Oracle® Transportation Management

Content Management System Integration
Guide

Release 6.5.1

Part No. F48326-01

October 2021

ORACLE®

Copyright Notice

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Contents

Tables

Figures

Send Us Your Comments

Oracle® Transportation Management Content Management System Integration Guide, Release 6.4.2

Part No. F48326-01

Oracle welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this manual?

If you find any errors or have any other suggestions for improvement, please indicate the title and part number of the documentation and the chapter, section, and page number (if available). You can send comments to us in the following ways:

- Electronic mail: otm-doc_us@oracle.com

If you have problems with the software, contact Support at https://support.oracle.com or find the Support phone number for your region at http://www.oracle.com/support/contact.html.

Intended Audience

The Content Management System Integration Guide is intended for clients, system administrators, and consultants.

Related Documents

- Oracle Transportation Management Administration Guide
- Oracle Transportation Management Installation Guide
- Oracle Transportation Management Technical Architecture Guide
- Oracle WebCenter Content Documentation Library

Definition of Related Content Management Terms and Concepts

- **CMS (Content Management System)**: A set of processes and technologies that support the collection, managing, and publishing of information in any form or medium. A CMS typically provides for long-term storage of information, information retrieval and global search capabilities.
- **Document**: An electronic file containing information, seen by the CMS as a unique entity. The document typically has metadata associated with it for identification and search.
- **Folder**: A named collection holding documents and other folders. Folders can be hierarchical.
- **Versioning**: Support for a checkin/checkout paradigm of document editing. A CMS that supports versioning allows a single document to hold multiple versions of information.
- **Unfiling**: Support for the addition of a CMS Document without a specified folder.

Definition of Related Protocol Terms and Concepts

- **WCC (Oracle WebCenter Content)**: Oracle's enterprise-class content management solution.
- **ODC (Oracle Documents Cloud)**: The Oracle Documents Cloud Service is a enterprise ready, secure, and powerful content collaboration solution.

Definition of Related Oracle Transportation Management Terms and Concepts

- **Business Object:** An Oracle Transportation Management entity that can own content and has a transactional life cycle. Business object examples include shipments and trade transactions.
- **Report**: Generation of business content from an Oracle Transportation Management business object and a BI Publisher template.
- **Document**: A piece of content managed by Oracle Transportation Management. The content may be stored within the Oracle Transportation Management schema or an external CMS.
- **Document Type:** A classification of documents within Oracle Transportation Management. E.g., Bill of Lading is a type of document/report owned by shipments.
- **Standard Document**: The association of a document type with a particular business object. E.g., a Bill of Lading may be required for a particular shipment.
- **Revision**: The instantiation of content for a standard document. A business object may have one or more revisions for a given standard document. Each revision corresponds to a single document.

Change History

Date	Document Revision	Summary of Changes
10/2021	-01	6.5.1 Release.

1. Introduction

The system provides optional integration to the following Content Management Systems:

- Oracle WebCenter Content (WCC)
- Oracle Documents Cloud(ODC)

CMS Integration Levels

When coupling Oracle Transportation Management with your CMS, the system supports three levels of integration.

- **Storage**: The CMS is used solely to store document content managed exclusively by Oracle Transportation Management. This option removes content overhead from the transactional schema. In a pure storage model, the focus of the CMS is to provide scalable and reliable long-term storage for application-specific content.
- **Content Organization**: The CMS is used to store and organize document content. This option allows CMS users to access transportation documents within an enterprise document manager. They may browse a hierarchical tree or use global search to retrieve content.
- **Custom Search**: The CMS is used to store, organize, and search for document content. Searches may include transportation attributes, attached as custom metadata values to CMS documents.

2. How to Configure Storage

To set up a CMS for storing document content created by Oracle Transportation Management, first create a content management system to represent the CMS connection. Three types of content management systems are supported:

- **Database**: This is a reserved type used by the **DATABASE** content management system. It stores document content in the Oracle Transportation Management **GLOGOWNER** schema.
- **WebCenter Content**: Documents are stored and managed in an Oracle WebCenter Content instance. With the WebCenter Content Java API, Oracle Transportation Management tightly integrates both products.
- **Oracle Documents Cloud:** Documents are stored and managed in an Oracle Documents Cloud instance. With the Oracle Documents Cloud REST API, Oracle Transportation Management tightly integrates both products.

Preparing the CMS for Storage

Certain custom installation steps are necessary for Oracle Transportation to use a CMS for document storage. These steps need to be run by an administrator of the CMS.

WebCenter Content

Oracle Transportation Management communicates with WCC over the Remote Intradoc Client (RIDC). To properly enable this protocol, the WCC server must set the Intradoc server port and register all Oracle Transportation Management application servers as trusted hosts for RIDC. For information on how to configure WCC for RIDC, see
http://docs.oracle.com/cd/E29542_01/doc.1111/e14495/newui.htm#CHDBABIJ.

