



Version 10.2
Import/Export Guide

Inhalt

1	Einleitung	1
2	Import	1
2.1	Allgemeines.....	1
2.3	XML Import	4
2.3.1	Struktur der XML-Datei	4
2.3.2	Parameter	6
2.3.3	Templates	7
2.3.4	Import von Multimedia Dateien	7
2.4	CSV Import	8
3	Export.....	9
4	Rechtliche Hinweise / Impressum.....	12
4.1	Fragen und Anregungen.....	12
4.2	Impressum	12
4.3	Rechtliche Hinweise	12

1 Einleitung

Der hyper Content & Digital Asset Management Server erlaubt den Import und Export von Objekten (Seiten, Komponenten, Multimedia Komponenten) über eine definierte Schnittstelle. Damit lassen sich Inhalte aus externen Systemen automatisiert in das System importieren und aus dem System exportieren.

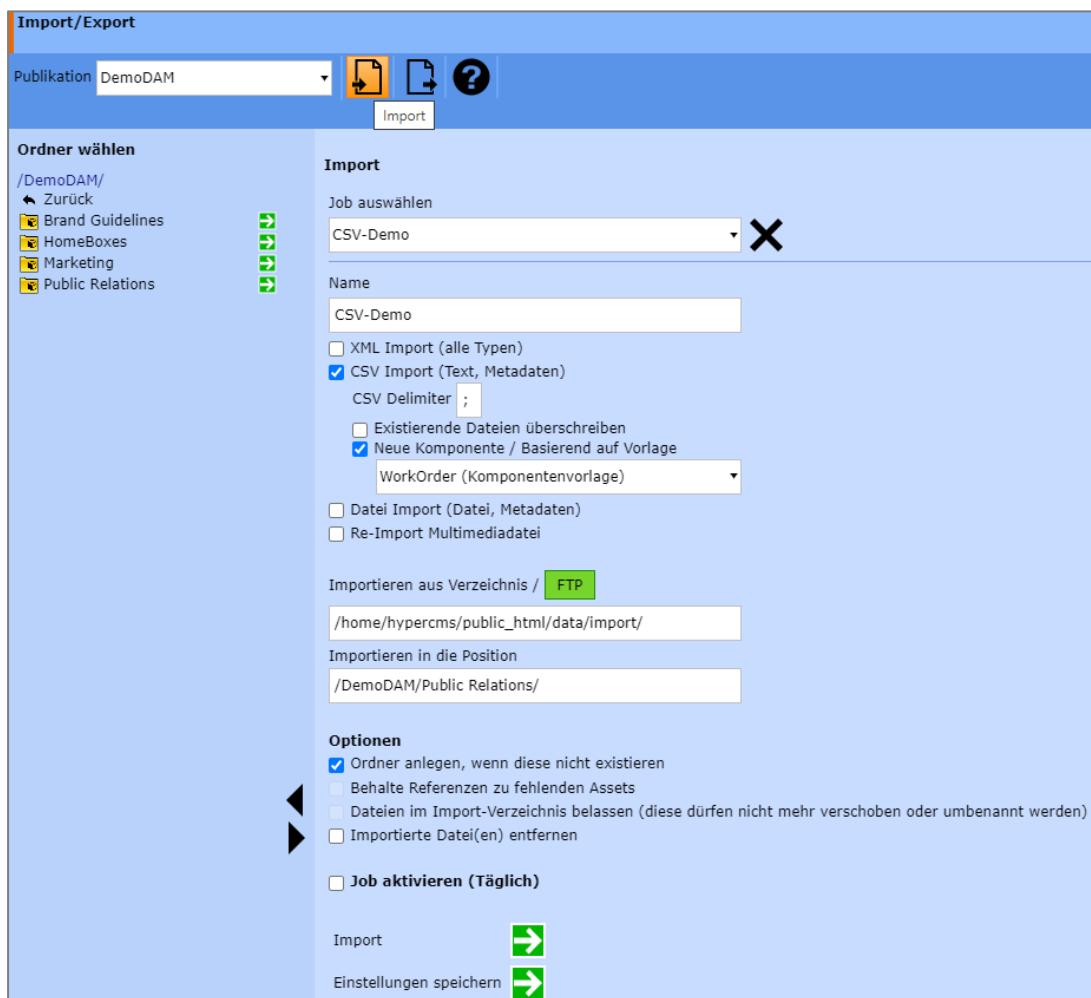
Die grafische Benutzeroberfläche für den Import und Export befindet sich im Navigator im Administrationsbereich, sofern das Connector Modul installiert ist.

