#
path.logs: /usr/local/var/log/elasticsearch/
#
```



Home » Administration » Configuration » Search and metadata » Elasticsearch Connector » elasticsearch

CLUSTER NAME	STATUS	NUMBER OF NODES
elasticsearch	yellow	1

Administrative cluster name *

elasticsearch

Machine name: elasticsearch

Enter the administrative cluster name that will be your Elasticsearch cluster unique identifier.

Server URL *

http://localhost:9200

URL and port of a server (node) in the cluster. Please, always enter the port even if it is default one. Nodes will be automatically discovered. Examples: http://localhost:9200 or https://localhost:443.

☑ Make this cluster default connection

If the cluster connection is not specified the API will use the default connection.

☐ Use multiple nodes connection

Automatically discover all nodes and use them in the cluster connection. Then the Elasticsearch client can distribute the query execution on random base between nodes.

☐ Use authentication

Use HTTP authentication method to connect to Elasticsearch.

Connection timeout *

3

After how many seconds the connection should timeout if there is no connection to Elasticsearch.

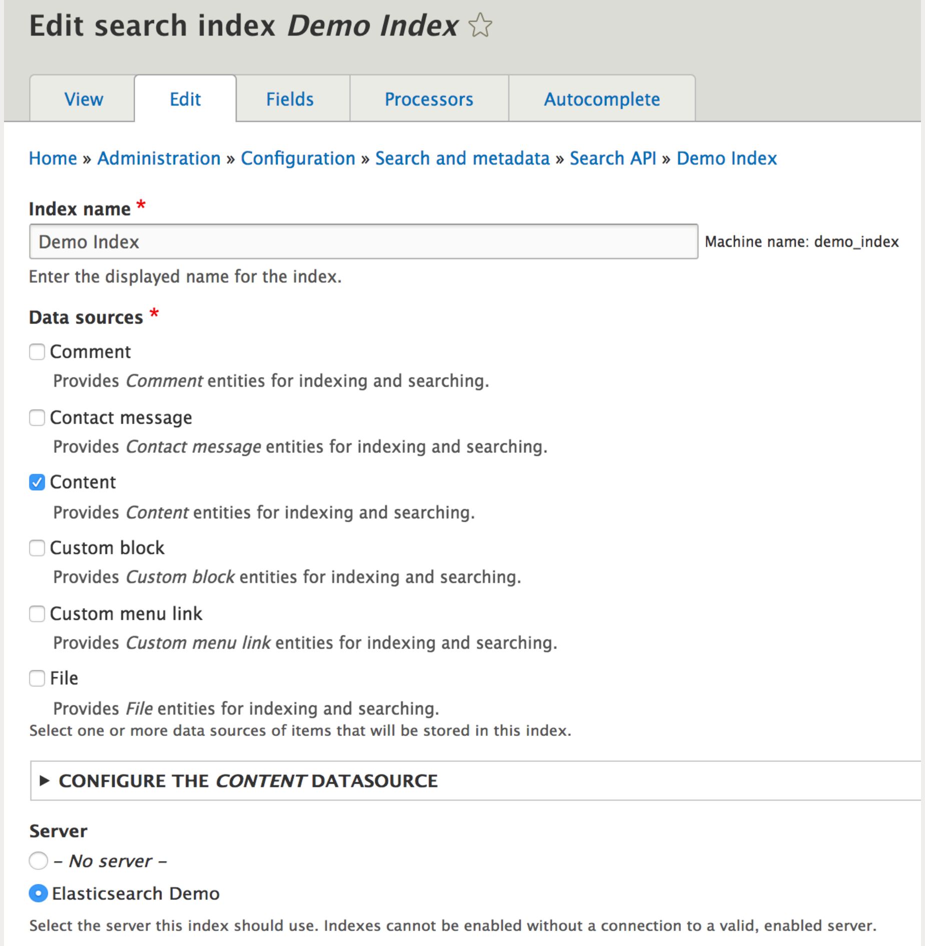
Status *

● Active

○ Inactive

Add search index (same as Solr)

Go to Configuration > Search and metadata > Search API. Click "Add Index"



Edit search index *Demo Index* ☆

View | Edit | Fields | Processors | Autocomplete

Home » Administration » Configuration » Search and metadata » Search API » Demo Index

Index name *

Demo Index — Machine name: demo_index

Enter the displayed name for the index.

Data sources *

□ Comment
Provides *Comment* entities for indexing and searching.

□ Contact message
Provides *Contact message* entities for indexing and searching.

☑ Content
Provides *Content* entities for indexing and searching.

□ Custom block
Provides *Custom block* entities for indexing and searching.

□ Custom menu link
Provides *Custom menu link* entities for indexing and searching.

□ File
Provides *File* entities for indexing and searching.

Select one or more data sources of items that will be stored in this index.