Oracle Documents Cloud

Oracle Transportation Management communicates with Oracle Documents Cloud using the REST API. For information on how to configure Oracle Documents Cloud, see [Oracle Content and Experience Cloud](). For more information on the Rest API see [REST API for Documents]().

Defining CMS Connections for Storage

A CMS connection in Oracle Transportation Management is defined with a Content Management System record. This can be accessed via the main menu from **Business Process Automation > Power Data > Document Generation > Content Management Systems**.

Every CMS connection requires the following fields:

- **URL**: The URL needed to communicate with the CMS over a specified protocol.
- **User**: The CMS user name used to authenticate the connection.
- **Password**: The CMS credentials used to authenticate the connection.

Architectures requiring more complex credentialing, e.g. those involving certificate authentication, are not supported.

WebCenter Content

For WebCenter Content, Oracle Transportation Management supports only the IDC protocol. This requires the application server be a trusted client from the standpoint of the document server. The connection URL should be of the form:

```
idc://<WCC host>:<IDC port>

or
https://<WCC host>:<WCC SSL Port>/cs/idcplg
```

Typically, the IDC port is 4444. Note that the user and password are ignored for the initial connection but allow for document-based security on subsequent document requests.

The certificates required for Oracle Transportation and Global Trade Management to trust the URL should be loaded to the Java trust store.

Oracle Documents Cloud

For Oracle Documents Cloud, Oracle Transportation Management supports REST API calls. The connection URL would usually be of the form:

```
https://<ODC host>:<ODC port>/documents/api/<ODC release>
```

Note that the user and password are needed to make calls to Oracle Documents Cloud.

Storage Document Identification

Though each CMS has its own concept of a document, a version and the relationship between them, three basic attributes identifying a document are critical to proper storage:

- **A GUID**: Each document must have a globally unique identifier to distinguish it from other documents. For some systems, this may be supplied by the user; for others it may be generated. Note that the GUID is stored with the Oracle Transportation Management document record after the content is created or queried.
- **A primary name**: For systems supporting versioning, a document must have a primary name. Documents that share a primary name are grouped together as versions of the initial document of that name. For systems without versioning, the primary name acts as an alternate ID to identify a document when browsing or searching.
- **A folder**: Many content management systems support the browsing of documents via hierarchical folders. When storing content, Oracle Transportation Management optionally passes a folder specification to the CMS. This takes the form `<parent folder>/<child folder>/`…. On storage, folders and subfolders are created by the CMS to meet the folder specification and the content is stored in the last descendant on the folder branch. Note that the folder specification is optional. Documents stored without a folder are considered *unfiled*. Some CMS systems, however, do not support unfiled documents and a folder specification is required for storage.

WebCenter Content

GUID

For WebCenter Content, the GUID is given by the Document Name parameter, corresponding to the **dDocName** argument in WCC RIDC calls. The GUID is limited to 30 characters due to WCC constraints when backed by an Oracle database. By default, the Document Name is set to:

```
OTM-<sequence>1
```

where the sequence is based on an Oracle sequence, **CMS_DOCUMENT_SEQ**, and left padded with zeros. This guarantees the uniqueness of the GUID across all Oracle Transportation Management documents. If an alternate Document Name is needed, consult the Document Parameters section for instructions on custom parameter generation.

Primary Name

The primary name for WebCenter Content is given by the Document Title parameter, corresponding to the **dDocTitle** argument in WCC RIDC calls. As the Document Title has no length constraints, it is the primary attribute users can use to identify a document. For owned documents, the default Document Title is set to:

```
OTM-<document owner type> <document owner>-<document type>2
```

where the *document owner type* is the type of business object that owns the document (e.g. SHIPMENT). The *document owner* is the unique ID (GID) of the business object. The *document type* is the document type GID (e.g. BILL OF LADING). For unowned documents, the default Document Title is set to:

```
OTM-<document GID>
where the document GID is the unique identifier of the document within OTM,
including the domain.
```

The defaults for the Document Title assume that the implementation uses the full versioning support inherent in WCC. Consider the generation of a shipment Bill of Lading. Since the Document Title is only unique by shipment and document type, all bills of lading for the shipment will share the same Document Title. Any standard revisions or regeneration of the bill of lading will result in a new version being added to the existing CMS document. The user sees a single bill of lading with the name **OTM-SHIPMENT <shipment GID>-BILL OF LADING**. By viewing the details of the document, the user can see each revision and open up its contents.

To suppress the default versioning support in WCC, uncheck the **Use Versioning** checkbox on the Content Management System record corresponding to the WCC instance. Then modify the Document Title parameter generator to use the **WCC_DOCTITLE.DEFAULT** business number rule. This rule sets the Document Title to:

```
OTM-<document owner type>-<document owner>-<document type>-nnnn
```

where *nnnn* is a sequence for the particular owner and document type. Each revision of the document is then represented by a unique WCC document. Note that if versioning support in WCC is suppressed but the default Document Title parameter generator is not reset; adding a revision to a standard document will overwrite content of the older revision.

Folder

WebCenter Content supports unfiled documents so a folder specification is not needed for simple storage.

