

docuflow Guides

CONTENTS

1. Welcome	7
2. What's New	8
3. Repository Specific Guides	12
3.1. docuflow for Box	12
3.2. Untitled	13
3.3. docuflow for FileNet	14
3.4. Untitled	15
4. Agnostic Guides	16
4.1. Solution Guide	16
4.1.1. Purpose of Guide	17
4.1.2. Customer Resources Required	18
4.1.3. Configuration Overview	19
4.1.4. Viewing and Attachment List Options	21
4.1.5. DocManager	23
4.1.5.1. Creating a DocManager Rule	25
4.1.5.2. Enabling Store New and Assign / Attachment List Menu Path	32
4.1.5.3. Required Metadata Properties	34
4.1.5.4. SAPLink / Inbound to SAP Processing	37
4.1.5.4.1. Using Microsoft Graph Explorer to Debug SharePoint Filters in Link Enterprise	43
4.1.5.5. Attachment List Options	45
4.1.5.6. Delete Link Management via docuflow Link Manager	51
4.1.5.7. SAP Logon Pad Security Configuration	56
4.1.5.8. Fiori DocManager - SAP Fiori Launchpad Plugin (cloned) (cloned)	58
4.1.6. Custom API Usage	64
4.1.7. Choice Lists	65
4.1.8. Workflow Connect	66
4.1.9. AuditLink	67
4.1.10. DocMigrator	68
4.1.10.1. Migration Table Flags	70
4.1.10.2. Configuration	71
4.1.10.3. Execution: Load	74
4.1.10.4. Execution: Migration	77
4.1.10.5. Post-Processing Tasks	80
4.1.10.6. Additional Utilities	81
4.1.10.7. Migration Concepts and Strategies	82
4.1.10.8. Box - Additional Best Practices	86
4.1.11. Accounts Payable	87
4.1.11.1. Functional Overview	89
4.1.11.2. Configuration	93
4.1.11.2.1. Classification	95
4.1.11.2.2. Invoice Processing Rules	98
4.1.11.2.3. Inbox Mapping	101
4.1.11.2.4. Business Rules	102
4.1.11.2.5. Rejection Codes	103
4.1.11.2.6. Exceptions	104
4.1.11.2.7. Agents: Core	106
4.1.11.2.8. Agents: Approvers (ASL)	108
4.1.11.2.9. Agents: Substitutions and Delegations	114
4.1.11.2.10. Agents: Email Mapping	115
4.1.11.3. Working with TrackLink	116
4.1.11.4. Customizing	120
4.1.12. Publisher	123
4.1.13. System Messages	124
4.1.14. Utilities: CMS Load (Beta)	125
4.1.15. Utilities: Link Maintenance	126
4.1.16. Utilities: Metadata Push (MDPro)	127
4.1.17. Utilities: Cleaning Gateway Errors	128
4.1.18. Utilities: Archiving Gateway Data	130
4.1.19. Utilities: Attachment List Dashboard	131
4.1.20. Utilities: Determine Attachments	132
4.1.21. Working with the API Gateway and SCRUD Test Harness	134
4.2. Container Deployment Guide	138
4.2.1. Step by Step Deployment OCP	140

4.3. Developers Guide	143
4.3.1. Purpose of Guide	144
4.3.2. Class/Method Plug-In's	145
4.3.2.1. DocManager Metadata maps	146
4.3.2.2. DocManager Inbound Processing	148
4.3.2.3. DocMigrator Custom Source Link Determination	151
4.3.2.4. Publisher: Triggering Storing and Linking	152
4.3.3. docuflow BADI's	155
4.3.3.1. Base Content API	156
4.3.3.2. DocManager	157
4.3.4. SCRUD API's	160
4.3.4.1. CMS_API_SEARCH	161
4.3.4.2. CMS_API_CREATE	168
4.3.4.3. CMS_API_READ	177
4.3.4.4. CMS_API_UPDATE	181
4.3.4.5. CMS_API_DELETE	185
4.3.5. Ping and Load API's	188
4.3.5.1. CMS_API_PING	189
4.3.5.2. CMS_API_LOAD	192
4.3.6. docuflow SCRUD Web Services	195
4.3.6.1. User Permissions	196
4.3.6.2. HTTPS Setup (JKS/PEM) (cloned)	197
4.3.6.3. Box Endpoints	198
4.3.6.3.1. Upload	198
4.3.6.3.2. Search	201
4.3.6.3.3. Create	205
4.3.6.3.4. Read	218
4.3.6.3.5. Update	223
4.3.6.3.6. Delete	226
4.3.6.3.7. Tasks	227
4.3.6.3.8. Sign (available version > 6.6.5)	232
4.3.7. docuflow BAPI Web Services	239
4.3.8. Useful / Common docuflow Methods	241
4.3.8.1. Store and Link a Document	242
4.3.8.2. Send Metadata	247
4.3.9. Overall CMS API Notes	248
4.3.10. SAP Code Vulnerability Analyzer (CVA)	249
4.3.11. log4j Update	250
4.4. VersaFile Release Management	251
4.4.1. Release Schedule	252
4.4.2. Source Code Review	253
4.4.3. Vulnerability Assessment	254
4.4.4. Security Testing	255
4.4.5. Managing Security Issues	256
4.5. Change Log	257
4.5.1. SAP	258
4.5.1.1. 3.1 (beta)	259
4.5.1.2. 3.0	261
4.5.1.3. 2.9	262
4.5.2. Middleware	263
4.5.2.1. 6.5.7	264
5. SuccessFactors	266
5.1. Agnostic Guide	266
5.1.1. SuccessFactors Technical Overview	267
5.1.1.1. Introduction	268
5.1.1.2. Architecture Overview	269
5.1.1.3. Minimum Requirements	270
5.1.2. BTP Configuration	271
5.1.2.1. Linking SF to BTP Landscape	272
5.1.2.1.1. BTP Sub Account Creation	276
5.1.2.2. HTML5 Application Registrations	277
5.1.2.2.1. BTP Command Line Interface (CLI)	286
5.1.2.3. Extension Admin Permissions	287
5.1.2.4. SuccessFactors App Setup	288
5.1.3. SuccessFactors Configuration	293
5.1.3.1. API User	294

5.1.3.2. Create Pick List	296
5.1.3.3. Create Approval Workflow	297
5.1.3.4. SuccessFactors Objects	298
5.1.3.4.1. Category Link Card 1	299
5.1.3.4.2. Category Link Card 3	300
5.1.3.4.3. Document Type Link Card 1	301
5.1.3.4.4. Document Type Link Card 3	302
5.1.3.4.5. Link Card Type 1	303
5.1.3.4.6. Link Card Type 2	304
5.1.3.4.7. Link Card Type 3	305
5.1.3.4.8. Link Card Type 4	306
5.1.3.4.9. ESS Link Card Child	308
5.1.3.4.10. ESS Link Card Parent	311
5.1.3.4.11. HR Link Card Child	313
5.1.3.4.12. HR Link Card Parent	315
5.1.3.4.13. d4iFolderLink	317
5.1.3.5. Permission Provisioning	318
5.1.3.6. Export Category and Document Type	321
5.1.3.7. Import Category and Document Types	323
5.1.3.8. UI Configuration	324
5.1.4. docuflow for Integrations (d4i) Middleware Configuration	326
5.1.4.1. Start & Stop Order	327
5.1.4.2. docuflow Middleware Installation	328
5.1.4.2.1. Unpacking ZIP Package	329
5.1.4.2.2. Initial Configuration	330
5.1.4.2.3. Setting up Firefox Certificate	333
5.1.4.2.4. Connecting to docuflow Configuration	337
5.1.4.2.5. SSL Certificates Modifications	338
5.1.4.3. docuflow Base Configuration	342
5.1.4.3.1. Engines	343
5.1.4.3.2. Connectors	344
5.1.4.3.2.1. Box	346
5.1.4.3.2.2. IBM FileNet	347
5.1.4.3.2.3. Filesystem	348
5.1.4.3.3. Connections	350
5.1.4.3.3.1. Box	351
5.1.4.3.3.2. IBM FileNet	355
5.1.4.3.3.3. Filesystem	359
5.2. SuccessFactors for Box	360
5.2.1. SuccessFactors Technical Overview	360
5.2.1.1. Introduction	361
5.2.1.2. Architecture Overview	362
5.2.1.3. Minimum Requirements	363
5.2.2. BTP Configuration	364
5.2.2.1. Linking SF to BTP Landscape	365
5.2.2.1.1. BTP Sub Account Creation	369
5.2.2.2. HTML5 Application Registrations	370
5.2.2.2.1. BTP Command Line Interface (CLI)	379
5.2.2.3. Extension Admin Permissions	380
5.2.2.4. SuccessFactors App Setup	381
5.2.3. SuccessFactors Configuration	386
5.2.3.1. API User	387
5.2.3.2. Create Pick List	389
5.2.3.3. Create Approval Workflow	390
5.2.3.4. SuccessFactors Objects	391
5.2.3.4.1. Category Link Card 1	392
5.2.3.4.2. Category Link Card 3	393
5.2.3.4.3. Document Type Link Card 1	394
5.2.3.4.4. Document Type Link Card 3	395
5.2.3.4.5. Link Card Type 1	396
5.2.3.4.6. Link Card Type 2	397
5.2.3.4.7. Link Card Type 3	398
5.2.3.4.8. Link Card Type 4	399
5.2.3.4.9. ESS Link Card Child	401
5.2.3.4.10. ESS Link Card Parent	404
5.2.3.4.11. HR Link Card Child	406

5.2.3.4.12. HR Link Card Parent	408
5.2.3.4.13. d4iFolderLink	410
5.2.3.5. Permission Provisioning	411
5.2.3.6. Export Category and Document Type	414
5.2.3.7. Import Category and Document Types	416
5.2.3.8. UI Configuration	417
5.2.4. docuflow for Integrations (d4i) Middleware Configuration	419
5.2.4.1. Start & Stop Order	420
5.2.4.2. docuflow Middleware Installation	421
5.2.4.2.1. Unpacking ZIP Package	422
5.2.4.2.2. Initial Configuration	423
5.2.4.2.3. Setting up Firefox Certificate	426
5.2.4.2.4. Connecting to docuflow Configuration	430
5.2.4.2.5. SSL Certificates Modifications	431
5.2.4.3. docuflow Base Configuration	435
5.2.4.3.1. Engines	436
5.2.4.3.2. Connectors	437
5.2.4.3.2.1. Box	439
5.2.4.3.2.2. Filesystem	440
5.2.4.3.3. Connections	442
5.2.4.3.3.1. Box	443
5.2.4.3.3.2. Filesystem	446
5.3. SuccessFactors for FileNet	447
5.3.1. SuccessFactors Technical Overview	447
5.3.1.1. Introduction	448
5.3.1.2. Architecture Overview	449
5.3.1.3. Minimum Requirements	450
5.3.2. BTP Configuration	451
5.3.2.1. Linking SF to BTP Landscape	452
5.3.2.1.1. BTP Sub Account Creation	456
5.3.2.2. HTML5 Application Registrations	457
5.3.2.2.1. BTP Command Line Interface (CLI)	466
5.3.2.3. Extension Admin Permissions	467
5.3.2.4. SuccessFactors App Setup	468
5.3.3. SuccessFactors Configuration	473
5.3.3.1. API User	474
5.3.3.2. Create Pick List	476
5.3.3.3. Create Approval Workflow	477
5.3.3.4. SuccessFactors Objects	478
5.3.3.4.1. Category Link Card 1	479
5.3.3.4.2. Category Link Card 3	480
5.3.3.4.3. Document Type Link Card 1	481
5.3.3.4.4. Document Type Link Card 3	482
5.3.3.4.5. Link Card Type 1	483
5.3.3.4.6. Link Card Type 2	484
5.3.3.4.7. Link Card Type 3	485
5.3.3.4.8. Link Card Type 4	486
5.3.3.4.9. ESS Link Card Child	488
5.3.3.4.10. ESS Link Card Parent	491
5.3.3.4.11. HR Link Card Child	493
5.3.3.4.12. HR Link Card Parent	495
5.3.3.4.13. d4iFolderLink	497
5.3.3.5. Permission Provisioning	498
5.3.3.6. Export Category and Document Type	501
5.3.3.7. Import Category and Document Types	503
5.3.3.8. UI Configuration	504
5.3.4. docuflow for Integrations (d4i) Middleware Configuration	506
5.3.4.1. Start & Stop Order	507
5.3.4.2. docuflow Middleware Installation	508
5.3.4.2.1. Unpacking ZIP Package	509
5.3.4.2.2. Initial Configuration	510
5.3.4.2.3. Setting up Firefox Certificate	513
5.3.4.2.4. Connecting to docuflow Configuration	517
5.3.4.2.5. SSL Certificates Modifications	518
5.3.4.3. docuflow Base Configuration	522
5.3.4.3.1. Engines	523

5.3.4.3.2.	Connectors	524
5.3.4.3.2.1.	IBM FileNet	526
5.3.4.3.2.2.	Filesystem	527
5.3.4.3.3.	Connections	529
5.3.4.3.3.1.	IBM FileNet	530
5.3.4.3.3.2.	Filesystem	534

1. Welcome



<https://www.versafire.com>

SAP® Certified
Integration with SAP NetWeaver®

2. What's New

SAP

Under Development

1. docManager for DMS v.2
2. Box Sign Integration enhancements:
 1. linking back of signed copy and log to SAP
 2. specification of due date
 3. multi step signature

3.1

docuflow middleware version required: 6.6.4

1. Expose ABAP SCRUDWL as callable RFC's
2. System Dashboard improvement: Add version number to System Dashboard
3. Bug fix for SAPLink example
4. Bug fix for docMigrator for ArchiveLink report (header disappears after hitting refresh)
5. Updated AP Workflow PRE-POP NON-PO BADI
6. docManager for DMS v.20
 1. various improvements - will be documented upon completion
7. docMigrator for DMS v2.0
 1. Enhanced migrator dashboard
 2. Support both docuflow ECM and ArchiveLink as source
 3. Additional Post-Processing and Utilities, namely validate and rollback functionality
8. Updates to Micro-Process workflow module
 1. Additional config for exceptions
 2. Sample plug-in rule ABAP
 3. Vendor email plug-in for EKKO
 4. Logging polish
9. Box:
 1. Integration with Box Sign
 1. Single step signature
 2. TrackLink integration
10. M365:
 1. Nested folder creation with file placement support
11. Enhanced Activeworx support
 1. Additional data exports
 2. Exposed all data exports as optional RFC enabled function modules

12. docManager ArchiveLink authorizations have been updated

1. Please review your S_WFAR_OBJ authorizations
 2. Storing a document from SAP via docManager now requires 01 activity
 3. To select a document type when storing a document from SAP via docManager, activity 01 is required for it to show up in the drop down list
- futher detail: <https://docs.versafire.com/docuflow-guides/31-beta>

3.0

docuflow middleware version required: 6.6.0

1. SAP CVA Compliance
 2. CMS Load - load all content repository classed, objects, metadata templates, libraries, and properties into SAP
 3. System Dashboard improvements
 1. Speed to display
 2. Show activity for a particular rule
 4. Enhanced Micro-Process Workflow support
 5. SFO/GOS Attachment List refinements
 6. M365:
 1. New flat folder creation with file placement support
 2. Added System Dashboad Load support
 7. Box:
 1. Workflow CRUDPL"W" for Tasks integration:
 1. General Tasks
 2. Approval Tasks
 2. Added File placement support in flat folder creation
 3. Nested folder creation with file placement support
- futher detail: <https://docs.versafire.com/docuflow-guides/30>

2.9

docuflow middleware version required: 6.5.7

1. New Micro-Process Workflow
 2. New docManager Attachment List:
 1. Navigation pane with exapandable folders for document flow, classification, and file type folder navigation
 2. Download support
 3. SAP DB Note creation support
 4. CMS Note creation support
 3. AP Workflow enhancements
 4. Publisher bug fix
- further detail: <https://docs.versafire.com/docuflow-guides/29>

Middleware

Under development

1. M365: Content Server

6.6.1

docuflow SAP version required: 3.0

1. GUI Improvements/Bug Fixes:
 1. Copy functionality on all settings (SAP system, Inbound Processing Rules, etc)
 2. Wider field for filter spring on Inbound Processing Rules
 3. Load support for Box
 4. Added support for configurable keystore password for web services (JKS).
 5. Added support to enable/disable SSL for web services.
 6. Added redirect to home page when refreshing page.
 7. Removed version from config.properties file.
 8. Bug fix with webservises authentication.
 9. Bug fix with SAP SNC path location

6.6.0

docuflow SAP version required: 3.0

1. Box:
 1. File placement support in flat folder creation
 2. Nested folder creation with file placement support
 3. Tasks integration
 1. General Tasks
 2. Approval Tasks
2. M365:
 1. Flat folder creation with file placement support
 2. Load API support

6.5.7

docuflow SAP version required: 2.9

1. Box:
 1. READ API supports folders for viewing URL
 2. Folder URL expiry time now configurable
 3. Auto applying cascade policy on folder creation
 4. Content Server bypass: support for Box object_id
 5. Support for metadata query in SAPLink (previously only supported Box Search API with 10 minute delay)

6. New web browser configuration UI (to replace Profile Wizard)
 7. Support for Box flat folder creation
 8. Kubernetes support
- further detail: <https://docs.versafire.com/docuflow-guides/657>

3. Repository Specific Guides

3.1. docuflow for Box

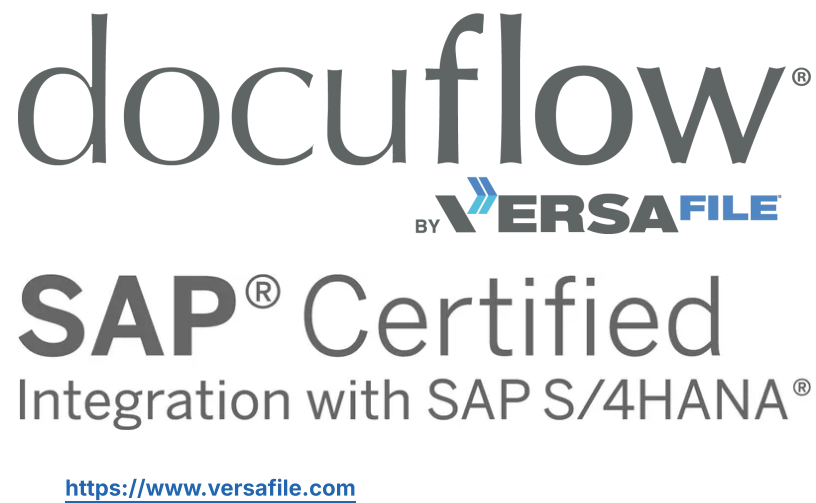
3.2. Untitled

3.3. docuflow for FileNet

3.4. Untitled

4. Agnostic Guides

4.1. Solution Guide



4.1.1. Purpose of Guide

The goal of this guide is to provide detailed configuration of specific integration patterns and docuflow add-ons as well as an overview of all docuflow utilities.

For detailed installation and base configuration, and as a precursor, please refer to the docuflow Installation and Base Configuration Guide (separate).

For details on architecture, supported environments, and system requirements please refer to the docuflow Technical Overview guides below:

1. [Box](#)
2. [SharePoint/M365](#)

For custom development, please refer to the [Developers Guide](#).

4.1.2. Customer Resources Required

Customer resources required for a docuflow configuration and utility usage typically falls on the shoulders of an SAP Basis and/or an SAP technical resource (often an SAP developer who has experience in or nominated to be the document management guru).

Configuration flows between SAP ArchiveLink (OAC2, OAC3) and docuflow SAP configuration, so can be handled by one resource with the appropriate authorizations.

In situations where docuflow plug-ins are utilized (such as more complex meta-data mapping or inbound processing SAP Key manipulation or custom usage), working with your SAP developer (if not that document admin already) will be necessary.

On-going correspondence and collaboration will of course be required between your Middleware and CMS teams for configuration and testing of new patterns.

As new document types are configured, SAP Security may need to grant the necessary authorizations.

4.1.3. Configuration Overview

This document assumes as a precursor that docuflow SAP and middleware has been installed, base configuration has been completed, and that you have connectivity between docuflow SAP ↔ docuflow Connector ↔ CMS System as well as the base repository ID, CMS profiles, tags, destinations and repository ID's ready to roll.

SAP Security is assumed to already be in place for the docuflow transactions and that future SAP authorization for adding and viewing documents (ex: S_WFAR_OBJ) has been considered.

Please refer to the docuflow Installation and Base Configuration Guide (separate) for further information on the above.

For docuflow SAP ABAP plug-in or direct web services development please refer to the Developer guide (separate).

For technical overview please refer to the Technical Overview guide for that particular repository (separate).

The below outlines an overview of the various configuration components of docuflow covered in this guide:

1. **SAPLink:** Provides automated Inbound processing from your CMS system to SAP for a wide variety of integration: automated link creation, triggering of workflows, execute custom ABAP code, and more
2. **DocManager:** Controls outbound document storage (via message output as well as Service for Object) as well as inbound (SAPLink) options such as linking and custom code triggering, as well as metadata transfer in both directions
3. **Custom API Usage:** Typically for use when only the base docuflow API product is installed
4. **ChoiceLists:** Send SAP metadata in the form of flat and hierarchical list for use as drop-down and choice menus in your CMS system
5. **Workflow Connect:** Code free triggering of SAP Workflow including container fulfillment. WFCConnect leverages DocManager for SAPLink inbound processing.
6. **AuditLink:** Utility to extract documents from your CMS system and generate HTML report with drill down to the extracted images for a disconnected from your SAP and CMS system access to an Audit "bundle"
7. **DocMigrator:** Migrate content (and metadata) from various sources to a docuflow ECM endpoint, creating new links, deletion of old ones, as well as various utilities to go with
8. **Accounts Payable:** A powerful yet flexible SAP based Invoice Processing and Approval workflow built on the SAP workflow runtime system.
9. **Publisher:** Routines to allow for message output (SmartForms, AdobeForms, etc.) to be stored with metadata in your CMS system
10. **LinkManager:** Provides an approval process for end users to delete links, with link managers controlling the final deletion as well as the content as an option
11. **Viewing and Attachment List Options:** Configure whether docuflow DocManager is leveraged to store and view content including advanced viewing options, SAP DB Attachment and Note option, and more
12. **System Messages:** Prevent or warn users and API usage when needed during downtime or critical changes across the docuflow platform
13. **Utilities:**
 1. **CMS Load:** Load various CMS system components into SAP (properties, docclasses, etc.)
 2. **Link Maintenance:** Maintain ArchiveLink links
 3. **Metadata Push:** Manually or scheduled metadata updates from SAP to your CMS system based on configured DocManager rules
 4. **Cleaning Gateway Errors:** Maintenance of error in the docuflow API gateway
 5. **Archiving Gateway Data:** Maintenance and archiving of docuflow API gateway log tables that may grow in size over time
 6. **Attachment List Dashboard:** Report on ArchiveLink links with an enhanced selection criteria across all link tables

14. **Working with the API Gateway and SCRUD Test Harness:** All docuflow processing is logged and reportable, including the testing of basic SCRUD API's with no coding.
15. **Fiori-based docuflow DocManager Plugin** - This is an optional component for customers using Fiori user experience for end-users and wish to extend the docuflow **DocManager** functionality to SAP Fiori Apps allowing users to attach and view documents via Fiori Launchpad on browser.

All configuration and utilities can be accessed via transaction `/dflow/docuflowsys`

4.1.4. Viewing and Attachment List Options

Viewing content is accomplished via two access points :

1. From your SAP transaction / record
2. Direct to your content repository (optional)

From SAP

Viewing from SAP is achieved via the SAP or docuflow DocManager Attachment List (Object for Services) or Originals List (DMS).

Accessing content from these attachment lists perform the following authorization checks:

1. Access to your SAP transaction / record with standard SAP authorizations
2. Access to the ArchiveLink or DMS attachment with standard SAP authorizations
3. Optional: Additional SAP BADI authorization checks
4. Optional: Additional SAP docuflow BADI authorization checks

From there, docuflow provides 2 options to retrieve and display the content from your CMS system for URL based viewing.

1. Unauthenticated Access

The method leveraged to achieve this differs based on your CMS system:

1. Box:
 1. Unauthenticated short lived one time use expiring embedded URLs via web browser
 2. docuflow Content Server via a non-dialogue system user (thin and thick client options)
2. FileNet:
 1. ICC4AP Content Server via a non-dialogue system user (thin and thick client options)
3. M365.
 1. Unauthenticated short lived shared URLs via web browser
 2. docuflow Content Server via a non-dialogue system user (thin and thick client options)

2. Authenticated Access via CMS System

Whilst still leveraging SAP standard ArchiveLink and DMS links, docuflow can compile a viewing URL directing the user to log into the CMS system view their web browser.

This option leverages any security capabilities and options based on your CMS system. These are in addition to the check outlined in "From SAP".

Refer to the section "[CMS Security based on SAP Security / Data](#)" below for possible additional security checks based on SAP values.

Please refer to your CMS Systems' security documentation for further options.

SAP SFO/GOS Attachment List Options

You can leverage the standard SAP Attachment List or (highly recommended) the docuflow DocManager Attachment List.

Refer to the section "Attachment List Options" under the DocManager section for advantages in using the docuflow DocManager Attachment List plus additional SFO/GOS configuration options with docuflow.

Direct to your CMS System

Allowing access to content directly via your CMS system relies solely on that system's security model and capabilities.

Refer to the section "[CMS Security based on SAP Security / Data](#)" below for possible additional security checks based on SAP values.

Please refer to your CMS Systems' security documentation for further options.

FileNet Security Overview is provided below:

https://www.ibm.com/support/knowledgecenter/SSNW2F_5.5.0/com.ibm.p8.security.doc/p8pso000.htm

CMS Security based on SAP Security / Data

docuflow does not replicate the SAP security model to your CMS system. If however your CMS system can leverage SAP data in the form of metadata or CVL's (choice lists), you can leverage docuflow SAP to inform your CMS system with those values (ie: metadata transfer).

4.1.5. DocManager

DocManager is the central component, actioned by "rules", that enables and controls most docuflow processing in both directions inbound and outbound from SAP. This includes enabling further docuflow add-on's that might leverage DocManager such as Custom API Usage, DocMigrator, WfConnect and Publisher for example. It also allows you to specify what metadata is transferred in both directions, as well as viewing capabilities, options, and link management.

We also provide a Fiori User Experience-based docuflow DocManager that works seamlessly standard Fiori Apps and delivered as a Fiori Launchpad plugin. Check page "**Fiori DocManager - Launchpad Plugin**" for more details.

Inbound to SAP DocManager processes

1. SAPLink Create Link
2. SAPLink Workflow Connect (WfConnect)
3. SAPLink Accounts Payable
4. SAPLink DMS
5. SAPLink Custom Processing (class-method ABAP)

Note that all *inbound* to SAP DocManager processes require a rule in the docuflow Enterprise Connector to determine what content or events in your content management system are relevant to docuflow processing. [Refer to the SAPLink section of this document for more information.](#)

Outbound from SAP DocManager processes:

1. Store New and Assign
2. Publisher
3. Custom API usage
4. DocMigrator

Outbound from SAP DocManager processes requires only base installation and configuration. Please keep in mind however as you add outbound DocManager processes to consider load balancing the volume for performance reasons. [Refer to the docuflow Installation and Base Configuration Guide for more information for information on these considerations.](#)

[Note that you can also use docuflow to store business documents to non-docuflow Content Servers.](#) Simply configure the repository ID and relevant DocManager rule within docuflow for a consistent user interface for all your storage, linking, and viewing.

DocManager SFO/GOS Interface and Viewing

Below are some of the advantage to using the docuflow DocManager SFO/GOS interface:

1. Enhanced link attributes
2. Link Attributes for older systems with no implementation of table TOAAT
3. [Delete Link Management via docuflow Link Manager](#)
4. Multiple file / drag & drop that also support SAP GUI for JAVA
5. [Virtual Link Manager](#) to view links from connected SAP records without the duplication of link entries
6. Activity Logging and Viewing
7. [Allows storage/linking/viewing to both docuflow and non-docuflow connections](#)
8. Advanced viewing options to show and hide SAP DB notes, SAP DB attachments, as well as all ArchiveLink attachments

1. The "hide" option is very useful for in-flight migrations for example
9. Advanced BADl options for a wide variety of customizing options
10. Allow "Notes" to be created in your external repository instead of your SAP database
11. Supports Viewing attachments by auto-group via classification, docuflow flow, or file extension

Configuration

DocManager leverages standard SAP ArchiveLink and therefore entries for specified Document Types must exist already in OAC2, and have their link entry with SAP Business Object configured in OAC3.

Additionally, with docuflow ECM it is highly recommended that you clear/disable the value in OAG1 "Always Copy Document Class from Document Type". This allows you to use the "*" in the OAC2 document class column whilst at the same time letting docuflow add the appropriate file extension (ie: document class) in the Reserve field of the SAP ArchiveLink table (eg: PDF, XLSX, etc).

Configuring DocManager requires the below components:

1. Creating a DocManager Rule
 1. Required for both inbound and outbound
2. Enabling Store New and Assign / Attachment List Menu Path
 1. Required if you choose to use the docuflow DocManager SFO/GOS dialogue window for either document linking, viewing, storage to non-docuflow repositories, or usage of docuflow Link Manager
3. SAPLink / Inbound to SAP Processing
 1. Required for inbound processing only
 2. Configured at the docuflow middleware level
4. Delete Link Management via docuflow Link Manager
5. Attachment List Options
6. SAP Logon Security Configuration

Note on ArchiveLink Doctypes

When creating document types in OAC2, consider the following:

1. Doctypes drive SAP create and viewing authorization
2. For docuflow ECM or Content Server with Callback/Callback File move enabled: Doctypes drive location and metadata of document in SharePoint (and optional folder creation)
 1. Reminder: Native Content Server (without docuflow Callback File Move) uses repository ID for location.
3. If storing through content server AND viewing through content server (vs. any source of storage and viewing with ECM), you must create a document type for every file type/extension/mime-type ie: pdf, xls, xlsx, doc, docx, etc - which can result in a LOT of doctypes.
 1. This can be avoided by using an * in OAC2 and ensuring URL viewing is used.
 1. No further config required when using ECM to view via URL, however for content server basic Content Server Callback must be configured/enabled.

4.1.5.1. Creating a DocManager Rule

To create a DocManager Rule, enter transaction /dflow/docuflowsys and choose the menu path Configure→API Rules→DocManager.

Parameter	Description	Sample Value
Object Type	The SAP Business Object that is relevant to this DocManager rule. This and Document Type make up the unique rule key.	BKPF
Document Type	The ArchiveLink Document Type configured in OAC2 that is relevant to this DocManager rule. This and SAP Business Object make up the unique rule key. For link creation, and entry in OAC3 must exist for these two and the configured Cont.Rep.ID below (and in the docuflow Base product and OAC0).	ZWFINVSP01
En. DM Store New SFO	Enable Services for Object (SFO or GOS) for SAP to CMS business document storage.	True or False
En. DM Link Ex SFO	Enable Link Existing document via drag and drop from CMS to SAP via OLE (future use)	True or False
En. SL CS CB	Enable SAPink Content Server Callback file mover. If enabled, for Content Server "create": SAPLink will move from a Content Server dropzone (via a re-store) to a final location. This allows for a full feature set even though a document is stored via content server, such as: folder creation, dynamic location, external BPM and task creation, etc....in short: everything that a document stored via ECM can do. (metadata and viewing is handled above with En. SL CS CB).	
En. SL CS Move	Enable SAPink Content Server Callback file mover. If enabled, for Content Server "create": SAPLink will move from a Content Server dropzone (via a re-store) to a final location. This allows for a full feature set even though a document is stored via content server, such as: folder creation, dynamic location, external BPM and task creation, etc....in short: everything that a document stored via ECM can do. (metadata and viewing is handled above with En. SL CS CB).	
En. DM SL Cr. Link	Enable SAPLink Create Link which automatically creates ArchiveLink links for CMS inbound to SAP processing requests. This is used with CMS M.data Key or S.Link Cust Class/Method to determine the SAP key to link to (ie: must match exactly ArchiveLink object_id).	True or False
En. SL WFCo nnect	Enable Workflow Connect to trigger SAP workflow code free (and fill workflow containers with metadata from CMS) for CMS inbound to SAP processing requests.	True or False
En. SL Custo m	Enable custom ABAP code for CMS inbound to SAP processing requests (code is contained in .Link Cust Class/Method).	True or False
En. DM SL v1B	Enable passing of CSM inbound to SAP processing requests to docuflow V1.	True or False
Allow SL NoCon t.	For repositories that support contentless records, enable this to allow these to send inbound to docuflow SAP SAPLink requests	True or False

Parameter	Description	Sample Value
En.DM Publish	Enable SAP to CMS outbound saving of message output data with metadata.	True or False
En.DM Custom	Enable custom API usage for SAP to CMS outbound processing (code is container in your own Z namespace ABAP).	True or False
En.Migration	Enable DocMigrator for this rule.	True or False
Hide Att.List	Hide this SAP Business Object/DocType combination on the docuflow SFO/GOS attachment list menu for existing links.	True or False
Hide Act.List	Hide the Activity List tab on the SFO/GOS attachment list menu to prevent showing activity for this SAP Business Object/DocType.	True or False
CMS Profile	Determines which configured CMS Profile in docuflow SAP to utilize. Combination of CMS Profile and CMS Tag determine the API destination and thus load across an RFC destination.	VersaFile_Demo
CMS Use Tag	Determine which configured CMS Profile in docuflow SAP to utilize. Combination of CMS Profile and CMS Tag determine the API destination and thus load across an RFC destination.	VersaFile_Demo
Cont.Rep.ID	Determines which configured repository in docuflow SAP to utilize for SAP to CMS processing.	ZF
CMS Type	Determines which CMS Types to use for docuflow SAP to CMS outbound processing (typically Create). FileNet: - document class - for create, defaults to the "Document" document class if unspecified. Documentum: - object type - for create, defaults to the "dm_document" class if unspecified. Box: - metadata template - for create, defaults to creating a document with no metadata template if not specified. SharePoint On-Prem/SharePoint O365: - content type - for create, defaults to creating a document of content type Document if not specified.	accountsPayable
CMS Location	Specifies where the content resides for docuflow SAP to CMS outbound processing (typically Create). ** CMS Location allows Environment Variable usage. In short these allow for different values based on your SAP system id (ex: DEV / QA1 / PRD). Please refer to the docuflow Installation and Base Configuration Guide under	5523993

Parameter	Description	Sample Value
	"Environment Variables" for detail on how to configure and utilize. ** FileNet: - if unspecified documents are left unfiled. Documentum: - If unspecified documents are stored in root of the user's cabinet. Box: - Best practice: folder number/id for the exact folder location can be specified - If unspecified documents are stored in the configured root folder of the Enterprise Connector - relative folder names can be used with a / being the root. If a root folder is configured for the Box CMS in the Enterprise connector all relative folder locations will be relative to the root folder configured. - Relative folder names are not recommended for performance reasons. SharePoint On-Prem/SharePoint O365: - optional sub-site URL (added on to repository URL) with mandatory document library specified last	
Own Folder	Determines whether a file is stored in the CMS Location (not enabled), versus when enabled the Enterprise Connector will check if there is a folder with the name "SAP_<sapkey>". If there is one (as a subfolder to CMS Location), the file is stored there. Otherwise it will create one automatically and store the file there.	True or False
Link Folder	If Own Folder is enabled, whether to create an ArchiveLink link to the folder vs. the file. Linking to a folder, when viewed from SAP, will show the folder view and all of its contents. Linking to a file will bring up the Box preview of the file being viewed.	True or False
CMS M.data Key	For inbound from CMS to SAP processing requests for "En. DM SL Cr. Link" (ie: automatic link creation), if an SAP key does not need to be formatted or retrieve and is an exact match in your CMS system to the ArchiveLink key required (with formatting/padding, etc..), you can specify that property name here. Otherwise, you must use a S.Link Cust Class/Method to determine the SAP key to link to	InvoiceSAPKey
S.Link Cust Class	For inbound from CMS to SAP processing requests for "En. DM SL Cr. Link" (ie: automatic link creation), if an SAP key needs to be formatted or retrieve or is NOT an exact match in your CMS system, you can compile the SAPKey here (either with the metadata or SAP lookup). Otherwise, you must specify the key in the CMS M.data Key field to determine the SAP key to link to. Refer to the docuflow Developers Guide for more information on how to program these class/methods.	/DFLOW/CL_DOCMANAGER_HELPER
S.Link Cust Method	See above "S.Link Cust Class"	SAPLINK_EXAMPLE

Parameter	Description	Sample Value
En. MD Trans. Map	Applicable to both inbound and outbound processing patterns, this <u>no-code option</u> generates metadata mapping from SAP to your CMS system If enabled, a metadata map must exist under the Configure→Metadata mapping on the docuflow SAP System Dashboard.	True or False
En. MD Trans. Cus.	Applicable to both inbound and outbound processing patterns, this <u>CODE_option</u> generates metadata mapping from SAP to your CMS system If enabled, the below metadata class/method must be specified.	True or False
Metadata Class	Refer to the docuflow Developers Guide for more information on how to program these class/methods.	/DFLOW/CL_ DOCMANAG ER_HELPER
Metadata Method	See above "Metadata Class".	METADATA_ EXAMPLE
Description	English description for cosmetic reasons on the overall System	"FI Invoice Processing"

Example

Object Type **BKPF**
 Document type **ZWFINVSP01**

DocuFlow DocManager: Configuration

☒ En. DM Store New SFO	
☐ En. DM Link Ex SFO	
☒ En. DM SL Cr.Link	
☐ En. SL WFCConnect	
☐ En. DM SL Custom	
☐ En. DM SL v1B	
☐ Allow SL NoCont.	
☐ En. DM Publish	
☐ En. DM Custom	
☐ En. Migration	
☐ Hide Att.List	
☐ Hide Act.List	
CMS Profile	VersaFile_Demo
CMS Use Tag	VersaFile_Demo
Cont.Rep.ID	ZF
CMS Type	accountsPayable
CMS Location	<AP_location>
☐ Own Folder	
☐ Link Folder	
CMS M.data Key	
S.Link Cust Class	/DFLOW/CL_DOCMANAGER_HELPER
S.Link Cust Method	SAPLINK_EXAMPLE
☐ En. MD Trans.Map	
☒ En. MD Trans.Cus.	
Metadata Class	/DFLOW/CL_DOCMANAGER_HELPER
Metadata Method	METADATA_EXAMPLE
Description	

This example enables the SAP Object Type for parked FI invoices (BKPF) and the ArchiveLink doctype 'ZWFINVSP01' to be configured for outbound business document storage from SAP to your CMS (En. DM Store New SFO) with the metadata compiled in the class/method /DFLOW/CL_DOCMANAGER_HELPER→METADATA_EXAMPLE.

Inbound from your CMS to SAP processing is also enabled (En. DM SL Cr.Link) to automatically link content to SAP, with the SAPLink /DFLOW/CL_DOCMANAGER_HELPER→

SAPLINK_EXAMPLE being the class/method that helps compile and format the key for an FI Invoice (company code, document number, fiscal year – all concatenated with document number zero padded).

Also notice the CMS Location value of <AP_Location> leveraging the Environment Variables configuration.

4.1.5.2. Enabling Store New and Assign / Attachment List Menu Path

For all Store New and Assign (ie: Business Document Storage), or to use the docuflow Attachment List for any other purposes, you must add the DocManager menu path to the Services for Object / Global Object Services configuration table via the instructions below:

1. Maintain the table SGOSATTR via SM30 in your development configuration client. Note that this configuration is client independent, but will generate a transport to must be moved with other ArchiveLink and docuflow configuration.
2. Find the entry with the name INFO_SERVICE and double click to change the entry.
3. Change the field value for "Next Service" to DOCUFLOWSTOR as shown below. Save your entry and green arrow back.

Table View Edit Goto Selection Utilities System Help

Change View "SGOS: Attribute of Generic Services": Details

New Entries

Name of service: INFO_SERVICE

SGOS: Attribute of Generic Services

Description	Help for object services
Quick info	Help for the object services
Class f.Gen.Service	CL_GOS_SRV_INFO
Service Type	Single Service
Icon	ICON_INFORMATION
Next service	DOCUFLOWSTOR
Subservice	
☐ Control	
☐ Commit required	

4. Click the button "New Entries" and create the following entry named "DOCUFLOWSTOR":

- ensure the class name /DFLOW/CL_DOCMANAGER is spelled correctly and that the "Commit Required" is enabled!

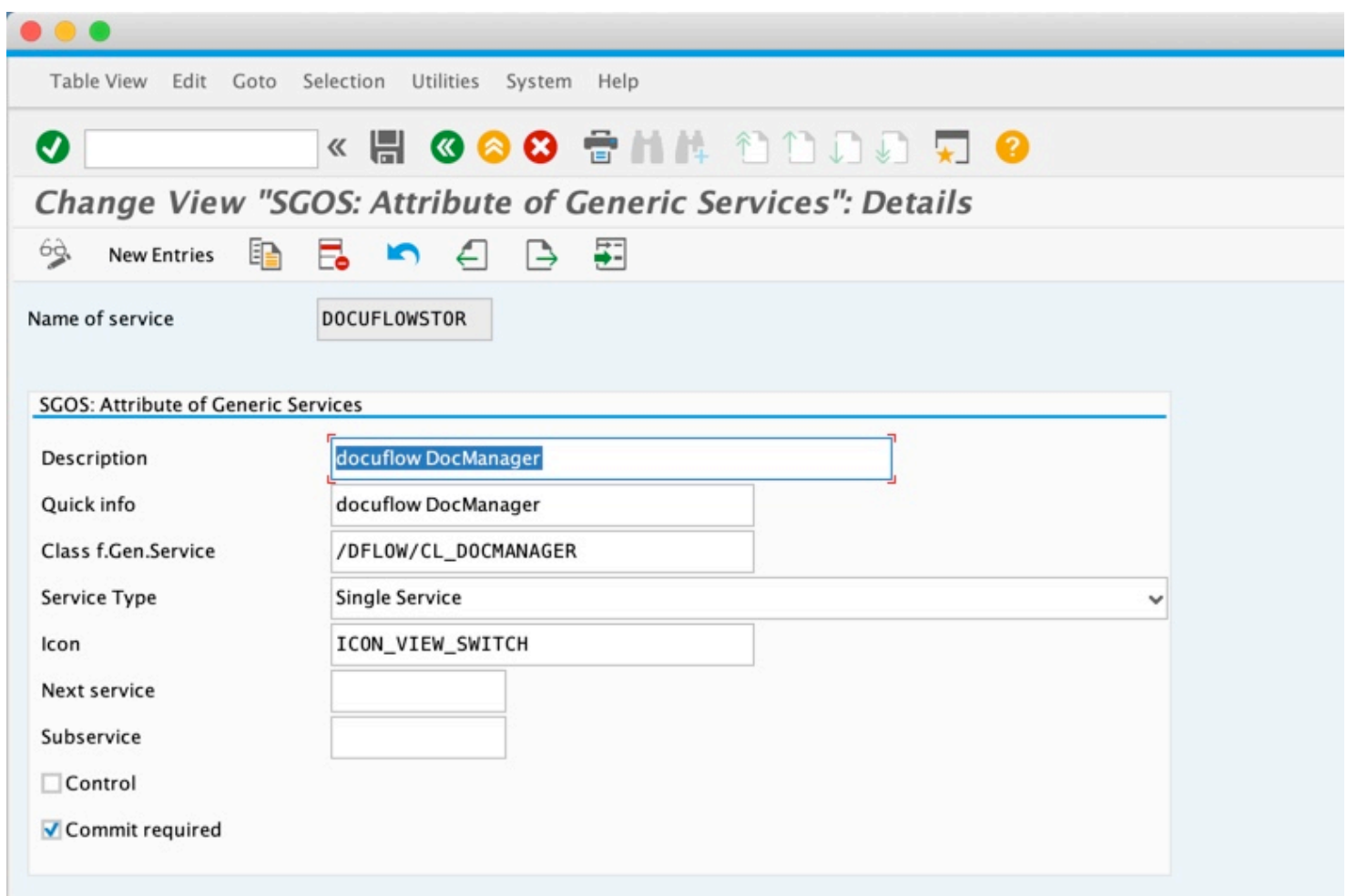


Table View Edit Goto Selection Utilities System Help

Change View "SGOS: Attribute of Generic Services": Details

New Entries

Name of service DOCUFLOWSTOR

SGOS: Attribute of Generic Services

Description docuflow DocManager

Quick info docuflow DocManager

Class f.Gen.Service /DFLOW/CL_DOCMANAGER

Service Type Single Service

Icon ICON_VIEW_SWITCH

Next service

Subservice

☐ Control

☒ Commit required

5. Save your entry and include this transport in your migration to QUAL and PROD list of transports.

4.1.5.3. Required Metadata Properties

Note: you might already have some properties during your Base Installation and Configuration.

*** It is mandatory that all properties (both ECM and Content Server) be added for any content type/metadata template/document class used for docuflow integration ***

ECM Required Metadata Properties

These names are suggestions and recommended, but are configurable in the Enterprise Connector.

SharePoint Columns:

Attribute Name (suggested)	Attribute Type	Attribute Settings
docuflowUsername	single line of text	
docuflowTimestamp	single line of text	
docuflowSaplinkTrig ger	single line of text	The only values docuflow will recognize are "True" and "False" as triggers. The resulting values are True and False by default, but can be changed in the config.properties file under your installation directory. saplinkfalse.value=<value> (for success) saplinkerror.value=<value> (for error) - eg: "DoNotProcessSuccess" and "DoNotProcessError" - saplinkfalse.value=DoNotProcessSuccess - saplinkerror.value=DoNotProcessError
docuflowSaplinkErro r	single line of text	

Box Properties:

Attribute Name (suggested)	Attribute Type	Attribute Settings
docuflowUsername	Text	
docuflowTimestamp	Date	
docuflowSaplinkTrigger	Dropdown Single Value	The only values docuflow will recognize are "True" and "False" as triggers. The resulting values are True and False by default, but can be changed in the config.properties file under your installation directory. saplinkfalse.value=<value> (for success) saplinkerror.value=<value> (for error) - These values must be added to the available values in the Box Dropdown Single Value list - eg: "DoNotProcessSuccess" and "DoNotProcessError" - saplinkfalse.value=DoNotProcessSuccess - saplinkerror.value=DoNotProcessError
docuflowSaplinkError	Text	

FileNet P8 Properties:

Property Name (suggested symbolic name)	Property Type	Property Attributes
docuflowUsername	String	Single, Length of 64
docuflowTimestamp	Date/Time	Single
docuflowTrigger	Date/Time	Single
docuflowSaplinkTrigger	Date/Time (preferred) or Boolean	Single
docuflowSAPLinkError	String (preferred), Boolean, or Date/Time	Single, Length of 256 (if String)

Documentum Properties:

Attribute Name (suggested)	Attribute Type	Attribute Settings
docuflowUsername	String	Single, Length of 64
docuflowTimestamp	Date/Time	Single
docuflowSaplinkTrigger	Date/Time (preferred) or Boolean	Single
docuflowSaplinkError	String(preferred), Boolean, or Date/Time	Single, Length of 256 (if String)

Content Server Required Metadata Properties

*** Please ensure these are EXACT casing as below *****

Attribute Name (case sensitive)	Attribute Type	Attribute Setting
contRep	Text	
docID	Text	
dateC	Text	
compId	Text	
pVersion	Text	
docProt	Text	
contentLength	Text	
accessMode	Text	
charset	Text	
compStatus	Text	
sapVersion	Text	

4.1.5.4. SAPLink / Inbound to SAP Processing

*Unsupported in S3

SAPLink is the inbound processing from your content repository **to** SAP. This can be leveraged to execute a number of docuflow delivered functions in SAP or your own custom logic. Thus, why we call SAPLink the "do anything pipe" into SAP.

There are three components involved in SAPLink processing:

1. Ensure the CMS document class/object type/metadata template has the required fields for SAPLink processing
2. Configuration a "rule" in the Enterprise Connector middleware: This determines what content is to be processed, when, and what (if any) metadata is to be transferred. What is processing is dependent on the CMS system being used (outline more below).
3. Configuration of a corresponding DocManager rule in SAP that matches the document type configured in Step 2: This determines what action is to be performed in SAP (link a document, trigger custom code, start Accounts Payable workflow, etc.)

Note: the docuflow system account specified in the middleware must have access to the transaction /DFLOW/DOCUFLOWGTW in its security profile to execute inbound to SAP Jco calls. Please refer to the section "Create SAP System Account" in the Base Installation and Configuration guide.

The inbound processing rule determines:

1. What is to be processed
2. What metadata is to be sent to SAP

How to configure a SAPLink rule in Enterprise Connector

1. Load up in the browser the onfigurator <https://localhost:82/>
2. Under "Mappings" click inbound to SAP.
3. Hover over each of the input boxes in the UI to look at sample values for each input box.
4. Click Save.
5. Restart the docuflow service.

Notes

The DocManager rule in SAP determines what action is to be performed in SAP for each item "found" from a SAPLink rule in the connector.

Each request is processing and calls a docuflow BAPI within the SAP, which then reads the DocManager rule configuration found for that doctype.

There can only be one ArchiveLink document type and SAP business object combination ever configured in the DocManager configuration, thus avoid sharing document types across SAP object (to avoid an inbound collision).

docuflow will then execute the configured action and pass back a return, either an error or success.

In the case of a success, the trigger property is set too False to prevent a re-processing of that item found.

In the case of an error, the error property, if relevant to your CMS system, will be updated.

In both scenarios, the docuflow username and time stamp fields are updated.

Upon a success SAPLink, metadata can also be enabled to be transferred back to your CMS system.

Rule Config: What is to be processed

Inbound to SAP entries search for documents that meet specific criteria and send the information to SAP for processing. If processing completes successfully, the trigger property is adjusted so that the document is not processed again. If the processing were to fail by default, the processing is tried again. In order to not retry failed documents, adjustments can be made to the filter string to omit documents that were in error.

Setting	Description	Sample Value(s)
Enable Inbound to SAP	If checked, configuration for Inbound to SAP is enabled and can be configured for inbound to SAP processing.	
SAP	The SAP system to send document metadata too.	SAP DEV ED1
CMS	The CMS system to get document metadata from.	
Trigger data Type	Data type by which the saplink rule is triggered. Task and Sign task type only applicable for Box workflow tasks.	Boolean, TimeStamp, task and signtask.
SAP Repo.	The SAP ArchiveLink repository that should be associated with this metadata being passed.	ZC
SAP Type	The SAP ArchiveLink document type associated with this metadata being passed.	ZCONTRACT
CMS Repo.	The content management system object store or repository. For Box implementations, use "Default"	Default
CMS Type	The content management system document class, object type or metadata template to look for the requested metadata in.	scmContracts
Interval	The Inbound to SAP process searches for documents matching the configured criteria and then sends the information to SAP. This interval determines how often the underlying content management system is searched. For Box implementations, setting this value less than 5 minutes has no value.	10
Trigger Prop.	The content management system property or attribute associated with when to trigger sending information to SAP. By default, docuflow sets this property to True or False based on the result of a SAPLink call to SAP. These values can be manipulated via the Config.properties file with the below entries: saplinkfalse.value=<value> (for success) saplinkerror.value=<value> (for error)	docuflowSaplinkTrigger saplinkfalse.value=DoNotProcessSuccess saplinkerror.value=DoNotProcessError
Type	The trigger property or attribute type. Boolean triggers happen when the Boolean property or attribute is set to true. Timestamp triggers happen when the configured date/time property or attribute has no value.	Boolean

Setting	Description	Sample Value(s)
	Currently, Box and M365 implementations only support Boolean trigger types (even though in Box the type is Single Value Dropdown Text and SharePoint is single line of text)	
Error Prop.	If an error happens while trying to process a particular file you can optionally store the error information in the content management system.	docuflowSaplinkError
Type	The error property or attribute type. Boolean error types are only set to true if an error happens. Timestamp error types only set a date/time stamp when an error happens. String errors put as much of the error as they can into the configured error property or attribute. This is the preferred configuration. Currently M365 implementations only support String (single line of text) and Box supports Text.	String
Max	An optional setting to limit the number of entries returned when searching for documents. A setting of 0 has no limit.	0
Filter	An optional (with FileNet and Documentum and implementations but required for Box) filter string to finely tune what is searched for in the underlying content management system. Refer to the where clause documentation for working with the Search Utility in the appendix for help configuring a filter clause. For Documentum and FileNet filter clauses you must start with AND or OR since the documents are already being searched for by the configured trigger type so this additional filter logic needs to be added on with AND or OR notations. Some examples are: AND <insert error property name> IS NULL (which omits documents that have already errored when using a String based Error Property) OR LastModifier <> '<insert DocuFlow FileNet Username>' (which handles updates to documents as well as the first time they are created) OR LastModifier <> '<insert DocuFlow FileNet Username>' AND <insert error property name> IS NULL (which omits error documents and also handles updates to documents)	(Box example): Search API <pre>mdfilters=[{"templateKey":"scmContracts", "scope":"Hi enterprise","filters": {"docuflowSaplinkTrigger":"True"}}] &limit=50&offset=0</pre> (Box example): Metadata query: <pre>{"from":"enterprise_761921826.metadatatemplate","q uery":"docuflowSaplinkEnabled = :value","query_params": {"value":"True"},"ancestor_folder_id":"131865107948" }</pre> (FileNet example not including documents in error) <pre>AND DocuFlow_SAPLinkError IS NULL</pre> (Sharepoint example): <pre>fields/SAPFacility eq 'Complete' and fields/docuflowSaplinkTrigger eq 'True'</pre>

Setting	Description	Sample Value(s)
	OR r_modifier <> '<insert DocuFlow Documentum Username>' (which handles updates to documents as well as the first time they are created) OR r_modifier <> '<insert DocuFlow Documentum Username>' AND <insert error property name> IS NULL (which omits error documents and also handles updates to documents) For Box implementations you must enter in a filter clause which is the input to the metadata search web service for Box.	
Location	An optional location to further limit where documents are searched for. This is only possible with FileNet and Documentum implementations and not possible with Box and SharePoint implementations.	
Order By	An optional order by clause to order the returned results. Make sure to include the metadata being ordered by in the metadata comma separated list. This is only possible with FileNet and Documentum implementations and not possible with Box and SharePoint implementations.	
Source Trigger Property	An optional property or attribute associated to trigger when sign task is completed. (BOX SIGN)	middlewareOriginalDocId
LineItemColumn	An optional property which is referred to csv file containing line item information. (Reference to a csv file)	
Sub.	If enabled sub folders will be searched. For more information refer to the Search Utility in the appendix for help. This is only possible with FileNet and Documentum implementations and not possible with Box and SharePoint implementations.	

Rule Config: What metadata is to be sent to SAP

What metadata is sent from your repository to SAP is determined by the Metadata column:

Setting	Description	Sample Value(s)
Metadata	The comma separated list of metadata to send to SAP in the specified order.	title,contract,name

Rule Config: Sign Task

sign task Config

Setting	Description	Sample Value(s)
sourceTriggerProperty	The content management system property or attribute associated to trigger when the sign task is completed.	middlewareOriginaldocId

4.1.5.4.1. Using Microsoft Graph Explorer to Debug SharePoint Filters in Link Enterprise

1. Purpose

Demonstrated how to use Microsoft Graph Explorer to debug SharePoint filters in docuflow middleware. This helps identify whether issues originate from Microsoft Graph or the middleware layer.

2. Tools & References

- **Graph Explorer:** <https://developer.microsoft.com/en-us/graph/graph-explorer>
- **Demo Tenant:** yourtenant.sharepoint.com
- **Sample Site ID:** [https://graph.microsoft.com/v1.0/sites/\(yourtenant\).sharepoint.com:/sites/VFSGMiddleware](https://graph.microsoft.com/v1.0/sites/(yourtenant).sharepoint.com:/sites/VFSGMiddleware)
- **Sample Library:** [/VFSGMiddleware/Equip/](#)

3. Benefits of Using Graph Explorer

- Test filters instantly without restarting middleware
- Isolate root causes between Graph and docuflow
- Discover internal SharePoint field names
- Simulate customer scenarios for debugging

4. Corrected Graph API Filter Example

Correct Syntax:

```
https://graph.microsoft.com/v1.0/sites/.../list/items?$expand=fields&$filter=fields/SAPFacility eq 'Complete' and fields/docuflowSaplinkError0 ne 'True'
```

URL-Encoded Version (for code):

```
...?$expand=fields&$filter=fields/SAPFacility%20eq%20'Complete'%20and%20fields/docuflowSaplinkError0%20ne%20'True'
```

5. Boolean Field Values

Value	Boolean
0	False
1	True

6. Common Pitfalls

Mistake	Result
Using = instead of eq	Filter ignored
Missing fields/ prefix	Query fails or returns incorrect results
Unindexed fields	Filter errors unless workaround header used
Null Boolean values	Ignored by Graph API

4.1.5.5. Attachment List Options

Using the docuflow Attachment List provides the below additional benefits:

1. docuflow Attachment list displays all ArchiveLink Attachments whether they were created via docuflow ECM or other content services
2. Allow document storage to both docuflow ECM supported CMS system AND non-docuflow supported CMS systems (ie: any ArchiveLink endpoint)
3. docuflow Virtual Link Manager allows you display links that are not linked to the SAP record you are viewing, but rather linked to an associated SAP record. Instead of implementing a process that duplicates links, you can use Virtual Link Manager to show these links as if they were attached. For example, when viewing a Purchase Order and its attachments, you can also display any attachments that were directly linked to the Goods Receipt.
4. docuflow Virtual Link Manager also allows you to manipulate what is shown on the attachment list as a whole: meaning you can hide links, blank out sensitive information, add additional security checks, and so forth.
5. You can choose to display or hide SAP DB Attachments and Notes
6. You can choose to hide particular SAP Object / doctype combinations. This is used often during migrations when for a window of time, you may have duplicate links from the source system as well as the new target system prior to validation and link deletion.
7. For those earlier release systems that do not have table TOAAT implemented, the docuflow Attachment List fills the gap in providing the columns of who created the attachment, time stamp, filename, and extension
Regardless of #1 above, the docuflow Attachment List provides the full ArchiveLink link record (repository UUID, doctype, etc)
8. Easy "tab" based interface for drag and drop document storage and viewing in one window
9. Consistent interface regardless of windows or java based SAP GUI
10. Allow "Notes" to be created in your external repository instead of your SAP database
11. Supports Viewing attachments by auto-group via classification, docuflow flow, or file extension

docuflow provides the above functionality config and code free (with the exception of the Virtual Link Manager - refer to the below).

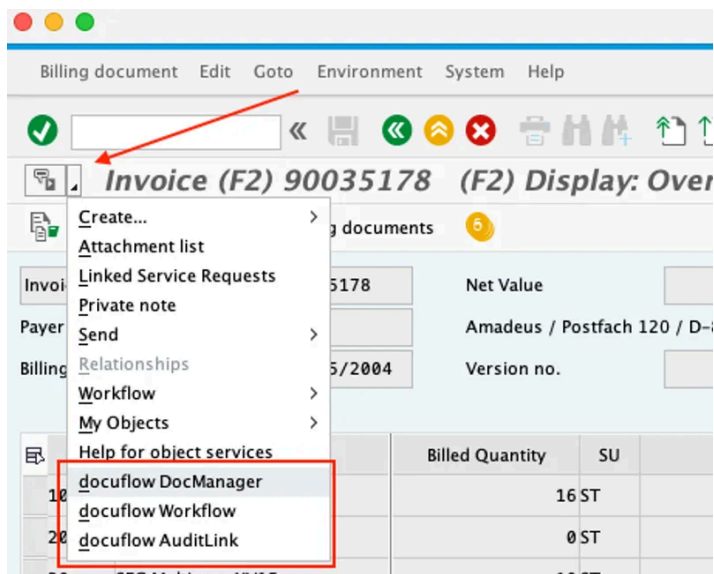
Configuration is located within the docuflow System Dashboard (/dflow/docuflowsys) and choose the menu path Configure→Services for Object

Below is an overview of the configuration required to enable the docuflow DocManager Attachment List:

SGOSATTR Entries / Utility

This utility is in beta and should not be used.

Instead, the following entries must be manually created in the table SGOSATTR. These entries enable the docuflow SFO/GOS menu path(s) to be shown (note: the docuflow DocManager, Workflow and AuditLink will only be displayed IF that option is *independently and separately* configured for a particular SAP Object), else they will not appear.



Below are the entries that must be created in SGOSATTR via transaction SM30 (note this is cross-client):

Name of service: DOCUFLOWSTOR

Name of service: **DOCUFLOWSTOR**

SGOS: Attribute of Generic Services

Description	docuflow DocManager
Quick info	docuflow DocManager
Class f.Gen.Service	/DFLOW/CL_DOCMANAGER
Service Type	Single Service
Icon	ICON_VIEW_SWITCH
Next service	DOCUFLOWWFM
Subservice	
☐ Control	
☒ Commit required	

Name of service: DOCUFLOWWFM

Name of service: **DOCUFLOWWFM**

SGOS: Attribute of Generic Services

Description	docuflow Workflow
Quick info	docuflow Workflow
Class f.Gen.Service	/DFLOW/CL_WORKFLOW_GOS
Service Type	Single Service
Icon	ICON_WORKFLOW_ACTIVITY
Next service	DOCUFLOWAUDC
Subservice	
☐ Control	
☒ Commit required	

Name of service: DOCUFLOWAUDC

Name of service	DOCUFLOWAUDC
SGOS: Attribute of Generic Services	
Description	docuflow AuditLink
Quick info	docuflow AuditLink
Class f.Gen.Service	/DFLOW/CL_AUDITLINK_GOS
Service Type	Single Service
Icon	ICON_WORKFLOW_ACTIVITY
Next service	
Subservice	
☐ Control ☒ Commit required	

TOAEX Utility

Except in the case where a (docuflow) Content Server is being used for viewing, an entry in the table TOAEX must be created to allow for docuflow to handle viewing request and integrate your web browser.

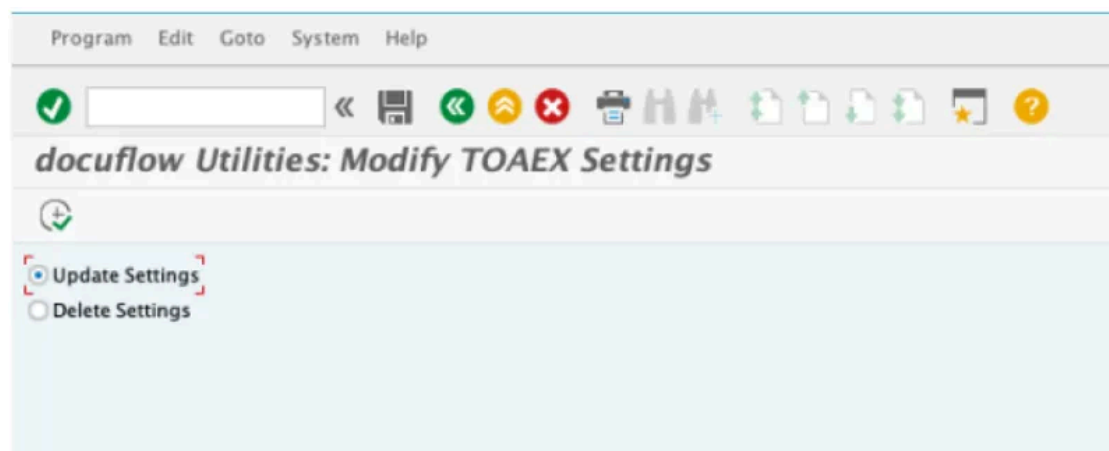
First check whether you have an existing entry in table TOAEX (use transaction SE16 or SE17) under the two exit ID's:

1. OA_DISPLAYJAVA_01
2. OA_OBJECTDISPLAY_01

If you DO, please consult your development team to work with your docuflow implementer to merge your existing logic within those function modules and the docuflow function module "/DFLOW/ZUTIL_TOAEX_FUNCTION_01".

If you DO NOT have entries in the TOAEX table for the above exit ID's, choose the sub menu path Configure→Services for Object→**TOAEX Utility**.

From there simply execute using the "Update Settings" option. If utilizing this method, ensure you execute this in each SAP environment that docuflow will be used.



Your TOAEX table should now look like the below for the two exit ID's:

EXIT ID	Status (ACTIVE)	Function Module (EXIT_FUB)	VALUE
OA_DISPLAYJAVA_01	X	/DFLOW/ZUTIL_TOAEX_FUNCTION_01	
OA_OBJECTDISPLAY_01	X	/DFLOW/ZUTIL_TOAEX_FUNCTION_01	

MENU OPTIONS

Below are the menu options for the docuflow DocManager Attachment List:

Object Type

This is your SAP Object such as BKPF, BUS2012, etc. This must match the business object for the SAP transaction you are configuration. Note you can also use a wildcard "*" to set options system wide. You can subsequently add individual SAP Objects that will override the system "*" wildcard.

SAP Key1/2/3 Length

Specifies the SAP Business Object's key length. Only required if "En. WF TrackLink" or "En. Late WF" is enabled below.

For example, vendor numbers are a single key field, and are 10 characters in length for LFA1. Put a 10 in the SAPKey1 Length.

For example, vendor numbers at the company code level are a composite key (ie: two fields) for LFB1. 10 characters in length. Put a 10 in the SAPKey1 Length, and 4 in SAPKey2 Length.

En. Atta.w/ Fold

If enabled, the 2nd generation of the docuflow DocManager attachment list will be used. This newer version contains the following changes:

1. Supports Viewing attachments by auto-group via classification, docuflow flow, or file extension
2. Move the attachment list tab to the first tab, and the "Store" tab to the 2nd position.

SAP DB Attach

If enabled, Attachments stored in the SAP database will be displayed in the attachment list.

SAP DB Notes

If enabled, Notes stored in the SAP database will be displayed in the attachment list.

Attachments Link Manager

If enabled, users will be allowed to delete items from the attachment list leveraging docuflow Delete Link Management functionality. Please refer to the section Delete Link Management via docuflow Link Manager for more information.

En. Download

If enabled, users will be allowed to download content in the attachment list to their local/network drive.

En. Cr.Note DB

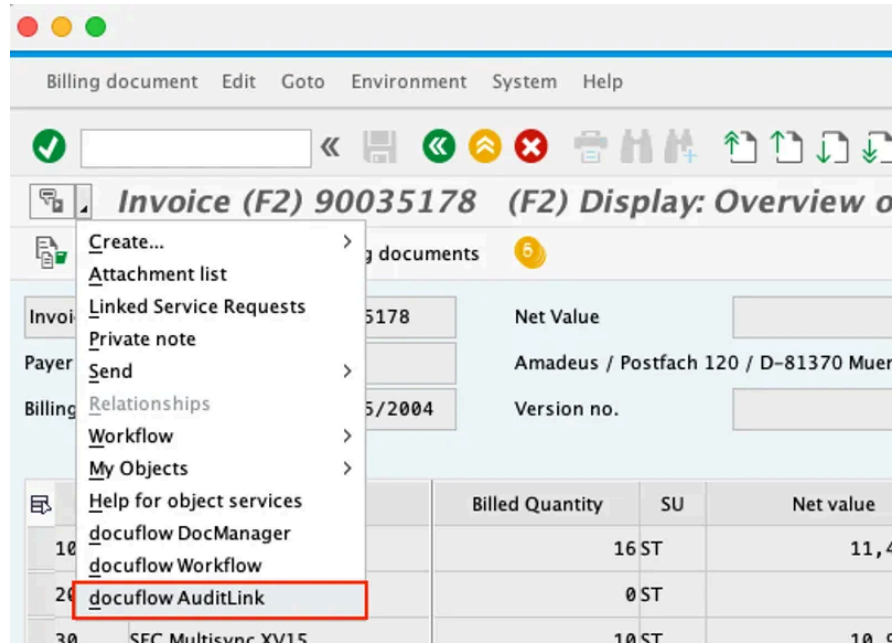
If enabled, users will be allowed to create/store Notes in the SAP database.

En. Cr.Note GOS

If enabled, users will be allowed to create/store Notes in your content repository (stored as .txt files).

En. AuditLink GOS

If enabled, the docuflow AuditLink TrackLink option will show on the SFO/GOS menu path:



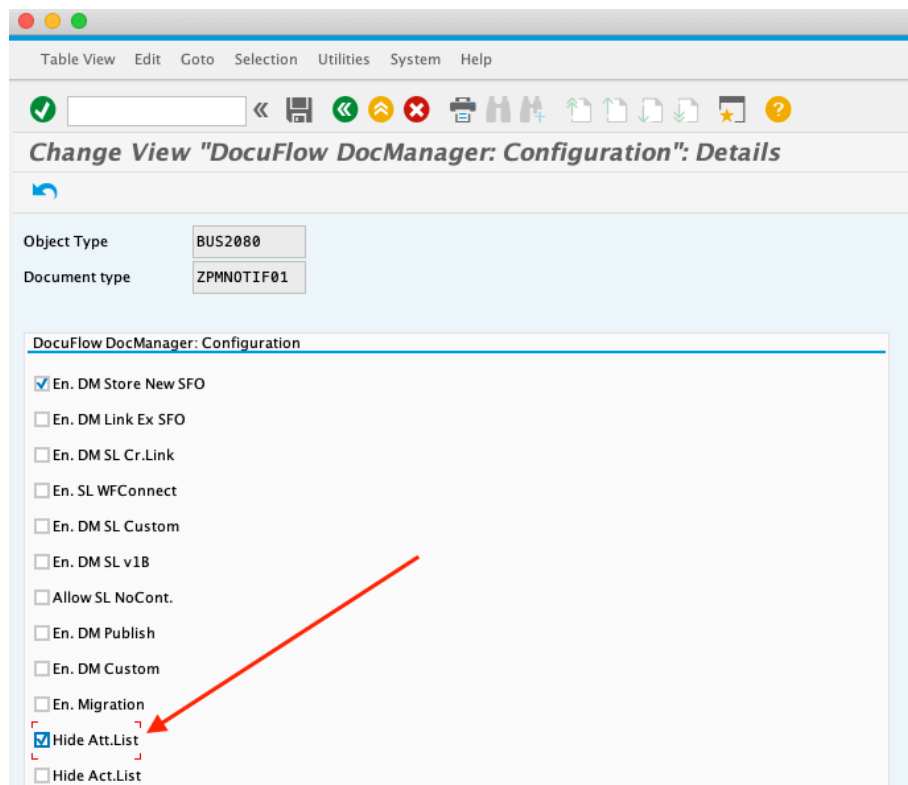
En. WF TrackLink

En. Late WF

Document Type

Hiding particular SAP Object / doctype combinations

1. via the docuflow System Dashboard (/dflow/docuflowsys) choose the menu path Configure→API Rules→Document Manager
2. Create/find the entry for the SAP Object/document type rule.
3. Enable the "Hide Att.List" checkbox:



Virtual Link Manager

Please refer to the docuflow Developers Guide on more information on implementing.

Delete Link Manager

Configuration and usage is outlined in the next section.

4.1.5.6. Delete Link Management via docuflow Link Manager

docuflow's Link Manager provides the mechanism for a user to create a "delete request" via the docManager Attachment List within their SAP transaction, and for that delete request to be triaged by 1 or more Link Managers.

Dependent on the configuration of Link Manager, the content will also be, or not allowed, to be deleted from the content repository along with the link.

Both end-users and Link Managers can add notes to the requests to give reason for the request, or for the approval or rejection of the request.

You may also choose for certain SAP Objects and doctypes to allow the end-user to perform these actions.

Link Manager functionality can be categorized into 3 components:

1. Configuration
2. End-User Requests
3. Link Manager Actions

Configuration

1. Via the docuflow System Dashboard (/dflow/docuflowsys) choose the menu path Configure→Attachment List→ Link Manager
2. For any SAP Object / doctype combination, specify what is or is not allowed to be taken as link and content action:
3. Wildcards (*) are supported and therefore you can create global rules in addition to more specific rules. Below is an example showing that system wide, links and content are allowed to be delete by the Link Manager only. In addition, for and HR object, only links are allowed to be delete but not the content:

ObjectType	Doc. type	Link	Content
*	*	Link Manager Allowed	Link Manager Allowed
BUS1065	ZHREMPGEN1	Link Manager Allowed	Not Allowed

End-User Requests

An end-user creates a delete request directly from within the docManager Services for Object Attachment List by highlighting the attachment they want to delete and choosing the the Delete icon:

LT	Storage date	Strge time	Filename	Creator	Description	
	07/07/2020	12:06:44	Inbound AMM-XA42-0040-001-8.00-Aircraft-Maintenance-Manual.pdf	DOCUFLOW	Demo - File Linking	Zf
		10:57:11	Folder View: T-3000-N02	DOCUFLOWDEMO	Demo - Folder Creation / Folder Linking	Zf
		10:54:41	SL_1265.pdf	DOCUFLOWDEMO	Demo - File Linking	Zf

The end user is then prompted whether they would like to add a note (optional) to explain the reason to the Link Manager:

Create note

Title of note: Delete Request Note: Initial

Whoops, this is a picture of my cat. Please delete :->

Once submitted, the delete request is sent to the Link Managers and the link status is updated "Pending" as indicated on the left hand column:

Store New and Assign

Attachment List

Activity

Del. Status	LT	Storage date	Strge time	Filename
PENDING		07/07/2020	12:06:44	Inbound AMM-XA42-0040-001-B.00-Aircraft-Maintenance-Manu
			10:57:11	Folder View: T-3000-N02
			10:54:41	SL_1265.pdf

Note that pending attachments cannot be viewed.

If the delete request is approved, the link entry will automatically disappear.

If the delete request is rejected, the link entry will remain and be taken out of Pending status.

An end-user may view all their historical and current delete requests via the "My Delete Request" button:

Store New and Assign

Attachment List

Activity

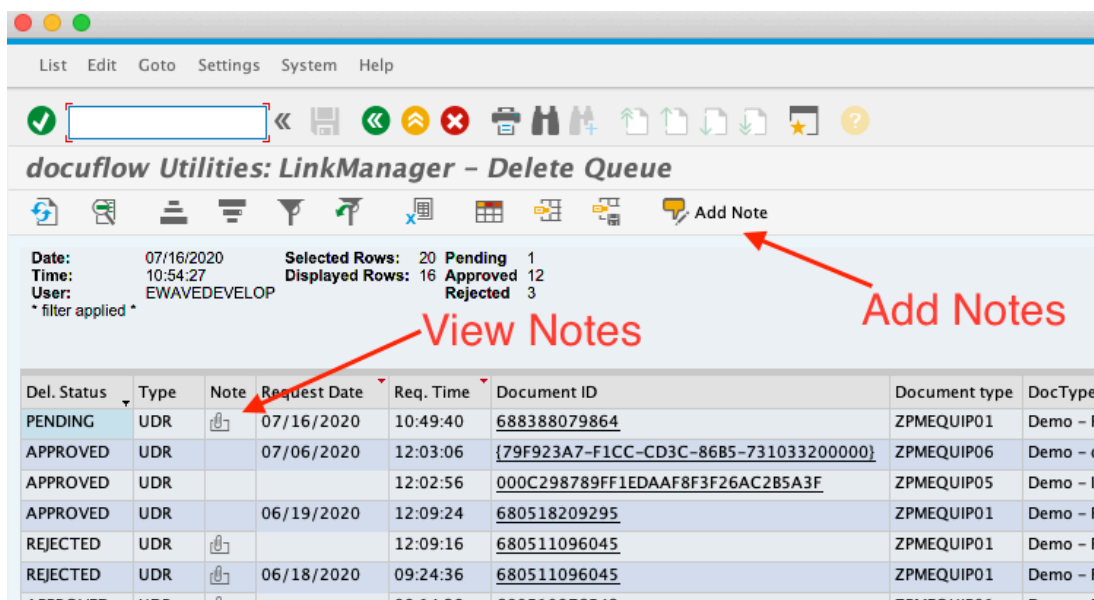
Fullscreen

My Delete Requests

Del. Status	LT	Storage date	Strge time	Filename
PENDING		07/07/2020	12:06:44	Inbound AMM-XA42-0040-001-B.00-Aircraft-Maintenance-Manu
			10:57:11	Folder View: T-3000-N02
			10:54:41	SL_1265.pdf

Note that the delete requests viewed are for that user only, however - is across the entire SAP system regardless of which SAP record or transaction they generated a request from.

This delete queue also allows the end user to view or add notes:



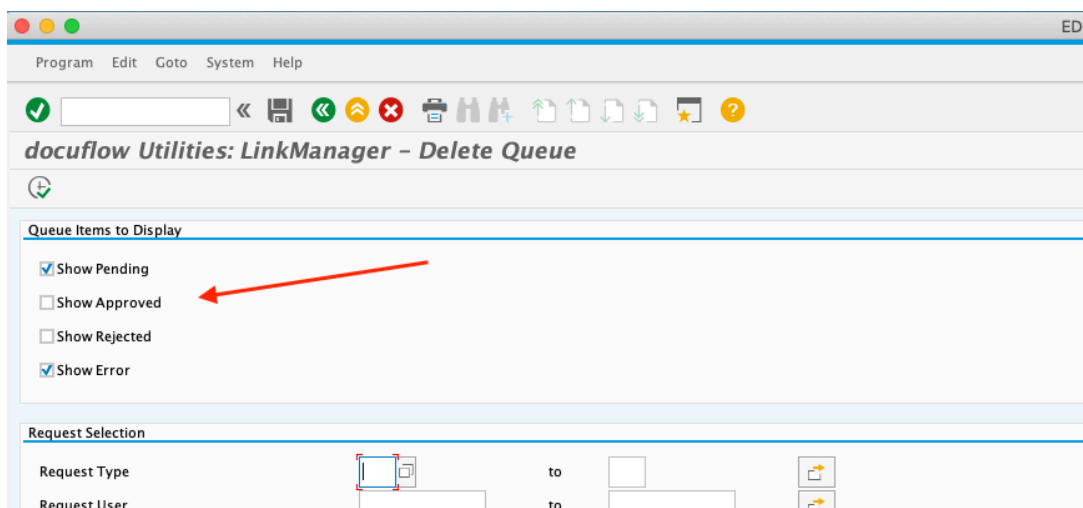
Link Manager Actions

Link Managers are users who have the permission to approve or delete requests.

Permission takes the form of granting access to the Delete Queue as an Administrator via either:

1. Via the docuflow System Dashboard (/dflow/docuflowsys) choose the menu path Dashboards→Link Manager→Delete Management
2. Granting the manager access to the transaction /DFLOW/DOCUFLOWLINKM

By default, the report will display all outstanding delete requests (indicated by the checkboxes):

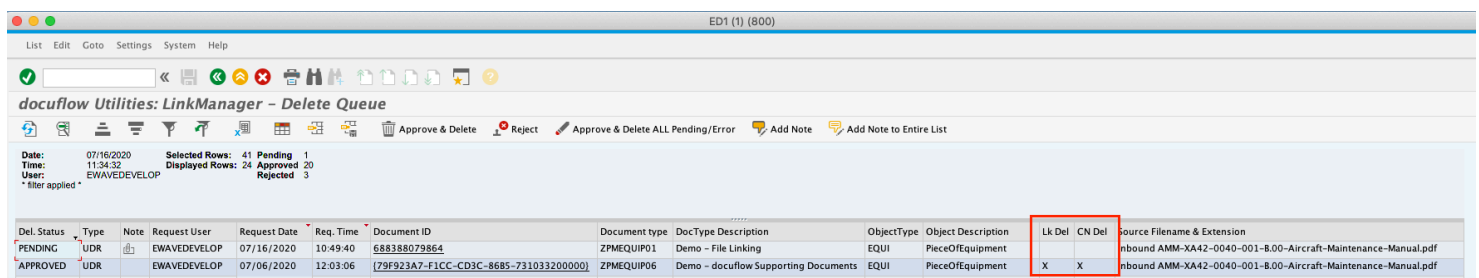


"Outstanding" is considered to be requests that need approval/rejection, OR, that encountered an error during the delete process (whether it be at the delete link or delete content level).

A request that is approved but an error occurs, is considered deleted from the end-user perspective, and it is the responsibility of the Link Manager to determine what the error is and what action needs to be taken.