▸ CONFIGURE THE *CONTENT* DATASOURCE

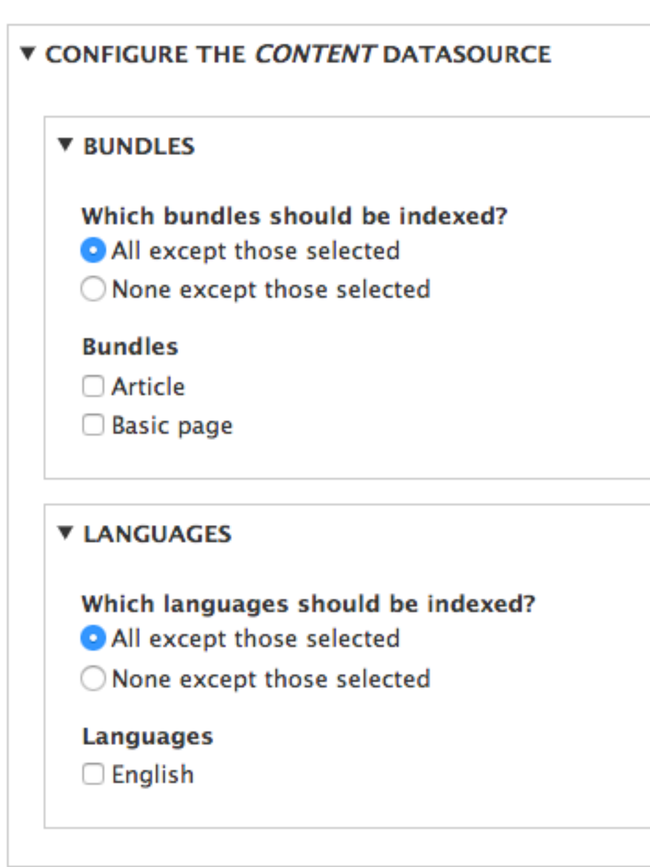
Server

○ – *No server* –

● Elasticsearch Demo

Select the server this index should use. Indexes cannot be enabled without a connection to a valid, enabled server.

Selecting the "Content" data source, options are presented to select which bundles are to be indexed



▾ CONFIGURE THE *CONTENT* DATASOURCE

▾ BUNDLES

Which bundles should be indexed?

● All except those selected

○ None except those selected

Bundles

□ Article

□ Basic page

▾ LANGUAGES

Which languages should be indexed?

● All except those selected

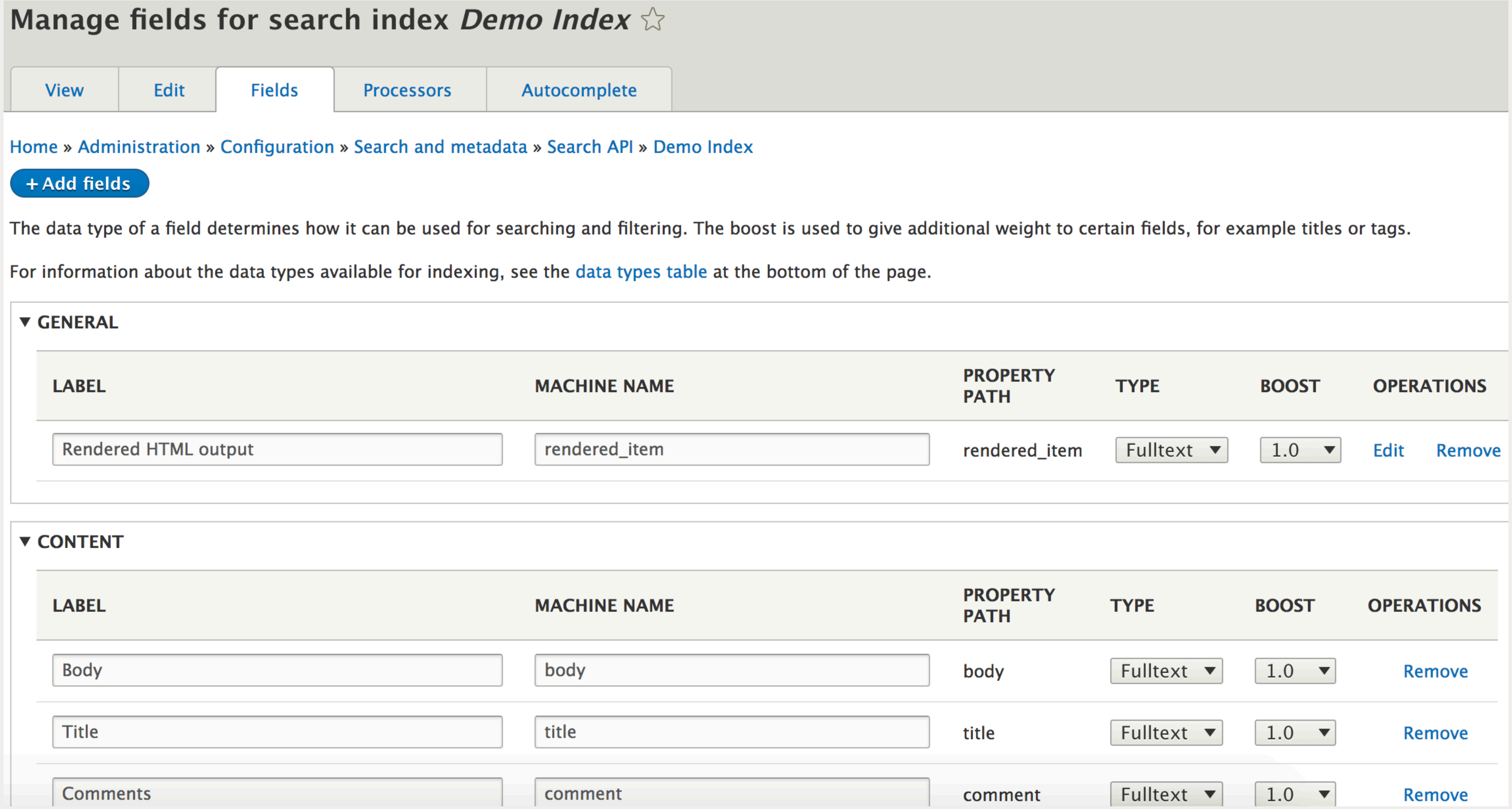
○ None except those selected

Languages

□ English

Add fields to index (same as Solr)

Before search can be performed, select all the fields that should be available to search. That is configured in the "Fields" tab.



