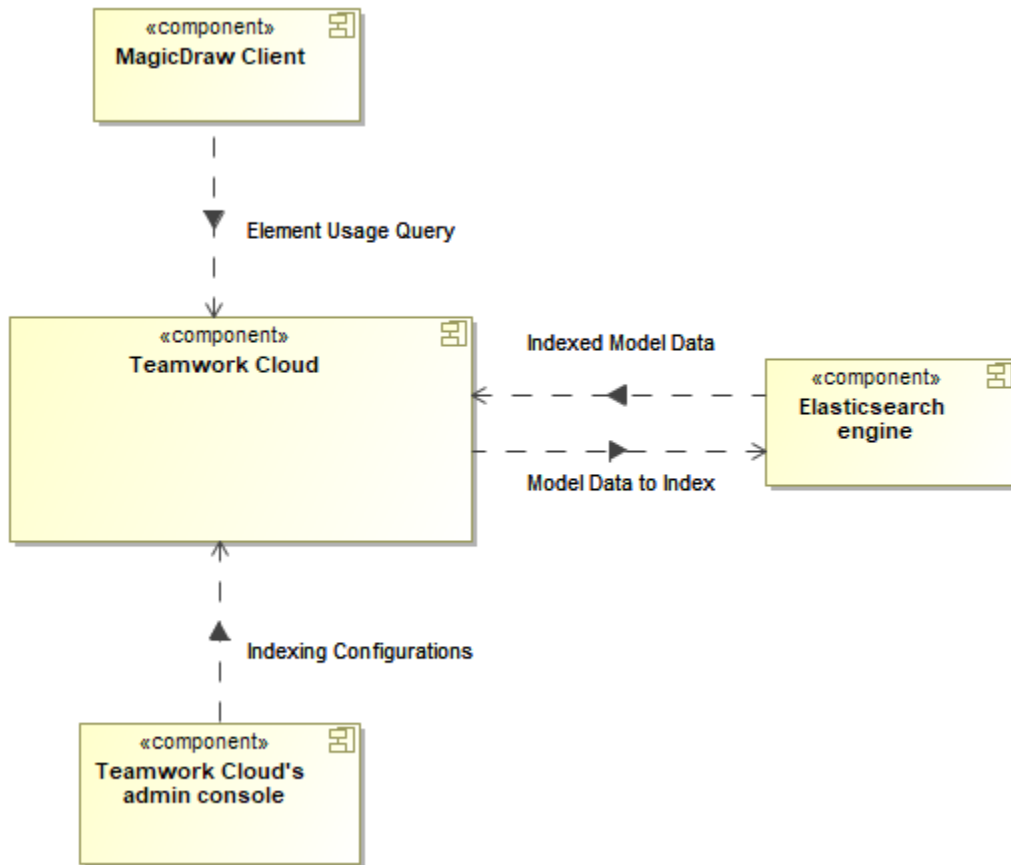


# Global element usage indexing and search

On this page

- [Installing and setting up Elasticsearch](#)
  - [System requirements](#)
  - [Installing Elasticsearch](#)
  - [Enabling global element usage indexing and search](#)
  - [Enabling Encrypted Communication \(SSL\) with Elasticsearch Server](#)

You can index resources for their element usages within the selected scope of the Teamwork Cloud repository and later query these usages through the modeling tool. This enables you to evaluate what impact modifying or deleting certain model elements will have on other models that are using it. To enable this functionality, Teamwork Cloud needs to be incorporated with a dedicated search engine - Elasticsearch. This component makes it possible to index model element usage data and serve it quickly when queried from a modeling tool.



Conceptual solution for the global element usage indexing and search functionality.

To start using the global element usage search functionality, you need to install and set up Elasticsearch (v7 series) as described below.

## Installing and setting up Elasticsearch

### System requirements


- We recommend using NVMe SSD disks for the Elastic search instance.
- The **-Xmx** value should be set to 4GB as a minimum.
- The expected indexing speed under recommended hardware settings is roughly 1 revision per second.
- The following formula can be used to roughly determine the required disk space per single Indexing Configuration:


*Used Project size (element count) x 400 bytes x N x # of Using Projects (specified scope),*

where N equals the number of times both using and used elements get changed throughout their history (N tends to range from 4 to 10 as noted from experiments with production data).  
The formula also assumes that actual element-level reference ratio is 15-20% from all of the elements in a Used Project.  
E.g. A DB size of 250GBs yields an index size of 60GBs under the above mentioned conditions.