A common scenario is that the actual content in the CMS system was deleted without SAP knowing, thus the delete content stage would fail

Below is a sample queue display:



Del. Status	Type	Note	Request User	Request Date	Req. Time	Document ID	Document type	DocType Description	ObjectType	Object Description	Lk Del	CN Del	Source Filename & Extension
PENDING	UDR		EWAVEDEVELOP	07/16/2020	10:49:40	<u>688388079864</u>	ZPMEQUIP01	Demo - File Linking	EQUI	PieceOfEquipment			mbound AMM-XA42-0040-001-8.00-Aircraft-Maintenance-Manual.pdf
APPROVED	UDR		EWAVEDEVELOP	07/06/2020	12:03:06	[79F923A7-F1CC-CD3C-8685-731033200000]	ZPMEQUIP06	Demo - docuflow Supporting Documents	EQUI	PieceOfEquipment	X	X	mbound AMM-XA42-0040-001-8.00-Aircraft-Maintenance-Manual.pdf

Once displaying the Delete queue, the following actions are available:

1. Approve and Delete - this is a single request action and requires you to highlight (click once) on the request to action. You will be prompted to add an optional note.
2. Reject - this too is a single request action and requires you to highlight (click once) on the request to reject. You will be prompted to add an optional note (in the case of rejections it is recommended to add a note).
3. Approve and Delete All Pending/Error - this will approve and delete all entries that are in those two status'. Approved and rejected entries are ignored.
4. Add note - this is a single request action and requires you to highlight (click once) on the request to action.
5. Add Note to Entire List - this adds a note to EVERY request on the screen regardless of status.

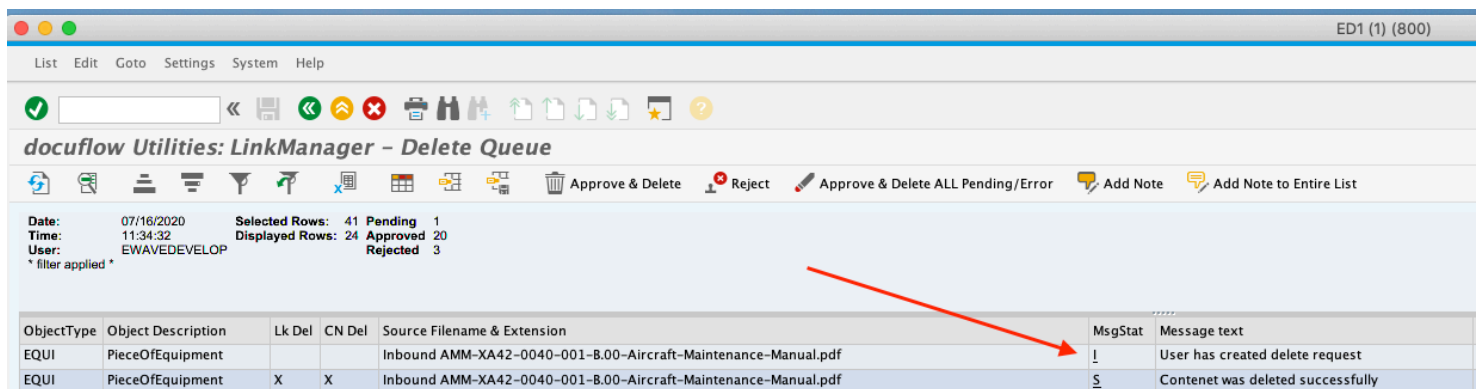
It is important to note the two flag columns on the report:

Lk Del - "X" indicates the link was deleted

CN Del - "X" indicates the content was deleted

Clicking on the underlined "Document ID" field will display the content prior to content deletion.

Scrolling to the right, you will also see the MsgStatus column. This provides a detailed log of all activity of the delete request. Click on the letter (I, E, or S) to show this detail.



Object Type	Object Description	Lk Del	CN Del	Source Filename & Extension	MsgStat	Message text
EQUI	PieceOfEquipment			Inbound AMM-XA42-0040-001-8.00-Aircraft-Maintenance-Manual.pdf	I	User has created delete request
EQUI	PieceOfEquipment	X	X	Inbound AMM-XA42-0040-001-8.00-Aircraft-Maintenance-Manual.pdf	S	Content was deleted successfully

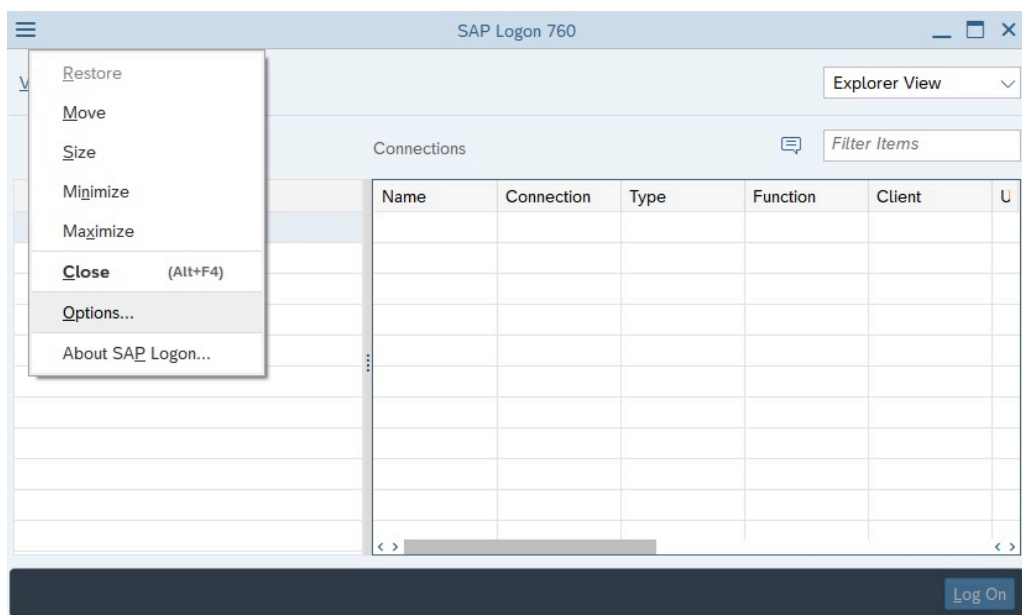
4.1.5.7. SAP Logon Pad Security Configuration

By default, the SAP Logon Pad generates a pop-up warning each time a local file is accessed or a file is accessed from an external content management system.

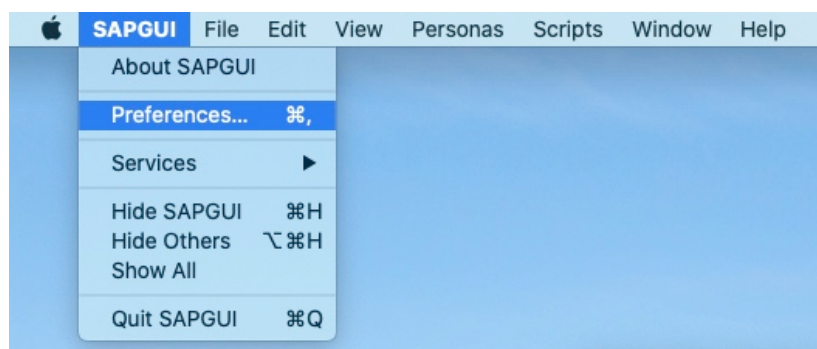
Due to the fact that docuflow DocManager does this...a LOT!, it is recommended that you disable this warning pop-up to avoid end-user frustration.

Follow the steps below to avoid this pop-up message:

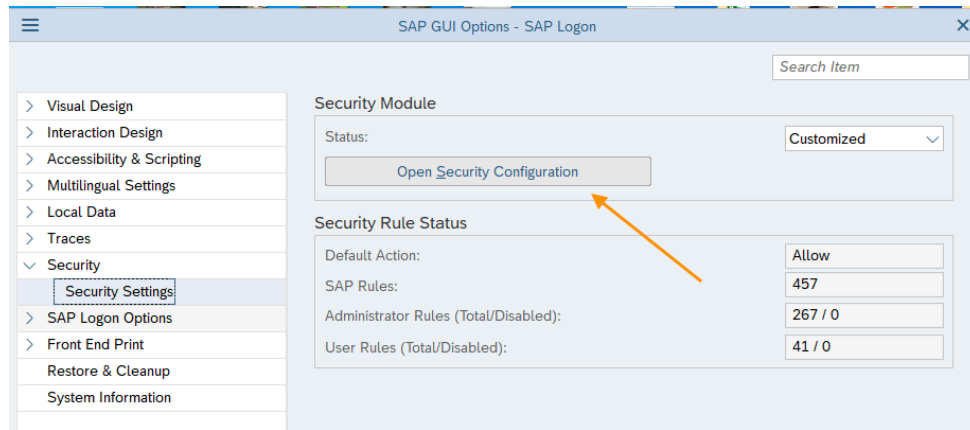
- From the SAP GUI Logon Pad, choose:
- for Windows: Click on the top left icon and select Options or Preference (depending on your SAP GUI version):



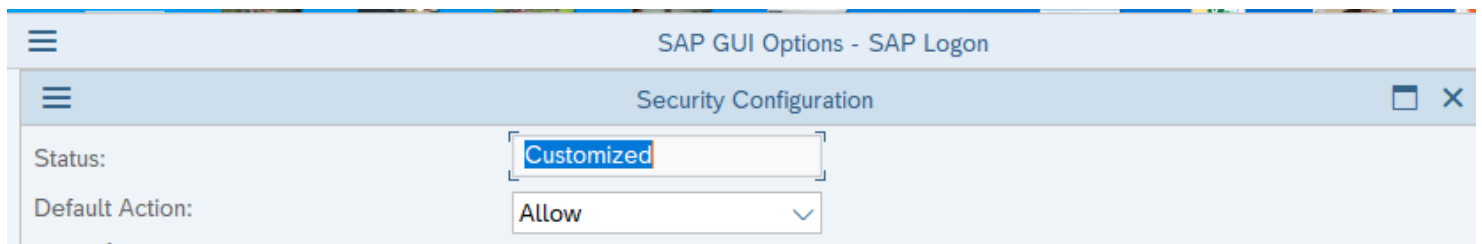
- for Java: via the menu bar path SAPGUI→Preferences:



- Location the Security folder on the left, highlight Security Settings, and then choose "Open Security Configuration":



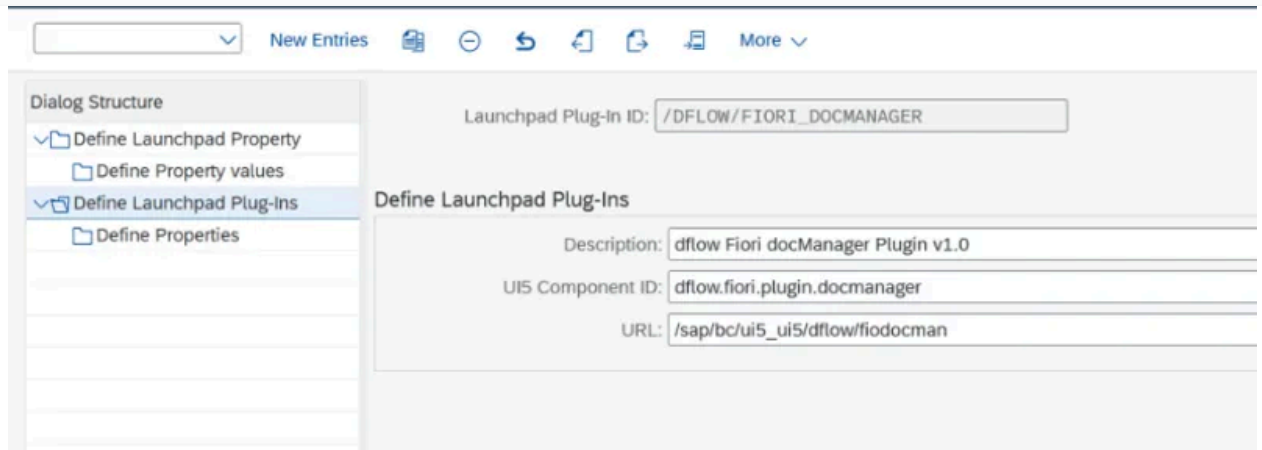
- Change the default Action to "Allow", and click OK/Apply:



4.1.5.8. Fiori DocManager - SAP Fiori Launchpad Plugin (cloned) (cloned)

Deployment Guide for DocManager Fiori plugin (v3.0)

Topic	Description
Plugin Features	
Required Back-End Product	
Required Back-End ECM Components	SAP Transport (ECM) add-on
Minimum UI5 Version Requirements	SAP UI5 1.114.11
Description	
Implementation Information - SAP Transports	Contact Account Manager or VersaFile point of contact.
Implementation Information - Installation & Configuration	- The app consists of front-end components (such as the user interfaces) and back-end components (such as the OData service). The back-end and front-end components are delivered as SAP transports and have to be installed in a system landscape that is enabled for SAP Fiori. Front-End Components - SAP BSP Application - /DFLOW/FIODOCMAN - ICF Path - /sap/bc/ui5_ui5/dfLOW/fiodocman - UI5 Component - dflow.fiori.plugin.docmanager.Component - Package - /DFLOW/DOCUFLOW_FIO_DOCMANAGER Back-End Components (ABAP) - OData Service – - External Service Name – /DFLOW/FIORI_DOCMANAGER_SRV - Technical Service Name - /DFLOW/FIORI_DOCMANAGER_SRV_0001 - Version – 0001 - Package – /DFLOW/DOCUFLOW_FIO_DOCMANAGER - Plugin Definition – - Navigate to transaction - /UI2/FLP_CONF_DEF to define properties for your plugin as shown in screenshot below. (taken care by transports so more of validation)



Dialog Structure

- Define Launchpad Property
 - Define Property values
 - Define Launchpad Plug-Ins
 - Define Properties

Launchpad Plug-In ID: /DFLOW/FIORI_DOCMANAGER

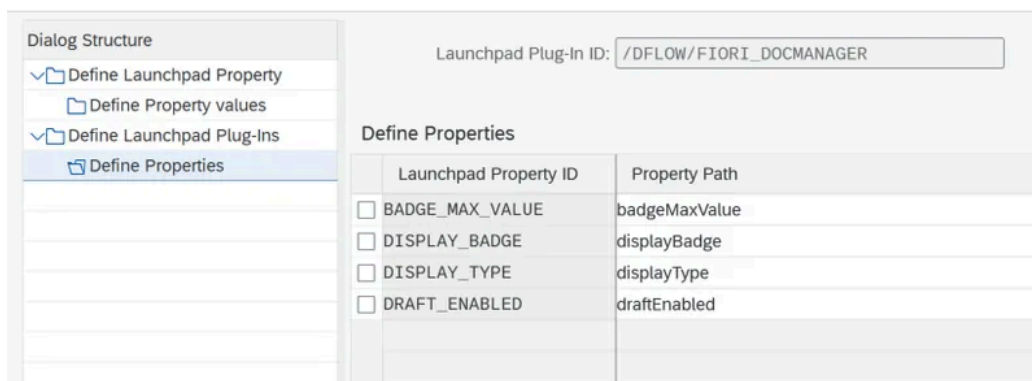
Define Launchpad Plug-Ins

Description: dflow Fiori docManager Plugin v1.0

UIS Component ID: dflow.fiori.plugin.docmanager

URL: /sap/bc/ui5_ui5/dflow/fiodocman

Properties: Below property definitions explained are key to the overall functioning of the plugin component. (These entries are delivered with package. Please consult your product support contact for any questions.)



Dialog Structure

- Define Launchpad Property
 - Define Property values
 - Define Launchpad Plug-Ins
 - Define Properties

Launchpad Plug-In ID: /DFLOW/FIORI_DOCMANAGER

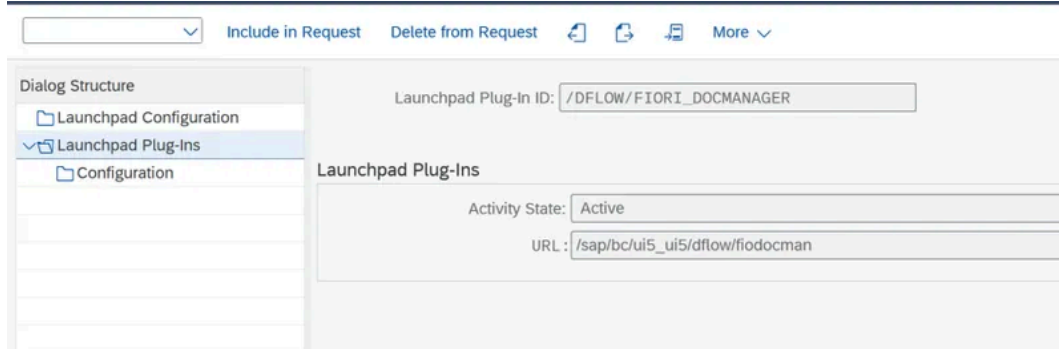
Define Properties

Launchpad Property ID	Property Path
☐ BADGE_MAX_VALUE	badgeMaxValue
☐ DISPLAY_BADGE	displayBadge
☐ DISPLAY_TYPE	displayType
☐ DRAFT_ENABLED	draftEnabled

- BADGE_MAX_VALUE = badgeMaxValue
 - The maximum number to display in the buttons' badge. If the number of attachments is greater than the maximum number, the maximum number followed by a plus sign will be displayed.
- DISPLAY_BADGE = displayBadge
 - Controls whether or not to display a badge next to the plugin button with the number of attachments.
- DISPLAY_TYPE = displayType
 - Is linked to the appearance or user experience of the plugin which basically is defaulted to 'Dialog' but alternatively 'Popover' is supported too.
- DRAFT_ENABLED = draftEnabled
 - This property is used to control the availability of draft feature in general.

Enabling/Activating the Plugin (Customizing):

/UI2/FLP_CUS_CONF → To make your plugin available in specific clients



Dialog Structure

- Launchpad Configuration
- Launchpad Plug-Ins
- Configuration

Launchpad Plug-In ID: /DFLOW/FIORI_DOCMANAGER

Launchpad Plug-Ins

Activity State: Active

URL: /sap/bc/ui5_ui5/dflow/fiodocman

Property Values: Providing values to the property defined is performed in this transaction. These values govern the app

behavior and shouldn't be changed without consulting product support. Values are case-sensitive too.

The screenshot shows the 'Launchpad Configuration' dialog. On the left, the 'Dialog Structure' tree is expanded to 'Launchpad Plug-Ins' > 'Configuration'. The main area shows the 'Launchpad Plug-In ID' as '/DFLOW/FIORI_DOCMANAGER'. Below this is a 'Configuration' table with the following data:

Launchpad Property ID	Property Value
☐ BADGE_MAX_VALUE	9
☐ DISPLAY_BADGE	true
☐ DISPLAY_TYPE	Dialog
☐ DRAFT_ENABLED	true

- UI Adaptation

The DocManager plugin was developed to replace SAP's standard Attachment Services UI, addressing its limitations and offering additional functionality. To prevent users from seeing the default "Attachments" tab in Fiori apps, use the Key User Adaptation approach to hide the standard "Attachment" feature wherever DocManager is enabled.

Our solutions team will help adapting apps during implementation but for further details please refer to standard SAP guide here to learn more about pre-requisites and approach.

- Plugin Activation Control: In addition to the core ECM configurations in order for the Fiori plugin to work we also deliver an configuration cluster /DFLOW/FDMCFG that can be accessed via T-Code **/DFLOW/SYS → Menu Path → Configure → "Attachment Services" → "Fiori Options"** to control the plugin activation per any standard or customer Fiori apps and along with other additional technical details. Our solutions team works with customers to identify potential SAP Business Processes to be enabled and accordingly help populate required information based on your SAP S/4HANA version. Below is a sample entry for activating DocManager plugin for Business Partner app.

The screenshot shows the 'Semantic Object' configuration table. The 'Dialog Structure' tree on the left is expanded to 'Semantic Object'. The table has the following data:

Semantic Object	Obj. Type	Enabled	Draft
☐ BusinessPartner	BUS1006	☒	☒

Validation Checklist

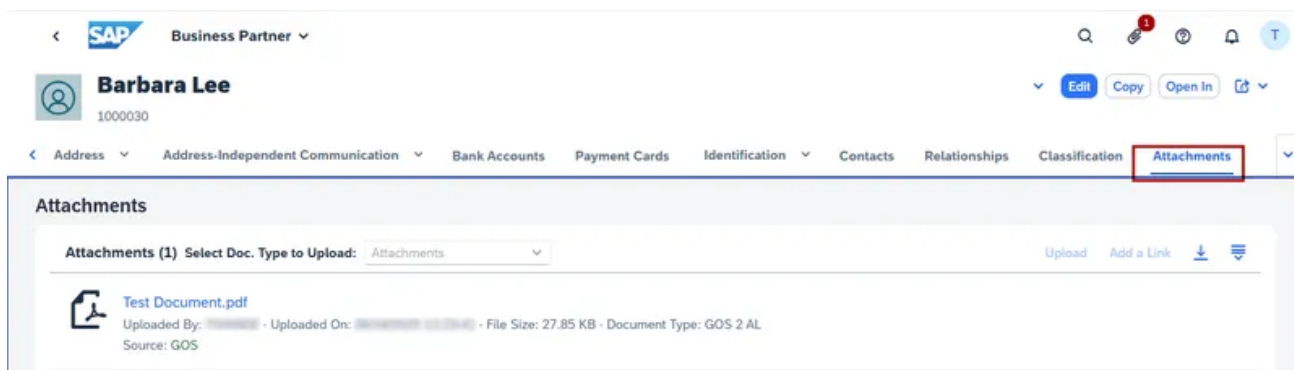
Technical Considerations – HDM Dependancies

- DocManager plugin works in absence of HDM functionality however if HDM is enabled below matrix will help clarify any limitation towards support for draft-mode in Fiori Apps.

HDM Status	GOS Framework	Draft-mode support
HDM Inactive	N/A	Yes
HDM Active	GOS Enabled	Yes
HDM Active	GOS Disabled	No

Technical Considerations – Standard Fiori Apps

- Only limitation with standard Fiori Apps would be with support for Draft-Mode attachment feature as the plugin only supports SAP Standard Fiori Apps that are built using BOPF framework that provides the ability to handle drafts. Learn more [here](#) about SAP draft handling concept and dependencies.
- URLs are well-formed and provide required information like Semantic Object, Action and Object Keys for the plugin to support effectively. Business Objects that are single-valued e.g. Purchase Order number or multi-valued object keys e.g. Purchase Requisition Item must be coded into URL with appropriate key-value pairing e.g.
PurchaseRequisition='1000489';PurchaseRequisitionItem='00030',
- Recommendation - When customers leverage Fiori DocManager plugin, we recommend to hide the standard "Attachments" tab in SAP Fiori Apps to ensure the users are not storing attachments using standard SAP options. "Attachments" tab can be hidden using Fiori "**Key User Adaptation**" as explained here - [click here](#)



Technical Considerations – Custom Fiori Apps

S4 CORE 106 or above (for draft mode support)

SAP S/4HANA FPS 2021 and above

The docuflow DocManager Fiori plugin is a reusable component that can be integrated to any standard SAP Fiori Object based app e.g. Purchase Order based application to enable attachment capabilities in SAP S/4HANA.

Attachments are an integral part of business applications and processes. In SAP S/4HANA, the DocManager Fiori plugin is offered as a central unified solution for uploading attachments across applications. The applications can integrate operations such as upload, download, delete and access to virtual links.

- Launch and validate Fiori DocManager plugin on all respective standard Fiori apps.
- Support customers to complete UI Adaptation on standard SAP Fiori Apps to hide "Attachment" tab.
- Identify and enable plugin to desired SAP Standard Fiori apps.
- Validate entries for Fiori DocManager Plugin definition and activation entries in SAP Fiori Launchpad customizing.
- Activate (if needed) SICF services for oData and Plugin.
- Import DocManager related transports into development environment.
- Validate Customers S/4HANA and Fiori Versions for compatibility and support.

- Custom Apps built using non-standard SAP Fiori floorplans need to be reviewed and tested on a case-to-case basis provided other points above meet the requirements.
- Custom Apps built using SAP's standard guidelines for handling draft will be able to take advantage of draft-mode support in DocManager plugin. (*Learn more [here](#) about SAP draft handling concept and dependencies*).
- Custom Apps must be associated to a BOR Business Object (standard or custom) to associate with ArchiveLink document types.
- Custom Apps must have a valid Semantic Object and Action associated with it.

To conclude, this page is updated from time to time and any discrepancies found please reach out to your customer success contact.

4.1.6. Custom API Usage

Custom API usage is typically for use when only the base docuflow API product is installed. Usage of the docuflow (S)earch (C)reate (R)ead (U)pdate (D)eleate API platform within SAP is outlined in detail the docuflow Developer Guide.

When using the SCRUD API set in a custom scenario and you have DocManager installed on your system, this section is may not relevant as you are encouraged to leverage DocManager rules to control API usage.

In the scenario you have only the base CMS API installed, you may choose to utilized the creation of Custom API Rules to house relevant config, especially the mandatory input parameters for the SCRUD API set such as CMS Profile, CMS Use Tag, Repository ID, Location, etc, allowing you to leverage a unique "Label".

You are responsible for what you use the available fields for as well the retrieval of them, as the SCRUD API set will only log itself in the docuflow API Gateway by default in terms of functionality.

To create a Custom API Rule when DocManager is not installed, enter transaction /dflow/docuflowsys and choose the menu path Configure->API Rules->Custom.

Please also consult the docuflow Developer Guide for more information.

4.1.7. Choice Lists

ChoiceList allows you to create list items from SAP in a SharePoint list which helps to track and manage data.

This can be used in many scenarios where you want to store a collection of documents under each list item.

Middleware also allows to link these list items to a document by attaching a metadata field of lookup type and value while creating or updating the document.

SAP requires two fields to be filled to create a list using the following BAPI: /DFLOW/ZCMSCREATECHOICELIST_WS

1) ZCHOICELISTNAMEPREFIX:

Required: Determines list name inside a specific site or subsite to create new items in SharePoint.

Example: EquipmentMaster.

2) ZFORCE:

Optional: A flag that allows you to clear all the existing items in a list and add new set of items to it.

Example: "X" to enable the flag.

4.1.8. Workflow Connect

Workflow Connect (WFConnect) allows code free triggering of SAP Workflow including container fulfillment for "inbound to SAP" processing (ie: also called SAPLink).

There are three parts to the WFConnect configuration:

1. Configure the docuflow Enterprise for the inbound SAPLink rule. Refer to the SAPLink section in this document for more information.
2. Enable the "En.SL WF Connect" checkbox in the appropriate DocManager rule. Refer to the DocManager configuration for information
3. Configure the WFConnect rule via transaction /dflow/docuflowsys and choose the menu path Configure→WF Connect.

There are two WFConnect rule elements to configure:

1. Triggers: this determine what method of workflow trigger to execute based on an inbound SAP DocType and label.
2. Metadata mapping (*optional*): if metadata is to be transferred to the corresponding workflow container, the inbound metadata must be mapped to container elements.

WFConnect Triggers

There are three workflow trigger methods available:

1. Publish BOR Event: This publishes an event on the SAP workflow runtime system. This in turn can be set as your triggering event in your workflow task or definition.
 1. Specify the SAP business object and event to publish.
2. Start Workflow: This starts your workflow directly calling the function "SAP_WAPI_CREATE_EVENT".
 1. Specify the workflow task or definition to start.
3. Start Class/Method: This starts your workflow by called the class-method specified.

WFConnect Metadata Mapping

Regardless of the triggering method, if inbound metadata is to be passed to the workflow container or event, you must configure metadata mapping.

1. Document Type: This is optional as you may configure mapping at the global level or only at the label level. If you choose to configure by Document type specify the ArchiveLink SAP ar_object.
2. Label: This is optional as you may configure mapping at the global level or only at the document type level. If you choose to configure by Label, this is a free form text field that can be used to split up document type or categorize larger groups of metadata buckets.

A global rule is indicated with both an empty document type and empty label.

4.1.9. AuditLink

We are in the final stages of migrating our document to our new online platform! This section will be updated soon, please check back in a few days!

4.1.10. DocMigrator

docuflow DocMigrator provides an SAP based toolset to migrate content from a source system to a target docuflow support repository, including SAP linkage maintenance, metadata transfer, plus additional follow up processing tasks.

DocMigrator supports the following **sources**:

1. SAP Database Attachments
2. SAP Database Notes (these are represented as .txt files in the target system or export)
3. SAP ArchiveLink Content Server
4. Custom ABAP Class-Method
5. FileSystem
6. docuflow ECM

And the following **target** end-points:

1. SAP Database
2. docuflow ECM
3. Content Server

A docuflow migration consists of **4 stages**:

1. Configuration
2. Load Migration Table - sets the stage for what is to be moved, where it is to go, and what options. These can be done via four methods:
 1. ArchiveLink link table entries (most common)
 2. .CSV file containing entries to be migrated and corresponding data files
 3. A custom ABAP Class-Method to create either of the above two options
 4. Content stored in your SAP database
3. Migrating Entries in the Migration Table - migrates the content w/ metadata if applicable and updates the status of the Migration Table
4. Perform any post-processing tasks or utilities - allows you to delete source content and links if applicable, plus more utilities

A very useful utilities is the ability to export content to a filesystem for review or validation.

In addition, for SAP Database Notes and SAP Database Attachments, the DocMigrator can be used to see what you have in your database if you are unsure. Often users will have been allowed to store content in the SAP Database without document administration or business owners knowing.

In addition, you can delete the entire, or a portion (using all the Source and Target selection criteria) of the docManager load/migration table using the "Delete Migration Table" option. Take extreme caution when doing so as your entries cannot be retrieved (albeit they can be re-loaded).

As you process your migration steps, the DocMigrator updates link entries in the Migration table with the status and flags each step as it is completed.

At any point you can display your Migration Table and its current state using the "Display" option.

Compiled metadata (if applicable) and status messages can be drilled down/single clicked into from the ALV report for more detail (indicated by an underline).

It is highly recommended to read the **Migration Concepts and Strategies** section on how to architect and execute your migration as it pertains to docuflow setup.

In addition, each content repository may have an additional notes or best practices section , please ensure you reference if applicable.

To configure DocMigrator, enter transaction /dflow/docuflowsys and choose the menu path Configure→DocMigrator.

To enter the DocMigrator Dashboard, enter transaction /dflow/docuflowsys and choose the menu path Dashboards→DocMigrator.

4.1.10.1. Migration Table Flags

Over the entire migration process (including optional steps and flags), below are the list of flags set (if applicable) in the migration table as DocMigrator is executed across the different phases of the migration, post-processing, etc.

These flags are included as input to DocMigrator, allowing you to report on, or execute migration tasks based on.

eg: Only process migration table entries for a particular SAP object and document type that have NOT been migrated yet - Target Cont. Migrated NE 'X'

1. Source MD Exists (if applicable)
2. New MD Exists (if applicable)
3. Source Validated (during validation runs only)
4. Target Cont. Migrated (during migration)
5. Metadata Migrated (if applicable)
6. Target Link Created (during migration)
7. Src Content Exported (used as an optional utility)
8. Src Content Deleted (used as an optional post-processing step)
9. Src Link Deleted (used as an optional post-processing step)
10. Target Content Deleted (used as an optional utility)
11. Target Content Link Deleted (used as an optional utility)
12. Target Content Exported (used as an optional utility)

These flags are extremely useful when running the migrator in Display mode to report on subsets of your migration content and activity.

4.1.10.2. Configuration

There are three levels of configuration for DocMigrator:

1. SAP ArchiveLink Document Class to Mime-type Mappings - these entries are in the table TOADD and contain the file extension (document class) to mime-type mapping. This is required for all file types to be migrated regardless of their source.
2. DocMigrator - source repository id setup and mapping, doctype mapping, and source metadata mapping.
3. DocManager Rule - all configuration needed to store content from SAP to your target repository such as location, metadata, etc. *All business objects and content must be configured in DocManager.*
4. If you are using flat files as input, create a logical filename "ZDOCUFLOW_MIGRATOR_SOURCE_FILES"
5. If you are export file copies, create a logical filename "ZDOCUFLOW_MIGRATOR_EXPORT_COPY_FILES"

SAP ArchiveLink Document Class to Mime-Type Mappings

SAP comes with pre-populated mappings. If a combination is missing in your TOADD table, you can add new entries via the menu path:

SPRO→SAP Reference IMG→SAP Net weaver→Application Server→ Basic Services→Archive Links→Basic Customizing→ Edit Document Classes

DocMigrator

Locate the DocMigrator config under the docuflow System Dashboard (txn /dflow/sys) and the menu path Configure→DocMigrator

- Source Connectivity
 - Not required for SAP database attachments or notes
 - Choose between a docuflow endpoint, a non-docuflow ArchiveLink content server, the SAP database, or File input for each repository ID that will be used in the Load Table
 - Provided CMS Use Tag. For non-docuflow endpoints, this is for description purposes only. For docuflow endpoints, this will be used along with standard docuflow configuration to determine the RFC destination to perform the READ from.
- Repository Mapping
 - Not required for SAP database attachments or notes
 - This is required in all cases except where you are reading from the same source docuflow repository as the target, or if you have/will populate the Load table from an ABAP class/method or file method (as those determine the target repository ID).
 - It is strongly recommended not to sharing repository IDs for different repository type.
- DocType Mapping
 - Not required for SAP database attachments or notes
 - Mapping of document types is optional in all cases.
 - Typically only recommended if SAP security is to be different between source and target content, or if you want to merge document types.
 - Remember that new document types typically requires updated SAP Security roles/profiles.
 - Wildcards (*) can be used in the Source Object Type and Source Document Type columns and take second precedence over an exact matching entry.
- Original MD Map
 - Not required for SAP database attachments or notes

- If metadata is supported from the source content, mapping of original/source metadata names to new target metadata field/property names can be done.
- If only new metadata from SAP is being sent to the target system as part of the migration, this mapping is not required.
- Wildcards (*) can be used in the Source Object Type and Source Document Type columns and take second precedence over an exact matching entry.
- [SAP DB as Source Map](#)
 - Required *only* for SAP database attachments or notes
 - For all SAP database attachments or notes you must specify a target repository ID and doctype per SAP Object.
 - A wildcard (*) can be used in the Source Object Type column and take second precedence over an exact matching entry.

DocManager Rule

Locate the DocManager config under the docuflow System Dashboard (txn /dflow/sys) and the menu path Configure→API Rules→Document Manager.

Since DocMigrator utilizes DocManager for all content to be migrated to determine location, metadata, etc, a DocManager rule must exist for every business object and doctype combination to be migrated.

For detail on this configuration, refer to the DocManager section of the docuflow Solution Guide.

Logical Filename: Source Files

Create the logical filename "ZDOCUFLOW_MIGRATOR_SOURCE_FILES" via transaction FILE. Assign to a logical path of the same name and configure that logical path to the directory to save the log files. The physical path must have the filename placeholder as <FILENAME>.

Example below:

Log. File	ZDOCUFLOW_MIGRATOR_SOURCE_FILES
Name	docuflow Migrator: Source Files
Physical file	
Data format	DIR
Applicat.area	
Logical path	ZDOCUFLOW_MIGRATOR_SOURCE_FILES

Logical path	ZDOCUFLOW_MIGRATOR_SOURCE_FILES
Name	docuflow Migrator: Source Files

Syntax group	WINDOWS NT Microsoft Windows NT
Physical path	\\directory\ <FILENAME>

Logical Filename: Export Copy

Create the logical filename "ZDOCUFLOW_MIGRATOR_EXPORT_COPY_FILES" via transaction FILE. Assign to a logical path of the same name and configure that logical path to the directory to save the log files. The physical path must have the filename placeholder as .

Example below:

Log. File	ZDOCUFLOW_MIGRATOR_EXPORT_COPY_FILES
Name	docuflow Migrator: Export Copy Files
Physical file	
Data format	DIR
Applicat.area	
Logical path	ZDOCUFLOW_MIGRATOR_EXPORT_COPY_FILES

Logical path	ZDOCUFLOW_MIGRATOR_EXPORT_COPY_FILES
Name	docuflow Migrator: Export Copy Files

Syntax group	WINDOWS NT Microsoft Windows NT
Physical path	\directory\<FILENAME> Short description of syntax group

4.1.10.3. Execution: Load

Loading the migration table sets the stage for:

- what content is to be migrated and from what source (a docuflow endpoint, file, SAP content server, SAP DB)
- any SAP repository ID or doctype mapping (via configuration)
- where the content is to be migrated to (ie: what docuflow endpoint repository ID)

This can be done using 3 separate "Source" methods:

1. ArchiveLink Links
2. .CSV file containing file references and required ArchiveLink data
3. Custom ABAP Class-Method
4. Content stored in your SAP database

Regardless of the method used, it is highly recommended to enter a useful descriptor/tag that identifies this migration run.

You can run the load process in test mode without making any changes to the load table.

The results of the Load phase are displayed on the screen according to the option "Display Results in ALV" checkbox. If this option is chosen, all entries are displayed in the ALV report format, else a simple message indicating the number of entries loaded successfully, with info message, or errors is displayed.

If ALV is chosen, the header will display information as to the breakdown of entries loaded successfully, with info message, or errors.

Be careful when choosing "Display Results in ALV" as for large datasets this will significantly increase runtime and potential SAP system memory usage and timeouts. It is therefore suggested to use the "Display" mode of the migrator afterwards instead of displaying the results in ALV format.

Clicking on the source or target hotspot/hyperlink in the ALV allows you to view the image.

The DocManager rule and/or metadata mapping *does not happen* at the Load stage. This is done in the Migrate phase.

No additional flags are set at this stage, as the Load phase creates the entries in the migration table.

ArchiveLink Links

This method relies entirely on existing link entries in your ArchiveLink link tables (TOA01, etc).

1. Specify the parameters in the section "Source ArchiveLink Selection Criteria" to determine which entries will be read by the DocMigrator
2. Each entry found will equate to an entry in the load table
3. Before writing the entry to the load table, DocMigrator reads the DocMigrator configuration to determine repository ID and doctype mapping

The source of this content must be accessible through an ArchiveLink content server or equivalent to be read.

.CSV File

This method relies entirely on an input .CSV file that you compile yourself to add entries to the Migration table. The source type of these entries are data files only (ie: *you cannot* use this method to load ArchiveLink link entries into the migration table).

The .CSV file must follow the below columns, in order:

- sap_object
- object_id
- archiv_id
- arc_doc_id
- ar_object
- ar_date
- del_date (optional)
- reserve (this must align with your viewing platform, thus either * or file extension)
- source file path
- source file name

These files must be accessible by the DocMigrator program, be sure to take into consideration if the migration run will be executed in the background (and therefore the files must be accessible by the SAP server).

Custom ABAP Class-Method

This method relies entirely on an ABAP class-method that you can develop yourself to add entries to the Migration table, which takes the form of ArchiveLink links.

Refer to /DFLOW/CL_MIGRATOR_HELPER⇒GET_ABAP_LINKS_EXAMPLE for an example of this logic.

The custom class method MUST set the following:

ArchiveLink Source:

- EV_SOURCE_TYPE:
 - = "A" for ArchiveLink content server
 - = "D" for docuflow ECM
- sap_object
- object_id
- archiv_id
- arc_doc_id
- ar_object
- ar_date
- del_date (optional)
- reserve (this must align with your viewing platform, thus either * or file extension)

SAP Database

With the information available to us for this as a source being limited, this method relies on specifying and SAP Business Object and the corresponding object ID "range".

Not that as part of the LOAD phase, unlike the other sources, mime-type and therefore extension (reserve) will not be known until migration is executed (or even in validation mode).

An example would be "EQUI" for Equipment, and then specifying the object ID "range", ie: the equipment numbers that you like to migrate the content for.

Select either/both Attachments or Notes options. The SAP Object and Object ID (the SAP Key) are optional. If you are unsure of what is in your database, you can run this wide open to see!

4.1.10.4. Execution: Migration

The migration process is the phase in which for the entries selected (via the selection screen) in the migration table, content is READ from the source type (ArchiveLink content server, docuflow ECM endpoint, data file, or SAP DB) and created in the target system (ArchiveLink content server, docuflow ECM endpoint, or SAP DB).

This process uses all the input criteria available for selection parameters.

Below is a description of your migration options.

Keep Original SAPDocID

This option allows you preserve the source SAPDocID (ie: <link table>-arc_doc_id. This should only be done if required by another 3rd party or other dependency.

This option is only available for ECM target endpoints.

Check Multiple Links

This option will check if a other migration entries exist for the the same:

1. source repository ID
2. source document ID (ie: SAPDocID / arc_doc_id)

This is only valid if you have an artifact assigned to more than 1 SAP record.

If you know or are unsure if you have 1 to many linkages, this option is safe to choose. There is no harm in choosing this option however it comes with decreased performance as each entry, before it migrated, reads the entire migration table to see if an entry with the same source repository ID/document ID has been migrated.

If it finds an entry that is migrated, it does not "re-migrate" the document. Instead, it preserves the 1 to <n> multiple link entries and simply creates the appropriate ArchiveLink link and marks the entry as migrated.

Additional note: if you have or suspect multiple links, it is safest to choose single stream processing (For more information on these consideration, refer to the section/node "Migration Concepts and Strategies - Box" below). In theory, if parallel batches contain entries of the same source repository ID/document ID combination, and if they are attempted to be migrated at the same time, duplicate documents in your document repository could occur. This is the worst case. The migration will still be valid.

Force RFC (recommended)

This option allows you to override the RFC that the DocManager rule for the target entry will determine. If this is shared with day-to-day docuflow processing, it is highly recommended to have a dedicated RFC (and migration docuflow connector) to migration only. For more information on these consideration, refer to the section/node "Migration Concepts and Strategies - Box" below.

Force Source Rep ID

This option allows you to force a source repository ID different than the source repository ID in the load table.

Should be used for unique testing in non-production systems only and should only be used if required.

Force Unique Filename

This option allows you to force a unique filename in the target content repository, and in some cases (such as Box), this is mandatory to prevent versioning.

The target filename will have the format <original filename>_<unique SAP UUID>.<original file extension>

Replace Characters with SP

This option allows you to identify a list of special characters all in a sequence (eg: %, or %`@) and have them replaced by a space.

Include Metadata (docuflow ECM only)

Get/Map Source Metadata to Target Metadata

If the source is a docuflow ECM endpoint - metadata can be *read* from the source and moved along with the content to the migrated.

This requires the DocMigrator configuration described above "Source Metadata Mapping". Only properties configured will be transferred.

You can make source metadata mandatory, in which case the entry will be flagged with an error if not found.

Enhance/Compile New Target Metadata

If the source does not support metadata, metadata can still be *added* to the content being migrated via DocManager, OR in addition to. This metadata can come from any available dataset since you can create an ABAP class-method to compile that metadata, thus you have full control. Of course, you can also add SAP metadata code free using the metadata map option in DocManager.

You can make New Target metadata mandatory, in which case the entry will be flagged with an error if not found or compiled via the found DocManager rule.

**If both sources of metadata are leveraged, the metadata compiled by DocManager will override any metadata with duplicate target property names.*

Validate Only / Store Content

You can choose to run a validation run only to ensure your configuration and metadata options are correct. A validation run will read data from the source, compile metadata if applicable, but will NOT create the content in the source system nor an ArchiveLink link. In a validation run, the following flags are set:

1. Source MD Exists (if applicable)
2. New MD Exists (if applicable)
3. Source Validated

Auto Delete Return Table on Success - This flag helps delete the return table upon success.

Auto Delete Source Content and Link on Success (SAP Db Only). - This flag is relevant for Source as SAP Database or GOS Attachments and if True, post migration the Source Content and link will be deleted including the reorganization of the database space used by the attachment in source db/repository.

Auto Delete Source Link on Success (ArchiveLink Only). - Upon successful migration of the source document this flag if set will ensure deletion of the Archivelink entry from the link table defined in OAC3 against the Business Object and Source Archive Doc Type and Source Repository.

Store content WILL create migrate the content (with metadata if applicable) and if successful, will also create an ArchiveLink link entry and set the following flags:

1. Source MD Exists (if applicable)
2. New MD Exists (if applicable)
3. Target Cont. Migrated
4. Metadata Migrated (if applicable)
5. Target Link Created

Note that if you migrate an entry twice, the DocMigrator will skip that entry and place a message in the status table.

The results of the Load phase are displayed on the screen according to the option "Display Results in ALV" checkbox. If this option is chosen, all entries are displayed in the ALV report format, else a simple message indicating the number of entries loaded successfully, with info message, or errors is displayed.

If ALV is chosen, the header will display information as to the breakdown of entries loaded successfully, with info message, or errors.

Be careful when choosing "Display Results in ALV" as for large datasets this will significantly increase runtime and potential SAP system memory usage and timeouts. It is therefore suggested to use the "Display" mode of the migrator afterwards instead of displaying the results in ALV format.

Clicking on the source or target hotspot/hyperlink in the ALV allows you to view the image.

4.1.10.5. Post-Processing Tasks

Delete Source Content

You can use DocMigrator to delete source content after a successful migration. The source link must exist for the DocMigrator to do so. This is only applicable to source types:

1. SAP DB - *In this case both Source Content and Link is deleted at same time.*
2. ArchiveLink Content Server
3. ArchiveLink docuflow ECM

Actual Source File deletion is not supported at this time for Archivelink based attachments.

This process uses all the input criteria available for selection parameters.

Delete Source Link

You can use DocMigrator to delete source ArchiveLink links after a successful migration. Typically this is done after validation of your migration. If during the migration and validation time frame, you wish to not show duplicate entries in the SAP attachment list (if using the DocManager attachment list), you can optionally hide the target entries via the DocManager config until after all is completed and verified. Then unhide the new attachments and delete the old, or optionally keep the old links and simply hide them as well.

This process uses all the input criteria available for selection parameters.

The results of the Load phase are displayed on the screen according to the option "Display Results in ALV" checkbox. If this option is chosen, all entries are displayed in the ALV report format, else a simple message indicating the number of entries loaded successfully, with info message, or errors is displayed.

If ALV is chosen, the header will display information as to the breakdown of entries loaded successfully, with info message, or errors.

Be careful when choosing "Display Results in ALV" as for large datasets this will significantly increase runtime and potential SAP system memory usage and timeouts. It is therefore suggested to use the "Display" mode of the migrator afterwards instead of displaying the results in ALV format.

Clicking on the source or target hotspot/hyperlink in the ALV allows you to view the image prior to deleting the link or content, but of course is not supported after.

4.1.10.6. Additional Utilities

Export Copy of Source / Target Content

DocMigrator can be used as a content exporter for any entries in the Migration Table. Specify a valid path available to the program (ie: foreground or background).

This process uses all the input criteria available for selection parameters.

Delete Target Content

You can use DocMigrator to delete target content. This is typically only done when backing out a migration or portion thereof and should be approached using EXTREME CAUTION.

This process uses all the input criteria available for selection parameters.

Delete Target Link

You can use DocMigrator to delete target ArchiveLink links. This is typically only done when backing out a migration or portion thereof and should be approached using EXTREME CAUTION.

This process uses all the input criteria available for selection parameters.

The results of the Load phase are displayed on the screen according to the option "Display Results in ALV" checkbox. If this option is chosen, all entries are displayed in the ALV report format, else a simple message indicating the number of entries loaded successfully, with info message, or errors is displayed.

If ALV is chosen, the header will display information as to the breakdown of entries loaded successfully, with info message, or errors.

Be careful when choosing "Display Results in ALV" as for large datasets this will significantly increase runtime and potential SAP system memory usage and timeouts. It is therefore suggested to use the "Display" mode of the migrator afterwards instead of displaying the results in ALV format.

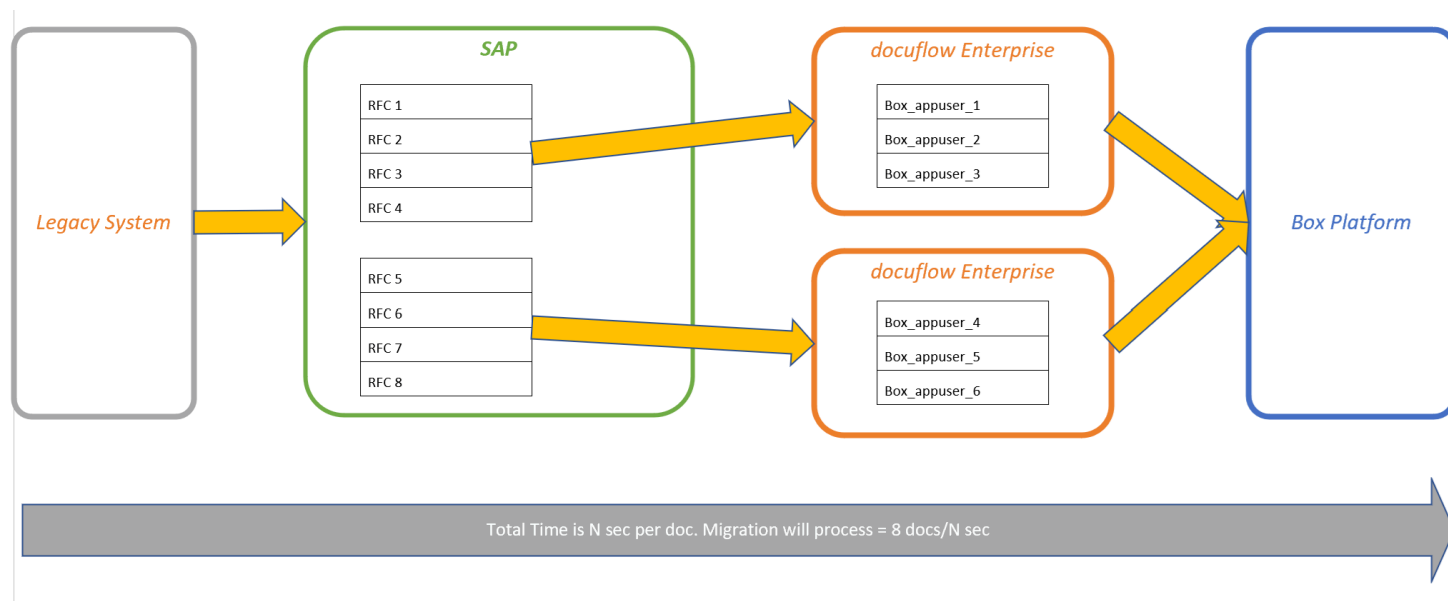
Clicking on the source or target hotspot/hyperlink in the ALV allows you to view the image prior to deleting the link or content, but of course is not supported after.

4.1.10.7. Migration Concepts and Strategies

Migration content from a source to docuflow Target with DocMigrator is a combination of DocManager business rules (SAP Business Objects and DocTypes), RFCs and their connections, and SAP background jobs.

All of these, plus consideration of day-to-day docuflow processing and how quickly you want to migrate your content determines the constellation of these components.

The diagram below provides a high level overview:



Typically there is a timeline for a migration, and together with the volume of content to be migrated, a seconds/per document or document/per second target is determined.

The below sections outline the components, consideration, and best practices involved in order to achieve a successful migration.

Along with overall best practices, each content repository may have an additional notes or best practices section, please ensure you reference if applicable.

Sharing of DocManager Rules and RFCs

The first determination is whether content to be migrated will be sharing DocManager rules (which are at the SAP Business Object and/or DocType level) with day-to-day docuflow processing.

Things that will force separate DocManager rules:

- If there is a need for separate folder locations in Box
- If there is a need for separate SAP security
- If there is a need for different meta-data

In short, RFC destinations (and their number of RFC connections per destination) are how docuflow can be load balanced to allow:

1. multiple docuflow Migrator SAP background jobs to be ran in parallel
2. sharing or separation of RFC processing between migration processing and day-to-day processing

In typical day-to-day docuflow processing, the DocManager rules determine the RFC destination, however DocMigrator can override this to force use of dedicated RFC destinations at runtime (more on this below).

Parallel vs. Single Stream Processing

The second determination is whether your volume vs. timeline/milestones require multiple migration processes to run in parallel, or whether a single processing stream will suffice, in other words, how fast do you want or need your content to be migrated.

Single Stream

- Additional consideration into carving and chunking migration is not required
- The only consideration is whether to share existing RFCs with day-to-day processing using a higher number of RFC connections (more on this below)

Parallel Processing (most common and recommended)

- Ideally is executed using separate RFC destinations than day-to-day processing, and further ideal over separate docuflow middleware connectors
- The number of dedicated SAP background tasks available for migration dictates how many parallel processes can be executed
- For Parallel Processing, there are 2 ways to carve and chunk a migration within the docuflow DocMigrator, both leverage the input/selection criteria to the DocMigrator:
 - ArchiveLink Link parameters (most common and recommended)
 - All fields in TOA01 are available, both for "source" and "target" link entries in the migrator table once loaded/staged
 - Typical and easiest levels of carving are SAP Business Object and/or DocType
 - The advantage of using these two elements is it typically prevents from job running in parallel for the same migration load table entry at the exact same time
 - The other fields in a link entry can be used (such as SAP record/object_id range) but extra caution must be given to not processing a load table entry at the same time over multiple jobs
 - Loading with custom ABAP to assign unique tags (requires development)
 - This method typically has the chunks set by Tag name set at time of the load process, and thus migration of the load table is done at the tag level

*note: it is imperative that parallel jobs do not have input criteria that allow it to process the same rule at the same time - they must be "unique" jobs to avoid runtime conflict. Running a migration for a document that is already migrated is OK, docuflow will not migrate or process a document twice.

RFC Destinations and Connections

Regardless of the processing style chosen able, but imperative to consider for Parallel Processing, is how to load balance the migration process over RFC destinations and RFC connections.

Sharing DocManager Rules

If sharing the same DocManager rule as day-to-day processing is not concern, and load is not a concern, DocMigrator can be run to leverage the DocManager rule in it's entirety, including which RFC to use

In this scenario, it is recommended to increase the number of RFC connections at the middleware level for each RFC to be shared, thus reducing the chance of migration activity and day-to-day processing waiting for an available RFC destination connection

Forcing RFC Destination(s)

To take full control over which RFC destination is to be used, DocMigrator can be run explicitly specifying the RFC destination to use for that background job by using the "Force RFC" input parameter **(most common and recommended)**

- In this scenario, ideally use 1 RFC destination per background job
- If background jobs are to share an RFC destination, increase the number of RFC connections to match the number of SAP jobs running against that RFC at the same time (assuming no day-to-day processing is using that same RFC)
- Ideally each background job uses 1 RFC destination with 1 RFC connection
- It is recommended to have a maximum of 10 RFC destinations per docuflow middleware connectors
- It is recommended to have a maximum of 10 RFC connections per RFC destination
- In theory, that is 100 RFC destination/connection combinations registered in your SAP Gateway, therefore mind your maximum allowed users as logged in clients in SMGW

Additional Best Practises

- SAP Background Jobs
 - **Never have migration table entries shared between background jobs for parallel processing.** A potential collision could occur to process the same entry at the same time across more than 1 background job
- Unique Filenames from Source Content
 - If your source files may contain non-unique filenames, and there is a chance they may be migrated to the same repository location for a CMS that does *not* allow same name files, **versioning must be prevented**. This can be handled in two ways:
 - 1) In the DocMigrator, choose the "Force Unique Filename". This will append a unique UUID stamp to the source filename.
 - 2) In the docuflow middleware, at the RFC destination level - enable "Error On Duplicate". This will result in an error for documents attempted to be migrated to the same folder if there is a file with that existing name. These errors can then be triaged after the fact to either modify the filename in the source system, or re-run the docuflow migration for those errors and choose "Force Unique Filename" above.
- RFC's and docuflow Middleware
 - Ideally avoid sharing docuflow migration RFC destinations with production day-to-day docuflow processing RFC destinations
 - Ideally avoid sharing docuflow migration with production day-to-day docuflow processing across the same middleware instance
 - Ideally have dedicated 1 or more docuflow middleware instances for migration (unless shared doctypes and DocManager rules for migration and day-to-day)
 - Consider total number of RFC destinations/RFC connections in your SAP Gateway maximum user count
 - For production day-to-day docuflow processing and migration processing, servers hosting docuflow middleware should be dedicated to docuflow only
 - Often 1 non-prod docuflow middleware instance/server is enough, ideally 1 per SAP instance (DEV/QUAL/PROD/TRAINING) but that is optional/nice to have. This only affects Inbound to SAP processing/testing as when you add a document to Box, how does the connector know which SAP system to go to (hint for the fancy pants out there: yes you can add more than one metadata field at the SAPLink rule level in the mdfilters string).
 - If your source data is also from a docuflow endpoint - ensure the CMS Use Tag in the DocMigrator configuration is separate from DocManager business rules and/or is configured to a dedicated RFC as a docuflow READ will be performed on that RFC
 - Also important to ensure there are multiple RFC connections for a READ RFC destination to ensure maximum throughput and performance.
- Repository ID's and Doctype's
 - Mapping to new Repository ID's is almost always required and highly recommended
 - Mapping to new document types is optional. Remember that new document types typically requires updated SAP Security roles/profiles.
 - theory recap: ArchiveLink uses S_WFAR_OBJ authorization object which has repository id, SAP business object, and doctype
 - thus if repository ID is already part of your security model, you will have to accommodate the new repository ID at a minimum

- recap section “Sharing of DocManager Rules and RFCs” above to determine if you have to have new doctypes (in addition to reasons you may have as well)
- Deleting Links / Attachments List
 - During migration of SAP Business Objects/Doctypes buckets, if the source content is to remain intact, and if using then docuflow DocManager attachment list ONLY, consider “hiding” those from the attachment list until migration for that combination is complete (DocManager config)
 - After migration buckets of SAP Business Objects/Doctypes are completed, and if using then docuflow DocManager attachment list ONLY, consider “hiding” those from the attachment list until final verification (DocManager config)
 - Deletion of source links is only recommended after a verification window post-migration. This is also optional if using the DocManager attachment list, as you can also simply hide the orinal links as well.

Performance Notes

- Set logging in docuflow middleware to ERROR only. Less logging, less writing to disk
- Increase the JVM size for docuflow middleware server to 2048 initial/4096 max.
- Have docuflow Profile Wizard closed during processing
- Including metadata as part of the migration requires an additional API call as part of the “create”, but allows for the migration to process in one enchilada. However, If migration milestones are short and if metadata isn't required Day 1, or if API max calls are a concern, consider transferring metadata as a second wave using the docuflow MDPro in SAP to execute a mass push of metadata after the fact.
- Use Box folder numbers instead of names, this reduces the number of API calls and increases performance **significantly**.

4.1.10.8. Box - Additional Best Practices

Box Rate Limits

Before you go and create a hundred RFC destinations with 10 RFC connections each to migrate a million documents over lunch, keep in mind there are documented Box Rate Limits at two levels:

1. Overall Enterprise Account - 100,000 API calls per month are allowed
 1. docuflow uses 3 API calls per create, but if need be, this can be decreased to 2 by not transferring metadata (which you may or may not need, or you can always transfer at a later date)
 2. This limit is shared at the Enterprise level, so consider day-to-day processing as well during migration window
2. App Users - Each app user is allowed 1,000 API calls per minute
 1. To work within App User limits, docuflow allows multiple App Uses to be configured per middleware connector. Yay docuflow.

Example: If a document overall takes 3 seconds to read from the source and create in Box with metadata, and you have 100,000 documents to migrate, and you can carve out chunks to process 3 background jobs in SAP over dedicated RFC connections:

- You can move 3 documents every 3 seconds, thus your virtual migration time is 1 second per document.
- At 100,000 documents, you can move them in 100,000 seconds = *27.8 hours.

*keep in mind max API calls as the above example would use $100k * 3$ API calls per = 300k API calls total

Box App Users

- Never share app users across docuflow middleware instances
- Consider naming convention of docuflow_mig_<app user number> so you know who/the migration process added the content to box

4.1.11. Accounts Payable

Introduction

docuflow Accounts Payable is an extremely powerful SAP based Invoice Processing and Approval workflow built on the SAP workflow runtime system.

Even with its powerful design, the docuflow Accounts Payable Workflow is simple and fully configurable to handle all types of invoice processing: from straightforward to complex to fully customized.

Compatible with any scanning or ingestion method to your content repository, docuflow AP is a powerful and flexible solution to process, track, manage, and automate your accounts payable process - resulting in greater efficiencies and promoting shorter invoice processing times, enhanced exception management, and much more - allowing you to pay invoices faster, easier, and achieve vendor discounts.

Features

Below are some of the docuflow AP features:

- automatic creation of invoices
- data pre-population for manual invoice creating
- full integration with your content repository
- enhanced exception management
- auto posting
- built in agent management
- multi level approvals inside and outside of SAP (including mobile phone)
- ASL integration
- built in notifications
- leverages your standard SAP screen and functions to ensure 2 and 3 way matching and data validation
- back-population of metadata to your content management system along entire invoice lifecycle
- deadline monitoring to ensure no invoices stall
- powerful and fast searching and tracking of invoices
- work queues with powerful sort, filter, and load balancing capabilities
- based on the powerful and robust SAP workflow runtime system
- a suite to tools to support administrative functionality including year end rollover
- supports delegation
- works with SAP UWL
- works with SAP Fiori
- fully configurable
- code-free implementations
- customizable
- fast implementation timelines
- no SAP workflow in-house skill set required

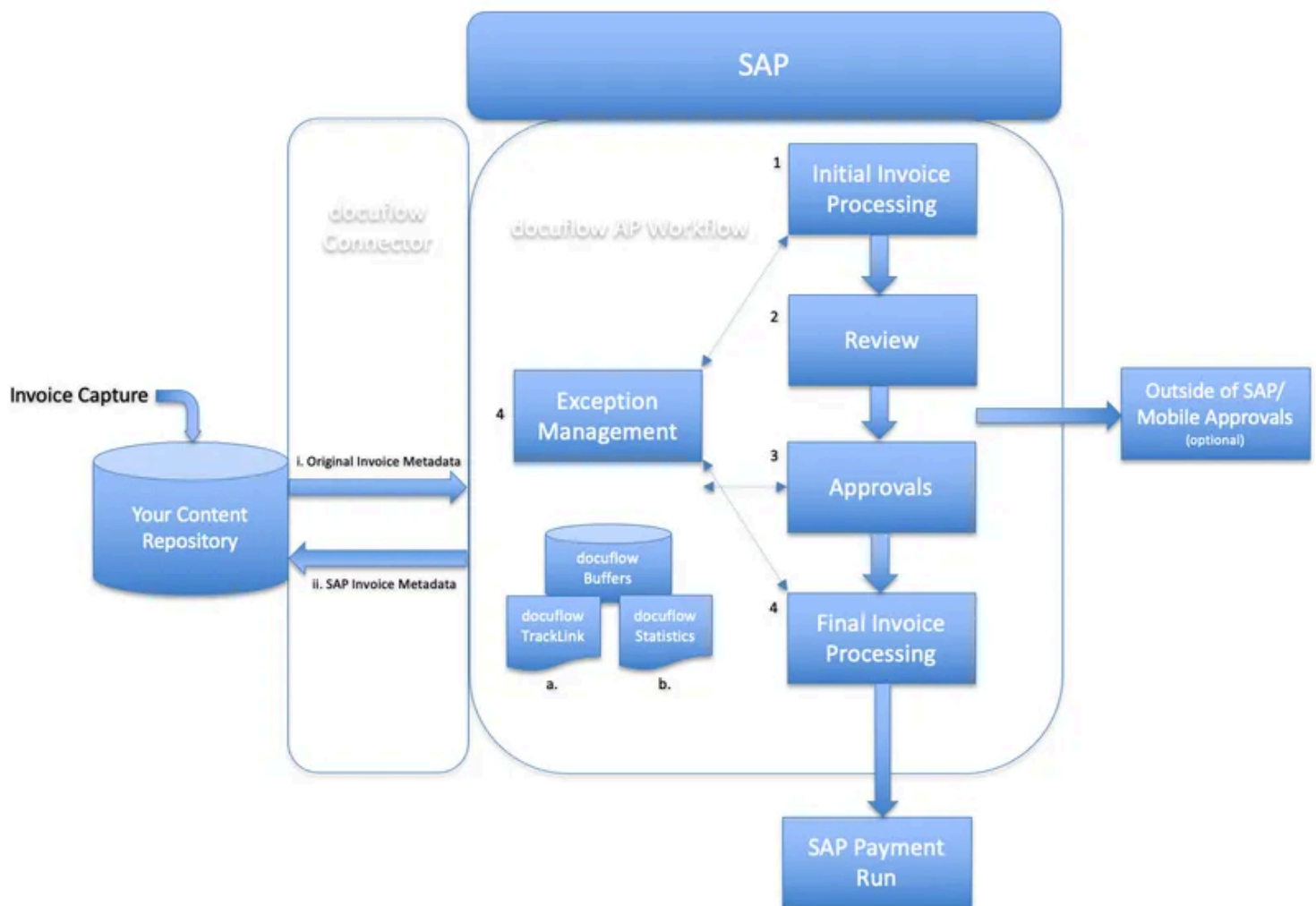
4.1.11.1. Functional Overview

Summary

The functional diagram below illustrates docuflow AP at a high level followed by a brief summary of each step.

For more detail on each step and their configuration and options, please refer to the additional sections in this document:

1. Configuration
2. Working with TrackLink
3. Customization



Steps

Invoice Capture

docuflow allows you to choose your method of capturing/scanning and ingesting your invoices. Once the invoice is in your content repository, docuflow is configured to recognize they are ready to be sent to docuflow in SAP to start the docuflow Accounts Payable Workflow.

i. Original Invoice Metadata

It is ideal that your Invoice Capture process is able to recognize original invoice metadata at the time of comital to your content repository. Any metadata captured can be sent to SAP to assist the docuflow Accounts Payable Workflow and drives Step 1 Initial Invoice Processing, drastically improving efficiencies in the overall processing of an invoice.

This step is done automatically in the background by docuflow.

1. Initial Invoice Processing

This is the first step of the docuflow Accounts Payable Workflow within SAP. This step can be configured for:

1. Pre-population of data to decrease data entry time and errors
2. Optional auto invoice creation
3. Exception Mangement
4. Optional auto exceptions
5. Dynamic Prioritization
6. Reviewer Selection (optional)
7. Approver Selection (optional)
8. Initial AP Inbox Queue Load Management and Agent Routing
9. Optional Coordinator stage

As mentioned above in the Invoice Capture Stage, any original invoice metadata that is available can be leveraged in the above configuration docuflow Accounts Payable Workflow options.

Users work from the docuflow TrackLink queue which can be filtered and sorted and customized to be the most effective for your AP processing team. In addition to processing the invoice work-item, it also has a host of additional functionality for note taking, raising exceptions, reprocessing, and much more.

In addition to flexible docuflow Accounts Payable Workflow agent / role resolution setup, AP processing team can also configure docuflow TrackLink to load balance work-items across a group.

It is in the Initial Invoice Processing stage that once an invoice is created (parked, held, posted), docuflow automatically creates a link to this invoice in your content management system.

2. Invoice Review

After Initial Invoice Processing is an optional Review stage. Reviewers are determined by the docuflow Core Agent table, or can selected manually by the AP Clerk if configured. If manually selected by the AP Clerk, there can be only 1 potential Reviewer chosen.

The review step can be configured to be skipped for PO based invoices as well as skip invoices up to a maximum amount.

Rejections (based on configurable rejection codes), note taking, raising exceptions, display original attachments as well as adding optional additional document to your invoice and content repository are all standard functionality. Rejections return the invoice to the Initial AP stage.

3. Invoice Approvals

After Initial Invoice Processing and optional Review, and after any exceptions are triaged, is the approval stage. docuflow Accounts Payable Workflow comes with up to 5 levels of approval. Approvers are determined by the docuflow Agent Approver table (with dollar value thresholds), or can selected manually by the AP Clerk or Reviewer. If manually selected by the AP Clerk or Reviewer, there can be only 1 potential approver chosen.

Approver determination is also fully customizable if you would like to implement your own role resolution.

The approval step can be configured to be skipped for PO based invoices as well as skip invoices up to a maximum amount.

Rejections (based on configurable rejection codes), note taking, raising exceptions, display original attachments as well as adding optional additional document to your invoice and content repository are all standard functionality. Rejections return the invoice to the Initial AP stage.

In addition, external / outside of SAP approvals are available based on docuflow's integration with your content repository or any non-SAP workflow runtime workflow engine, included mobile applications.

4. Final Invoice Processing

The last step in the docuflow Accounts Payable Workflow is final Invoice Processing.

This step can be configured for:

1. Automatic posting of invoices
2. Skipping of this step if invoice is already posted
3. Exception Management
4. Optional auto exceptions
5. Dynamic Prioritization
6. Final AP Inbox Queue Load Management and Agent Routing

This step is optional, and is often a different agent or group of agents than initial AP, as the last step in the workflow it is also the last chance to manipulate/check the invoice before posting and thus submitted to your standard SAP payment run.

5. Exception Management

Exception Management is a big part and strength of the docuflow Accounts Payable Workflow. Each dialogue step (ie: a work-item that is delivered to a user or group of users) has the option to "manually" raise an exception.

We say "manual", as each step whether it be a dialogue step or a background step (a step in the workflow that is behind the scenes/automatic), can also be configured for pre delivered auto-exceptions as well as custom auto-exceptions. Auto Exceptions are those that occur before and Initial AP Clerk, Reviewer, Approver, or Final AP Agent receive a work-item.

For exceptions of either variety, a work-item is generated to an agent/set of agents via configuration who in turn process / resolve the exception. When resolved, the work-item returns to the stage in which the exception originated.

Exceptions can also be configured to allow termination of the workflow (duplicates, etc) as well as customized to trigger a non-docuflow process, as well as optionally select approvers.

Note that you can also choose to configure or customize the auto-completion of an exception as well (regardless whether it was raised automatically or manually).

An example of raising an auto-exception would be for a PO based invoice in which the goods have not yet been received. The invoice (whether parked or not created) work-item can be automatically routed to an "Goods Not Received" exception.

An example of auto-completion of an exception would be in the same scenario - you may choose to have a nightly job run that determine whether any work-items in the "Goods Not Received" exception have now been fully received. If they have been received, the exception is automatically set to resolved and the work-item returns to the next stage in the workflow.

ii. SAP Invoice Metadata

You can optionally configure any SAP metadata to be sent "back" to your content management system throughout the lifecycle of the invoice in the docuflow Accounts Payable Workflow.

This sending of metadata from SAP to your content management system before and after each every stage in the workflow, ensuring your content management system (CMS) is in sync.

Some potential reasons to do this are:

1. Enable reporting in your CMS system
2. Enable retention in your CMS system
3. Enable workflows in your CMS system

This step is done automatically in the background by docuflow.

a. docuflow TrackLink

docuflow TrackLink is your home base for almost all elements related to the docuflow Accounts Payable Workflow.

Leveraging an SAP ALV "report" approach, the TrackLink toolset provides the following functionality:

1. Allows all agents (ie: Initial AP Agents, Reviewers, Approvers, Final AP Agents, and Exception Agents) to process their assigned work-items
2. Acts as shared Inbox for work-items with multiple agents (no SAP Office required)
3. Fully configurable sorting and filtering of customer defined layouts to allow for prioritization, queue management, and different "views" of invoices to be processed
4. Advanced note taking and viewing
5. Viewing of images from your content management system
6. A suite of Administrator utilities such as:
 1. forcing work-items to complete or terminated
 2. mass note taking
 3. year end rollover
 4. agent re-assignment
 5. and more.

For more detail please refer to the section "Working with TrackLink" located in this document.

DeadLine Monitoring

Each dialogue step of the docuflow Accounts Payable Workflow can be configured with deadline monitoring. Deadline monitoring is a feature of the SAP workflow runtime engine of which the docuflow Accounts Payable Workflow is built upon.

Each rule configured can be assigned a number of days in which a work-item must be processed within. If this does not occur, the work-item is automatically escalated. Escalation agents (or deadline agents) can be configured per rule or according to your needs such as based on your HR organizational structure.

Notifications

Each dialogue step of the docuflow Accounts Payable Workflow can be configured for basic and customized notifications.

Basic notifications send an email based on the agents SAP profile. This can be optionally customized to send notifications based on your preferred corporate communication platform.

4.1.11.2. Configuration

Summary

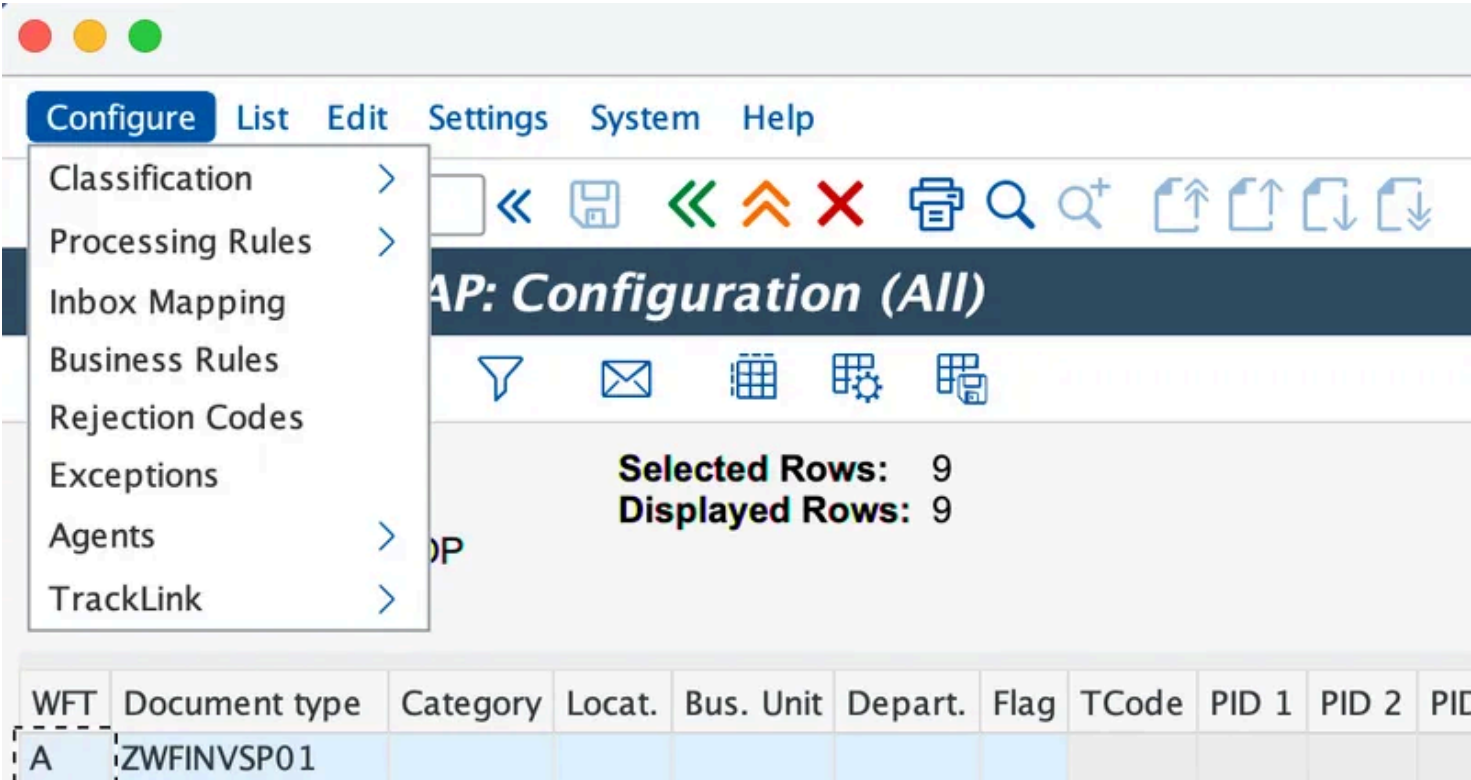
All docuflow AP configuration can accessed via two methods:

- 1. Directly via transaction `/dflow/wfconfig`
- 2. Via the docuflow System Dashboard (transaction `/dflow/sys`) and the menu path Configure→WFConnect→Workflow Configuration.

Please note: all docuflow AP configuration is NOT transprotable

When you first enter the config transaction, you will see a report showing the processing rule configuration.

Choose the menu path "Configure" in the top left of the screen to choose the components to configure.



Below are the categories of configuration for docuflow AP. Below is a brief summary. Refer to each categories section in this same document for complete details.

1. Classification

Locations, Invoice categories, business units, departments, and invoice flags help classify your invoices for processing and set how the below components can be configured.

2. Invoice Processing Rules

Based on the classification of an invoice, processing rules determine the baseline behaviour and flow the docuflow AP workflow.

3. Inbox Mapping

For invoices received via email, addresses can be mapping to invoices classification

4. Business Rules

Allows additional information to be configured and leveraged for auto-creating invoices.

5. Rejection Codes

Setup rejection codes for reviewers and approvers.

6. Exceptions

Define exceptions that suite your business. These can be raised automatically or via workflow agents.

7. Agents

Setup role resolution to determine standard agent assignment for all stages of the workflow broken down by the following:

- 7.a** Core: Initial AP, Reviewers, Exceptions, Final AP, Errors, Admins, Catch-Alls.
- 7.b.** Approvers: ASL based approval
- 7.c** Substitution and Delegation
- 7.d** Email Mapping - allows notifications to be send out

8. TrackLink

If custom fields are used in TrackLink, you may assign labels to them.

Note on Customization

Albeit docuflow AP can be implemented code-free with its flexible and configurable platform, you can tailor the workflow even further to suite your precise needs via customization. Customization is available throughout the docuflow AP Workflow in the form of customer BADI's. Refer to each section for customization options available relevant to that workflow stage, as well the section "BADI Customizing" for a view of all technical customizations available.

4.1.11.2.1. Classification

Overview

Classification of your invoices allows you to configure the rest of the docuflow AP Workflow accordingly. This include:

1. Rule processing options such as auto-create, auto-post, etc
2. Approval levels
3. Agent Determination
4. Agent Load Balancing
5. Applicable Manual and Auto Exceptions
6. Prioritization

Further more, classification of invoices provides you many more options when working within docuflow TrackLink for:

1. Inbox Work-item queue management
2. Load Balancing
3. Sorting
4. Filtering
5. Prioritization
6. Searching

In additional, for docuflow Statistics as well as your own, classification can provide further options for analysis.

Each tier of classification is optional, and you can also created additional custom tiers if you need.

Each tier can be logically associated with additional tiers to create a hierarchal structure, but this is optional as well.

Classification Sources

In order for classification to be functional, the values of the classification have to come from somewhere. These typically come from the below scenarios but again this is flexible to suite your AP process and business:

1. Metadata from the time of invoice capture / ingestion - direct values
2. Metadata from the time of invoice capture / ingestion - derived values based on business logic
3. Values within SAP based on business logic and metadata from 1. and 2. above
4. Values obtained only after the creation of an invoice and updated via a docuflow BADI.
5. BADI customization
6. Inbox Mapping - refer to the section Inbox Mapping in this Configuration document.

Values of classification are completely for your purpose only, and each tier can represent whatever you wish them to represent, with each tier being optional

Classification tiers that come with docuflow are:

Document Type

This represent the ArchiveLink Document Type determine by the configuration of docuflow DocManager which receives the inbound to SAP request from you content management system.

The document type (not accounting document type) is the highest overall classification of an invoice. Companies often just have 1 document type.

This is the minimum classification you must have for the and is already built into the docuflow system.

Note: Document type determine who can view images attached to invoices (outside of the docuflow AP workflow). If you require certain invoices to have different security requirements, consider implementing more document types.

Locations

Location typically represent geographical regions.

Invoice Categories

Invoice categories are the highest level of invoice specific classification. Some common examples are:

Commercial

Critical

Standard

Request for Funds

Confidential

Business Units

Business Units and Departments can also be used stand alone or with a relationship to Location, Invoice Categories, or each other.

Departments

Business Units and Departments can also be used stand alone or with a relationship to Location, Invoice Categories, or each other.

Invoice Flags

Invoice Flags tend to be very granular in nature and typically identify a very specific instance of an invoice that requires "special" processing along its lifecycle. Whether that be high priority processing from the AP team such as Early Pay Discounts to an Auto Exception such as Admin Kick-out.

Invoice Types

Invoice Types are pre-built in docuflow and consist of:

- PO
- NON-PO

docuflow determines this as to whether there is a PO number at time of capture/ingestion *or* based on which type of invoice is chosen to be created by the system or the Initial AP team.

Invoice Type is the minimum an invoice can be classified as.

Workflow Container Names

In the Workflow Trigger metadata mapping (or your customizing options), below are the corresponding workflow container names (must be upper case):

ZAR_OBJECT
ZLOCATION
ZINVOICECATEGORY
ZBUSINESSUNIT
ZDEPARTMENT
ZINVOICEFLAG
ZINVOICETYPE

BADI Customizing Options

The following BADI customizing options are available for Invoice Processing. For information on how to implement, please refer to the section "Customizing" later in this document.

1. **Custom Invoice Classification:** To manipulate the Classification for an invoice.

4.1.11.2.2. Invoice Processing Rules

Overview

Processing rules determine the baseline behaviour for an invoice based on its classification.

Albeit you can dynamically change the processing rule during the life cycle of an invoice via customizing (optional — refer to the Customizing section later in this document), the processing rule will set the default behaviour.