2 Import

2.1 Allgemeines

Der Import wird vom REST API unter "hypercms/connector/import", welches sich im Connector Modul befindet, ausgeführt. Der Import kann über einen Job in regelmäßigen Zeitabständen oder auch nur bei Bedarf angestoßen werden.

Die Namen der zu importierenden Dateien sind nebensächlich, da alle Dateien im Importverzeichnis gelesen werden. Wichtig ist jedoch, dass bei der Verwendung von XML-Dateien die Dateinamen unterschiedlich sind und die Dateiendung .xml besitzen. Wird die Container ID im der zu importierenden XML-Datei übergeben, so wird das existierende Objekt überschrieben, sofern es sich auf der gleichen Position befindet. Neben dem REST API gibt es auch eine grafische Benutzeroberfläche für den automatisierten Import.



Name:

Sie müssen dem Importjob einen Namen geben, wenn Sie die Importparameter speichern möchten.

XML Import:

Der Import von Inhalten erfolgt über die Ablage von XML-Dateien in einem definierten Ordner. Falls keine Angabe erfolgt, im Ordner /data/import.

Damit lassen sich aus einem anderen System exportierte Inhalte in das eigene System importieren.

importtype ... „xml“ für den Import von XML-Dateien

CSV-Import:

Der Textinhalt von CSV-Dateien kann importiert werden. Wenn kein CSV-Trennzeichen angegeben wurde, bestimmt das System es automatisch anhand des Dateiinhalts. Normalerweise ist das Trennzeichen ; oder ,. Wenn der Inhalt bestehender Objekte oder Assets ersetzt werden soll, müssen Sie die Option „Vorhandene Dateien überschreiben“ aktivieren. Wenn Sie neue Objekte (Komponenten) basierend auf einer bestimmten Komponentenvorlage erstellen möchten, müssen Sie die Option „Neue Komponente / Basierend auf Vorlage“ aktivieren und eine Vorlage auswählen.

importtype ... „csv“ für den Import von CSV-Dateien

delimiter ... Trennzeichen, das in den CSV-Dateien verwendet wird

updateobjects ... true/false, aktualisiert den Inhalt bestehender Objekte

createobjects ... true/false, neue Objekte erstellen

template ... Name der Komponentenvorlage, die für das neue Objekte verwendet werden soll

Datei Import:

Import aller Dateien und Ordner aus dem angegebenen Importverzeichnis.

importtype ... „file“ für den Import von Dateien und Ordnern

Re-Import Multimediatei:

Wurden zuvor Multimediateien exportiert, wobei die Daten im System behalten wurden (siehe hierzu Kapitel Export), so können diese auch wieder reimportiert werden.

Dabei werden die symbolischen Links im Repository wieder durch die Originaldateien ersetzt.

Beachten Sie dabei, dass dieser Import keine Unterverzeichnisse analysiert. Die genaue Angabe des Verzeichnisses ist daher wichtig.

importtype ... „symlink“ für den Import von zuvor exportierten Dateien (Symbolische Links werden durch die Datei ersetzt)

Importieren aus Verzeichnis:

importdir ... Verzeichnispfad für die Ablage der zu importierenden Dateien, Standardwert ist data/import oder FTP Server Pfadangabe

Importieren in die Position:

Der Import von Ordnerstrukturen und Dateien kann auch ohne die Verwendung von XML-Dateien erfolgen. In diesem Fall durchsucht der Import-Prozess alle Dateien und Ordner im angegebenen Pfad und importiert diese als Ordner bzw. Assets im System. Dabei werden alle importierten Dateien analysiert und indiziert.

location ... Position für den Import der Ordner und Assets, Pflichteingabe für importtype „csv“ und „file“

Ordner anlegen, wenn diese nicht existieren:

Sollen nicht existierende Unterverzeichnisse automatisch beim Import angelegt werden, kann hierfür ein Parameter genutzt werden. Bei der Aktivierung des Parameters ist darauf zu achten, dass die Anlage der Unterordnern bei einem Import nicht mehr erfolgen kann und somit Informationen dieser Ordner daher nachträglich nur über die Bereitstellung des Container Namens im Zielsystem erfolgen kann.

createfolders ... true/false, Ordner automatisch erzeugen, Standardwert = false

Behalte Referenzen zu fehlenden Assets

Werden Links auf nicht existierende Komponenten importiert, so kann das System den Import dieser Beziehungen verhindern. Da beim Importieren das fehlende Objekt u.U. erst später importiert wird, ist das Ignorieren dieses Fehlers und der Import der Beziehungen möglich.

Diese Option wird beim Import von Daten aus CSV Dateien nicht verwendet.

linkignore ... true/false, Ignorieren von Links, Standardwert = false

Dateien im Import-Verzeichnis belassen:

Wenn diese Option gewählt wird, verwendet das System die Dateien im Quellverzeichnis und merkt sich die Referenz zu der Datei. Dies macht Sinn, wenn die Dateien gemeinsam genutzt werden und daher nicht aus dem Import-Verzeichnis entfernt werden dürfen. Dabei ist jedoch zu beachten, dass die Dateien im Quellverzeichnis weder umbenannt noch verschoben werden dürfen, da damit die Referenz auf die Datei ungültig wird. Diese Option gilt nicht beim Importieren von CSV-Inhalten, da diese nicht als Multimediateile betrachtet werden.

leavefiles ... true/false, Standardwert = false

Importierte Dateien entfernen:

Wenn diese Option gewählt wird, werden die Dateien im Quellverzeichnis nach dem erfolgreichen Importieren entfernt.

deletefiles ... true/false, Standardwert = false

Passwortschutz:

Sofern ein Passwortschutz für den Import/Export in der Datei „config.inc.php“ eingerichtet wurde, ist folgender Parameter zu setzen, siehe auch \$mgmt_config['passcode'].

passcode ... Passwort für den Zugriff auf den Import-Dienst

Beispiel eines Aufrufs des Imports via REST:

<https://domain.com/hypercms/connector/import/?passcode=mypasscode&importtype=xml&linkignore=true>

2.2 XML Import

2.2.1 Struktur der XML-Datei

Die abgelegten Dateien müssen folgendem XML-Format entsprechen:

```
<?xml version="1.0" encoding="UTF-8" ?>
<object>
    <hyperCMS>
        <contentcontainer></contentcontainer>
        <contentuser>username</contentuser>
        <contentlocation>%comp%/Publication/folder/</contentlocation>
        <contentobject>objectname</contentobject>
        <contenttemplate>templatename.comp.tpl</contenttemplate>
        <contentpublished>true</contentpublished>
        <contentworkflow>Publikation.Workflowname.xml</contentworkflow>
    </hyperCMS>
    <head>
        <pagetitle>title</pagetitle>
        <pageauthor>name of the author</pageauthor>
        <pagedescription>description</pagedescription>
        <pagekeywords>keyword list</pagekeywords>
        <pagecontenttype>text/html; charset=iso-8859-1</pagecontenttype>
        <pagelanguage>language shortcut (e.g. en)</pagelanguage>
        <pagerevisit>20 days</pagerevisit>
        <pagetracking>private.track.dat</pagetracking>
    </head>
    <textcollection>
        <text>
            <text_id>summary</text_id>
            <textuser>username</textuser>
            <textcontent>This is the summary!</textcontent>
        </text>
        <text>
            <text_id>detailed</text_id>
            <textuser>username</textuser>
            <textcontent><![CDATA[<font size="2">This is the detailed content!]]></textcontent>
        </text>
    </textcollection>
    <mediacollection>
        <media>
            <media_id>image</media_id>
            <mediauser>username</mediauser>
            <mediafile>Publication/Manual_hcm0325418.pdf</mediafile>
            <mediaobject>%comp%/Publication/Docs/Manual.pdf</mediaobject>
            <mediaalttext>User Manual</mediaalttext>
            <mediaalign></mediaalign>
            <mediawidth></mediawidth>
            <mediaheight></mediaheight>
        </media>
    </mediacollection>
    <linkcollection>
        <link>
            <link_id>link</link_id>
            <linkuser>username</linkuser>
            <linkhref>http://www.hypercms.net</linkhref>
            <linktarget>_blank</linktarget>
            <linktext>www.hypercms.net</linktext>
        </link>
    </linkcollection>
    <componentcollection>
        <component>
```

```
<component_id>component</component_id>
<componentuser>username</componentuser>
<componentcond>filename</componentcond>
<componentfiles>%comp%/Publication/folder/component1.jsp|%comp%/P
ublication/folder/component2.jsp</componentfiles>
</component>
</componentcollection>
<articlecollection>
<article>
<article_id>article</article_id>
<articletitle>News</articletitle>
<articledatefrom></articledatefrom>
<articledateto></articledateto>
<articlestatus>active</articlestatus>
<articleuser>username</articleuser>
<articletextcollection>
</articletextcollection>
<articlemediacollection>
</articlemediacollection>
<articlelinkcollection>
</articlelinkcollection>
<articlecomponentcollection>
</articlecomponentcollection>
</article>
</articlecollection>
</object>
```