## Installing Elasticsearch

Go to <https://www.elastic.co/downloads/elasticsearch> and install Elasticsearch (v7 series).

 We highly recommend deploying Elasticsearch on a machine separate from Teamwork Cloud and Cassandra due to additional server resource consumption.

 In the Elasticsearch installation directory, open the **jvm.options** file and make sure the **-Xms** and **-Xmx** properties are uncommented.

## Enabling global element usage indexing and search

Once you install Elasticsearch, configure Teamwork Cloud as described below to start using the global element usage search functionality.

To enable global element usage search

1. Go to *<Teamwork\_Cloud\_install\_directory>\configuration*, open the **application.conf** file, and add the following lines at the end of the file:
  - a. To enable the querying component, add the following property and set it to *true*:

```
esi.server.actor.query.component-enabled=true
```

- b. To enable resource indexing, add the following property and set it to *true*:

```
esi.indexer.enabled = true
```

- c. If Elasticsearch and Teamwork Cloud run on different machines, add the following line:

```
esi.query.es.node.host = "<es.host.or.ip>"
```


- d. If Elasticsearch is set up to listen on a different port from the default one (9200), add the following line:

```
esi.query.es.node.port = <es.port>
```

2. Restart Teamwork Cloud services.

Now you can [index resources for their element usages](#) in the Settings application and use the [global element usage search functionality](#) in a modeling tool.

## Enabling Encrypted Communication (SSL) with ElasticSearch Server

 SSL setup with Elasticsearch is a general security feature, required both to configure Elasticsearch itself and Teamwork Cloud. You must specify a username and password. See more information on setting up passwords in Elasticsearch: <https://www.elastic.co/guide/en/elasticsearch/reference/7.16/security-minimal-setup.html>

To enable encrypted communication between Teamwork Cloud and ElasticSearch servers

1. Prepare type PKCS #12 (\*.p12) certificate. You can:
  - Use an existing certificate for the machine that will host Elasticsearch, or
  - Generate a new certificate for Elasticsearch using `elasticsearch-certutil` (<https://www.elastic.co/guide/en/elasticsearch/reference/current/certutil.html>).

Example:

```
elasticsearch-certutil cert --self-signed --name <elasticsearch.host.name>
```

Alternatively, if the certificate subject name does not match the hostname, you can generate the certificate to facilitate the **Subject Alternative Name** extension:

```
lasticsearch-certutil cert --self-signed --name <arbitrary.subject.name> --dns <elasticsearch.host.name> --ip <elasticsearch.host.ip>
```

## 2. Configure *elasticsearch.yml*.

Enable security:

```
xpack.security.enabled: true
```

Enable SSL for transport:

```
xpack.security.transport.ssl.enabled: true
xpack.security.transport.ssl.verification_mode: certificate
xpack.security.transport.ssl.keystore.path: <path.to.p12.file>
xpack.security.transport.ssl.keystore.password: changeit
```

Enable SSL for HTTP (REST API):

```
xpack.security.http.ssl.enabled: true
xpack.security.http.ssl.verification_mode: certificate
xpack.security.http.ssl.keystore.path: <path.to.p12.file>
xpack.security.http.ssl.keystore.password: changeit
```

## 3. Import the certificate into Java's trusted certificates keystore *cacerts*.

First, export the X.509 certificate (usually \*.crt or \*.cer file) from the \*.p12 file (for this use tools like *OpenSSL* or *KeyStore Explorer*). Then import the certificate file using Java's keytool:

```
keytool.exe -import -file <path.to.certificate.file> -cacerts -alias <certificate.subject>
```

## 4. Set up Teamwork Cloud: in *application.conf*, set the following properties:

```
esi.query.es.security.enabled = true
esi.query.es.security.auth.username = "elastic"
esi.query.es.security.auth.password = "<elastic.password>"
esi.query.es.node.protocol = "https"
```

## Related pages

- [Creating indexing configurations](#)
- [Global element usage search](#)