---

[1] This is based on the use of the **WCC_DOCNAME** Parameter Generator.

[2] This is based on the use of the **WCC_DOCTITLE.VERSIONS** Business Number rule as a default generator for Document Title.

Oracle Documents Cloud

GUID

For Oracle Documents Cloud, the GUID is generated by Oracle Documents Cloud itself for each and every object created on the server. This is a standard unique GUID for all the objects getting stored in Oracle Documents Cloud and is generally hidden from the user. This GUID is used by Oracle Transportation system while trying to perform any operation on any object in Oracle Documents Cloud.

Primary Name

The primary name for ODC content is given by the Document Title parameter, corresponding to the **primaryFile** argument in REST calls. The value for this **primaryFile** parameter is formed based pn the generator provided in the Document Title Generator.

As the Document Title has no length constraints, it is the primary attribute users can use to identify a document. For owned documents, the default Document Title is set to:

```
OTM-<document owner type> <document owner>-<document type>3
```

where the **document owner type** is the type of business object that owns the document (e.g. SHIPMENT). The **document owner** is the unique ID (GID) of the business object. The **document type** is the document type GID (e.g. BILL OF LADING). For unowned documents, the default Document Title is set to:

```
OTM-<document GID>
```

where the document GID is the unique identifier of the document within OTM, including the domain.

The defaults for the Document Title assume that the implementation uses the full versioning support inherent in ODC. Consider the generation of a shipment Bill of Lading. Since the Document Title is only unique by shipment and document type, all bills of lading for the shipment will share the same Document Title. Any standard revisions or regeneration of the bill of lading will result in a new version being added to the existing CMS document. The user sees a single bill of lading with the name **OTM-SHIPMENT <shipment GID>-BILL OF LADING**. By viewing the details of the document, the user can see each revision and open up its contents.

To suppress the default versioning support in ODC, uncheck the **Use Versioning** checkbox on the Content Management System record corresponding to the ODC instance. Then modify the Document Title parameter generator to use the **ODC_DOCTITLE.DEFAULT** business number rule. This rule sets the Document Title to:

```
OTM-<document owner type>-<document owner>-<document type>-nnnn
```

where *nnnn* is a sequence for the particular owner and document type. Each revision of the document is then represented by a unique ODC document. Note that if versioning support in ODC is suppressed but the default Document Title parameter generator is not reset; adding a revision to a standard document will overwrite content of the older revision.

---

[3] This is based on the use of the **ODC_DOCTITLE.VERSIONS** Business Number rule as a default generator for Document Title.

Copyright © 2014, 2021, Oracle and/or its affiliates.

Folder

Oracle Documents Cloud supports unfiled documents so a folder specification is not needed for simple storage.

Storage Workflow

Figure 2-1: Document Storage Workflow summarizes how the CMS and its settings impact how Oracle Transportation Management stores content.
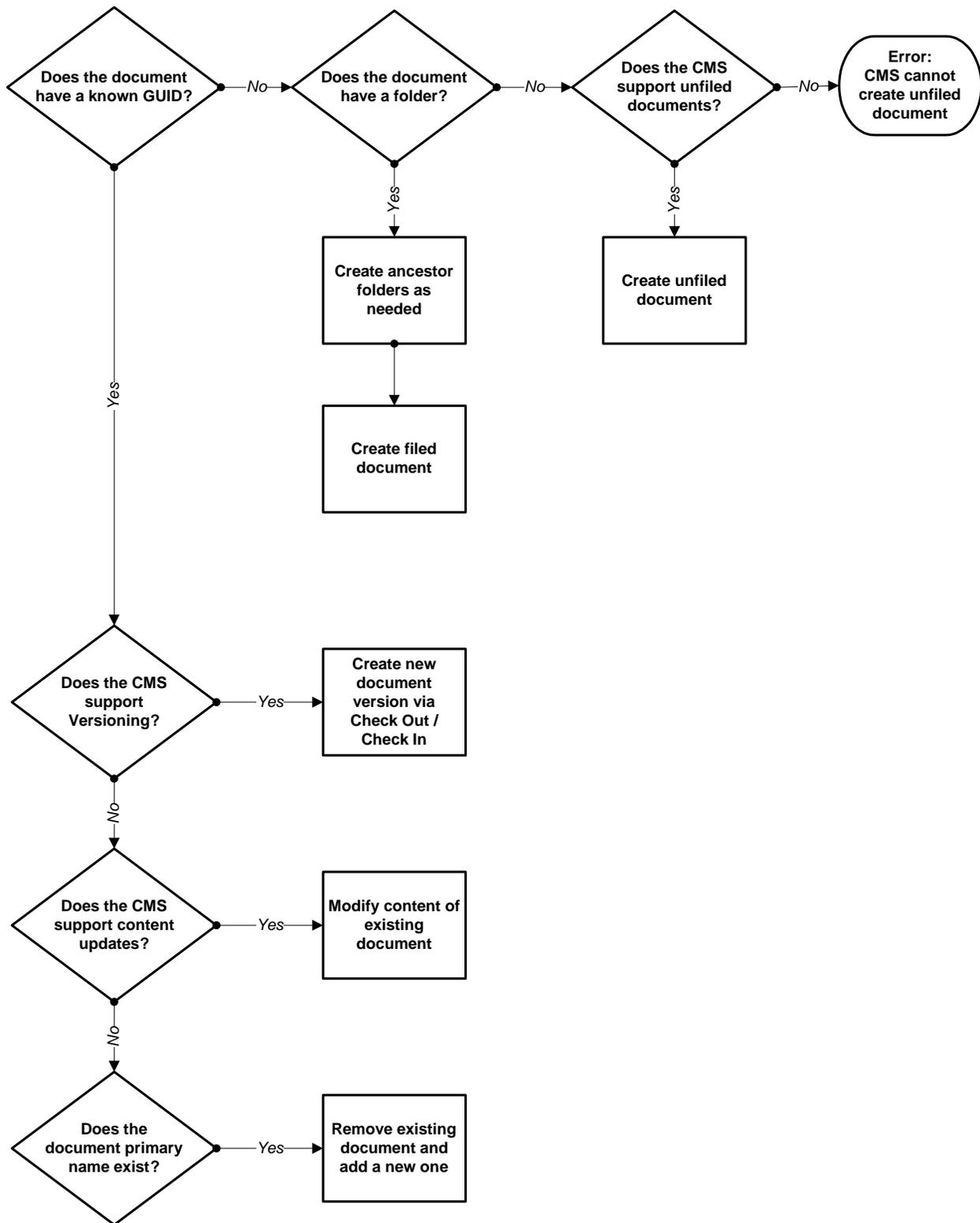
Figure 2-1: Document Storage Workflow