Setup

Based on the docuflow Classification configuration, below are the processing rule options. Unless specified otherwise, the options below are relevant to *both* NON-PO and PO Invoice Types.

*note that any changes to an invoice processing rule only takes affect *for net new or reprocessed* invoices.

1. **Enable Initial AP:** Enable the first step of the Accounts Payable workflow. In only special cases should this be off. Normal settings is ON.
2. **Enable Auto Create NON-PO:** If ON, docuflow will attempt to automatically park the invoice given metadata from the content management system for NON-PO invoices.
3. **Enable Auto Create PO:** If ON, docuflow will attempt to automatically park the invoice given metadata from the content management system for PO invoices.
4. **Enable PO Wizard:** If enabled, the AP Initial AP user will be presented with a popup asking whether the invoices is a NON-PO invoice vs. a PO invoice. If the invoices is a PO based invoice, and the metadata is captured and mapped correctly, the PO number will show up in the PO number box. If it is a NON-PO invoice, no value will be shown in the PO number box. This is basically a manual/visual confirmation whether the invoice is NON-PO vs. PO. If this is disabled, the system will 100% rely on the PO metadata value captured and mapped at the ingestion process.
5. **Enable Pre-Pop NON-PO/PO:** If Auto Create is not enabled, or is enabled and fails, the Initial AP team must first create the invoice (or send for exception). This option allows you to pre-populate the SAP invoice entry screen with metadata from the content management system if available.
6. **Enable Auto Send IA:**
 1. Auto Create successful: work-item will be sent to the next processing stage after Initial AP (ie: Initial AP will be skipped)
 2. Invoice "manually" created successfully (ie: Auto Create is not enabled or fails, or the Initial AP clerk parks the invoice successfully), the docuflow pop-up window can be skipped and the invoice can be sent directly for the next processing stage.
7. **Deadline (Days) Initial AP:** In days, how long a work-item will stay at the Initial AP stage before escalation. Who receives this escalation is configured in the Agents: Core configuration.
8. **Review Type:** this determines whether the Review step is enabled, and if enabled how Reviewers are determined.
 1. No Review: the Review step of the workflow will always be DISABLED.
 2. docuflow Core: Review step of the workflow is ENABLED and the agents are determined by the docuflow Core Agent table configuration.
 3. Manual Selection - Initial AP: Review step of the workflow is ENABLED and the agents are determined by manual selection from the InitialAP clerk.
9. **Skip Rev POs:** If enabled, the review step will be skipped for PO based invoices.
10. **Allow Skip Review:** If enabled, the Review step will be skipped if less than or equal to (LE) the amount specified in the field below Skip Rev Max. Amt.
11. **Skip Rev Max. Amt.:** If Allow Skip Review above is enabled, this is the value that determines the max amount an invoice will be skipped.

12. **Deadline (Days) Review:** In days, how long a work-item will stay at the Review stage before escalation. Who receives this escalation is configured in the Agents: Core configuration.
13. **Approval Type:** this determines whether the Approval step(s) are enabled, and if enabled how Approvers are determined.
 1. No Approvals: Approval step of the workflow will always be DISABLED.
 2. docuflow ASL: Approval step of the workflow will be enabled ENABLED and the agents and levels are determined by the docuflow Agent: ASL table configuration.
 3. Manual Selection - Initial AP: Approval step of the workflow is ENABLED and the agents are determined by manual selection from the InitialAP clerk.
 4. Manual Selection - Review: Approval step of the workflow is ENABLED and the agents are determined by manual selection from the Reviewer
 5. Manual Selection - Either: Approval step of the workflow is ENABLED and the agents are determined by manual selection from BOTH the InitialAP clerk and the Reviewer.
14. **Enable Skip Approvals for PO Invoices:** Skip all approval levels for PO based invoices.
15. **Deadline (Days) Approval Level 1 thru 5:** In days, how long a work-item will stay at particular Approval stage before escalation. Who receives this escalation is configured in the Agents configuration (outlined later in this document).
16. **Enable AutoPost NON-PO/PO:** docuflow can attempt to Automatically post the invoice at the Final AP stage. This is only attempted if "Enable Final AP" is enabled above. If Auto Post fails, a work-item is automatically sent to a Final AP agent.
 1. En. FA AP NON-PO: Enable auto post for NON-PO invoices
 2. En. FA AP PO: Enable auto post for PO invoices.
17. **En. FA NON-PO:** Enable Final AP for NON-PO invoice's
18. **En. FA PO:** Enable Final AP for PO invoices.
19. **Enable Auto Send FA:**
 1. Auto Post successful: workflow will be set to complete (ie: Final AP will be skipped).
 2. Invoice "manually" posted successfully (ie: Auto Post is not enabled or fails, or the Final AP clerk posts the invoice successfully), the docuflow pop-up window can be skipped and the workflow will be set to complete automatically.
20. **Force Post:** If Final AP is enabled, and in the case that a work-item is sent to an agent (ie: AutoPost is enabled and either fails or AutoKeep is enable / Skip Final AP is disabled) if the invoice is *not* posted, then the "Complete Workflow" on the docuflow pop-up window will be disabled.
21. **Deadline (Days) Final AP:** In days, how long a work-item will stay at the Final AP stage before escalation. Who receives this escalation is configured in the Agents configuration (outlined later in this document).
22. **WF ID Sync:** if enabled, the metadata property for the workflow ID will be sent from SAP to your CMS repository. The CMS property name is specified below.
23. **WFID MData Name:** The CMS metadata property name for workflow ID.

BADI Customizing Options

The following BADI customizing options are available for Invoice Processing. For information on how to implement, please refer to the section "Customizing" later in this document.

1. **Custom Rule Determination:** To manipulate the Processing Rule for an invoice.
2. **Custom Auto Create:** To implement your own Auto Create logic.
3. **Custom Pre-Pop:** To implement your own Pre-Pop logic.
4. **Custom Work-item Text:** To implement your own Work-item text.
5. **Custom Auto Exceptions:** To implement your Auto Exceptions.

- 6. **Custom Auto Post Logic:** To implement your own Auto Post logic.
- 7. **Custom Approval ASL Check:** To implement your own ASL check
- 8. **Custom Role Resolution / Agent Determination:** To implement your own logic to determine work-item recipients
- 9. **Custom Notifications:** To implement your own method of notifying agents.
- 10. **Custom Pre-Processor Logic:** Powerful BADI to take control of all pre-processing logic
- 11. **Custom Post-Processor Logic:** Powerful BADI to take control of all post-processing logic
- 12. **Custom POPUP PBO Logic:** To implement your own changes to the docuflow pop-up windows for processing.
- 13. **Custom POPUP PAI Logic:** To implement your own logic to the docuflow pop-up windows for processing.

4.1.11.2.3. Inbox Mapping

Inbox Mapping is one of the options to drive Classification (please reference the section Classification located in this document).

This configuration maps:

Recipient Email Address ↔ The classification hierarchy.

In order to implement this, the BADI /DFLOW/IF_APWF~CLASSIFICATION must be leveraged.

Read table /DFLOW/ZAPWFC16 and fill the appropriate cs_zapwfl01 fields for classification:

ZAR_OBJECT (this is document type)

ZLOCATION

ZCATEGORY

ZBUSINESSUNIT

ZDEPARTMENT

ZFLAG

4.1.11.2.4. Business Rules

Business rules provide a structure to store additional to be leveraged in the Accounts Payable process.

This table is not automatically used by the product, you must implement a docuflow BADI to use it as you require.

A common application for this is to house the details needed to auto-park invoices. Each customer has slightly different data entry screens and thus the BADI DFLOW/APWF_BADI~AUTOCREATE

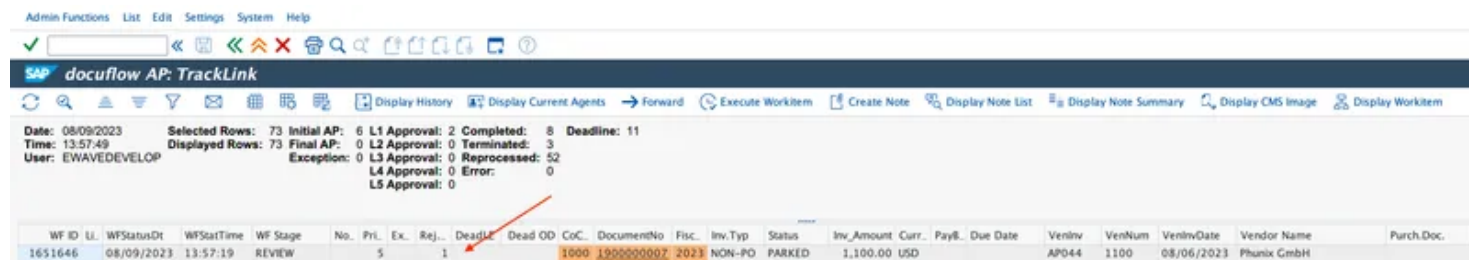
4.11.2.5. Rejection Codes

Overview

Rejection codes are a quick indicator of why a rejection has occurred at either the Review or Approval stages.

Notes/comments can also be made for further detail, but rejection codes provide a quick reason why an invoice is rejected.

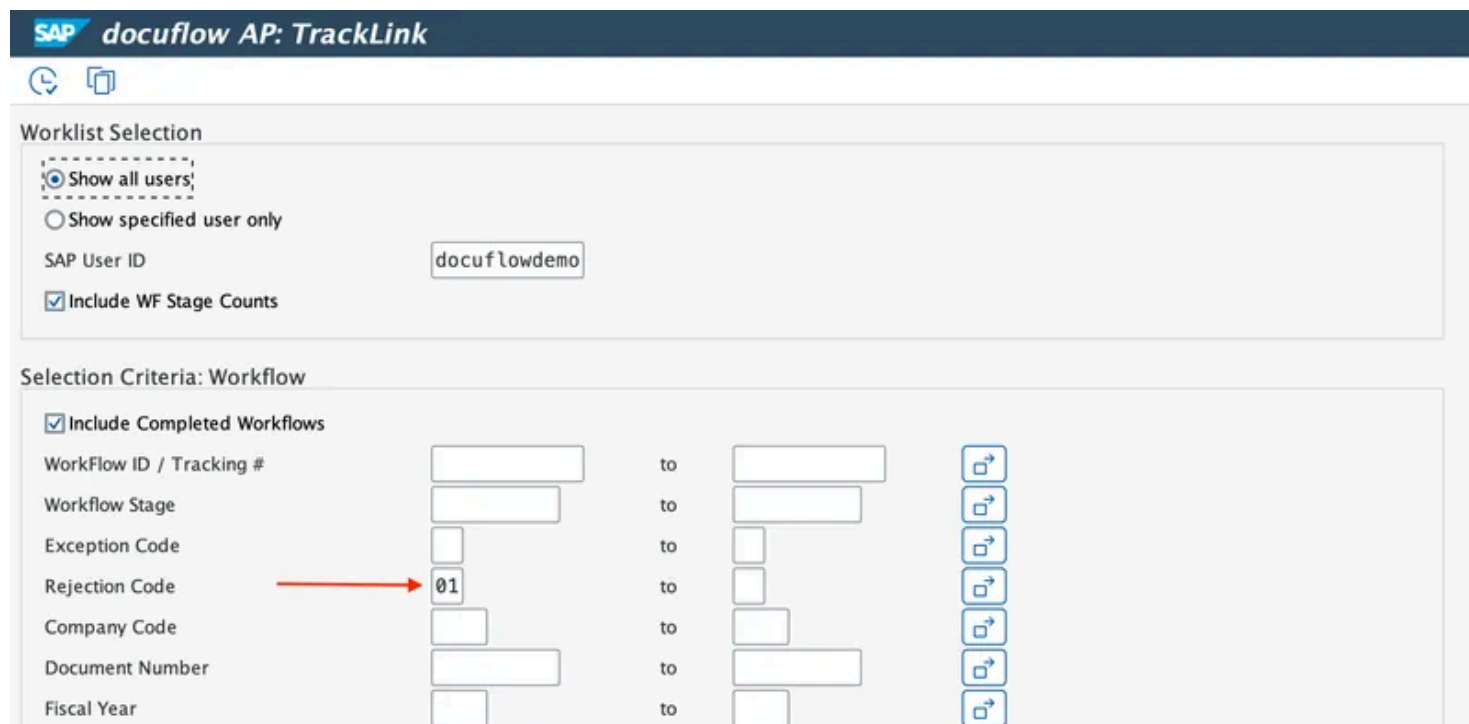
When a workitem is rejected, the rejection code shows up in TrackLink until the invoice is processed further, at which point the rejection code is automatically removed.



The screenshot shows the SAP docuflow AP: TrackLink interface. At the top, there's a menu bar with 'Admin Functions', 'List', 'Edit', 'Settings', 'System', and 'Help'. Below it is a toolbar with various icons. The main area displays a summary of the workflow status, including 'Date: 08/09/2023', 'Time: 13:57:49', 'User: EWAVEDEVELOP', and various counts for 'Selected Rows', 'Displayed Rows', 'Initial AP', 'Final AP', 'L1 Approval', 'L2 Approval', 'L3 Approval', 'L4 Approval', 'L5 Approval', 'Completed', 'Terminated', 'Reprocessed', and 'Error'. Below this is a table with columns: 'WF ID', 'LI', 'WFStatusDt', 'WFStatTime', 'WF Stage', 'No.', 'Pri.', 'Ex.', 'Rej.', 'DeadLc', 'Dead OD', 'CoC', 'DocumentNo', 'Fisc.', 'Inv.Typ', 'Status', 'Inv.Amount', 'Curr.', 'PayB.', 'Due Date', 'VenInv', 'VenNum', 'VenInvDate', 'Vendor Name', and 'Purch.Doc.'. A red arrow points to the 'Rej.' column, which contains the value '01' for the first row.

WF ID	LI	WFStatusDt	WFStatTime	WF Stage	No.	Pri.	Ex.	Rej.	DeadLc	Dead OD	CoC	DocumentNo	Fisc.	Inv.Typ	Status	Inv.Amount	Curr.	PayB.	Due Date	VenInv	VenNum	VenInvDate	Vendor Name	Purch.Doc.
1651646		08/09/2023	13:57:19	REVIEW	5			1			1000	19000000007	2023	NON-PO	PARKED	1,100.00	USD			AP044	1100	08/06/2023	Phunix GmbH	

Rejection codes are also an optional input parameter to TrackLink, allowing you to easily search for all invoices in a rejection status:



The screenshot shows the 'Worklist Selection' and 'Selection Criteria: Workflow' sections of the SAP docuflow AP: TrackLink interface. In the 'Worklist Selection' section, the 'Show all users' radio button is selected, and the 'SAP User ID' field contains 'docuflowdemo'. The 'Include WF Stage Counts' checkbox is checked. In the 'Selection Criteria: Workflow' section, the 'Include Completed Workflows' checkbox is checked. The 'Rejection Code' field is highlighted with a red arrow and contains the value '01'. The 'Workflow ID / Tracking #' field is empty. The 'Workflow Stage' field is empty. The 'Exception Code' field is empty. The 'Company Code' field is empty. The 'Document Number' field is empty. The 'Fiscal Year' field is empty. The 'to' field is empty. The 'Workflow ID / Tracking #' field is empty. The 'Workflow Stage' field is empty. The 'Exception Code' field is empty. The 'Company Code' field is empty. The 'Document Number' field is empty. The 'Fiscal Year' field is empty. The 'to' field is empty.

Note: Rejection codes are language specific. Each one MUST have the language key identified.

4.1.11.2.6. Exceptions

Overview

Exception management is a very powerful and flexible component of the docuflow Accounts Payable Workflow.

Exceptions take two forms with regard to how they are raised:

1. Manually by the agent of a work-item
2. Automatically by either docuflow pre-configured Auto Exceptions or by customization via customer BADI's

Exceptions are in turn resolved via two methods:

1. Manually by the agent of a work-item
2. Automatically by either docuflow pre-configured Auto Exceptions or by customization via customer BADI's

Exceptions can also be configured to call custom logic to suite unique customer requirements.

Setup

Below are the options for configuring Exceptions:

1. **Exception ID:** The exception ID is a 2-digit code that you create. You can use numbers such as 01-99 or letters. Typical use is 01 thru 99 with a zero padding for 01 thru 09.
2. **Language:** Exception codes are language specific. Each one MUST have the language key identified.
3. **Workflow Stage:** Exceptions are configured per "workflow stage" levels.
There are three levels to the docuflow Accounts Payable Workflow:
 - a. Initial AP
 - b. Approval
 - c. Final AP
 If an exception is relevant to all three stages you can leave that field flag empty and it will be available in all three levels.
4. **Exception Code Text:** This is a free form description of the exception.
5. **Allow WF Termination:** Enabling this option provides the exception agent the option to terminate the entire workflow for this invoice. If this is disabled that pushbutton is greyed out.
6. **Force Resolve:** Enabling this option forces the originator of the exception to acknowledge that the exception has been resolved before processing further.
7. **Deadline (Days):** In days, how long a work-item will stay at the Exception stage before escalation. Who receives this escalation is configured in the Agents configuration (outlined later in this document).

Role Resolution / Agent Determination

Who will receive dialogue (foreground) work-items for Exceptions is configured in the Agent configuration. Refer to the section "Agents" later in this document.

BADI Customizing Options

The following BADI customizing options are available for Exception Management. For information on how to implement, please refer to the section "Customizing" later in this document.

1. **Custom Workitem Text:** To implement your own Workitem text.
2. **Custom Role Resolution / Agent Determination:** To implement your own logic to determine work-item recipients

- 3. **Custom Notifications:** To implement your own method of notifying agents.
- 4. **Custom Pre-Processor Logic:** Powerful BADI to take control of all pre-processing logic
- 5. **Custom Post-Processor Logic:** Powerful BADI to take control of all post-processing logic
- 6. **Custom POPUP PBO Logic:** To implement your own changes to the docuflow pop-up windows for processing.
- 7. **Custom POPUP PAI Logic:** To implement your own logic to the docuflow pop-up windows for processing.

4.1.11.2.7. Agents: Core

Overview

Core Agents are based on the docuflow Classification configuration. Approvers are separate from core agents. There are 4 categories of agents associated with docuflow "core" processing:

1. Initial AP Agents
2. Review Agents
 1. Note: Review Agents can also be configured according to ASL Thresholds (such as approvers). Please refer to the next section Agents: Approvers for more information.
3. Final AP Agents
4. Exception Agents

Each one of those agent categories can also have assigned deadline and catch-all agent.

You can specify as many agents as you would like to 1 processing rule, or you can also choose org units that reference multiple agents. This creates virtual "teams" of processors per invoice classification (called "possible agents").

docuflow AP and the underlying SAP workflow runtime system manages the sharing of these workitems and prevents work-items from being processed more than once. `Once a user has "touched" a work-item, it is reserved by them for processing until it is either completed or resubmitted back the larger "team".`

Setup

Below are the options for configuring Agents:

1. **Classification:** standard classification hierarchy
2. **WF Stage Core:** which stage of the workflow is this rule relevant to (this is mandatory unless you are configuring a "Admin" or "Deadline" agent)
3. **Invoice Type:** Allows you to split out agents between PO vs. NON-PO
4. **Exception Code:** For exception stage, specifies a particular exception.
5. **Admin:** `There must be at least 1 Admin agent in your docuflow implementation.` These are agents who can configure the system and access via TrackLink, and also who can perform Administrator functions via TrackLink.
6. **Deadline:** Indicates this agent is a deadline agent for a particular stage, approval amount, or exception.
7. **Catch-All:** `There must be at least 1 catch-all agent in your docuflow implementation.` These are agents which will receive a work-item if no agent is found. *For Catch-All agents, classification is ignored (only the flag that the agent is a catch-all is utilized).*
8. **Workflow Agent Type:** Choose the applicable agent type.
9. **Workflow Agent:** Specify the appropriate agent value based on the workflow type specified above.

BADI Customizing Options

The following BADI customizing options are available for Agent determination. For information on how to implement, please refer to the section "Customizing" later in this document.

1. **Custom Role Resolution / Agent Determination:** To implement your own exception processing logic in place/in addition to a docuflow AP Workflow Exception work-item.
2. **Custom Notifications:** To implement your own method of notifying agents.

4.11.2.8. Agents: Approvers (ASL)

When you first enter the configuration screen for approvers, a report is displayed showing your current matrix. To change configuration, choose the "Configure" tool-button:

Document type	Locat.	Categ.	Bus. Unit	Depart.	Flag	CoCode	Vendor	Cost Ctr	LineNum	Skip Appr.	SkipAprAmt	L1 Min.Amt	L1 Max.Amt	Level 1 Approver	L2 Min.Amt	L2 Max.Amt	Level 2 Approver
ZWFINVSP01			ACME_SOUTH	ADMIN				9030	1	X	999.99	1,000.00	1,999.00	WFAPPROVERL1	2,000.00	2,999.00	WFAPPROVERL2
ZWFINVSP01			ACME_SOUTH	ADMIN				9030	2	X	999.99	1,000.00	1,999.00	WFAPPROVERL1	2,000.00	2,999.00	WFAPPROVERL2

Approval Agents are based on the docuflow Classification configuration hierarchy + three additional (optional) elements:

1. company code
2. vendor number
3. cost center

This means you can use any combination of those before docuflow will move up the classification hierarchy:

docuflow will use the following order to find a relevant approver, and will continue up the hierarchy combining the agents for a particular approver level:

1. Classification hierarchy match + company code + vendor + cost center
2. Classification hierarchy match + company code + vendor (*cost center = space*)
3. Classification hierarchy match + vendor (*company code/cost center = space*)
4. Classification hierarchy match + company code + cost center (*vendor = space*)
5. Classification hierarchy match + cost center (*company code/cost center = space*)

....and then will continue up the classification hierarchy. A classification hierarchy match is any combination of the 6 classification elements, with a space allowed to create your combinations.

1. Document Type + Category + Location + Business Unit + Department + Flag + Company Code
2. Document Type + Category + Location + Business Unit + Department + Flag
3. Document Type + Category + Location + Business Unit + Department (*Flag = space*)
4. Document Type + Category + Location + Business Unit
5. Document Type + Category + Location
6. Document Type + Category
7. Document Type
8. Global rule (everything blank)

In each of the above scenarios, you must leave the field blank for it to traverse up the hierarchy, and leaving blanks will allow you to move up the hierarchy.

Each rule found will be appended. At the end of agent determination, duplicate agents are automatically removed to a single of that agent.

If you do not have an invoice with 1 or more classification elements they are considered an empty space.

example: If you do not have a Category for an invoice, you can leave Category a space and still use Location + Business Unit - this rule will be satisfied:

Document Type + Category + Location + Business Unit

Based on those elements, up to 5 levels can be configured.

You may also choose to skip approval for certain invoices that match the elements above, and optionally with a max dollar amount.

example: skip approvals for a particular location and business if the invoice amount is lower than \$1,000.00

Below is an example of skipping approvals up to and equal to \$999.99. From there:

Level 1 approval is amount greater than or equal to \$1,000.00 but less than or equal to \$1,999.99, going up by thousand dollar increments, with Level 5 approval being for any invoice greater than or equal to \$5,000.00

Document type	ZWFINVSP01
Location	
Category	
Business Unit	ACME_SOUTH
Department	ADMIN
Flag	
Company Code	
Vendor	
Cost Center	9030
Line Number	1

Agents – ASL Approval

☒ Skip Approval

Skip Apr Max. Amt.	999.99
L1 Minimum Amt.	1,000.00
L1 Maximum Amt.	1,999.00
L1 Approver	WFAPPROVERL1
L2 Minimum Amt.	2,000.00
L2 Maximum Amt.	2,999.00
L2 Approver	WFAPPROVERL2
L3 Minimum Amt.	3,000.00
L3 Maximum Amt.	3,999.00
L3 Approver	WFAPPROVERL3
L4 Minimum Amt.	4,000.00
L4 Maximum Amt.	4,999.00
L4 Approver	WFAPPROVERL4

L4 Approver	WFAPPROVERL4
L5 Minimum Amt.	5,000.00
L5 Maximum Amt.	999,999,999.00
L5 Approver	WFAPPROVERL5

If you have the need for more than 1 *possible agent* for a level, duplicate the entry and use the "Line Number" field to make the new entry unique. You can have unlimited *possible agents* for an invoice.

To have more than 1 approval, you can change your dollar amounts to require multiple levels:

In this example, an invoice would be relevant always to 2 levels of approval because Level 5 is set to all amounts above the skip level:

Document type	ZWFINVSP01 
Location	
Category	
Business Unit	ACME_SOUTH
Department	ADMIN
Flag	
Company Code	
Vendor	
Cost Center	9030
Line Number	2

Agents – ASL Approval

☒ Skip Approval

Skip Apr Max. Amt.	999.99
L1 Minimum Amt.	1,000.00
L1 Maximum Amt.	1,999.99
L1 Approver	WFAPPROVERL1
L2 Minimum Amt.	2,000.00
L2 Maximum Amt.	2,999.99
L2 Approver	WFAPPROVERL2
L3 Minimum Amt.	3,000.00

L3 Maximum Amt.	3,999.99
L3 Approver	WFAPPROVERL3
L4 Minimum Amt.	4,000.00
L4 Maximum Amt.	4,999.99
L4 Approver	WFAPPROVERL4
L5 Minimum Amt.	1,000.00
L5 Maximum Amt.	999999,999.00
L5 Approver	WFAPPROVERL5

BADI Customizing Options

The following BADI customizing options are available for Agent determination. For information on how to implement, please refer to the section "Customizing" later in this document.

1. **Custom Role Resolution / Agent Determination:** To implement your own exception processing logic in place/in addition to a docuflow AP Workflow Exception work-item.
2. **Custom Notifications:** To implement your own method of notifying agents.

4.1.11.2.9. Agents: Substitutions and Delegations

4.1.11.2.10. Agents: Email Mapping

4.1.11.3. Working with TrackLink

Overview

docuflow TrackLink is considered home-base for all users and administrators. TrackLink is used as the end users processing queue as well as a host of additional functionality for inbox management, searching and tracking, and administrator functions.

Below is a list of functionality offered:

1. User inbox processing
2. Load balancing across processing group
3. Dynamic filtering and sorting
4. Wide range of search options
5. Configurable layout at the global or user level
6. Customizable layout content
7. Drill-down functionality to invoices, purchase orders, etc.
8. Integration with your SAP payment run status
9. Enhanced Work-Item Processing function:
 1. Security based forwarding (optional)
 2. Displaying image from content repository
 3. Enhanced "note" taking and viewing
 4. Exception processing
 5. Reprocessing
10. Administrator functions:
 1. Single and mass Reassignment
 2. Single and mass forced completion and buffer status management
 3. Single and mass enhanced "note" taking
 4. Single and mass Role Resolution update
 5. Full invoice year-end rollover toolset with link management

Searching

The ability to search and find a work-item quickly for customer support or other reasons is essential.

docuflow TrackLink and its underlying structure means the data is at your fingertips without any data compilation that would result in long wait times. A wide variety of search options are available in standard SAP format (ie: select-options).

Results are displayed in a straightforward layout and then be further filtered and sorted.

Search results are independent of whether the user searching is able to process the work-item or not (they are of course prevented from processing if they are not a "possible agent").

You can choose to display in-progress invoices only or included completed workitems.

User Inbox Processing and Functions

User inbox processing is the day to day work queue of your AP team and Exception managers. Albeit TrackLink is also used by approvers when outside-of-sap or mobile approval is not configured, TrackLinks flexibility is realized when leveraged as home-base for end-users main job is to process work-items.

Working in tandem with configured role-resolution to determine "possible agents," docuflow AP and the underlying SAP workflow runtime system manages the sharing of these work-items and prevents work-items from being processed more than once. Once a user has "touched" a work-item, it is reserved by them for processing until it is either completed or resubmitted back to the larger "team".

Therefore, at the highest level, a user can view any list of work-items (which is basically equivalent to Searching), but they can also choose to view work-items that they are a "possible agent" for / are allowed to process.

A user can further filter and sort by configurable and saveable layouts to provide different "views" of their queue.

Functions are categorized into three buckets:

1. Toolbar functions - functions everyone has access too
2. User functions - "special" user functions
3. Administrator functions

Toolbar Functions

1. **Execute Work-item:** The most common function, this allows the user to execute the work-item of which they are a possible agent of. What happens upon execution depends on the workflow stage and configured functionality (ie: create invoices, display invoice, approve invoice, triage an exception, etc).
2. **Display History:** Every detail about an invoice is logged extensively and is viewable for light reading enjoyment as well as analysis into the history of an invoice and its processing.
3. **Forwarding:** If *and only if* a user has the appropriate authorization, forwarding a work-item outside of the possible agent group is available. Enabling forwarding is completely optional and is typically driven by a customers internal process. A user must have the below authorization to forward a workitem:
 1. AUTHORITY-CHECK OBJECT 'S_WF_WI'
 - ID 'WI_TYPE' FIELD '*
 - ID 'TASK_CLASS' FIELD '*
 - ID 'WFACTVT' FIELD '35'.
 2. OR AUTHORITY-CHECK OBJECT 'S_WF_WI'
 - ID 'WI_TYPE' FIELD '*
 - ID 'TASK_CLASS' FIELD '*
 - ID 'WFACTVT' FIELD '25'.
4. **Create Note:** Allows you to notes to a work-item.
5. **Display Note List:** Display a list of all notes for a work-item. You can then view the note itself by drilling down into that note.
6. **Display Note Summary:** Displays a summary of all notes so you do not have to search through all the notes if there are a number of them.
7. **Display Work-item:** Displays the SAP workflow runtime system work-item.
8. **Display CMS Image:** Displays the original invoice from your content management system.
9. **Drill-down:** If the SAP invoice has been created, you can drill down to display without processing the work-item. You can also drill down into the purchase order.

User Functions

1. **User Functions → Re-process Invoice:** In some scenarios, you may choose to restart the workflow. This is called "re-processing" an invoice. If the workflow already has an SAP invoice created, you can choose to retart using the same invoice or create a new one.

2. **User Functions → Re-classify Invoice:** In some scenarios, you may choose to re-classify the invoice (ie: change location, business unit, department, category, or flag). This does not re-process an invoice, and in most cases, a re-process is not required. Only Initial AP, Review, and Exception stages can potentially re-classify an invoice.

Administrator Functions

If configured as a docuflow Administrator, the below functions are available. These functions can be processed for single work-items as well as mass processing:

1. **Lock/Unlock:** Administrators can lock workitems so they cannot be processed until unlocked. This could be when a batch of invoices is under investigation, or if there is a system issue.
2. **Terminate:** A workflow can be terminated/stopped immediately.
3. **Re-Assign:** Individual work-items can re-assigned to a different invoice.
4. **Display ASL Log:** All ASL checks that are performed during approval are logged. This displays such information.
5. **Mass Functions:** These functions perform activity on all workitems shown on the TrackLink screen/list:
 1. **Lock/Unlock:** Same as above but en mass
 2. **Create Note:** Above what is available at the end-user level, Administrators can add mass notes to the list of work-items displayed.
 3. **Forward:** Above what is available at the end-user level, Administrators can mass forward the list of work-items displayed. A user must have the below authorization to forward a workitem:
 1. AUTHORITY-CHECK OBJECT 'S_WF_WI'
 - ID 'WI_TYPE' FIELD '*'
 - ID 'TASK_CLASS' FIELD '*'
 - ID 'WFACTVT' FIELD '35'.
 2. OR AUTHORITY-CHECK OBJECT 'S_WF_WI'
 - ID 'WI_TYPE' FIELD '*'
 - ID 'TASK_CLASS' FIELD '*'
 - ID 'WFACTVT' FIELD '25'.
 4. **Reprocess:** Same as above but en mass
 5. **Terminate:** Same as above but en mass
 6. **Display ASL Log:** Same as above but en mass
 7. **Year End Rollover:** SAP document number contain the fiscal year. For in-process work-items with SAP invoices that are not posted and year-end occurs, the year end process changes the key for the SAP invoice (including the SAP document number itself depending on your SAP configuration). The docuflow Year End Rollover Utility automatically:
 1. Re-creates the invoice in the new year
 2. Updates the docuflow AP database
 3. Re-establishes any linkages to your content management system attachments
 4. Can optionally set the previous years invoice as marked for deletion
 8. **Force WF Update:** Allows the forced update of TrackLink invoices fields
 9. **Force Agent Update:** If agent configuration is changed, this allows the update of the role-resolution logic to update the appropriate agents/possible agents.

Versions of TrackLink

There are two versions of TrackLink available:

TrackLink Full

Transaction: /DFLOW/APTRACKLINK

This has all functionality available and the initial screen allows you to choose to see all workitems regardless of whether you are a possible agent of that workflow. Completed workflows are also viewable and searchable.

*Note that you can only process a workitem if you are an assigned possible agent

Best suited for:

1. Managers and supervisors
2. Any user that will perform "administrator functions" listed above
3. Any user that will perform "user functions" listed above
4. Users that need to search all invoices regardless of who their possible agent(s) are, including historical invoices

TrackLink User Only

Transaction: /DFLOW/APTRACKLINKUS

Provides access to:

1. Only In-flight (ie: *not* completed) workflows
2. Only Workitems of which you are a possible agent for
3. Does not allow and user or administrator function menu paths (still has access to toolbar functions)

Best suited for:

1. Users that only need an inbox / processing queue

4.1.11.4. Customizing

Albeit docuflow AP can be implemented code-free with its flexible and configurable platform, you can tailor the workflow even further to suite your precise needs via customization.

All docuflow customization is accomplished via provided docuflow BADI's. Each BADI's interface is described more in the docuflow Developers Guide.

Below is a list of all docuflow AP Workflow BADI's:

ENHANCEMENT SPOT: /DFLOW/APWF

Initialization - Classification

BADI: /DFLOW/APWF_BADI~CLASSIFICATION

Initialization of the workflow container occurs once per workflow at the start. This BADI allows you to modify the Classification of an invoice entry prior identification of a process rule (see below).

Initialization - Processing Rule

BADI: /DFLOW/APWF_BADI~PROCESSING_RULE

Initialization of the workflow container occurs once per workflow at the start. This BADI allows you to modify the Processing Rule for an invoice.

Workitem Text

BADI: /DFLOW/APWF_BADI~WORKITEMTEXT

docuflow comes with default workitem text:

docuflow AP: <workflow stage> <workflow ID>

This BADI allows you to modify or create your own.

Auto Create

BADI: /DFLOW/APWF_BADI~AUTOCREATE

docuflow comes with built in Auto Create logic for both NON-PO and PO based invoices. This BADI allows you to modify or create your own logic.

Pre-Population

BADI: /DFLOW/APWF_BADI~PREPOPULATION

docuflow comes with built in Pre-Population logic (in the form of BDC routines) for both NON-PO and PO based invoices. This BADI allows you to modify or create your own logic.

Auto Post

BADI: /DFLOW/APWF_BADI~AUTOPOST

docuflow comes with built in Auto Post logic for both NON-PO and PO based invoices. This BADI allows you to modify or create your own logic.

Approver ASL Check

BADI: /DFLOW/APWF_BADI~ASLCHECK

docuflow comes with built in ASL logic against its own docuflow tables. This BADI allows you to modify or create your own ASL logic.

Role Resolution / Agent Determination

BADI: /DFLOW/APWF_BADI~ROLERESOLUTION

docuflow comes with built in Role Resolution logic against its own docuflow tables. This BADI allows you to modify or create your own ASL logic.

Notifications

BADI: /DFLOW/APWF_BADI~NOTIFICATIONS

docuflow comes with built in Notifications logic against a work-item recipient using their SAP profile. This BADI allows you to modify or create your own ASL logic.

TrackLink

BADI: /DFLOW/APWF_BADI~TRACKLINK_DATA

docuflow comes with built in TrackLink queue monitoring, searching, and reporting. This BADI allows you to modify what data is being constantly updated and tracked by docuflow.

TrackLink ALV

BADI: /DFLOW/APWF_BADI~TRACKLINK_ALV

docuflow comes with built a TrackLnk reporting ALV. This BADI allows you to modify columns and data displayed.

Statistics

BADI: /DFLOW/APWF_BADI~STATISTICS_DATA

docuflow comes with built in Statistics gathering and reporting. This BADI allows you to modify or create your own analysis logic.

Statistics ALV

BADI: /DFLOW/APWF_BADI~STATISTICS_ALV

docuflow comes with built a Statistics reporting ALV. This BADI allows you to modify columns and data displayed.

Pre-Processor

BADI: /DFLOW/APWF_BADI~PREPROCESSOR

This BADI allows you to modify the workflow in-progress buffer and rule processing entry *prior* to each step of the docuflow AP workflow, for both foreground/dialogue work-items (Initial AP, Approvals, Final AP, Exceptions) as well as background steps (such as Auto Create, Auto Post).

CAUTION: This BADI allows you to take complete control of the pre-process aside from possible agents. Use with caution.

Post-Processor

BADI: /DFLOW/APWF_BADI~POSTPROCESSOR

This BADI allows you to modify the workflow in-progress buffer and rule processing entry *after* to each step of the docuflow AP workflow. This typically determines the "next stage" in processing.

CAUTION: This BADI allows you to take complete control of the post-process. Use with caution.

Pop-Up PBO

BADI: /DFLOW/APWF_BADI~POPUP_PBO

This BADI allows you to modify the docuflow processing popup screen for Initial AP, Exception Manager, Approvals, and Final AP. You can enable and disable button, as well as change text.

Pop-Up PAI

BADI: /DFLOW/APWF_BADI~POPUP_PAI

This BADI allows you to modify the docuflow processing popup for logic Initial AP, Exception Manager, Approvals, and Final AP based on which pushbutton is selected.

4.1.12. Publisher

Publisher provides the ability to store your message output (such as SAPScript, SmartForms, AdobeForms, as well as any PDF, OTF, or Spool output) from SAP to your CMS system using docuflow ECM (an alternative to docuflow Content Server).

Typically a background process, your SAP generated content is stored, optionally linked to your SAP record, and metadata can be transferred from SAP to your CMS system. This may be implemented using various methods depending on your SAP module such as but not limited to: message output, spool generation, workflow event, business transaction events, as well as customer enhancements spots.

When utilizing Publisher for a particular message output you will need to add a call to a docuflow class-method at the end of your message output (program/function module). If the message output is the delivered message output (program/function module) from SAP and has not be customized (sometimes referred to as a "Z" version) you will need to create an implicit enhancement point to include the docuflow code. Creating the implicit enhancement point is outside the scope of this document.

Calling the docuflow Publisher code snippet

In your custom code or enhancements, add code that calls the "publish_message" method of the "/dflow/cl_publishing".

Please refer to the docuflow Development Guide for more info and an example.

4.1.13. System Messages

System messages prevent or inform/warn users and API usage when needed during downtime or critical changes across the docuflow platform.

Examples of this may include:

1. When there is an issue storing documents to the backend content repository and you wish to inform the users to take a chill pill rather than create an abundance of help desk tickets
2. You are in the middle of making a change to production that requires a temporary hands off
3. The content management system is undergoing maintenance and a particular interface that uses the docuflow SCRUD API needs to stop processing temporarily

There are two types of system messages (see below for the "Message Type"):

"E" - stops processing and informs the user/process

"I" - informs the user/process but continues processing

For user based docuflow processing, such as Store New and Assing (ie: GOS menu based document storage and linking), the message (regardless of type I or E) will be displayed to the end user and will also will be added to the return table in the docuflow API gateway.

For API or background based docuflow processing, such as SCRUD usage, the message (regardless of type I or E) will be added to the return table in the docuflow API gateway.

To create a System Message, enter transaction /dflow/docuflowsys and choose the menu path Configure→System Messages.

Setting	Description	Sample Value(s)
API Source	The particular type of processing that needs to have a message displayed on.	DocManager: Store New and Assign
CMS Profile	The content management system profile associated with this particular message	<any profile that you have configured>
CMS Use Tag	The content management system use tag associated with this particular message.	<any tag that you have configured>
API Type	Optional API type of call that you wish the message to be associated with.	
Enable/Disable	If enabled this message will be displayed to users utilizing the associated processing tied to this message.	
Message Type	The type of message which can be just informational or an error which will restrict further processing.	Information
Message Text	The first line of text to display to the user.	We are currently experiencing a network slowdown.
Message Text	The second line of text to display to the user.	Please be patient!
Message Text	The third line of text to display to the user.	
En. Verbose Logging	If enabled additional logging will be added to the gateway dashboard logging. Note this has a small hit to performance.	

4.1.14. Utilities: CMS Load (Beta)

CMS Load (in beta) allows you to load various CMS system components into docuflow SAP (properties, docclasses, etc.). Please note this utility is currently under beta and no further documentation is available at this time.

Once loaded, any docuflow configuration that references a document property can be chosen via the drop down matchode for possible values.

To execute a CMS Load, enter transaction /dflow/docuflowsys and choose the menu path Configure→Execute→CMS Load.

4.1.15. Utilities: Link Maintenance

Link Maintenance allows a simple tool to maintain ArchiveLink links.

This is often used after client refreshes in order to "reset" the links in non-production environments as well.

This utility differs from the Attachment List Dashboard in that allows for multiple links to be deleted at one time while the dashboard only allows for one deletion at a time.

To execute Link Maintenance, enter transaction /dflow/docuflowsys and choose the menu path Configure→Dashboards→Link Manager->Mass Maintenance.

Make sure to limit the search results by at least a Doctype and Archive ID (required fields) to help narrow down the results.

This utility can also be run in the background for large processing to be completed in the background to avoid timeout issues with foreground/dialogue processing.

Setting	Description	Sample Value(s)
Test Only	If selected the actual processing of the ArchiveLink entries will only be simulated and not actually run.	
Production System Override	If selected this allows for ArchiveLink entries to be adjusted in a production environment.	
Wipe ArchiveLink Links	If selected the ArchiveLink entries filtered on will be deleted from the underlying ArchiveLink tables.	
Link Table (eg: TOA01)	The link table to restrict processing on.	TOA01
SAP Object	The SAP business object in particular to filter the ArchiveLink entries on.	BUS2014
Object ID	A particular unique object id to filter the ArchiveLink entries on. This unique object id will only be unique to a particular SAP business object so make sure to include additional filter criteria.	0000000001
Archive ID	The ArchiveLink repository to filter the ArchiveLink entries on.	ZC
CMS ID	The content management system document id in particular to filter the ArchiveLink entries on.	1232456
Doctype	The ArchiveLink document type to filter the ArchiveLink entries on.	ZCONTRACT
Archive Date	The ArchiveLink archive date to filter the ArchiveLink entries on.	12/25/2020
Reserve	The ArchiveLink reserve entry to filter the ArchiveLink entries on.	

4.1.16. Utilities: Metadata Push (MDPro)

Metadata Push (or MDPro), is a utility that allows you to manually trigger or schedule metadata updates from SAP to your CMS system based on configured DocManager rules.

There are two way to execute a metadata push:

1. Directly via the MDPro transaction/menu path and you enter the input criteria to determine what gets pushed
 1. To execute MDPro, enter transaction /dflow/docuflowsys and choose the menu path Configure→Execute→Metadata Push.
 2. From a docuflow API Gateway results set, and then "loading" MDPro from that result set.

Using the docuflow API gateway allows you to use entries in the gateway to populate the input selection screen to MDPro.

If entering MDProd directly, leverage the Link Selection to determine which ArchiveLink link entries MDPro will push metadata for.

Note that all MDPro activity is logged in the docuflow API Gateway.

Link Selection

Use any combination of the standard ArchiveLink link table elements to filter which links you would like to push metadata for

Action

Setting	Description
Production System Override	Enable this for production environments
Descriptive Label	Give your "run" a meaningful label. This will assist you in searching the docuflow API Gateway.
View Only	This does not execute the (U)pdate API call to the content repository. It is equivalent to a test run.
Push Metadata	This is an actual run that will update the content repository.

Gateway Update

Setting	Description
Update Existing (Slower)	If there are any previous ERROR entries from (U)pdate for a particular SAP object / Object ID / Archive ID / CMS ID / AR Object unique key in the docuflow API gateway, it will be updated to reflect a success. Therefore, you can report on all ERRORS in the gateway for outstanding errors if desired. There is performance overhead with this option. You can always report (via date or label or other criteria) the MDPro run as an alternative.
No Update (Faster)	This option does not update historical gateway entries.

Display Options

Setting	Description
Detail ALV	Results are displayed in ALV report format
Summary Text Only	A simple text summary only is displayed instead of the ALV report

4.1.17. Utilities: Cleaning Gateway Errors

This function allows you to maintain (ie: delete) errors in the docuflow API gateway that may have subsequently been processed successfully. This allows the API gateway to be more "clean" and have less false negatives as its history grows and as corrections are made.

We recommend you run the program monthly, with the input variant set for a date equal to + 1 day since the last time it was ran, or, aligned with how often you run the API gateway to use the errors to trigger a resolution action.

To execute a gateway error clean, enter transaction /dflow/docuflowsys and choose the menu path Configure→Execute→Clean Gateway Errors.

Selection Criteria

Setting	Description	Sample Value(s)
CMS Profile	The content management system profile to limit cleaning on.	
CMS Use Tag	The use tag to limit cleaning on.	
CMS Repository Type	The content management system repository type to limit cleaning on.	
CMS RFC Destination	The remote function call (RFC) destination to limit cleaning on.	
API Source	The source of the api calls to limit cleaning on.	SAPLINK
API Type	The particular api call type to limit cleaning on.	
SAP object ID	The particular SAP ArchiveLink object keys to limit cleaning on.	
Document Type	The SAP ArchiveLink document types to limit cleaning on.	
Date	The date range to limit cleaning to.	
Time	The time range to limit cleaning on.	
User Name	The SAP user name to limit cleaning on.	
CMS SAP User	The SAP user name to limit cleaning on.	
CMS Rep./Store	The content management system repository or object store to limit cleaning on.	
CMS ID	The content management system id(s) to limit cleaning on.	
CMS Filename	The content management system filename(s) to limit cleaning on.	
CMS Type	The content management system document class, object type or metadata template to limit cleaning on.	
Test Only	If selected the cleaning isn't actually performed and the process is just simulated.	

Creating and Scheduling Background Job

1. Consult with your SAP Basis team to create the job using the program "/DFLOW/ZCMSAPI_GTW_ERROR_CLEAN"
2. Create a variant for the above program using a dynamic date selection per the below (as an example)
 1. Select "Save as Variant..." from the "Goto" menu under the "Variants".

2. Adjust the date range by Selecting "X:Dynamic Date Calculation (System Date)" for the Selection variable of the "Date" Field name.
3. Configure the "Name of Variable (Input Only Using F4)" to select the particular date range to archive.
 1. We suggest selecting I for include (the I in the I/E column) with the LT (for less than) option the "Current date +/- ??? days" selection from the choices.
4. Enter 31- for the ??? value.
5. This cleans errors that have been processed successfully older than 31 days.

4.1.18. Utilities: Archiving Gateway Data

An overview and instructions on how to schedule the docuflow API maintenance job can be found in our Base Installation and Configuration Guide located here:

<https://docs.docuflow.general.sg.versafire.com/utilities-archiving-gateway-data>

4.1.19. Utilities: Attachment List Dashboard

This is a simple report on ArchiveLink links with an enhanced selection criteria across all link tables, that also allows for adjustment of individual link entries.

This utility differs from the Link Maintenance utility in that it only allows for single links to be maintained/deleted versus the Link Maintenance dashboard allows for mass deletion.

To execute the attachment list report, enter transaction /dflow/sys and choose the menu path Configure→Dashboards→Attachment List.

Make sure to limit the search results by at least an SAP Object or AR Object(Doctype) to help narrow down the results.

The toolbar that runs along the top of the attachment list dashboard can allow you to further refine your view of the attachment list table. Included in the toolbar is some additional functionality like the ability to customize layouts and delete link entries.

Setting	Description	Sample Value(s)
SAP Object	The SAP business object in particular to filter the ArchiveLink entries on.	BUS2014
Object ID	A particular unique object id to filter the ArchiveLink entries on. This unique object id will only be unique to a particular SAP business object so make sure to include additional filter criteria.	0000000001
Archive ID	The ArchiveLink repository to filter the ArchiveLink entries on.	ZC
CMS ID	The content management system document id in particular to filter the ArchiveLink entries on.	1232456
AR Object (Doctype)	The ArchiveLink document type to filter the ArchiveLink entries on.	ZCONTRACT
Archive Date	The ArchiveLink archive date to filter the ArchiveLink entries on.	12/25/2020
Reserve	The ArchiveLink reserve entry to filter the ArchiveLink entries on.	
Layout	Optionally you can select a previously saved layout for the gateway dashboard before displaying the results.	DEFAULT

4.1.20. Utilities: Determine Attachments

This section provides tips and tricks to determine what links exist in your SAP system. There are 2 main types of links (outside of SAP DMS DIR KRPO):

1. ArchiveLink (External Repository)
2. GOS (SAP Database)

ArchiveLink (External Repository)

ArchiveLink links are stored in the SAP table TOA01. However, via transaction OAC3 (table TOAOM), you can specify other link tables at the SAP object and ArchiveLink doctype level. Some customers will have other link tables such as TOA02, TOA03, or may have copied this table to ZOA01 referenced in OAC3.

One additional table TOAAT will contain the filename and created by. Not this table has not always been available in the SAP landscape, therefore if you have very old entries in a link table such as TOA01 you might not have a corresponding entry in TOAAT. This table is referenced by field ARC_DOC_ID.

There are a couple options to determine how many links you have and potentially group these by SAP object and/or ArchiveLink doctype.

1. Manual method
 1. Enter transaction OAC3 (or view table TOAOM) and determine what link table are *potentially* being used
 2. Export each one of those tables to Excel using transaction SE16 or SE17.
 3. Group by SAP_OBJECT and if desired, AR_OBJECT (ArchiveLink doctype). The other fields are self explanatory.
2. Script method
 1. Enlist your SAP basis team to write scripts on the backend againsts your database. For a sample of SAP Hana scripts please reach out to your VersaFile representative.
3. docuflow Migrator
 1. If you have docuflow ECM installed in your SAP system you can run the transaction /DFLOW/MIGRATOR to analyze your ArchiveLink attachments.
4. ABAP
 1. You can create a small basic ABAP program in DEV and promote to your SAP production instance and execute. For a sample of SAP ABAP code please reach out to your VersaFile representative.

GOS (SAP Database)

GOS links and attachment content are stored in three primary tables, two with the binary relationship (ie: link) SRGBTBREL for business object links and SOFFPHIO for SAP office notes, attachments and outputs) and one for the content itself (unlike ArchiveLink where the content is stored in an external repository) SOFFCONT1.

One additional table SOOD will contain the filename, size, and storage date for *most* entries. This table is referenced by fields of the composite key OBJTP, OBJJR, and OBJNO.

There are a couple options to determine how many links you have and potentially group these by SAP object and/or ArchiveLink doctype.

1. Manual method
 1. Export each one of the tables above to Excel using transaction SE16 or SE17.
 2. For a count of attachments, SRGBTBREL can be used and the SAP Object column is TYEPID_A. Be aware of column RELTYPE that determines if the entry is an attachment or a note.

3. For size and storage date, lookup against SOOD.

2. Script method

1. Enlist your SAP basis team to write scripts on the backend againsts your database.

3. docuflow Migrator

1. If you have docuflow ECM installed in your SAP system you can run the transaction /DFLOW/MIGRATOR to analyze your ArchiveLink attachments.

4. ABAP

1. You can create a small basic ABAP program in DEV and promote to your SAP production instance and execute. For a sample of SAP ABAP code please reach out to your VersaFile representative.

4.1.21. Working with the API Gateway and SCRUD Test Harness

The docuflow API gateway provides the below functions:

1. Reporting of all docuflow processing, useful for a wide array of reporting and analysis
2. An entry point to the SCRUD (and PL) Test Harness
3. Load MDPro (metadata push) from an API gateway results set

Regardless of the function above, execute the docuflow API gateway via either:

1. Transaction /dflow/docuflowsys and choose the button API Gateway
2. Enter the transaction /dflow/docuflowgtw

Once a results set is displayed, you see all data columns left to right - **these can be filtered** to provide a more focused view depending on what you are looking for/at by choosing the `"Views" menu path` when viewing the report results.

Within the both summary and detail level line, you can drill down on the columns indicated with a green checkbox to view further data, eg: viewing metadata, viewing the full return table, etc.

Due to the volume of logging, it is recommended you routinely purge the gateway (primarily at the detail level) to keep table space to a minium. Refer to the section in this document "Utilities: Archiving Gateway Data".

Selection Criteria

Setting	Description	Sample Value(s)
CMS Profile	The CMS profile to filter on.	Box_1
CMS Tag	The CMS Use Tag to filter on.	DocuFlow Document Storage
CMS Repository Type	The CMS repository type (FileNet P8, Documentum, or Box)	BX
RFC Destination	The RFC destination to filter on.	ZDOCUFLOW_CMSAPI_01
API Source	The source of the process to filter on like viewing or choice list processing.	STORENEW
API Type	The underlying API call to filter on (search, create, read, update, delete, ping and load). These calls are used together to form various processing so it may be interesting to focus the results on just a port of say the viewing process.	READ
SAP Key	The unique SAP ArchiveLink key for a particular object to filter on.	00000000013000
DocType	The ArchiveLink document type to filter on.	ZCONTRACT
Call Date	The date when the api call was made to filter on	12/25/2020
Call Time	The time when the api call was made to filter on	00:00:00
Call User	The originating user that made the API call.	DOCUFLOW
Call Return Message Type	The return that was associated with the API call which can either be an (E)rror, (S)uccess, (W)arning, or (I)nfo.	E
SAP User	The originating user that made the API call.	DOCUFLOW
CMS ID	The content management system document id number to filter on.	123456
Label	Migration and MDPro labels	docuflow is great
Hide System Ping	If selected the ping API messages are not reported.	
Hide DocViewing	If selected, and if docuflow DocManager or the TOAEX method is used for viewing content, then those entries are not reported.	
Hide Detail Entries	If selected the detail API messages are not reported. By default, the gateway summarizes API calls into 1 "summary" entry in the gateway for ease of reporting. If you wish to see the underlying API calls (for example, the CREATE and UPDATE for a DocManager Storage request), click this off.	

Setting	Description	Sample Value(s)
Default Layout	Optionally you can select a previously saved layout for the gateway dashboard before displaying the results.	DEFAULT

Reporting on docuflow Processing

All docuflow processing is logged at a summary and detail level within the API Gateway, and therefore reporting can be done so for various reasons such as:

1. Viewing general activity
2. Troubleshooting and unhappy path scenarios
3. Exporting results for performance and load analysis
4. etc...

Familiarize yourself with the input selection criteria and drill down information available. Also note that by default document viewing activity and "detailed entries" are hidden. See input criteria above "Hide DocViewing" and "Hide Detail Entries" for more information and to see what additional data is available.

SCRUD "Test" Harness

The SCRUD "Test" Harness allows the testing of basic SCRUD API's with no coding via the docuflow API Gateway. The (S)earch, (C)reate, (R)ead, (U)pdate, and (D)elele API's that are executed are identical to those called when using the ABAP SCRUD API set (refer to the docuflow Developers Guide) or the docuflow Enterprise Connector test harness.

All docuflow processing leverages the same endpoint API calls and resulting calls in the docuflow middleware and therefore the test harness can provide you the ability to simulate all docuflow processing.

To enter the SCRUD "Test" Harness:

1. Execute the API gateway using input parameters that will return:
 1. a small dataset *or*
 2. NO dataset *or*
 3. a dataset that contains the line(s) that you wish to use to action (or "re-process") using SCRUD functions
 1. "Re-process" in this context means that for Read and Update, you can choose an existing line in the API gateway report and the test harness will use those values as the input to the test harness (you have the option to change those values of course, but they can be handy to reduce entering all the data).
2. Choose the "Execute" menu path and choose the applicable SCRUD function.
 1. The"New" options bring up a new/empty SCRUD input screen
 2. The"Again" options require you to select a reported line (click once anywhere on the desired results screen on the line of choice) and basically pre-populates the SCRUD input screen with already utilized values in the case of (R)ead and (U)pdate. Not to worry....these values can be changed prior to execution.

**** NOTE **** Even though this is referred to as "test harness", all actions are processed fully to the CMS endpoint. So albeit you can use this to test (Search/Read of course have no lasting effect other than load and performance), in the case of Create/Update/Delete, especially in a production environment, these actions must be approached with caution.

Loading MDPro

MDPro allows you to push metadata from SAP to your CMS endpoint, and can be executed directly *or* alternatively, you can "load" the input selection criteria from the API Gateway if you are reporting on (U)pdate API activity.

Refer to the section "Utilities: Metadata Push (MDPro)" for more detail on how to run MDPro.

4.2. Container Deployment Guide

Overview

This guide explains how to deploy Docuflow using sample YAML files or integrating with your CI/CD pipeline.

Provided YAML Configurations

We offer example YAML files for:

- Deployment
- Service
- Route

These can be used as templates or modified to suit your environment.

Container Registry & Tags

Docuflow containers are published to our Container Registry: `versafiler.azurecr.io`

Pull credential for the container registry will be provided by your account manager

Available image tags:

- `latest` tags the latest image of that version
- `qa-latest` tags the qa-latest image of that version, will be typically released 1 week before tagged as latest
- `date-stamp` (e.g., `20250710`) the tag will stay and is available for up to 12 months.

Example: `versafiler.azurecr.io/docuflow/6.9/df4sap:latest`

You may:

- Pull directly from our Container Registry
- Copy to your own container registry

Ingress Configuration

Supported ingress methods:

- Ingress or Route
- Service Type: LoadBalancer

Ingress or Route: Currently each port should have a dedicated ingress/route configuration:

- 1 for Docuflow GUI
- 1 per ArchiveLink or CMIS content server
- 1 per WebSocket RFC Connection

Service Type: LoadBalancer

- One per Docuflow deployment

□ Storage and Configuration

Docuflow can operate without RWX (ReadWriteMany) storage:

- GUI runs in **Read-Only** mode
- Configuration files loaded via **ConfigMap**
- SAP certificate added to an **Secret**
- Logs written to `stdout` and `stderr`

(Assumes container output is collected by a monitoring solution)

4.2.1. Step by Step Deployment OCP

Intro

Example how to deploy on Red Hat OpenShift Container Platform (`RH OCP`)

- Use configMap for configuration
- No RWX storage
- GUI read only

Bash/Shell commands to deploy on RH OCP


```

##set vars to use in script
vfns="dflow-ns"
vflocalfilepath="/home/myuser/dflow-files"

#make local folder
mkdir -p $vflocalfilepath/$vfns/ssl-cert
mkdir -p $vflocalfilepath/$vfns/config
mkdir -p $vflocalfilepath/$vfns/sap-cert

##copy the main.ini file to the 'config' file.
##copy the 'certificate.crt' and 'cert-private.key' to the 'ssl-cert' folder
##copy the CS1P81-Y7.cer to the 'sap-cert' folder, after send cert in SAP check /dflow/config/certificates folder for the cert.
### naming convention is <<content.system>>-<<Archive Name>>.cer

## create NS
kubectl create namespace $vfns
##set as default ns
kubectl config set-context --current --namespace=$vfns

## create image pull secret ##
kubectl -n $vfns create secret docker-registry image-repo-login-secret \
  --docker-username=my-username \
  --docker-password=pass123 \
  --docker-server=versafire.azurecr.io \
  --docker-email=dummy@localdummy.local

##deploy config (via configMaps and secret)
#kubectl -n $vfns delete cm dflow-config
kubectl -n $vfns create configmap dflow-config --from-file=$vflocalfilepath/$vfns/config

## import SSL cert and key as secret
#kubectl -n $vfns delete secret dflow-x509
kubectl -n $vfns create secret tls dflow-x509 \
  --cert=$vflocalfilepath/$vfns/ssl-cert/certificate.crt \
  --key=$vflocalfilepath/$vfns/ssl-cert/cert-private.key

##add SAP certs we trust (seckey sigValidation) as secret
#kubectl -n $vfns delete secret dflow-sap-cert
kubectl -n $vfns create secret generic dflow-sap-cert --from-file=$vflocalfilepath/$vfns/sap-cert

##yaml with deployment and service kind.
kubectl -n $vfns apply -f dflow-deploy.yml

## deploy route for GUI and CS
## modify first to set the hostname to correct hostname.

#check hostname
yq .spec.host route-dflow-cs.yaml

kubectl -n $vfns apply -f route-dflow-gui.yaml
kubectl -n $vfns apply -f route-dflow-cs.yaml

##tips how to scale up/down

```

YAML files

 [dflow-deploy.yml](#) [route-dflow-cs.yaml](#) [route-dflow-gui-scrud.yaml](#)

Example of config folders

 [main.ini](#) [Config.properties](#)

4.3. Developers Guide



SAP[®] Certified
Integration with SAP S/4HANA[®]

<https://www.versafire.com>

4.3.1. Purpose of Guide

The goal of this guide is to provide detailed overview of all developer related components within docuflow. These components fall into two categories:

1. docuflow SAP development:

1. Class/Method Plug-In's: Class-methods that may be leveraged in standard docuflow configuration
2. docuflow BADI's: Enhancement spots for additional customization of docuflow processing
3. S(earch) C(reate) R(read) U(pdate) D(elete) API's: Class-methods for each SCRUD functionality in which the standard docuflow processing leverages, which is also open for direct customer usage in custom scenarios.
(P)ing and (L) API's: Class-methods for multiple levels of ping to the middleware and CMS, as well as the ability to load CMS repository info into SAP.

2. docuflow Web services usage:

1. HTTPS Setup
2. docuflow SCRUD web services
3. docuflow BAPI web service

Lastly, as our customers leverage the docuflow logic under the hood more and more, we hear of common routines and methods that may be of use in your custom requirements. These are located under Useful / Common docuflow Methods.

4.3.2. Class/Method Plug-In's

Class/method ABAP development may be necessary in the below docuflow processing scenarios:

1. DocManager/DocMigrator/Publisher: Metadata maps for sending SAP data to your CMS
2. DocManager: Custom processing during Inbound Processing/SAPLink
3. DocManager: SAPKey determination during Inbound Processing/SAPLink
4. DocMigrator: Custom source link determination and compilation
5. Publisher: Triggering storing and linking via docManager to your CMS

Each one of these is optional within their respective docuflow configuration. Refer to the docuflow Solution Guide (separate) for more information on this configuration.

4.3.2.1. DocManager Metadata maps

As a more powerful option to using the metadata map configuration, class/method metadata maps allow you to compile the SAP data via ABAP that will update your CMS documents (if configured) for outbound docuflow storage and mass metadata push/updates.

Example Method: /DFLOW/CL_DOCMANAGER_HELPER⇒METADATA_EXAMPLE

The below example is provided with the docuflow product.

Signature:

Type	Parameter	Description	Sample Value
Importing	IV_SAP_OBJECT	The SAP business object associated with this process	BKPF
Importing	IV_DOCTYPE	The ArchiveLink document type to use for linking documents with this process	ZINVOICE01
Importing	IV_SAPKEY	The unique identifier of the SAP records the link will be created to	300019000000002013
Importing	IV_ARC_DOC_ID	The unique identifier of the CMS document	{83281172-DD17-4B8A-8913-9E5E69BA3125}
Exporting	ET_METADATA	This is the resulting table of key (case sensitive CMS property name) value (SAP data) pairs of the SAP data to update the CMS document with that you must fill with your ABAP code	

Notes:

- If you want to update a multi-line property (only supported in Documentum and FileNet CMS systems), place an X in the field ZMULTI
- For Box CMS systems where a date field is part of the metadata being sent, the format must be MM/dd/yyyy hh:mm:ss a z, where a z = the AM/PM and TIMEZONE
 - example: 12/31/9999 12:00:00 PM MST
- To assist with the date format for Box, you may choose to utilize the DocuFlow helper method '/dflow/cl_utilities_helper⇒format_date_plus_sys_timezone' to format a system date field to the format desired and also appending the system timezone
- To assist with the date format for FileNet and Documentum, you may choose to utilize the DocuFlow helper method '/dflow/cl_utilities_helper⇒format_date' to format a system date field to the format desired

4.3.2.2. DocManager Inbound Processing

In your CMS system, the property configured to be polled for as the indicator whether a document should be processed/call SAP (referred to as the docuflow SAPLink Trigger property or filter), when found, will trigger a call to the docuflow SAP gateway BAPI at a configured interval.

The DocManager Inbound Processing class/method allows you to either:

1. Custom: Execute your own custom process entirely or in combination with any other DocManager configured option mentioned above
2. Link Create: Optionally manipulate all exporting and changing method parameters for subsequent DocManager configured Create Link processing (called after Custom if configured). Primarily this is used to find/confirm/format/compile/specify the "SAPKey" for what SAP record to link to.

Example Method: **/DFLOW/CL_DOCMANAGER_HELPER⇒SAPLINK_EXAMPLE**

The below example is provided with the docuflow product.

Signature:

Type	Parameter	Description	Sample Value
Importing	IV_MO_DE	This class/method is shared in the docuflow configuration for both “types” of processing (custom vs. SAP Key determination). Which processing is being executed by docuflow configuration is indicated by this value.	SAPKEY or CUSTOM
Importing	IV_DOCUMENT_ID	The CMS unique identifier (ie: UUID/GUID)	485029877609
Exporting	EV_GO_ON_FLAG	CUSTOM processing only: setting the goon flag to ABAP_TRUE (ie: X)	ABAP_FALSE or empty vs. ABAP_TRUE (X)
Exporting	ET_RETURN	Anything other than a line added with the value ‘E’ in the TYPE field (ie: any line with S or I or W as a TYPE) will result in the docuflow middleware considering the process a success and the inbound request will be set to complete (ie: the SAPLink “trigger” property set to false and will NOT be reprocessed). If a line in the table with the value ‘E’ in the TYPE field is found, the entry in the CMS system will be reprocessed at the next interval. Regardless of the TYPE value, all entries added to this table will be logged and viewable in the docuflow API Gateway and thus can be leverage to log/report on the processing of your logic using any fields of the table (Message Text being the most common)	
Exporting	EV_SAPKEY	Used extensively in Link Create, this is the fully formatted SAPKey that will be used to create the ArchiveLink	FI Invoice: 30001900000 2222019 Vend or Master: 0000033985
Exporting	EV_GUID	Used only in CRM: this is the GUID that will be used in place of EV_SAPKEY to generate the viewing URL to link	
Channeling	CT_METADATA	This table contains the metadata from your CMS system (what is sent over to SAP is determined by the docuflow middleware configuration for Inbound Processing metadata). This table it not typically changed, but rather read to leverage the metadata from your CMS system to either execute a custom process or in the case of Link Create, determine the EV_SAPKEY. This might be to take different metadata fields and do a lookup to get a key, or maybe simply formatting the key to be padded so	

Type	Parameter	Description	Sample Value
		ArchiveLink will recognize it, or possible to verify the incoming key is correct (and if not, add an E type line to the return table). The values of this table can be viewed under the "SMD" (SAPLink Metadata) column in the docuflow API Gateway.	
Channeling	CT_FILE	Table containing the CMS inbound files related to the triggering of the SAPLink process.	
Channeling	CT_CUSTOM	For future use.	
Channeling	CS_ZD	Table containing the existing DocManager configuration.	
Channeling	CS_ZC	Table containing the existing Repository ID configuration.	
Channeling	CS_ZC	Table containing the existing CMS Profile configuration.	

Notes:

- Custom processing is executed first, follow by Link Create (SAPKey) processing '2nd' (if no goon flag is received).
- Any exporting or changing parameter values manipulated by Custom processing will be passed on to Link Create (SAPKey) processing.

4.3.2.3. DocMigrator Custom Source Link Determination

Not specified in the configuration as the other docuflow class/method plug-ins, but rather at runtime in the DocMigrator load screen, this ABAP method allows you to compile the source entries for a migration. This is useful in scenarios where the standard ArchiveLink source selection criteria does not satisfy your query (ie: all PO based invoices from a particular company code, thus you cannot use the object_id to determine those from a company code since MM invoices do not have company code as part of the SAPKey).

Example Method: /DFLOW/CL_MIGRATOR_HELPER⇒ GET_ABAP_LINKS_EXAMPLE

The below example is provided with the docuflow product.

Signature:

Type	Parameter	Description	Sample Value
Exporting	EV_SOURCE_TYPE	The source type is either an ArchiveLink Content Server (A) or docuflow ECM (D)	A or D
Exporting	ET_CONNECT	The ArchiveLink table entries that you either custom retrieve or custom build.	Structure of TOAV0 – standard SAP ArchiveLink links

4.3.2.4. Publisher: Triggering Storing and Linking

In your custom code or enhancements (primarily in the "processing" or appropriate form of your Z version of the message output), add code that calls the "publish_message" method of the "/dflow/cl_publishing".

The example below shows the retrieval of the spool id as well as various other input parameters like the filename (used for the document that will be stored in the content management system) so that the call to the "publish_message" method can be made.

Calls to the method can also be made using the SAP OTF table or by passing in the PDF as a string instead of using the spool id. Adjustments will need to be made for the particular message output program/function module that is getting DocuFlow functionality added to it.

Consult with an ABAP developer for assistance in gathering the correct information so that a call to the "publish_message" can be made successfully.

Example

* Ensure the user has selected to PRINT the document (ie. Not Fax/EDI/etc)

```
IF finaa-nacha = 1. "Print
```

```
    " define the variables we need to make the call to publisher
```

```
    DATA lo_publishing TYPE REF TO /dflow/cl_publishing.
```

```
    DATA lv_object_id TYPE /dflow/cl_publishing=>gt_object_id.
```

```
    DATA lv_arc_doc_id TYPE /dflow/cl_publishing=>gt_id.
```

```
    DATA lv_filename TYPE file_table-filename.
```

```
    DATA lv_document_linked type abap_bool.
```

```
    DATA lv_document_stored type abap_bool.
```

```
    DATA lv_message type string.
```

```
    " get an instance of the publisher object
```

```
    lo_publishing = /dflow/cl_publishing=>get_instance( ).
```

```
    " set FILENAME (format the filename to how you would like)
```

```
    CONCATENATE bkorm-konto bkorm-bukrs INTO lv_filename SEPARATED BY '_'.
```

```
    " determine the SAP object id for this particular object
```

```
    lv_object_id+0(10) = bkorm-konto. "Customer number
```

```
    lv_object_id+10(4) = bkorm-bukrs. "Company code
```

```
    " try to make the call to the Publisher method and catch any exceptions
```

" in this example the sap object type and document type are hard coded

TRY.

CALL METHOD lo_publishing→publish_message

EXPORTING

iv_spool_id = itcpp-tdspoolid

* it_otf =

* iv_pdf_xstring =

iv_filename = lv_filename

iv_sap_object = 'KNB1'

iv_ar_object = 'ZCRMPUBL01'

iv_object_id = lv_object_id

IMPORTING

ev_arc_doc_id = lv_arc_doc_id

ev_error = lv_error

ev_document_stored = lv_document_stored

ev_document_linked = lv_document_linked

et_return = lt_return.

.

CATCH /dflow/cx_common .

lv_message = lo_error→get_text().

lv_error = abap_true.

MESSAGE lv_message TYPE 'I'.

ENDTRY.

READ TABLE lt_return INTO ls_return WITH KEY type = 'E'.

IF sy-subrc EQ 0.

MOVE: 'ZTC' TO syst-msgid,

'E' TO syst-msgty,

'018' TO syst-msgno,

ls_return-message(50) TO syst-msgv1,

ls_return-message+50(50) TO syst-msgv2,

ls_return-message+100(50) TO syst-msgv3,

ls_return-message+150(50) TO syst-msgv4.

PERFORM protocol_update.

MOVE: '/DFLOW/ZDOCUFLOW' TO syst-msgid,

'E' TO syst-msgty,

'001' TO syst-msgno,

'docuflow storage connection error - contact admin' TO syst-msgv1.

PERFORM protocol_update.

MOVE c_rc01 TO g_subrc.

```
ELSE.  
  MOVE: '/DFLOW/ZDOCUFLOW' TO syst-msgid,  
        'S' TO syst-msgty,  
        '001' TO syst-msgno,  
        'docuflow storage success' TO syst-msgv1.  
  PERFORM protocol_update.  
ENDIF.
```

```
ENDIF.
```

4.3.3. docuflow BADI's

docuflow provides BAPI's/enhancements spots in the following modules:

1. Base Content API
2. DocManager

4.3.3.1. Base Content API

The below DocuFlow BADI's/enhancements spots are available within the Content API base component:

/DFLOW/ATTACHMENT_LIST

- BADI: /DFLOW/BADI_ATTACHMENT_LIST
 - Allows custom processing of the attachment list(s) used within DocuFlow components (including API Gateway and DocManager). Useful for hiding attachments in certain situations that might be confidential to certain users (ex: document title for HR) above and beyond SAP authorization checks as well as to enable Virtual Linking functionality.
 - Virtual Linking is the concept of being able to display links in an attachment list that are from another SAP record (or object) as if they were linked to the SAP record being viewed. Examples of this are to show original MM invoices on the resulting FI invoice, original outline agreement attachments when viewing a purchase order, etc.

/DFLOW/GENERATE_VIEWING_URL

- BADI: /DFLOW/BADI_GEN_VIEWING_URL
 - Allows custom manipulation of the URL generated and launched when viewing documents from the SAP or DocManager attachment list.

4.3.3.2. DocManager

The below BADI's/enhancement spots are available within the DocManager Add-On:

/DFLOW/DOCMANGER_CRM_BO_DETAIL

- BADI: /DFLOW/BADI_DOCMGR_CRM_BO_DTL
 - Allows custom manipulation of CRM Store New processing business object, sap_object, doctype, object_id, and filename details.

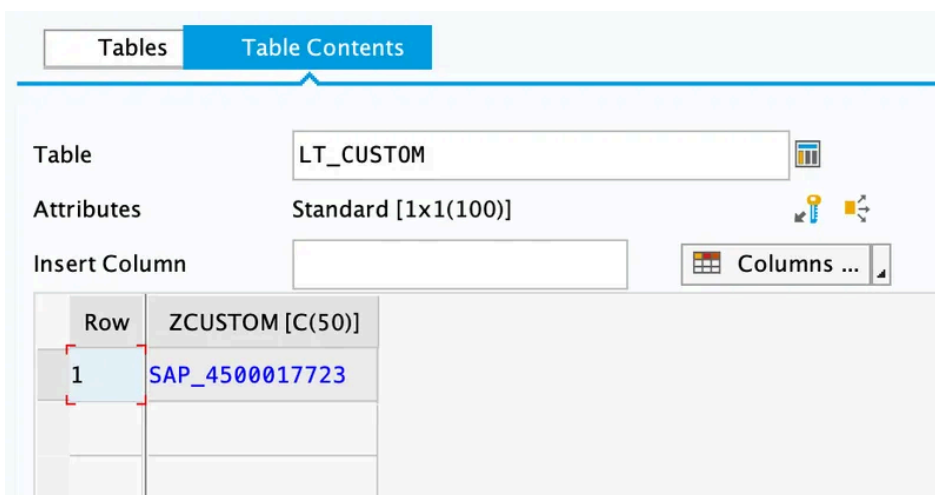
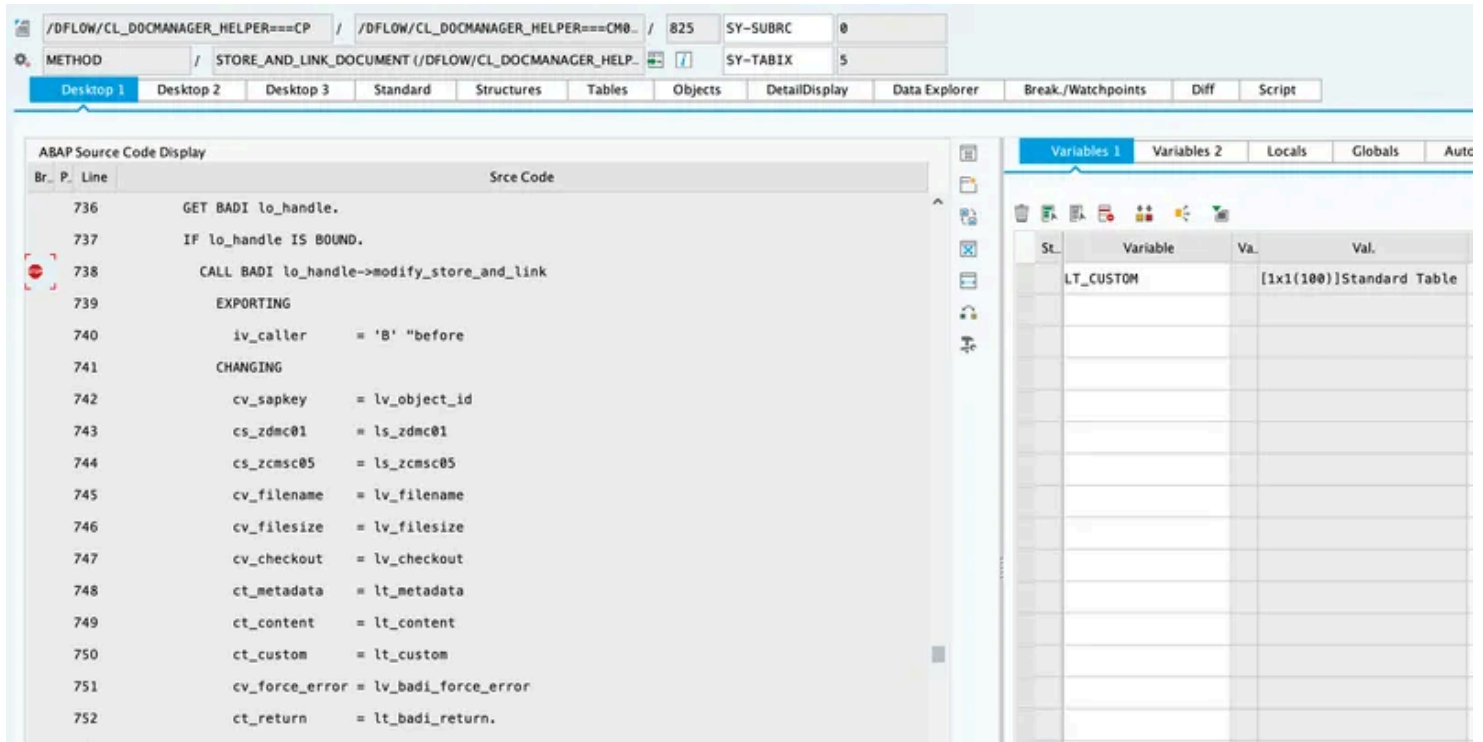
/DFLOW/DOCMANGER_SFO_STATUS

- BADI: /DFLOW/BADI_DOCMGR_SFO_STATUS
 - Allows custom determination of whether to enable or disable the DocManager SFO menu option.
 - In the code, leverage the input parameter IS_LPORB to determine the SAP Object and ArchiveLink document type
 - Set the parameter CV_EP_STATUS with one of the following values:
 - 0 Active
 - 1 Inactive
 - 2 Invisible
 - Note that you do NOT have to set to 0 - Active, as this is the default

/DFLOW/DOCMANGER_STOREANDLINK

- BADI: /DFLOW/BADI_DOCMGR_STORE_LINK
 - Allows custom processing of the Store New process *prior* to the Content API "CREATE" call with iv_caller 'B'.
 - One of the most common reasons to implement this BADI under iv_caller 'B' is to create a folder structure during the document creation process.
 - To do so, you have two options:
 - 1. 1 parent with <n> flat folders
 - 2. All folders as nested folders: in this case, set the variable cv_nested_folders to "X".
 - Note: you can pass the folder name each time you create a document. docuflow will first check to see if that name exists under the configured location/folder ID in the DocManager rule. If it does not exist, it will create it. If it exists, no action is performed for a folder creation.
 - When you create a multi level folder structure, you will be able to indicate which folder to store the document in. By default, the document is stored in the top level sub-folder. To place the content (if you are storing content, which is optional) in a folder other than the top parent folder, indicate this with a ~ tilde.
 - Example of CT_CUSTOM to create 4 folders, and place content in parent folder:
 - SAP_4500017723
SAP_SubFolder1
SubFolder2
SubFolder3
 - or
 - ~SAP_4500017723
SAP_SubFolder1
SubFolder2
SubFolder3

- Example of CT_CUSTOM to create 4 folders, and place content in lowest sub-folder:
 - SAP_4500017723
 - SAP_SubFolder1
 - SubFolder2
 - ~SubFolder3
- Below is an example to create 1 sub folder showing the ABAP debugger:



- Allows custom processing of the Store New process *after* the Content API "CREATE" call and also the ArchiveLink create link with iv_caller 'A'.
 - This is called in both success and error occurrences which is indicated by the entries in the return table
 - In the case of success, IV_ARC_DOC_ID will be populated
 - Some customers implement this to interpret the return table CT_RETURN when creating folders. In some scenarios the folder ID's created are useful to know.