Manage fields for search index *Demo Index* ☆

Home » Administration » Configuration » Search and metadata » Search API » Demo Index

The data type of a field determines how it can be used for searching and filtering. The boost is used to give additional weight to certain fields, for example titles or tags.

For information about the data types available for indexing, see the data types table at the bottom of the page.

▼ GENERAL

LABEL	MACHINE NAME	PROPERTY PATH	TYPE	BOOST	OPERATIONS	
Rendered HTML output	rendered_item	rendered_item	Fulltext ▼	1.0 ▼	Edit	Remove

▼ CONTENT

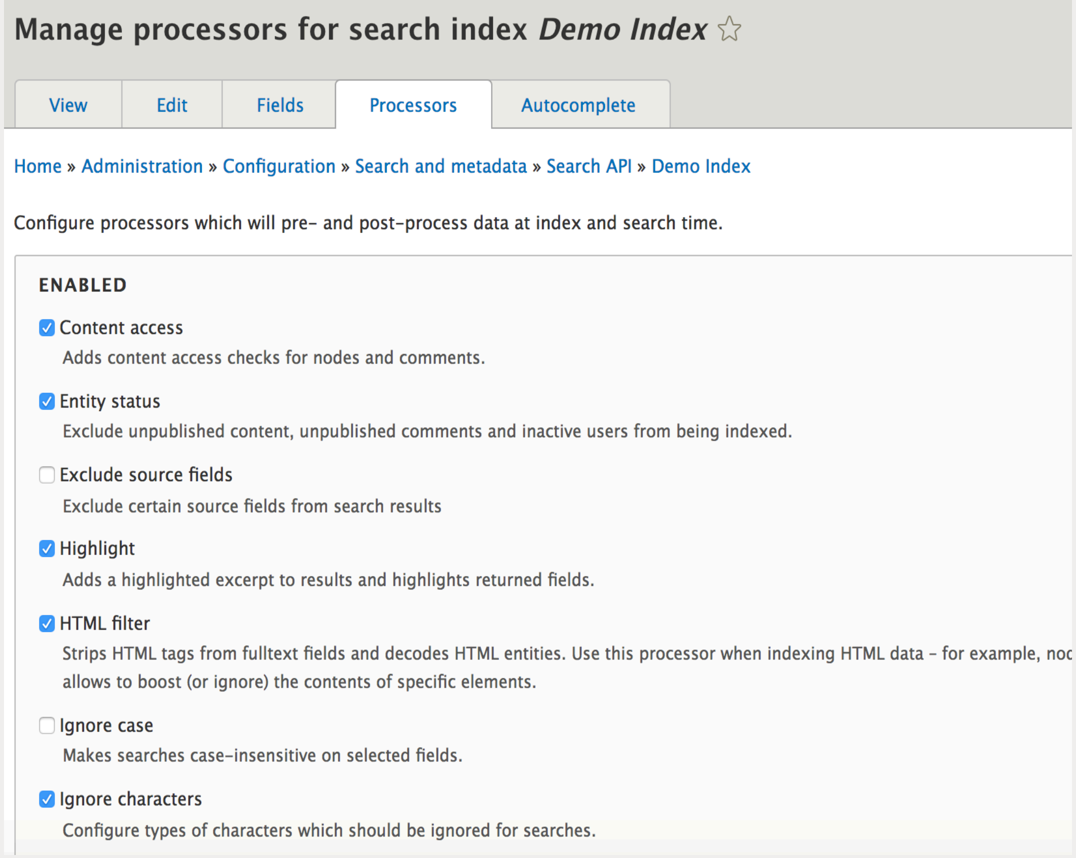
LABEL	MACHINE NAME	PROPERTY PATH	TYPE	BOOST	OPERATIONS
Body	body	body	Fulltext ▼	1.0 ▼	Remove
Title	title	title	Fulltext ▼	1.0 ▼	Remove
Comments	comment	comment	Fulltext ▼	1.0 ▼	Remove

Add processors to the index (same as Solr)

Last step is to add additional ’processors’.

This includes items such as:

- Content access
- Ignore case (case-insensitive search)
- Tokenizer (split into individual words)



Manage processors for search index *Demo Index* ☆

Home » Administration » Configuration » Search and metadata » Search API » Demo Index

Configure processors which will pre- and post-process data at index and search time.