Storage Lifetime Management

For many implementations, the lifetime of document content is generally managed by the CMS. This allows important documents to exist long past the lifecycle of their transactional owners in Oracle Transportation Management. For example, it may be critical for commercial invoices to be stored and queried years after their shipments have been removed. Using Oracle Transportation Management's schema to store the invoice, it would be deleted when the shipment is deleted. Using an external CMS, the invoice can be stored based on business rules independent of immediate transportation needs.

There are use cases, however, where maintaining documents in the CMS can add unnecessary overhead. For example, assume you have a PICKUP RECEIPT document type associated with shipments. A user may upload a PICKUP RECEIPT to the system and realize it is incorrect. He then uploads an updated copy of the same receipt. Depending on the versioning support and setup of the CMS, both versions of the PICKUP RECEIPT may be stored in the CMS for an extended period when only the latest version is useful.

When removing content links from the system, Oracle Transportation Management categorizes the activity by use case. If the use case is managed by the CMS, the link is removed but the content remains in the CMS; if not, the content is removed from the CMS. These use cases include:

- **preview**: the link was used for a temporary, asynchronous preview.
- **overlay**: the link was intentionally overlaid by a user (e.g. a user uploads new content into a document).
- **revision**: the link was to a revision explicitly removed by the user. Revisions are available for Standard Documents in Oracle Transportation Management.
- **user**: the link was explicitly removed by the user from the Document finder.
- **owner**: the link was removed indirectly when its owner was removed.

A property controls the system responsible for content removal. Its format is:

```
glog.document.external.persistRemoveCases=<comma-delimited list of use cases>
```

where the default setting is:

```
glog.document.external.persistRemoveCases=preview,overlay,revision,user
```

This means that content is removed from the CMS when it was 1) used for a temporary preview or 2) manually deleted or overlaid by a user. When a shipment is removed, the shipment documents are not removed from the CMS. It is the responsibility of each CMS to manage document content rules, ensuring scalability as the number and size of documents grow.

3. How to Configure Content Organization

To setup a CMS for storage and content organization, first perform the steps from Section 2 to create a Content Management System for storage in the CMS. Then, proceed with the following.

Preparing the CMS for Content Organization

Certain custom installation steps are necessary to enable CMS features for content organization. These steps need to be run by an administrator of the CMS.

WebCenter Content

A number of features in WebCenter Content can be leveraged to better identify and secure Oracle Transportation Management documents within WCC. These features are optional, but require modifications to your WCC configuration.

- **Document Filing**: Oracle Transportation Management can support the filing of documents into WCC Framework Folders. Contribution Folders (i.e. folders_g) are a deprecated feature in WCC and are not supported by the integration layer. To enable Framework Folders, follow the instructions at https://docs.oracle.com/cd/E23943_01/doc.1111/e10978/c08_folders.htm#CSMRC1540 and https://docs.oracle.com/cd/E23943_01/doc.1111/e10792/toc.htm.