2.2.2 Parameter

Um ein neues Objekt (Seite, Komponente oder Multimedia Asset) zu erzeugen, sind folgende Informationen notwendig:

```
<hyperCMS>
    <contentcontainer>0014020.xml</contentcontainer>
    <contentuser>username</contentuser>
    <contentlocation>%comp%/Publication/folder/</contentlocation>
    <contentobject>objectname</contentobject>
ODER
    <contentfolder>foldername</contentfolder>
    <contenttemplate>templatename.comp.tpl</contenttemplate>
    <contentpublished>true</contentpublished>
    <contentworkflow>Publikation.Workflowname.xml</contentworkflow>
</hyperCMS>
```

Pflichtangaben:

<contentuser> ... Name des Benutzers
<contentlocation> ... Pfad für die Ablage; %page% steht für den Root der Seiten;
%comp% steht für den Root der Komponenten; Danach muss die Angabe des Namens
der Publikation erfolgen sowie weitere Angaben zum Pfad, wobei jeder Ordner mit / zu
trennen ist
<contentobject> ... Name des Objektes ohne Dateierweiterung ODER
<contentfolder> ... Name des Folders (Folder VOR Object, falls beide Parameter
vorkommen)
<contenttemplate> ... Name des verknüpften Templates

Damit kann das neue Object einer Publikation zugeordnet werden und am richtigen Ort
auf Basis eines Templates erstellt werden.

Passwortschutz:

<contentpasscode> ... Wurde in der Datei „config.inc.php“ ein Passwort für die Nutzung
definiert, so ist die Übergabe des korrekten Passwortes notwendig.

Optionale Angaben sind:

<contentcontainer> ... Angabe eines bereits vorhandenen Containers, damit wird der
Inhalt des bestehenden Containers überschrieben. Bleibt der Eintrag leer, wird ein neuer
Container für den Inhalt erzeugt.
<contentdate> ... Datum im Format yyyy-mm-dd hh:mm
<contentpublish> ... Soll der Inhalt publiziert werden (true/false oder Datum/Leereintrag)
<contentworkflow> ... Zuordnung zu einem bestehenden Workflow

Der Bereich bzw. XML-Node <head> dient für die Übermittlung von Meta-Informationen
einer Seite. Dieser XML-Node kann auch leer bleiben, wenn keine Informationen
übermittelt werden sollen.

Danach folgen die XML-Nodes <textcollection>, <mediacollection>, <linkcollection>,
<componentcollection> und <articlecollection>.

Diese stellen den eigentlichen Inhalt des Objektes dar und sind gleichzeitig die
editierbaren Elemente, die das System erkennt. Auch diese XML-Nodes sind optional und
können leer bleiben, was jedoch bedeutet, dass ein Objekt ohne Inhalte erzeugt wird.

