

# Export tool

---

The export tool exports data from a Daisy repository to a directory or zip file conforming to the [Daisy import/export format](#)<sup>1</sup>.

For a basic usage scenario, see the [introduction](#)<sup>2</sup>.

## Specifying the set of documents to export

The documents to export are specified in an XML file which is passed to the export tool using the `-f` option.

The format of this file is as follows:

```
<?xml version="1.0"?>
<documents>
  <query>select id where true</query>
  <document id="841-DSY" branch="main" language="default"/>
</documents>
```

Both the query and document elements can be used as many times as desired. The branch and language attributes on the document element are optional.

Only the documents explicitly included via the query or document elements will be part of the export. So for example images included in these documents will not be automatically added.

When using queries, if you also want to export which documents are retired, then you need to include the retired documents too, via a query option:

```
<?xml version="1.0"?>
<documents>
  <query>select id where true option include_retired = 'true'</query>
</documents>
```

## Specifying extra schema types, namespaces and collections to export

By default, the export tool will export the subset of the repository schema used by the documents in the export. The same is true for namespaces and collections. If you want to force certain of these items to be part of the export, even when they are not used by the exported documents, you can also list them in the export set file, by using the following elements:

```
<documentType>...</documentType>
<fieldType>...</fieldType>
<partType>...</partType>
<namespace>...</namespace>
<collection>...</collection>
```

These elements can be mixed between the document and query elements.

It is allowed, but optional, to use the root element `exportSet` instead of `documents`.

Example:

```
<?xml version="1.0"?>
<exportSet>
```

- 
1. [daisy:332-cd \(Import/export format\)](#)
  2. [daisy:333-cd \(Import/export introduction\)](#)

```

<query>select id where true</query>
<document id="841-DSY" branch="main" language="default"/>
<documentType>SimpleDocumentContent"/>
<namespace>foo</namespace>
</exportSet>

```

## Configuration

The behaviour of the export tool can be influenced by a number of options. In addition, the documents and the schema can be altered upon export.

The configurations are done in an XML file which is supplied via the `-o` argument. To see an example of the such a file, which also shows the default configuration, use the `-d` argument:

```
daisy-export -d > exportoptions.xml
```

## Options

Option	Description
<code>exportLastVersion</code>	Sets whether the data from the last version of the document should be exported, rather than the live version. Default false.
<code>failOnError</code>	Sets whether the export should be interrupted when a failure occurs during exporting the document. Default false. When false, the export will simply continue with the next document (a list of failures is however kept and provided at the end of the import). When true, the export will stop in case of an error.
<code>stackTracesOfFailures</code>	Sets whether full stack traces of failures should be kept (for logging in the summary file). Default false.
<code>includeListOfRetiredDocuments</code>	Sets whether a list of retired documents should be included in the export. Default true.
<code>exportDocumentOwners</code>	Sets whether the owners of documents should be exported. Default false.
<code>enableLinkExtraction</code>	Sets whether link extraction should occur on document content. Default true. When true, the content of supported parts (e.g. Daisy HTML, navigation, books) are parsed for links. The branches, languages and namespaces used in those links are included in the export as non-required (the registration of these branches, languages and namespaces in the target repository is important for link extraction to work, and in case of the namespaces to make sure they are associated with the correct fingerprint). Setting this to false can save the export tool some work (the export will run a bit faster).
<code>exportVersionState</code>	Sets whether the state of the exported version should be excluded in the export. Default true. When the option <code>exportLastVersion</code> is false (the default), then the version state will always be 'publish' and thus this option serves little purpose in that case.
<code>exportReferenceLanguage</code>	Sets whether the reference language should be exported. Default true.

## DocumentExportCustomizer

See the description of the document import customizer of the [import tool](#)<sup>3</sup>.

The document export customizer is very similar, it is able to manipulate documents before being exported. The default implementation is able to drop fields, parts and collections, and can exclude out-of-line links and custom fields from the export. The interface to be implemented for custom implementations is:

```
org.outerj.daisy.tools.importexport.export.config.DocumentExportCustomizer
```

## SchemaCustomizer

Exactly the same as for the [import tool](#)<sup>4</sup>.

## Using the export tool programatically

It is possible to embed export tool in other applications. The export code is separate from the export tool command-line interface. The entry point to run an export is the following method:

```
org.outerj.daisy.tools.importexport.export.Exporter.run(...)
```

(This is a static method, but is no problem to run multiple exports concurrently in the same JVM).

If you want to have a look at the sources of the exporter, the code is found in the Daisy source tree below `applications/importexport`.

- 
3. [daisy:334-cd \(Import tool\)](#)
  4. [daisy:334-cd \(Import tool\)](#)