- **Security Groups**: The initial security groups available in WCC are **Public** and **Secure**. Any user can view a document in the **Public** group. Only administrators can view a document in the **Secure** group. Since Oracle Transportation Management assigns all documents to the **Secure** group by default, its documents are not available for viewing or searching in WCC. To allow transportation documents to be viewed as an enterprise document in WCC, either assign the documents to the Public security group or create one or more security groups for these documents. For more information on creating custom security groups in WCC, see http://docs.oracle.com/cd/E23943_01/doc.1111/e10792/c05_security.htm#BGBEAIFH

- **Accounts**: WebCenter Content supports a more granular level of document security than security groups. Hierarchical accounts can be set up such that a user given access to an account can view documents in that account or child accounts. To distribute documents across accounts and setup valid users for those accounts requires coordination between WCC and Oracle Transportation Management. For more information on creating accounts in WCC, see http://docs.oracle.com/cd/E23943_01/doc.1111/e10792/c05_security.htm#BGBDIFIJ.

Oracle Documents Cloud

No additional installation steps are necessary to support document filing in Oracle Documents Cloud. Note that Oracle Transportation Management release 6.4.2 does not support any custom security in Oracle Documents Cloud.

Determining a Document's Folder for Content Organization

The Content Management System manager provides three options for determining the folder assignment for a document:

- **Fixed**: All documents are stored in a root folder.
- **Domain**: Documents are stored in subfolders off of the root folder. The subfolder is determined by the document's domain. E.g., assume the document domain is **CLIENTS/YELLOW** and the root folder is **OTM**. Then the document is stored in a hierarchy of folders given by **OTM/CLIENTS/YELLOW**.

- **Generated**: Documents are stored in a subfolder based on a parameter generator. This allows folder assignment to be driven by data on the document or its owner. See Section 5 for details on setting up parameter generators.

4. How to Configure Custom Search

To set up a CMS to enable searches on custom Oracle Transportation Management attributes, first perform the steps from Sections 2 and 3 to create a Content Management System for storage and content organization. Then, proceed with the following.

Preparing the CMS for Custom Search

Certain custom installation steps are necessary to enable CMS features for content organization. These steps need to be run by an administrator of the CMS.

WebCenter Content

WCC supports the use of custom document parameters when searching and displaying documents. Two options are available from the WCC Configuration Manager applet:

- Adding custom search parameters to the list of information fields. These fields are automatically added to the default search and document view pages. For more information adding information fields, see http://docs.oracle.com/cd/E23943_01/doc.1111/e10978/c03_repository.htm#DAFIGAJA.

- Adding custom search parameters to the list of additional fields. These fields are available to rules and profiles. A profile can be associated with specific users or documents so that transportation-specific fields are only searchable and viewable on custom search and view pages. For more information adding additional fields, see http://docs.oracle.com/cd/E23943_01/doc.1111/e10978/c03_repository.htm#DAFEBHCI. For information tying these additional fields to profiles and rules, see http://docs.oracle.com/cd/E23943_01/doc.1111/e10978/c04_metadata.htm#DAFIIEEI.

Note that any parameters added to information fields or additional fields should be prefixed with a lower-case x in Oracle Transportation Management. WCC maps any document parameters starting with **x** to custom fields. E.g., to add a **LogisticsParty** field to WCC, add an **xLogisticsParty** parameter to the Content Management System record.

Oracle Documents Cloud

ODC does not support the use of custom document parameters when searching and displaying documents.

5. Document Parameters

Both standard and custom information is passed from Oracle Transportation Management to the CMS via document parameters. A document parameter is simply a key and value attached to the Oracle Transportation Management document record where the value is:

- generated when the document is created or modified[4]; or
- manually entered by a user on the Document manager

Parameters may be used to drive internal API calls to the CMS or sent directly to the CMS.

Parameter Generation

The following types of parameter generation are supported:

- **Business Number Rule**: The parameter value is calculated from a business number rule in the domain of the document. A business rule provides the ability to combine fixed strings, information regarding the document or its owner, and numeric sequences into an identifier for the document. Business rules can also be used to retrieve simple data from the document owner like a reference number or involved party.
- **SQL**: The parameter value is retrieved from a SQL select statement. The SQL may include a single bind parameter representing the document owner.
- **Constant**: The parameter value is fixed.
- **Java**: A special type of **JAVA_PLUGIN** can be added that generates a parameter using custom Java code.

Updating Parameters with Current Information

Document parameters may be immutable. Once generated on creation, an immutable parameter is never updated. This is important for parameters that act as unique identifiers on the document.

Other parameters, however, may reflect information regarding the document or its owner (e.g. a shipment pickup location). This information may change as transactional data is updated in the system. By default, document parameters are only updated when changes are made to the document. It is possible, though, to trigger an update of all mutable document parameters when changes are detected in the document's owner.

- A user may manually invoke the **Update Document Parameters** action from the Document manager, the manager for the document owner (e.g. the Shipment manager) or the Documents tab on the owner.
- An automation agent may be set up to listen to events on the document owner and, using a data type association[5], run the **UPDATE DOCUMENT PARAMETERS** agent action for each owned document.
- An automation agent may be set up to list to events on the document owner and run the **UPDATE STANDARD DOCUMENT PARAMETERS** agent action to update the parameters for a particular standard document type.