Beim XML-Node für <mediacollection> ist wesentlich, dass es sich hierbei nur um eine Referenz auf eine Multimedia Komponente handelt. Die Multimedia Komponente selbst muss im System verfügbar sein, siehe Lokationsangabe im Node <mediaobject>. Findet das System beim Import die zugehörige Multimedia Komponente, so wird der Node <mediafile> automatisch befüllt. Sollte die Multimediadatei nicht existieren, so wird der XML-Node <media> nicht importiert.

```
<media>
    <media_id>image</media_id>
    <mediauser>username</mediauser>
    <mediafile>Publication/Manual_hcm0325418.pdf</mediafile>
    <mediaobject>%comp%/Publication/Docs/Manual.pdf</mediaobject>
    <mediaalttext>User Manual</mediaalttext>
    <mediaalign></mediaalign>
    <mediawidth></mediawidth>
    <mediaheight></mediaheight>
</media>
```

Gleiches gilt auch für den XML-Node <componentcollection>. Auch hier wird geprüft, ob die Referenzen unter im Node <componentfiles> auf existierende Objekte zeigen.

```
<componentcollection>
    <component>
        <component_id>component</component_id>
        <componentuser>username</componentuser>
        <componentcond>filename</componentcond>
        <componentfiles>%comp%/Publication/folder/component1.jsp|%comp%/Publication/folder/component2.jsp</componentfiles>
    </component>
</componentcollection>
```

2.2.3 Templates

Wichtig ist, dass das zugeordnete Template die Elemente mit der richtigen ID besitzt, damit die importierten Inhalte auch im System angezeigt und editiert werden können. Die Inhalte werden in jedem Fall übernommen, da auch zu einem späteren Zeitpunkt das Template erweitert oder getauscht werden kann und somit die Inhalte angezeigt und bearbeiten werden können.

2.2.4 Import von Multimedia Dateien

Beim Import von Dateien (Multimedia Assets) ist darauf zu achten, dass das System auf den übermittelten Pfad auch Zugriff besitzt. Die Dateien müssen also nicht im Ordner „data/import“ abgelegt werden.

Damit die Dateien ohne XML importiert werden können, sind folgende Informationen an das Import-Script zu übergeben:

location ... Pfadangabe, wie z.B. %comp%/publication/folder/

Wenn XML-Dateien für den Import verwendet werden:

In der XML-Datei können die Metadaten in der <textcollection> hinterlegt werden. Auch hier ist darauf zu achten, dass eine Metadaten-Template existiert, das die Daten bearbeitbar macht.

Um auf die zu importierende Datei zu referenzieren, ist folgender Eintrag (fett markiert) notwendig:

```
<hyperCMS>
    <contentcontainer></contentcontainer>
    <contentuser>username</contentuser>
    <contentlocation>%comp%/Publication/folder/</contentlocation>
    <contentobject>filename</contentobject>
    <contenttemplate>default.meta.tpl</contenttemplate>
    <b><contentmediafile>image.jpg</contentmediafile></b>
    <contentpublish></contentpublish>
    <contentworkflow></contentworkflow>
</hyperCMS>
```

Der Name der Multimedia-Komponente <contentobject> muss dabei nicht dem Dateinamen der zu importierenden Datei <contentmediafile> entsprechen.

Nach dem erfolgreichen Import werden die Daten (XML-Dateien) aus dem Ordner „data/import“ vom System entfernt. Die Multimedia-Dateien werden ebenfalls entfernt, wenn sie sich direkt im Ordner „import“ befinden oder auch an einer anderen Lokation im Dateisystem.

2.3 CSV Import

Wenn Sie Inhalte oder Metadaten importieren möchten, können Sie hierzu CSV-Dateien (Comma Separated Values) im Importverzeichnis bereitstellen. Es ist wichtig, dass die CSV-Datei den Pfad (Ordner und Name), Namen des Objekts oder die Container-ID als ersten Spaltentitel und die in den Vorlagen für die Inhaltskennung verwendeten Inhalts-IDs als Spaltentitel enthält. Höchstwahrscheinlich haben Sie den Dateinamen und nicht die eindeutige Container-ID, die das System erstellt, wenn ein neues Objekt erstellt wird.