ENABLED

☑ Content access
Adds content access checks for nodes and comments.

☑ Entity status
Exclude unpublished content, unpublished comments and inactive users from being indexed.

☐ Exclude source fields
Exclude certain source fields from search results

☑ Highlight
Adds a highlighted excerpt to results and highlights returned fields.

☑ HTML filter
Strips HTML tags from fulltext fields and decodes HTML entities. Use this processor when indexing HTML data – for example, noc
allows to boost (or ignore) the contents of specific elements.

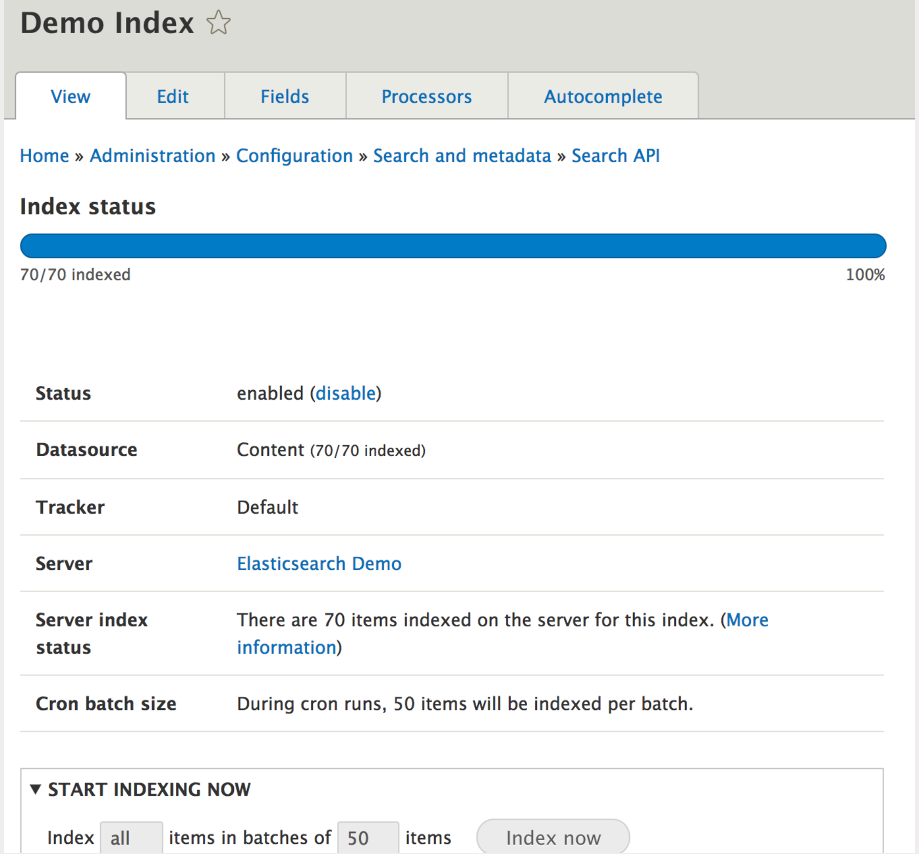
☐ Ignore case
Makes searches case-insensitive on selected fields.

☑ Ignore characters
Configure types of characters which should be ignored for searches.

Verify search index is working (same as Solr)

Once fields and processors are set up, go back to the ”View” tab. It will show the status of the index, and at this point, the content is ready to be indexed if not already set to index immediately when the index is created.

Indexing of content is done via cron and any new content will get indexed then.



Demo Index ☆

Home » Administration » Configuration » Search and metadata » Search API

Index status

70/70 indexed 100%

Status	enabled (disable)
Datasource	Content (70/70 indexed)
Tracker	Default
Server	Elasticsearch Demo
Server index status	There are 70 items indexed on the server for this index. (More information)
Cron batch size	During cron runs, 50 items will be indexed per batch.