- When you use the IT_CUSTOM folder to create folders, the return table will also pass back the folder ID's created. The parent folder is indicated by a ! exclamation point, all sub-folders are indicated by a ~ tild.
 - Example of return table when folders are created:
!SAP_4500017723,147811962292
~SAP_SubFolder1,938382828
~SAP_SubFolder2,938382829
~SAP_SubFolder3,938382830

4.3.4. SCRUD API's

All standard docuflow functionality and processing relies on 5 methods within the class **/DFLOW/CL_CMS_API**.

These methods are listed below and are free to use directly.

1. CMS_API_SEARCH
2. CMS_API_CREATE
3. CMS_API_READ
4. CMS_API_UPDATE
5. CMS_API_DELETE

When using the SCRUD API's, you may wish to use the docuflow Custom API Rule configuration table. This table with a key of "Label", allows you to specify input parameters to the SCRUD API set and thus avoid potential hard coding of the input parameters required.

By using a "Label" to uniquely identify a Custom API Rule, you can then query the table /DFLOW/ZCMSC06 using that label to get such parameters as CMS Profile, CMS Use Tag, Repository ID, Location, etc.

To create a Custom API Rule when DocManager is not installed, enter transaction /dflow/sys and choose the menu path Configure→API Rules→Custom.

4.3.4.1. CMS_API_SEARCH

Functionality	Description	FileNet Notes	Documentum Notes	Box Notes	O365/Share Point Notes
IV_CM_SP_RO_FIL_E	Required: Determines which configured CMS Profile in docuflow SAP to utilize. Combination of CMS Profile and CMS Tag determine the API destination and thus load across an RFC destination.				
IV_CM_ST_AG	Required: Determine which configured CMS Profile in docuflow SAP to utilize. Combination of CMS Profile and CMS Tag determine the API destination and thus load across an RFC destination.				
IV_RE_PO_SIT_OR_Y	Required: Determines which configured repository in docuflow SAP to utilize	Object Store	Docbase	Default is the only Store/Repository supported since Box currently doesn't support the notion of different repositories.	Site, site collection
IV_TY_PE	Optional: what CMS type to create.	DOCUMENT CLASS Defaults to the searching "Document" document class if unspecified. If "Include Subfolders" configuration is enabled on	OBJECT TYPE Defaults to the searching "dm_document" class and all sub types if unspecified.	METADAT TEMPLATE Includes all documents with or without metadata	CONTENT TYPE Defaults to searching a document of content type

Functionality	Description	FileNet Notes	Documentum Notes	Box Notes	O365/Share Point Notes
		connector it will also include all sub classes of the document class.		templates if empty.	Document if not specified.
IV_US_E_MA_PPI_NG	Optional: use mapping of IV_TYPE to CMS type in Enterprise Connector if CMS. Used if ArchiveLink doctype name to CMS type mapping is required.				
IV_WHER_E	Allows for adding additional search criteria with the search. With Box if used it is the only thing that can be specified while searching.	Refer to the "Relational Queries" section from the online FileNet P8 Info Center documentation for help with constructing advanced where clauses. Examples: DocumentTitle = 'TEST.tif' <Property Name> IS NULL DateCreated > 2016-08-01T12:00:00-08:00	Refer to the Documentum DQL reference material for more information on constructing advanced where clauses. Examples: <Property Name> IS NULL r_modifier <> '<username>' <Property Name> IS NOT NULL r_modify_date > DATE('08/01/2016')	Refer to the Box api for the search api call for what can be used as the where clause. Examples: mdfilter s= [{"templateKey": "<template name>", "scope": "enterprise", "filters": {"<metadata name>": "True"}}] &limit=50&offset=0	Refer to the Graph api for the search api call for what can be used as the where clause. Examples: mdfilters= [{"templateKey": "<template name>", "scope": "enterprise", "filters": {"<metadata name>": "True"}}] &limit=50&offset=0
IV_LOCA_TION_N	Limit the search results to the specified location.	Relative folder names are used with / being the root. Currently unfiled documents cannot be searched from the Profile Wizard utilities.	Relative cabinet/folder names are used with / being the root.	A number for the exact folder location can be specified. Relative folder names are used with a / being the	

Functionality	Description	FileNet Notes	Documentum Notes	Box Notes	O365/Share Point Notes
				root. If a root folder is configured for the Box CMS in the Enterprise connector all relative folder locations will be relative to the root folder configured.	
IV_MAX_RESULTS	Limits the returned results to the amount specified.	Generally the maximum results by default are around 5000. Tests should be made to confirm that the system is configured to return a max of 5000 results though.	The default max search results is currently unknown. Tests should be performed to determine if there is a default maximum if it is of concern.	With a where clause type of query the max results you can get is 200. You may retrieve more with sub folders on the API destination configuration in the connector.	The default max search results is currently unknown. Tests should be performed to determine if there is a default maximum if it is of concern.
IV_MAX_TIME	Limits the returned results to the time specified.				
IV_DATETIME_FORMAT	IF date/time is specified, what format is it in. Leave empty for default configured in enterprise connector (recommended).			Must be MM/dd/yyyy hh:mm:ss a z, where a z = the AM/PM and TIMEZONE example: 12/31/9999 12:00:00 PM MST	Must be MM/dd/yyyy hh:mm:ss a z, where a z = the AM/PM and TIMEZONE example: 12/31/9999 12:00:00 PM MST
ITEM	If specified, only metadata	See notes in ET_METADATA_ID.	See notes in ET_METADATA_ID.	Not supported.	Not supported

Functionality	Description	FileNet Notes	Documentum Notes	Box Notes	O365/Share Point Notes
TA DA TA_ FIL TE R	properties indicated will be returned. By default of no filter is provided, all metadata for a document is retrieved.			See notes in ET_METADATA_ID.	See notes in ET_METADATA_ID.
IT_ ME TA DA TA_ OR DE R	Allows ordering of search results by the specified properties in the order selected.	If a single property is passed in the results are ordered in ascending order for the property name. If unspecified the results are ordered by "Id". Optionally you can specify ASC or DESC after the property name to make the sort happen ascending or descending respectively. *The property being ordered by must also be specified as one of the selected properties.	If a single property is passed in the results are ordered in ascending order for the property name. If unspecified the results are ordered by "r_object_id". Optionally you can specify ASC or DESC after the property name to make the sort happen ascending or descending respectively. *The property being ordered by must also be specified as one of the selected properties.	Not supported	Not supported
ET_ ME TA DA TA_ ID	Contains the metadata retrieved.	If no properties are specified (IT_METADATA_FILTER) then by default only the below will be returned: Id DocumentTitle Some properties can't be returned in searches like: ContentElements FoldersFiledIn Some key properties that have special handling via Search: VersionSeries = will return the version series Id ClassDescription = Gets the symbolic name StorageArea = Gets the display name StoragePolicy = Gets the policy name SecurityPolicy = Gets the policy name OwnerDocument = Gets the Id of	If no properties are specified (IT_METADATA_FILTER) then by default only the below will be returned: r_object_id title Some properties can't be returned in searches like: ContentElements FoldersFiledIn VersionSeries	If no metadata is specified (IT_METADATA_FILTER – currently not supported by Box API endpoint) then by default only the below will be returned: id name	If no metadata is specified (IT_METADATA_FILTER – currently not supported by Graph API endpoint) then by default only the below will be returned: Id_Object Name_Object

Functionality	Description	FileNet Notes	Documentum Notes	Box Notes	O365/Share Point Notes
		the owner document SourceDocument = Gets the Id of the source document ReplicationGroup = Gets the display name of it SecurityFolder = Gets the folder path name PublishingSubsidiaryFolder = Gets the folder path name			
ET_RE_TU_RN	Table that indicates a success of an error. The TYPE field will contain an 'S' or 'E' respectively and the MESSAGE field will contain information in the case of an error.				
CT_CU_ST_OM	Future use.				

General comments:

- Behind the scenes at the connector level, the search uses either a select or a where clause.
- What determines this from the SAP side is the usage of the IV_WHERE parameter vs. not (in tern uses a SELECT).

General FileNet comments:

- If not using paged search results and if the number of results exceeds the content caches default max search results you will only get that many results and you won't be able to see the other results. Generally this is around 5,000 results.
- Using the Profile Wizard search utility screen as an example the generated SQL behind the scenes (if "Include Subfolders" is selected)
 - SELECT TOP <Max Results> <Select Clause> FROM <From Clause> WITH INCLUDESUBCLASSES WHERE <Where Clause> AND <FROM Clause>.this INSUBFOLDER '<Location>' ORDER BY <Order By Clause>
- Using the Profile Wizard search utility screen as an example the generated SQL behind the scenes (if "Include Subfolder" isn't selected)
 - SELECT TOP <Max Results> <Select Clause> FROM <From Clause> WHERE <Where Clause> AND <FROM Clause>.this INFOLDER '<Location>' ORDER BY <Order By Clause>
- Refer to the relational queries section on the IBM InfoCenter documentation for FileNet P8 for help with configuring advanced where clauses.

http://www.ibm.com/support/knowledgecenter/SSNW2F_5.2.1/com.ibm.p8.ce.dev.ce.doc/query_sql_syntax_rel_queries.htm?cp=SSNW2F_5.2.1%2F10-2-1-34-4-1&lang=en

General Documentum comments:

If not using paged search results and the number of results exceeds the default max search results you will only get that many results and you won't be able to see the other results.

- Using the Profile Wizard search utility screen as an example the generated SQL behind the scenes (if "Include Subfolders" is selected)
 - `SELECT <Select Clause> FROM PUBLIC <From Clause> WHERE <Where Clause> AND FOLDER ('<Location>',DESCEND) ORDER BY <Order By Clause> enable (return_top <Max Results>)`
- Using the Profile Wizard search utility screen as an example the generated SQL behind the scenes (if "Include Subfolder" isn't selected)
 - `SELECT <Select Clause> FROM PUBLIC <From Clause> WHERE <Where Clause> AND FOLDER ('<Location>') ORDER BY <Order By Clause> enable (return_top <Max Results>)`
- Refer to the Documentum DQL reference manual for help with configuring advanced where clauses.

General Box comments:

- Currently paged search results are only supported when using a where clause.
- If using the where clause the max search results can be a maximum of 200.
- If using the select clause the max search results is 1000 per folder.
- Adjusting the metadata templates requires restarting the DocuFlow services and Profile Wizard for the changes to be visible.
- Updated metadata can take upwards of 10 minutes before a search using a where clause will find it.
- Locations can be specified either using the exact folder number or via a relative path with / being the root. If the location is specified via a relative path the folder names can take upwards of 10 minutes before a search using a relative path in the location field will find it.
- Searches can either be performed by using a where clause (and nothing else) or by using a select clause and optional location and optional max results.
- When searching via select clause (instead of a where clause) there is a performance hit when searching with a specified from clause since documents are just listed out by folders and specifying metadata template information to query requires each document to be read from a folder listing. Use searches with a select clause and from clause sparingly if possible. In those cases using a where clause search is advisable.
- Refer to the Box Content API for the search api call for help in configuring advanced searches using the where clause.
<https://docs.box.com/reference#searching-for-content>

4.3.4.2. CMS_API_CREATE

Functionality	Description	FileNet Notes	Documentum Notes	Box Notes	O365/SharePoint Notes
IV_CM_SP_RO_FIL_E	Required: Determines which configured CMS Profile in docuflow SAP to utilize. Combination of CMS Profile and CMS Tag determine the API destination and thus load across an RFC destination.				
IV_CM_ST_AG	Required: Determine which configured CMS Profile in docuflow SAP to utilize. Combination of CMS Profile and CMS Tag determine the API destination and thus load across an RFC destination.				
IV_REPOSITORY	Required: Determines which configured repository in docuflow SAP to utilize	Object Store	Docbase	Default is the only Store/Repository supported since Box currently doesn't support the notion of different repositories.	Site, site collection
IV_TYPE	Optional: what CMS type to create.	DOCUMENT CLASS Defaults to the "Document" document class if unspecified.	OBJECT TYPE Defaults to the "dm_document" class if unspecified.	METADATA TEMPLATE Defaults to creating a document with no metadata template if not specified.	CONTENT TYPE Defaults to creating a document of content type Document if not specified.
IV_US_E_MA	Optional: use mapping of IV_TYPE to CMS type in Enterprise Connector if CMS. Used if ArchiveLink doctype name to CMS type mapping is required.				

Functionality	Description	FileNet Notes	Documentum Notes	Box Notes	O365/SharePoint Notes
PPI NG					
IV_ID	Optional: id to use for versioning a document. When specified a new version of the existing id is created with the given content.	GUID	r_object_id	id	Versioning not supported
IV_FILENAME	Optional: the filename of the content being added to the CMS. If omitted some systems allow for a contentless document to be added.	Used for the "content element" as well as the "DocumentTitle" unless "DocumentTitle" is set in the metadata which will override this. May be empty for a contentless document.	Used as the "object name" as well as the "title" metadata property unless title is set in the metadata which will override this. If the filename is empty then the "title" is set to the "object id." May be empty for a contentless document.	Used as the "name" metadata property unless "name" is set in the metadata which will override this. Box requires that a filename and subsequent content be passed in since currently there is no support for a contentless document.	Used as the "LinkFilename" metadata property
IV_CHECKIN_AS_MIN	Optional: performs the check-in as a minor version.			Not supported	Not supported
IV_CHECKOUT	Optional: allows you to checkout a document prior to creating a new version of the document.	Only available with new versions.	Only available with new versions.	Checks out the document prior to creation and cancels the checkout after the creation has occurred to help guarantee no other person or process modifies the document while the creation call is happening.	Not supported

Functionality	Description	FileNet Notes	Documentum Notes	Box Notes	O365/SharePoint Notes
IV_FINDCURRENTID	Optional: allows you to find the latest version from the given id (since the id might not be the latest version of the document).			Not supported since currently Box ids are the same for every version.	Not supported since currently SharePoint ids are the same for every version.
IV_LOCALTION	Optional: (Mandatory for O365/SharePoint) allows you to specify where to add the document in the content management system.	If unspecified documents are left unfiled.	If unspecified documents are stored in root of the users cabinet.	A number for the exact folder location can be specified. If unspecified documents are stored in the configured root. Relative folder names can be used with a / being the root. If a root folder is configured for the Box CMS in the Enterprise connector all relative folder locations will be relative to the root folder configured. Relative folder names are not recommended for performance reasons.	This is required and is the Document Library root
IT_FOLDER_METADATA	Optional: metadata will be applied at the folder level instead of the file level.	Not supported	Not supported	Applies metadata at the top level folder only	Applies metadata at the top level folder only
IV_FOLDERERNE	Optional: set to X if the use of CT_CUSTOM you choose to use nested folders.	Not supported	Not supported	X = nested folders	Not supported

Functionality	Description	FileNet Notes	Documentum Notes	Box Notes	O365/SharePoint Notes
STED					
IT_METADATA	Optional: metadata to set during creation. Some metadata cannot be set during creation and is only read only (some system properties). Updating some system properties in certain repositories may lead to unknown results. The docuflow username and timestamp properties are automatically set as metadata during creation if they are configured on the particular CMS tab in the docuflow Enterprise Connector. When the same keys are used multi valued properties are set with all of the specified values. The ZMULTI flag is ignored during a create since the same key signifies a multi valued property by default.	To assist with the date format for FileNet, you may choose to utilize the DocuFlow helper method <code>'/dflow/cl_utilities_helper⇒format_date'</code> to format a system date field to the format desired	To assist with the date format for Documentum, you may choose to utilize the DocuFlow helper method <code>'/dflow/cl_utilities_helper⇒format_date'</code> to format a system date field to the format desired	Currently doesn't support multi valued metadata. Where a date field is part of the metadata being sent, the format must be MM/dd/yyyy hh:mm:ss a z, where a z = the AM/PM and TIMEZONE example: 12/31/9999 12:00:00 PM MST To assist with the date format for Box, you may choose to utilize the DocuFlow helper method <code>'/dflow/cl_utilities_helper⇒format_date_plus_sys_timezone'</code> to format a system date field to the format desired and also appending the system timezone	Currently doesn't support multi valued metadata.
IT_CONFIG	Optional if CMS supports folder creation and 1 or more folder are specified				
EVENT_RETURN_ID	The id of the document added to the CMS.				
RETURN_TABLE	Table that indicates a success of an error. The TYPE field will contain an 'S' or 'E' respectively and the MESSAGE field will contain information in the case of an error. When you use the IT_CUSTOM folder to create folders, the return table will also pass back the folder ID's created. The				

Functionality	Description	FileNet Notes	Documentum Notes	Box Notes	O365/SharePoint Notes
	parent folder is indicated by a ! exclamation point, all sub-folders are indicated by a ~ tild. Example of return table when folders are created: <pre>!SAP_4500017723,147811962292 ~SAP_SubFolder1,938382828 ~SAP_SubFolder2,938382829 ~SAP_SubFolder3,938382830</pre> Refer to the below examples for more return table samples.				
CT_CUSTOM	Optional: If supported by your CMS repository, the list of folder(s) to create. This requires the BADI /DFLOW/DOCMANGER_STOREANDLINK to be implemented. You have two choices: 1. Flat Folder Structure 2. Nested Folder Structure For nested, you must set the flag IV_FOLDER_NESTED The first line is the parent folder, lines 2 and above are subfolders (optional). By default, docuflow places the content (when applicable) in the first folder/first line (when flat) or top level folder (when nested). You can override this by placing a ~ tild preceeding the folder name. Example 1: <pre>SAP_PO_HEADER_4500012349 SAP_PO_LINE_00010 SAP_PO_LINE_00020</pre> (content will be placed in first/top folder by default) Example 2: <pre>SAP_PO_HEADER_4500012349 SAP_PO_LINE_00010</pre>				Not supported

Functionality	Description	FileNet Notes	Documentum Notes	Box Notes	O365/SharePoint Notes
	~SAP_PO_LINE_00020 (content will be placed in last/lowest sub-folder) *note: the flag CS_ZDMC01-ZOWNFOLDER must be set for folders to be created. This can be set either in the docManager rule or directly via the BADI.				

General Box comments:

- It is recommended to configure the "error on duplicate" in the docuflow Enterprise Connector since automatically versioning of documents with the same name can produce unpredictable results.
- Locations can be specified either using the exact folder number or via a relative path with / being the root. If the location is specified via a relative path the folder names can take upwards of 1-2 minutes before a create using a relative path in the location field will find it.
- General cleanup utilities are present in Box so that if an error happens when creating a document the document is attempted to be permanently deleted if permissions allow it. Otherwise subsequent cleanup may be required for failed documents. For instance if setting the metadata fails but creation of the document succeeds. They are handled via separate calls to Box so grouping them into a single transaction that can be rolled back is not possible.
- By default (for performance reasons), docuflow comes with the updating of Box system properties (referred to as Basic Metadata) turned OFF. This affects both the CREATE and UPDATE API's. If you want to update Basic Metadata (note the any metadata templates and properties you create on your own are considered Custom Metadata, of which is always enabled), you will need to change the config.properties property box.basicMetadataUpdate=YES. Refer to the section "Adjusting Box Rate Limit and Basic Metadata" in the Base Installation and Configuration guide for more detail.

Example

Below is an example call from storing a document via Services for Object. Place a breakpoint at

ABAP Source Code Display

```

825 lr_api->cms_api_create(
826     EXPORTING
827         iv_cmsprofile = ls_zdmc01-zcmsprofile
828         iv_cmstag     = ls_zdmc01-zcmstag
829         iv_repository = ls_zdmc05-zrepository
830         "iv_id        = p_id
831         iv_filename   = lv_filename
832         iv_size       = lv_filesize
833         iv_type       = ls_zdmc01-ztype
834         "iv_use_mapping = p_map
835         "iv_checkin_as_min = p_cimin
836         "iv_checkout   = p_co
837         "iv_find_curfrom_id = p_fcur
838         iv_location    = ls_zdmc01-zcmslocation
839         "iv_date_time_fmt = lc_dtfmt
840         iv_folder_metadata = ls_zdmc01-zenable_metadataatransfer_fol
841         it_metadata     = lt_metadata
842         it_content      = lt_content
843     IMPORTING
844         ev_returnid    = ev_arc_doc_id
845         et_return      = lt_return
846     CHANGING
847         ct_custom      = lt_custom
848 ).
  
```

Variable Explorer

St.	Variable	Va.	Val.
	GV_SLINK_OBJ		
	LS_ZDMC01-ZCMSPROFI		Box_VF1
	LS_ZDMC01-ZCMSTAG		docuflow VHC1
	LV_FILENAME		Sample Contract Amendmen
	LV_FILESIZE		14091
	LS_ZDMC01-ZTYPE		purchaseOrder
	LS_ZDMC01-ZCMSLOCAT		150020108595
	LS_ZDMC01-ZENABLE_M		
	LT_METADATA		[6x3(2150)]Standard Table
	LT_CONTENT		[14x1(1024)]Standard Tab
	EV_ARC_DOC_ID		884254634276
	LT_RETURN		[2x14(1088)]Standard Tab
	LT_CUSTOM		[1x1(100)]Standard Table

Metadata passed

Table:

Attributes: Standard [6x3(2150)]

Insert Column:

Columns ...

Row	ZMULTI [C(1)]	ZKEY [C(50)]	ZVALUE [C(1024)]
1		poNumber	4500017723
2		linkedToSap	True
3		companyCode	3000
4		paymentTerm	ZB01
5		vendorNumber	0000003740
6		vendorName	POS Paper.com

Custom table passed

In this example, we instructed docuflow to create a 1 level subfolder. This is done by filling the CUSTOM table with the folder name:

Tables

Table Contents

Table

LT_CUSTOM

Attributes

Standard [1x1(100)]

Insert Column

Columns ...

Row	ZCUSTOM [C(50)]
1	SAP_4500017723

Return table received

note: in this return table, we see that we also have docuflow configured to create a folder. When this occurs, the '!' indicates the folder name and the second value is the folder ID.

4.3.4.3. CMS_API_READ

Functionality	Description	FileNet Notes	Docu-ment Notes	Box Notes	O365/SharePoint Notes
IV_C MSP ROFI LE	Required: Determines which configured CMS Profile in docuflow SAP to utilize. Combination of CMS Profile and CMS Tag determine the API destination and thus load across an RFC destination.				
IV_C MST AG	Required: Determine which configured CMS Profile in docuflow SAP to utilize. Combination of CMS Profile and CMS Tag determine the API destination and thus load across an RFC destination.				
IV_RE POSI TORY	Required: Determines which configured repository in docuflow SAP to utilize	Object Store	Docbase	Default is the only Store/Repository supported since Box currently doesn't support the notion of different repositories.	Site, site collection
IV_TY PE	Optional: what CMS type to create.	DOCUMENT CLASS	OBJECT TYPE	METADATA TEMPLATE	CONTENT TYPE
IV_US E_MA PPIN G	Optional: use mapping of IV_TYPE to CMS type in Enterprise Connector if CMS. Used if ArchiveLink doctype name to CMS type mapping is required.				
IV_ID	Optional: id to use for versioning a document. When specified a new version of the existing id is created with the given content.	GUID	r_object_id	id	Versioning not supported
IV_C HEC KOUT	Optional: allows you to checkout a document.	Only available in newer versions.	Only available in newer versions.		Not supported
IV_GE T_CO	Optional: Indicates whether to pass the content of the document back in table ET_CONTENT_MULTI				

Functionality	Description	FileNet Notes	Documentum Notes	Box Notes	O365/SharePoint Notes
NTE NT					
IV_LO CATI ON	Optional: (Mandatory for O365/SharePoint only and not used in any other CMS system) Specifies where to read the document from.	Not utilized	Not utilized	Not utilized	This is required and is the Document Library. For documents created by docuflow, this is stored at the time of create in a docuflow table in the scenario that the config changes and thus can be used for subsequent reads.
IT_M ETAD ATA_ FILTE R	If specified, only metadata properties indicated will be returned. By default of no filter is provided, all metadata for a document is retrieved.				
ET_M ETAD ATA	Contains the metadata retrieved.				
ET_FI LES	Table containing the filename, size, and mime-type of the files retrieved if get content is enabled.				
ET_R ETUR N	Table that indicates a success of an error. The TYPE field will contain an 'S' or 'E' respectively and the MESSAGE field will contain information in the case of an error.				
CT_C UST OM	Future use				

General FileNet comments:

- For performance considerations it's best to limit the read to the particular metadata that is necessary for processing and not retrieve all of the available metadata.

General Documentum comments:

- For performance considerations it's best to limit the read to the particular metadata that is necessary for processing and not retrieve all of the available metadata.

General Box comments:

- For performance considerations it's best to limit the read to the particular metadata that is necessary for processing and not retrieve all of the available metadata.
- Currently VersionSeries isn't supported with Box since there is currently no unique id that represents all versions of a document.
- FoldersFiledIn only returns the parent folder for a Box document.

Example

docuflow has a test harness that is located under the API Gateway. You can choose to either:

1. Single click on a gateway entry and choose the menu path Execute→Read Again. This will populate values into the test harness from the line you have selected
2. Choose the the menu path Execute→Read New. This provide a blank test harness.

To set a break-point to see how this call is executed, the program the test harness uses for READ is:

Program SE38: /DFLOW/ZCMSAPI_READ_01

Below is a sample call:

docuflow Content API: Read

CMS Profile	Box_VF1
CMS Tag	docuflow VHC1
Repository	Default
CMS ID	874177989172
Type	purchaseOrder
☐ Use Mapping	
☐ Check Out	
☐ Get Content	
Location	
Date/Time Format	
Metadata Filter 1	
Metadata Filter 2	
Metadata Filter 3	
Metadata Filter 4	
Metadata Filter 5	
Filepath to save to	

4.3.4.4. CMS_API_UPDATE

Functionality	Description	FileNet Notes	Documentum Notes	Box Notes	O365/SharePoint Notes
IV_CMS PROFILE	Required: Determines which configured CMS Profile in docuflow SAP to utilize. Combination of CMS Profile and CMS Tag determine the API destination and thus load across an RFC destination.				
IV_CMS TAG	Required: Determine which configured CMS Profile in docuflow SAP to utilize. Combination of CMS Profile and CMS Tag determine the API destination and thus load across an RFC destination.				
IV_REPOSITORY	Required: Determines which configured repository in docuflow SAP to utilize	Object Store	Docbase	Default is the only Store/Repository supported since Box currently doesn't support the notion of different repositories.	Site, site collection
IV_TYPE	Optional: what CMS type to update.	DOCUMENT CLASS	OBJECT TYPE	METADATA TEMPLATE	CONTENT TYPE
IV_US_E_MA_PPI_NG	Optional: use mapping of IV_TYPE to CMS type in Enterprise Connector if CMS. Used if ArchiveLink doctype name to CMS type mapping is required.				
IV_CHECKOUT	Optional: allows you to checkout a document.	Only available in newer versions.	Only available in newer versions.		Not supported

Functionality	Description	FileNet Notes	Documentum Notes	Box Notes	O365/SharePoint Notes
IV_CANCEL_CHECKOUT	Optional: allows you to cancel the checkout of a document. This will return an error if the document is already checked out.				
IV_LOCALE_UPDATE	Optional: (Mandatory for O365/SharePoint only and not used in any other CMS system) Specifies where to document to update resides.	Not utilized.	Not utilized.	Not utilized.	This is required and is the Document Library. For documents originally created by docuflow, this is stored at the time of create in a docuflow table in the scenario that the config changes and thus can be used for subsequent updates.
IV_FOLDER_METADATA	Optional: If supported by your CMS, setting this flag to true will apply metadata at the folder level. The ID's located in the IT_IDS table will be folder IDs.			Applied metadata at the top level folder only	Applied metadata at all levels of folders
IT_IDS	Table of file ID's (or folders ID's if supported by your CMS)				

Functionality	Description	FileNet Notes	Documentum Notes	Box Notes	O365/SharePoint Notes
IT_METADATA	Mandatory: metadata to set during updates. Some metadata cannot be set during update and is only read only (some system properties). Updating some system properties in certain repositories may lead to unknown results. The docuflow username and timestamp properties are automatically set as metadata during update if they are configured on the particular CMS tab in the docuflow Enterprise Connector. When the same keys are used multi valued properties are set with all of the specified values. The ZMULTI flag is ignored during an update since the same key signifies a multi valued property by default.	To assist with the date format for FileNet, you may choose to utilize the DocuFlow helper method <code>'/dflow/cl_utilities_helper⇒format_date'</code> to format a system date field to the format desired	To assist with the date format for Documentum, you may choose to utilize the DocuFlow helper method <code>'/dflow/cl_utilities_helper⇒format_date'</code> to format a system date field to the format desired	Currently doesn't support multi valued metadata. Where a date field is part of the metadata being sent, the format must be MM/dd/yyyy hh:mm:ss a z, where a z = the AM/PM and TIMEZONE example: 12/31/9999 12:00:00 PM MST To assist with the date format for Box, you may choose to utilize the DocuFlow helper method <code>'/dflow/cl_utilities_helper⇒format_date_plus_sys_timezone'</code> to format a system date field to the format desired and also appending the system timezone	Supports multi valued metadata for Lookup, Choices and multi line of text.
ETURN	Table that indicates a success of an error. The TYPE field will contain an 'S' or 'E' respectively and the MESSAGE field will contain information in the case of an error.				
CUSTOM	Future use				

4.3.4.5. CMS_API_DELETE

Functionality	Description	File Notes	Document Notes	Box Notes	O365/SharePoint Notes
IV_C MSP ROFI LE	Required: Determines which configured CMS Profile in docuflow SAP to utilize. Combination of CMS Profile and CMS Tag determine the API destination and thus load across an RFC destination.				
IV_C MST AG	Required: Determine which configured CMS Profile in docuflow SAP to utilize. Combination of CMS Profile and CMS Tag determine the API destination and thus load across an RFC destination.				
IV_R EPO SITORY	Required: Determines which configured repository in docuflow SAP to utilize	Object Store	Docbase	Default is the only Store/Repository supported since Box currently doesn't support the notion of different repositories.	Site, site collection
IV_T YPE	Optional: what CMS type to delete.	DOCUMENT CLASS	OBJECT TYPE	METADATA TEMPLATE	CONTENT TYPE
IV_U SE_MAP PING	Optional: use mapping of IV_TYPE to CMS type in Enterprise Connector if CMS. Used if ArchiveLink doctype name to CMS type mapping is required.				
IV_IDS	Table of file ID's (or folders ID's if supported by your CMS)				
IV_L OCA TION	Optional: (Mandatory for O365/SharePoint only and not used in any other CMS system) Specifies where to delete the document from.	Not utilized.	Not utilized.	Not utilized.	This is required and is the Document Library. For documents originally created by docuflow, this is stored at the time of create in a docuflow table in the scenario that the config changes and thus can be used for subsequent deletes.

Functionality	Description	FileNotes	Document Notes	Box Notes	O365/SharePoint Notes
ET_R ETU RN	Table that indicates a success of an error. The TYPE field will contain an 'S' or 'E' respectively and the MESSAGE field will contain information in the case of an error.				
CT_ CUS TOM	Future use				

4.3.5. Ping and Load API's

Ok, we weren't completely truthful, there are more than 5 API's that made up the docuflow SCRUD platform. Actually it is SCRUDPL. But it doesn't sound the same when you say it.

Here are the two additional API's:

1. [CMS_API_PING](#)
2. [CMS_API_LOAD](#)

4.3.5.1. CMS_API_PING

Functionality	Description	FileNet Notes	Docuflow Notes	Box Notes	0365/SharePoint Notes
IV_CMS_PROFILE	Required: Determines which configured CMS Profile in docuflow SAP to utilize. Combination of CMS Profile and CMS Tag determine the API destination and thus load across an RFC destination.				
IV_CMS_TAG	Required: Determine which configured CMS Profile in docuflow SAP to utilize. Combination of CMS Profile and CMS Tag determine the API destination and thus load across an RFC destination.				
IV_PING_MIDDLEWARE	Checks that status of the connection to the connector only.				
IV_PING_CMS	Checks the status of the connection right through to the CMS system (requires IV_TYPE)				
IV_REPOSITORY	Required: Determines which configured repository in docuflow SAP to utilize	Object Store	Docbase	Default is the only Store/Repository supported since Box currently doesn't support the notion of different repositories.	Site, site collection
IV_TYPE	What CMS type to PING (mandatory for CMS ping)	DOCUMENT CLASS	OBJECT TYPE	METADATA TEMPLATE	CONTENT TYPE
IV_USE_MAPPING	Optional: use mapping of IV_TYPE to CMS type in Enterprise Connector if CMS. Used if ArchiveLink doctype name to CMS type mapping is required.				
ET_RETURN	Table that indicates a success of an error. The TYPE field will contain an 'S' or 'E' respectively and the MESSAGE field will contain information in the case of an error.				

Functionality	Description	FileNet Notes	Docum entum Notes	Box Notes	0365/S harePoi nt Notes
CT_CUS TOM	Future use				

4.3.5.2. CMS_API_LOAD

Functionality	Description	File Name Notes	Document Notes	Box Notes	0365/SharePoint Notes
IV_CMSPRO FILE	Required: Determines which configured CMS Profile in docuflow SAP to utilize. Combination of CMS Profile and CMS Tag determine the API destination and thus load across an RFC destination.				
IV_CMSTAG	Required: Determine which configured CMS Profile in docuflow SAP to utilize. Combination of CMS Profile and CMS Tag determine the API destination and thus load across an RFC destination.				
IV_LOAD_RE POSITORIES	Load list of repositories for a CMS tab and profile				
IV_LOAD_M ETADATA	Load list of metadata available for each repository				
IV_LOAD_TY PESANDPR OPS	Load list of types and their properties for each repository				
IV_LOAD_DA TABASE	Save results to docuflow SAP tables				
IV_LOAD_INI TIALIZE_DB	When Load DB is specified, whether to clear existing docuflow SAP tables to be loaded into				
IV_REPOSIT ORY	Which repositories to load (Mandatory if Load Repositories is specified).				
ET_REPOSIT ORIES	Retrieved list of repositories.				
ET_METADA TA_STRUCT	Retrieved list of metadata				
ET_TYPER SANDPROPS	Retrieved list of types and properties				

Functionalit y	Description	FileNe t Notes	Docume ntum Notes	Box Note s	O365/Sha rePoint Notes
ET_RETURN	Table that indicates a success of an error. The TYPE field will contain an 'S' or 'E' respectively and the MESSAGE field will contain information in the case of an error.				
CT_CUSTO M	Future use				

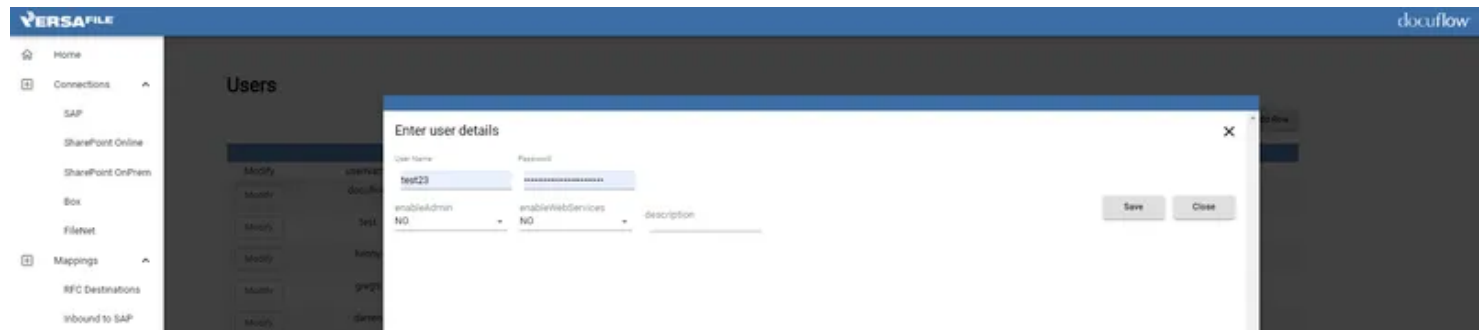
4.3.6. docuflow SCRUD Web Services

4.3.6.1. User Permissions

Every user created in Docuflow has the ability to be restricted to different parts of the product. Currently, docuflow allows users to the following:

- 1) Enable/disable users access to admin interface.
- 2) Enable/disable user access to web services.

To add or remove users visit the admin interface. Click on the users menu item on the left navigation panel. Click add row, and enter the details of the user. Hit save to complete creating the user.



4.3.6.2. HTTPS Setup (JKS/PEM) (cloned)

JKS

Docuflow supports utilizing the JKS format for it's web services API. If you are not familiar with java key stores you can learn more about them [here](#). In most cases inside a keystore a private key is accompanied by the corresponding public key certificate.

To view the certificates currently installed in your docuflow middleware for web services you can view the JKS file using a key store viewer such as <https://keystore-explorer.org/>

Once downloaded you can open the "keystore.jks" file using the password !DFLOW! to open the JKS file.

The JKS file is located at **C:\<docuflow-installation-directory>\config\certificates\keystore.jks**

For information on **trusted certificates** and importing them you can visit the following link <https://keystore-explorer.org/doc/5.4/trustedCertificates.html>

For information on using a generated key-pair certificate you can visit the following link <http://keystore-explorer.org/doc/5.4/keyPairs.html#generate-a-key-pair>

Alternatively you can utilize pem and cer files for encrypting the docuflow web service.

Upon any changes of the JKS the service must be restarted.

PEM/CER

To enable docuflow to use the following formats you must add to the ini file **cert.type = pem** under [General Preferences].

Using a generated private key file in pkcs8 format for the server and place in the config folder of the docuflow instance.

```
* Using openssl and your private key this is created by:
* openssl pkcs8 -topk8 -inform pem -in your_private_key.pem -outform pem -nocrypt -out your_private_key_pkcs8.pem
```

Generating the public certificate for the server. Place in the config folder of the docuflow instance.

```
*
* openssl req -new -x509 -key your_private_key.pem -out your_public.cer -days 3650
*
```

Update Config.properties and remove the default values for content.cer.file and content.pem.file with the ones generaged above. Save the file and restart the service.

content.cer.file=your_public.cer

content.pem.file=your_private_key_pkcs8.pem

4.3.6.3. **Box Endpoints**

4.3.6.3.1. **Upload**

Docuflow provides an endpoint to upload files to. This is strictly for uploading files, and if you require metadata to be transferred following up with an update endpoint call is required.

Upload

Upload a file using HTTP multi part form data. HTTP headers specifying folder id, and the content system named must be specified.

POST

https://localhost:82/scrud/upload

Request

HEADER PARAMETERS

cmsName	String	required
Name of the content management system configured in docuflow.		
location	String	optional
Folder id where docuflow is uploading the file. (box folder id) else root folder used.		
Content-Type	String	optional
multipart/form-data; boundary=(insert boundary)		
Example is included in code examples		

EXAMPLE

```
-----
> POST /scrud/upload HTTP/1.1
> Host: localhost:82
> User-Agent: insomnia/2021.7.2
> Cookie: vertx-web.session=8c1a5313de4babf8825fe2114505da7e
> Content-Type: multipart/form-data; boundary=X-INSOMNIA-BOUNDARY
> location: 154829431337
> cmsName: demoproduct
> Authorization: Basic ZG9jdWZsb3c6ZG9jdWZsb3c=
> Accept: */*
> Content-Length: 3693

| --X-INSOMNIA-BOUNDARY
| Content-Disposition: form-data; name=""; filename="Versafile_4C.png"
| Content-Type: image/png
|  PNG
|  
-----
EXAMPLE
```

SUCCESS EXAMPLE

```
{
  "id": "910645979785",
  "result": "success"
}
```

FAILED EXAMPLE

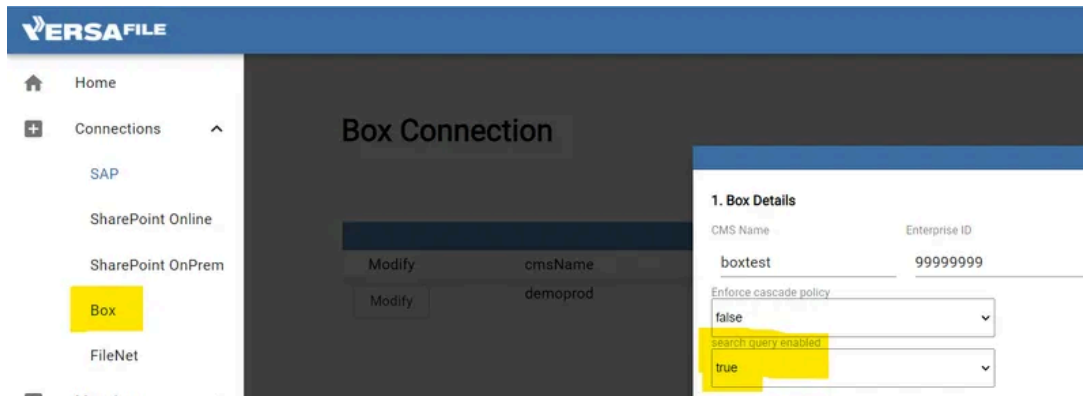
```
{
  "id": null,
  "result": "fail"
}
```

4.3.6.3.2. Search

Docuflow provides a web service endpoint that will allow you to retrieve a subset of documents/folders based on a search criteria. Docuflow supports two types of search within box. One using [metadata filters](#) and another using [metadata queries](#) (real-time).

You can learn about the pros and cons of which one to utilize [here](#).

To enable or disable metadata queries (real-time search) you can do so within the docuflow configurator interface.



We are providing below an example of both a metadata filter example (with search query disabled in the configurator), and a metadata query example (with search query enabled in the configurator).

Search (query)

Search for all documents using a where clause. If succesfull will return a JSON Array with a size greater then or equal to zero with status code of 200. Otherwise status code 500 is returned.

Precondition: insertName.searchQueryEnabled=true (located in inifile)

POST
<https://localhost:82/scrud/search>


Request

BODY PARAMETERS

cmsName

String

required

Name of the box system configured in docuflow.

whereClause

String

required

A where clause (JSON in string form). Used to query box to return a set of files/folders depending on criteria set.

selectedCMS

String

required

Value should always be set to "box" to let docuflow know which configured content system to search.

```
{
  "cmsName": "insertProductionSystemName (located in docuflow gui)",
  "whereClause": "{ \"from\": \"enterprise_InsertEnterpriseId.insertMetaDataTemplate\", \"query\": \"insertMetaDataColumnName = :value\", \"selectedCMS\": \"box\" }",
}
```

```
[
  {
    "name": "check",
    "id": "895736641171",
    "type": "file"
  },
  {
    "name": "demo1.txt",
    "id": "897859484481",
    "type": "file"
  }
]
```

Search (mdfilters)

Search for all files/folders using a where clause. If succesfull will return a JSON Array with a size greater then or equal to zero with status code of 200. Otherwise status code 500 is returned.

POST<https://localhost:82/scrud/search>

Request

BODY PARAMETERS

cmsName

String

required

Name of the box system configured in docuflow.

whereClause

String

optional

A where clause (JSON in string form). Used to query box to return a set of files/folders depending on criteria set.

selectedCMS

String

optional

Value should always be set to "box" to let docuflow know which configured content system to search.

```
{
  "cmsName": "insertProductionSystemName (located in docuflow gui)",
  "whereClause": "mdfilters=[{\"templateKey\": \"insertMetaDataTemplateName\", \"scope\": \"enterprise\", \"filters\": {\"insertMetaData\"}}]",
  "selectedCMS": "box"
}
```

```
[
  {
    "name": "check",
    "id": "895736641171",
    "type": "file"
  },
  {
    "name": "demo1.txt",
    "id": "897859484481",
    "type": "file"
  },
  {
    "name": "Cataray.pdf",
    "id": "901051603311",
    "type": "file"
  },
  {
    "name": "Version3.pdf",
    "id": "901048740163",
    "type": "file"
  },
  {
    "name": "TEST XLS.xls",
    "id": "900741822394",
    "type": "file"
  },
  {
    "name": "TEST HTM.htm",
    "id": "900747633313",
    "type": "file"
  },
  {
    "name": "TEST GIF.gif",
    "id": "900747915499",
    "type": "file"
  }
]
```

4.3.6.3.3. Create

Docuflow provides a web service endpoint that will allow you to create a document or folder.

The following endpoint allows the following capabilities.

1. Create a File
2. Create a File with metadata
3. Create a folder
4. Create a folder with metadata
5. Create a nested folder structure (with parent obtaining metadata)
6. Create a single level tree structure (with parent obtaining metadata)

Create a File

Create a file with no metadata

POST<https://localhost:82/scrud/create>

Request

BODY PARAMETERS

type

String

required

Metadata template name in box.

location

String

required

Folder id where the file will be located in box.

selectedCMS

String

required

Value should always be set to "box" to let docuflow know which configured content system to search.

cmsName

String

required

Name of the box system configured in docuflow.

```
{
  "cmsName": "Name of the box system configured in docuflow.",
  "fileLocation": "C:\\someFolder\\someFile.txt",
  "type": null,
  "location": "Insert unique box folder id",
  "selectedCMS": "box"
}
```

Create a File (metadata)

Create a file with metadata

POST<http://localhost:82/scrud/create>

Request

BODY PARAMETERS

type

String

required

Metadata template name in box.

location

String

required

Folder id where the file will be located in box.

selectedCMS

String

required

Value should always be set to "box" to let docuflow know which configured content system to search.

cmsName

String

required

Name of the box system configured in docuflow.

mdArray

Array

required

A JSON array where each index is a json object with the keys "md1" and "md2". Where md1 is the column name, and md2 is the value to be inserted in that column.

```
{
  "cmsName": "Name of the box system configured in docuflow.",
  "fileLocation": "C:\\someFolder\\someFile.txt",
  "type": "insertMetaDataTemplate",
  "location": "Insert unique box folder id",
  "selectedCMS": "box",
  "mdArray": [
    {
      "md1": "insertColumnName",
      "md2": "insertValue"
    },
    {
      "md1": "someVendor",
      "md2": "VersaFile Inc"
    }
  ]
}
```

Create a Folder

Create a folder with no metadata

POST<https://localhost:82/scrud/create>

Request

BODY PARAMETERS

type

String

required

Metadata template name in box.

location

String

required

Folder id where the file will be located in box.

selectedCMS

String

required

Value should always be set to "box" to let docuflow know which configured content system to search.

cmsName

String

required

Name of the box system configured in docuflow.

custom1

String

required

Name of the folder you would like to create.

```
{
  "cmsName": "Name of the box system configured in docuflow.",
  "type": null,
  "location": "Insert unique box folder id",
  "selectedCMS": "box",
  "custom1": "insert Folder Name"
}
```

Create a Folder (metadata)

Create a Folder with metadata

POST<https://localhost:82/scrud/create>

Request

BODY PARAMETERS

type

String

required

Metadata template name in box.

location

String

required

Folder id where the file will be located in box.

selectedCMS

String

required

Value should always be set to "box" to let docuflow know which configured content system to search.

cmsName

String

required

Name of the box system configured in docuflow.

custom1

String

required

Name of the folder you would like to create.

mdArray

Array

required

A JSON array where each index is a json object with the keys "md1" and "md2". Where md1 is the column name, and md2 is the value to be inserted in that column.

metaDataFolderFlag

Boolean

required

true if setting metadata for folders
false if not setting metadata for folders

```
{
  "cmsName": "Name of the box system configured in docuflow.",
  "fileLocation": "C:\\someFolder\\someFile.txt",
  "type": "insertMetaDataTemplate",
  "location": "Insert unique box folder id",
  "selectedCMS": "box",
  "mdArray": [
    {
      "md1": "insertColumnName",
      "md2": "insertValue"
    },
    {
      "md1": "someVendor",
      "md2": "VersaFile Inc"
    }
  ],
  "custom1": "folderName",
  "metaDataFolderFlag": true
}
```

Create a Folder (metadata and nested)

Create a Folder with metadata

POST<https://localhost:82/scrud/create>

Request

BODY PARAMETERS

type

String

required

Metadata template name in box.

location

String

required

Folder id where the file will be located in box.

selectedCMS

String

required

Value should always be set to "box" to let docuflow know which configured content system to search.

cmsName

String

required

Name of the box system configured in docuflow.

custom1

String

required

Name of the folder you would like to create. If using nested the delimiter is #DFLOW#

mdArray

Array

required

A JSON array where each index is a json object with the keys "md1" and "md2". Where md1 is the column name, and md2 is the value to be inserted in that column.

enableNested

Boolean

required

true (if wanting each folder to have one child in a nested manner)

false (if all children are located under the first folder name)

metaDataFolderFlag

Boolean

required

true if setting metadata for folders

false if not setting metadata for folders


```
{
  "cmsName": "Name of the box system configured in docuflow.",
  "fileLocation": "C:\\someFolder\\someFile.txt",
  "type": "insertMetaDataTemplate",
  "location": "Insert unique box folder id",
  "selectedCMS": "box",
  "mdArray": [
    {
      "md1": "insertColumnName",
      "md2": "insertValue"
    },
    {
      "md1": "someVendor",
      "md2": "VersaFile Inc"
    }
  ],
  "custom1": "folderName#DFLOW#folderName2#DFLOW#folderName3",
  "metaDataFolderFlag": true,
  "enableNested": true
}
```

Create a Folder (metadata and nested false)

Create a Folder with metadata

POST<https://localhost:82/scrud/create>

Request

BODY PARAMETERS

type

String

required

Metadata template name in box.

location

String

required

Folder id where the file will be located in box.

selectedCMS

String

required

Value should always be set to "box" to let docuflow know which configured content system to search.

cmsName

String

required

Name of the box system configured in docuflow.

custom1

String

required

Name of the folder you would like to create. If using nested the delimiter is #DFLOW#

mdArray

Array

required

A JSON array where each index is a json object with the keys "md1" and "md2". Where md1 is the column name, and md2 is the value to be inserted in that column.

enableNested

Boolean

required

true (if wanting each folder to have one child in a nested manner)

false (if all children are located under the first folder name)

metaDataFolderFlag

Boolean

required

true if setting metadata for folders
false if not setting metadata for folders

```
{
  "cmsName": "Name of the box system configured in docuflow.",
  "fileLocation": "C:\\someFolder\\someFile.txt",
  "type": "insertMetaDataTemplate",
  "location": "Insert unique box folder id",
  "selectedCMS": "box",
  "mdArray": [
    {
      "md1": "insertColumnName",
      "md2": "insertValue"
    },
    {
      "md1": "someVendor",
      "md2": "VersaFile Inc"
    }
  ],
  "custom1": "folderName#DFLOW#folderName2#DFLOW#folderName3",
  "metaDataFolderFlag": true,
  "enableNested": false
}
```

Create a Folder (no metadata and nested false)

Create a Folder with metadata

POST<https://localhost:82/scrud/create>

Request

BODY PARAMETERS

type

String

required

Metadata template name in box.

location

String

required

Folder id where the file will be located in box.

selectedCMS

String

required

Value should always be set to "box" to let docuflow know which configured content system to search.

cmsName

String

required

Name of the box system configured in docuflow.

custom1

String

required

Name of the folder you would like to create. If using nested the delimiter is #DFLOW#

mdArray

Array

required

A JSON array where each index is a json object with the keys "md1" and "md2". Where md1 is the column name, and md2 is the value to be inserted in that column.

enableNested

Boolean

required

true (if wanting each folder to have one child in a nested manner)

false (if all children are located under the first folder name)

metaDataFolderFlag

Boolean

required

true if setting metadata for folders

false if not setting metadata for folders

```
{  
  "cmsName": "Name of the box system configured in docuflow.",  
  "fileLocation": "C:\\someFolder\\someFile.txt",  
  "type": null  
  "location": "Insert unique box folder id",  
  "selectedCMS": "box",  
  "custom1": "folderName#DFLOW#folderName2#DFLOW#folderName3",  
  "metaDataFolderFlag": false,  
  "enableNested": false  
}
```

4.3.6.3.4. Read

Docuflow provides a web service endpoint that will allow you to retrieve a file or folders metadata or binary content of a file.

Included below are examples of reading a files metadata and it's binary content. (Two seperate calls)

Read (metadata)

Retrieve a file or folders metadata by using its unique box identifier.

POST<https://localhost:82/scrud/read>

Request

BODY PARAMETERS

cmsName

String

required

Name of the box system configured in docuflow.

type

String

required

Name of metadata template in box.

id

String

required

Unique id of box file or folder.

selectedCMS

String

required

Value should always be set to "box" to let docuflow know which configured content system to search.

getContent

Boolean

optional

true/false (will retrieve binary of file)

```
{  
  "cmsName": "insertProductionSystemName (located in docuflow gui)",  
  "type": "insertMetaDataTemplateName",  
  "id": "unique box id specfic to folder/id",  
  "selectedCMS": "box"  
}
```



```

{
  "ContentElements": "demo1.txt#DFLOW#text/plain#DFLOW#2111",
  "comment_count": "0",
  "companyCode": "",
  "content_created_at": "10/12/2021 12:48:52 p.m. PDT",
  "content_modified_at": "10/12/2021 12:48:52 p.m. PDT",
  "created_at": "12/20/2021 07:29:31 a.m. PST",
  "created_by": "Bhavani Singetam",
  "description": "",
  "docuflowSaplinkEnabled": "",
  "docuflowTimestamp": "",
  "docuflowUsername": "",
  "docuflowtaskid": "15103846926",
  "docuflowtaskstatus": "in-process",
  "docuflowworkflowid": "",
  "embed1": "https://versafire-box.app.box.com/preview/expiring_embed/gvact6FE!hc4EhSwuz8Zl-y-Sf80bT25vPui0lkTN3P_G11ELtgvcTHI4XYWYP-",
  "embed2": "Kne3s4qBnDfTAL8DoHZoq7qssnDf9P-aznmofY-lZnZ0Bk8oek5Qcgf_NChMzkPyNgAT6_RSgwhMNgGh7hUuNMN0dak5_xs-bZANve68DtQwwBqwwG6kGydr",
  "etag": "1",
  "expiring_embed_link": "https://versafire-box.app.box.com/preview/expiring_embed/gvact6FE!hc4EhSwuz8Zl-y-Sf80bT25vPui0lkTN3P_G11ELt",
  "extension": "txt",
  "fileVersion": "967518965567",
  "file_version": "967518965567",
  "has_collaborations": "true",
  "id": "897859484481",
  "is_package": "false",
  "item_status": "active",
  "lastUpdated": "1640195853",
  "linkedToSap": "",
  "lock": "",
  "modified_at": "12/22/2021 09:57:30 a.m. PST",
  "modified_by": "someone",
  "name": "demo1.txt",
  "owned_by": "docuflow Demo",
  "parent": "1474026705828",
  "path_collection": "/test",
  "paymentTerm": "",
  "permissions": "can_download:true,can_preview:true,can_upload:true,can_comment:true,can_rename:true,can_delete:true,can_share:true,",
  "poNumber": "5900",
  "poStatus": "",
  "purged_at": "",
  "sequence_id": "1",
  "sha1": "013efd1020dea4fcdc4f9118433acc8ded3bf895",
  "shared_link": "",
  "size": "2111",
  "tags": "",
  "trashed_at": "",
  "type": "file",
  "vendorName": "",
  "vendorNumber": "",
  "version_number": "2"
}

```

Read (Binary content)

Retrieve a files binary by using its unique box identifier.

POST

https://localhost:82/scrud/read

Request

BODY PARAMETERS

cmsName	String	required
Name of the box system configured in docuflow.		
type	String	required
Name of metadata template in box.		
id	String	required
Unique id of box file or folder.		
selectedCMS	String	required
Value should always be set to "box" to let docuflow know which configured content system to search.		

```
{
  "cmsName": "insertProductionSystemName (located in docuflow gui)",
  "type": "insertMetaDataTypeName",
  "id": "unique box id specific to folder/id",
  "selectedCMS": "box",
  "getContent": true
}
```

```
...binary
```

4.3.6.3.5. Update

Docuflow provides a web service endpoint that will allow you to update a document or folders metadata.

Update File/Folder metadata

Update a file or folder metadata

POST<https://localhost:82/scrud/update>

Request

BODY PARAMETERS

id

String

required

Unique id of box file or folder.

cmsName

String

required

Name of the box system configured in docuflow.

selectedCMS

String

required

Value should always be set to "box" to let docuflow know which configured content system to search.

type

String

required

Name of metadata template in box.

metaDataFolderFlag

Boolean

required

true if setting metadata for folders
false if not setting metadata for folders

mdArray

Array

required

A JSON array where each index is a json object with the keys "md1" and "md2". Where md1 is the column name, and md2 is the value to be inserted in that column.

```
{
  "cmsName": "insertProductionSystemName (located in docuflow gui)",
  "type": "insertMetaDataTemplateName",
  "id": "insert box folder id",
  "mdArray": [
    {
      "md1": "someColumnToBeUpdated",
      "md2": "newValue"
    }
  ],
  "selectedCMS": "box",
  "metaDataFolderFlag": true
}
```

4.3.6.3.6. Delete

Docuflow provides a web service endpoint that will allow you to delete a file.

Delete a file.

Delete a File

POST<https://localhost:82/scrud/delete>

Request

BODY PARAMETERS

id	String	required
id of the document		

cmsName	String	required
Name of the box system configured in docuflow.		

selectedCMS	String	required
Value should always be set to "box" to let docuflow know which configured content system to search.		

```
{  
  
  "cmsName": "insertProductionSystemName (located in docuflow gui)",  
  "id": "127595030790",  
  "selectedCMS": "box"  
}
```

4.3.6.3.7. Tasks

Currently docuflow has an endpoint to create tasks on top of files and assign them to a user (among other things).

Create Tasks

Create a general task on a file

POST<https://localhost:82/scrud/createTask>

Request

BODY PARAMETERS

id(ZID)

String

required

Unique id of box file or folder.

type(ZTYPE)

String

required

Name of metadata template in box.

selectedCMS

String

required

Value should always be set to "box" to let docuflow know which configured content system to search.

body(ZTEXTBODY)

Array

required

Json Array of strings (each index being a seperate line).

Or can take a single string value "singleLineExample"

dueDate(ZDUEDATE)

String

required

Date the task is due

2100-12-31 (YYYY-MM-DD)

recipients(ZRECIPIENTS)

Array

required

Array of strings of recipients of the task (email id)

*User may require permissions to file before being able to be assigned a task.

taskType(ZTASKTYPE)

String

required

Type of task assigned. For a General Task, task Type is "COMPLETE".


```
{
  "cmsName": "insertProductionSystemName (located in docuflow gui)",
  "id": "insert box file id",
  "taskType": "insert Task type (described in params)",
  "recipients":
    [
      "admin@versafire.com",
      "admin2@versafire.com"
    ],
  "body": [
    "Hi",
    "Thank you."
  ],
  "dueDate": "2021-12-30",
  "selectedCMS": "box",
}
```

Create Tasks

Create a review task on a file

POST<https://localhost:82/scrud/createTask>

Request

BODY PARAMETERS

id(ZID)

String

required

Unique id of box file or folder.

type(ZTYPE)

String

required

Name of metadata template in box.

selectedCMS

String

required

Value should always be set to "box" to let docuflow know which configured content system to search.

body(ZTEXTBODY)

Array

required

Json Array of strings (each index being a seperate line).

Or can take a single string value "singleLineExample"

dueDate(ZDUEDATE)

String

required

Date the task is due

2100-12-31 (YYYY-MM-DD)

recipients(ZRECIPIENTS)

Array

required

Array of strings of recipients of the task (email id)

*User may require permissions to file before being able to be assigned a task.

taskType(ZTASKTYPE)

String

required

Type of task assigned. For a review task, task type is "REVIEW"

```
{
  "cmsName": "insertProductionSystemName (located in docuflow gui)",
  "id": "insert box file id",
  "taskType": "insert Task type (described in params)",
  "recipients":
    [
      "admin@versafire.com",
      "admin2@versafire.com"
    ],
  "body": [
    "Hi",
    "Thank you."
  ],
  "dueDate": "2021-12-30",
  "selectedCMS": "box",
}
```

4.3.6.3.8. Sign (available version > 6.6.5)

Currently docuflow has an endpoint to create a request to sign documents and send them to multiple users to sign the file.

Note: If the document assigning sign request is in a specific folder under root. Specify the root folder id in box Configuration.

Sign Documents

Create a sign request on a document

POST<https://localhost:82/scrud/createTask>

Request

BODY PARAMETERS

id String **required**
Unique id of box file or folder.

type String **required**
Name of metadata template in box.

selectedCMS String **required**
Value should always be set to "box" to let docuflow know which configured content system to search.

body Array **required**
Json Array of strings (each index being a separate line).

Or can take a single string value "singleLineExample"

dueDate String **required**
Number of days from today the sign request is valid for. Example: 5

recipients Array **required**
Array of strings of recipients of the task (email id)

*User may require permissions to file before being able to be assigned a sign request.

enableSameDestination Boolean **optional**
If true, signed document and signing log is stored in the same location as source file (destination id auto filled is generated by middleware).

If false, Signed Document and signing log is stored in the destination location.

destinationId String **required**
location where the signed document and signing log is stored (Only when enableSameDestination is false)

taskType String **required**
Task Type of a sign request is "SIGN"

```
{  
  "cmsName": "demoprod",  
  "id": "917235734365",  
  "taskType": "SIGN",  
  "recipients": [  
    "bhavanis@versafire.com"  
  ],  
  "destinationId": "156906751612",  
  "type": "purchaseOrder",  
  "selectedCMS": "box",  
  "dueDate": "1",  
  "enableSameDestination": false  
}
```

Sign Documents

Use case scenario: Store signed document and signing log where original document is.

When the enableSameDestination is set to true the signed document and signing log will be stored in the same location as source file (i.e; the parent folder of id)

POST<https://localhost:82/scrud/createTask>

Request

BODY PARAMETERS

id

String

required

Unique id of box file or folder.

type

String

required

Name of metadata template in box.

selectedCMS

String

required

Value should always be set to "box" to let docuflow know which configured content system to search.

body

Array

required

Json Array of strings (each index being a seperate line).

Or can take a single string value "singleLineExample"

dueDate

String

required

Number of days from today the sign request is valid for.

recipients

Array

required

Array of strings of recipients of the task (email id)

*User may require permissions to file before being able to be assigned a sign request.

enableSameDestination

Boolean

optional

If true, signed document and signing log is stored in the same location as source file (destination id auto filled is generated by middleware).

If false, Signed Document and signing log is stored in the destination location.

destinationId

String

required

location where the signed document and signing log is stored (Only when enableSameDestination is false)

taskType

String

required

Task type of a sign request is "SIGN"

```
{
  "cmsName": "demoprod",
  "id": "917235734365",
  "taskType": "SIGN",
  "recipients": [
    "bhavanis@versafire.com"
  ],
  "destinationId": "156906751612",
  "type": "purchaseOrder",
  "selectedCMS": "box",
  "dueDate": "1",
  "enableSameDestination": false
}
```


Sign Documents

Use case scenario: Multiple Sign Recipients.

The request can be sent to multiple signers. If one signer declines the request the whole request will be cancelled.

POST<https://localhost:82/scrud/createTask>

Request

BODY PARAMETERS

id

String

required

Unique id of box file or folder.

type

String

required

Name of metadata template in box.

selectedCMS

String

required

Value should always be set to "box" to let docuflow know which configured content system to search.

body

Array

required

Json Array of strings (each index being a separate line).

Or can take a single string value "singleLineExample"

dueDate

String

required

Number of days from today the sign request is valid for.

recipients

Array

required

Array of strings of recipients of the task (email id)

*User may require permissions to file before being able to be assigned a sign request.

enableSameDestination

Boolean

optional

If true, signed document and signing log is stored in the same location as source file (destination id auto filled is generated by middleware).

If false, Signed Document and signing log is stored in the destination location.

destinationId

String

required

location where the signed document and signing log is stored (Only when enableSameDestination is false)

taskType

String

required

Task Type of a sign request is "SIGN"

```
{
  "cmsName": "demoprod",
  "id": "917235734365",
  "taskType": "SIGN",
  "recipients": [
    "bhavanis@versafire.com",
    "singetambhavani7@gmail.com"
  ],
  "destinationId": "156906751612",
  "type": "purchaseOrder",
  "selectedCMS": "box",
  "dueDate": "1",
  "enableSameDestination": false
}
```

4.3.7. docuflow BAPI Web Services

Docuflow provides two end points to access SAP BAPI's. Using the GET request on the "any" endpoint we can retrieve what data is available to the developer. Docuflow BAPI web services use basic authorization with a default username: **docuflow**, and a default password: **docuflow**. Ensure SSL cert validation is disabled prior to accessing the web services.

The following examples can be imported into Insomnia Rest Client via this link:

<https://docuflow.box.com/s/em0ckv2zarcai15cr5az6rewxvfw18p5>

For example using a get request,

https://localhost:82/bapi/any?bapi=BAPI_INCOMINGINVOICE_CREATE

would return the following data below:

<https://jsoneditoronline.org/#left=cloud.b9e0a63872034234850ac9de5709b564>

We can use this response to learn what all the available fields/data are available in that particular bapi.

To send data to BAPI_INCOMINGINVOICE_CREATE we would send a post request to <https://localhost:82/bapi/any> using the json body included in this link:

<https://jsoneditoronline.org/#left=cloud.dbd7f72e2ade41eab9301c17081b2a04>

You should see the following json response here

<https://jsoneditoronline.org/#left=cloud.73c35d070cb546f683dc05e82865cac1>

Notice in the export parameters the response given from the BAPI that was executed

```

▼ EXPORT_PARAMETERS [2]
  ▼ 0 {3}
    NAME : FISCALYEAR
    TYPE : NUM
    VALUE : 2021
  ▼ 1 {3}
    NAME : INVOICEDOCNUMBER
    TYPE : CHAR
    VALUE : 5105609715

```

Another example is given here with BAPI_PO_CHANGE using post.

<https://jsoneditoronline.org/#left=cloud.b9a0aeb968344aeb94b1153ca69b1959>

and it's corresponding response,

<https://jsoneditoronline.org/#left=cloud.d6ef93f919e443d7a8153c1fd6b856ec>

Common tables to look at for response from SAP are the **return** table, and the **export parameters**.

4.3.8. Useful / Common docuflow Methods

Below are some common docuflow methods used by our customers in their custom requirements.

Feel free to use them!

1. Store and Link a Document and/or Create Folders: `/dflow/cl_docmanager_helper⇒store_and_link_document`
2. Send Metadata: `/dflow/cl_docmanager_helper⇒send_metadata_update`
3. Create an ArchiveLink Link: `/dflow/cl_utilities_helper⇒create_archivelink_link`
4. View an attachment that has an ArchiveLink entry: `/dflow/cl_utilities_helper⇒display_image`

4.3.8.1. Store and Link a Document

Storing and linking a document is an extremely common functionality.

Storing be done by either providing content via a filename to upload or by providing the binary content itself.

The docuflow store and link method can also create folder structures (with and without content...note content is stored in the top level folder), and in that scenario optionally apply metadata to the top level folder as well (to apply metadata to subfolders or documents, use the method `/dflow/cl_docmanager_helper⇒send_metadata_update`. You can also choose which level of multi-folder heirachy you create in which to place content if applicable.

`/dflow/cl_docmanager_helper⇒store_and_link_document`

Parameters

Parameter	Description	Sample Value
IV_CALLER	Describes who is calling the method. Logged in the docuflow API Gateway as such.	/dflow/cl_common_super⇒ gc_caller_docmanager_custom
IV_MODE	Create can be passed file content (C), the name and path of a file to upload and store (F), as well just create folder with no content (N).	'C' = Content 'F' = File 'N' = No Content
IV_SAP_OBJECT	ArchiveLink SAP Object	BKPF
IV_ARCHIVE_OBJECT	ArchiveLink doctype	ZINVOICE01
IV_ARCHIVE_OBJECT_ID	ArchiveLink SAP key. Note this can be a compound key for certain SAP objects or line item/company code level.	300019000000012020
IV_FILE_WITH_PATH	Determine File, Content, or Contentless	'C' = Content 'F' = File 'N' = No Content
IV_NO_CREATE_LINK	Optional: does NOT create a link after successfull document storage	
IV_NO_METADATA_TRANSFER	Optional: Overrides the docManager rule of whether to send metadata	
IV_SIZE	Mandatory if File or Content mode. Size of content in bytes.	
IV_RFC	Optional: Name of RFC to override docManager rule	
IV_FOLDER_METADATA	Optional: Boolean to indicate if update is to a folder. Note that this is applied to the top level folder only.	X = folder metadata EMPTY = file metadata
IT_METADATA	Table of key value pairs of your CMS properties and the values to update	sapCompanyCode 3000
IT_CONTENT	Optional (if contentless or file mode). Table of binary content.	

Parameter	Description	Sample Value
IT_SA PLINK _MET ADAT A	Optional: no used at this point - future functionality	
IT_CU STOM	Optional: If supported by your CMS repository, the list of folder(s) to create. The first line is the parent folder, lines 2 and above are subfolders (optional). By default, docuflow places the content (when applicable) in the top level folder (ie: the first line). You can override this by placing a ~ tild preceeding the folder name. Also by default, the structure will be created as 1 parent and the rest subfolders of that parent. If you wish to create a nested folder structure, set the flag IV_FOLDER_NESTED to X.	SAP_PO_HEADER_4500012349 SAP_PO_LINE_00010 SAP_PO_LINE_00020 (content will be placed in top folder) SAP_PO_HEADER_4500012349 SAP_PO_LINE_00010 ~SAP_PO_LINE_00020 (content will be placed in lowest sub-folder)
EV_D OCU MENT _STO RED	N/A if mode is Contentless, else flag indicating whether a document was stored successfully	
EV_D OCU MENT _LINK ED	N/A if mode is Contentless, else flag indicating whether a document was linked successfully after storaget (assuming IV_NO_CREATE_LINK is empty)	
EV_AR C_DO C_ID	document ID created in the CMS repository	{FC2F11EF-9AB9-C08E-86B0-74E484400000} or 124057761318
EV_FI LENA ME	docuflow automaticall trims the extension off and passed back just the filename as a parameters	
ET_RE TURN	Table that indicates a success of an error. The TYPE field will contain an 'S' or 'E' respectively and the MESSAGE field will contain information in the case of an error. When you use the IT_CUSTOM folder to create folders, the return table will also pass back the folder ID's created. The parent folder is indicated by a ! exclamation point, all sub-folders are indicated by a ~ tild.	Example of return table when folders are created: !SAP_4500017723,147811962292 ~SAP_SubFolder1,938382828 ~SAP_SubFolder2,938382829 ~SAP_SubFolder3,938382830 Refer to the below examples for more return table samples.

NOTE: a commit work is required.

Example

note Folder creation is only if supported by your CMS

An easy way to see an example for the parameter is to put a breakpoint in the docuflow DocManager code that will be triggered when storing a document from any Services for Object enabled transaction:

/dflow/cl_docmanager_helper⇒store_and_link_document, approximately line 713.

Below are sample values:

The screenshot shows the ABAP Source Code Display for the method `store_and_link_document` in the class `cl_docmanager_helper`. The code includes parameters for document storage and linking, and a variable explorer on the right showing the values of these variables.

St.	Variable	Val.	Val.
	/DFLOW/CL_COMMON_SU...	STORENEW	
	GC_FILE_TYPE	F	
	GV_SAP_OBJECT	BUS2012	
	GV_DOCTYPE	ZPURCHORD1	
	GV_OBJECT_ID	4500017723	
	LS_FILE	/Users/gregsoltys/Docume...	
	LV_DOCUMENT_STORED	X	
	LV_LINK_CREATED	X	
	LV_ARC_DOC_ID	884254634276	
	LV_FILENAME		
	LT_RETURN	[12x14(1088)]Standard Ta...	

Metadata passed

Table	LT_METADATA		
Attributes	Standard [6x3(2150)]		
Insert Column	Columns ...		
Row	ZMULTI [C(1)]	ZKEY [C(50)]	ZVALUE [C(1024)]
1		poNumber	4500017723
2		linkedToSap	True
3		companyCode	3000
4		paymentTerm	ZB01
5		vendorNumber	0000003740
6		vendorName	P0S Paper.com

Custom table passed

In this example, we instructed docuflow to create a 1 level subfolder. This is done by filling the CUSTOM table with the folder name:

Tables

Table Contents

Table

LT_CUSTOM

Attributes

Standard [1x1(100)]

Insert Column

Columns ...

Row	ZCUSTOM [C(50)]
1	SAP_4500017723

To implement this folder creation with Store and Link, you can implement in your code that calls Store and Link, or, implement the BADI /DFLOW/BADI_DOCMGR_STORE_LINK at the point BEFORE the document storage (ie: IV_CALLER 'B' in the BADI). Please refer to the BADI section of this document for more information. Using the BADI ensures that all methods of storing business documents (ie: Migrator, Publisher, standard docuflow DocManager via Services for Object, custom use of Store and Link) all follow the same logic.

Return Table Received

Row	TYPE [C(1)]	ID [C(20)]	NUMBER [N(3)]	MESSAGE [C(220)]
1	I		000	Uploaded file successfully: Sample Contract Amendment.pdf
2	I		000	Metadata map compiled successfully.
3	I		000	Storing document in own folder...
4	I		000	Attempting CREATE api. Return message below:
5	I		000	storing file via docuflow ECM
6			000	!SAP_4500017723,150018499187
7	S		000	Success
8	I		000	Attempting link creation.
9	S		000	ArchiveLink link was created successfully.

note: in this return table, we see that we also have docuflow configured to create a folder. When this occurs, the '!' indicates the folder name and the second value is the folder ID.

4.3.8.2. Send Metadata

Sending metadata to a file ID or folder ID directly can be accomplished via the below method.

Note that if the metadata template is not yet applied (in the case of Box), it will be added (note: for Box more than one metadata template can be applied).

/dflow/cl_docmanager_helper⇒send_metadata_update

Parameters

Parameter	Description	Sample Value
IV_CALLER	Describes who is calling the method. Logged in the docuflow API Gateway as such.	/dflow/cl_common_super⇒gc_caller_docmanager_custom
IV_SAP_OBJE CT	ArchiveLink SAP Object	BKPF
IV_AR_OBJEC T	ArchiveLink doctype	ZINVOICE01
IV_OBJECT_ID	ArchiveLink SAP key. Note this can be a compound key for certain SAP objects or line item/company code level.	300019000000012020
IV_ARC_DOC_I D	CMS ID to update	{FC2F11EF-9AB9-C08E-86B0-74E484400000} or 124057761318
IV_LABEL	Cosmetic value logged in the docuflow API gateway for reporting and tracking purposes.	MIGRATE_DIRECT_UPDATE
IV_FOLDER_M ETADATA	Boolean to indicate if update is to a folder.	X = folder EMPTY = file
IT_METADATA	Table of key value pairs of your CMS properties and the values to update	sapCompanyCode 3000
EV_DOCUME NT_UPDATED	Flag to indicate whether the file or folder was updated successfully.	
ET_RETURN	Table that indicates a success of an error. The TYPE field will contain an 'S' or 'E' respectively and the MESSAGE field will contain information in the case of an error.	

4.3.9. Overall CMS API Notes

- Although usage of each one does not require docuflow Add-On configuration (such as DocManager), it does require based configuration (CMS Profile(s), CMS Tag(s), Repository ID(s) and API Destination(s))
- Each API logs activity in the docuflow API Gateway automatically and will register itself as the API source "CUSTOM" which can be used in the selection criteria
- System message are checked (refer to the docuflow Solution Guide for more information)

4.3.10. SAP Code Vulnerability Analyzer (CVA)

docuflow passes all P1/P2/P3 SAP CVA SLIN_SEC checks.

All requirements are handed via inline docuflow code with the exception of access (READ) to standard some standard SAP table (considered level P3).

To pass this requirement, an additional transport is available from VersaFile to implement BADI SLIN_BADI_DBTAB_ACCESS.

4.3.11. log4j Update

Below is an overview of the process to upgrade an existing instalation of docuflow. This must be performed for each docuflow instance (ie: docuflow windows service).

[The latest log4j jar files can be found here.](#)

At the time of this writing, 2.17.1 is the latest log4j release. In the future if new versions are to be utilized, change the version numbers in the below steps accordingly.

1. Stop the VersaFile docuflow service
2. Take note of the "Log On" tab if you are running the service under a named account.
3. Download and move new log4j files (log4j-core-2.16.0.jar, log4j-api-2.16.0.jar) to /lib/log4j/
4. Delete/rename old l4j file(s)
5. Go to /services folder
6. Uninstall the service
 - Start a Windows CMD prompt as administrator
 - In the CMD prompt run /services/uninstallService.bat
 - Refresh your Windows Service list to confirm it is no longer present
 - If the docuflow service still exists, please edit the uninstallService.bat file to reflect the correct service name.
7. Edit installService.bat in the /services directory
 1. For upgrade from 1.2.15:
 - Delete the string "\lib\log4j\log4j-1.2.16.jar;"
 - Add a new entry "\lib\log4j\log4j-core-2.16.0.jar;\lib\log4j\log4j-api-2.16.0.jar;"
 - This entry should proceed "\lib\connector-service-6.5.7-fat.jar"
 2. For upgrade from 2.16.0 or later:
 1. Replace the two instances of "2.16.0" <or your current version> with "2.17.1"
8. Reinstall the service
 - Start a Windows CMD prompt as administrator
 - In the CMD prompt run the updated /services/installService.bat
9. Refresh your Windows Service list to confirm the service has been re-installed
10. **Note:** If you were running the service under a named account in the Log On tab of the Windows Service, you must reapply those changes.
11. Open the docuflow.txt log file with baretail
12. Start the VersaFile docuflow service
13. View baretail and ensure docuflow activity is being logged

4.4. VersaFile Release Management

4.4.1. Release Schedule

VersaFile plans to release an updated version of their products on a quarterly schedule. These releases will includes any security issue fixes, identified software defects and possible functionality updates.

For each updated release, product release notes will be included that cover all resolutions and updates included.

If a critical vulnerability is found and VersaFile determines an immediate fix is required, an interim fix (IF) will be issued. Clients that would require these IFs will be notified along with special installation instructions if required.

Any accumulated IFs will be rolled into the next official release.

Release Phases:

Release Phase	Description
Planning	Planning of features and fixes to be included in release
Development	Design and development of fixes and features
Testing and Acceptance	Testing and Acceptance of Results
Preparations and Communications	Documentation updates and preparations for release including communications
Distribution	Distribution of links/downloads of software

4.4.2. Source Code Review

All source code is maintained and secured within Github (<https://github.com>) and each source code repository has Dependabot Alerts enabled.

Dependabot is a technology that detects vulnerable dependencies and sends alerts when:

- A new vulnerability is added to the GitHub Advisory Database. The GitHub Advisory Database contains a curated list of security vulnerabilities. Vulnerabilities are added to the GitHub Advisory Database from the following sources:
 - The National Vulnerability Database
 - A combination of machine learning and human review to detect vulnerabilities in public commits on GitHub
 - Security advisories reported on GitHub
 - The npm Security advisories database
- New vulnerability data from WhiteSource is processed. WhiteSource is the worlds largest open source vulnerability tracking database. <https://www.whitesourcesoftware.com/vulnerability-database/>
- The dependency graph for a repository changes. For example, when a contributor pushes a commit to change the packages or versions it depends on, or when the code of one of the dependencies changes

Additionally, GitHub allows the team to review any dependencies added, updated, or removed in a pull request made against the default branch of a repository, and flag any changes that would introduce a vulnerability into the project. This allows the team to spot and deal with vulnerable dependencies before, rather than after, they reach the codebase.

VersaFile also maintains partnerships with SAP, IBM and other software companies we integrate with. As part of our partner agreement with these vendors, as we are updated to security issues. These issues are evaluated by the team to see if VersaFile code is affected. If it is, fixes are planned and applied as per our release schedule.

4.4.3. Vulnerability Assessment

VersaFile software applications are scanned for vulnerabilities using the Open Web Application Security Project's Zed Attack Proxy (ZAP) software.

OWASP ZAP provides penetration testing of web applications, providing coverage analysis for software security risks such as:

- Broken Authentication, Broken Access Control, Security Misconfiguration
- Sensitive Data Exposure, Insecure Deserialization, Cross-Site Scripting
- Injection, XML External Entities, Insufficient Logging & Monitoring
- Using Components with Known Vulnerabilities

4.4.4. Security Testing

VersaFile Software includes security testing as part of release management. The following security testing is done for each release of the software:

- **Source Code Review** – The system code undergoes a detailed review and analysis looking specifically for security vulnerabilities.
- **Vulnerability Assessment** – The system is scanned and analyzed for security issues.

4.4.5. Managing Security Issues

If critical security issues are found, security advisories are used to manage the issue and communications around it.