Dies ist ein Beispiel für die Definition in einer Vorlage. Dies wurde von einem Benutzer mit Berechtigungen zum Erstellen und Bearbeiten von Vorlagen im System definiert. Beachten Sie, dass jeder Inhalt oder jedes Feld durch eine ID definiert ist, z. id="Titel". Sie können den „Titel“ selbst manuell bereitstellen, indem Sie ihn in das entsprechende Feld eingeben oder den in einer CSV-Datei gespeicherten Inhalt importieren.

Definitionen in einer Vorlage:

```
[hyperCMS: textu id = 'Title' label = 'Title' infotype = 'meta']
[hyperCMS: textu id = 'Description' label = 'Description' infotype = 'meta']
[hyperCMS: textk id = 'Schlüsselwörter' label = 'Schlüsselwörter' infotype = 'meta']
```

The CSV file must have the following structure:

```
Name;Title;Description;Keywords
"flowers.xlsx";"Flowers";"A list of all flowers;Flowers, Nature"
"/roses/red.jpg";"Red roses";" Roses, Red, Flowers, Field, Nature"
"/roses/yellow.jpg";"Yellow roses";"Roses, Yellow, Flowers, Field, Nature"
```

Die Spalte, die den Objektpfad oder nur den Namen enthält, muss „objectpath“ oder „name“ (Groß-/Kleinschreibung egal) heißen.

Die Spalte mit der Container-ID (alternativ zum Objektpfad oder -namen) muss „containerid“, „container-id“ oder „container_id“ (Groß-/Kleinschreibung egal) heißen.

3 Export

Der Export von Inhalten erfolgt über die Speicherung von XML-Dateien und Multimediateien in einem zu definierenden Ordner, oder, falls keine Angabe erfolgt, im Ordner „data/export“.

Der Export wird vom REST API in „hypercms/connector/export“ ausgeführt. Der Export kann über einen Job in regelmäßigen Zeitabständen oder auch nur bei Bedarf angestoßen werden.

Neben dem REST API bietet das Connector Modul des Systems auch eine grafische Benutzeroberfläche für den Datenexport.

Achtung: Beim Export wird nur die letzte Version eines Objektes verwendet, alle anderen Versionen werden nicht exportiert.

The screenshot shows the 'Import/Export' interface with the following configuration:

- Publication:** DemoDAM
- Job auswählen:** BackupLargeOldFiles
- Name:** BackupLargeOldFiles
- Folgenden Ordner exportieren:** /DemoDAM/Marketing/
- Export in Verzeichnis /:** /home/hypercms/public_html/data/export/
- Alle Inhalte ab Position exportieren:**
- Container ID exportieren (Überschreibt existierenden Container mit gleicher ID!):**
- Exportierte Objekte löschen:**
- Mediendateien in Exportverzeichnis verschieben (Alle Daten werden im System behalten):**
- Bedingungen:**
 - Erstellung muss älter sein als: 3650 Tage
 - Letzte Änderung muss älter sein als: 730 Tage
 - Letzter Zugriff muss älter sein als: 365 Tage
 - Dateigröße muss größer sein als: 1000 MB
- Neue Bedingung hinzufügen:** AND 360view (Komponente) -> stageWidth =
- Filter:** AND (object.template = "default.meta.tpl" AND txtnodes.text_id = "LicenseEnd" AND txtnodes.textcontent < "2016-05-01")
- Job aktivieren (Täglich):**
- Export:**
- Einstellungen speichern:**

Damit der Inhalt eines Objektes exportiert werden kann, sind folgende Informationen an das Export-Script zu übergeben:

location ... Pfadangabe, wie z.B. %comp%/publication/folder/
object ... Name des Objektes (optional)

Definition des Exportverzeichnisses:

exportdir ... Verzeichnispfad für die Ablage der exportierten Dateien, Standardwert ist data/export

Möchte man alle Objekte einer gewissen Position exportieren, so kann man sich einer zusätzlichen Einstellung bedienen:

all ... true/false, Standardwert = false, funktioniert nicht bei Verwendung von contentcontainer

Diese Einstellung kann auf Ordner und Objekte angewendet werden. Bei Angabe eines Ordners werden alle Inhalte im Ordner exportiert. Bei einem Objekt, werden alle Inhalte ab der gleichen Position des Objektes exportiert.