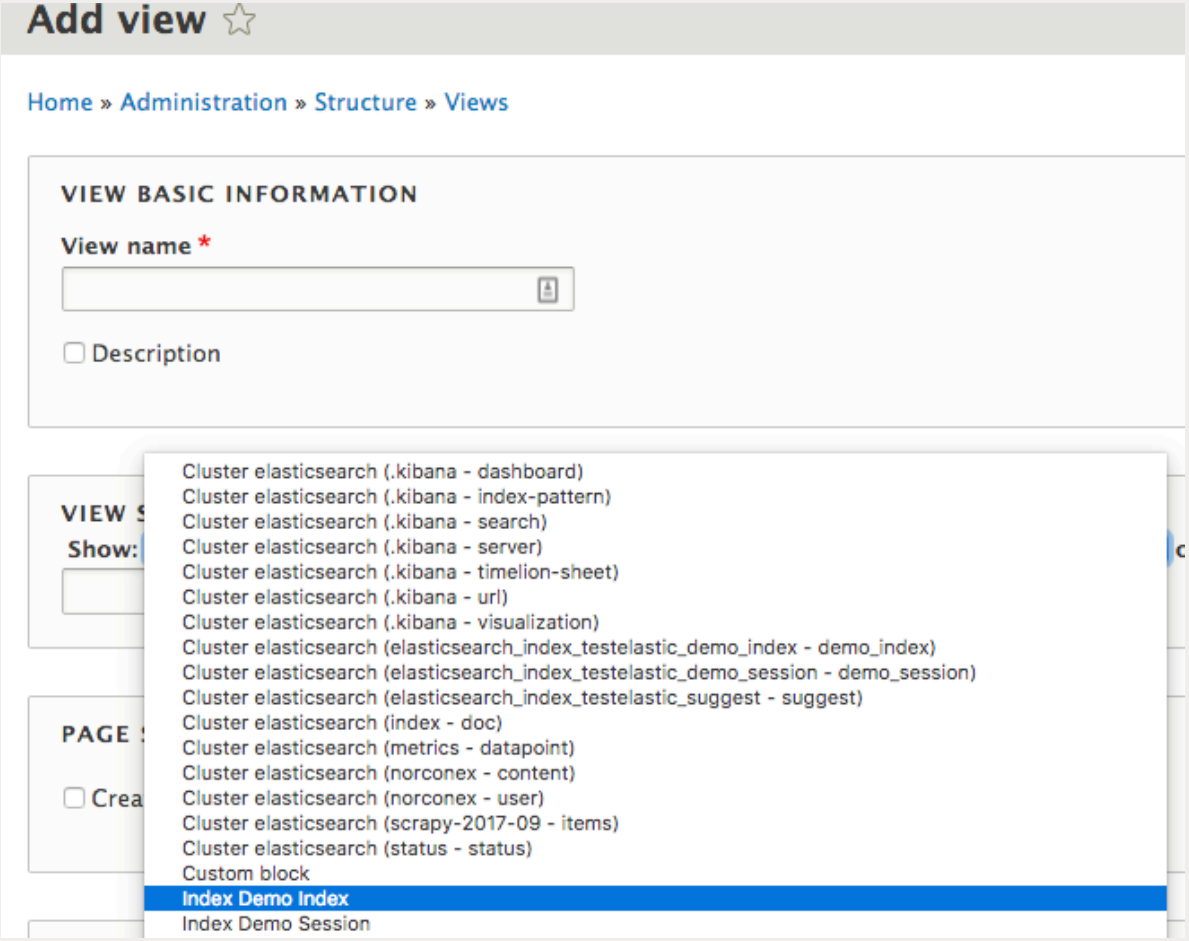
▼ START INDEXING NOW

Index all items in batches of 50 items

Create a view page

1. Go to Structure > Add view

2. Provide a view name and select your index name as the view source



Add view ☆

Home » Administration » Structure » Views

VIEW BASIC INFORMATION

View name *

□ Description

VIEW S

Show:

PAGE S

□ Crea

Cluster elasticsearch (.kibana - dashboard)
Cluster elasticsearch (.kibana - index-pattern)
Cluster elasticsearch (.kibana - search)
Cluster elasticsearch (.kibana - server)
Cluster elasticsearch (.kibana - timelion-sheet)
Cluster elasticsearch (.kibana - url)
Cluster elasticsearch (.kibana - visualization)
Cluster elasticsearch (elasticsearch_index_testelastic_demo_index - demo_index)
Cluster elasticsearch (elasticsearch_index_testelastic_demo_session - demo_session)
Cluster elasticsearch (elasticsearch_index_testelastic_suggest - suggest)
Cluster elasticsearch (index - doc)
Cluster elasticsearch (metrics - datapoint)
Cluster elasticsearch (norconex - content)
Cluster elasticsearch (norconex - user)
Cluster elasticsearch (scrapy-2017-09 - items)
Cluster elasticsearch (status - status)
Custom block
Index Demo Index
Index Demo Session

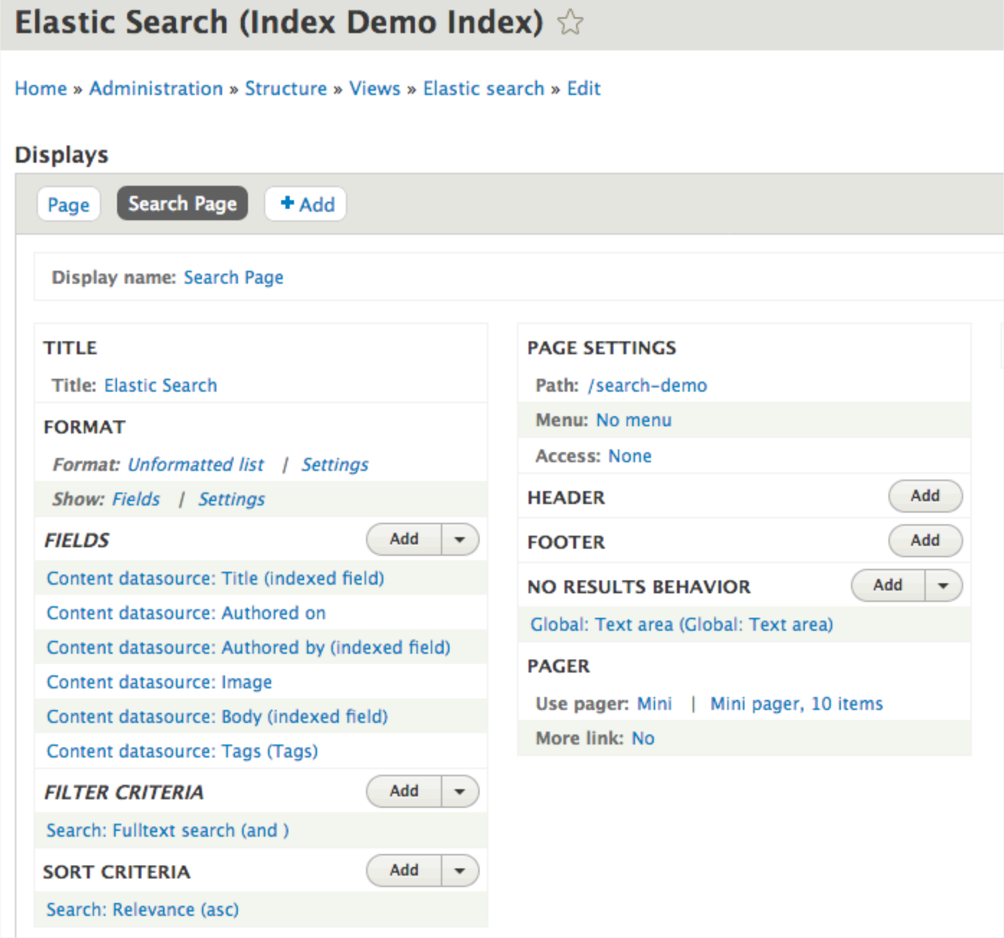
Create a view page (continue)

3. Under Format > Show, select Rendered Entity

Or, you can select "Fields" and add each field you would like to display in the Fields section.

4. Under Filter Criteria, add "Fulltext search" field and expose the field for filtering

5. Add Sort Criteria: The best one to use is "Relevance (desc)"



Elastic Search (Index Demo Index) ☆