The following security advisory template is used to track the security issue:

- Title/Name/Description
- Package/Code Name
- Affected Versions
- Patched Versions
- Severity is defined as the following:
 - Low (1-3)
 - Moderate (4-6)
 - High (7-8)
 - Critical (9-10)
- This severity level will be determined using the industry standard Common Vulnerability Scoring System (CVSS) v3.1.
- A write-up on the issue that can be sent out using the following template:

```
### Impact
_What kind of vulnerability is it? Who is impacted?_

### Patches
_Has the problem been patched? What versions should users upgrade to?_

### Workarounds
_Is there a way for users to fix or remediate the vulnerability without upgrading?_

### References
_Are there any links users can visit to find out more?_

### For more information
If you have any questions or comments about this advisory:
* Email us at [VersaFile Support] (mailto:support@versafire.com)
```

4.5. Change Log

4.5.1. SAP

4.5.11. 3.1 (beta)

General

Added

SCRUDWL exposed as callable RFC function module: Each method for the class /DFLOW/CL_CMS_API has its own function module wrapper with RFC enabled. The functions are located under the function group /DFLOW/ZDFCMSAPI_SCRUD_WS. This is useful if you want to call the class-methods externally.

Improved

docuflow System Dashboard: Added the docuflow SAP version number to the ALV header.

Fixed

SAPLink Example: The SAPLink example provided under /DFLOW/CL_DOCMANAGER_HELPER⇒SAPLINK_EXAMPLE did not have the correct interface.

Updates to Micro-Process:

1. Split out exceptions codes in config for both Send errors and Return "rejection" (Return is only for approval tasks)
2. Provided plug-in /DFLOW/CL_WFCONNECT⇒SET_TASK_RECIP_VEND_FROM_EKKO for vendor email address retrieval from ekko (uses ADR6)
3. Moved WORKFLOW api so summary level in docuflow API Gateweay
4. Added GOON_FLAG option to plug-in logic
5. Broke out logging between storage and workflow separately

Improved

Fixed

docMigrator for ArchiveLink: fixed ALV header bug - when user hits refresh button the "counts" for Error/Success/Warning/Info dissapeared into the abyss

docMigrator for DMS

1. Comprehensive information on both source and target content now shown on dashboard
2. Retry on delete to avoid locking issues
3. Added Post-Processing: Force Metadata Update
3. Added Post-Processing: Validate Target Content
4. Added Utilities: Rollback Target DIR
5. Add support for docuflow ECM as source

Improved

Improved

docManager for DMS: this will be updated as improvements continue

Improved

Bug fix for AP Workflow PRE-POP NON-PO BADI: The auto-create logic for NON-PO uses the PRE-POP logic instead of net new/duplicate logic. PO is stil seperate as auto-create uses BAPI to create, and BDC for PRE-POP.

Fixed

docManager Store and Link Authorizations: Changed authorization checks for adding a document. ***** Users must now have "01" activity on S_WFAR_OBJ (previously was "03") *****

Added

docManager Store and Link Authorizations: Implemented new authorization checks showing document type drop down values. ***** Users musthave "01" activity on S_WFAR_OBJ in order for the document type to appear in the choice list (previously there was no check) *****

Fixed

Accounts Payable TrackLink: fixed bug to allow viewing SharePoint image prior to invoice creation (ie: uses config location instead of a READ)

Box

Added

Box Sign Integration

1. Single Step signature
2. Integration with Micro-Process workflow and TrackLink

M365

Added

Nested folder creation with file placement support

4.5.1.2. 3.0

General

Added

SAP CVA Compliance: ABAP code changes required to satisfy SAP CVA P1 and P2 exceptions. Some P2 and all P3 exceptions remedied via CVA BADI.

Added

CMS Load: All repositories supported to load repository, class, template, library, and properties into docuflow SAP.

Improved

System Dashboard: Improved time it takes to load the dashboard by not duplicating middleware and CMS pings.

Improved

System Dashboard: Improved navigation to activity for a particular rule. By selecting the rule on the main dashboard, then select the pushbutton "Show Activity". This takes you to the gateway log for that document type.

Improved

Micro-Process Workflow: TrackLink can now be configured to specify what the "key" is for the workflow (to support all generic SAP record). This supports up to 3 key fields.

Improved

Micro-Process Workflow: TrackLink can be configured to hide/show/label the 20 custom/configurable/ customer "useable" columns.

Improved

Micro-Process Workflow: TrackLink now accepts a business object on the selection screen. This allows you to define layout sets at a more granular level per micro-process workflow. In turn, this default layout set can be configured in the TrackLink config.

Improved

docuflow docManager Attachment List: improved size and alignment across tabs with new navigation pane for better user experience.

Fixed

docuflow docManager Attachment List: when no attachment exist, top level folder in navigation pane would dump when double clicked.

Box

Added

New workflow API for Box Task integration /DFLOW/CL_CMS_API→CMS_API_WORKFLOW. Both "General" and "Approval" Box tasks are supported including multi-recipients, due date support, and multi-line task "body" description.

Improved

"Flat" folder creation via CREATE API (using ZCUSTOM table) now supports where the file (ie: which folder) is placed in by use of the tilde (~). Refer to the BADI for docManager for more detail on how to implement: <https://docs.versafire.com/docuflow-guides/hMui-docmanager>

Added

"Nested" folder creation via CREATE API (using ZCUSTOM table) - in addition to the "flat folder" creation above, nested folders are now supported. Each line of the ZCUSTOM table indicates a new nested (ie: sub) folder and the usage of the tilde (~) tell docuflow which folder to place the file within. Refer to the BADI for docManager for more detail on how to implement: <https://docs.versafire.com/docuflow-guides/hMui-docmanager>

Microsoft

Added

"Flat" folder creation via CREATE API (using ZCUSTOM table) - support for flat folder creation with file placement. Each line of the ZCUSTOM table indicates a new sub folder under the parent folder (which is line 1 of the ZCUSTOM table). The usage of the tilde (~) tells docuflow which folder to place the file within. Refer to the BADI for docManager for more detail on how to implement: <https://docs.versafire.com/docuflow-guides/hMui-docmanager>

Added

Load functionality in docuflow System Dashboard now supports SharePoint libraries. This allows you to see the property names which are used for configuration, metadata maps, etc.

4.5.1.3. 2.9

General

Added

Micro-Process Workflow: New workflow module to support any business use case. Sharing the same docuflow SAP workflow engine as our Accounts Payable workflow, this multi-step workflow is fully configurable for agent resolution, deadline monitoring, and integration with workflow outside of SAP for task completion.

Added

docManager Attachment List expandable folders: Support for navigation pane on left hand side of the docManager attachment list. Expandable folders include a) docuflow flow b) document type classification and c) file type (mimetype).

Improved

docManager Attachment List tab: Moved attachment list to first tab as this is the most common usage. Storing business documents is now on the second tab.

Added

docManager Attachment List download support: You can now download attachments directly from within the attachment list.

Added

docManager Attachment List DB note creation support: For customers still wanting to save "notes" in their SAP database, they can do so directly from the docuflow attachment list.

Added

docManager Attachment List CMS note creation support: You can now create "notes" in your content repository instead of relying on the SAP database. These are stored in your CMS system as .txt files.

Fixed

Publisher: corrected call to /dflow/cl_docmanager_helper⇒sotre_and_link_document to be static method syntax (caused dump in older versions of SAP ABAP stack)

4.5.2. Middleware

4.5.2.1. 6.5.7

Box

Added

Folder URL's - When using the READ API and passing the parameter "documentid", if docuflow determines that document ID does not exist, it will then check if the id is a folder. If the id exists, the read functionality will assume that we are reading a folder and return a folder URL. If we were seeking a document and its corresponding document id existed, then the read API would function as before.

Added

Auto applying cascade policy - In the even we want to enable cascade policy on a folder upon its creation we would add to the ini under [Box Connection]<insert-cms-name>.box.autoCascadePolicy=true. By default, this is set to false.

Added

Content server search bypass - Traditionally when retrieving content via content server there are extra API calls due to how retrieval in box works utilizing content server metadata. In some scenarios we may instead limit the meta data utilized and instead retrieve the document via it's box document id. By not retrieving the document by its content server document id we reduce the number of API calls required. Inside SAP the BOX document id would be linked. Discretion is advised upon using this functionality as it is not meeting SAP specifications. To enable this performance feature in the active ini file under section [Content Server] add <insert-cs-name>.contentServer.searchBypass=YES to the corresponding content server connection.

Added

Folder URL expiry time - When sharing box folder URL's, we can set the expiry time. Default is 60 seconds. You can change the expiry time by modifying the active ini profile under section [Content Server] we can add the following rule<insert-cs-system-name>.box.boxURLFolderExpiryTime=360 (the following example would expire a shared folder URL after 10 minutes).

Real time search - Box released a new search API to retrieve items. However, that requires using a new query language. The advantage of using this new query language is we can avoid the previous 10-minute delay when documents are initially indexed using the previous API. Anywhere a traditional "mdfilter" (metadata filter) array was used it would be replaced with a new box search query. An example is given below,

Added

```
{ "from": "enterprise_761921896.sdInvoice", //the enterprise id followed by the metadata template
  "query": "docuflowSaplinkEnabled = :value", //the field we are wanting to check
  "query_params": { "value": "True" //the value of the item it would hold }, "ancestor_folder_id": "131865107848" //the folder
  where want to start looking }
```

This is traditionally used in SAPLink, we would just modify the filter field in the sap link rule in the active ini file. To enable this functionality we would add <insert-cms-name>.box.searchQueryEnabled=true by modifying the active ini profile under section [Content Server]

Nested folder creation with file placement - A new tree structure folder creation. In the custom table, the first row holds the folder hierarchy that is comma delimited. <folder1>,<folder2>,<folder3> where folder 1 and folder 2 belong to folder 1. To enable metadata creation along with the folder creation send along ZFOLDER_METAFLAG='X'.

Added

~ indicates where file placement occurs

In the return table expect to see a format similar to this,

```
!<folder-name>,id (! is an identifier for parent when parsing)
~<folder-name>,id (~ is an identifier for children when parsing)
```

GUI

Added

New web user interface to configure docuflow middleware - It can be accessed via https://localhost:82 (username: docuflow, password: docuflow) **Can be changed.

General

Added

Kubernetes support

5. SuccessFactors

5.1. Agnostic Guide

5.1.1. SuccessFactors Technical Overview



5.1.1.1. Introduction

docuflow for SuccessFactors integrates documents between SuccessFactors and external content repositories.

This integration enables:

- Document storage, linking, and metadata transfer from SuccessFactors to the repository platform.
- Documents captured in SuccessFactors are stored within the repository, and documents added or managed in the repository can be linked back to SuccessFactors employee records.

To enable users to access the documents in Box, two HTML5 applications give them access:

- My Documents
- Document Search

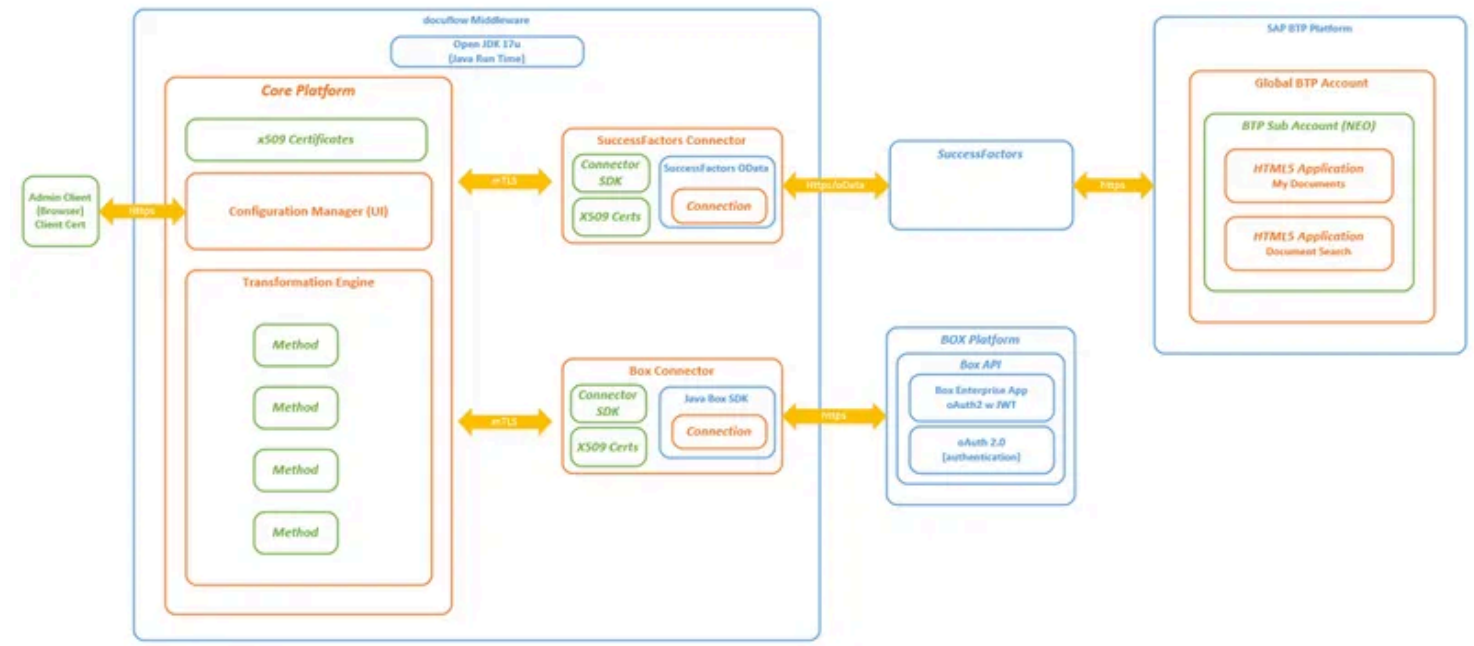
My Documents will show a user all of their employee documents that have been linked from Box.

Document Search will allow HR Managers to search for documents belonging to employees that report to them.

The process to capture and link documents is done through the docuflow middleware while the My Documents and Document Search is done through HTML5 applications deployed on the SAP BTP Platform.

5.1.1.2. Architecture Overview

The below diagram outlines the basic components of the solution



docuflow Middleware Responsibilities

- Discover new employees and create an **Employee Folder** in the repository, linking this folder back to their employee profile.
- Discover new employee documents in SuccessFactors, add them to the repository, and link them back to SuccessFactors using **LinkCards**.
- Discover new employee documents in the repository and link them to SuccessFactors using **LinkCards**.

SAP BTP Platform Responsibilities

- Provide a user interface (**MyDocuments**) for employees to view the LinkCards attached to their employee profile.
- Provide a user interface (**DocumentSearch**) for HR managers to view LinkCards for employees they are related to.

5.1.1.3. Minimum Requirements

docuflow Middleware Deployment and Network Requirements

Deployment Options

The **docuflow middleware component** (docuflow Enterprise Connector) can be deployed in various environments including:

- On-premises servers
- Private cloud infrastructure
- Fully managed service offered by VersaFile

Recommended Server Resources for On-Premises or Private Cloud Deployments

Component	Recommended Specification
Operating Systems	Windows Server 2012, 2012 R2, 2016, 2019
CPU	Minimum 4 cores or 4 physical CPUs, each at least 2.0 GHz
Disk Space	Minimum 40 GB
RAM	Minimum 16 GB
Java Runtime Environment	OpenJDK 17 LTS or later

Additional Requirements

- **JCO Version:** 3.0.14 and above
 - Versions 3.1.x and above require the [Visual C++ Redistributable Packages for Visual Studio 2013](#)
 - See SAP Note [2786882](#) for details

Network Access and Firewall Configuration for Repository API Access

To enable full functionality, the middleware server must have outbound network access to the content repository's API endpoints.

5.1.2. BTP Configuration

This section describes the typical configuration required to integrate the three HTML5 applications into SuccessFactors.

- My Documents
- Document Search
- Document Search (HRBP)

Required Accounts:

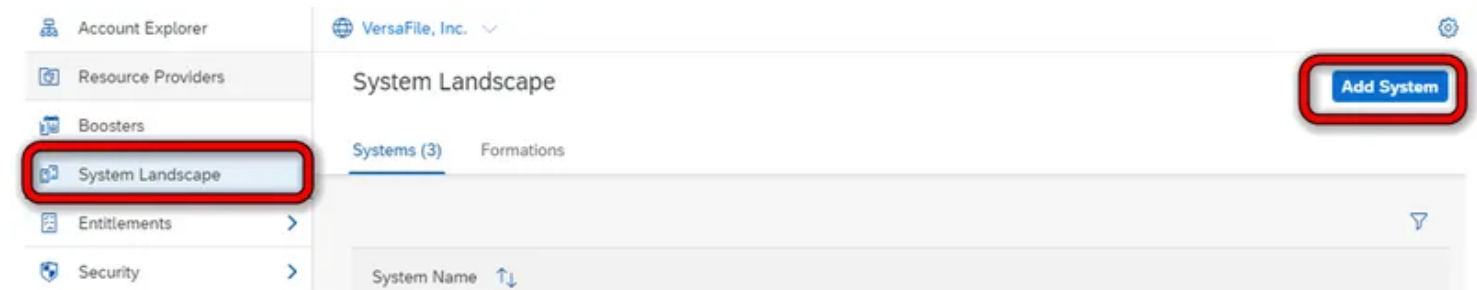
- Administrative BTP Access
- Administrative SuccessFactors Access

5.1.2.1. Linking SF to BTP Landscape

The following steps register the SuccessFactors instance to the BTP system Landscape. This is the first task in configuring the integration.

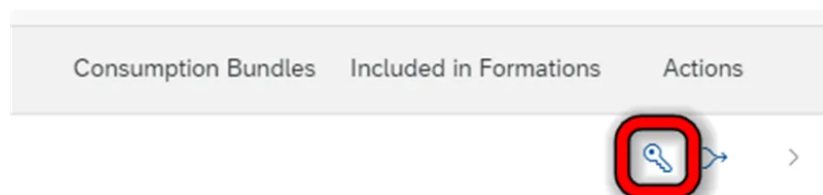
At the BTP Global Account, navigate to **System Landscape**.

Click the **'Add System'** button

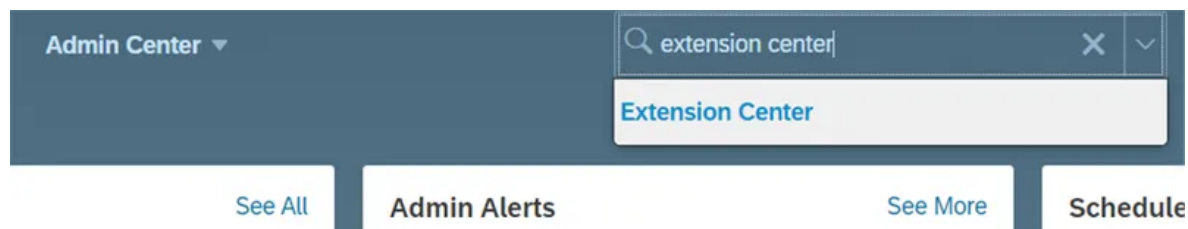


Supply a descriptive System Name and choose **'SAP SuccessFactors'** from the Type dropdown

Once the system has been added, click on the 'Get Token' icon. Copy this Token Value. This Token will be used within SuccessFactors to register the SuccessFactors system with the BTP Landscape.

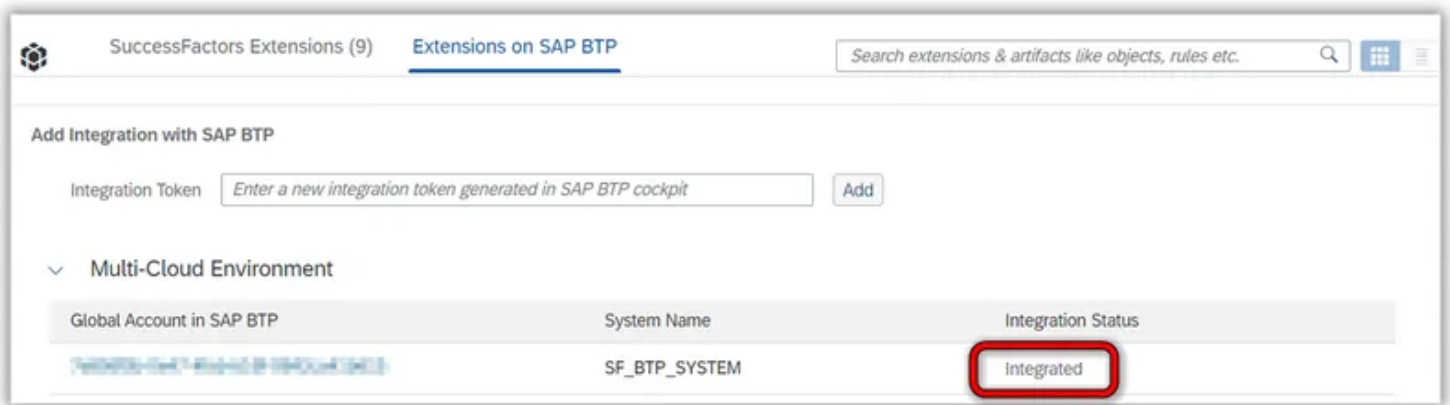


Next Log into **SuccessFactors** with an Administrative Account and Navigate to **'Extensions Center'**



Within the Extension Center navigate to the **'Extensions on SAP BTP'** tab. In the section **'Add Integration with SAP BTP'** paste the Token and click the **'Add'** button. This will connect the SuccessFactors instance with the BTP landscape.

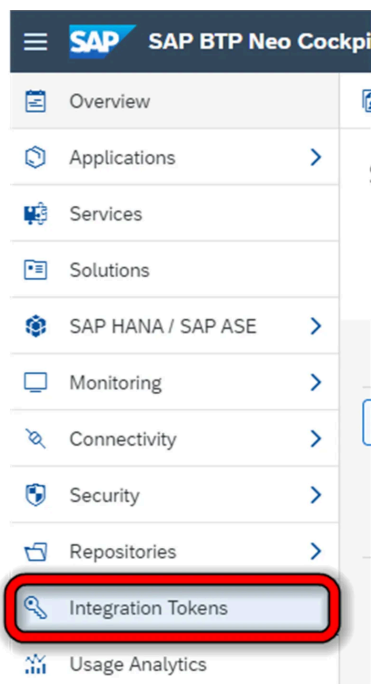
A message indicating the Systems are integrating will be shown. After a short period the status will show **'Integrated'**



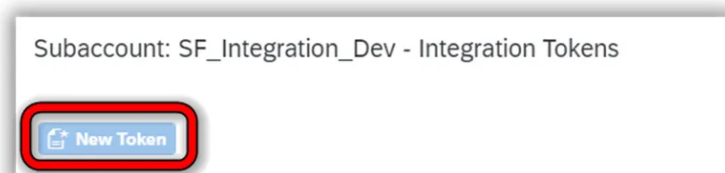
Next, link the BTP Sub Account to the Extensions on SAP BTP

- **NOTE:** The BTP Sub Account MUST be a NEO account (SAP Region)
- **NOTE:** See '**BTP Sub Account Creation**' sub-topic if you have not yet created a Sub Account

Open your BTP Sub Account and Navigate to '**Integration Tokens**'



Click on the '**New Token**' option to generate a new integration token for this BTP Sub Account. This token will then be used in SuccessFactors to integrate to the BTP Sub Account



Navigate back to '**Extentions Center**' in SuccessFactors

Once again, use the 'Add' Integration Token feature

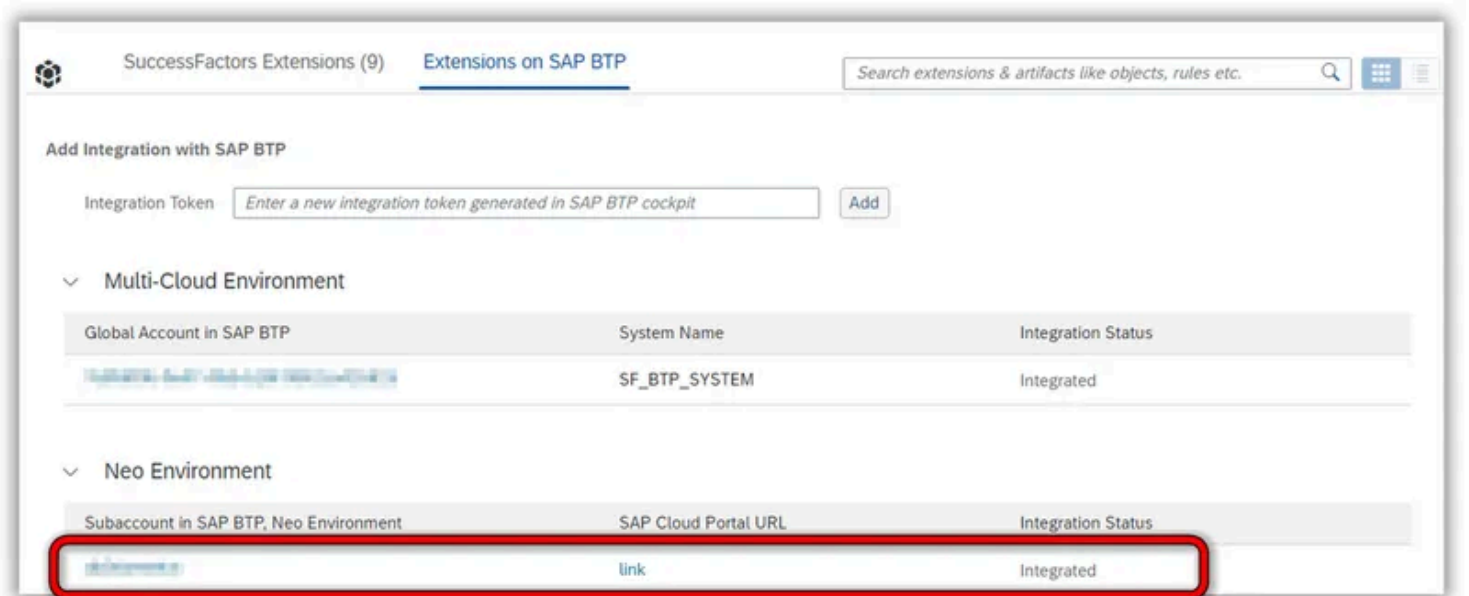


SuccessFactors Extensions (9) Extensions on SAP BTP Search extensions & artifacts like objects, rules etc.

Add Integration with SAP BTP

Integration Token Add

Once this token is added, it will show up in the '**Neo Environment**' Section



SuccessFactors Extensions (9) Extensions on SAP BTP Search extensions & artifacts like objects, rules etc.

Add Integration with SAP BTP

Integration Token Add

Multi-Cloud Environment

Global Account in SAP BTP	System Name	Integration Status
Global Account in SAP BTP	SF_BTP_SYSTEM	Integrated

Neo Environment

Subaccount in SAP BTP, Neo Environment	SAP Cloud Portal URL	Integration Status
Subaccount in SAP BTP, Neo Environment	link	Integrated

A Trust has been created to BTP. To see this, Navigate to **BTP Cockpit** → **Security** → **Trust**

 **SAP BTP Neo Cockpit**

Overview

Applications

Services

Solutions

SAP HANA / SAP ASE

Monitoring

Connectivity

Security

Trust

Authorizations

OAuth

Repositories

Integration Tokens

Usage Analytics

Entitlements

Members

VersaFile, Inc. / SF_Integration_Dev

Subaccount: SF_Integration_Dev - Trust

Trust Management

Local Service Provider | Application Identity Provider | Custom Application Domains Settings | Basic Authentication

 **Manage Local Provider Settings**
for yukrury3mx

Configuration Type: Custom

Local Provider Name: *

Signing Key: *

Signing Certificate: *

Certificate Valid From: Wed Nov 09 22:24:42 UTC 2022

Certificate Valid To: Tue Nov 09 22:24:42 UTC 2032

Principal Propagation: Enabled

Force Authentication: Disabled

Edit

Get Metadata

275 / 534

docuflow Guides

Modified: 07/02/2026

5.1.2.1.1. BTP Sub Account Creation

This step only needs to be done if there is not already a sub account created for the BTP environment.

At the Global Account Level, navigate to **'Account Explorer'** and click the **'Create'** button and select **'Sub Account'**

NOTE: Select one of the SAP (NEO) Regions to run the integration

Create Subaccount

Display Name*

SF_Integration_Dev

Subdomain ⓘ

Region*

US West (Colorado Springs) ▼

Singaporecf-ap21

US East (VA)cf-us21

US West (WA)cf-us20

SAP

Australia (Sydney)neo-ap1

Brazil (São Paulo)neo-br1

Canada (Toronto)neo-ca1

China (Shanghai)neo-cn1

Europe (Amsterdam)neo-eu3

Europe (Frankfurt)neo-eu2

Japan (Tokyo)neo-jp1

KSA (Riyadh)neo-sa1

UAE (Dubai)neo-ae1

US West (Colorado Springs)neo-us4

Description

HTML 5 Application integrations to SuccessFactors

Parent*

VersaFile, Inc. 

Account structure.

Not global account or to an parent.



Create

Cancel

5.1.2.2. HTML5 Application Registrations

This section will demonstrate how install the HTML5 applications into the BTP Sub Account that has been integrated into the SuccessFactors system in the previous section.

SAP Web IDE Full-Stack Service

If not already enabled, SAP Web IDE Full-Stack Service must be enabled in the sub account.

To enable this, in the sub account navigate to 'Services' and then select the tile/option 'SAP Web IDE Full-Stack' and then click '**Enable**'.

Security Permissions

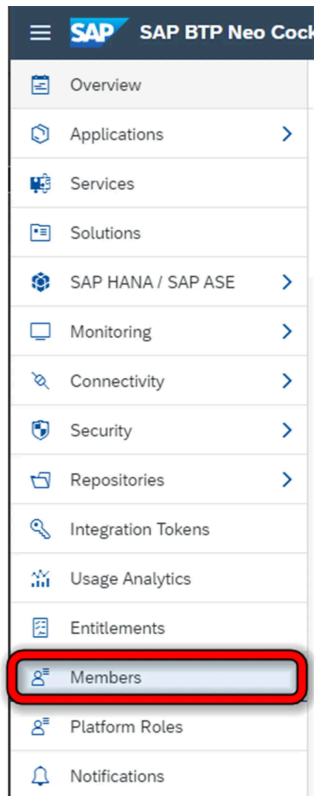
NOTE: This section will require the BTP Command Line Interface (CLI). If you do not have this installed please see the sub section '**BTP Command Line Interface**'

First, Permissions must be given to allow configuration of the BTP Sub Account.and Web Applications.

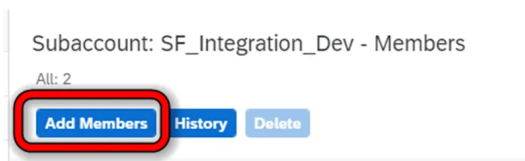
To Enable security for the Web Applications, navigate to **Security→Authorizations**

- Enter the S User ID and click 'Show Assignments'
- Click 'Assign' and then select
 - Subaccount 'SAP Web IDE xxx'
 - Application: di
 - Role: DiAdministrator
- Click 'Assign' and then select:
 - Subaccount 'SAP Web IDE xxx'
 - Application: featurediscovery
 - Role: DevXAdministrator

Next, Navigate to the '**Members**' section of the BTP Sub Account



In the Members section, click **'Add Members'**

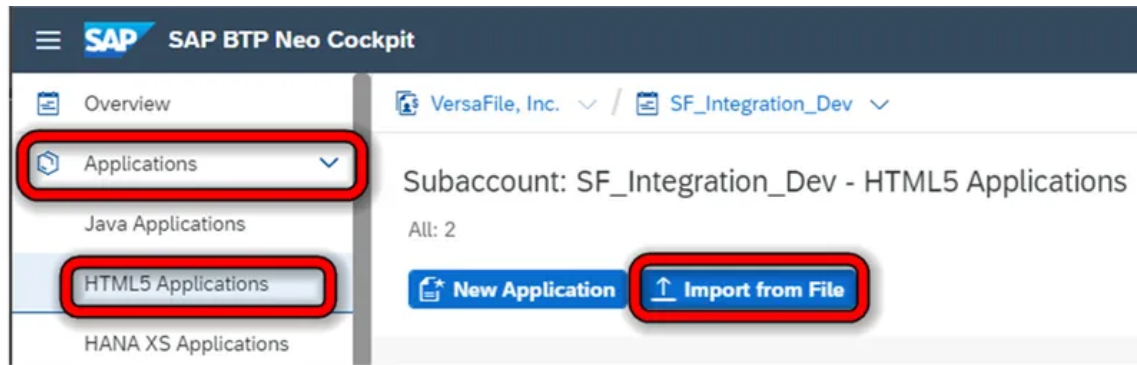


Enter the **User id** and select the permissions you want this user account to have the BTP Sub Account.

 A screenshot of the 'Add Members' dialog box. It contains a text input field for 'User IDs' with the placeholder text 'Sxxxxxxxxx'. To the right, under 'Assign roles:', there are five checkboxes, all of which are checked: 'Administrator (predefined role)', 'Developer (predefined role)', 'Support User (predefined role)', 'Application User Admin (predefined role)', and 'Cloud Connector Admin (predefined role)'. At the bottom, there is a text input field for 'Comment (optional, visible to all members):'. The 'Add Members' and 'Cancel' buttons are at the bottom right.

Next, upload the HTML5 Applications to the BTP Sub Account. These applications are supplied in a .zip format.

Navigate to **Applications→HTML5 Applications** in the BTP Sub Account and click **'Import from File'**



In the dialog, select the '**sfmydocuments.zip**', enter the version number 1 and click 'Import'

Follow the same procedure as above for '**sfdocumentsearch.zip**'.

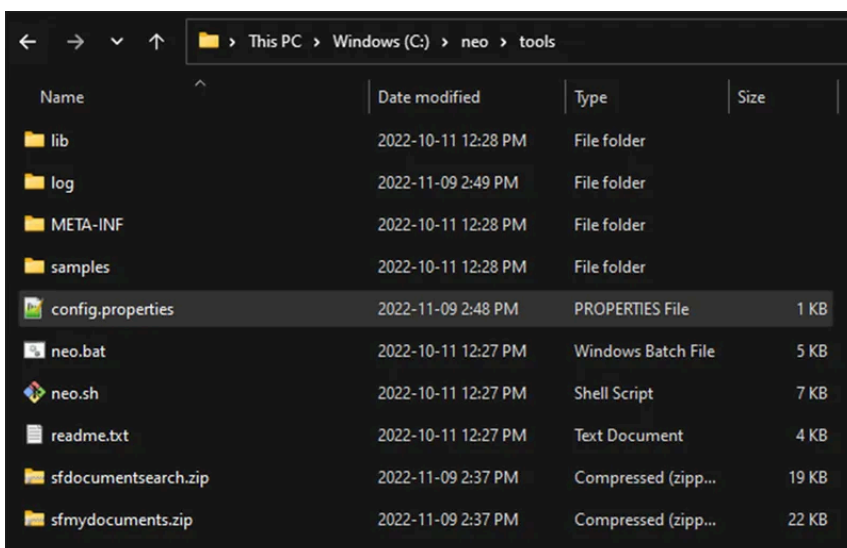
Once completed the applications should be listed in the HTML5 Applications screen.

Enable Application Access

NOTE: This next step required the BTP Command Line Interface (CLI). If you do not have this toolset installed, see the sub topic '**BTP Command Line Interface (CLI)**'

NOTE: The following examples are shown on windows but can be adapted for any other command line/shell

Open a Command Prompt/Shell and navigate to the '**Tools**' directory of the BTP Tools installation



Setup a properties file (e.g. **config.properties**) with the information that will be required for the connection. This will make the process easier.

NOTE: For a list of BTP Host values/locations: <https://help.sap.com/docs/btp/sap-btp-neo-environment/regions-and-hosts-available-for-neo-environment>

```
#####
# General settings - relevant for all commands #
#####

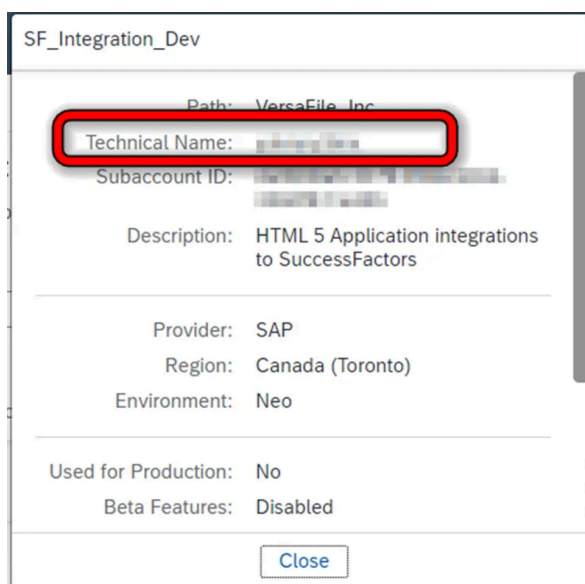
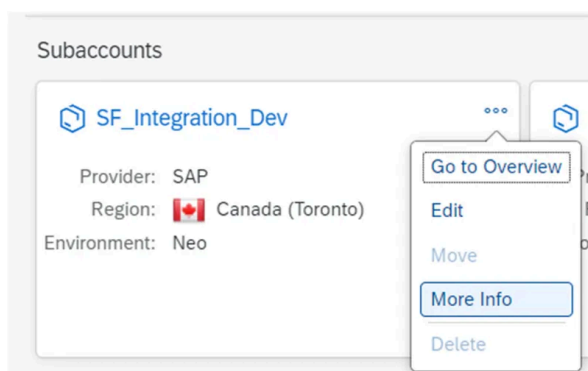
# Your account name
account=sfdocumentsearch

# Application name
application=sfdocumentsearch
application-type=html5

# User for login to hana.ondemand.com.
user=sapuser@hana.ondemand.com

# URL of the landscape admin server. Optional. Defaults to
hana.ondemand.com.
host=hana.ondemand.com
```

account = the ID of your BTP Sub Account



application = name of the application be registered. This process will be done twice, once for sfdocumentsearch and once for sfmydocuments. Note: use one of these values, run the command below, then alter this value and run the command again.

user = your SAP User Account. If you have a SAP Universal ID you will want to specify your S-ID, otherwise you can use the email associated with your S-ID

With the Command Prompt BTP CLI tools, execute the following command (use your properties file name)

neo.bat hcmcloud-enable-application-access config.properties

NOTE: if you are not using the properites file method, the full command would be:

```
neo hcmcloud-enable-application-access --application <my_application> --application-type java --account
<my_extension_subaccount> --user <user id> --host us1.hana.ondemand.com
```

Next, modify the config.properties file so that **application=sfmydocuments** to allow registration of the **sfmydocuments** application

```
#####
# General settings - relevant for all commands #
#####

# Your account name
account=sfmydocuments

# Application name
application=sfmydocuments
application-type=html5

# User for login to hana.ondemand.com.
user=sfmydocuments@hana.ondemand.com

# URL of the landscape admin server. Optional. Defaults to
hana.ondemand.com.
host=hana.ondemand.com
```

With the Command Prompt BTP CLI tools, execute the following command (use your properties file name)

neo.bat hcmcloud-enable-application-access config.properties

After registration example:

```

host      : https://ca1.hana.ondemand.com
account   : y[REDACTED]
application : sfmydocuments
application-type: html5
SDK version : 4.36.3
user      : [REDACTED]

Password for your user:
Enabling application access.....

Application access was enabled for application 'sfmydocuments' from subaccount 'y[REDACTED]':
Application Main URL      : https://sfmydocuments-[REDACTED].dispatcher.ca1.hana.ondemand.com
Service Provider Audience URL : https://ca1.hana.ondemand.com/[REDACTED]
Service Provider Logout URL  : https://authn.ca1.hana.ondemand.com/saml2/sp/slo/[REDACTED]

C:\Users\kyleh\Documents\SAP BTP\SAP NEO 2\tools>neo hcmcloud-enable-application-access vf2.properties

Neo Console Client

Executing [hcmcloud-enable-application-access] command with arguments:
host      : https://ca1.hana.ondemand.com
account   : [REDACTED]
application : sfdocumentsearch
application-type: html5
SDK version : 4.36.3
user      : S0024416632

Password for your user:
Enabling application access.....

Application access was enabled for application 'sfdocumentsearch' from subaccount '[REDACTED]':
Application Main URL      : https://sfdocumentsearch-[REDACTED].dispatcher.ca1.hana.ondemand.com
Service Provider Audience URL : https://ca1.hana.ondemand.com/[REDACTED]
Service Provider Logout URL  : https://authn.ca1.hana.ondemand.com/saml2/sp/slo/[REDACTED]

```

Register oData Connection

Finally, using the CLI register an oData connection (only one required for both applications)

Edit the **config.properties** file and comment out the application and:

Edit out the application application type properties:

```
#application=sfdocumentsearch
```

```
#application-type=html5
```

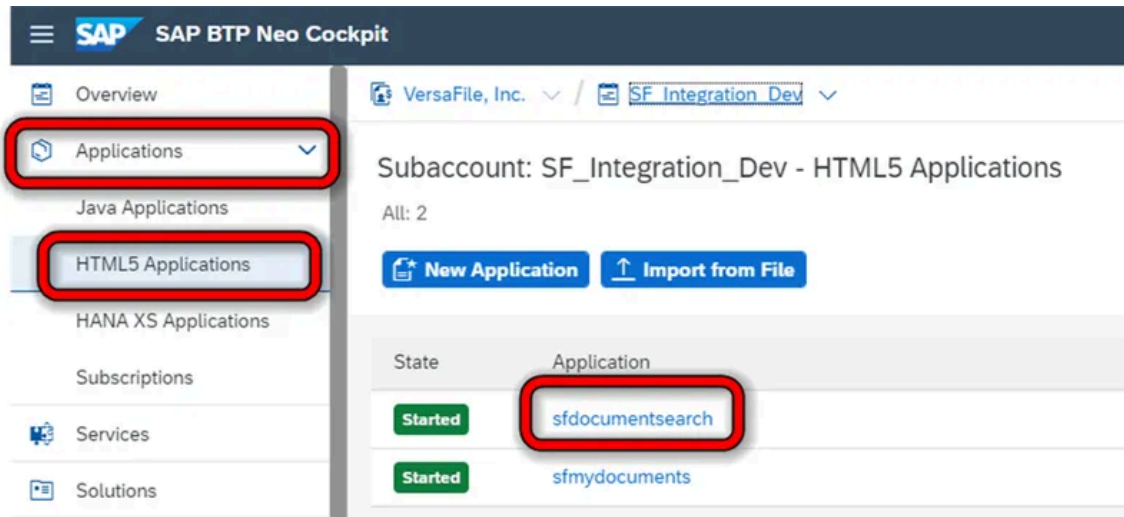
Add a new value for the name of the connection:

```
name=sfapps_odata
```

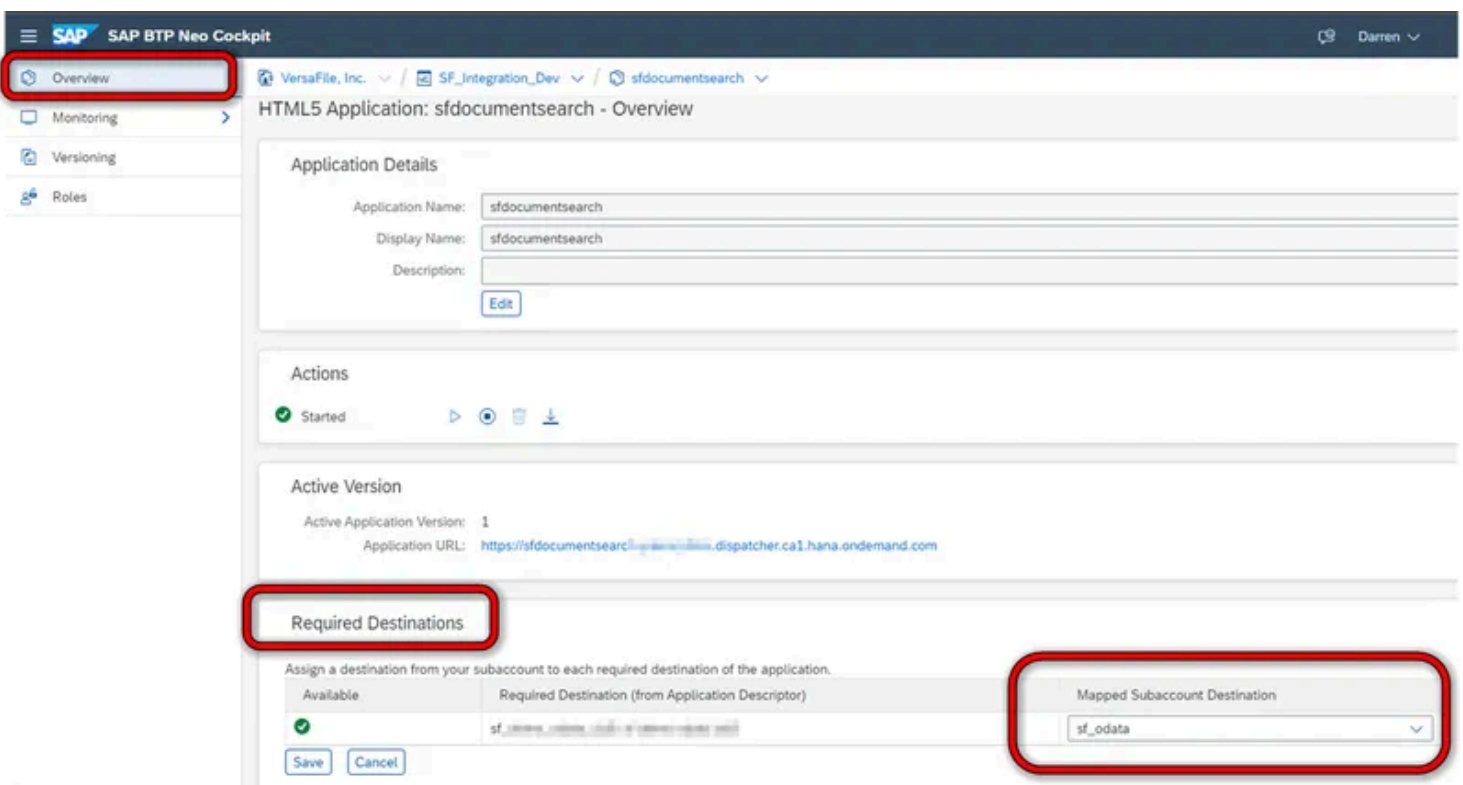
Neo.bat *hcmcloud-create-connection config.properties*

Set oData Connection in HTML5 Applications

For the next step, return to the **BTP Cockpit** and navigate to the **BTP Sub Account → Applications → HTML5 Applications** and click the sfdocumentsearch application



Under 'Required Destinations' set the 'Mapped Subaccount Destination'



Next, Click the 'Versioning' section, under 'History' section select the 'Versions' and set the 'Mapped Subaccount Destination'

SAP BTP Neo Cockpit

Overview Monitoring **Versioning** Roles

VersaFile, Inc. / SF_Integration_Dev / sfdocumentsearch

HTML5 Application: sfdocumentsearch - Versioning

Source Location

Git Repository URL: <https://git.ca1.hana.ondemand.com/> /sfdocumentsearch

Edit Online

History

Commits Versions

Refresh Import from File

Version	Active	Actions
1	✓	✕ ↓

Required Destinations for Version "1"

Assign a destination from your subaccount to each required destination of the application.

Available	Required Destination (from Application Descriptor)	Mapped Subaccount Destination
✓	sf_odata	sf_odata

Save Cancel

Next, register the second HTML5 application sfmydocuments. Return to the **BTP Cockpit** and navigate to the **BTP Sub Account** → **Applications** → **HTML5 Applications** and click the **sfmydocuments** application

SAP BTP Neo Cockpit

Overview **Applications** Java Applications HANA XS Applications Subscriptions Services Solutions

VersaFile, Inc. / SF_Integration_Dev

Subaccount: SF_Integration_Dev - HTML5 Applications

All: 2

New Application Import from File

State	Application
Started	sfdocumentsearch
Started	sfmydocuments

Under '**Required Destinations**' set the '**Mapped Subaccount Destination**'

Overview

Monitoring

Versioning

Roles

HTML5 Application: sfmydocuments - Overview

Application Details

Application Name: sfmydocuments

Display Name: sfmydocuments

Description:

Edit

Actions

Started

Active Version

Active Application Version: 1

Application URL: <https://sfmydocuments.dispatcher.ca1.hana.ondemand.com>

Required Destinations

Assign a destination from your subaccount to each required destination of the application.

Available	Required Destination (from Application Descriptor)	Mapped Subaccount Destination
		sf_odata

Edit

Next, Click the **'Versioning'** section, under **'History'** section select the **'Versions'** and set the **'Mapped Subaccount Destination'**

The screenshot displays the SAP BTP Neo Cockpit interface. On the left, the 'Versioning' tab is selected in the navigation menu. The main area shows the 'HTML5 Application: sfmydocuments - Versioning' page. Under the 'History' section, the 'Versions' button is highlighted. Below this, a table lists the versions, with version '1' being active. At the bottom, the 'Required Destinations for Version 1' section shows a table with columns 'Available' and 'Required Destination (from Application Descriptor)'. The 'Available' column has a green checkmark, and the 'Required Destination' column shows 'Mapped Subaccount Destination' with the value 'sf_odata'.

Version	Active	Actions
1	✓	✕ ↓

Required Destinations for Version "1"

Assign a destination from your subaccount to each required destination of the application.

Available	Required Destination (from Application Descriptor)
✓	Mapped Subaccount Destination sf_odata

5.1.2.2.1. BTP Command Line Interface (CLI)

This section has information on how to install the BTP CLI if it does not already exist in your environment.

The CLI can be downloaded from: <https://tools.hana.ondemand.com/#cloud>

SAP BTP SDK for the Neo environment

The SAP BTP SDK for the Neo environment includes platform API JARs, a local runtime for testing, documentation, samples, and the Neo console client. These are the major versions of the SAP BTP SDK for the Neo environment:

- SAP BTP SDK for the Neo environment, version for Java Web Tomcat 8 - provides support for some of the standard Java EE APIs (Servlet, JSTL, EL)
- SAP BTP SDK for the Neo environment, version for Java Web Tomcat 9 - provides support for some of the standard Java EE APIs (Servlet, JSTL, EL)
- SAP BTP SDK for the Neo environment, version for Java EE 7 Web Profile TomEE 7 - provides support for Java EE 7 Web Profile APIs

Available SAP BTP SDKs for the Neo environment

SAP BTP SDK for the Neo environment	Comment	Version	File Size	Download	Show old versions
Java Web Tomcat 8		3.166.7	56.8 MB	neo-java-web-sdk-3.166.7.zip (sha1)	
Java Web Tomcat 9		4.39.6	59.1 MB	neo-java-web-sdk-4.39.6.zip (sha1)	
Java EE 7 Web Profile TomEE 7		1.137.8	93.9 MB	neo-javaee7-wp-sdk-1.137.8.zip (sha1)	

NOTE: The above tools will require a Java Runtime Environment (JRE). If you do not have one scroll down on the page to the SAP JVM section and download a version appropriate for your Operating System

SAP JVM

The SAP JVM is a prerequisite for local profiling with the SAP JVM Profiler. It is a standard compliant certified JDK, supplemented by additional supportability and developer features and extensive monitoring and tracing facilities. For more information, see the [SAP JVM documentation](#)

Available SAP JVMs

Operating System*	Architecture	Version	File Size	Download
Linux	x86_64	8.1.091	130.0 MB	sapjvm-8.1.091-linux-x64.zip (sha1)
Linux	x86_64	8.1.091	126.0 MB	sapjvm-8.1.091-linux-x64.rpm (sha1)
Linux	ppc64le	8.1.091	126.0 MB	sapjvm-8.1.091-linux-ppc64le.zip (sha1)
Linux	ppc64le	8.1.091	119.0 MB	sapjvm-8.1.091-linux-ppc64le.rpm (sha1)
Mac OS X	x86_64	8.1.091	131.0 MB	sapjvm-8.1.091-macosx-x64.zip (sha1)
Windows	x86_64	8.1.091	162.0 MB	sapjvm-8.1.091-windows-x64.zip (sha1)

*SAP JVM is supported for:
SUSE Linux Enterprise Server 12 and 15, Redhat Enterprise Linux 7 and 8, Oracle Linux 7 and 8, Windows 10, Windows Server 2012, 2012 R2, 2016, 2019 and 2022, Mac OS X 10.14 (Mojave), 10.15 (Catalina), 11 (Big Sur) and 12 (Monterey)

Download the above tools and unpack to a system directory. Install the JRE if required.

If you just installed a JRE, the tools require 'JAVA_HOME' be defined. This will be the path where the JRE was downloaded to.

set JAVA_HOME=C:*your directory path* \JVM\sapjvm_8\jre

5.1.2.3. Extension Admin Permissions

The extension administrators will require permissions to the BTP environments to complete the Application setup in the SuccessFactors Extensions Center portlet.

- In the SuccessFactors Admin Center, navigate to the **Permissions Role Detail**
- Select the Administrative Role that is doing the solution configuration
- Navigate to the Permission setting '**Permission not requiring target**'
- Select '**Manage Extensions on SAP BTP**'
 - Allow '**Create Integration with SAP BTP**'

Add the Administrative user who will be accessing the Portal Configuration to the Extensions Administrators Group. In SuccessFactors:

- Navigate to '**Manage Permission Groups**'
- Select the group name: **Extension Administrators**
- Add the selected UserIds to the role

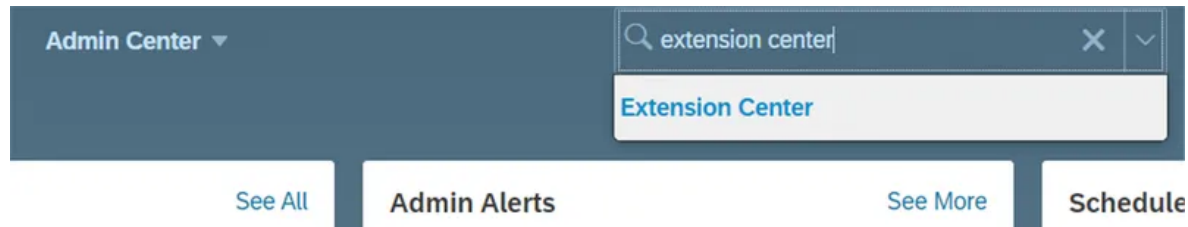
SAP Tech Note:

<https://help.sap.com/docs/btp/sap-business-technology-platform/register-sap-successfactors-system-in-global-account-in-sap-btp>

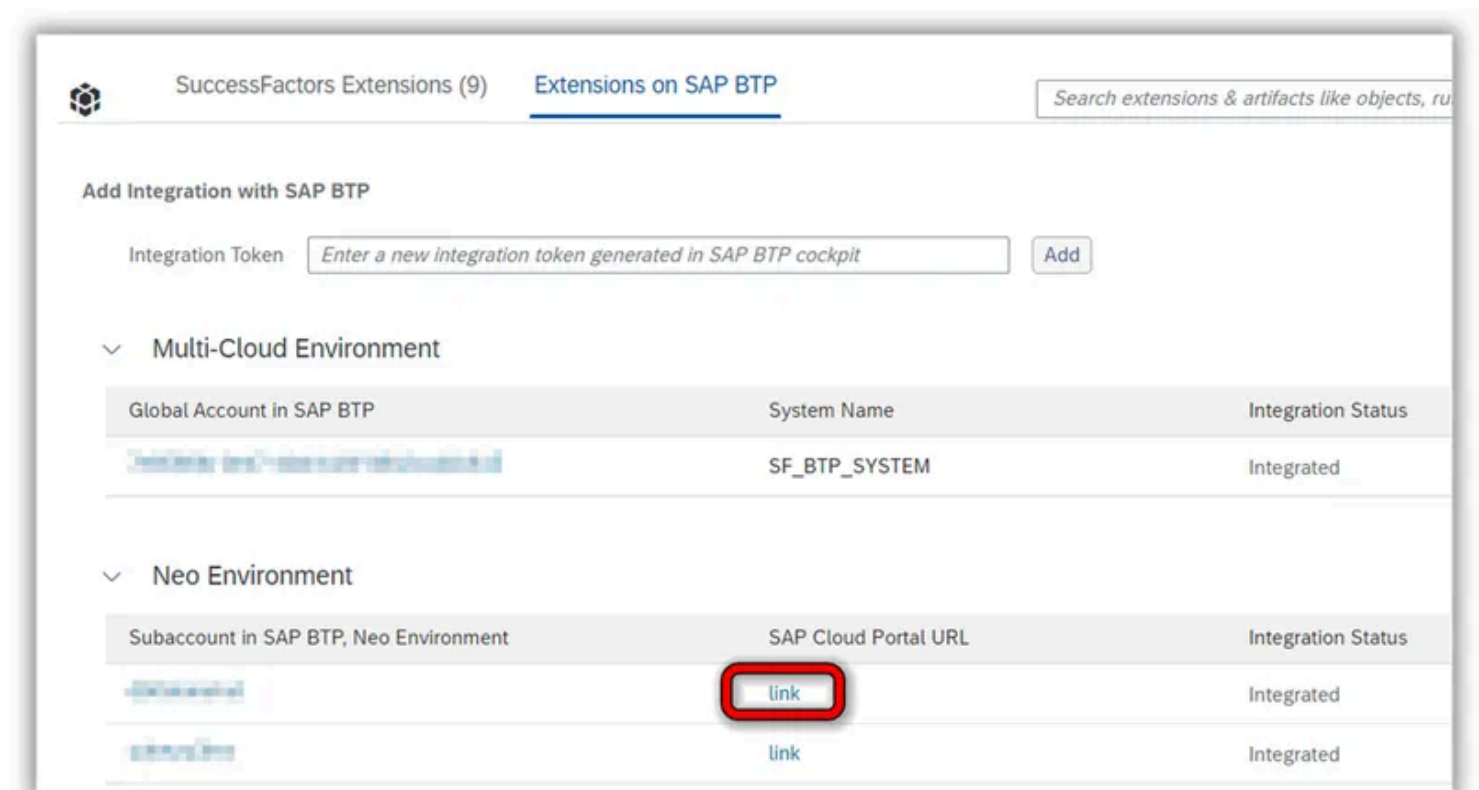
5.1.2.4. SuccessFactors App Setup

In this section, the HTML5 applications will be configured in SuccessFactors.

Log into SuccessFactors with an Administrative account and navigate to the **Extension Center**



Navigate to the '**Extensions on SAP BTP**' tab and click on the '**link**' under '**SAP Cloud Portal URL**'



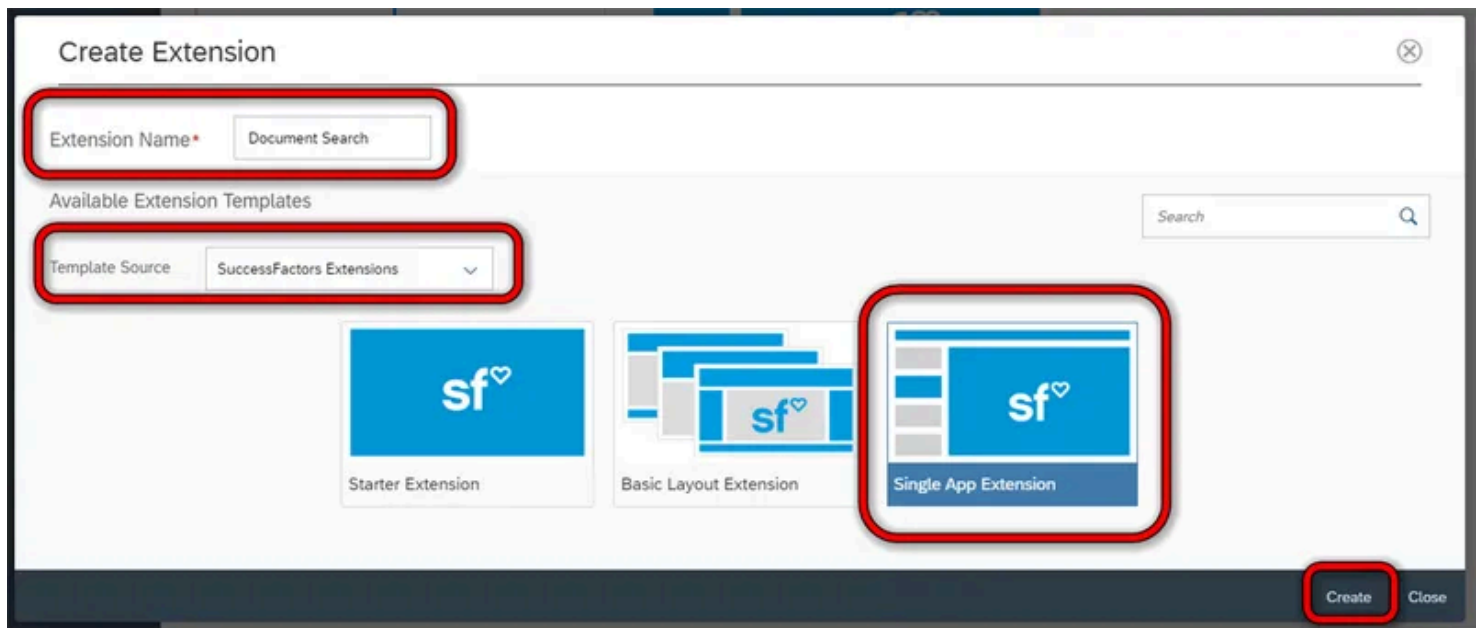
The SuccessFactors **Extension Directory** will open. Click on the giant plus button.



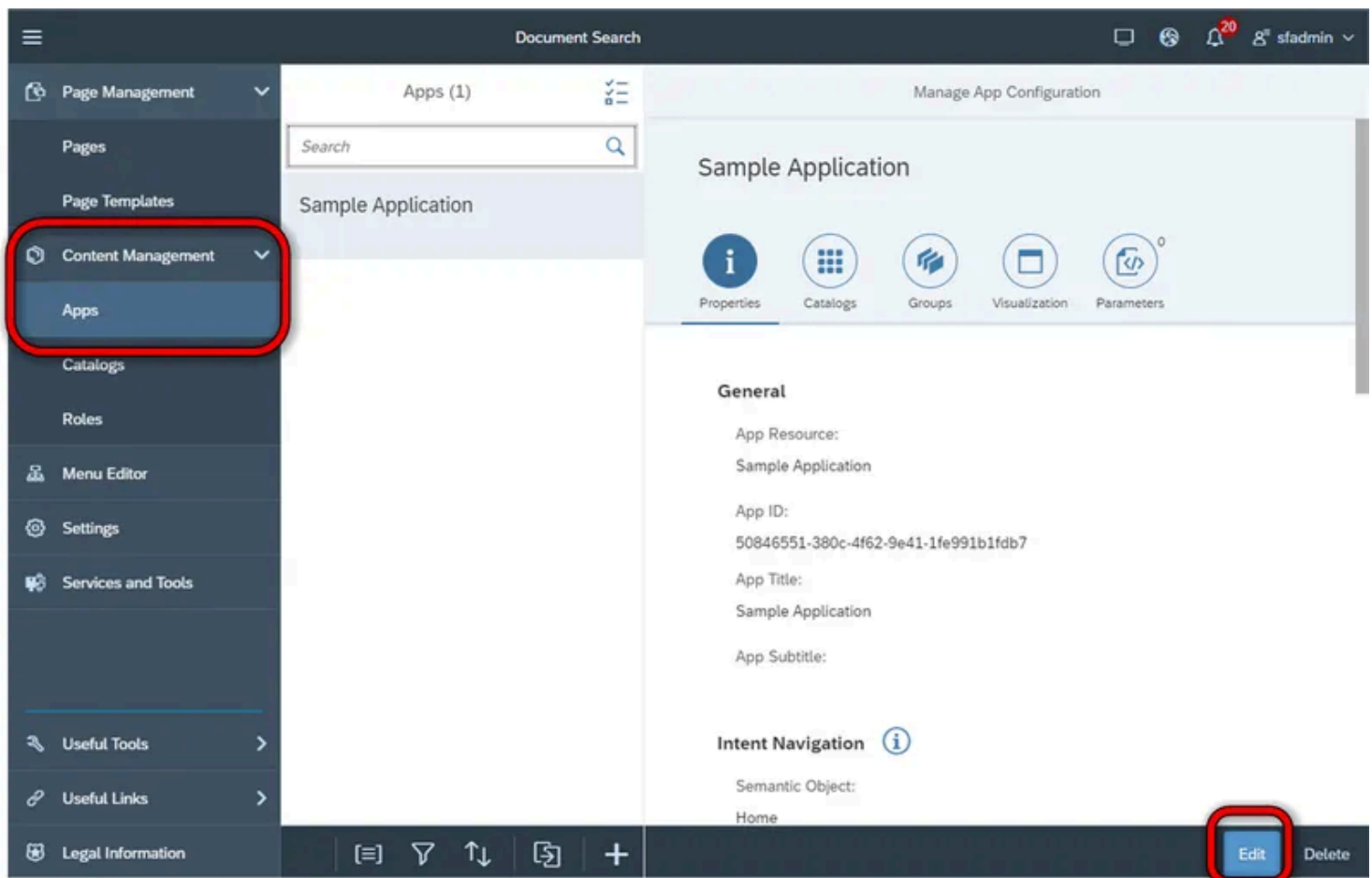
Set the following values for the new Extension and click the '**Create**' button

- Extension Name: **Document Search**

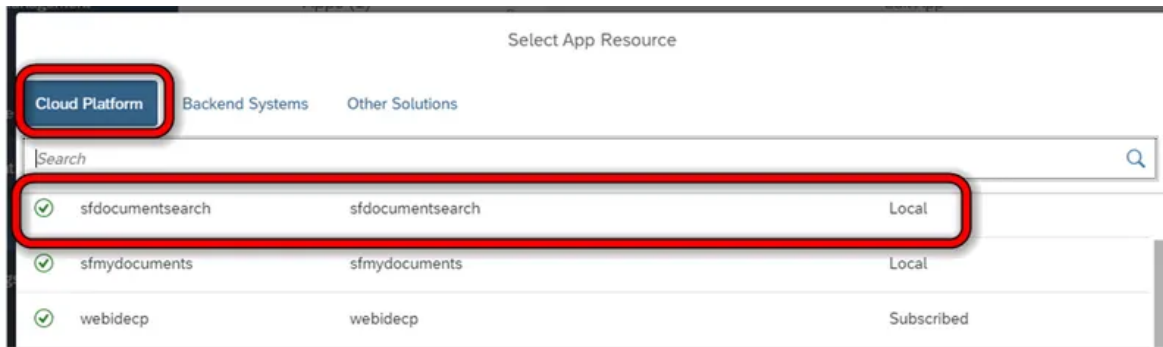
- Template Source: **SuccessFactors Extensions**
- Choose **Single App Extension**



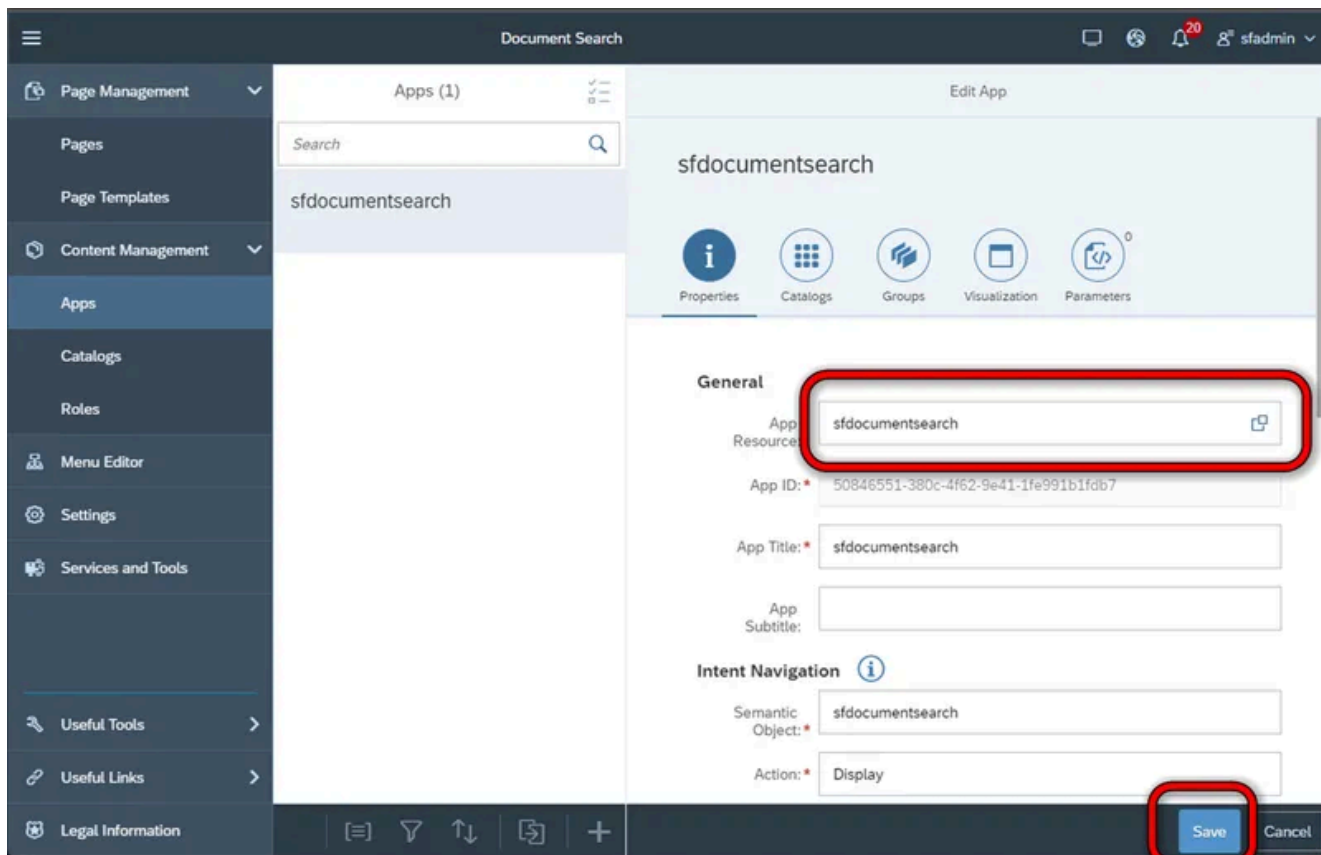
Once the Extension has been created, navigate to **Content Management** → **Apps** and click the **Edit** button



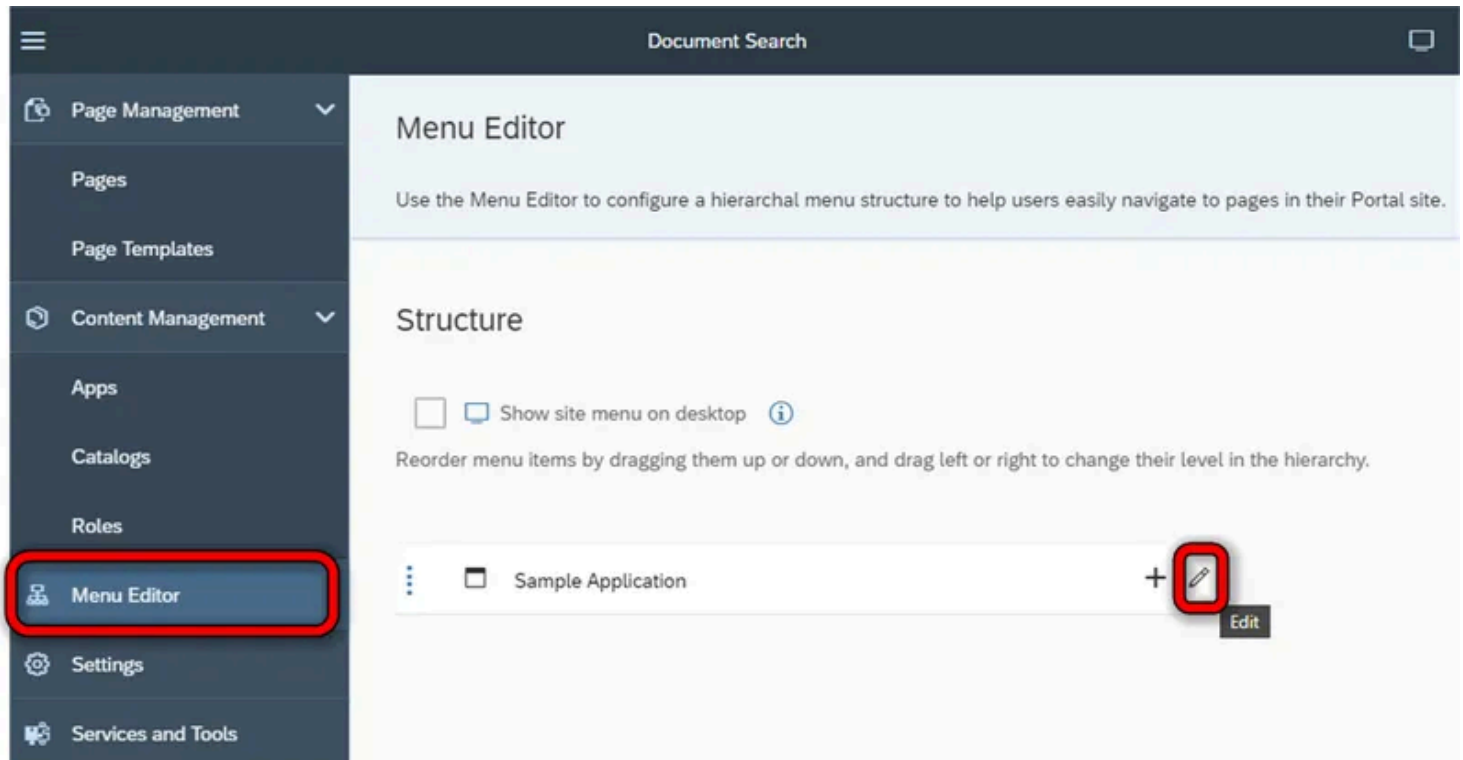
Choose the registered application from the BTP Platform (sfdocumentsearch)



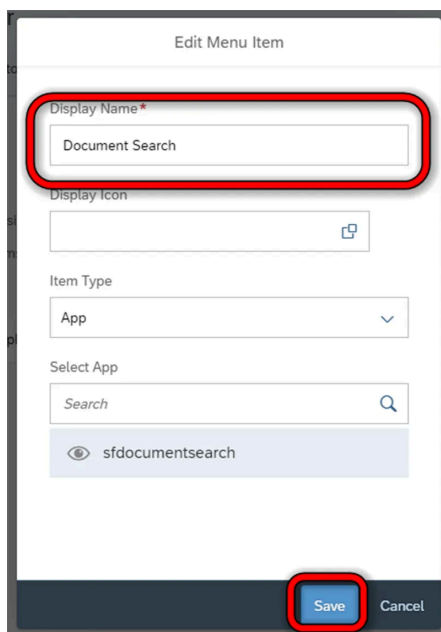
After selection the BTP Application click **SAVE**



Next, navigate to **'Menu Editor'** and click the **edit** pencil icon

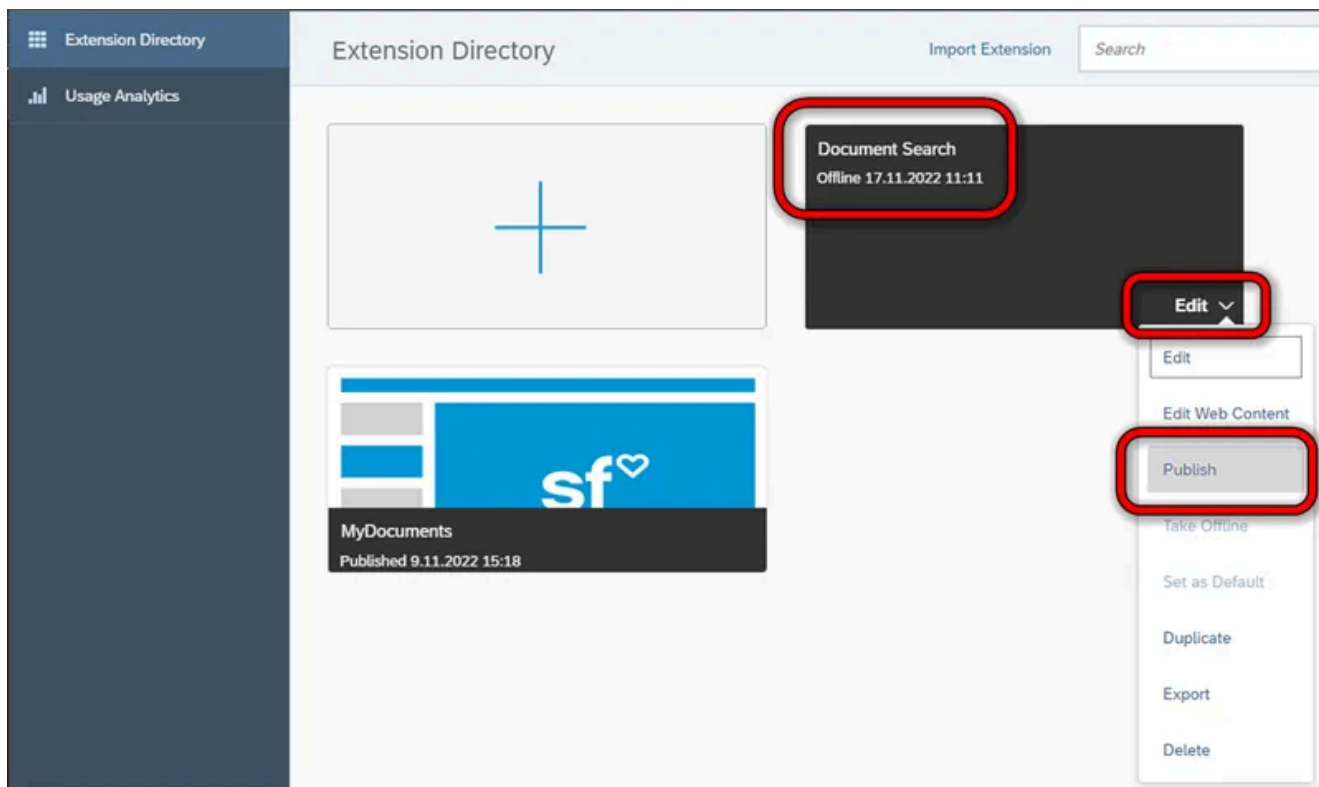


Set the display name to 'Document Search' and click the 'Save' button



After updating the menu display name, click the '**Save**' button in the lower right. You can now close the 'Menu Editor' browser tab and return to the '**Extension Directory**'

Select your new Extension, click the '**Edit**' button and select '**Publish**'



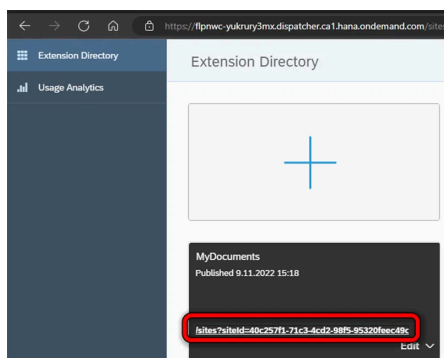
Custom Navigation Menu Setup

Next, register the Extension with the Custom Navigation in SuccessFactors

Navigate to the '**Configure Custom Navigation**' in SuccessFactors admin tools

Add a new line with the following values:

- Menu Item Label: Document Search
- ID: <next unused value>
- Link Type: External Link - New Tab/Window
- URL: Copy from the Extension Published



- Menu Location: <select where the menu should appear>
- Permissions: <select groups to see menu option>

NEXT Complete all of the Above Steps for the 'My Documents' application

5.1.3. SuccessFactors Configuration

This section describes the following configuration that must be completed in SuccessFactors.

5.1.3.1. API User

An API user is required for the docuflow for SuccessFactors middleware server.