Any time document parameters are updated, they are resent to the CMS to synchronize CMS data with Oracle Transportation Management data.Reserved Parameters.

---

[4] Parameters can be optionally regenerated when the business object owning the document changes. See Updating Parameters with Current Information.

[5] E.g., **SHIPMENT TO DOCUMENT**

For Oracle Documents Cloud (ODC), only the name(title) of the document can be updated through the parameter **dDocTitle**.Other parameters of the document uploaded into the Oracle Documents Cloud server are immutable.

Table 5-1: Standard Parameter Mappings shows how standard parameter generators are mapped to the CMS based off fields in the Content Management System record.

<p align="center">**Table 5-1: Standard Parameter Mappings**</p>

CMS Type	Parameter Name	Mutable	Notes	Parameter Source	Default Value
WCC	**dDocName**	N	Holds the GUID for WCC. Must be < 30 characters and globally unique across the enterprise.	**Document Name Generator**	Parameter Generator: **WCC_DOCNAME** This is a SQL generator, basing the value on the **DOCUMENT_CMS_SEQ** sequence. Values are of the form **OTM-*nnnnnnn*** where *nnnnnnn* is the next sequence value left padded with zeroes.
	dDocTitle	N	Holds the display name for the document. Documents sharing display name are stored as versions of a single document.	**Document Title Generator**	If the document has an owner, a business number generator based on rule **WCC_DOCTITLE.VERSIONS**. This produces values of the form **OTM-*ownerType owner-docType*** where *ownerType* and *owner* are the business object type and GID of the owner. E.g. SHIPMENT S2013-11-23-001. The *docType* is the document type. E.g. BILL OF LADING. Note that this default value assumes the CMS has versioning support. If the document has no owner, **OTM-*documentGID*** where *documentGID* is the ID of the document record.
	dDocAccount	Y	Only added if Use Accounts is checked. Account must exist within WCC	**Account Generator**	The domain name of the document.

CMS Type	Parameter Name	Mutable	Notes	Parameter Source	Default Value
	dDocAuthor	Y		**User**	The user of the Content Management System.[6]
	dSecurityGroup	Y	Security group must exist within WCC	**Security Group**	**Secure**
	primaryFile	N	Used to determine to add the document to a WCC contribution folder	**Folder Type, Root Folder, Document Filename**	The specified folder and document filename.
ODC	**dDocTitle**	N	Holds the display name for the document. Documents sharing display name are stored as versions of a single document.	**Document Title Generator**	If the document has an owner, a business number generator based on rule **ODC_DOCTITLE.VERSIONS**. This produces values of the form **OTM-*ownerType owner-docType*** where *ownerType* and *owner* are the business object type and GID of the owner. E.g. SHIPMENT S2013-11-23-001. The *docType* is the document type. E.g. BILL OF LADING. Note that this default value assumes the CMS has versioning support. If the document has no owner, **OTM-*documentGID*** where *documentGID* is the ID of the document record.
	primaryFile	N	Used to determine to add the document to a ODC folder	**Folder Type, Root Folder, Document Filename**	The specified folder and document filename.

Each of these standard parameters is driven by fields on the Content Management System Manager. They apply to every document stored with that CMS.

---

[6] Since WCC requires the **dDocAuthor** to be a valid WCC user, you cannot easily audit document activity via the logged in user. Activity can be tracked via domain (via the account). Alternatively, multiple Content Management Systems can point to the same CMS but with different CMS users. This gives more granularity to auditing.

Additional Parameters

Additional parameters can be generated for a CMS. The **Additional Parameters** grid on the Content Management System Manager allows a user to add a parameter, specify its generator and note whether it is mutable. These additional parameters are generated for every document stored with the CMS and sent to the CMS when the document is created or updated.

WebCenter Content

WCC allows arbitrary parameters to be passed along with most IDC calls. Parameters not used by WCC are ignored without warning or error. To use a parameter in WCC, a user should prefix custom properties with a lower case **x**. For example, **xShipFrom** could represent the shipment pickup location for a bill of lading.

Oracle Documents Cloud

Oracle Transportation Management does not support the use of additional document parameters by Oracle Documents Cloud for search or display.Parameter Precedence.

Parameter Precedence

The parameters listed in Table 5-1 assume that parameters are associated with a Content Management System and apply to all documents stored in that system.

Parameter generation, however, can be defined or overridden at the document type level. Assume, for example, a WCC content management system record has the security group set to **Public**. All documents, by default, are visible by all users. For commercial invoices, though, you want the security group set to **Financials**, a custom security group you have added to WCC. This can be done by editing the **COMMERCIAL INVOICE** document type[7] and adding a Content Management System Parameter with name **dSecurityGroup**. The generator should be set to a **Fixed** generator with value **Financials**. The parameter generator defined at the **COMMERCIAL INVOICE** level overrides the one specified on the Content Management System.