Home » Administration » Structure » Views » Elastic search » Edit

Displays

Page | Search Page | + Add

Display name: Search Page

TITLE

Title: Elastic Search

FORMAT

Format: Unformatted list | Settings

Show: Fields | Settings

FIELDS Add ▾

Content datasource: Title (indexed field)

Content datasource: Authored on

Content datasource: Authored by (indexed field)

Content datasource: Image

Content datasource: Body (indexed field)

Content datasource: Tags (Tags)

FILTER CRITERIA Add ▾

Search: Fulltext search (and)

SORT CRITERIA Add ▾

Search: Relevance (asc)

PAGE SETTINGS

Path: /search-demo

Menu: No menu

Access: None

HEADER Add

FOOTER Add

NO RESULTS BEHAVIOR Add ▾

Global: Text area (Global: Text area)

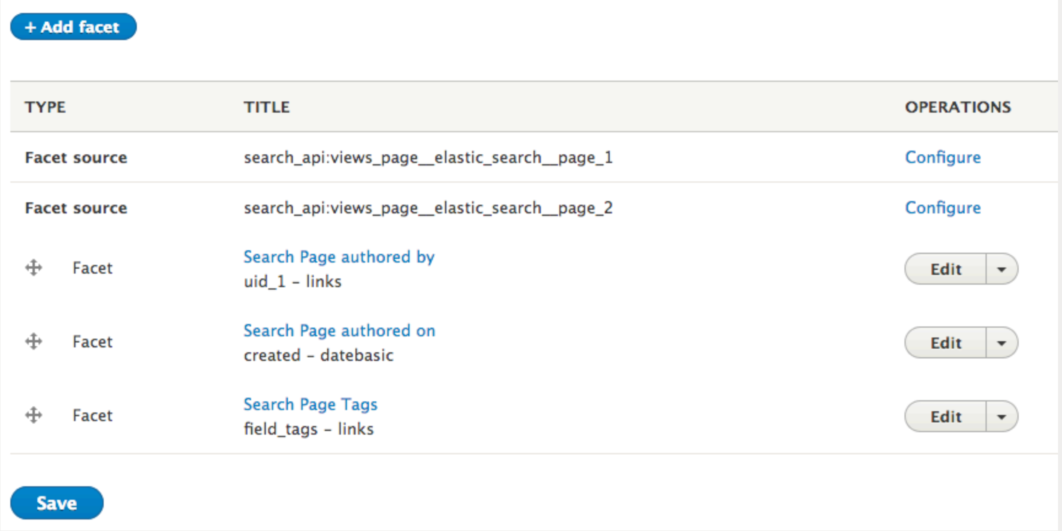
PAGER

Use pager: Mini | Mini pager, 10 items

More link: No

Add search facets (same as Solr)

With the search page setup now, we want to add facets to let users filter down content. Navigate to Configuration > Search and metadata > Facets then click "Add facet"



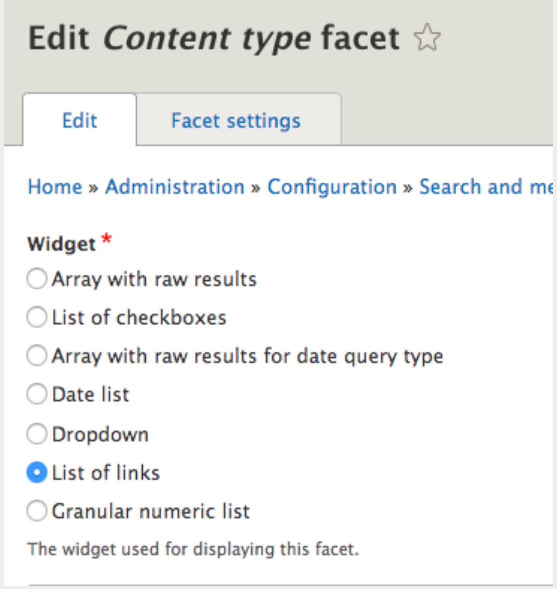
+ Add facet

TYPE	TITLE	OPERATIONS
Facet source	search_api:views_page__elastic_search__page_1	Configure
Facet source	search_api:views_page__elastic_search__page_2	Configure
✥ Facet	Search Page authored by uid_1 - links	Edit ▾
✥ Facet	Search Page authored on created - datebasic	Edit ▾
✥ Facet	Search Page Tags field_tags - links	Edit ▾

Save

Facets have a number of settings to configure:

- Widget
- Show the amount of results
- Sorting (by count, display value…)
- Operator (OR and AND - AND filters are exclusive and narrow the result set. OR filters are inclusive and widen the result set.)