Create a new user in SuccessFactors. This should be easily identifiable as an API user (i.e. apidflow)

Create a permission role for this user (i.e. apidflow_access)

The following permissions are required for the permission role:

Employee Data:

(Employee Profile)

- Country
- Email
- Employee Id
- Manager
- User ID
(User Information)
- Hire Date
(Employment Details)
- Employment Details MSS
- Hire Date

General User Permission:

- SFAPI User Login
 - User Login
- SAP System Configuration:
- View

Miscellaneous Permissions:

- Category Link Card 1 (view / edit)
- Category Link Card 3 (view / edit)
- Document Type Link Card 1 (view / edit)
- Document Type Link Card3 (view / edit)
- Employee Documents (view / view history / create / insert / correct / delete)
- Employee Documents.EmployeeDocumentChild (view / correct / create / adjust order / delete)
- HR Employee Documents (view / view history / create / insert / correct / delete)
- HR Employee Documents.HREmployeeDocumentChild (view / correct / create / adjust order / delete)
- Link Card Type 1 (View / Edit)
- Link Card Type 2 (View / Edit)
- Link Card Type 3 (View / Edit)
- Link Card Type 4 (View / Edit)
- cust_d4iFolderLink (View / Edit)
- SAP System Configuration (View)

Manage User

- User Account oData entity

Metadata Framework

- Admin access to MDF OData API

Manage Integration Tools

- oData API Attachment Export
- Allow Admin to Access OData API through Basic Authentication

Manage System Properties

- Picklist Management and Picklists Mappings Set Up

5.1.3.2. Create Pick List

The section will create the Pick List for document approval workflow status

Navigate to the Admin tool '**Pick List Center**'

Click '**New Pick List**'

In the Create Dialog, set the following values:

Name	Value	Notes
Code	LinkCard_DocStatus	
Name	Link Card Doc Status	
Status	Active	
Effective Start Date	<today's date>	
Display Order	Alpha	
Parent Picklist	<blank>	
Legacy Picklist ID	LinkCard_DocStatus	

Click the '**Save**' button

Create the following values:

Name	Value	Notes
External Code	1	
Label	InApproval	
Status	Active	
Non-unique External Code	1	

Name	Value	Notes
External Code	2	
Label	Rejected	
Status	Active	
Non-unique External Code	2	

Name	Value	Notes
External Code	3	
Label	Approved	
Status	Active	
Non-unique External Code	3	

5.1.3.3. Create Approval Workflow

The section will create the document approval workflow

Navigate to the Admin tool '**Manage Organization, Pay and Job Structures**'

Create New: Workflow

In the main workflow section, set the following values:

Name	Value	Notes
Workflow ID	d4i_WF	
Name	d4i_Approval	

In the section Step 1, set the following values:

Name	Value	Notes
Approver Type	Role	
Approver Role	Employee HR	
Context	Source	
Edit Transaction	Edit with Route Change	
No Approver Behaviour	Skip this Step	
Relationship to Approver	Employee	
Respect Permission	No	

5.1.3.4. SuccessFactors Objects

The following custom objects must be defined within the SuccessFactors administrative tools:

1. Category Link Card 1
2. Category Link Card 3
3. Document Type Link Card 1
4. Document Type Link Card 3
5. Link Card Type 1
6. Link Card Type 2
7. Link Card Type 3
8. Link Card Type 4
9. ESS Link Card Child
10. ESS Link Card Parent
11. HR Link Card Child
12. HR Link Card Parent
13. d4iFolderLink

NOTE: To access the administrative tools, navigate to '**Admin→Configure Object Definitions**'

5.1.3.4.1. Category Link Card 1

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	Category	
Effective Dating	None	
API Visibility	Editable	
Status	Active	
Label	Category Link Card 1	

In the '**Fields Section**', for '**external code**' select **details**

- In the details screen, set the value of '**Visibility**' to '**Not Visible**'
- NOTE: This prevents the UI from showing a count of the categories

In the '**Fields Section**', for '**external name**' select **details**

- In the details screen, set the value of '**Data Type**' to '**Translatable**'

Update the **Security Section** with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	<blank>	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	<blank>	

5.1.3.4.2. Category Link Card 3

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	Category_LinkCard3	
Effective Dating	None	
API Visibility	Editable	
Status	Active	
Label	Category Link Card 3	

In the '**Fields Section**', for '**external code**' select **details**

- In the details screen, set the value of '**Visibility**' to '**Not Visible**'

In the '**Fields Section**', for '**external name**' select **details**

- In the details screen, set the value of '**Data Type**' to '**Translatable**'

Update the **Security Section** with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	<blank>	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	<blank>	

5.1.3.4.3. Document Type Link Card 1

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	DocumentType_LinkCard1	
Effective Dating	None	
API Visibility	Editable	
MDF Version History	Complete History	
Status	Active	
Label	Document Type Link Card 1	

In the **Fields Section** for '**external code**' select **details**

- In the details screen, set the value of '**Visibility**' to '**Not Visible**'

In the '**Fields Section**', for '**external name**' select **details**

- In the details screen, set the value of '**Data Type**' to '**Translatable**'

In the **Associations** section set the following values:

Property	Value	Notes
Name	category	It will show as custom_category after its been created
Multiplicity	One to Many	
Destination Object	Category Link Card 1	Use the dropdown to find the custom object created for 'Category'
Type	Valid When	

Update the **Security** section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	<blank>	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	<blank>	

5.1.3.4.4. Document Type Link Card 3

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	DocumentType_LinkCard3	
Effective Dating	None	
API Visibility	Editable	
MDF Version History	Complete History	
Status	Active	
Label	Document Type Link Card 3	

In the **Fields Section** for '**external code**' select **details**

- In the details screen, set the value of '**Visibility**' to '**Not Visible**'

In the '**Fields Section**', for '**external name**' select **details**

- In the details screen, set the value of '**Data Type**' to '**Translatable**'

In the **Associations** section set the following values:

Property	Value	Notes
Name	CategoryLinkCard3	It will show as custom_CategoryLinkCard3 after its been created
Multiplicity	One to Many	
Destination Object	Category Link Card 3	Use the dropdown to find the custom object created for 'Category LinkCard3'
Type	Valid When	

Update the **Security** section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	<blank>	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	<blank>	

5.1.3.4.5. Link Card Type 1

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	d4i_LinkCard_1	
Effective Dating	None	
API Visibility	Editable	
Status	Active	
Label	Link Card Type 1	

Add the following fields to the d4i_LinkCard_1 object:

Name	Length	Data Type	More
user	100	user	Under details, set 'Required' to 'Yes'
Category	100	Generic Object	Valid Values Source: cust_Category_LinkCard3
DocumentType	100	Generic Object	Valid Values Source: cust_DocumentType_LinkCard3
Title	256	String	
UserID	38	String	
EmployeeName	256	String	
Country	128	String	
d4i_URL	256	String	
d4iUploadStatus	64	String	
CreatedDateTime	35	DateTime	
ownerId	38	String	
d4i_Status	64	String	

Update the Security section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	cust_user	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	mdfSystemEffectiveEndDate	

5.1.3.4.6. Link Card Type 2

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	d4i_LinkCard_2	
Effective Dating	None	
API Visibility	Editable	
Status	Active	
Label	Link Card Type 2	

Add the following fields to the d4i_LinkCard_2 object:

Name	Length	Data Type	More
user	100	user	Under details, set 'Required' to 'Yes'
Category	100	Generic Object	Valid Values Source: cust_Category_LinkCard3
DocumentType	100	Generic Object	Valid Values Source: cust_DocumentType_LinkCard3
Title	256	String	
UserID	38	String	
EmployeeName	256	String	
Country	128	String	
d4i_URL	256	String	
d4iUploadStatus	64	String	
CreatedDateTime	35	DateTime	
ownerId	38	String	
d4i_Status	64	String	

Update the Security section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	cust_user	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	mdfSystemEffectiveEndDate	

5.1.3.4.7. Link Card Type 3

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	d4i_LinkCard_3	
Effective Dating	None	
API Visibility	Editable	
Status	Active	
Label	Link Card Type 3	

Add the following fields to the d4i_LinkCard_3 object:

Name	Length	Data Type	More
user	100	user	Under details, set 'Required' to 'Yes'
Category	100	Generic Object	Valid Values Source: cust_Category_LinkCard3
DocumentType	100	Generic Object	Valid Values Source: cust_DocumentType_LinkCard3
Title	256	String	
UserID	38	String	
EmployeeName	256	String	
Country	128	String	
d4i_URL	256	String	
d4iUploadStatus	64	String	
CreatedDateTime	35	DateTime	
ownerId	38	String	
d4i_Status	64	String	

Update the Security section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	cust_user	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	mdfSystemEffectiveEndDate	

5.1.3.4.8. Link Card Type 4

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	d4i_LinkCard_4	
Effective Dating	None	
API Visibility	Editable	
Status	Active	
Label	Link Card Type 4	

Fields

Add the following fields to the d4i_LinkCard_4 object:

Name	Length	Data Type	More
user	100	user	Under details, set 'Required' to 'Yes'
Category	100	Generic Object	Valid Values Source: cust_Category_LinkCard3
DocumentType	100	Generic Object	Valid Values Source: cust_DocumentType_LinkCard3
Title	256	String	
UserID	38	String	
EmployeeName	256	String	
Country	128	String	
d4i_URL	256	String	
d4iUploadStatus	64	String	
CreatedDateTime	35	DateTime	
ownerId	38	String	
d4i_Status	64	String	

Security

Update the Security section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	cust_user	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	mdfSystemEffectiveEndDate	

5.1.3.4.9. ESS Link Card Child

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	d4i_LinkCard_ESS_Child	
Effective Dating	From Parent	
API Visibility	Editable	
Status	Active	
Label	Employee Document Child	
MDF Version History	Complete History	
Subject User Field		
Workflow Routing		
Pending Data	No	

Fields

In the '**Fields Section**', for '**externalName**' select **details**

- In the details screen, set the value of 'Visibility' to 'Not Visible'

Set the following fields on the cust_d4i_LinkCard_ESS_Child:

Name	Length	Data Type	More
externalCode	100	AutoNumber	This should exist, set type
Category	100	Generic Object	In Details , Set Valid Values Source to cust_category Set label to Document Category Set Required to Yes
DocumentType	100	Generic Object	In Details , Set Valid Values Source to cust_DocumentType_LinkCard1 Set Label to Document Type Also in Details, for Field Criteria, set 'Source Field Name' to cust_Category.mdfSystemInternalCode and 'Destination Field Name' to cust_Category Set Required to Yes
CollectionType	256	String	
d4iUploadStatus	38	Picklist	In Details , Set Valid Values Source to LinkCard_DocStatus
attachment	38	Attachment	Set Required to Yes

Security

Update the Security section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field		
CREATE Respects Target Criteria	No	
Base Date Field For Blocking		

Business Rule

The next step is to create a rule that will concatenate Category and Document Type into a single field that is used for data collection

Click on the plus next to 'Save Rules' to open '**Configure Business Rules**'

Select 'Basic'

Name	Value	Notes
Rule Name	LinkCard_CollectionType_Child	
Rule ID	LinkCard_CollectionType_Child	
Base Object	Employee Document Child	

In the **IF** statement section, select the '**Always True**' checkbox

Set the following Concatenation Rule:

Configure Business Rules

Search :Rule No Selection

LinkCard_CollectionType_Child (LinkCard_CollectionType_Child)

Scenario: Basic

Basic Information

Start Date 01/01/1900

Rule Type

Description

Parameters

Name	Object
Context	System Context 
Employee Document Child	Employee Document Child

 Variables If

 This rule is always true.
To add an expression please uncheck the Always True checkbox.

Then

Concatenate string

Save your changes to complete the rule

Navigate back to the Admin tool Configure Object Definitions

Open the ESS Link Card object and take action→make correction

Go to the section 'Save Rule'

- Select the **LinkCard_CollectionType_Child** rule that was just created and save the object

5.1.3.4.10. ESS Link Card Parent

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	d4i_LinkCard_ESS_Parent	
Effective Dating	Basic	
API Visibility	Editable	
Status	Active	
Label	Employee Documents	
MDF Version History	Complete History	
Default Screen	Employee Document Parent	
Subject User Field	externalCode	
Workflow Routing	d4i_Approval	
Pending Data	Yes	

Fields

Set the following fields on the cust_d4i_LinkCard_ESS_Parent object:

Name	Length	Data Type	More
externalCode	100	user	This should exist, set to user type Set Required: True

Associations

In the Associations section, add the following values:

Name	Multiplicity	Destination Object	Type
cust_EmployeeDocumentChild	One to Many	Employee Document Child	Composite

Security

Update the Security section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	externalCode	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	mdfSystemEffectiveEndDate	

5.1.3.4.11. HR Link Card Child

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	d4i_LinkCard_HR_Child	
Effective Dating	From Parent	
API Visibility	Editable	
Status	Active	
Label	HR Employee Document Child	
MDF Version History	Complete History	
Subject User Field		
Workflow Routing		
Pending Data	No	

Fields

In the '**Fields Section**', for '**externalName**' select **details**

- In the details screen, set the value of 'Visibility' to 'Not Visible'

Set the following fields on the cust_d4i_LinkCard_HR_Child:

Name	Length	Data Type	More
externalCode	100	AutoNumber	This should exist, set type
Category	100	Generic Object	In Details , Set Valid Values Source to cust_Category_LinkCard3 Set label to Document Category (set localization) Set Required to Yes
Document Type	100	Generic Object	In Details , Set Valid Values Source to cust_DocumentType_LinkCard3 Set Label to Document Type Also in Details, for Field Criteria, set 'Source Field Name' to cust_CategoryLinkCard3.mdfSystemInternalCode and 'Destination Field Name' to cust_Category Set Required to Yes
CollectionType	256	String	
d4iUpload Status	38	Picklist	In Details , Set Valid Values Source to LinkCard_DocStatus
attachment	38	Attachment	Set Required to Yes

Security

Update the Security section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field		
CREATE Respects Target Criteria	No	
Base Date Field For Blocking		

Business Rule

The next step is to create a rule that will concatenate Category and Document Type into a single field that is used for data collection

Click on the plus next to 'Save Rules' to open '**Configure Business Rules**'

Select '**Basic**'

Name	Value	Notes
Rule Name	HR_LinkCard_CollectionType_Child	
Rule ID	HR LinkCard_CollectionType_Child	
Base Object	HR Employee Document Child	

In the **IF** statement section, select the '**Always True**' checkbox

Set the following Concatenation Rule:

If

*This rule is always true.
To add an expression please uncheck the Always True checkbox.*

Then

Concatenate()

Concatenate string

- Sub string: ESS Link Card.Category Value.externalName
- Sub string: Text -
- Sub string: ESS Link Card.Document Type Value.externalName

Save your changes to complete the rule

Navigate back to the Admin tool Configure Object Definitions

Open the ESS Link Card object and take action→make correction

Go to the section 'Save Rule'

- Select the **HR_LinkCard_CollectionType_Child** rule that was just created and save the object

5.1.3.4.12. HR Link Card Parent

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	d4i_LinkCard_HR_Parent	
Effective Dating	Basic	
API Visibility	Editable	
Status	Active	
Label	HR Employee Documents	Localize
MDF Version History	Complete History	
Default Screen	HR Employee Documents	
Subject User Field	externalCode	
Workflow Routing	d4i_Approval	
Pending Data	Yes	

Fields

Set the following fields on the d4i_LinkCard_HR_Parent object:

Name	Length	Data Type	More
externalCode	100	user	This should exist, set to user type Required: True

Associations

In the Associations section, add the following values:

Name	Multiplicity	Destination Object	Type
HREmployeeDocumentChild	One to Many	HR Employee Document Child	Composite

Security

Update the Security section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	externalCode	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	mdfSystemEffectiveEndDate	

5.1.3.4.13. d4iFolderLink

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section

Property	Value	Notes
Code	d4iFolderLink	
Effective Dating	None	
API Visibility	Editable	
Status	Active	
Label	d4iFolderLink	

Fields

Set the following fields on the cust_d4iFolderLink object:

Name	Length	Data Type	More
EcmFolderID	64	String	
User	100	user	Required = True

Searchable Fields

Update the Security section with the following values:

Field	Status	Notes
cust_User	Active	
cust_BoxFolderID	Active	

Security

Update the Security section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	cust_User	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking		

5.1.3.5. Permission Provisioning

Navigate to the Admin tool '**Manage Permission Roles**'

Employee Self Service Permissions to Objects

Apply Document Upload to People Profile

- Select 'ESS' Role
- Select 'Permission...'
- Select 'Employee Views'
- Check 'Document Upload'

Apply Permissions to Objects

- Select 'ESS'
- Select 'Permission...'
- Select 'Miscellaneous Permissions'
- Set the following Values:

Name	Setting	Notes
Category Link Card 1	View	
Document Type Link Card1	View	
Employee Documents	View / Create	
Employee Documents.EmployeeDocument Child	View Current / Create	Field Overrides Collection Type Value : No Access d4iUploadStatus: No Access
Link Card Type 1	View	
Link Card Type 4	View	

Manager Self Service Permissions to Objects

- Select 'MSS'
- Select 'Permission...'
- Select 'Miscellaneous Permissions'
- Set the following Values:

Name	Setting	Notes
Category Link Card 3	View	
Document Type Link Card3	View	
Link Card Type 1	View	
Link Card Type 2	View	

HRBP Permissions to Objects

- Select 'HRBP Role'
- Select 'Permission...'
- Select 'Miscellaneous Permissions'
- Set the following Values:

Name	Setting	Notes
Category Link Card 1	View	
Category Link Card 3	View	
Document Type Link Card 1	View	
Document Type Link Card 3	View	
HR Employee Documents	View Current / View History / Create / Correct / Delete	
HR Employee Documents.HREmployeeDocumentChild	View / Correct / Create / Delete	Field Overrides Collection Type Value : No Access d4iUploadStatus: No Access
Link Card Type 1	View / Edit	
Link Card Type 2	View / Edit	
Link Card Type 3	View / Edit	
Link Card Type 4	View / Edit	

Client Administrator Permissions to Objects

- Select 'Client Administrators Role'
- Select 'Permission...'
- Select 'Miscellaneous Permissions'
- Set the following Values:

Name	Setting	Notes
Category Link Card 1	View	
Category Link Card 3	View	
Document Type Link Card 1	View	
Document Type Link Card 3	View	
HR Employee Documents	View Current / View History / Create / Correct / Delete	
HR Employee Documents.HREmployeeDocumentChild	View / Correct / Create / Delete	Field Overrides Collection Type Value : No Access d4iUploadStatus: No Access
Link Card Type 1	View / Edit	
Link Card Type 2	View / Edit	
Link Card Type 3	View / Edit	
Link Card Type 4	View / Edit	

5.1.3.6. Export Category and Document Type

Use these instructions to get a set of CSV files to use to import the Categories and Document types into the system

For Employee Document Upload Rules/Categories:

Go to **'Import and Export Data'**

For Categories:

- Select the action to perform: Download Template
- Select Generic Object: Category Link Card 1
- Include dependencies: No
- Click 'Download'

For Document Type LinkCard1:

- Select the action to perform: Download Template
- Select Generic Object: Document Type LinkCard 1
- Include dependencies: No
- Click 'Download'

For the Relationship between Category and Document Types for Link Card 1

- Select the action to perform: Download Template
- Select Generic Object: Document Type LinkCard 1 - Category
- Include dependencies: No
- Click 'Download'

For HR Document Upload Rules/Categories:

Go to **'Import and Export Data'**

For Categories:

- Select the action to perform: Download Template
- Select Generic Object: Category
- Include dependencies: No
- Click 'Download'

For Document Type LinkCard3:

- Select the action to perform: Download Template
- Select Generic Object: Document Type LinkCard 3
- Include dependencies: No
- Click 'Download'

For the Relationship between Category and Document Types for Link Card 3

- Select the action to perform: Download Template
- Select Generic Object: Document Type LinkCard 3 - Category

- Include dependencies: No
- Click 'Download'

5.1.3.7. Import Category and Document Types

Use these instructions to use a set of CSV files to use to import the Categories and Document types into the system

For Employee Document Upload Rules/Categories:

Go to **'Import and Export Data'**

For Categories:

- Select the action to perform: Import Data
- Select Generic Object: Category
- File: CSV with your Category Data
- Purge Type: Incremental Load
- Click 'Import'

To check the imported data:

Go to **'Manage Data'**

- Under **'Search'** select Category
- Under the selection dropdown, the values should be shown

Complete the above steps for:

- Category Link Card 1.csv
- Category Link Card 3.csv
- Document Type Link Card 1.csv
- Document Type Link Card 3.csv
- Document Type Link Card 1-category.csv
- Document Type Link Card3-CategoryLinkCard3.csv

5.1.3.8. UI Configuration

This step will place the 'Upload Documents' portal option for the Workflow approval in the Employee File UI interface

Configure UI Element

Navigate to the admin tool '**Manage Configuration UI**'

Employee Documents

Create **Employee Documents**

- Click on 'Create New'
- Select '**Employee Documents**' in the Select Base Object
- Set the ID to '**Employee Documents Parent**'
- Click SAVE

Complete the following configuration:

- Employee Documents Header Element : Title Type User Defined = Employee Documents: (NOTE Set Translations as well)
- externalCode : Visible = No
- externalName : Visible = No
- effectiveStartDate : Visible = No
- In the Child Documents Header Section
 - Document Header : Title Type User Defined = Document (NOTE Set Translations as well)
 - Visible Fields Count = 8
 - Set Order of Properties: Document Category, Document Type, attachment
 - Hide the other fields

HR Employee Documents

Create **HR Employee Documents**

- Click on 'Create New'
- Select '**HR Employee Documents**' in the Select Base Object
- Set the ID to '**HR Employee Documents Parent**'
- Click SAVE

Complete the following configuration:

- HR Employee Documents Header Element : Title Type User Defined = HR Employee Documents: (NOTE Set Translations as well)
- externalCode : Visible = No
- externalName : Visible = No
- effectiveStartDate : Visible = No
- In the Child Documents Header Section
 - Document Header : Title Type User Defined = Document (NOTE Set Translations as well)
 - Visible Fields Count = 8

- Set Order of Properties: Document Category, Document Type, attachment
 - Hide the other fields

Configure People Profile

Navigate to the admin tool '**Configure People Profile**'

Click on '**Add New Section**' to insert a new UI section into the Employee Profile Screen

- Name the Section ***Document Upload***
- Name the Sub-Section: ***Documents***

Click on the Left side area

- Drop a 'Live Profile MDF Information' block
- Set the **MDF Screen ID** to '***Employee Document Parent***'

Click on the Right side area

- Drop a 'Live Profile MDF Information' block
- Set the **MDF Screen ID** to '***HR Employee Document Parent***'
- Save

5.1.4. docuflow for Integrations (d4i) Middleware Configuration

The goal of this guide is to establish base installation, initial configuration, and connectivity testing to a point that docuflow d4i components can communicate with one another and allows configuration of the software.

The docuflow application consists of a core Transformation engine and a series of connectors. Each connector has one or more connections defined which provide the information required to connect to the end applications.

NOTE: docuflow d4i was previously known as TRex and there are still occasional references to this name in the product

5.1.4.1. Start & Stop Order

d4i can be started and stopped via command line or as services (if installed as service)

Start Order

1. connectors (CNN-* eg. CNN-FS) (*no dependency*)
2. datastore (DAO)
3. config manager (CFGMGR)
4. transform engine (NGIN)

Stop Order

1. connectors (CNN-* eg. CNN-FS)
2. config manager (CFGMGR)
3. transform engine (NGIN)
4. datastore (DAO)

5.1.4.2. docuflow Middleware Installation

5.1.4.2.1. Unpacking ZIP Package

The docuflow middleware is delivered as a simple compressed 'zip' package. To install docuflow, simply unpack the zip file to a location you would like on the filesystem. For example d:\d4i

docuflow requires a java runtime environment (JRE) to run. By default, docuflow will reference the JRE deployed into the JDK directory in the location docuflow was installed. For example, if docuflow was installed to d:\d4i the JRE should be located at d:\d4i\JRE.

Please see '**Minimum Requirements**' in the technical overview for supported JRE versions.

5.1.4.2.2. Initial Configuration

The base docuflow installation will come pre-configured with the core system, Filesystem, Box and SuccessFactors connectors. The main task to complete for initial setup is the configuration of connections for these connectors.

The internal docuflow connectors and core components communicate through HTTPS/TLS encrypted connections. To enable these secure connections, x.509 certificates must be supplied. docuflow provides mechanisms to generate these certificates.

Options when generating certificates include:

- Configuration of which connector certificates to create
- Configuration of the server IP address/DNS entry when remote connectors are required
- Creation of passwords for certificates generated

Settings are configured in the file `<install location>\config\trex-certmgr-application.json`

The `trex-certmgr-application.json` file contains the following sections:

"defaults"

"defaults" : this section allows settings that will be used for all certificates generated. This can be overridden in the specifics of each individual certificate section

"master"

"master": This is the section for the master keystore for all configured connectors on the system. The default settings will be sufficient for a one box installation.

For each connector, the value for **"disabled"** should be set to **'false'** to have a certificate generated.

If distributing a connector on a remote server, the **"san"** section can be used to define the remote server name and/or ip address.

An example snippet for the file system connector is shown below, including a "san" section.

```
{
  "disabled": false,
  "alias": "trex.cnn-trex-fs.server",
  "dn": {
    "commonName": "VersaFile TRex File System Connector"
  },
  "san": {
    "dns": ["fully.qualified.name"],
    "ip": ["127.0.0.1"]
  }
}
```

"keystores"

"keystores": The keystores section is where docuflow configuration for the p12 keystore certificates is done. Each connector must have an alias entry in the section "keystore": **"trex.cnn.keystore.p12"**. The **"storepass"** entry is where a password for the keystore can be configured.

The section that will need configuration is the trex.cnn.keystore.p12. This contains references to each of the connectors being used.

"truststores"

"truststores": The truststores section is where docuflow configuration for the p12 truststore certificates is done. These trusts allow the connectors and core docuflow components to exchange their certificates.

The **"storepass"** entry allows for configuration of the password for the truststore.

There are two sections that will need configuration:

- trex.ngin.truststore.p12 (allows connections to the core d4i transformation engine)
- trex.cfgmgr.truststore.p12 (allows connections to the d4i configuration UI engine)

Both of these sections contain references to each of the connectors being used. **Note:** this should match the list from the keystore section.

An example of this section is shown below, with 5 connectors being setup:

```
"truststores": [
  {
    "keystore": "trex.dao.truststore.p12",
    "storepass": "<password>",
    "aliases": [ "trex.cfgmgr.server", "trex.ngin.server" ]
  },
  {
    "keystore": "trex.ngin.truststore.p12",
    "storepass": "<password>",
    "aliases": [
      "trex.dao.server",
      "trex.cfgmgr.server",
      "trex.cnn-trex-fs.server",
      "trex.cnn-cm8.server",
      "trex.cnn-elst.server",
      "trex.cnn-trex-util.server",
      "trex.cnn-ibm-p8ce.server"
    ]
  },
  {
    "keystore": "trex.cfgmgr.truststore.p12",
    "storepass": "<password>",
    "aliases": [
      "trex.cfgmgr.client",
      "trex.dao.server",
      "trex.ngin.server",
      "trex.cnn-trex-fs.server",
      "trex.cnn-cm8.server",
      "trex.cnn-elst.server",
      "trex.cnn-trex-util.server",
      "trex.cnn-ibm-p8ce.server"
    ]
  },
  {
    "keystore": "trex.cnn.truststore.p12",
    "storepass": "<password>",
    "aliases": [ "trex.cfgmgr.server", "trex.ngin.server" ]
  }
],
```

"exports"

"exports": The exports section configures the exported certificates for the system. Of note is the **"privatekeys"** section. This will generate a .p12 certificate that can be imported into a web browser to allow connectivity to the docuflow configuration service. Set a storepass that will be used. NOTE: Here you can configure a separate certificate/password for different environments (DEV vs PROD etc).

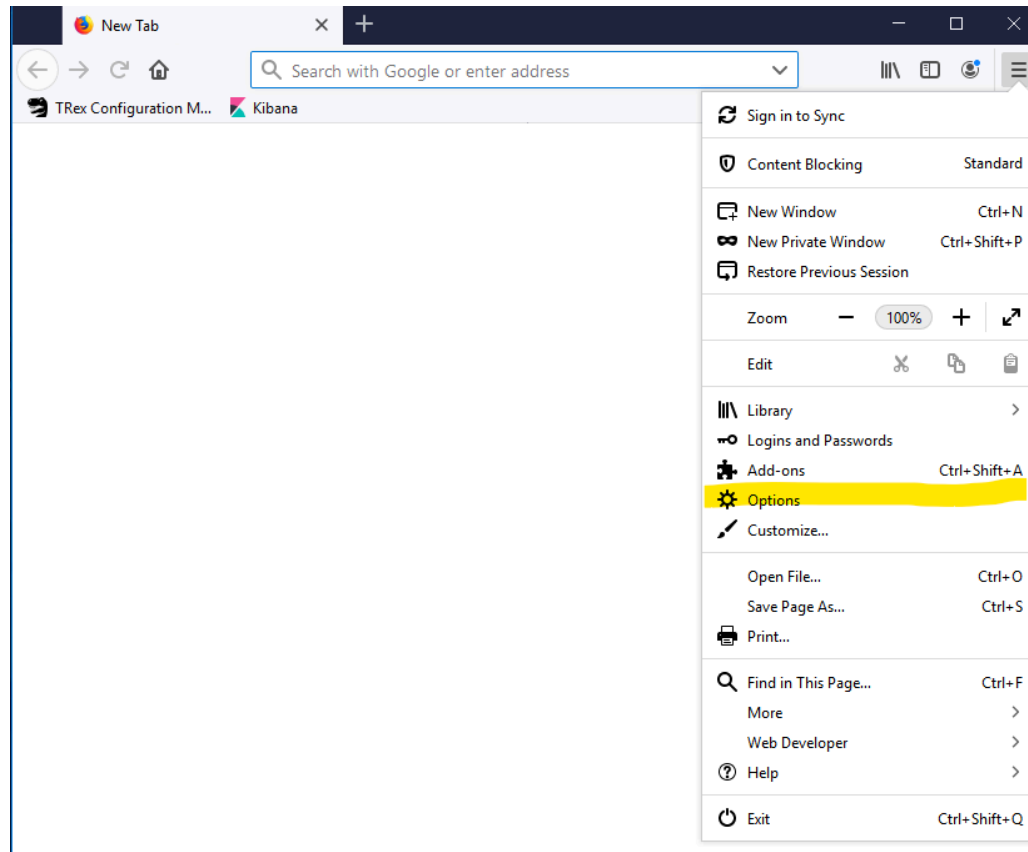
start-cert-mgr.ps1

Once configuration is complete, the powershell script '**<install location>start-cert-mgr.ps1**' should be run. This script will start a java application that will generate (or re-generate) the application certificates.

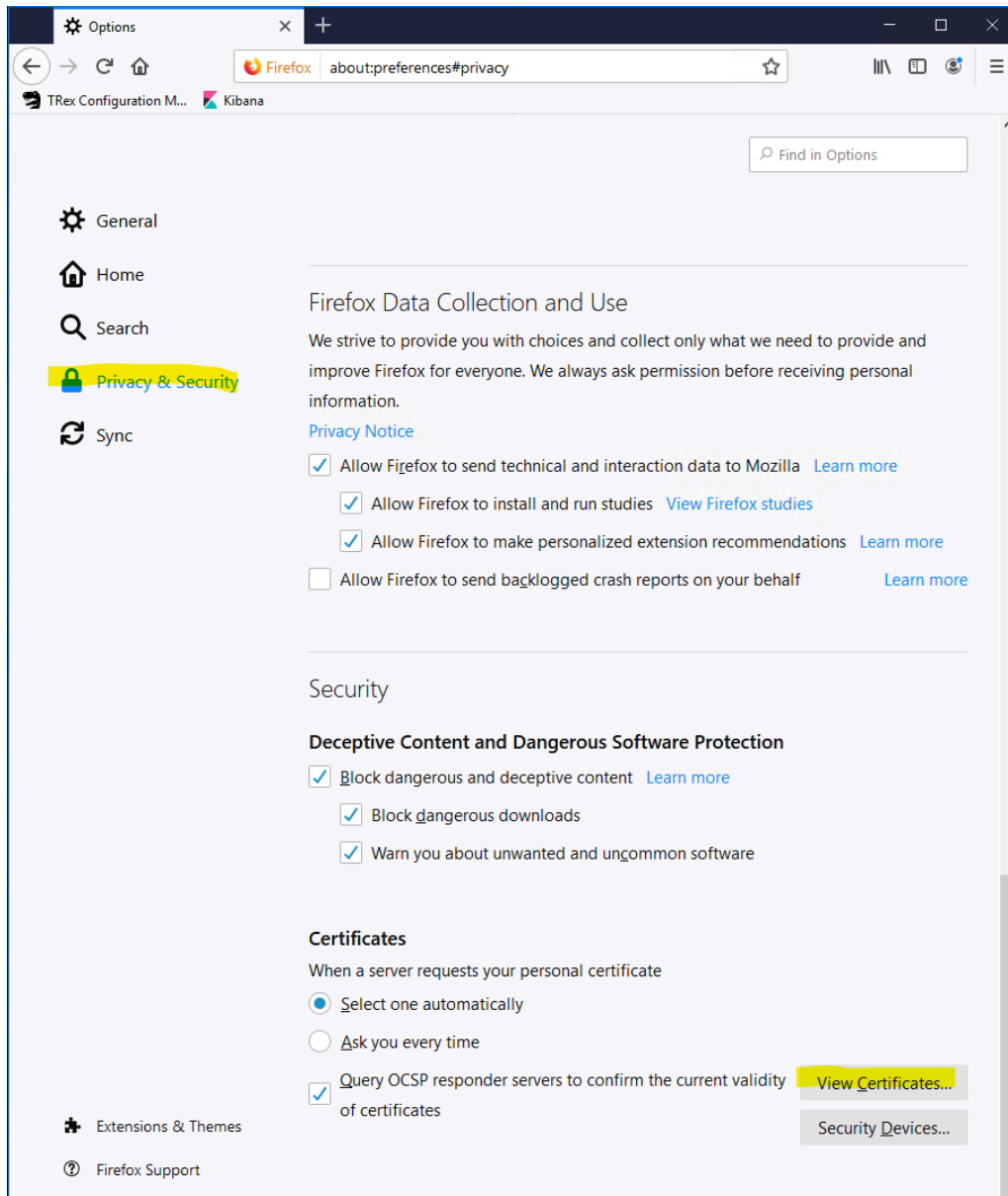
5.1.4.2.3. Setting up Firefox Certificate

The docuflow configuration service requires a HTTPS secured connection in order to communicate with a web browser. To enable this connection, a client certificate must be imported into the browser. This procedure explains how to install the certificate into Mozilla Firefox.

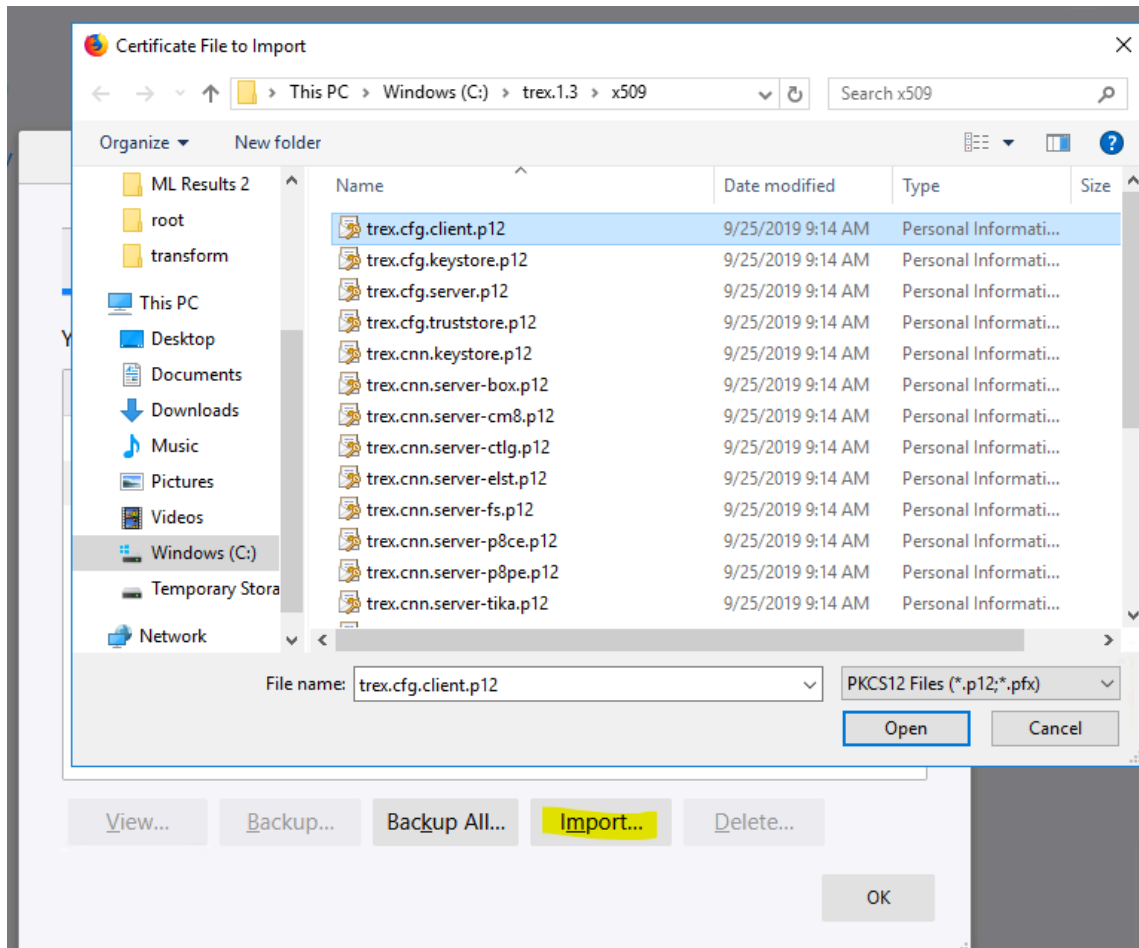
Open the Options section of Firefox



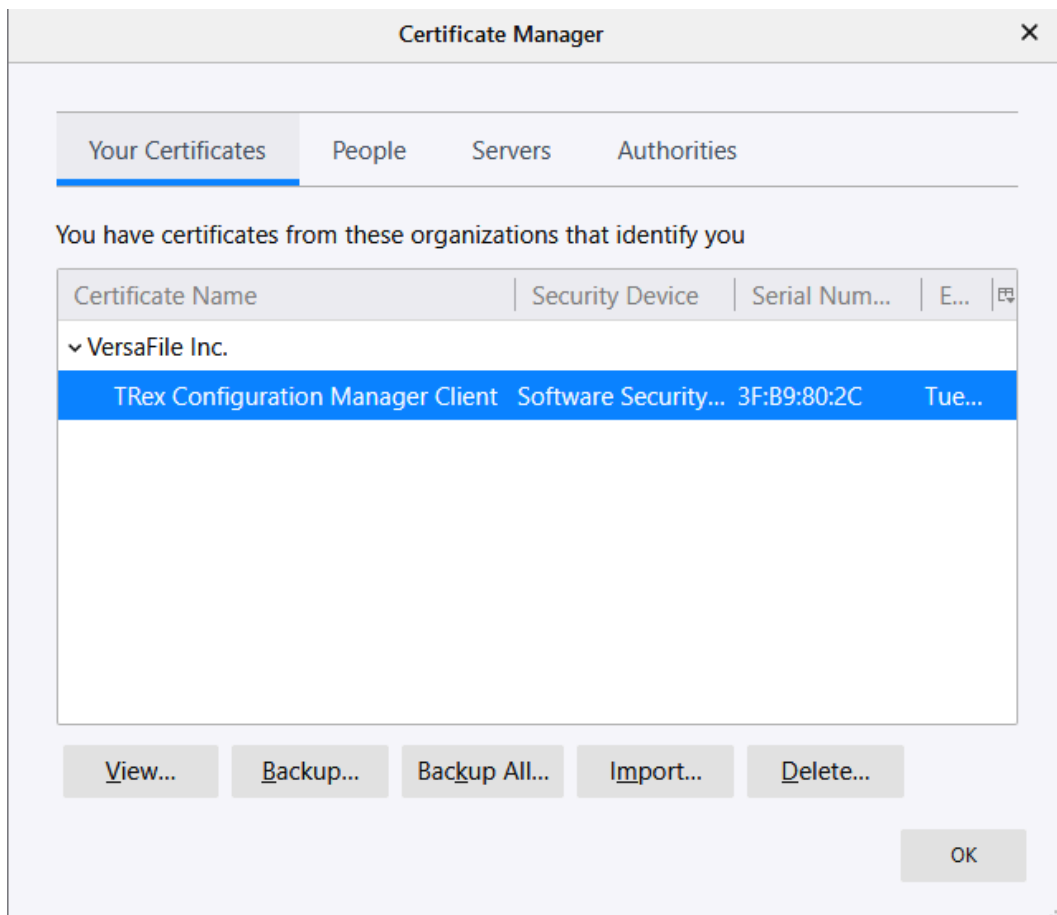
Select '**Privacy & Security**' section, scroll to the bottom and select View Certificates and then click '**Import**'



Select the 'exports' directory where you have created the x509 certificates (typically a directory called x509 in the root docuflow folder) and then select '**trex.cfgmgr.client.p12**' then supply the password that was configured in the '**trex-certmgr-application.json**' file.



You should see something similar to the following



5.1.4.2.4. Connecting to docuflow Configuration

Once all of the configuration steps are completed and the client certificate has been imported into your browser, Launch the windows power shell, go to File **<installocation>** , then run the powershell script **.\start-trex.ps1**.

If the file **<installocation>\start-trex.ps1** cannot be loaded, then the file **<installocation>\start-trex.ps1** is not digitally signed. And you cannot run this script on the current system. You will need to change the Execution policy to run these PowerShell scripts.

To display the execution policies for each scope in the order of precedence, use **"Get-ExecutionPolicy -List"**. To see the effective execution policy for your PowerShell session, use **"Get-ExecutionPolicy"** with no parameters.

The effective execution policy is determined by order of precedence as follows:

- MachinePolicy: Set by a Group Policy for all users of the computer.
- UserPolicy: Set by a Group Policy for the current user of the computer.
- Process: Affects only the current PowerShell session.
- CurrentUser: Affects only the current user.
- LocalMachine: Default scope that affects all users of the computer.

To change and set a new execution policy for the local computer for instance:

Use **Set-ExecutionPolicy** to specify the RemoteSigned policy. Then enter the Scope parameter to specify the default scope value, LocalMachine.

Set-ExecutionPolicy -ExecutionPolicy RemoteSigned -Scope LocalMachine

For more information on changing the execution policy, please visit <https://go.microsoft.com/fwlink/?LinkID=135170>.

After changing the execution policy, run **D:\docuflow> .\start-trex.ps1** A number of PowerShell windows will open up. Type "R" and run all scripts.

Once all scripts run, you can connect to the TRex Configuration Service UI. <https://<servername/localhost>:8443>

5.1.4.2.5. SSL Certificates Modifications

docuflow requires SSL certificates to secure communications amongst applications within the platform. docuflow does provide the ability to create its own self-signed certificates. Certificates are generated through the Certificate Manager application.

The following are execution guidelines for managing the certificates for docuflow .

Regenerate and Configure Certificates

After completing file modifications, such as those noted below, the following steps are required to regenerate and configure the new certificates for use with the docuflow platform.

1. Stop all docuflow applications
2. Rename (or delete) the <docuflow INSTALL LOCATION>\x509 directory
e.g. x509.old
3. Start the Certificate Manager application
 1. <docuflow INSTALL LOCATION>\start-cert-mgr.ps1
 2. Upon successful completion, exit/stop the Certificate Manager application
4. Start docuflow
5. Close and open a new browser instance
6. Import the newly generated certificates; follow the existing steps [here](#).
 1. Note: it not required, but recommended, to remove the previously configured certificates:
 1. Personal > VersaFile Configuration Manager Client
 2. Trusted Root Certification Authorities > VersaFile TRex Root CA

Modify the Expiry Length

- The expiry length for certificates is located within the <docuflow INSTALL LOCATION>\config\trex-certmgr-application.json file
 - within the defaults.validity object
 - default value 3650 (units in days; 10 years)
- To modify the expiry length of all certificates, edit the trex-certmgr-application.json file
 1. Modify the defaults.validity value
e.g. for 1 year expiry length

```
"validity": 365
```
 2. Save the file edit
 3. Execute the [Regenerate and Configure Certificates](#) steps, noted above.

Modify a Password

- Each application with the docuflow platform is created with its own truststore and keystore, to permit or restrict communication amongst applications.
- Passwords exist for each application's truststore and keystore. These passwords exist in:

- <docuflow INSTALL LOCATION>\config\trex-certmgr-application.json
- One of the following:
 - <docuflow INSTALL LOCATION>\config\application-cfgmgr.yaml
 - <docuflow INSTALL LOCATION>\config\application-dao.yaml
 - <docuflow INSTALL LOCATION>\config\application-ngin.yaml
 - <docuflow INSTALL LOCATION>\config\application-cnn.yaml
 - each connector has its own respective yaml config file; however, the application-cnn.yaml is inherited in each case as well
- A password configuration also exists for the client certificate, which exists in <docuflow INSTALL LOCATION>\config\trex-certmgr-application.json. Specifically:
 - exports.privatekeys.storepass object
- To modify a password, edit the trex-certmgr-application.json file and edit the respective yaml file, as required.
 - For example, to modify the password for both Configuration Manager certificates and the client certificate:

1. trex-certmgr-application.json edits:

1. Navigate to keystores.keystore : "trex.cfgmgr.keystore.p12"

2. Edit the storepass object, e.g.

```
{
  "keystore": "trex.cfgmgr.keystore.p12",
  "storepass": "newPassword1",
  "aliases": ["trex.cfgmgr.server"]
},
```

3. Navigate to truststores.truststore : "trex.cfgmgr.truststore.p12"

Edit the storepass object, e.g.

```
{
  "keystore": "trex.cfgmgr.truststore.p12",
  "storepass": "newPassword2",
  "aliases": [
    "trex.cfgmgr.client",
    ...
    "trex.cnn-trex-websvc.server"
  ]
},
```

4. Navigate to exports.privatekeys

Edit the storepass object, e.g.

```
"privatekeys": [{
  "alias": "trex.cfgmgr.client",
  "file": "exports/trex.cfgmgr.client.p12",
  "storepass": "newPassword3"
}]
```

2. application-cfgmgr.yaml edits

1. Navigate to and edit the server.ssl object

2. Edit the key-store-password and trust-store-password values, e.g.

```
server:
  port: 8443
```

```

ssl:
  key-store: ./x509/trex.cfgmgr.keystore.p12
  key-store-password: newPassword1
  key-alias: trex.cfgmgr.server
  trust-store: ./x509/trex.cfgmgr.truststore.p12
  trust-store-password: newPassword2

```

3. Execute the [Regenerate and Configure Certificates](#) steps, noted above.

◦ Password creation guidelines:

- It is recommended not to use, but if using the \ or " characters, these must be escaped with the \ character within the json and yaml files, e.g.
 - not escaped: *this"isaPwd*
 - escaped: *this\"is||aPwd*
- Although not used by default, it is recommended to surround the passwords in the yaml file with double quotes. e.g.

```

server:
  port: 8443
  ssl:
    key-store: ./x509/trex.cfgmgr.keystore.p12
    key-store-password: "newPassword1"
    key-alias: trex.cfgmgr.server
    trust-store: ./x509/trex.cfgmgr.truststore.p12
    trust-store-password: "newPassword2"

```

Modify Access to Configuration Manager

- By default, Configuration Manager is configured as accessible locally through a browser through `http://localhost:8443`
- An IP or DNS configuration can be added and included in the certificate creation.

1. Edit the `trex-certmgr-application.json` file
2. Navigate to the `master.certificates` object
3. Edit the `"alias": "trex.cfgmgr.server"` object

1. For a new DNS entry, add in a new entry in the `"dns"` array. For example, to add a DNS config with a name of *trex.cfgmgrtest.com* with an IP of *188.77.166.88*

```

{
  "alias": "trex.cfgmgr.server",
  "dn": {
    "commonName": "VersaFile Configuration Manager Application"
  },
  "san": {
    "dns": ["localhost", "trex.cfgmgrtest.com"],
    "ip": ["127.0.0.1", "188.77.166.88"]
  }
},

```

Note: an IP can be added on its own, without a DNS entry

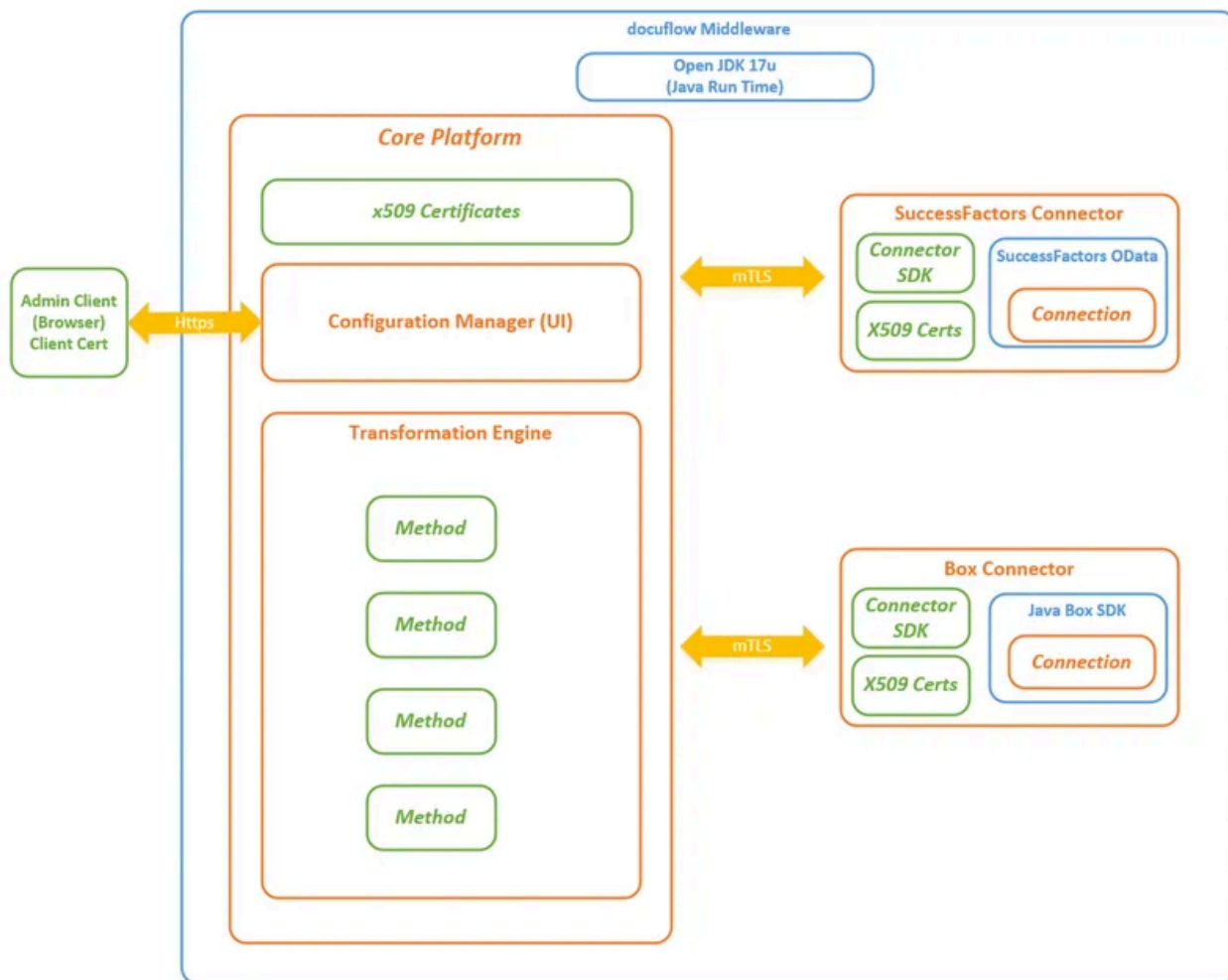
4. Execute the [Regenerate and Configure Certificates](#) steps, noted above.

5.1.4.3. docuflow Base Configuration

The Configuration of docuflow falls into four key components:

1. Engines
2. Connectors
3. Connections
4. Transformations

For docuflow for SuccessFactors and Box, the following architecture and connectors are used:



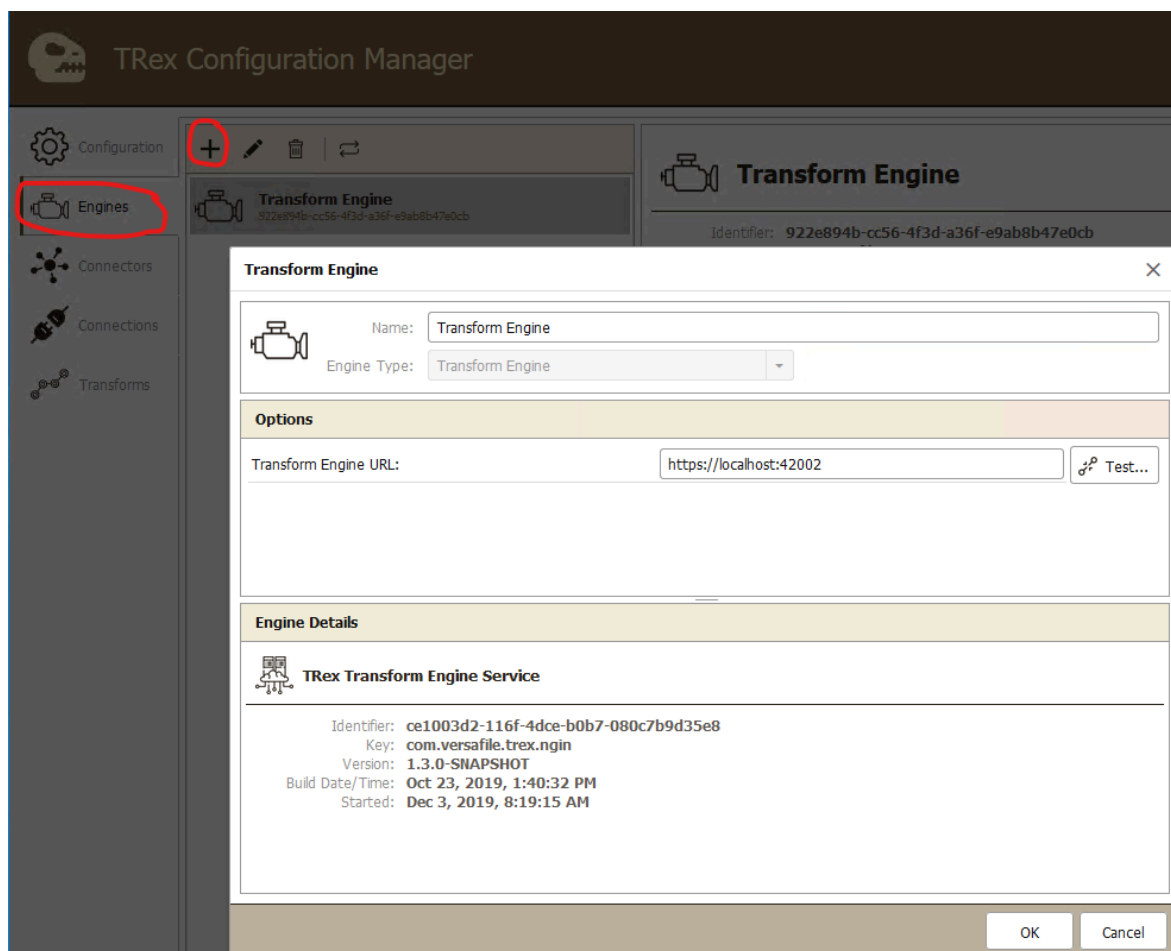
5.1.4.3.1. Engines

To add the main Transformation Engine to docuflow, select the 'Engines' tab on the right, then click on the '+' button

Fill in the URL using HTTPS protocol to the Transformation Engine service. By default this will be on the local machine on port 42002.

For example: `https://localhost:42002`

Click on the 'Test' button and after a successful connection, the Engine Details will be shown.



5.1.4.3.2. Connectors

Each connector is configured by linking

Select the 'Connectors' tab, then click the '+' button

Enter the URL to the connector endpoint using HTTPS and providing the port

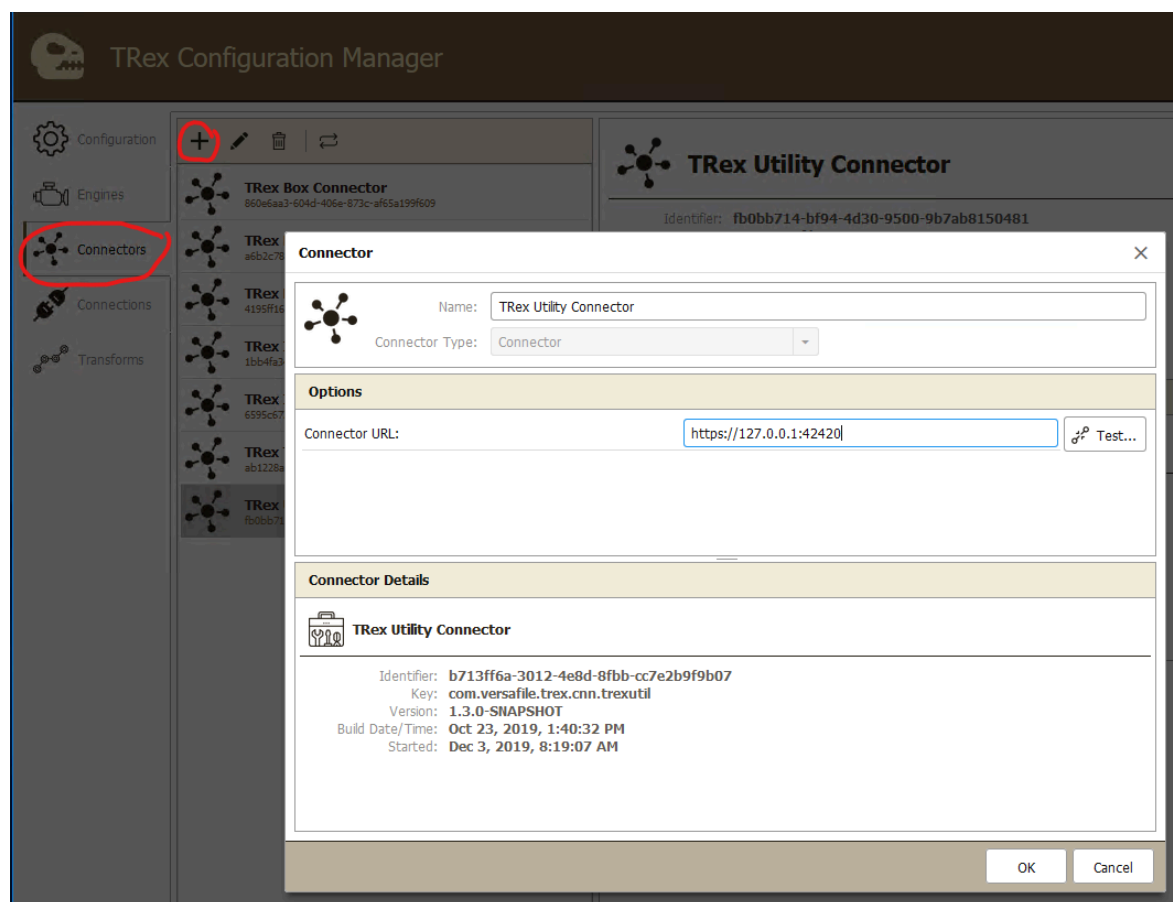
In default installations localhost or loopback addresses can be used. In advanced configurations where connectors are deployed on remote servers, the remote machine address/name should be used:

```
https://127.0.0.1:42420
```

```
https://localhost:42420
```

```
https://<remoteconnectorserver>:42420
```

Note: that 127.0.0.1 is a loopback ip4 address - the same as using localhost. This indicates the connector is running on the same machine as the TRex core services.



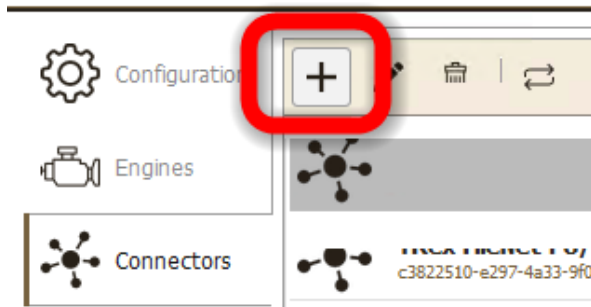
The default ports for the connectors assigned in the TRex configuration are listed in the table below:

Trex Utilities	42420
Filesystem	42421
IBM FileNet P8 Content Engine	42422
Apache Tika	42423
IBM Watson Cloud Services	42424
Elasticsearch	42425
IBM FileNet P8 Process Engine	42426
Box Cloud Content Management	42427
IBM Content Manager CM8	42429
IBM Forms Server	42430
Microsoft SharePoint	42431
Generic Web Services	42432

5.1.4.3.2.1. Box

To Create a new Box Connector in the docuflow Configuration UI:

1. Click on the 'Connectors' Tab
2. Click on the + icon
3. Enter the Connector URL (**<https://localhost:42427>**)



Adding a Connector

Connector Description

The Box connector provides connectivity to the Box Content Management, Workflow and Collaboration platform. See <http://www.box.com>

Box is a SaaS platform and all current releases of the **Enterprise** version are supported.

Methods Included:

File Operations

- Delete File
- Fetch File
- File Iterator
- Update File
- Upload File

Folder Operations

- Create Folder
- Delete Folder
- Folder Iterator
- Update Folder

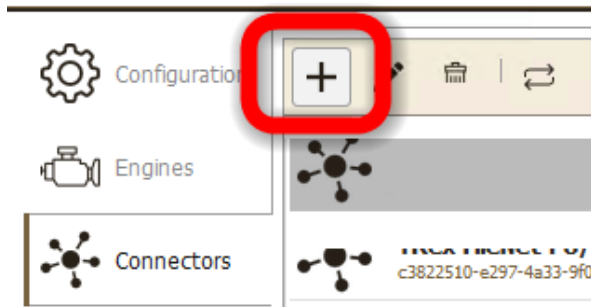
Search

- Metadata Query
- Search

5.1.4.3.2.2. IBM FileNet

To Create a new Box Connector in the docuflow Configuration UI:

1. Click on the 'Connectors' Tab
2. Click on the + icon
3. Enter the Connector URL (**https://localhost:42426**)



Adding a Connector

Connector Description

The Box connector provides connectivity to the IBM FileNet Content Platform Engine (CPE) for document storage, retrieval, and management.

IBM FileNet is an Enterprise Content Management (ECM) platform, and all currently supported versions of IBM FileNet P8 CPE are compatible with this connector. (??? for validation)

Methods Included:

File Operations

- Delete File
- Fetch File
- File Iterator
- Update File
- Upload File

Folder Operations

- Create Folder
- Delete Folder
- Folder Iterator
- Update Folder

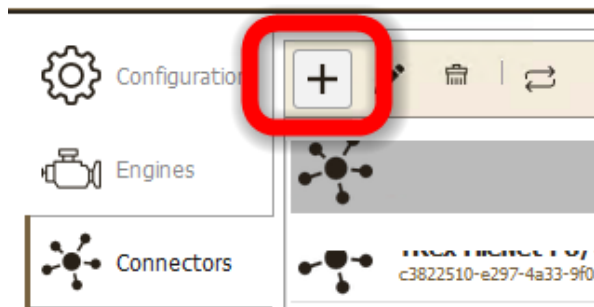
Search

- Metadata Query
- Search

5.1.4.3.2.3. Filesystem

To Create a new Filesystem Connector in the docuflow Configuration UI:

1. Click on the 'Connectors' Tab
2. Click on the + icon
3. Enter the Connector URL (**https://localhost:42421**)



Adding a Connector

Connector Description

The filesystem connector allows for connections to any file locations that are accessible from the server d4i is installed on.

The Filesystem connector takes an argument on its mount point to connect to. Only this location and folders/files below will be accessible.

This is configured in file `./config/application-cnn-fs.yaml`. By default, it will start looking for a folder called 'root' within the installation location of TRex. To change this, update the 'mount-point' variable.

```
plugin-args:
  mount-point: ./root
```

This can be changed to a local drive location such as C: or D: or D:\mystuff. Additionally, remote shares can be access using UNC paths. For example: \\servername\share

NOTE: If you are mounting a remote filesystem share, the Filesystem service should be configured to log in with an account that has access/priviledges to the remote share.

Methods Included:

CSV File Operations

- CSV Row Iterator
- CSV Write Row

File Conversion

- Convert to PDF

File Operations

- Copy File
- Delete File
- File Digest

- [File Iterator](#)
- [Move File](#)
- [Write Content To File](#)

Folder Operations

- [Delete Folder](#)
- [Folder Iterator](#)

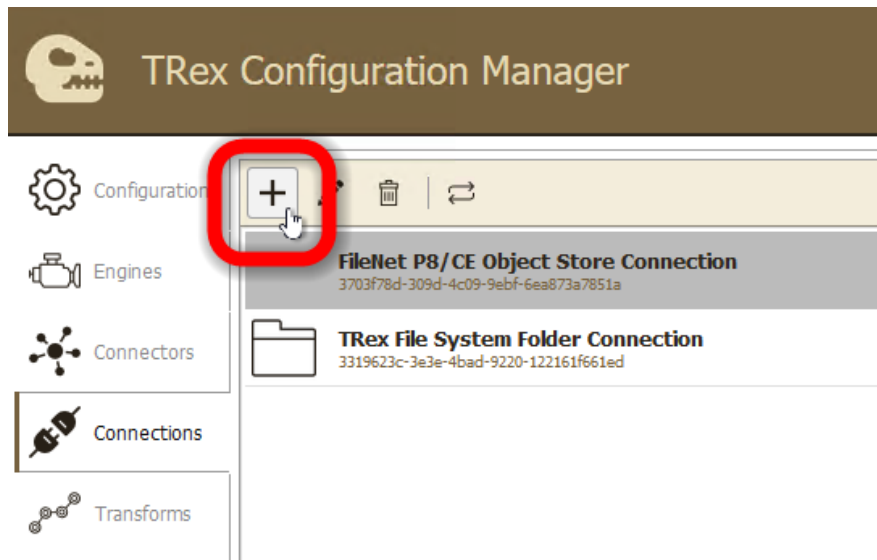
XML File Operations

- [XML Node Iterator](#)

5.1.4.3.3. Connections

Once Connectors have been created, the Connections can be created.

To create a new connection, select the '**Connections**' Tab and then the + (add...) button.



Each connection will require connection specific information and is detailed in the following sections.

5.1.4.3.3.1. Box

Minimum Requirements

Access to the following URLs is required from the middleware server:

Base url: <https://api.box.com/2.0/>

Base upload url: <https://upload.box.com/api/2.0/>

Token url: <https://api.box.com/oauth2/token>

Firewall Information:

- Source: the IP of the middleware server
- Port: https is 443
- Protocol: https

This provides the basic functionality that doucflow requires for its processing. It is strongly recommended however that you review the Box link below to ensure you have all the appropriate access for your particular environment and usage requirements above and beyond doucflow:

<https://support.box.com/hc/en-us/articles/360043696434-Configuring-A-Firewall-For-Box-Applications>

Box also provides a connectivity test page as well as documented firewall configuration if necessary:

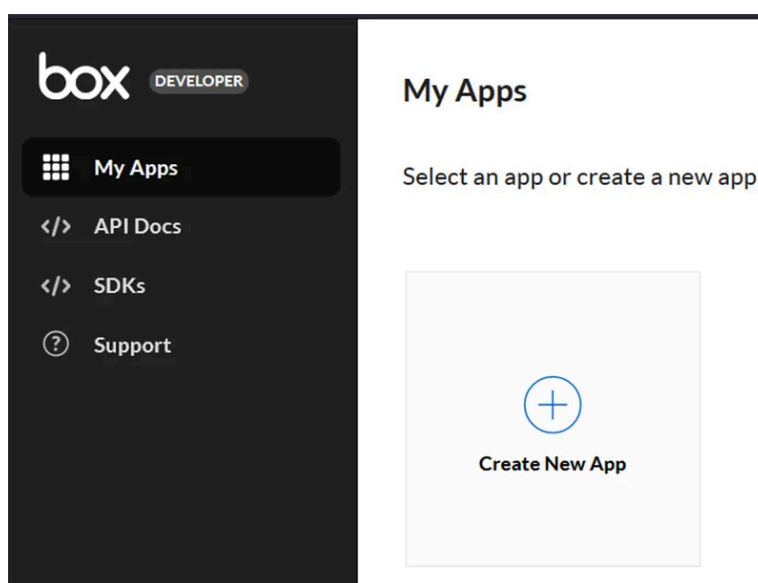
<https://support.box.com/hc/en-us/articles/360043694474-Box-Connectivity-Tests>

Configuring a Box Connection

Go to your Box developer console:

- <https://app.box.com/developers/console>

Under MyApps, click on the 'Create New App' button



Under 'Lets Get Started' - Select **Custom App**

Under 'Authentication Method' - Select **Server Authentication (with JWT)** and click Next

Custom App ✕

Select an authentication method and enter a name for your app. You can change the name later in General Settings.

Authentication Method

☒ **Server Authentication (with JWT)**

- Best for building integrations or apps with external collaborators - [Learn More](#)
- The app authenticates using a JSON Web Token. Requires a public/private key-pair for added security.

☐ **User Authentication (OAuth 2.0)**

- Best for building mobile or web apps - [Learn More](#)
- Users must authenticate with their Box login.

☐ **Server Authentication (Client Credentials Grant)**

- Best for scripting and back office integrations - [Learn More](#)
- The app authenticates using Client ID and Client Secret.

App Name

Enter an app name

By clicking 'Create App', you agree to the terms of the [Box Developer Agreement](#) and the [Box Privacy Policy](#).

Cancel Create App

Name your application e.g. '**d4iConnector**' and click 'Create App'

Click 'View Your App' and navigate to the '**Configuration**' tab

Under **Application Scopes** select the following options:

- Read all files and folders stored in Box
- Write all files and folders stored in Box
- Manage users (allows App users to be created)
 - Note: if you want to script out and create app users without docuflow, this setting is not require
 - Also: This setting can be used for the application to create the app users, and once the process is complete, it can be removed

Under **Advanced Features** select the following options

- Make API calls using the as-user header (allows scaling across app users)
- Generate user access tokens

Under **Add and Manage Public Keys** select 'Generate a Public/Private Keypair' and Save the json file somewhere secure. This JSON will be required for the Box Connection configuration. ***This private key cannot be regenerated and must be stored safely and securely.***

Under **App Settings** select the 'Download JSON' - this JSON file is required for the Box connector

After completing these steps go to the 'App Authorization' and click 'Submit for Authorization'

Return to the Admin console, go to 'Apps' → Custom Apps

- Click 'Authorize New App' and then enter the Client ID from the previous step
- Click the Authorize button

Creating a d4i Box Connection

There are two types of connections supported for connecting to Box: Enterprise and AppUser. When creating a new Box connection this can be selected in the 'connection type' dropdown.

Box Enterprise Connection

The Enterprise connection uses the default service account for connecting to Box.

The following information is required when creating a new Enterprise connection:

- **Enterprise ID:** This can be found in the Box Admin under the **Account and Billing** section
- **Client ID:** This can be found in the **OAuth 2.0 Credentials** section of the Box Enterprise Application configured earlier
- **Client Secret:** This can be found in the **OAuth 2.0 Credentials** section of the Box Enterprise Application configured earlier
- **Public Key ID:** This can be found in the **Add and Manage Public Keys** section of the Box Enterprise Application configured earlier
- **Private Key:** This is the Private Key that corresponds to the Public key that was registered with the Box Enterprise Application. Must be entered properly formatted with multiple lines (not only one long line).
- **Passphrase:** This is the passphrase for the Key registered with the Box Enterprise Application.

Box AppUser Connection

The AppUser connection uses a defined set of AppUsers associated with the Box Enterprise Application to connect to Box. The d4i Box connector will round robin between the provided App Users configured in the connector.

The following information is required when creating a new AppUser connection:

- **AppUser List:** A list of Box App users that have been previously created under the Box Enterprise Application
- **Enterprise ID:** This can be found in the Box Admin under the **Account and Billing** section
- **Client ID:** This can be found in the **OAuth 2.0 Credentials** section of the Box Enterprise Application configured earlier
- **Client Secret:** This can be found in the **OAuth 2.0 Credentials** section of the Box Enterprise Application configured earlier
- **Public Key ID:** This can be found in the **Add and Manage Public Keys** section of the Box Enterprise Application configured earlier
- **Private Key:** This is the Private Key that corresponds to the Public key that was registered with the Box Enterprise Application. Must be entered properly formatted with multiple lines (not only one long line).
- **Passphrase:** This is the passphrase for the Key registered with the Box Enterprise Application.

Once a connection is created, a connection test can be run to check if the configuration is valid.



Name:

Connection Type:

Options

AppUser List: ...  Test...

Client ID:

Client Secret:

Public Key ID:

Private Key:

-----BEGIN ENCRYPTED PRIVATE KEY-----
MIIFDjBABgkqhkiG9w0BBQ0wMzAbBgkqhkiG9w0BBQwwDgQIuWU3FegOf
MBQGCCqGS1b3DQMHBAlRNMHCvXxkeQSCBMhYJmfivCGwsTpAJcbEhXCHri
jm51ICgffZpG2rzFNQpwA0uCih0Z9mqh/Tb3vFf/FelktFAULkWM0yL5xWyD
P5kZ2nwsT+JV4geMd+B8CAkz9LLVTwrKBYOHlKtoIX037EQjxOWq3pw59!
< >

Passphrase:

5.1.4.3.3.2. IBM FileNet

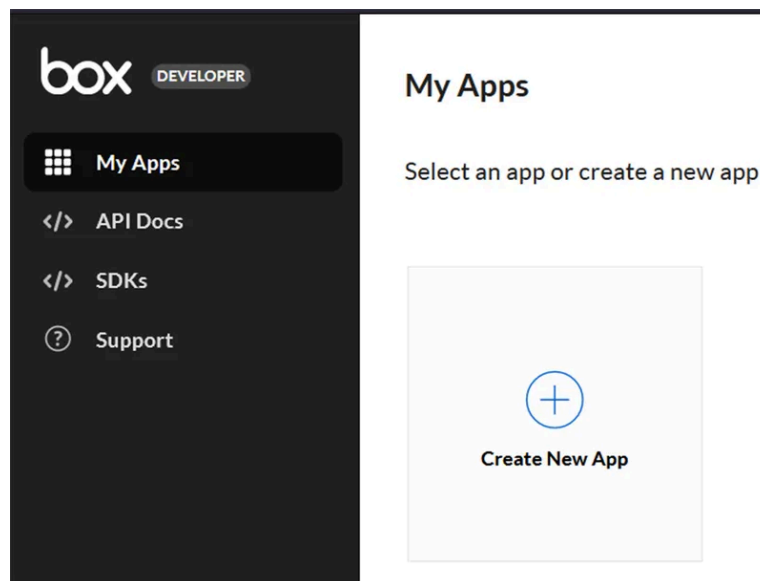
Configuring an IBM FileNet Connection

(TO BE UPDATED FOR FILENET)

Go to your Box developer console:

- <https://app.box.com/developers/console>

Under MyApps, click on the 'Create New App' button



Under 'Lets Get Started' - Select **Custom App**

Under 'Authentication Method' - Select **Server Authentication (with JWT)** and click Next

Custom App ✕

Select an authentication method and enter a name for your app. You can change the name later in General Settings.

Authentication Method

☒ **Server Authentication (with JWT)**

- Best for building integrations or apps with external collaborators - [Learn More](#)
- The app authenticates using a JSON Web Token. Requires a public/private key-pair for added security.

☐ **User Authentication (OAuth 2.0)**

- Best for building mobile or web apps - [Learn More](#)
- Users must authenticate with their Box login.

☐ **Server Authentication (Client Credentials Grant)**

- Best for scripting and back office integrations - [Learn More](#)
- The app authenticates using Client ID and Client Secret.

App Name

Enter an app name

By clicking 'Create App', you agree to the terms of the [Box Developer Agreement](#) and the [Box Privacy Policy](#).

Cancel Create App

Name your application e.g. '**d4iConnector**' and click 'Create App'

Click 'View Your App' and navigate to the '**Configuration**' tab

Under **Application Scopes** select the following options:

- Read all files and folders stored in Box
- Write all files and folders stored in Box
- Manage users (allows App users to be created)
 - Note: if you want to script out and create app users without docuflow, this setting is not require
 - Also: This setting can be used for the application to create the app users, and once the process is complete, it can be removed

Under **Advanced Features** select the following options

- Make API calls using the as-user header (allows scaling across app users)
- Generate user access tokens

Under **Add and Manage Public Keys** select 'Generate a Public/Private Keypair' and Save the json file somewhere secure. This JSON will be required for the Box Connection configuration. ***This private key cannot be regenerated and must be stored safely and securely.***

Under **App Settings** select the 'Download JSON' - this JSON file is required for the Box connector

After completing these steps go to the 'App Authorization' and click 'Submit for Authorization'

Return to the Admin console, go to 'Apps' → Custom Apps

- Click 'Authorize New App' and then enter the Client ID from the previous step
- Click the Authorize button

Creating a d4i Box Connection

There are two types of connections supported for connecting to Box: Enterprise and AppUser. When creating a new Box connection this can be selected in the 'connection type' dropdown.

Box Enterprise Connection

The Enterprise connection uses the default service account for connecting to Box.

The following information is required when creating a new Enterprise connection:

- **Enterprise ID:** This can be found in the Box Admin under the **Account and Billing** section
- **Client ID:** This can be found in the **OAuth 2.0 Credentials** section of the Box Enterprise Application configured earlier
- **Client Secret:** This can be found in the **OAuth 2.0 Credentials** section of the Box Enterprise Application configured earlier
- **Public Key ID:** This can be found in the **Add and Manage Public Keys** section of the Box Enterprise Application configured earlier
- **Private Key:** This is the Private Key that corresponds to the Public key that was registered with the Box Enterprise Application. Must be entered properly formatted with multiple lines (not only one long line).
- **Passphrase:** This is the passphrase for the Key registered with the Box Enterprise Application.

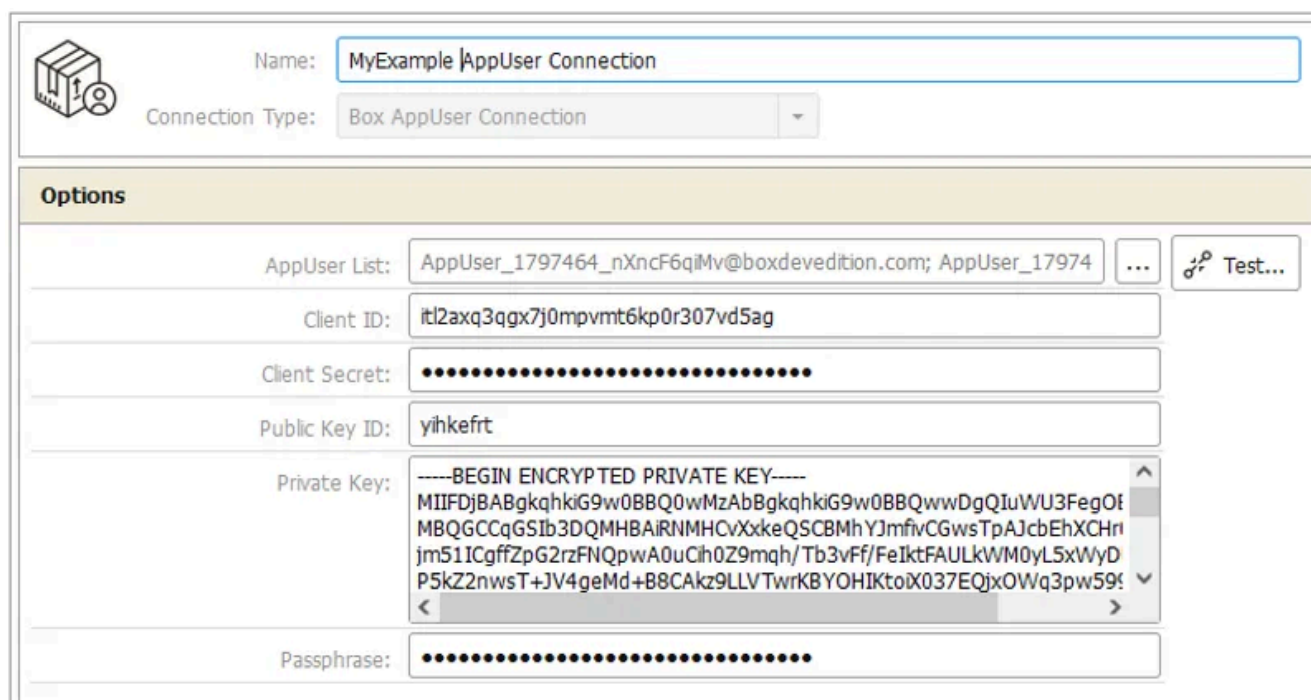
Box AppUser Connection

The AppUser connection uses a defined set of AppUsers associated with the Box Enterprise Application to connect to Box. The d4i Box connector will round robin between the provided App Users configured in the connector.