Parameters may also be set directly on the document object. For example, you could have edited a specific commercial invoice[8] and modified the **dSecurityGroup** parameter to set its value to **Financials**. This is not recommended for standard parameters, or parameters defined on the document type or Content Management System as any modification to the document will regenerate parameter values and overwrite your changes.

In general, then, the precedence of parameter definitions is:

1. Document
2. Document Type
3. Content Management System

---

[7] Business Process Automation > Power Data > Document Generation > Document Types

[8] Business Process Automation > Document Manager

Parameter Generation Reference Information

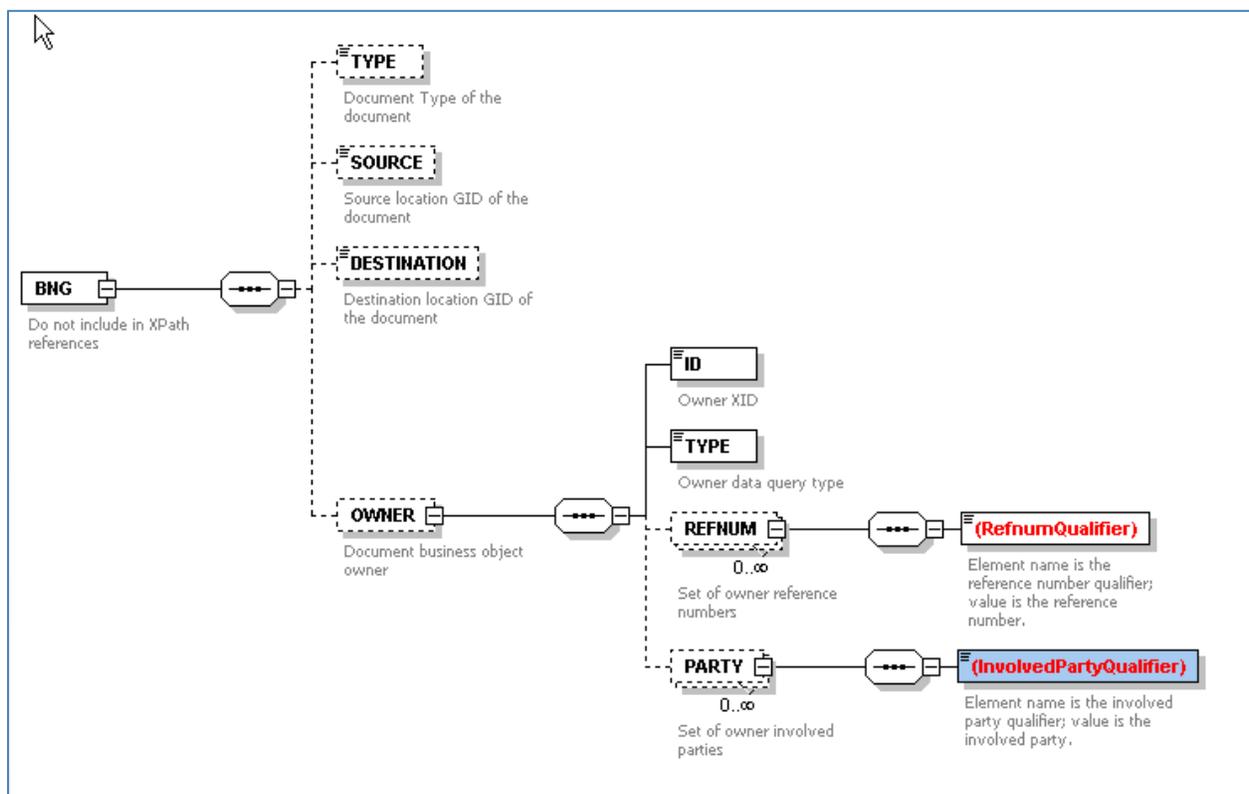### Schema for Business Number Parameter Generators

The business number parameter generator uses a business number (BN) rule to define a pattern for the parameter value. A special replacement pattern segment (**r***) is available to allow document and owner information to be part of the final value. The information is provided as an XML document and can be used with rudimentary XPath-like expressions in the pattern. Refer to online help for a comprehensive discussion of business number rules.

All business number rules used for document parameters must have the BN Type of **CMS_FIELD**. This, in turn, uses the **DOCUMENT_DEF_XID** BN Context to generate replacement information. The schema for the information is shown in Figure 5-1.

Assume you wanted to pass a document parameter for the logistics party. By using a BN Rule with pattern:

```
{r*:xml=OWNER/PARTY/LOGISTICS}
```

you extract out the LOGISTICS party from the list of involved parties associated with the owner of the document.

Figure 5-1: Schema for CMIS_FIELD BN Rules

Business Number Parameter Generators and Domains

In Oracle Transportation Management, a business number rule is identified by its XID. When executing the rule, the system checks for a BN rule record in the current domain with an XID matching the rule.

This has the advantage that a single rule can vary patterns by domain. But it increases the complexity of defining a BN rule to be used across many domains.

