

Exporting Documents from Content Central

Customer guide for exporting document files and their identifying metadata

Purpose

Content Central provides the building blocks needed to retrieve a customer's documents. For a small collection, use the built-in bulk download. For a complete or repeatable export, first create a document manifest and then retrieve each file through the Content Central transfer service.

Choose the right method

METHOD	BEST FOR	RESULT
Bulk download	A limited number of documents that can be selected in Search or a folder.	One ZIP file containing the selected current documents.
Scripted export	All documents, large collections, resumable runs, and exports that need an audit manifest.	A folder tree of current document files plus a CSV/XLS manifest and verification log.

Before you begin

- Use a dedicated Content Central account with access to every catalog and document type being exported.
- Confirm that the account can search, view, and download the required documents. The export will not bypass permissions.
- Choose a destination with enough free space for the document files, ZIP overhead, the manifest, and temporary files.
- Decide whether the requirement is the current document only or includes prior versions, deleted items, annotations, audit history, or other related data.

Important scope note

The download methods in this guide retrieve the current stored document. Prior versions and related records are separate data and should be discussed with Ademero before the export begins.

Method 1: built-in bulk download

- 1 Sign in to Content Central with the export account and open **Search** or the applicable folder.
- 2 Limit the results to a manageable batch, then select the documents to export.
- 3 Choose **Download**. Content Central prepares a ZIP containing the selected documents.
- 4 Save the ZIP, extract it, and repeat until every required result set is complete.

Use smaller batches if the browser, proxy, or local workstation has difficulty handling a large ZIP. Keep a simple batch log so that a failed or repeated batch can be identified.

Set the export scope before you start

- Document content: this procedure exports the current stored document. Historical versions and deleted items require separate planning.
- Metadata: include every required index field and system field in the manifest before generating the final report.
- Security: the export account's catalog and document-type permissions are enforced throughout the run.
- Related records: annotations, audit events, workflow history, and other linked information are outside the default document-file export.

Good control practice

Keep the original manifest, a batch or script result log, and a short note describing the destination folder convention. These records make completeness review and reruns straightforward.

Method 2: scripted, resumable export

A complete export is safest as a two-stage process: create an authoritative manifest, then download every document named in that manifest. This keeps enumeration separate from transfer and makes the run easy to resume and verify.

Stage A - create the manifest

- 1 In Content Central administration, create a **Documents** report that covers the required catalogs and document types.
- 2 Include at least these system fields: **Document ID**, **Catalog**, **Document Type**, **Document Filename**, **Document Folder**, **Document Version**, **Document Creation Date**, and **Document Modified Date**. Add business index fields if they are needed outside Content Central.
- 3 Generate the report and download it as CSV or XLS. Use Replace or Keep All output mode so the final manifest is a complete, identifiable file.
- 4 Review the manifest for expected counts, catalogs, and document types. Keep the original report unchanged as the export control file.

Stage B - retrieve each current document

Content Central exposes a SOAP transfer service at the application path below. Retain the Content Central application path supplied with the site address.

Purpose	Address
Transfer service	<code>https://<site>/<ContentCentral-path>/Services/2010/01/01/CCTransferService.svc</code>
Service description	<code>https://<site>/<ContentCentral-path>/Services/2010/01/01/CCTransferService.svc?wsdl</code>

Client sequence

- 1 Generate a SOAP client from the WSDL. Use HTTPS, MTOM message encoding, streamed transfer, and timeouts suitable for large files.
- 2 Send a **RequesterCredentials** SOAP header containing Domain, Username, and Password. The header namespace is `http://www.ademero.com/services/ContentCentral/DataContracts/2010/01/01/`.
- 3 Call **GetAuthenticatedUserId** once and retain the returned user ID.
- 4 For each Document ID in the manifest, call **DownloadDocument** with the Document ID and user ID. Save the returned stream using the returned filename.
- 5 Record Success, NoPermission, NotFound, or the received error. Retry transient failures with delay; do not silently skip a row.

Naming recommendation

Avoid filename collisions by writing each file beneath Catalog / Document Type / Document ID / Returned Filename. Sanitize characters that are invalid on the destination file system, but retain the original names in the manifest.

Implementation outline

The following C# outline shows the required client settings and call order. Add a Service Reference or use another SOAP client generator for the actual proxy and message-contract types.

```
// Outline only: generated proxy type names vary by client generator.
var binding = new BasicHttpBinding(BasicHttpSecurityMode.Transport) {
    MessageEncoding = WSMessagingEncoding.Mtom,
    TransferMode = TransferMode.Streamed,
    MaxReceivedMessageSize = long.MaxValue,
    ReceiveTimeout = TimeSpan.FromHours(24)
};
var client = new CCTransferServiceClient(binding,
    new EndpointAddress(transferServiceUrl));

using (new OperationContextScope(client.InnerChannel)) {
    var credentials = new RequesterCredentials {
        Domain = domain, Username = username, Password = password
    };
    var header = MessageHeader.CreateHeader(
        "RequesterCredentials", credentialsNamespace, credentials);
    OperationContext.Current.OutgoingMessageHeaders.Add(header);

    Guid userId = client.GetAuthenticatedUserId(
        new GetAuthenticatedUserIdRequest()).Id;

    foreach (Guid documentId in documentIds) {
        var response = client.DownloadDocument(
            new DownloadDocumentRequest(documentId, userId));
        // Require Result == Success, then copy response.Data to a file.
        // Log the document ID, returned name, byte count, and SHA-256.
    }
}
```

Verification and handoff

- Compare attempted, successful, failed, and skipped counts with the original manifest row count.
- For every successful download, record the Document ID, returned filename, byte length, destination path, and a SHA-256 checksum.
- Investigate every NoPermission and NotFound result. Correct access or confirm the source condition, then rerun only those rows.
- Open a representative sample from every catalog and document type, including large files and uncommon file formats.
- Store the original manifest, final result log, checksum list, and a description of the folder naming convention with the exported files.

Self-service requirements

This is a self-service procedure. Use an active Content Central account with access to the documents being exported. The transfer service enforces the account's existing permissions. A full administrator with access to all catalogs and document types can complete the entire export independently.

Protect credentials and exported documents in accordance with your organization's security and retention requirements. Remove temporary files after the export has been accepted.