Edit *Content type* facet ☆

Edit | Facet settings

Home » Administration » Configuration » Search and me

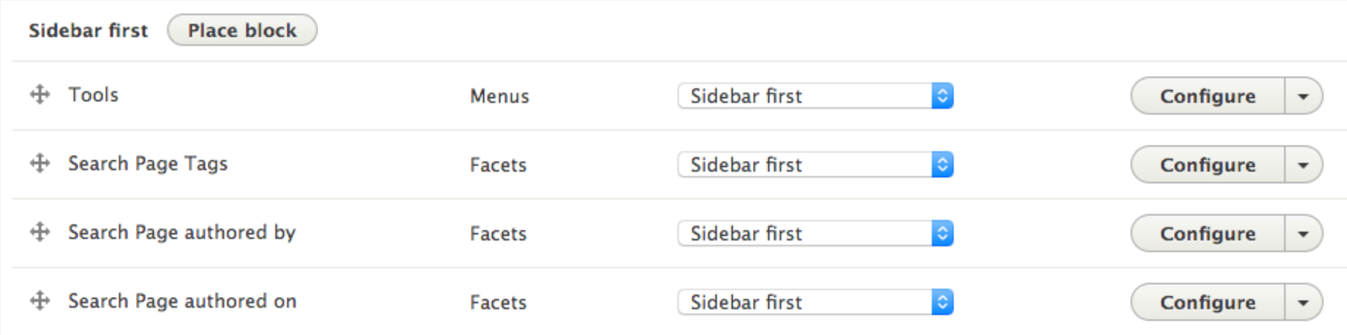
Widget *

○ Array with raw results
○ List of checkboxes
○ Array with raw results for date query type
○ Date list
○ Dropdown
● List of links
○ Granular numeric list

The widget used for displaying this facet.

Place facet blocks (same as Solr)

Last step, place the newly created Facet blocks on the Block Layout page



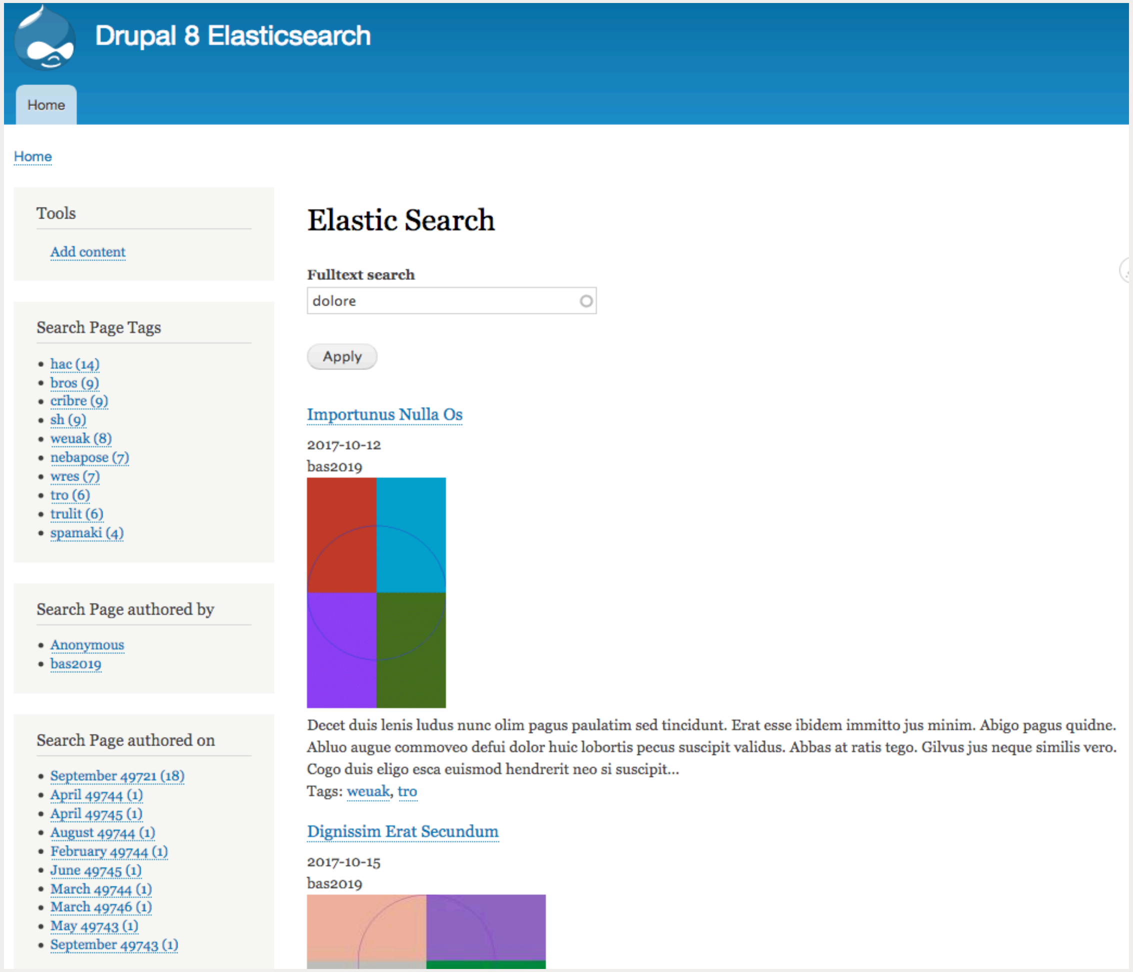
Sidebar first Place block

Tools	Menus	Sidebar first	Configure
Search Page Tags	Facets	Sidebar first	Configure
Search Page authored by	Facets	Sidebar first	Configure
Search Page authored on	Facets	Sidebar first	Configure

Note: Facet blocks don't need to have the page visibility set, it will automatically detect Each of the page a search page, otherwise, they will not be displayed.

Search Page

Finally, view the search page created.



Drupal 8 Elasticsearch

Home

Home

Tools

Add content

Search Page Tags

- hac (14)
- bros (9)
- cribre (9)
- sh (9)
- weuak (8)
- nebapose (7)
- wres (7)
- tro (6)
- trulit (6)
- spamaki (4)

Search Page authored by

- Anonymous
- bas2019

Search Page authored on

- September 49721 (18)
- April 49744 (1)
- April 49745 (1)
- August 49744 (1)
- February 49744 (1)
- June 49745 (1)
- March 49744 (1)
- March 49746 (1)
- May 49743 (1)
- September 49743 (1)

Elastic Search

Fulltext search

dolore

Apply

Importunus Nulla Os

2017-10-12

bas2019

Decet duis lenis ludus nunc olim pagus paulatim sed tincidunt. Erat esse ibidem immitto jus minim. Abigo pagus quidne. Abluo augue commoveo defui dolor huic lobortis pecus suscipit validus. Abbas at ratis tego. Gilvus jus neque similis vero. Cogo duis eligo esca euismod hendrerit neo si suscipit...

Tags: weuak, tro