Consider an example where you want to apply a BN rule, **LOGISTICS,** to populate an **xLogisticsParty** parameter on all applicable documents. Assume you have a **PUBLIC** Content Management System **WCC**, representing the WebCenter Content instance that will receive the custom parameter. To assign a parameter generator for **xLogisticsParty**, you create a **PUBLIC** parameter generator **LOGISTICS** that maps to a **PUBLIC** BN rule of **LOGISTICS**. But when the rule runs, it searches for the **LOGISTICS** rule in the domain of the document. So you need to duplicate the **PUBLIC LOGISTICS** rule in every domain it may run in.

SQL Statement Assumptions for SQL Parameter Generators

A SQL parameter generator executes a single-column SQL **select** statement and retrieves the first record of the result set. It returns the column value for the result set as the parameter value. The SQL statement optionally accepts the document owner GID as a single bind parameter.

Examples:

- `select 'OTM-' || document_cms_seq.nextval from dual`
- `select shipment_refnum_value from shipment_refnum where shipment_refnum_qual_gid = 'BOL' and shipment_gid=?`

Interface Requirements for Java Parameter Generators

For document parameters too complex for a Business Number or SQL rule, the system can be extended with a Java plugin to delegate parameter generation to a custom Java class. This is recommended only for advanced Oracle Transportation Management implementation teams.

To create a Java parameter generator,

- create a Java plugin of type **Parameter Generation**. This is accessed from **Configuration and Administration > Power Data > General > Java Plugins**.
- the Java class specified in the plugin must implement `glog.server.document.cms.CmsParamJavaPlugin`.
- create a Document Parameter Generator of type **Java Plugin**. The created plugin should appear in the drop-down list.

The **CmsParamJavaPlugin** interface requires implementation of a single **generate** method:

```
public String generate(String ownerDataQueryType, String ownerGid,
                T2SharedConnection conn) throws GLException;
```

Given the document owner and a database connection, the method returns the parameter value.

Note that custom plugins should be added sparingly. Future releases of Oracle Transportation Management may alter the base interface, requiring a migration effort for any custom code.

6. Reference

Properties Impacting CMS Integration

The following properties control CMS Integration:

- **glog.defaultMimeTypes.:** These are used by Oracle Transportation Management documents (as in the document manager). It maps file extensions to a mime type (an application). When using attachments with Oracle Transportation Management documents, this tells Oracle Transportation Management what application to use based on the extension of the attached file.

- **glog.document.attachment.maxSize:** While configuring a Stylesheet Profile, you can choose to embed a document. Document content is embedded in the notification. The attachment must be the same size or less of the bytes specified by this property. If not, Oracle Transportation Management Link is used.

- **glog.document.defaultCMS**: the default Content Management System. Report storage, document types and ad-hoc uploads will default to this CMS. After installation, it is set to **DATABASE**, storing document content in the Oracle Transportation Management schema. If all document content should default to a given external CMS, this property should be modified to reference it.

- **glog.document.external.appendExtension**.: Many CMS products provide standard URI links for document content. If you click on one of these links, the browser may use the file extension of the file specified by the link to render content. To support this use case, OTM optionally appends the file extension to the uname, based on the content's MIME type. These properties determine whether OTM appends the file extension to any uname it sends to a CMS.

    A mapping of MIME type to preferred file extension is available in glog.defaultMimeTypes property listed above.

    - o **glog.document.external.appendExtension.WCC**: a flag to determine whether the document file extension should be appended to **dDocTitle** in WCC. This is necessary for some browsers to match up the document with its proper renderer.

    - o **glog.document.external.appendExtension.ODC**: a flag to determine whether the document file extension should be appended to **dDocTitle** in ODC. This is necessary for some browsers to match up the document with its proper renderer.

- **glog.document.external.persistRemoveCases**: a comma-delimited list of use cases where removal of a document link in Oracle Transportation Management should delete the corresponding document in the CMS. This property controls the deletion of remote content by OTM. If your CMS does not support document removal, or one to which OTM does not have rights to remove documents, the property should be set to blank to avoid removal errors.

Valid use cases include:

- **preview**: Content was created to hold a temporary preview. Once viewed, the content should be removed.

- **overlay**: Content was intentionally overlayed by the user by uploading new content into the Document manager.

- **revision**: The user opted to remove a Standard Document revision.

- **user**: The user opted to delete a Document.

- **owner**: OTM removed a document when its owner was removed.

By default, OTM removes external content for preview, overlay, revision and user. This is to avoid the growth of unnecessary content on the remote CMS due to ad-hoc reports and erroneous revisions.

- **glog.document.retrieveURLMimeType:** Determines if the URL should be accessed to set the content type on the document. The property defaults to true for backward compatibility.

Logging CMS Integration Activity

The following log IDs can help diagnose problems with CMS integration:

- **ContentManagement**: Provides detailed logging of storage workflow decisions.

-

- **WCCCalls**: Logs session information for communication with WCC. Includes parameter details on each WCC call and response message.

- **ODCCalls**: Logs information for communication with ODC. Includes parameter details on each ODC REST call and response message in XML format.