The following information is required when creating a new AppUser connection:

- **AppUser List:** A list of Box App users that have been previously created under the Box Enterprise Application
- **Enterprise ID:** This can be found in the Box Admin under the **Account and Billing** section
- **Client ID:** This can be found in the **OAuth 2.0 Credentials** section of the Box Enterprise Application configured earlier
- **Client Secret:** This can be found in the **OAuth 2.0 Credentials** section of the Box Enterprise Application configured earlier
- **Public Key ID:** This can be found in the **Add and Manage Public Keys** section of the Box Enterprise Application configured earlier
- **Private Key:** This is the Private Key that corresponds to the Public key that was registered with the Box Enterprise Application. Must be entered properly formatted with multiple lines (not only one long line).
- **Passphrase:** This is the passphrase for the Key registered with the Box Enterprise Application.

Once a connection is created, a connection test can be run to check if the configuration is valid.



The screenshot shows a web form for configuring a 'Box AppUser Connection'. At the top, there is a 'Name' field containing 'MyExample AppUser Connection' and a 'Connection Type' dropdown menu set to 'Box AppUser Connection'. Below this is an 'Options' section with several input fields: 'AppUser List' (containing 'AppUser_1797464_nXncF6qIMv@boxdevedition.com; AppUser_17974'), 'Client ID' (containing 'itl2axq3qgx7j0mpvmt6kp0r307vd5ag'), 'Client Secret' (masked with dots), 'Public Key ID' (containing 'yihkefrt'), 'Private Key' (containing a multi-line encrypted key starting with '-----BEGIN ENCRYPTED PRIVATE KEY-----'), and 'Passphrase' (masked with dots). To the right of the 'AppUser List' field is a 'Test...' button with a play icon.

5.1.4.3.3. Filesystem

The filesystem connection allows for the setting of a subfolder root that is relative to where the connector is running.

For example, if the connector service was connected to the root of D:\ then the value entered would be a subdirectory of D:\. Use the default ./ to allow access to all subdirectory locations.



Name:

Connection Type: TRex File System Folder Connection ▼

Options

Root Path:

 Test...

Output Entity

 **Folder Entity**

Attributes

Breadcrumb:	WTI5dExuWmxjbk5oWm1sc1pTNTBjbVY0TG10dWJpNW1jdz09fl ^
Folder Name:	root
Folder Path:	.\

5.2. SuccessFactors for Box

5.2.1. SuccessFactors Technical Overview



5.2.1.1. Introduction

docuflow for SuccessFactors and Box integrates documents between SuccessFactors and the Box content platform.

This integration includes document storage, linking, and metadata transfer from SuccessFactors to Box. Documents can be captured from SuccessFactors and stored within Box and documents added to FileNet can be linked back to SuccessFactors.

To enable users to access the documents in Box, two HTML5 applications give them access:

- My Documents
- Document Search

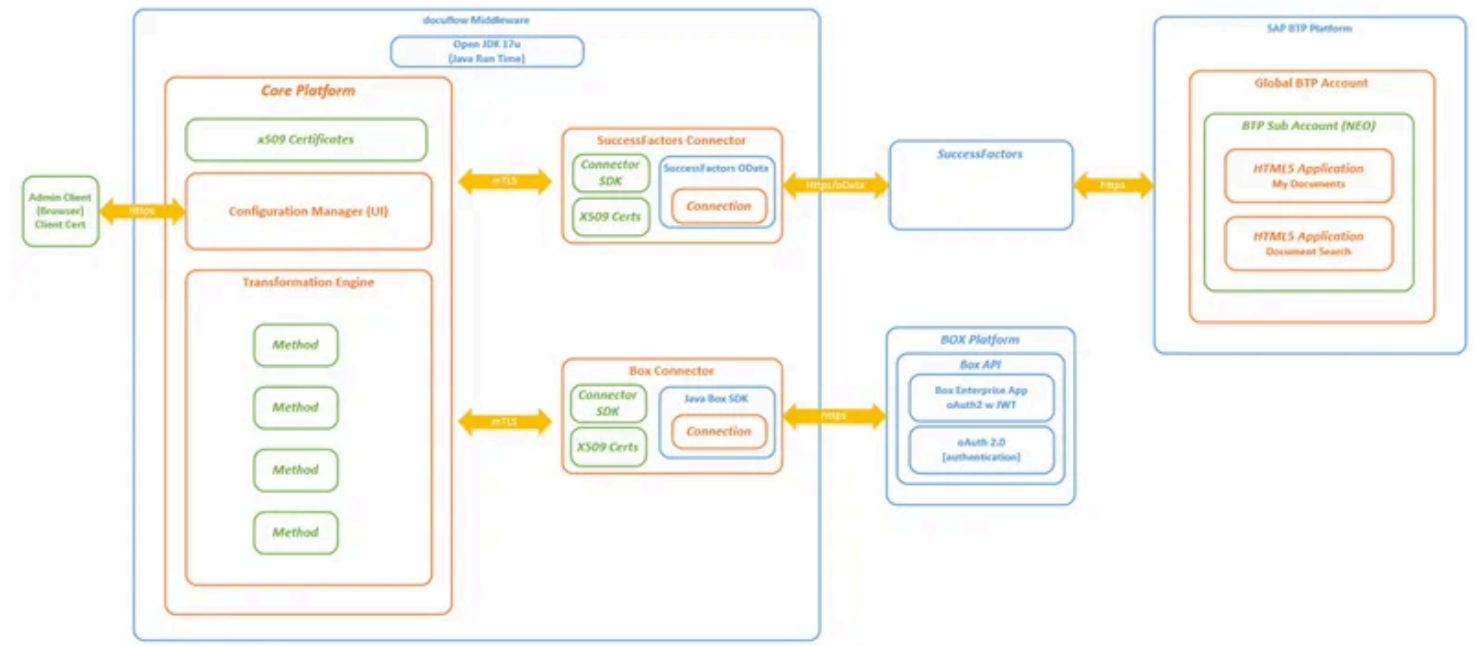
My Documents will show a user all of their employee documents that have been linked from Box.

Document Search will allow HR Managers to search for documents belonging to employees that report to them.

The process to capture and link documents is done through the docuflow middleware while the My Documents and Document Search is done through HTML5 applications deployed on the SAP BTP Platform.

5.2.1.2. Architecture Overview

The below diagram outlines the basic components of the solution



The docuflow Middleware is responsible for:

- Discovering new Employees and creating a Employee Folder in Box and linking this folder back to their Employee Profile
- Discovering new Employee Documents in SuccessFactors, adding them to Box and linking them back to SuccessFactors using LinkCards
- Discovering new Employee Documents in Box and linking them to SuccessFactors using LinkCards

The SAP BTP Platform is responsible for:

- A user interface (MyDocuments) for Employees to see the LinkCards that have been attached to their Employee Profile
- A user interface (DocumentSearch) for HR Managers to see the LinkCards for employees they are related to

5.2.1.3. Minimum Requirements

The docuflow middleware component (docuflow Enterprise Connector) can be deployed on-premise, in a private cloud or as a fully managed service from VersaFile.

When deploying on premise or in a private cloud environment, physical or virtual hardware is supported. The following server resources are recommended:

Operating Systems	CPU	Disk Space	RAM	Java Runtime Environment
Windows Server 2012, Windows Server 2012 R2, Windows Server 2016, Windows Server 2019	Minimum 4 cores or 4 physical CPUs with at least 2.0Ghz	40 GB	16 GB	OpenJDK 17 LTS or later

JCO requirements: 3.0.14 and above

(3.1.x and above requires Visual C++ Redistributable Packages for Visual Studio 2013)

<https://www.microsoft.com/en-ca/download/details.aspx?id=40784>

See <https://me.sap.com/notes/2786882>

URL and Firewall Configuration for Box API Access

Access to the following URLs is required from the middleware server:

Base url: <https://api.box.com/2.0/>

Base upload url: <https://upload.box.com/api/2.0/>

Token url: <https://api.box.com/oauth2/token>

Firewall Information:

- Source: the IP of the middleware server
- Port: https is 443
- Protocol: https

This provides the basic functionality that docuflow requires for its processing. It is strongly recommended however that you review the Box link below to ensure you have all the appropriate access for your particular environment and usage requirements above and beyond docuflow:

<https://support.box.com/hc/en-us/articles/360043696434-Configuring-A-Firewall-For-Box-Applications>

Box also provides a connectivity test page as well as documented firewall configuration if necessary:

<https://support.box.com/hc/en-us/articles/360043694474-Box-Connectivity-Tests>

5.2.2. BTP Configuration

This section describes the typical configuration required to integrate the three HTML5 applications into SuccessFactors.

- My Documents
- Document Search
- Document Search (HRBP)

Required Accounts:

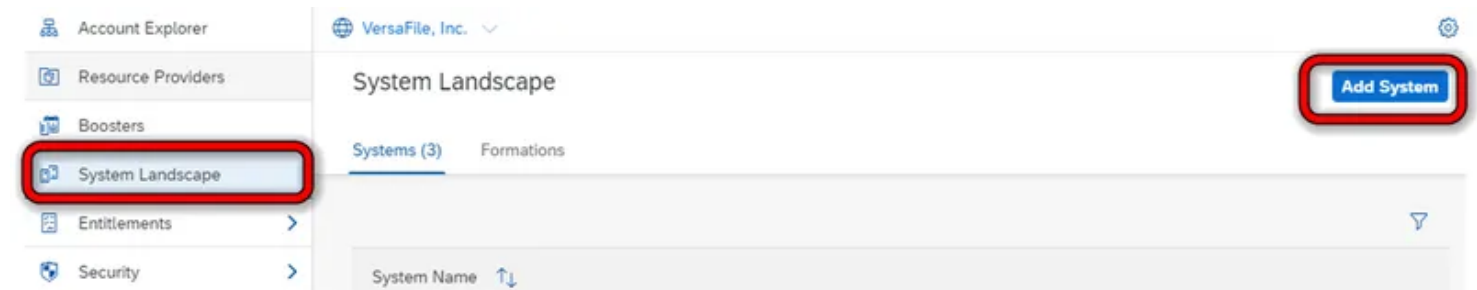
- Administrative BTP Access
- Administrative SuccessFactors Access

5.2.2.1. Linking SF to BTP Landscape

The following steps register the SuccessFactors instance to the BTP system Landscape. This is the first task in configuring the integration.

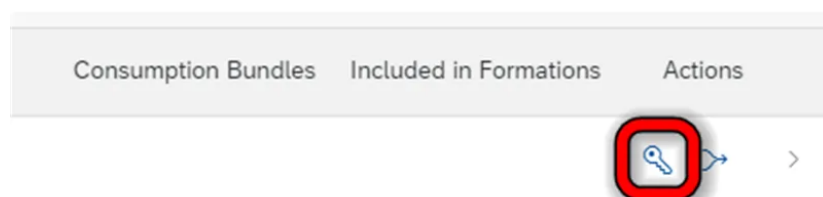
At the BTP Global Account, navigate to **System Landscape**.

Click the **'Add System'** button

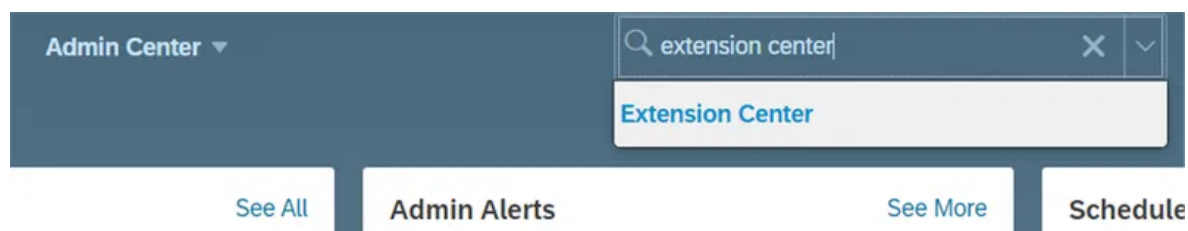


Supply a descriptive System Name and choose **'SAP SuccessFactors'** from the Type dropdown

Once the system has been added, click on the 'Get Token' icon. Copy this Token Value. This Token will be used within SuccessFactors to register the SuccessFactors system with the BTP Landscape.

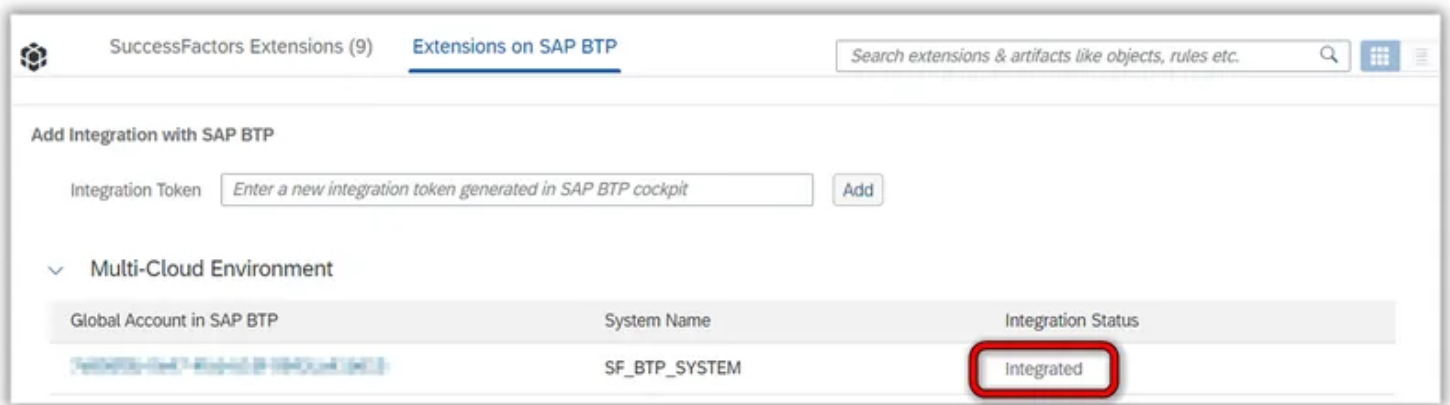


Next Log into **SuccessFactors** with an Administrative Account and Navigate to **'Extensions Center'**



Within the Extension Center navigate to the **'Extensions on SAP BTP'** tab. In the section **'Add Integration with SAP BTP'** paste the Token and click the **'Add'** button. This will connect the SuccessFactors instance with the BTP landscape.

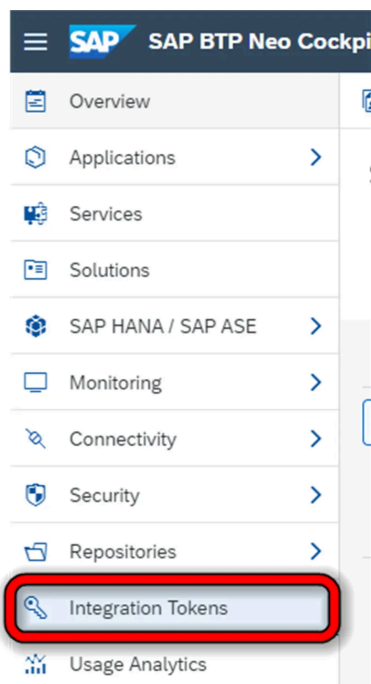
A message indicating the Systems are integrating will be shown. After a short period the status will show **'Integrated'**



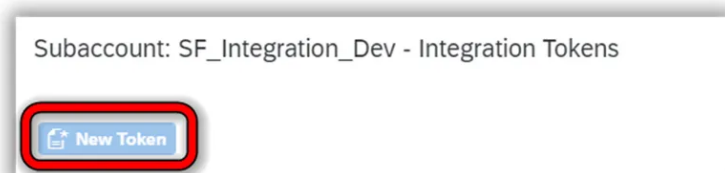
Next, link the BTP Sub Account to the Extensions on SAP BTP

- **NOTE:** The BTP Sub Account MUST be a NEO account (SAP Region)
- **NOTE:** See '**BTP Sub Account Creation**' sub-topic if you have not yet created a Sub Account

Open your BTP Sub Account and Navigate to '**Integration Tokens**'



Click on the '**New Token**' option to generate a new integration token for this BTP Sub Account. This token will then be used in SuccessFactors to integrate to the BTP Sub Account



Navigate back to '**Extentions Center**' in SuccessFactors

Once again, use the 'Add' Integration Token feature



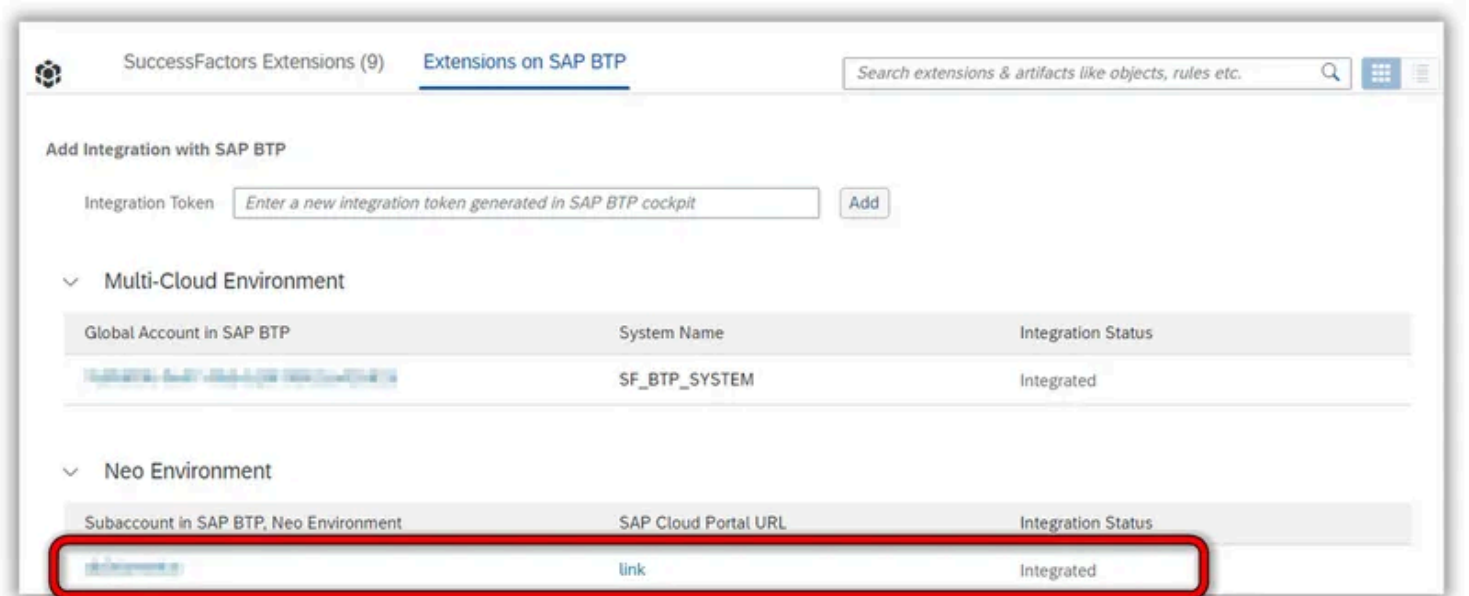
SuccessFactors Extensions (9) Extensions on SAP BTP

Search extensions & artifacts like objects, rules etc.

Add Integration with SAP BTP

Integration Token Add

Once this token is added, it will show up in the **'Neo Environment'** Section



SuccessFactors Extensions (9) Extensions on SAP BTP

Search extensions & artifacts like objects, rules etc.

Add Integration with SAP BTP

Integration Token Add

Multi-Cloud Environment

Global Account in SAP BTP	System Name	Integration Status
Global Account in SAP BTP	SF_BTP_SYSTEM	Integrated

Neo Environment

Subaccount in SAP BTP, Neo Environment	SAP Cloud Portal URL	Integration Status
Subaccount in SAP BTP, Neo Environment	link	Integrated

A Trust has been created to BTP. To see this, Navigate to **BTP Cockpit → Security → Trust**

 **SAP BTP Neo Cockpit**

Overview

Applications

Services

Solutions

SAP HANA / SAP ASE

Monitoring

Connectivity

Security

Trust

Authorizations

OAuth

Repositories

Integration Tokens

Usage Analytics

Entitlements

Members

VersaFile, Inc. / SF_Integration_Dev

Subaccount: SF_Integration_Dev - Trust

Trust Management

Local Service Provider | Application Identity Provider | Custom Application Domains Settings | Basic Authentication

 **Manage Local Provider Settings**
for yukrury3mx

Configuration Type: Custom

Local Provider Name: *

Signing Key: *

Signing Certificate: *

Certificate Valid From: Wed Nov 09 22:24:42 UTC 2022

Certificate Valid To: Tue Nov 09 22:24:42 UTC 2032

Principal Propagation: Enabled

Force Authentication: Disabled

Edit

Get Metadata

368 / 534

docuflow Guides

Modified: 07/02/2026

5.2.2.1.1. BTP Sub Account Creation

This step only needs to be done if there is not already a sub account created for the BTP environment.

At the Global Account Level, navigate to **'Account Explorer'** and click the **'Create'** button and select **'Sub Account'**

NOTE: Select one of the SAP (NEO) Regions to run the integration

Create Subaccount

Display Name*

Subdomain ⓘ

Description

Region*

Parent*

Region options:

- Singapore cf-ap21
- US East (VA) cf-us21
- US West (WA) cf-us20
- SAP**
- Australia (Sydney) neo-ap1
- Brazil (São Paulo) neo-br1
- Canada (Toronto) neo-ca1
- China (Shanghai) neo-cn1
- Europe (Amsterdam) neo-eu3
- Europe (Frankfurt) neo-eu2
- Japan (Tokyo) neo-jp1
- KSA (Riyadh) neo-sa1
- UAE (Dubai) neo-ae1
- US West (Colorado Springs) neo-us4

Account structure.

not global account or to an parent.

Create **Cancel**

5.2.2.2. HTML5 Application Registrations

This section will demonstrate how install the HTML5 applications into the BTP Sub Account that has been integrated into the SuccessFactors system in the previous section.

SAP Web IDE Full-Stack Service

If not already enabled, SAP Web IDE Full-Stack Service must be enabled in the sub account.

To enable this, in the sub account navigate to 'Services' and then select the tile/option 'SAP Web IDE Full-Stack' and then click '**Enable**'.

Security Permissions

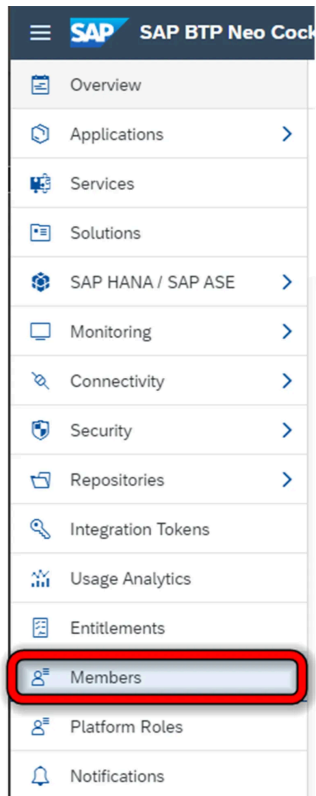
NOTE: This section will require the BTP Command Line Interface (CLI). If you do not have this installed please see the sub section '**BTP Command Line Interface**'

First, Permissions must be given to allow configuration of the BTP Sub Account.and Web Applications.

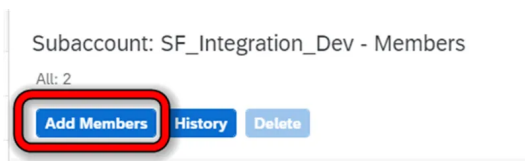
To Enable security for the Web Applications, navigate to **Security→Authorizations**

- Enter the S User ID and click 'Show Assignments'
- Click 'Assign' and then select
 - Subaccount 'SAP Web IDE xxx'
 - Application: di
 - Role: DiAdministrator
- Click 'Assign' and then select:
 - Subaccount 'SAP Web IDE xxx'
 - Application: featurediscovery
 - Role: DevXAdministrator

Next, Navigate to the '**Members**' section of the BTP Sub Account



In the Members section, click **'Add Members'**

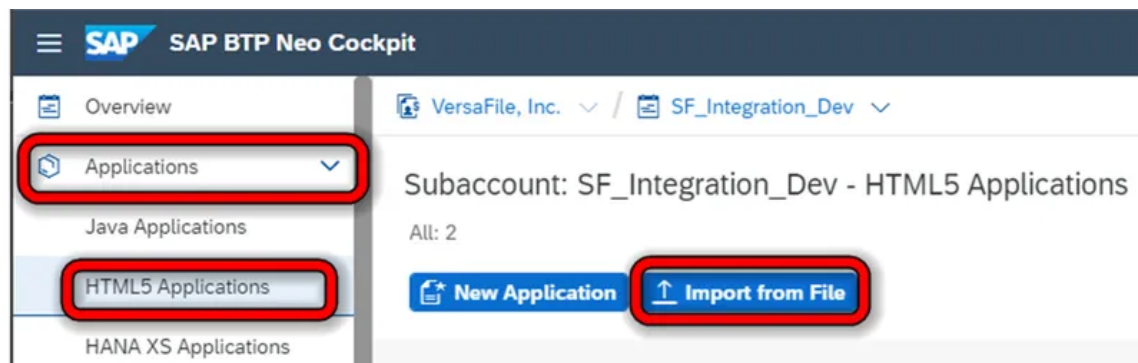


Enter the **User id** and select the permissions you want this user account to have the BTP Sub Account.

 A screenshot of the 'Add Members' dialog box. It contains a text input field for 'User IDs' with the placeholder text 'Sxxxxxxxxx'. To the right, under 'Assign roles:', there are five checkboxes, all of which are checked: 'Administrator (predefined role)', 'Developer (predefined role)', 'Support User (predefined role)', 'Application User Admin (predefined role)', and 'Cloud Connector Admin (predefined role)'. At the bottom, there is a text input field for 'Comment (optional, visible to all members):'. The dialog has 'Add Members' and 'Cancel' buttons at the bottom right.

Next, upload the HTML5 Applications to the BTP Sub Account. These applications are supplied in a .zip format.

Navigate to **Applications→HTML5 Applications** in the BTP Sub Account and click **'Import from File'**



In the dialog, select the '*sfmydocuments.zip*', enter the version number 1 and click 'Import'

Follow the same procedure as above for '*sfdocumentsearch.zip*'.

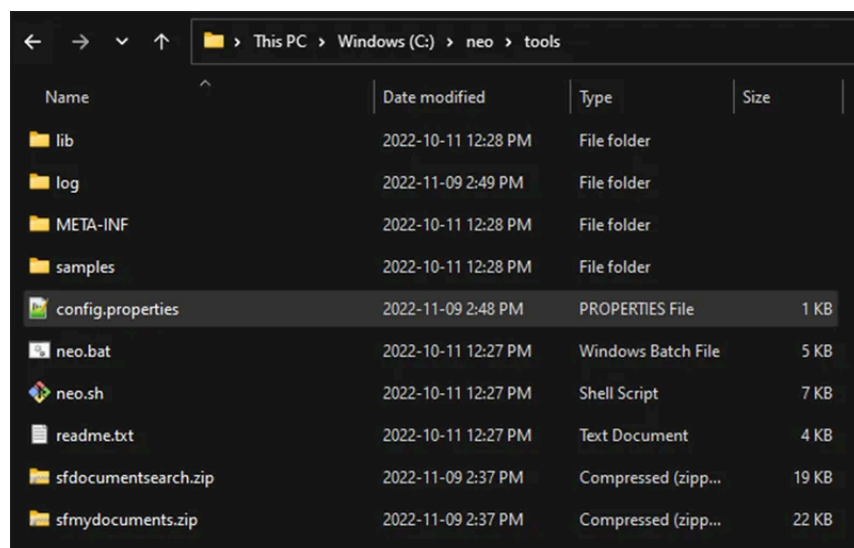
Once completed the applications should be listed in the HTML5 Applications screen.

Enable Application Access

NOTE: This next step required the BTP Command Line Interface (CLI). If you do not have this toolset installed, see the sub topic '**BTP Command Line Interface (CLI)**'

NOTE: The following examples are shown on windows but can be adapted for any other command line/shell

Open a Command Prompt/Shell and navigate to the '**Tools**' directory of the BTP Tools installation



Setup a properties file (e.g. **config.properties**) with the information that will be required for the connection. This will make the process easier.

NOTE: For a list of BTP Host values/locations: <https://help.sap.com/docs/btp/sap-btp-neo-environment/regions-and-hosts-available-for-neo-environment>

```
#####
# General settings - relevant for all commands #
#####

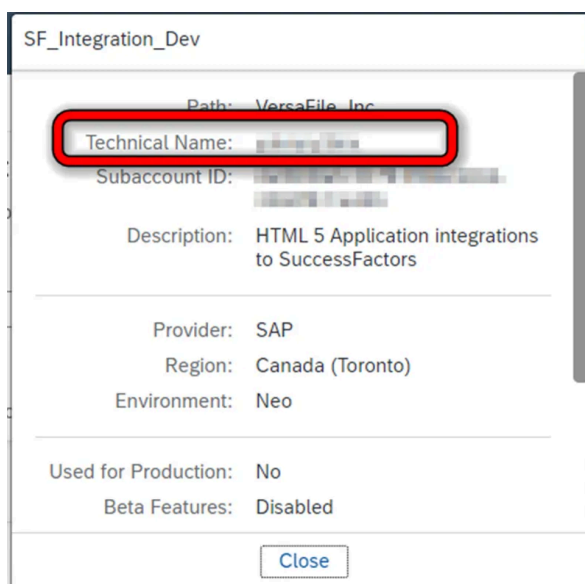
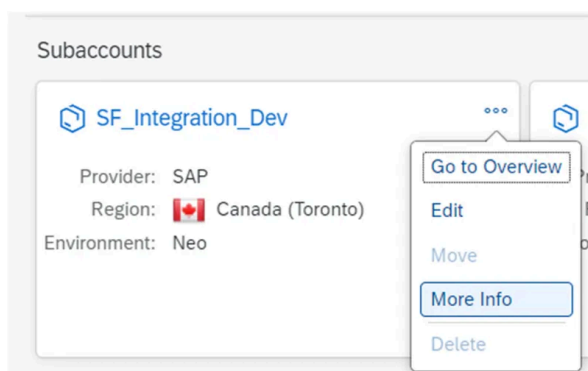
# Your account name
account=sfdocumentsearch

# Application name
application=sfdocumentsearch
application-type=html5

# User for login to hana.ondemand.com.
user=sapuser@hana.ondemand.com

# URL of the landscape admin server. Optional. Defaults to
hana.ondemand.com.
host=hana.ondemand.com
```

account = the ID of your BTP Sub Account



application = name of the application be registered. This process will be done twice, once for sfdocumentsearch and once for sfmydocuments. Note: use one of these values, run the command below, then alter this value and run the command again.

user = your SAP User Account. If you have a SAP Universal ID you will want to specify your S-ID, otherwise you can use the email associated with your S-ID

With the Command Prompt BTP CLI tools, execute the following command (use your properties file name)

neo.bat hcmcloud-enable-application-access config.properties

NOTE: if you are not using the properites file method, the full command would be:

```
neo hcmcloud-enable-application-access --application <my_application> --application-type java --account
<my_extension_subaccount> --user <user id> --host us1.hana.ondemand.com
```

Next, modify the config.properties file so that **application=sfmydocuments** to allow registration of the **sfmydocuments** application

```
#####
# General settings - relevant for all commands #
#####

# Your account name
account=sfmydocuments

# Application name
application=sfmydocuments
application-type=html5

# User for login to hana.ondemand.com.
user=sfmydocuments@hana.ondemand.com

# URL of the landscape admin server. Optional. Defaults to
hana.ondemand.com.
host=hana.ondemand.com
```

With the Command Prompt BTP CLI tools, execute the following command (use your properties file name)

neo.bat hcmcloud-enable-application-access config.properties

After registration example:

```

host      : https://ca1.hana.ondemand.com
account   : y[REDACTED]
application : sfmydocuments
application-type: html5
SDK version : 4.36.3
user      : [REDACTED]

Password for your user:
Enabling application access.....

Application access was enabled for application 'sfmydocuments' from subaccount 'y[REDACTED]':
Application Main URL      : https://sfmydocuments-[REDACTED].dispatcher.ca1.hana.ondemand.com
Service Provider Audience URL : https://ca1.hana.ondemand.com/[REDACTED]
Service Provider Logout URL  : https://authn.ca1.hana.ondemand.com/saml2/sp/slo/[REDACTED]

C:\Users\kyleh\Documents\SAP BTP\SAP NEO 2\tools>neo hcmcloud-enable-application-access vf2.properties

Neo Console Client

Executing [hcmcloud-enable-application-access] command with arguments:
host      : https://ca1.hana.ondemand.com
account   : [REDACTED]
application : sfdocumentsearch
application-type: html5
SDK version : 4.36.3
user      : S0024416632

Password for your user:
Enabling application access.....

Application access was enabled for application 'sfdocumentsearch' from subaccount '[REDACTED]':
Application Main URL      : https://sfdocumentsearch-[REDACTED].dispatcher.ca1.hana.ondemand.com
Service Provider Audience URL : https://ca1.hana.ondemand.com/[REDACTED]
Service Provider Logout URL  : https://authn.ca1.hana.ondemand.com/saml2/sp/slo/[REDACTED]

```

Register oData Connection

Finally, using the CLI register an oData connection (only one required for both applications)

Edit the **config.properties** file and comment out the application and:

Edit out the application application type properties:

```
#application=sfdocumentsearch
```

```
#application-type=html5
```

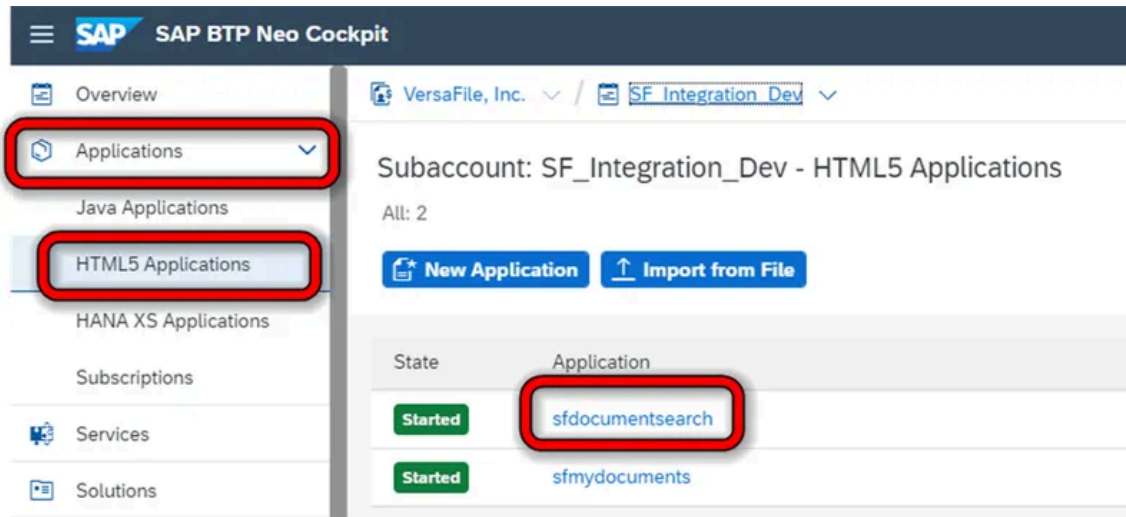
Add a new value for the name of the connection:

```
name=sfapps_odata
```

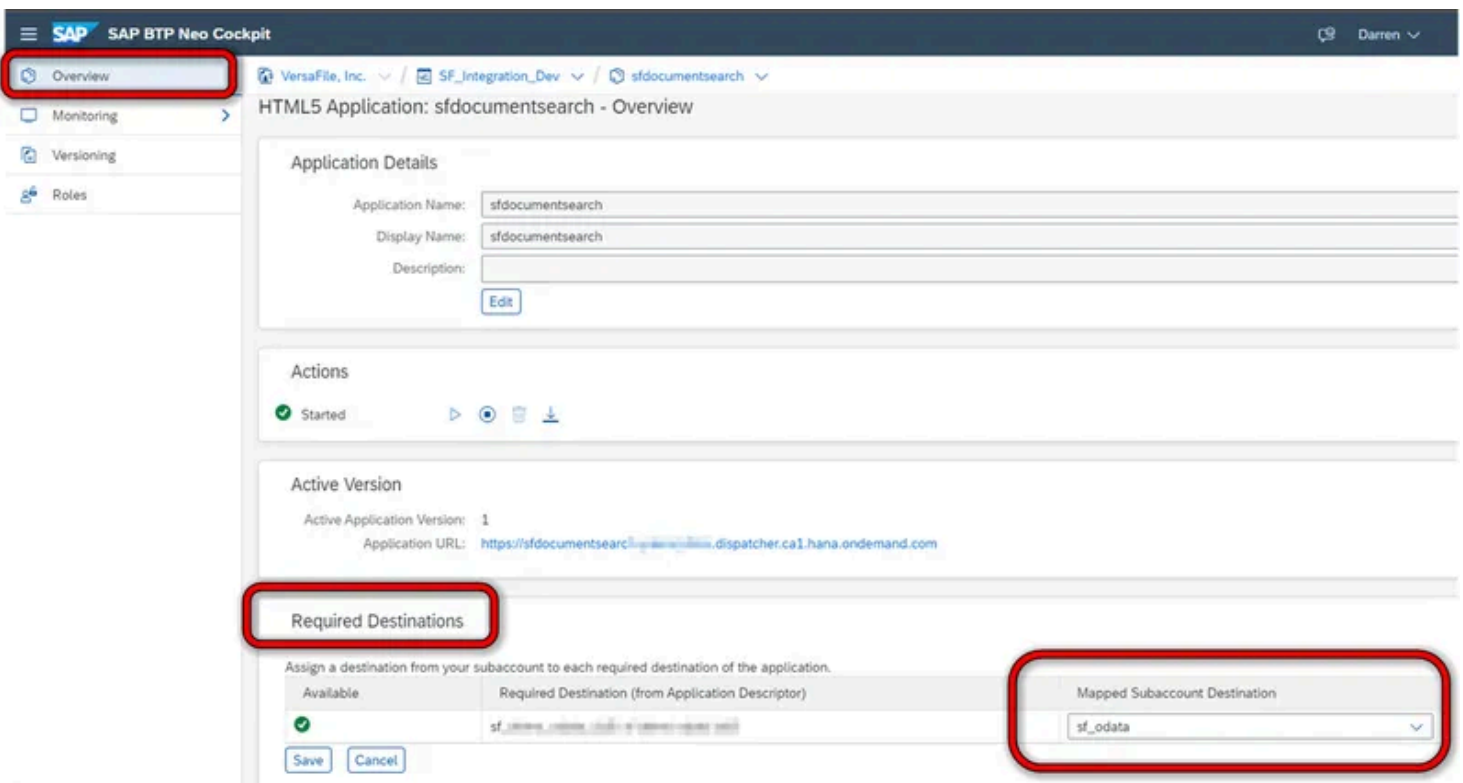
Neo.bat *hcmcloud-create-connection config.properties*

Set oData Connection in HTML5 Applications

For the next step, return to the **BTP Cockpit** and navigate to the **BTP Sub Account → Applications → HTML5 Applications** and click the sfdocumentsearch application



Under 'Required Destinations' set the 'Mapped Subaccount Destination'



Next, Click the 'Versioning' section, under 'History' section select the 'Versions' and set the 'Mapped Subaccount Destination'

SAP BTP Neo Cockpit

Overview | Monitoring | **Versioning** | Roles

VersaFile, Inc. / SF_Integration_Dev / sfdocumentsearch

HTML5 Application: sfdocumentsearch - Versioning

Source Location

Git Repository URL: <https://git.ca1.hana.ondemand.com/> /sfdocumentsearch

[Edit Online](#)

History

[Commits](#) | **[Versions](#)**

[Refresh](#) | [Import from File](#)

Version	Active	Actions
1	✓	Edit Download

Required Destinations for Version "1"

Assign a destination from your subaccount to each required destination of the application.

Available	Required Destination (from Application Descriptor)	Mapped Subaccount Destination
✓	sf_odata	sf_odata

[Save](#) | [Cancel](#)

Next, register the second HTML5 application sfmydocuments. Return to the **BTP Cockpit** and navigate to the **BTP Sub Account** → **Applications** → **HTML5 Applications** and click the **sfmydocuments** application

SAP BTP Neo Cockpit

Overview | **Applications** | Java Applications | HANA XS Applications | Subscriptions | Services | Solutions

VersaFile, Inc. / SF_Integration_Dev

Subaccount: SF_Integration_Dev - HTML5 Applications

All: 2

[New Application](#) | [Import from File](#)

State	Application
Started	sfdocumentsearch
Started	sfmydocuments

Under '**Required Destinations**' set the '**Mapped Subaccount Destination**'

SAP BTP Neo Cockpit

Overview | Monitoring | Versioning | Roles

VersaFile, Inc. / SF_Integration_Dev / sfmydocuments

HTML5 Application: sfmydocuments - Overview

Application Details

Application Name: sfmydocuments
 Display Name: sfmydocuments
 Description:
 Edit

Actions

Started

Active Version

Active Application Version: 1
 Application URL: https://sfmydocuments.dispatcher.ca1.hana.ondemand.com

Required Destinations

Assign a destination from your subaccount to each required destination of the application.

Available	Required Destination (from Application Descriptor)	Mapped Subaccount Destination
✓	sf_mydata	sf_odata

Edit

Next, Click the **'Versioning'** section, under **'History'** section select the **'Versions'** and set the **'Mapped Subaccount Destination'**

SAP BTP Neo Cockpit

Overview | Monitoring | Versioning | Roles

VersaFile, Inc. / SF_Integration_Dev / sfmydocuments

HTML5 Application: sfmydocuments - Versioning

Source Location

Git Repository URL: https://git.ca1.hana.ondemand.com/sfmydocuments
 Edit Online

History

Comm | **Versions**

Refresh Import from File

Version	Active	Actions
1	✓	✕ ⬇

Required Destinations for Version "1"

Assign a destination from your subaccount to each required destination of the application.

Available	Required Destination (from Application Descriptor)	Mapped Subaccount Destination
✓	sf_mydata	sf_odata

Edit

5.2.2.2.1. BTP Command Line Interface (CLI)

This section has information on how to install the BTP CLI if it does not already exist in your environment.

The CLI can be downloaded from: <https://tools.hana.ondemand.com/#cloud>

SAP BTP SDK for the Neo environment

The SAP BTP SDK for the Neo environment includes platform API JARs, a local runtime for testing, documentation, samples, and the Neo console client. These are the major versions of the SAP BTP SDK for the Neo environment:

- SAP BTP SDK for the Neo environment, version for Java Web Tomcat 8 - provides support for some of the standard Java EE APIs (Servlet, JSTL, EL)
- SAP BTP SDK for the Neo environment, version for Java Web Tomcat 9 - provides support for some of the standard Java EE APIs (Servlet, JSTL, EL)
- SAP BTP SDK for the Neo environment, version for Java EE 7 Web Profile TomEE 7 - provides support for Java EE 7 Web Profile APIs

Available SAP BTP SDKs for the Neo environment

SAP BTP SDK for the Neo environment	Comment	Version	File Size	Download	Show old versions
Java Web Tomcat 8		3.166.7	56.8 MB	neo-java-web-sdk-3.166.7.zip (sha1)	
Java Web Tomcat 9		4.39.6	59.1 MB	neo-java-web-sdk-4.39.6.zip (sha1)	
Java EE 7 Web Profile TomEE 7		1.137.8	93.9 MB	neo-javaee7-wp-sdk-1.137.8.zip (sha1)	

NOTE: The above tools will require a Java Runtime Environment (JRE). If you do not have one scroll down on the page to the SAP JVM section and download a version appropriate for your Operating System

SAP JVM

The SAP JVM is a prerequisite for local profiling with the SAP JVM Profiler. It is a standard compliant certified JDK, supplemented by additional supportability and developer features and extensive monitoring and tracing facilities. For more information, see the [SAP JVM documentation](#)

Available SAP JVMs

Operating System*	Architecture	Version	File Size	Download
Linux	x86_64	8.1.091	130.0 MB	sapjvm-8.1.091-linux-x64.zip (sha1)
Linux	x86_64	8.1.091	126.0 MB	sapjvm-8.1.091-linux-x64.rpm (sha1)
Linux	ppc64le	8.1.091	126.0 MB	sapjvm-8.1.091-linux-ppc64le.zip (sha1)
Linux	ppc64le	8.1.091	119.0 MB	sapjvm-8.1.091-linux-ppc64le.rpm (sha1)
Mac OS X	x86_64	8.1.091	131.0 MB	sapjvm-8.1.091-macosx-x64.zip (sha1)
Windows	x86_64	8.1.091	162.0 MB	sapjvm-8.1.091-windows-x64.zip (sha1)

*SAP JVM is supported for: SUSE Linux Enterprise Server 12 and 15, Redhat Enterprise Linux 7 and 8, Oracle Linux 7 and 8, Windows 10, Windows Server 2012, 2012 R2, 2016, 2019 and 2022, Mac OS X 10.14 (Mojave), 10.15 (Catalina), 11 (Big Sur) and 12 (Monterey)

Download the above tools and unpack to a system directory. Install the JRE if required.

If you just installed a JRE, the tools require 'JAVA_HOME' be defined. This will be the path where the JRE was downloaded to.

set JAVA_HOME=C:*your directory path* \JVM\sapjvm_8\jre

5.2.2.3. Extension Admin Permissions

The extension administrators will require permissions to the BTP environments to complete the Application setup in the SuccessFactors Extensions Center portlet.

- In the SuccessFactors Admin Center, navigate to the **Permissions Role Detail**
- Select the Administrative Role that is doing the solution configuration
- Navigate to the Permission setting '**Permission not requiring target**'
- Select '**Manage Extensions on SAP BTP**'
 - Allow '**Create Integration with SAP BTP**'

Add the Administrative user who will be accessing the Portal Configuration to the Extensions Administrators Group. In SuccessFactors:

- Navigate to '**Manage Permission Groups**'
- Select the group name: **Extension Administrators**
- Add the selected UserIds to the role

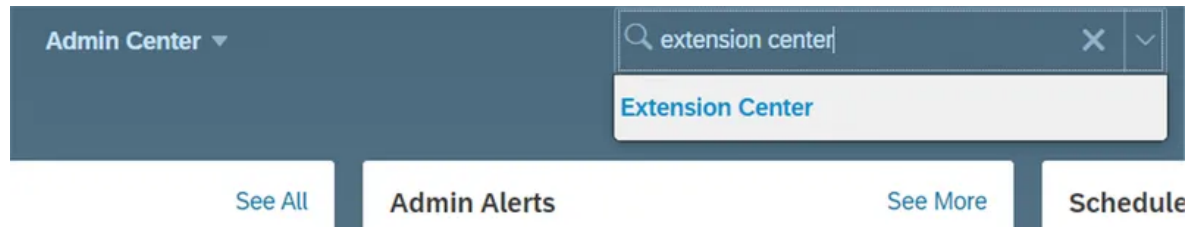
SAP Tech Note:

<https://help.sap.com/docs/btp/sap-business-technology-platform/register-sap-successfactors-system-in-global-account-in-sap-btp>

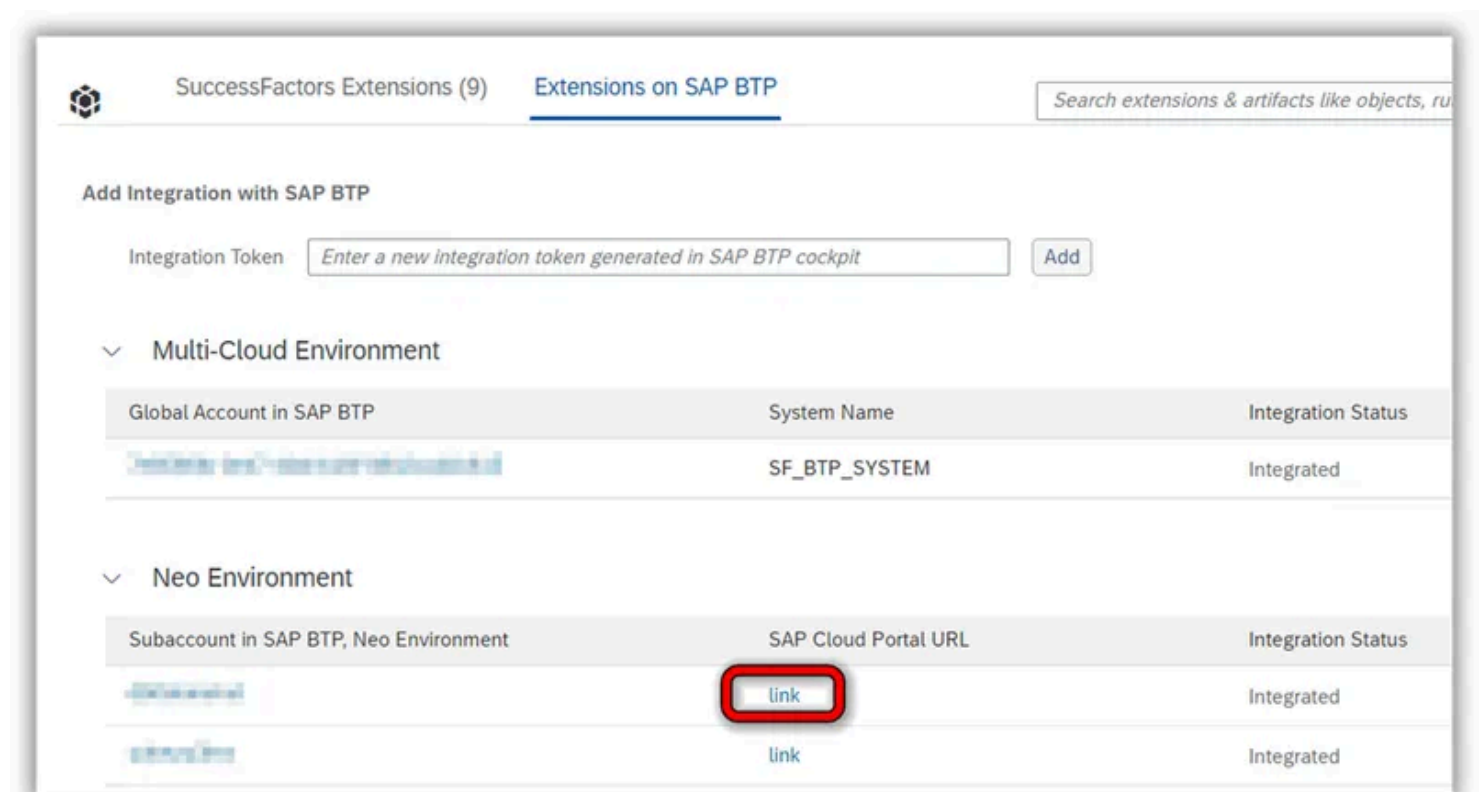
5.2.2.4. SuccessFactors App Setup

In this section, the HTML5 applications will be configured in SuccessFactors.

Log into SuccessFactors with an Administrative account and navigate to the **Extension Center**



Navigate to the '**Extensions on SAP BTP**' tab and click on the '**link**' under '**SAP Cloud Portal URL**'



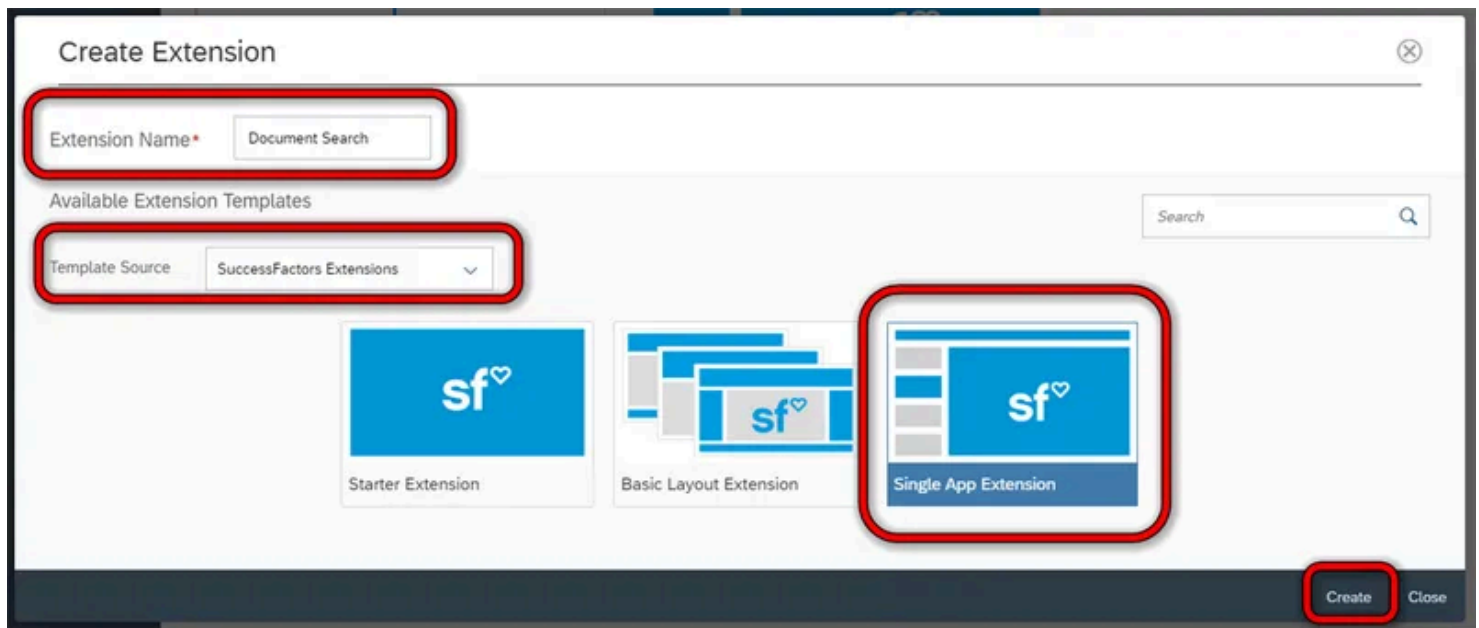
The SuccessFactors **Extension Directory** will open. Click on the giant plus button.



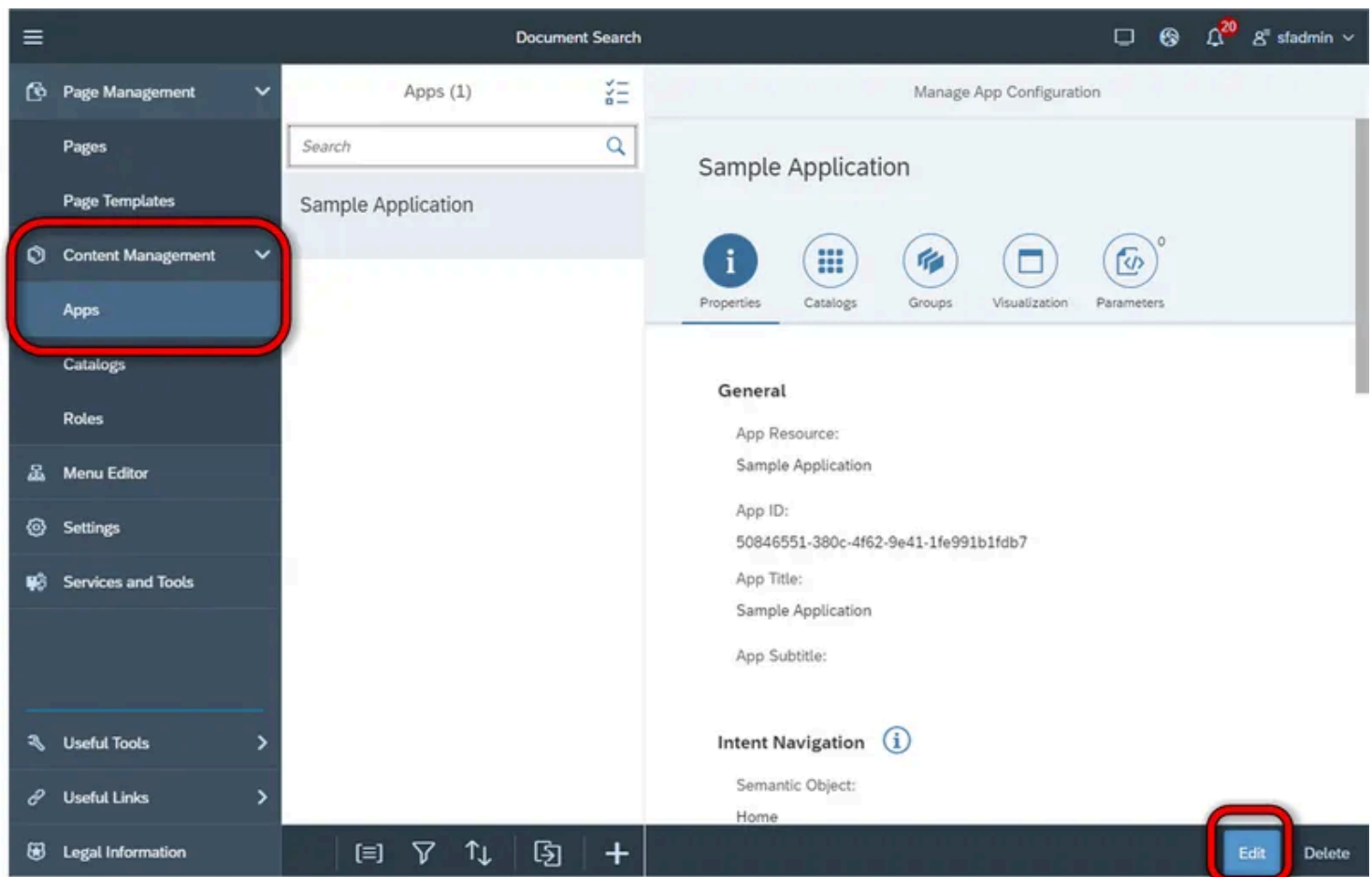
Set the following values for the new Extension and click the '**Create**' button

- Extension Name: **Document Search**

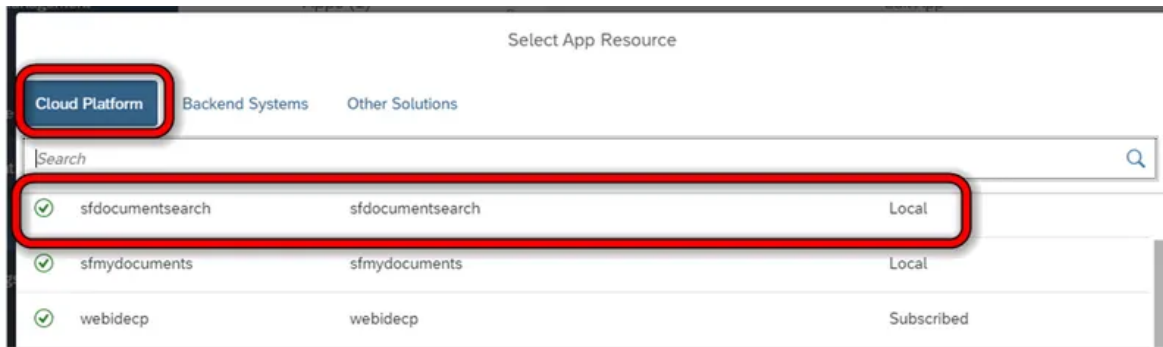
- Template Source: **SuccessFactors Extensions**
- Choose **Single App Extension**



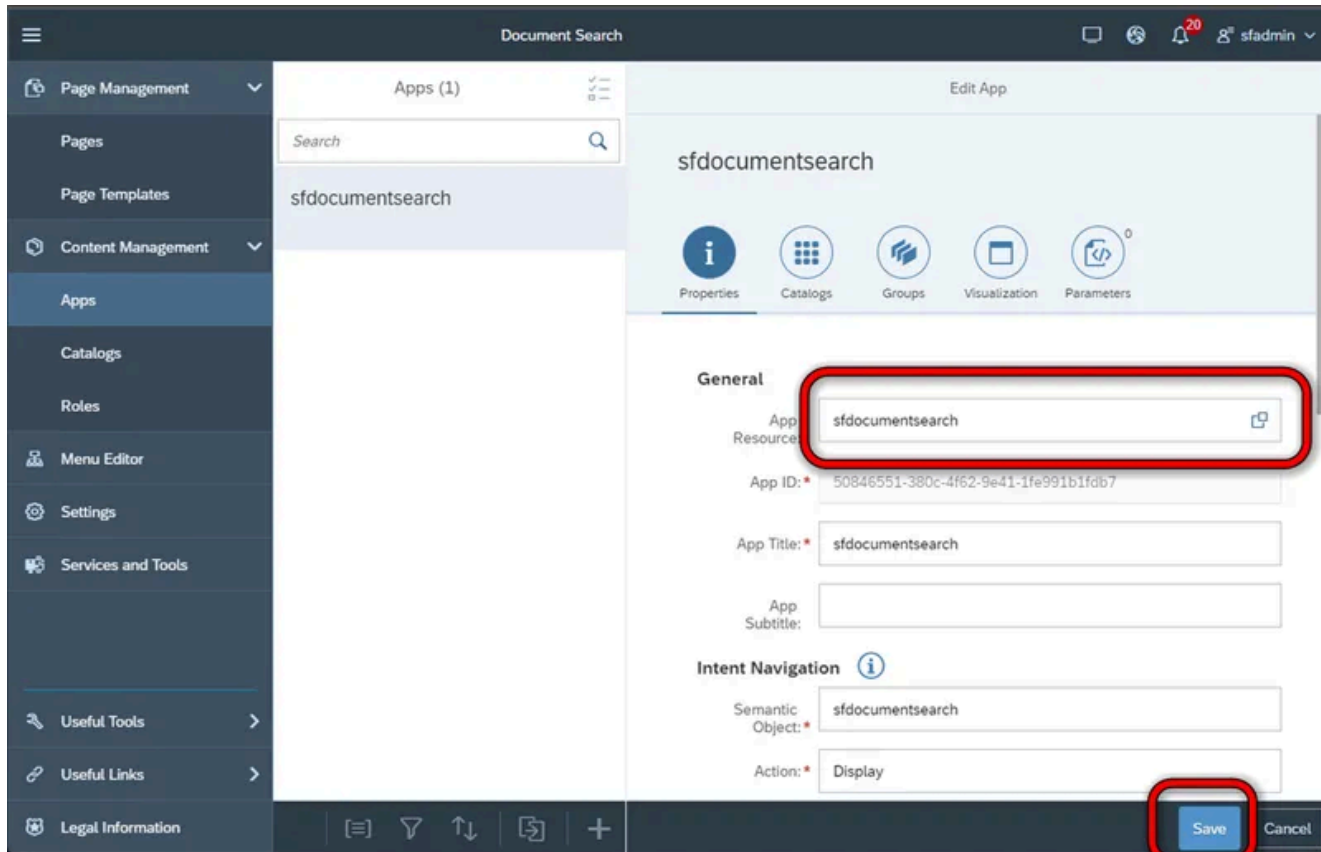
Once the Extension has been created, navigate to **Content Management** → **Apps** and click the **Edit** button



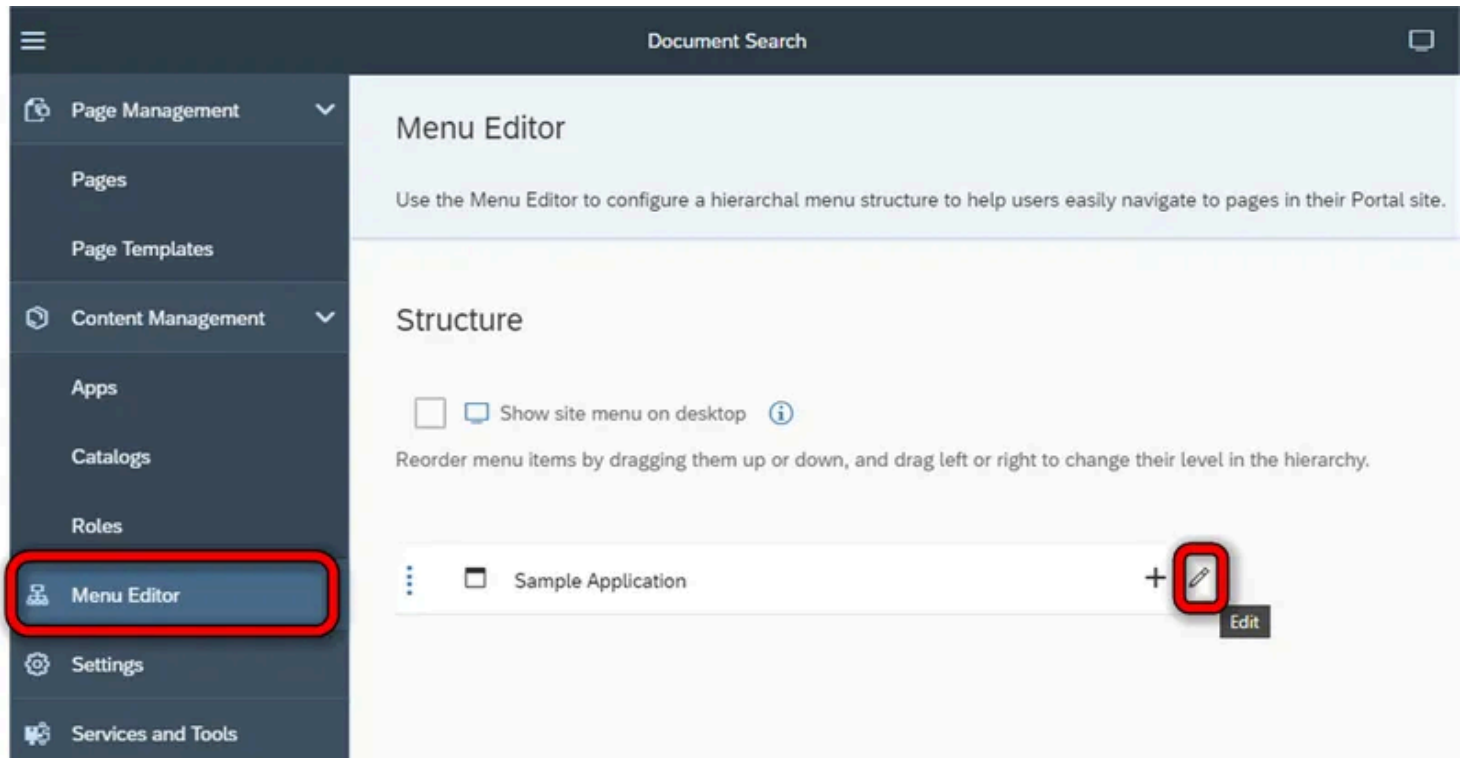
Choose the registered application from the BTP Platform (sfdocumentsearch)



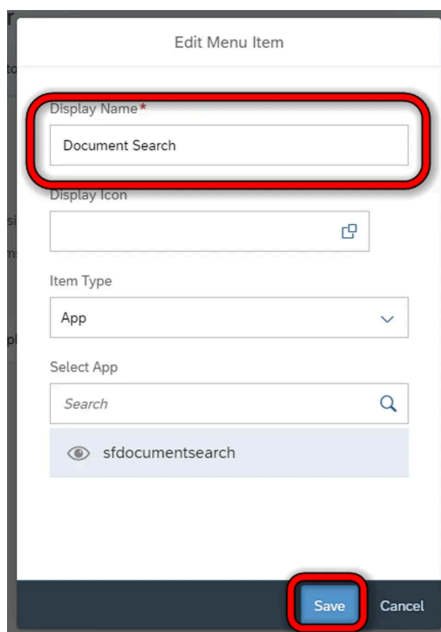
After selection the BTP Application click **SAVE**



Next, navigate to **'Menu Editor'** and click the **edit** pencil icon

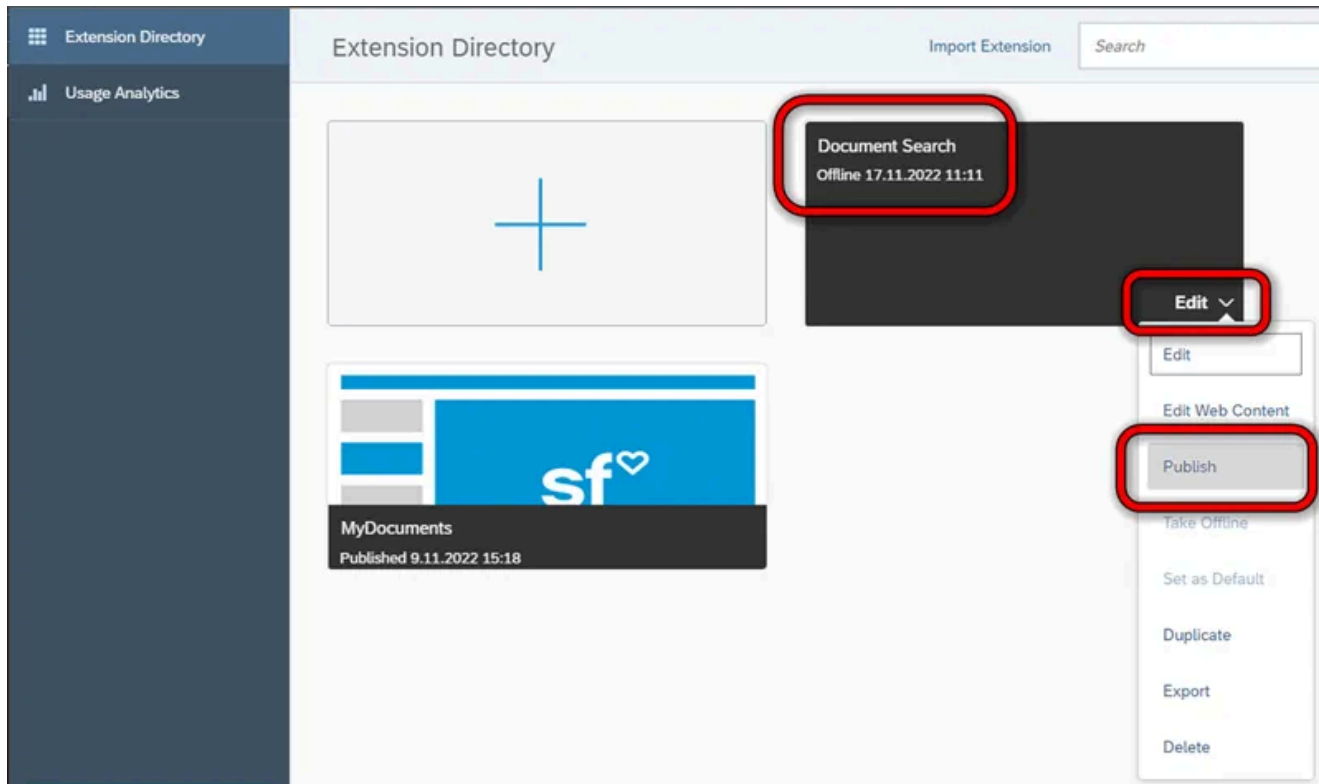


Set the display name to 'Document Search' and click the 'Save' button



After updating the menu display name, click the **'Save'** button in the lower right. You can now close the 'Menu Editor' browser tab and return to the **'Extension Directory'**

Select your new Extension, click the **'Edit'** button and select **'Publish'**



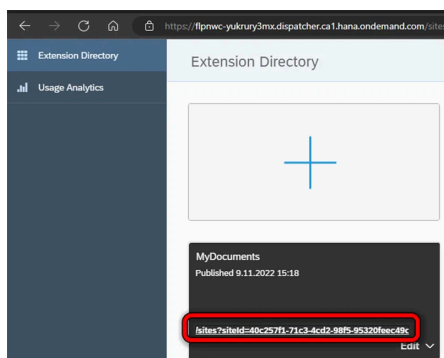
Custom Navigation Menu Setup

Next, register the Extension with the Custom Navigation in SuccessFactors

Navigate to the '**Configure Custom Navigation**' in SuccessFactors admin tools

Add a new line with the following values:

- Menu Item Label: Document Search
- ID: <next unused value>
- Link Type: External Link - New Tab/Window
- URL: Copy from the Extension Published



- Menu Location: <select where the menu should appear>
- Permissions: <select groups to see menu option>

NEXT Complete all of the Above Steps for the 'My Documents' application

5.2.3. SuccessFactors Configuration

This section describes the following configuration that must be completed in SuccessFactors.

5.2.3.1. API User

An API user is required for the docuflow for SuccessFactors middleware server.

Create a new user in SuccessFactors. This should be easily identifiable as an API user (i.e. apidflow)

Create a permission role for this user (i.e. apidflow_access)

The following permissions are required for the permission role:

Employee Data:

(Employee Profile)

- Country
- Email
- Employee Id
- Manager
- User ID
(User Information)
- Hire Date
(Employment Details)
- Employment Details MSS
- Hire Date

General User Permission:

- SFAPI User Login
- User Login
- SAP System Configuration:
 - View

Miscellaneous Permissions:

- Category Link Card 1 (view / edit)
- Category Link Card 3 (view / edit)
- Document Type Link Card 1 (view / edit)
- Document Type Link Card3 (view / edit)
- Employee Documents (view / view history / create / insert / correct / delete)
- Employee Documents.EmployeeDocumentChild (view / correct / create / adjust order / delete)
- HR Employee Documents (view / view history / create / insert / correct / delete)
- HR Employee Documents.HREmployeeDocumentChild (view / correct / create / adjust order / delete)
- Link Card Type 1 (View / Edit)
- Link Card Type 2 (View / Edit)
- Link Card Type 3 (View / Edit)
- Link Card Type 4 (View / Edit)
- cust_d4iFolderLink (View / Edit)
- SAP System Configuration (View)

Manage User

- User Account oData entity

Metadata Framework

- Admin access to MDF OData API

Manage Integration Tools

- oData API Attachment Export
- Allow Admin to Access OData API through Basic Authentication

Manage System Properties

- Picklist Management and Picklists Mappings Set Up

5.2.3.2. Create Pick List

The section will create the Pick List for document approval workflow status

Navigate to the Admin tool '**Pick List Center**'

Click '**New Pick List**'

In the Create Dialog, set the following values:

Name	Value	Notes
Code	LinkCard_DocStatus	
Name	Link Card Doc Status	
Status	Active	
Effective Start Date	<today's date>	
Display Order	Alpha	
Parent Picklist	<blank>	
Legacy Picklist ID	LinkCard_DocStatus	

Click the '**Save**' button

Create the following values:

Name	Value	Notes
External Code	1	
Label	InApproval	
Status	Active	
Non-unique External Code	1	

Name	Value	Notes
External Code	2	
Label	Rejected	
Status	Active	
Non-unique External Code	2	

Name	Value	Notes
External Code	3	
Label	Approved	
Status	Active	
Non-unique External Code	3	

5.2.3.3. Create Approval Workflow

The section will create the document approval workflow

Navigate to the Admin tool '**Manage Organization, Pay and Job Structures**'

Create New: Workflow

In the main workflow section, set the following values:

Name	Value	Notes
Workflow ID	d4i_WF	
Name	d4i_Approval	

In the section Step 1, set the following values:

Name	Value	Notes
Approver Type	Role	
Approver Role	Employee HR	
Context	Source	
Edit Transaction	Edit with Route Change	
No Approver Behaviour	Skip this Step	
Relationship to Approver	Employee	
Respect Permission	No	

5.2.3.4. SuccessFactors Objects

The following custom objects must be defined within the SuccessFactors administrative tools:

1. Category Link Card 1
2. Category Link Card 3
3. Document Type Link Card 1
4. Document Type Link Card 3
5. Link Card Type 1
6. Link Card Type 2
7. Link Card Type 3
8. Link Card Type 4
9. ESS Link Card Child
10. ESS Link Card Parent
11. HR Link Card Child
12. HR Link Card Parent
13. d4iFolderLink

NOTE: To access the administrative tools, navigate to '**Admin→Configure Object Definitions**'

5.2.3.4.1. Category Link Card 1

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	Category	
Effective Dating	None	
API Visibility	Editable	
Status	Active	
Label	Category Link Card 1	

In the '**Fields Section**', for '**external code**' select **details**

- In the details screen, set the value of '**Visibility**' to '**Not Visible**'
- NOTE: This prevents the UI from showing a count of the categories

In the '**Fields Section**', for '**external name**' select **details**

- In the details screen, set the value of '**Data Type**' to '**Translatable**'

Update the **Security Section** with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	<blank>	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	<blank>	

5.2.3.4.2. Category Link Card 3

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	Category_LinkCard3	
Effective Dating	None	
API Visibility	Editable	
Status	Active	
Label	Category Link Card 3	

In the '**Fields Section**', for '**external code**' select ***details***

- In the details screen, set the value of '**Visibility**' to '**Not Visible**'

In the '**Fields Section**', for '**external name**' select ***details***

- In the details screen, set the value of '**Data Type**' to '**Translatable**'

Update the **Security Section** with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	<blank>	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	<blank>	

5.2.3.4.3. Document Type Link Card 1

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	DocumentType_LinkCard1	
Effective Dating	None	
API Visibility	Editable	
MDF Version History	Complete History	
Status	Active	
Label	Document Type Link Card 1	

In the **Fields Section** for '**external code**' select **details**

- In the details screen, set the value of '**Visibility**' to '**Not Visible**'

In the '**Fields Section**', for '**external name**' select **details**

- In the details screen, set the value of '**Data Type**' to '**Translatable**'

In the **Associations** section set the following values:

Property	Value	Notes
Name	category	It will show as custom_category after its been created
Multiplicity	One to Many	
Destination Object	Category Link Card 1	Use the dropdown to find the custom object created for 'Category'
Type	Valid When	

Update the **Security** section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	<blank>	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	<blank>	

5.2.3.4.4. Document Type Link Card 3

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	DocumentType_LinkCard3	
Effective Dating	None	
API Visibility	Editable	
MDF Version History	Complete History	
Status	Active	
Label	Document Type Link Card 3	

In the **Fields Section** for '**external code**' select **details**

- In the details screen, set the value of '**Visibility**' to '**Not Visible**'

In the '**Fields Section**', for '**external name**' select **details**

- In the details screen, set the value of '**Data Type**' to '**Translatable**'

In the **Associations** section set the following values:

Property	Value	Notes
Name	CategoryLinkCard3	It will show as custom_CategoryLinkCard3 after its been created
Multiplicity	One to Many	
Destination Object	Category Link Card 3	Use the dropdown to find the custom object created for 'Category LinkCard3'
Type	Valid When	

Update the **Security** section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	<blank>	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	<blank>	

5.2.3.4.5. Link Card Type 1

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	d4i_LinkCard_1	
Effective Dating	None	
API Visibility	Editable	
Status	Active	
Label	Link Card Type 1	

Add the following fields to the d4i_LinkCard_1 object:

Name	Length	Data Type	More
user	100	user	Under details, set 'Required' to 'Yes'
Category	100	Generic Object	Valid Values Source: cust_Category_LinkCard3
DocumentType	100	Generic Object	Valid Values Source: cust_DocumentType_LinkCard3
Title	256	String	
UserID	38	String	
EmployeeName	256	String	
Country	128	String	
d4i_URL	256	String	
d4iUploadStatus	64	String	
CreatedDateTime	35	DateTime	
ownerId	38	String	
d4i_Status	64	String	

Update the Security section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	cust_user	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	mdfSystemEffectiveEndDate	

5.2.3.4.7. Link Card Type 3

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	d4i_LinkCard_3	
Effective Dating	None	
API Visibility	Editable	
Status	Active	
Label	Link Card Type 3	

Add the following fields to the d4i_LinkCard_3 object:

Name	Length	Data Type	More
user	100	user	Under details, set 'Required' to 'Yes'
Category	100	Generic Object	Valid Values Source: cust_Category_LinkCard3
DocumentType	100	Generic Object	Valid Values Source: cust_DocumentType_LinkCard3
Title	256	String	
UserID	38	String	
EmployeeName	256	String	
Country	128	String	
d4i_URL	256	String	
d4iUploadStatus	64	String	
CreatedDateTime	35	DateTime	
ownerId	38	String	
d4i_Status	64	String	

Update the Security section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	cust_user	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	mdfSystemEffectiveEndDate	

5.2.3.4.8. Link Card Type 4

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	d4i_LinkCard_4	
Effective Dating	None	
API Visibility	Editable	
Status	Active	
Label	Link Card Type 4	

Fields

Add the following fields to the d4i_LinkCard_4 object:

Name	Length	Data Type	More
user	100	user	Under details, set 'Required' to 'Yes'
Category	100	Generic Object	Valid Values Source: cust_Category_LinkCard3
DocumentType	100	Generic Object	Valid Values Source: cust_DocumentType_LinkCard3
Title	256	String	
UserID	38	String	
EmployeeName	256	String	
Country	128	String	
d4i_URL	256	String	
d4iUploadStatus	64	String	
CreatedDateTime	35	DateTime	
ownerId	38	String	
d4i_Status	64	String	

Security

Update the Security section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	cust_user	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	mdfSystemEffectiveEndDate	

5.2.3.4.9. ESS Link Card Child

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	d4i_LinkCard_ESS_Child	
Effective Dating	From Parent	
API Visibility	Editable	
Status	Active	
Label	Employee Document Child	
MDF Version History	Complete History	
Subject User Field		
Workflow Routing		
Pending Data	No	

Fields

In the '**Fields Section**', for '**externalName**' select **details**

- In the details screen, set the value of 'Visibility' to 'Not Visible'

Set the following fields on the cust_d4i_LinkCard_ESS_Child:

Name	Length	Data Type	More
externalCode	100	AutoNumber	This should exist, set type
Category	100	Generic Object	In Details , Set Valid Values Source to cust_category Set label to Document Category Set Required to Yes
DocumentType	100	Generic Object	In Details , Set Valid Values Source to cust_DocumentType_LinkCard1 Set Label to Document Type Also in Details, for Field Criteria, set 'Source Field Name' to cust_Category.mdfSystemInternalCode and 'Destination Field Name' to cust_Category Set Required to Yes
CollectionType	256	String	
d4iUploadStatus	38	Picklist	In Details , Set Valid Values Source to LinkCard_DocStatus
attachment	38	Attachment	Set Required to Yes

Security

Update the Security section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field		
CREATE Respects Target Criteria	No	
Base Date Field For Blocking		

Business Rule

The next step is to create a rule that will concatenate Category and Document Type into a single field that is used for data collection

Click on the plus next to 'Save Rules' to open '**Configure Business Rules**'

Select 'Basic'

Name	Value	Notes
Rule Name	LinkCard_CollectionType_Child	
Rule ID	LinkCard_CollectionType_Child	
Base Object	Employee Document Child	

In the **IF** statement section, select the '**Always True**' checkbox

Set the following Concatenation Rule:

Configure Business Rules

Search :Rule No Selection

LinkCard_CollectionType_Child (LinkCard_CollectionType_Child)

Scenario: Basic

Basic Information

Start Date 01/01/1900

Rule Type

Description

Parameters

Name	Object
Context	System Context 
Employee Document Child	Employee Document Child

 Variables If

 This rule is always true.
To add an expression please uncheck the Always True checkbox.