Dignissim Erat Secundum

2017-10-15

bas2019

Apache Solr or Elasticsearch?

Source: A scene from Annie Get Your Gun (1950)

When to use Apache Solr

- If there are few thousand nodes of content (or more, depending on server performance and whether hosted locally or externally)
- You do not want to put stress on the database
- If searching is a major feature of the site
- If you need to search multiple fields

When to use Elasticsearch

- The Elastic Stack (Elasticsearch, Logstash, and Kibana) can interactively search, discover, and analyze to gain insights that improve the analysis of time-series data.

- No need for upfront schema definition. Schema can be defined per type for customization of indexing process.

- Has an edge in the cloud environment - this is depend upon SolrCloud advancement.

- Has advantages of search for enterprise or higher-ed level where analytics plays a bigger role.

Additional resources

Solr:
Configuring the Search API server (D8)
https://www.drupal.org/node/2763137

Drupal 8 and Solr: Google-fast search on your own website | Why and how
https://blog.openlucius.com/en/blog/drupal-8-and-solr-google-fast-search-your-own-website-why-and-how

Set up faceted Apache Solr search on Drupal 8
https://www.jeffgeerling.com/blog/2016/set-faceted-apache-solr-search-on-drupal-8-2016-deprecated

When to consider Apache Solr
https://lastcallmedia.com/blog/when-consider-apache-solr

Elasticsearch:
Indexing content from Drupal 8 using Elasticsearch
https://www.lullabot.com/articles/indexing-content-from-drupal-8-to-elasticsearch

How to index Attachments and Files to Elasticsearch
https://qbox.io/blog/index-attachments-files-elasticsearch-mapper

----

Solr vs. Elasticsearch: 5 Factors to Consider
https://www.cmswire.com/information-management/solr-vs-elasticsearch-5-factors-to-consider/

Questions?