Möchte man die Container Namen exportieren, so kann man folgenden Parameter setzen:

preserve ... true/false, Standardwert = false, soll der Container Name exportiert werden

Wird der Container Name exportiert und die exportierte Datei anschließend wieder importiert, so wird der vorhanden Container damit überschrieben.

Sollen nur die Multimediadateien unter Verwendung eines symbolischen Links exportiert werden:

symlink ... true/false, Standardwert = false

Damit bleiben alle Daten im System erhalten, jedoch befinden sich anstatt der Dateien nur mehr symbolische Links auf die Dateien im Exportverzeichnis. Damit kann man die speicherplatzintensiven Dateien auf einen günstigeren Speicherplatz auslagern. Der Speicher sollte jedoch ebenfalls in der Lage sein, die Dateien in einem angemessen Zeitraum bereit zu stellen.

Die Multimediadateien werden wieder automatisch importiert, wenn diese in der Originalversion (durch die Funktion downloadfile) abgerufen werden.

Möchte man die Objekte nach dem Export automatisch entfernen, so kann man folgenden Parameter setzen:

delete ... true/false, Standardwert = false

Folgende Bedingungen können angewendet werden:

createdays ... Nur Objekte deren Erzeugungsdatum älter als der angegebene Wert in Tagen ist, werden exportiert

editdays ... Nur Objekte deren letztes Änderungsdatum älter als der angegebene Wert in Tagen ist, werden exportiert

accessdays ... Nur Objekte deren letztes Zugriffsdatum älter als der angegebene Wert in Tagen ist, werden exportiert

filesize ... Nur Objekte deren Dateigröße größer als der angegebene Wert in MB ist, werden exportiert

Die freie Definition von Bedingungen in Bezug auf Metadaten oder Inhalte von Objekten ist ebenfalls möglich. Hierzu kann man sich des Editors bedienen, um beliebige weitere Bedingungen zu erstellen. Der Editor für neue Bedingungen erstellt diese in SQL im Eingabefeld unterhalb des Editors. Hier kann der Code bei Bedarf auch entfernt werden oder weiter ergänzt werden.

Die Auswahlfelder des Editors zeigen die Operatoren sowie die Text-ID der hyperCMS tags der Vorlagen in der Form: Vorlagenname[Vorlagentyp] -> Text-ID

Passwortschutz:

Wurde in der Datei „config.inc.php“ ein Passwort für die Nutzung definiert, so ist die Übergabe des korrekten Passwortes über den Parameter „passcode“ notwendig.

passcode ... Passwort entsprechend der Hauptkonfigurationsdatei

Beispiel eines Aufrufs des Exports via REST:

<https://domain.com/hypercms/connector/export/?passcode=mypasscode&location=%comp%/publication/folder/&all=true>

Damit werden die gewünschten Objekte gelesen und deren Content Container ermittelt. Die Content Container werden im Exportverzeichnis mit den Container Namen (XML-Datei mit Container-ID) gespeichert.

Beim Export von Multimedia-Komponenten werden deren Dateien ebenfalls im Ordner „data/export“ abgelegt. Dabei wird sowohl der Content Container als auch die Datei exportiert.

Im Content Container befindet sich die Referenz auf die Datei. Der Dateiname der Mediendatei beinhaltet im Namen die Container ID in der Form: name_hcm0014876.jpg

4 Rechtliche Hinweise / Impressum

4.1 Fragen und Anregungen

Sollten Sie weitergehende Fragen oder Anregungen zum Produkt haben, so wenden Sie sich bitte an den Support.

hyperCMS Support:

support@hypercms.com
<http://www.hypercms.com>

4.2 Impressum

Verantwortlich für den Inhalt:

hyperCMS
Content Management Solutions GmbH
Rembrandtstr. 35/6
A-1020 Wien – Austria

office@hypercms.com
<http://www.hypercms.com>

4.3 Rechtliche Hinweise

Vorliegende Installationsanleitung basiert auf der zum Zeitpunkt der Verfassung des Dokumentes verfügbaren Programmversion.

Der Hersteller behält sich Programmänderungen und –Verbesserungen vor.

Fehler und Irrtümer vorbehalten.

© 2026 by hyperCMS Content Management Solutions