**Documentum Export Utility**

**Pre-Requisites:**

1. DFC version 6.5 or higher installed on host machine
2. Java 1.6 or higher installed on host machine

**Configurations:**

To run the tool/utility, configure the export.properties file to contain the necessary login credentials for your repository, and a query to gather the files you want to export. In addition, specify a location for the exported content on your file system. Here is the sample export.properties file.

export.query=select \* from dm\_sysobject where folder('/Temp',descend)  
export.user=dmadmin  
export.password=dmadmin  
export.repo=repo1  
export.path=c:/Temp/Export  
export.log=c:/Temp/Export/export.log

**Notes:**

* After the export tool runs once it will encrypt the password so it doesn’t sit around unprotected.
* The query must return r\_object\_id.

Once configured, run the export.bat batch file. The batch file does its best to create the correct class path, but you might have to help it if your Documentum environment is not installed in the default manner.

The export tool will export all of the objects it gathers from running the configured query. Each content object is exported to a file name that is its r\_object\_id (an easy way to avoid any file name collisions), followed by the DOS extension for its content type. The accompanying metadata is similarly named, followed with .metadata.xml. All files are exported to the path you specify in the export.properties file.   Note that no folder structures are created to mirror the structure in the repository. That is, the export directory will contain a “flat” file structure (that’s why avoiding file name collisions was important). Right now, the export tool can handle dm\_sysobjects, dm\_folders, virtual documents, and content-less objects. Since the export tool will export whatever the query returns, you can export versions by specifying ‘(all)‘ in the query, and subfolder structures by using the FOLDER(\*,'descend') function.

Here is an example .metadata.xml file.

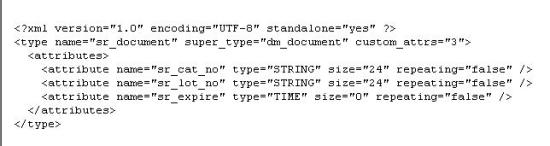
[](https://msroth.files.wordpress.com/2013/04/export_xml2.jpg)

**Notes:**

* This particular file is a virtual document, and it’s child nodes are listed in the <vd\_children> element.
* By default, the export tool exports metadata values for:  object\_name, object\_type, title, subject, acl\_domain, acl\_name, owner, version and a\_content\_type) (if applicable).
* If the type is a custom type (like this example is), values for all custom metadata properties will also be exported (note sr\_cat\_no, sr\_lot\_no, and sr\_expire above).

If the export tool encounters an object that appears to be a custom type, it will export a .typedef.xml file that contains definitions for all its custom attributes.

Here is an example of a .typedef.xml file.

[](https://msroth.files.wordpress.com/2013/04/type_xml.jpg)