Then

Concatenate string

Save your changes to complete the rule

Navigate back to the Admin tool Configure Object Definitions

Open the ESS Link Card object and take action→make correction

Go to the section 'Save Rule'

- Select the **LinkCard_CollectionType_Child** rule that was just created and save the object

5.2.3.4.10. ESS Link Card Parent

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	d4i_LinkCard_ESS_Parent	
Effective Dating	Basic	
API Visibility	Editable	
Status	Active	
Label	Employee Documents	
MDF Version History	Complete History	
Default Screen	Employee Document Parent	
Subject User Field	externalCode	
Workflow Routing	d4i_Approval	
Pending Data	Yes	

Fields

Set the following fields on the cust_d4i_LinkCard_ESS_Parent object:

Name	Length	Data Type	More
externalCode	100	user	This should exist, set to user type Set Required: True

Associations

In the Associations section, add the following values:

Name	Multiplicity	Destination Object	Type
cust_EmployeeDocumentChild	One to Many	Employee Document Child	Composite

Security

Update the Security section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	externalCode	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	mdfSystemEffectiveEndDate	

5.2.3.4.11. HR Link Card Child

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	d4i_LinkCard_HR_Child	
Effective Dating	From Parent	
API Visibility	Editable	
Status	Active	
Label	HR Employee Document Child	
MDF Version History	Complete History	
Subject User Field		
Workflow Routing		
Pending Data	No	

Fields

In the '**Fields Section**', for '**externalName**' select **details**

- In the details screen, set the value of 'Visibility' to 'Not Visible'

Set the following fields on the cust_d4i_LinkCard_HR_Child:

Name	Length	Data Type	More
externalCode	100	AutoNumber	This should exist, set type
Category	100	Generic Object	In Details , Set Valid Values Source to cust_Category_LinkCard3 Set label to Document Category (set localization) Set Required to Yes
Document Type	100	Generic Object	In Details , Set Valid Values Source to cust_DocumentType_LinkCard3 Set Label to Document Type Also in Details, for Field Criteria, set 'Source Field Name' to cust_CategoryLinkCard3.mdfSystemInternalCode and 'Destination Field Name' to cust_Category Set Required to Yes
CollectionType	256	String	
d4iUpload Status	38	Picklist	In Details , Set Valid Values Source to LinkCard_DocStatus
attachment	38	Attachment	Set Required to Yes

Security

Update the Security section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field		
CREATE Respects Target Criteria	No	
Base Date Field For Blocking		

Business Rule

The next step is to create a rule that will concatenate Category and Document Type into a single field that is used for data collection

Click on the plus next to 'Save Rules' to open '**Configure Business Rules**'

Select '**Basic**'

Name	Value	Notes
Rule Name	HR_LinkCard_CollectionType_Child	
Rule ID	HR LinkCard_CollectionType_Child	
Base Object	HR Employee Document Child	

In the **IF** statement section, select the '**Always True**' checkbox

Set the following Concatenation Rule:

If

*This rule is always true.
To add an expression please uncheck the Always True checkbox.*

Then

Concatenate()

Concatenate string

- Sub string: ESS Link Card.Category Value.externalName
- Sub string: Text -
- Sub string: ESS Link Card.Document Type Value.externalName

Save your changes to complete the rule

Navigate back to the Admin tool Configure Object Definitions

Open the ESS Link Card object and take action→make correction

Go to the section 'Save Rule'

- Select the **HR_LinkCard_CollectionType_Child** rule that was just created and save the object

5.2.3.4.12. HR Link Card Parent

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	d4i_LinkCard_HR_Parent	
Effective Dating	Basic	
API Visibility	Editable	
Status	Active	
Label	HR Employee Documents	Localize
MDF Version History	Complete History	
Default Screen	HR Employee Documents	
Subject User Field	externalCode	
Workflow Routing	d4i_Approval	
Pending Data	Yes	

Fields

Set the following fields on the d4i_LinkCard_HR_Parent object:

Name	Length	Data Type	More
externalCode	100	user	This should exist, set to user type Required: True

Associations

In the Associations section, add the following values:

Name	Multiplicity	Destination Object	Type
HREmployeeDocumentChild	One to Many	HR Employee Document Child	Composite

Security

Update the Security section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	externalCode	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	mdfSystemEffectiveEndDate	

5.2.3.4.13. d4iFolderLink

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section

Property	Value	Notes
Code	d4iFolderLink	
Effective Dating	None	
API Visibility	Editable	
Status	Active	
Label	d4iFolderLink	

Fields

Set the following fields on the cust_d4iFolderLink object:

Name	Length	Data Type	More
BoxFolderID	64	String	
User	100	user	Required = True

Searchable Fields

Update the Security section with the following values:

Field	Status	Notes
cust_User	Active	
cust_BoxFolderID	Active	

Security

Update the Security section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	cust_User	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking		

5.2.3.5. Permission Provisioning

Navigate to the Admin tool '**Manage Permission Roles**'

Employee Self Service Permissions to Objects

Apply Document Upload to People Profile

- Select 'ESS' Role
- Select 'Permission...'
- Select 'Employee Views'
- Check 'Document Upload'

Apply Permissions to Objects

- Select 'ESS'
- Select 'Permission...'
- Select 'Miscellaneous Permissions'
- Set the following Values:

Name	Setting	Notes
Category Link Card 1	View	
Document Type Link Card1	View	
Employee Documents	View / Create	
Employee Documents.EmployeeDocument Child	View Current / Create	Field Overrides Collection Type Value : No Access d4iUploadStatus: No Access
Link Card Type 1	View	
Link Card Type 4	View	

Manager Self Service Permissions to Objects

- Select 'MSS'
- Select 'Permission...'
- Select 'Miscellaneous Permissions'
- Set the following Values:

Name	Setting	Notes
Category Link Card 3	View	
Document Type Link Card3	View	
Link Card Type 1	View	
Link Card Type 2	View	

HRBP Permissions to Objects

- Select 'HRBP Role'
- Select 'Permission...'
- Select 'Miscellaneous Permissions'
- Set the following Values:

Name	Setting	Notes
Category Link Card 1	View	
Category Link Card 3	View	
Document Type Link Card 1	View	
Document Type Link Card 3	View	
HR Employee Documents	View Current / View History / Create / Correct / Delete	
HR Employee Documents.HREmployeeDocumentChild	View / Correct / Create / Delete	Field Overrides Collection Type Value : No Access d4iUploadStatus: No Access
Link Card Type 1	View / Edit	
Link Card Type 2	View / Edit	
Link Card Type 3	View / Edit	
Link Card Type 4	View / Edit	

Client Administrator Permissions to Objects

- Select 'Client Administrators Role'
- Select 'Permission...'
- Select 'Miscellaneous Permissions'
- Set the following Values:

Name	Setting	Notes
Category Link Card 1	View	
Category Link Card 3	View	
Document Type Link Card 1	View	
Document Type Link Card 3	View	
HR Employee Documents	View Current / View History / Create / Correct / Delete	
HR Employee Documents.HREmployeeDocumentChild	View / Correct / Create / Delete	Field Overrides Collection Type Value : No Access d4iUploadStatus: No Access
Link Card Type 1	View / Edit	
Link Card Type 2	View / Edit	
Link Card Type 3	View / Edit	
Link Card Type 4	View / Edit	

5.2.3.6. Export Category and Document Type

Use these instructions to get a set of CSV files to use to import the Categories and Document types into the system

For Employee Document Upload Rules/Categories:

Go to **'Import and Export Data'**

For Categories:

- Select the action to perform: Download Template
- Select Generic Object: Category Link Card 1
- Include dependencies: No
- Click 'Download'

For Document Type LinkCard1:

- Select the action to perform: Download Template
- Select Generic Object: Document Type LinkCard 1
- Include dependencies: No
- Click 'Download'

For the Relationship between Category and Document Types for Link Card 1

- Select the action to perform: Download Template
- Select Generic Object: Document Type LinkCard 1 - Category
- Include dependencies: No
- Click 'Download'

For HR Document Upload Rules/Categories:

Go to **'Import and Export Data'**

For Categories:

- Select the action to perform: Download Template
- Select Generic Object: Category
- Include dependencies: No
- Click 'Download'

For Document Type LinkCard3:

- Select the action to perform: Download Template
- Select Generic Object: Document Type LinkCard 3
- Include dependencies: No
- Click 'Download'

For the Relationship between Category and Document Types for Link Card 3

- Select the action to perform: Download Template
- Select Generic Object: Document Type LinkCard 3 - Category

- Include dependencies: No
- Click 'Download'

5.2.3.7. Import Category and Document Types

Use these instructions to use a set of CSV files to use to import the Categories and Document types into the system

For Employee Document Upload Rules/Categories:

Go to '**Import and Export Data**'

For Categories:

- Select the action to perform: Import Data
- Select Generic Object: Category
- File: CSV with your Category Data
- Purge Type: Incremental Load
- Click 'Import'

To check the imported data:

Go to '**Manage Data**'

- Under '**Search**' select Category
- Under the selection dropdown, the values should be shown

Complete the above steps for:

- Category Link Card 1.csv
- Category Link Card 3.csv
- Document Type Link Card 1.csv
- Document Type Link Card 3.csv
- Document Type Link Card 1-category.csv
- Document Type Link Card3-CategoryLinkCard3.csv

5.2.3.8. UI Configuration

This step will place the 'Upload Documents' portal option for the Workflow approval in the Employee File UI interface

Configure UI Element

Navigate to the admin tool '**Manage Configuration UI**'

Employee Documents

Create **Employee Documents**

- Click on 'Create New'
- Select '**Employee Documents**' in the Select Base Object
- Set the ID to '**Employee Documents Parent**'
- Click SAVE

Complete the following configuration:

- Employee Documents Header Element : Title Type User Defined = Employee Documents: (NOTE Set Translations as well)
- externalCode : Visible = No
- externalName : Visible = No
- effectiveStartDate : Visible = No
- In the Child Documents Header Section
 - Document Header : Title Type User Defined = Document (NOTE Set Translations as well)
 - Visible Fields Count = 8
 - Set Order of Properties: Document Category, Document Type, attachment
 - Hide the other fields

HR Employee Documents

Create **HR Employee Documents**

- Click on 'Create New'
- Select '**HR Employee Documents**' in the Select Base Object
- Set the ID to '**HR Employee Documents Parent**'
- Click SAVE

Complete the following configuration:

- HR Employee Documents Header Element : Title Type User Defined = HR Employee Documents: (NOTE Set Translations as well)
- externalCode : Visible = No
- externalName : Visible = No
- effectiveStartDate : Visible = No
- In the Child Documents Header Section
 - Document Header : Title Type User Defined = Document (NOTE Set Translations as well)
 - Visible Fields Count = 8

- Set Order of Properties: Document Category, Document Type, attachment
 - Hide the other fields

Configure People Profile

Navigate to the admin tool '**Configure People Profile**'

Click on '**Add New Section**' to insert a new UI section into the Employee Profile Screen

- Name the Section ***Document Upload***
- Name the Sub-Section: ***Documents***

Click on the Left side area

- Drop a 'Live Profile MDF Information' block
- Set the **MDF Screen ID** to '***Employee Document Parent***'

Click on the Right side area

- Drop a 'Live Profile MDF Information' block
- Set the **MDF Screen ID** to '***HR Employee Document Parent***'
- Save

5.2.4. docuflow for Integrations (d4i) Middleware Configuration

The goal of this guide is to establish base installation, initial configuration, and connectivity testing to a point that docuflow d4i components can communicate with one another and allows configuration of the software.

The docuflow application consists of a core Transformation engine and a series of connectors. Each connector has one or more connections defined which provide the information required to connect to the end applications.

NOTE: docuflow d4i was previously known as TRex and there are still occasional references to this name in the product

5.2.4.1. Start & Stop Order

d4i can be started and stopped via command line or as services (if installed as service)

Start Order

1. connectors (CNN-* eg. CNN-FS) (*no dependency*)
2. datastore (DAO)
3. config manager (CFGMGR)
4. transform engine (NGIN)

Stop Order

1. connectors (CNN-* eg. CNN-FS)
2. config manager (CFGMGR)
3. transform engine (NGIN)
4. datastore (DAO)

5.2.4.2. docuflow Middleware Installation

5.2.4.2.1. Unpacking ZIP Package

The docuflow middleware is delivered as a simple compressed 'zip' package. To install docuflow, simply unpack the zip file to a location you would like on the filesystem. For example d:\d4i

docuflow requires a java runtime environment (JRE) to run. By default, docuflow will reference the JRE deployed into the JDK directory in the location docuflow was installed. For example, if docuflow was installed to d:\d4i the JRE should be located at d:\d4i\JRE.

Please see '**Minimum Requirements**' in the technical overview for supported JRE versions.

5.2.4.2.2. Initial Configuration

The base docuflow installation will come pre-configured with the core system, Filesystem, Box and SuccessFactors connectors. The main task to complete for initial setup is the configuration of connections for these connectors.

The internal docuflow connectors and core components communicate through HTTPS/TLS encrypted connections. To enable these secure connections, x.509 certificates must be supplied. docuflow provides mechanisms to generate these certificates.

Options when generating certificates include:

- Configuration of which connector certificates to create
- Configuration of the server IP address/DNS entry when remote connectors are required
- Creation of passwords for certificates generated

Settings are configured in the file `<install location>\config\trex-certmgr-application.json`

The `trex-certmgr-application.json` file contains the following sections:

"defaults"

"defaults" : this section allows settings that will be used for all certificates generated. This can be overridden in the specifics of each individual certificate section

"master"

"master": This is the section for the master keystore for all configured connectors on the system. The default settings will be sufficient for a one box installation.

For each connector, the value for **"disabled"** should be set to **'false'** to have a certificate generated.

If distributing a connector on a remote server, the **"san"** section can be used to define the remote server name and/or ip address.

An example snippet for the file system connector is shown below, including a "san" section.

```
{
  "disabled": false,
  "alias": "trex.cnn-trex-fs.server",
  "dn": {
    "commonName": "VersaFile TRex File System Connector"
  },
  "san": {
    "dns": ["fully.qualified.name"],
    "ip": ["127.0.0.1"]
  }
}
```

"keystores"

"keystores": The keystores section is where docuflow configuration for the p12 keystore certificates is done. Each connector must have an alias entry in the section "keystore": **"trex.cnn.keystore.p12"**. The **"storepass"** entry is where a password for the keystore can be configured.

The section that will need configuration is the trex.cnn.keystore.p12. This contains references to each of the connectors being used.

"truststores"

"truststores": The truststores section is where docuflow configuration for the p12 truststore certificates is done. These trusts allow the connectors and core docuflow components to exchange their certificates.

The **"storepass"** entry allows for configuration of the password for the truststore.

There are two sections that will need configuration:

- trex.ngin.truststore.p12 (allows connections to the core d4i transformation engine)
- trex.cfgmgr.truststore.p12 (allows connections to the d4i configuration UI engine)

Both of these sections contain references to each of the connectors being used. **Note:** this should match the list from the keystore section.

An example of this section is shown below, with 5 connectors being setup:

```
"truststores": [
  {
    "keystore": "trex.dao.truststore.p12",
    "storepass": "<password>",
    "aliases": [ "trex.cfgmgr.server", "trex.ngin.server" ]
  },
  {
    "keystore": "trex.ngin.truststore.p12",
    "storepass": "<password>",
    "aliases": [
      "trex.dao.server",
      "trex.cfgmgr.server",
      "trex.cnn-trex-fs.server",
      "trex.cnn-cm8.server",
      "trex.cnn-elst.server",
      "trex.cnn-trex-util.server",
      "trex.cnn-ibm-p8ce.server"
    ]
  },
  {
    "keystore": "trex.cfgmgr.truststore.p12",
    "storepass": "<password>",
    "aliases": [
      "trex.cfgmgr.client",
      "trex.dao.server",
      "trex.ngin.server",
      "trex.cnn-trex-fs.server",
      "trex.cnn-cm8.server",
      "trex.cnn-elst.server",
      "trex.cnn-trex-util.server",
      "trex.cnn-ibm-p8ce.server"
    ]
  },
  {
    "keystore": "trex.cnn.truststore.p12",
    "storepass": "<password>",
    "aliases": [ "trex.cfgmgr.server", "trex.ngin.server" ]
  }
],
```

"exports"

"exports": The exports section configures the exported certificates for the system. Of note is the **"privatekeys"** section. This will generate a .p12 certificate that can be imported into a web browser to allow connectivity to the docuflow configuration service. Set a storepass that will be used. NOTE: Here you can configure a separate certificate/password for different environments (DEV vs PROD etc).

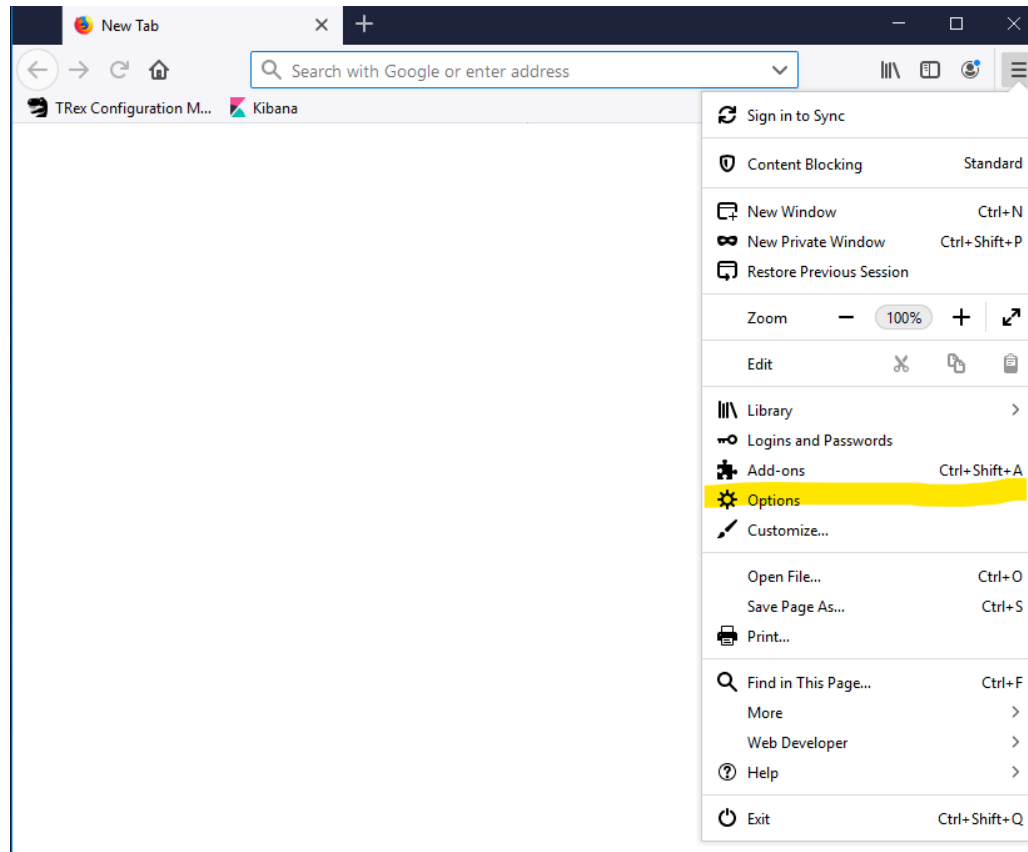
start-cert-mgr.ps1

Once configuration is complete, the powershell script '**<install location>start-cert-mgr.ps1**' should be run. This script will start a java application that will generate (or re-generate) the application certificates.

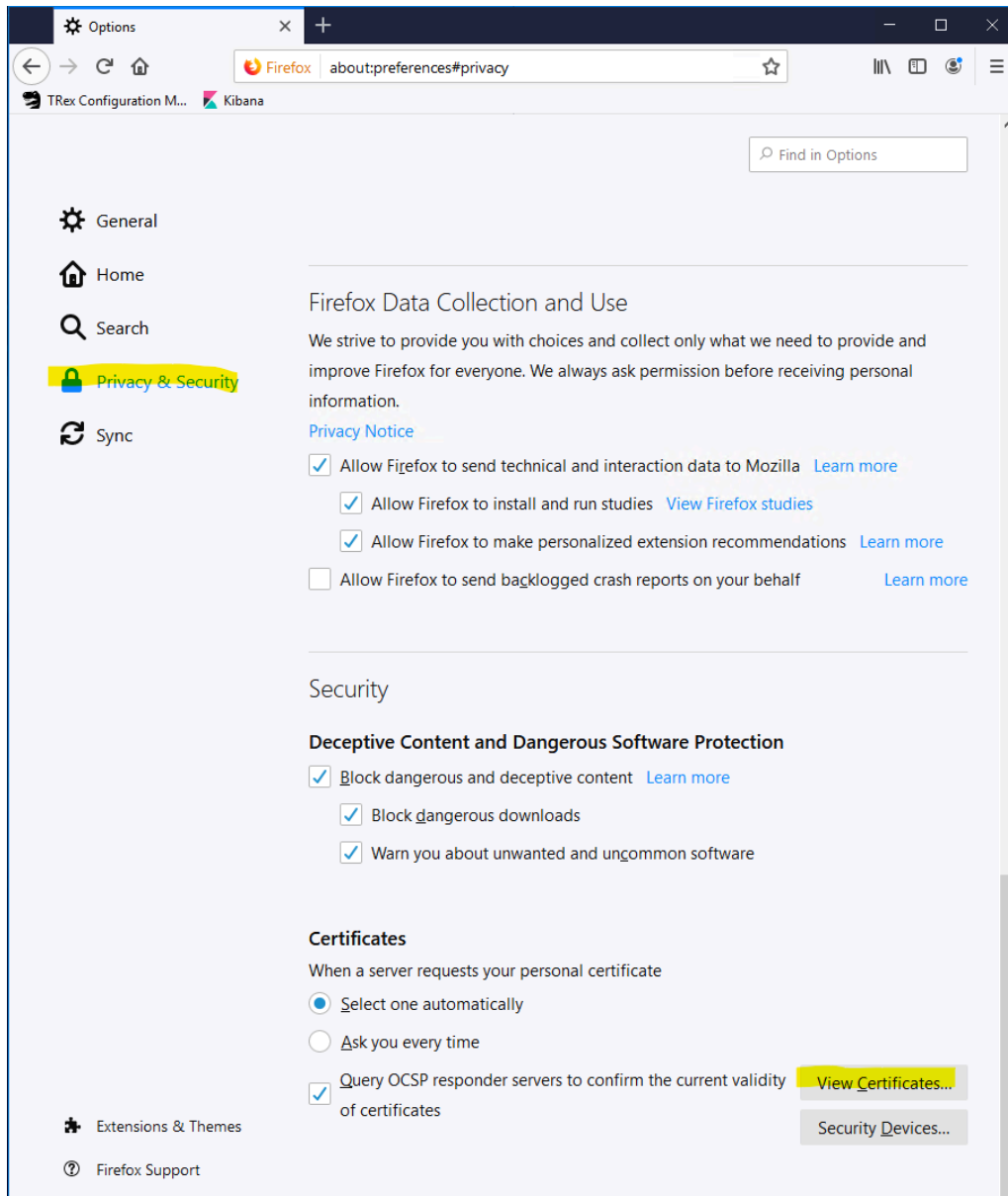
5.2.4.2.3. Setting up Firefox Certificate

The docuflow configuration service requires a HTTPS secured connection in order to communicate with a web browser. To enable this connection, a client certificate must be imported into the browser. This procedure explains how to install the certificate into Mozilla Firefox.

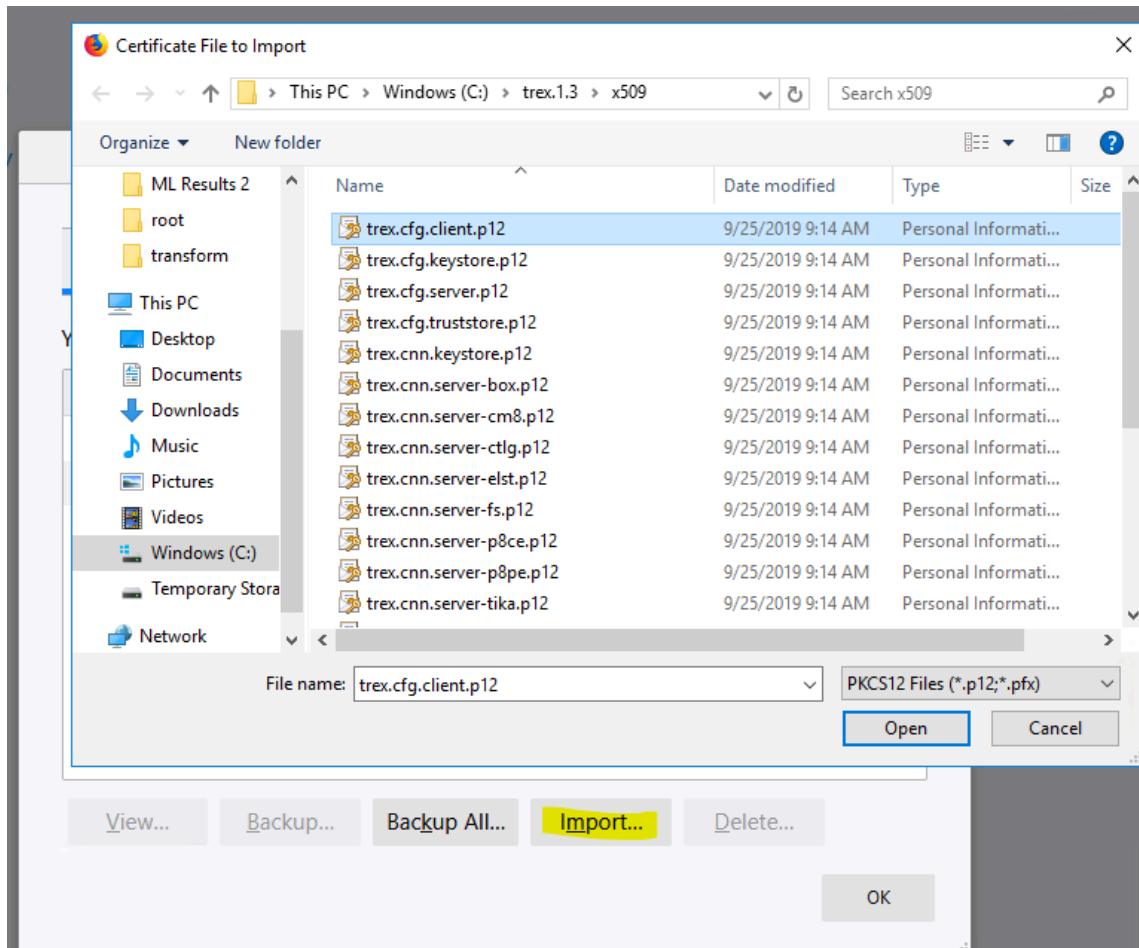
Open the Options section of Firefox



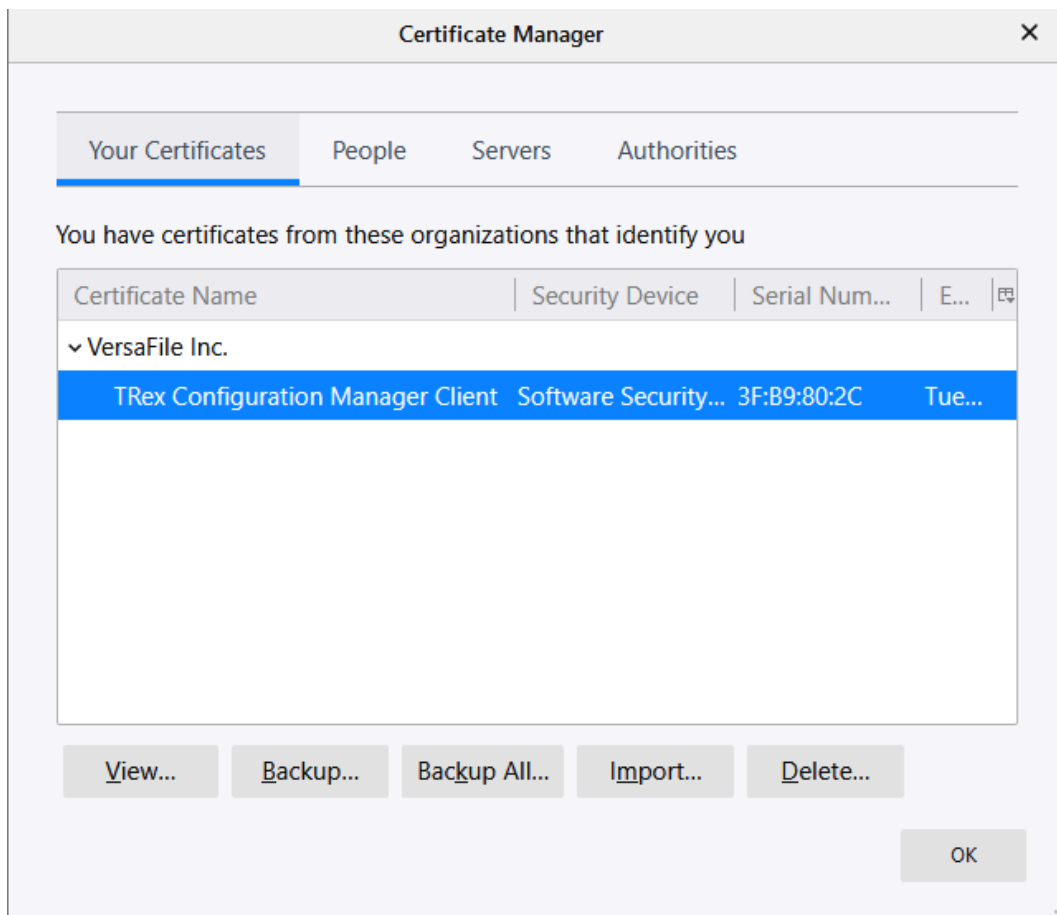
Select '**Privacy & Security**' section, scroll to the bottom and select View Certificates and then click '**Import**'



Select the 'exports' directory where you have created the x509 certificates (typically a directory called x509 in the root docuflow folder) and then select '***trex.cfgmgr.client.p12***' then supply the password that was configured in the '***trex-certmgr-application.json***' file.



You should see something similar to the following



5.2.4.2.4. Connecting to docuflow Configuration

Once all of the configuration steps are completed and the client certificate has been imported into your browser, Launch the windows power shell, go to File **<installocation>**, then run the powershell script **.\start-trex.ps1**.

If the file **<installocation>\start-trex.ps1** cannot be loaded, then the file **<installocation>\start-trex.ps1** is not digitally signed. And you cannot run this script on the current system. You will need to change the Execution policy to run these PowerShell scripts.

To display the execution policies for each scope in the order of precedence, use **"Get-ExecutionPolicy -List"**. To see the effective execution policy for your PowerShell session, use **"Get-ExecutionPolicy"** with no parameters.

The effective execution policy is determined by order of precedence as follows:

- MachinePolicy: Set by a Group Policy for all users of the computer.
- UserPolicy: Set by a Group Policy for the current user of the computer.
- Process: Affects only the current PowerShell session.
- CurrentUser: Affects only the current user.
- LocalMachine: Default scope that affects all users of the computer.

To change and set a new execution policy for the local computer for instance:

Use **Set-ExecutionPolicy** to specify the RemoteSigned policy. Then enter the Scope parameter to specify the default scope value, LocalMachine.

Set-ExecutionPolicy -ExecutionPolicy RemoteSigned -Scope LocalMachine

For more information on changing the execution policy, please visit <https://go.microsoft.com/fwlink/?LinkID=135170>.

After changing the execution policy, run **D:\docuflow> .\start-trex.ps1** A number of PowerShell windows will open up. Type "R" and run all scripts.

Once all scripts run, you can connect to the TRex Configuration Service UI. <https://<servername/localhost>:8443>

5.2.4.2.5. SSL Certificates Modifications

docuflow requires SSL certificates to secure communications amongst applications within the platform. docuflow does provide the ability to create its own self-signed certificates. Certificates are generated through the Certificate Manager application.

The following are execution guidelines for managing the certificates for docuflow .

Regenerate and Configure Certificates

After completing file modifications, such as those noted below, the following steps are required to regenerate and configure the new certificates for use with the docuflow platform.

1. Stop all docuflow applications
 2. Rename (or delete) the <docuflow INSTALL LOCATION>\x509 directory
e.g. x509.old
 3. Start the Certificate Manager application
 1. <docuflow INSTALL LOCATION>\start-cert-mgr.ps1
 2. Upon successful completion, exit/stop the Certificate Manager application
 4. Start docuflow
 5. Close and open a new browser instance
 6. Import the newly generated certificates; follow the existing steps [here](#).
 1. Note: it not required, but recommended, to remove the previously configured certificates:
 1. Personal > VersaFile Configuration Manager Client
 2. Trusted Root Certification Authorities > VersaFile TRex Root CA
-

Modify the Expiry Length

- The expiry length for certificates is located within the <docuflow INSTALL LOCATION>\config\trex-certmgr-application.json file
 - within the defaults.validity object
 - default value 3650 (units in days; 10 years)
 - To modify the expiry length of all certificates, edit the trex-certmgr-application.json file
 1. Modify the defaults.validity value
e.g. for 1 year expiry length

```
"validity": 365
```
 2. Save the file edit
 3. Execute the [Regenerate and Configure Certificates](#) steps, noted above.
-

Modify a Password

- Each application with the docuflow platform is created with its own truststore and keystore, to permit or restrict communication amongst applications.
- Passwords exist for each application's truststore and keystore. These passwords exist in:

- <docuflow INSTALL LOCATION>\config\trex-certmgr-application.json
- One of the following:
 - <docuflow INSTALL LOCATION>\config\application-cfgmgr.yaml
 - <docuflow INSTALL LOCATION>\config\application-dao.yaml
 - <docuflow INSTALL LOCATION>\config\application-ngin.yaml
 - <docuflow INSTALL LOCATION>\config\application-cnn.yaml
 - each connector has its own respective yaml config file; however, the application-cnn.yaml is inherited in each case as well
- A password configuration also exists for the client certificate, which exists in <docuflow INSTALL LOCATION>\config\trex-certmgr-application.json. Specifically:
 - exports.privatekeys.storepass object
- To modify a password, edit the trex-certmgr-application.json file and edit the respective yaml file, as required.
 - For example, to modify the password for both Configuration Manager certificates and the client certificate:

1. trex-certmgr-application.json edits:

1. Navigate to keystores.keystore : "trex.cfgmgr.keystore.p12"

2. Edit the storepass object, e.g.

```
{
  "keystore": "trex.cfgmgr.keystore.p12",
  "storepass": "newPassword1",
  "aliases": ["trex.cfgmgr.server"]
},
```

3. Navigate to truststores.truststore : "trex.cfgmgr.truststore.p12"

Edit the storepass object, e.g.

```
{
  "keystore": "trex.cfgmgr.truststore.p12",
  "storepass": "newPassword2",
  "aliases": [
    "trex.cfgmgr.client",
    ...
    "trex.cnn-trex-websvc.server"
  ]
},
```

4. Navigate to exports.privatekeys

Edit the storepass object, e.g.

```
"privatekeys": [{
  "alias": "trex.cfgmgr.client",
  "file": "exports/trex.cfgmgr.client.p12",
  "storepass": "newPassword3"
}]
```

2. application-cfgmgr.yaml edits

1. Navigate to and edit the server.ssl object

2. Edit the key-store-password and trust-store-password values, e.g.

```
server:
  port: 8443
```

```

ssl:
  key-store: ./x509/trex.cfgmgr.keystore.p12
  key-store-password: newPassword1
  key-alias: trex.cfgmgr.server
  trust-store: ./x509/trex.cfgmgr.truststore.p12
  trust-store-password: newPassword2

```

3. Execute the [Regenerate and Configure Certificates](#) steps, noted above.

- Password creation guidelines:

- It is recommended not to use, but if using the \ or " characters, these must be escaped with the \ character within the json and yaml files, e.g.
 - not escaped: *this"isaPwd*
 - escaped: *this|"is||aPwd*
- Although not used by default, it is recommended to surround the passwords in the yaml file with double quotes. e.g.

```

server:
  port: 8443
  ssl:
    key-store: ./x509/trex.cfgmgr.keystore.p12
    key-store-password: "newPassword1"
    key-alias: trex.cfgmgr.server
    trust-store: ./x509/trex.cfgmgr.truststore.p12
    trust-store-password: "newPassword2"

```

Modify Access to Configuration Manager

- By default, Configuration Manager is configured as accessible locally through a browser through <http://localhost:8443>
- An IP or DNS configuration can be added and included in the certificate creation.

1. Edit the `trex-certmgr-application.json` file
2. Navigate to the `master.certificates` object
3. Edit the `"alias": "trex.cfgmgr.server"` object

1. For a new DNS entry, add in a new entry in the `"dns"` array. For example, to add a DNS config with a name of *trex.cfgmgrtest.com* with an IP of *188.77.166.88*

```

{
  "alias": "trex.cfgmgr.server",
  "dn": {
    "commonName": "VersaFile Configuration Manager Application"
  },
  "san": {
    "dns": ["localhost", "trex.cfgmgrtest.com"],
    "ip": ["127.0.0.1", "188.77.166.88"]
  }
},

```

Note: an IP can be added on its own, without a DNS entry

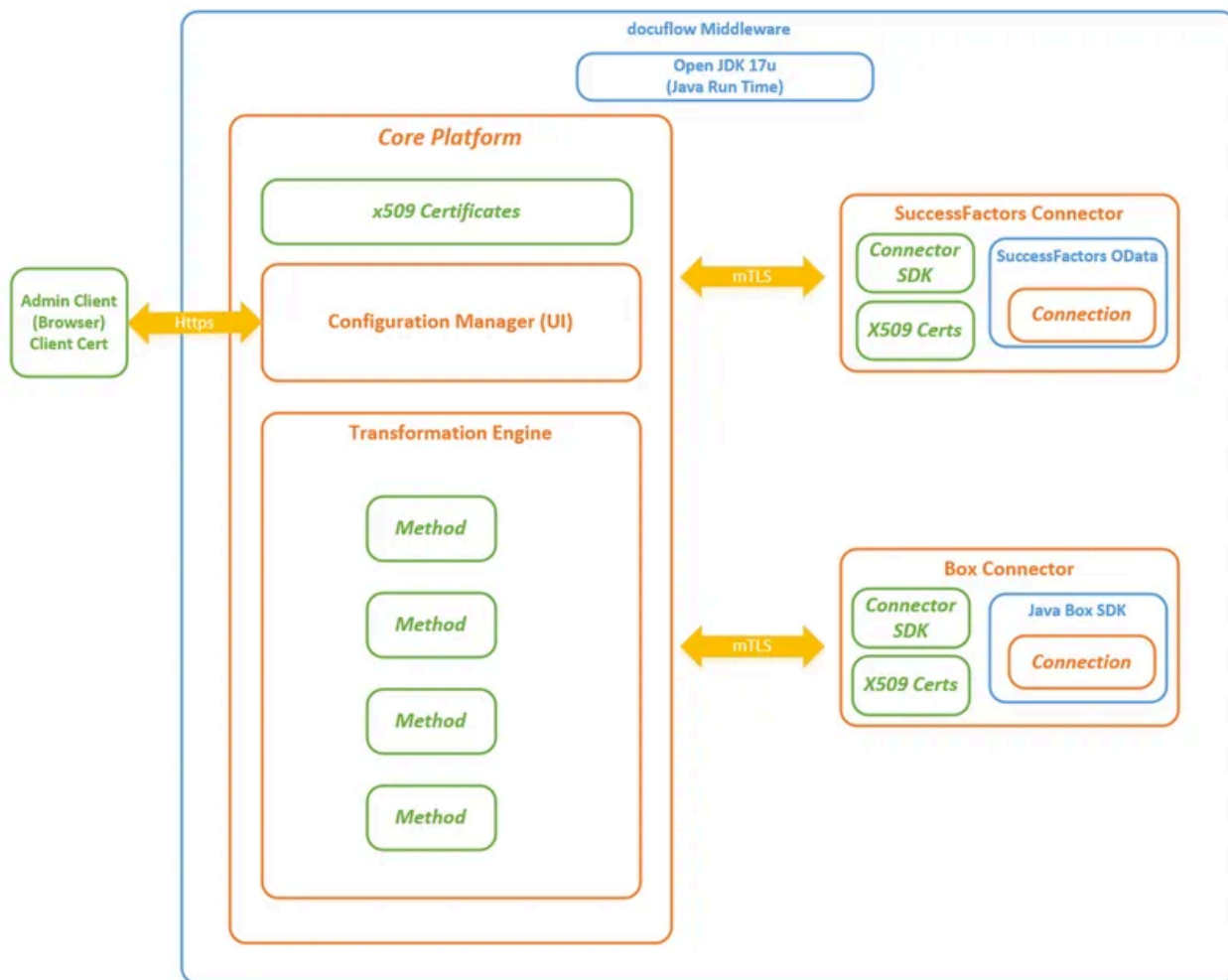
4. Execute the [Regenerate and Configure Certificates](#) steps, noted above.

5.2.4.3. docuflow Base Configuration

The Configuration of docuflow falls into four key components:

1. Engines
2. Connectors
3. Connections
4. Transformations

For docuflow for SuccessFactors and Box, the following architecture and connectors are used:



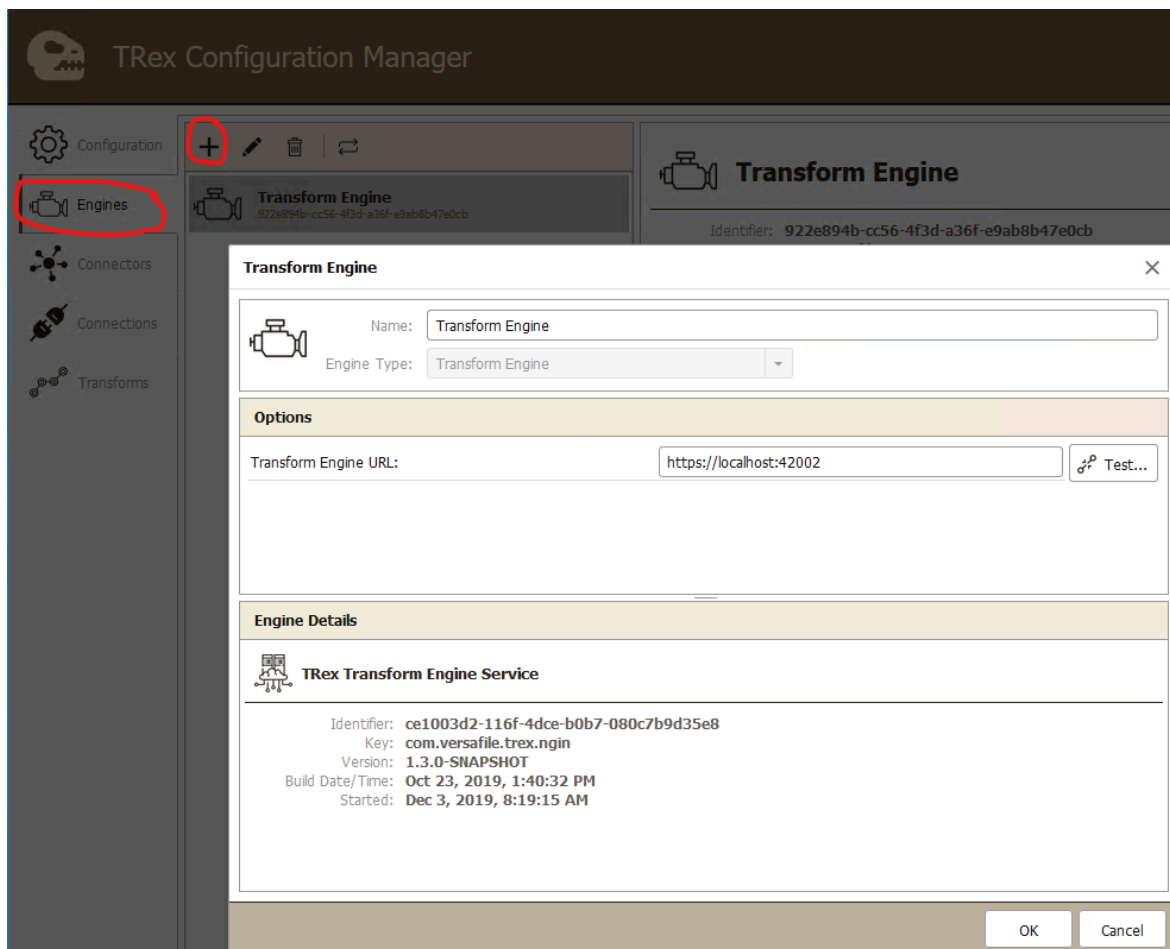
5.2.4.3.1. Engines

To add the main Transformation Engine to docuflow, select the 'Engines' tab on the right, then click on the '+' button

Fill in the URL using HTTPS protocol to the Transformation Engine service. By default this will be on the local machine on port 42002.

For example: `https://localhost:42002`

Click on the 'Test' button and after a successful connection, the Engine Details will be shown.



5.2.4.3.2. Connectors

Each connector is configured by linking

Select the 'Connectors' tab, then click the '+' button

Enter the URL to the connector endpoint using HTTPS and providing the port

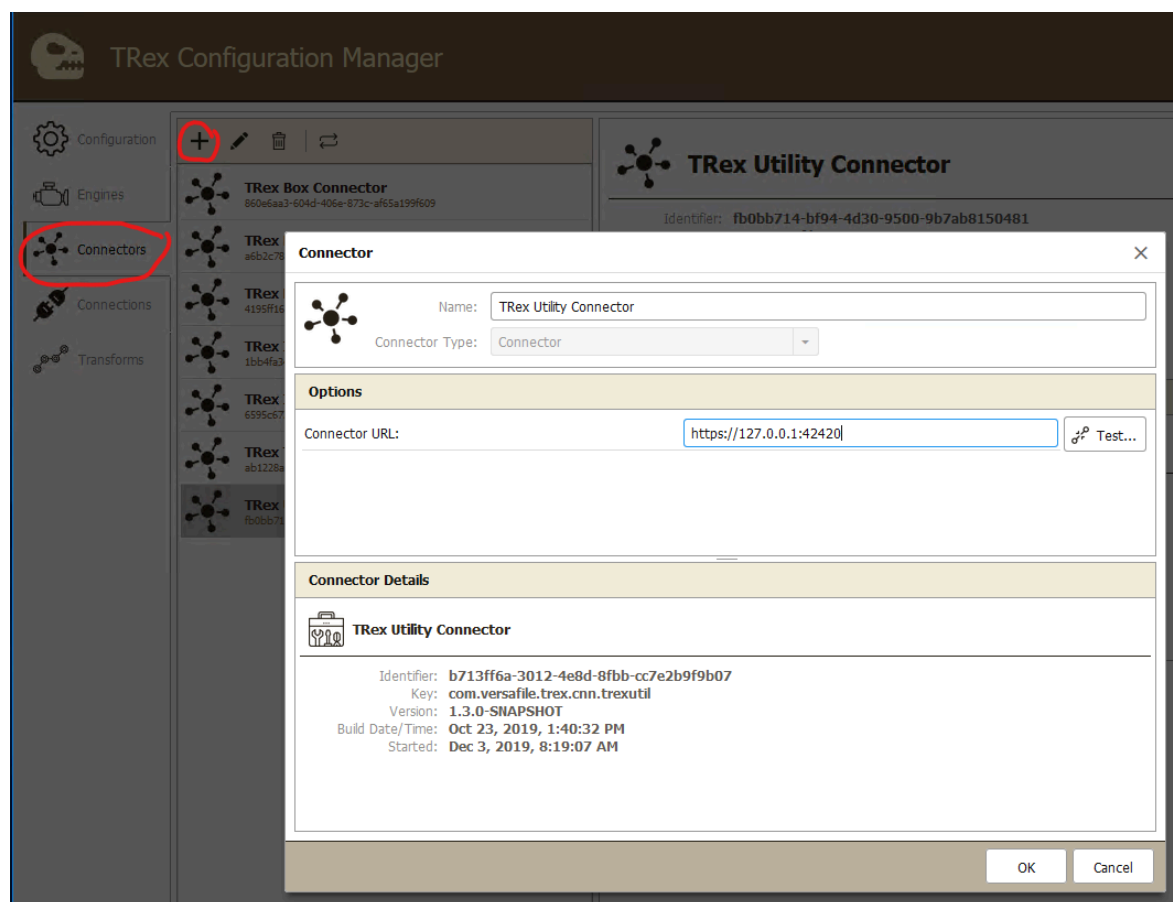
In default installations localhost or loopback addresses can be used. In advanced configurations where connectors are deployed on remote servers, the remote machine address/name should be used:

```
https://127.0.0.1:42420
```

```
https://localhost:42420
```

```
https://<remoteconnectorserver>:42420
```

Note: that 127.0.0.1 is a loopback ip4 address - the same as using localhost. This indicates the connector is running on the same machine as the TRex core services.



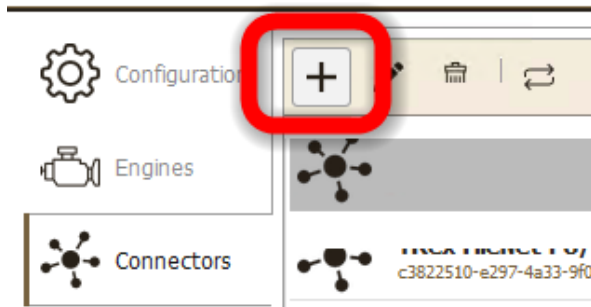
The default ports for the connectors assigned in the TRex configuration are listed in the table below:

Trex Utilities	42420
Filesystem	42421
IBM FileNet P8 Content Engine	42422
Apache Tika	42423
IBM Watson Cloud Services	42424
Elasticsearch	42425
IBM FileNet P8 Process Engine	42426
Box Cloud Content Management	42427
IBM Content Manager CM8	42429
IBM Forms Server	42430
Microsoft SharePoint	42431
Generic Web Services	42432

5.2.4.3.2.1. Box

To Create a new Box Connector in the docuflow Configuration UI:

1. Click on the 'Connectors' Tab
2. Click on the + icon
3. Enter the Connector URL (**<https://localhost:42427>**)



Adding a Connector

Connector Description

The Box connector provides connectivity to the Box Content Management, Workflow and Collaboration platform. See <http://www.box.com>

Box is a SaaS platform and all current releases of the **Enterprise** version are supported.

Methods Included:

File Operations

- Delete File
- Fetch File
- File Iterator
- Update File
- Upload File

Folder Operations

- Create Folder
- Delete Folder
- Folder Iterator
- Update Folder

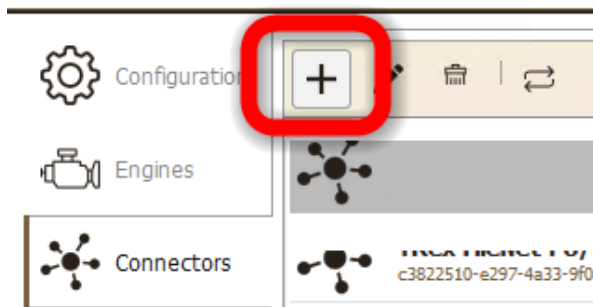
Search

- Metadata Query
- Search

5.2.4.3.2.2. Filesystem

To Create a new Filesystem Connector in the docuflow Configuration UI:

1. Click on the 'Connectors' Tab
2. Click on the + icon
3. Enter the Connector URL (**https://localhost:42421**)



Adding a Connector

Connector Description

The filesystem connector allows for connections to any file locations that are accessible from the server d4i is installed on.

The Filesystem connector takes an argument on its mount point to connect to. Only this location and folders/files below will be accessible.

This is configured in file `./config/application-cnn-fs.yaml`. By default, it will start looking for a folder called 'root' within the installation location of TRex. To change this, update the 'mount-point' variable.

```
plugin-args:
  mount-point: ./root
```

This can be changed to a local drive location such as C: or D: or D:\mystuff. Additionally, remote shares can be access using UNC paths. For example: \\servername\share

NOTE: If you are mounting a remote filesystem share, the Filesystem service should be configured to log in with an account that has access/priviledges to the remote share.

Methods Included:

CSV File Operations

- CSV Row Iterator
- CSV Write Row

File Conversion

- Convert to PDF

File Operations

- Copy File
- Delete File
- File Digest

- [File Iterator](#)
- [Move File](#)
- [Write Content To File](#)

Folder Operations

- [Delete Folder](#)
- [Folder Iterator](#)

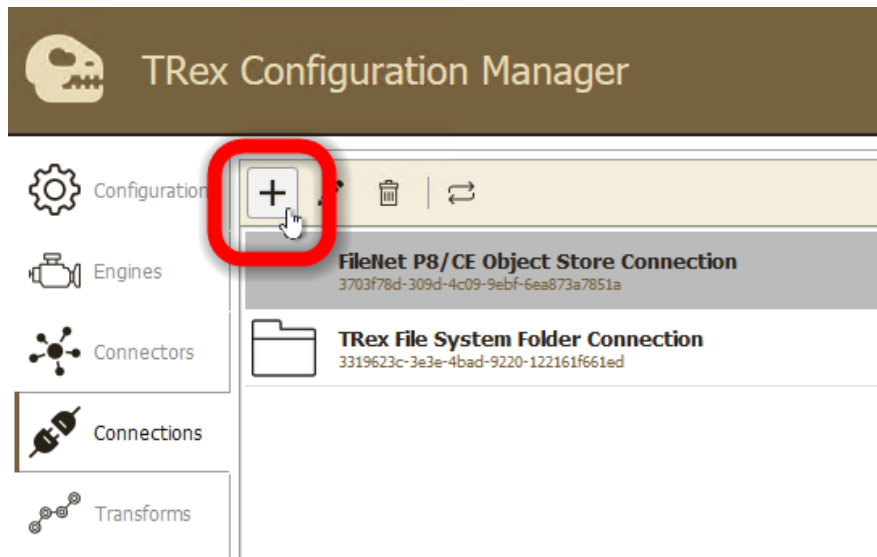
XML File Operations

- [XML Node Iterator](#)

5.2.4.3.3. Connections

Once Connectors have been created, the Connections can be created.

To create a new connection, select the '**Connections**' Tab and then the + (add...) button.



Each connection will require connection specific information and is detailed in the following sections.

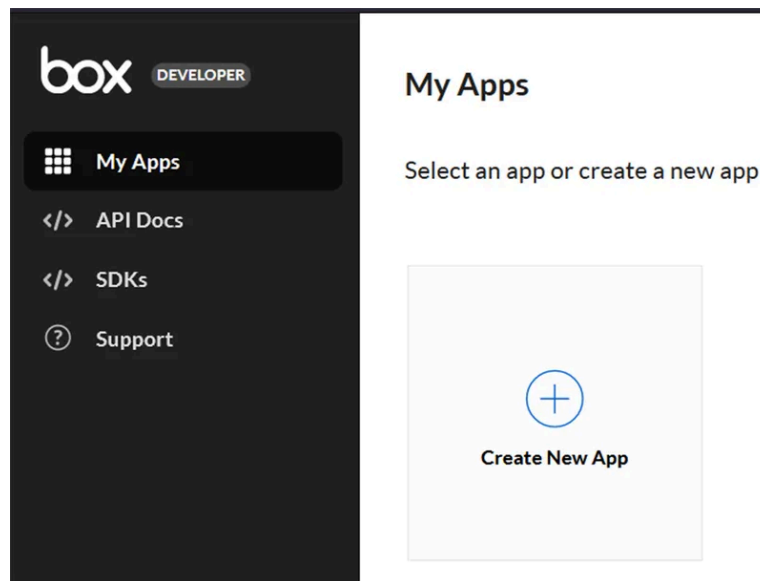
5.2.4.3.3.1. Box

Configuring a Box Connection

Go to your Box developer console:

- <https://app.box.com/developers/console>

Under MyApps, click on the 'Create New App' button



Under 'Lets Get Started' - Select **Custom App**

Under 'Authentication Method' - Select **Server Authentication (with JWT)** and click Next

Custom App

Select an authentication method and enter a name for your app. You can change the name later in General Settings.

Authentication Method

☒ **Server Authentication (with JWT)**

- Best for building integrations or apps with external collaborators - [Learn More](#)
- The app authenticates using a JSON Web Token. Requires a public/private key-pair for added security.

☐ **User Authentication (OAuth 2.0)**

- Best for building mobile or web apps - [Learn More](#)
- Users must authenticate with their Box login.

☐ **Server Authentication (Client Credentials Grant)**

- Best for scripting and back office integrations - [Learn More](#)
- The app authenticates using Client ID and Client Secret.

App Name

Enter an app name

By clicking 'Create App', you agree to the terms of the [Box Developer Agreement](#) and the [Box Privacy Policy](#).

Cancel

Create App

443 / 534

docuflow Guides

Modified: 07/02/2026

Name your application e.g. '**d4iConnector**' and click 'Create App'

Click 'View Your App' and navigate to the '**Configuration**' tab

Under **Application Scopes** select the following options:

- Read all files and folders stored in Box
- Write all files and folders stored in Box
- Manage users (allows App users to be created)
 - Note: if you want to script out and create app users without docuflow, this setting is not require
 - Also: This setting can be used for the application to create the app users, and once the process is complete, it can be removed

Under **Advanced Features** select the following options

- Make API calls using the as-user header (allows scaling across app users)
- Generate user access tokens

Under **Add and Manage Public Keys** select 'Generate a Public/Private Keypair' and Save the json file somewhere secure. This JSON will be required for the Box Connection configuration. ***This private key cannot be regenerated and must be stored safely and securely.***

Under **App Settings** select the 'Download JSON' - this JSON file is required for the Box connector

After completing these steps go to the 'App Authorization' and click 'Submit for Authorization'

Return to the Admin console, go to 'Apps' → Custom Apps

- Click 'Authorize New App' and then enter the Client ID from the previous step
- Click the Authorize button

Creating a d4i Box Connection

There are two types of connections supported for connecting to Box: Enterprise and AppUser. When creating a new Box connection this can be selected in the 'connection type' dropdown.

Box Enterprise Connection

The Enterprise connection uses the default service account for connecting to Box.

The following information is required when creating a new Enterprise connection:

- **Enterprise ID:** This can be found in the Box Admin under the **Account and Billing** section
- **Client ID:** This can be found in the **OAuth 2.0 Credentials** section of the Box Enterprise Application configured earlier
- **Client Secret:** This can be found in the **OAuth 2.0 Credentials** section of the Box Enterprise Application configured earlier
- **Public Key ID:** This can be found in the **Add and Manage Public Keys** section of the Box Enterprise Application configured earlier
- **Private Key:** This is the Private Key that corresponds to the Public key that was registered with the Box Enterprise Application. Must be entered properly formatted with multiple lines (not only one long line).
- **Passphrase:** This is the passphrase for the Key registered with the Box Enterprise Application.

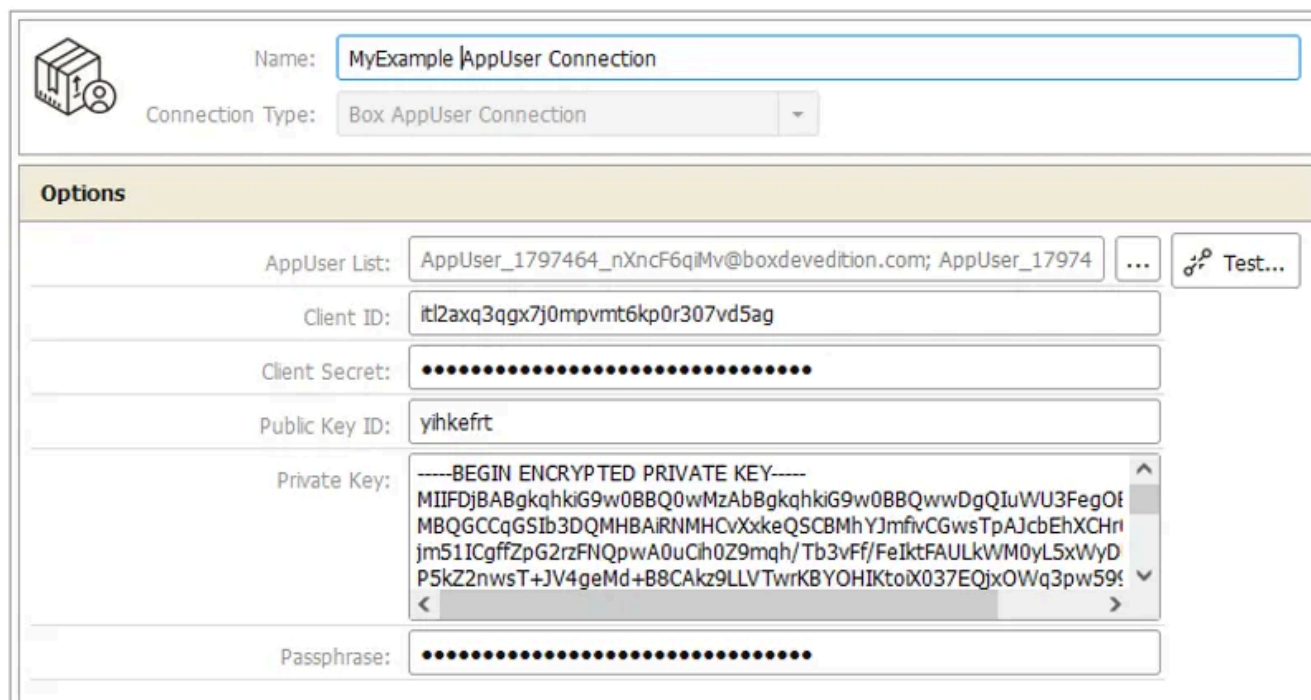
Box AppUser Connection

The AppUser connection uses a defined set of AppUsers associated with the Box Enterprise Application to connect to Box. The d4i Box connector will round robin between the provided App Users configured in the connector.

The following information is required when creating a new AppUser connection:

- **AppUser List:** A list of Box App users that have been previously created under the Box Enterprise Application
- **Enterprise ID:** This can be found in the Box Admin under the **Account and Billing** section
- **Client ID:** This can be found in the **OAuth 2.0 Credentials** section of the Box Enterprise Application configured earlier
- **Client Secret:** This can be found in the **OAuth 2.0 Credentials** section of the Box Enterprise Application configured earlier
- **Public Key ID:** This can be found in the **Add and Manage Public Keys** section of the Box Enterprise Application configured earlier
- **Private Key:** This is the Private Key that corresponds to the Public key that was registered with the Box Enterprise Application. Must be entered properly formatted with multiple lines (not only one long line).
- **Passphrase:** This is the passphrase for the Key registered with the Box Enterprise Application.

Once a connection is created, a connection test can be run to check if the configuration is valid.



Name: MyExample AppUser Connection

Connection Type: Box AppUser Connection

Options

AppUser List: AppUser_1797464_nXncF6qiMv@boxdevedition.com; AppUser_17974 ... **Test...**

Client ID: itl2axq3qgx7j0mpvmt6kp0r307vd5ag

Client Secret:

Public Key ID: yihkefrt

Private Key: -----BEGIN ENCRYPTED PRIVATE KEY-----
 MIIFDjBAGkqhkiG9w0BBQ0wMzAbBgkqhkiG9w0BBQwwDgQIuWU3FegOf
 MBQGCCqGS1b3DQMHBAIRNMHCvXxkeQSCBMhYJmfivCGwsTpAJcbEhXCHr
 jm51ICgffZpG2rzFNQpwA0uCiH0Z9mqh/Tb3vFf/FeIktFAULkWM0yL5xWyD
 P5kZ2nwsT+JV4geMd+B8CAkz9LLVTwrKBYOHIKtoX037EQjxOWq3pw59!
 < >

Passphrase:

5.2.4.3.3.2. Filesystem

The filesystem connection allows for the setting of a subfolder root that is relative to where the connector is running.

For example, if the connector service was connected to the root of D:\ then the value entered would be a subdirectory of D:\. Use the default ./ to allow access to all subdirectory locations.



Name:

Connection Type: TRex File System Folder Connection ▼

Options

Root Path:

 Test...

Output Entity

 **Folder Entity**

Attributes

Breadcrumb:	WTI5dExuWmxjbk5oWm1sc1pTNTBjbVY0TG10dWJpNW1jdz09fl ^
Folder Name:	root
Folder Path:	.\

5.3. SuccessFactors for FileNet

5.3.1. SuccessFactors Technical Overview



5.3.1.1. Introduction

docuflow for SuccessFactors and IBM FileNet integrates documents between SuccessFactors and the FileNet content platform.

This integration includes document storage, linking, and metadata transfer from SuccessFactors to FileNet. Documents can be captured from SuccessFactors and stored within FileNet, and documents added to FileNet can be linked back to SuccessFactors.

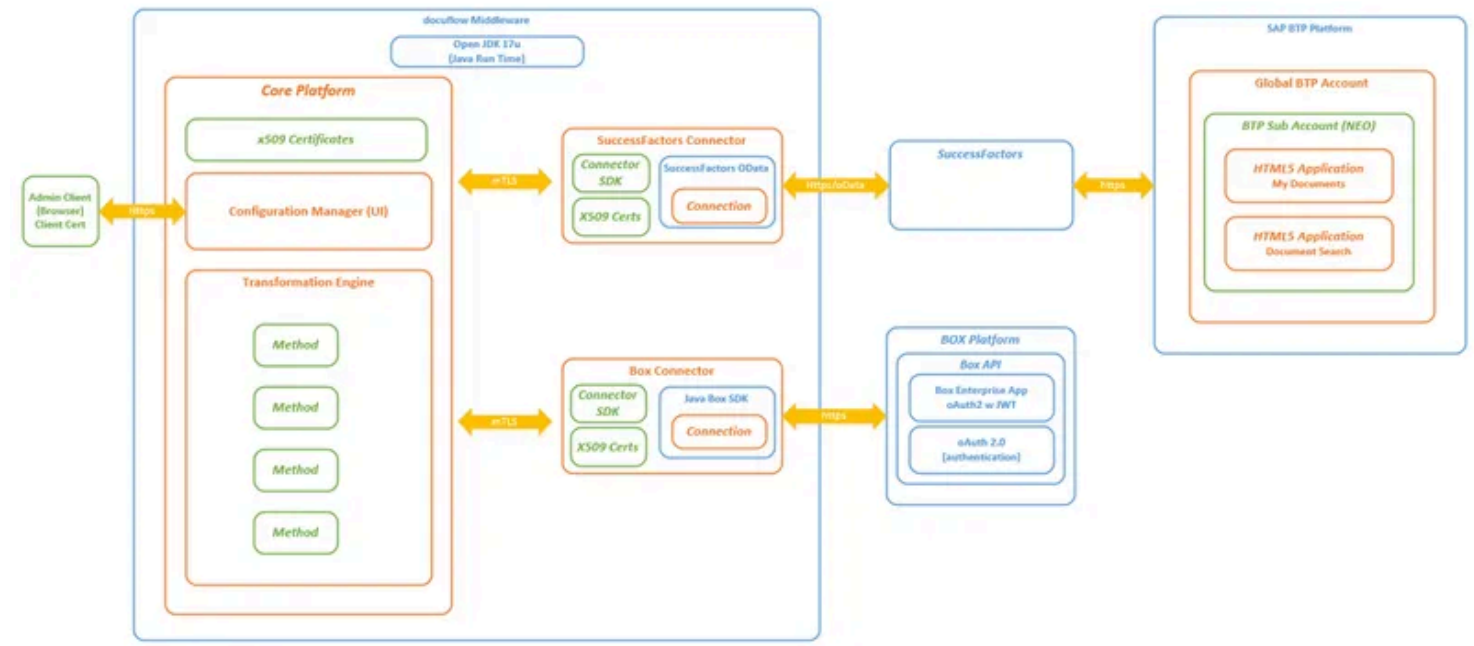
To enable users to access documents in FileNet, two HTML5 applications are available:

- My Documents – Shows a user all of their employee documents that have been linked from FileNet.
- Document Search – Allows HR Managers to search for documents belonging to employees who report to them.

The document capture and linking process is handled by the docuflow middleware, while My Documents and Document Search run as HTML5 applications deployed on the SAP BTP Platform.

5.3.1.2. Architecture Overview

The below diagram outlines the basic components of the solution



The docuflow Middleware is responsible for:

- Discovering new Employees and creating a Employee Folder in FileNet and linking this folder back to their Employee Profile
- Discovering new Employee Documents in SuccessFactors, adding them to FileNet and linking them back to SuccessFactors using LinkCards
- Discovering new Employee Documents in FileNet and linking them to SuccessFactors using LinkCards

The SAP BTP Platform is responsible for:

- A user interface (MyDocuments) for Employees to see the LinkCards that have been attached to their Employee Profile
- A user interface (DocumentSearch) for HR Managers to see the LinkCards for employees they are related to

5.3.1.3. Minimum Requirements

The docuflow middleware component (docuflow Enterprise Connector) can be deployed on-premise, in a private cloud or as a fully managed service from VersaFile.

When deploying on premise or in a private cloud environment, physical or virtual hardware is supported. The following server resources are recommended:

Operating Systems	CPU	Disk Space	RAM	Java Runtime Environment
Windows Server 2012, Windows Server 2012 R2, Windows Server 2016, Windows Server 2019	Minimum 4 cores or 4 physical CPUs with at least 2.0Ghz	40 GB	16 GB	OpenJDK 17 LTS or later

JCO requirements: 3.0.14 and above

(3.1.x and above requires Visual C++ Redistributable Packages for Visual Studio 2013)

<https://www.microsoft.com/en-ca/download/details.aspx?id=40784>

See <https://me.sap.com/notes/2786882>

URL and Firewall Configuration for IBM FileNet API Access

Access to the following URLs is required from the middleware server:

Base url: <https://api.box.com/2.0/>

Base upload url: <https://upload.box.com/api/2.0/>

Token url: <https://api.box.com/oauth2/token>

Firewall Information:

- Source: the IP of the middleware server
- Port: https is 443
- Protocol: https

This provides the basic functionality that docuflow requires for its processing. It is strongly recommended however that you review the Box link below to ensure you have all the appropriate access for your particular environment and usage requirements above and beyond docuflow:

<https://support.box.com/hc/en-us/articles/360043696434-Configuring-A-Firewall-For-Box-Applications>

Box also provides a connectivity test page as well as documented firewall configuration if necessary:

<https://support.box.com/hc/en-us/articles/360043694474-Box-Connectivity-Tests>

5.3.2. BTP Configuration

This section describes the typical configuration required to integrate the three HTML5 applications into SuccessFactors.

- My Documents
- Document Search
- Document Search (HRBP)

Required Accounts:

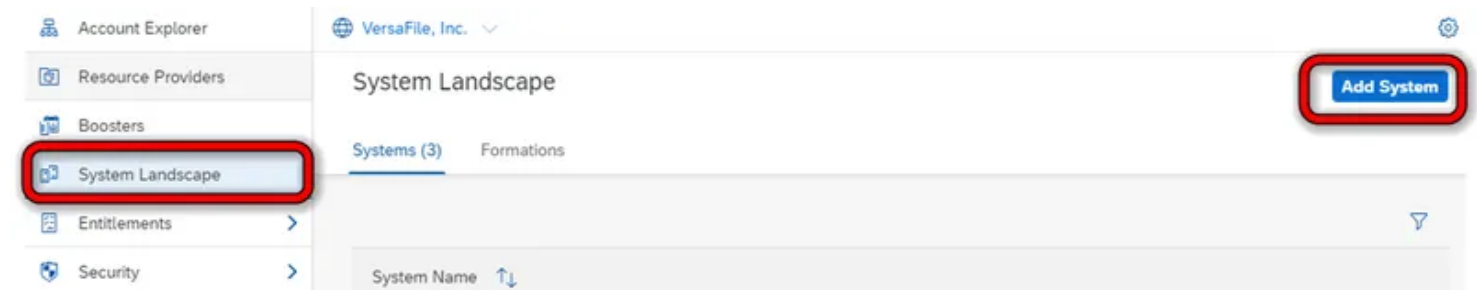
- Administrative BTP Access
- Administrative SuccessFactors Access

5.3.2.1. Linking SF to BTP Landscape

The following steps register the SuccessFactors instance to the BTP system Landscape. This is the first task in configuring the integration.

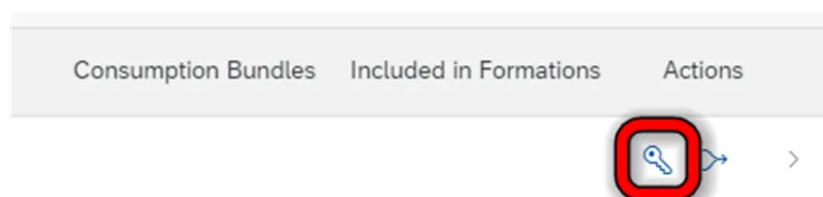
At the BTP Global Account, navigate to **System Landscape**.

Click the **'Add System'** button

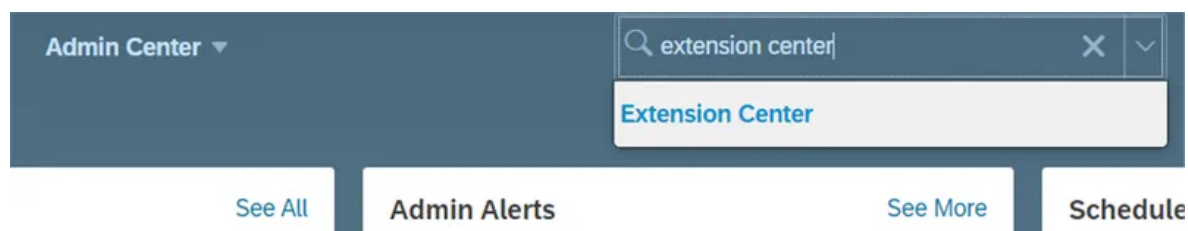


Supply a descriptive System Name and choose **'SAP SuccessFactors'** from the Type dropdown

Once the system has been added, click on the 'Get Token' icon. Copy this Token Value. This Token will be used within SuccessFactors to register the SuccessFactors system with the BTP Landscape.

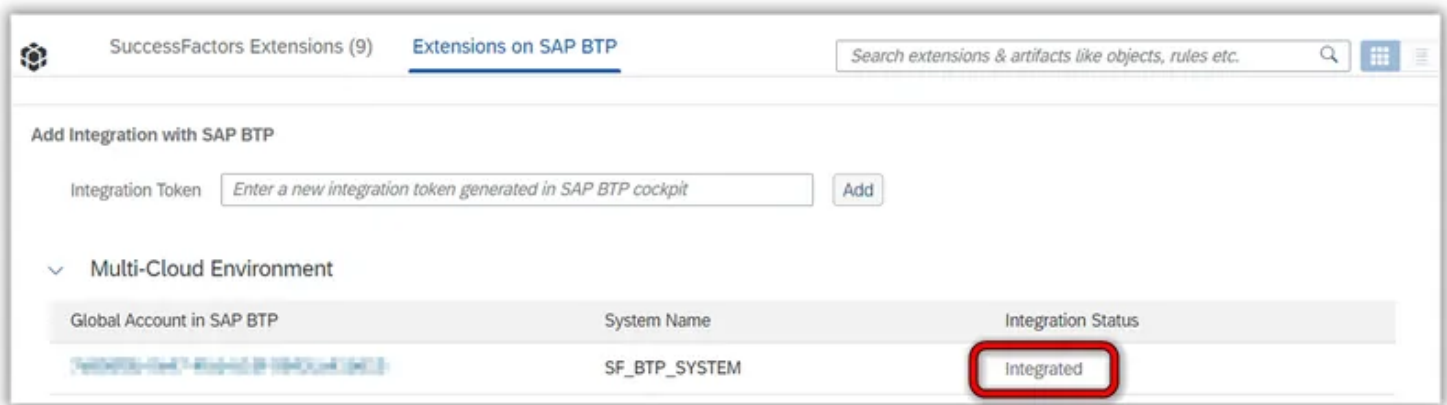


Next Log into **SuccessFactors** with an Administrative Account and Navigate to **'Extensions Center'**



Within the Extension Center navigate to the **'Extensions on SAP BTP'** tab. In the section **'Add Integration with SAP BTP'** paste the Token and click the **'Add'** button. This will connect the SuccessFactors instance with the BTP landscape.

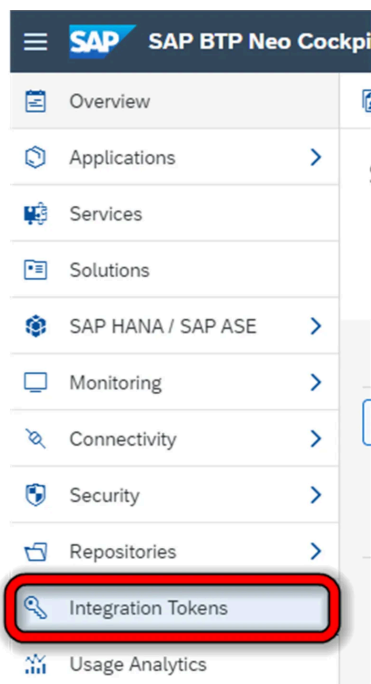
A message indicating the Systems are integrating will be shown. After a short period the status will show **'Integrated'**



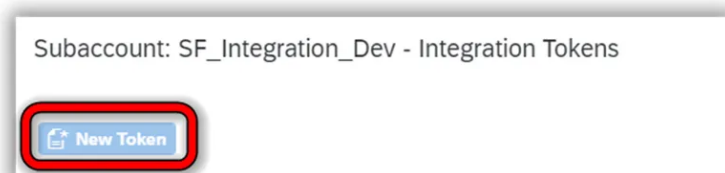
Next, link the BTP Sub Account to the Extensions on SAP BTP

- **NOTE:** The BTP Sub Account MUST be a NEO account (SAP Region)
- **NOTE:** See '**BTP Sub Account Creation**' sub-topic if you have not yet created a Sub Account

Open your BTP Sub Account and Navigate to '**Integration Tokens**'



Click on the '**New Token**' option to generate a new integration token for this BTP Sub Account. This token will then be used in SuccessFactors to integrate to the BTP Sub Account



Navigate back to '**Extentions Center**' in SuccessFactors

Once again, use the 'Add' Integration Token feature

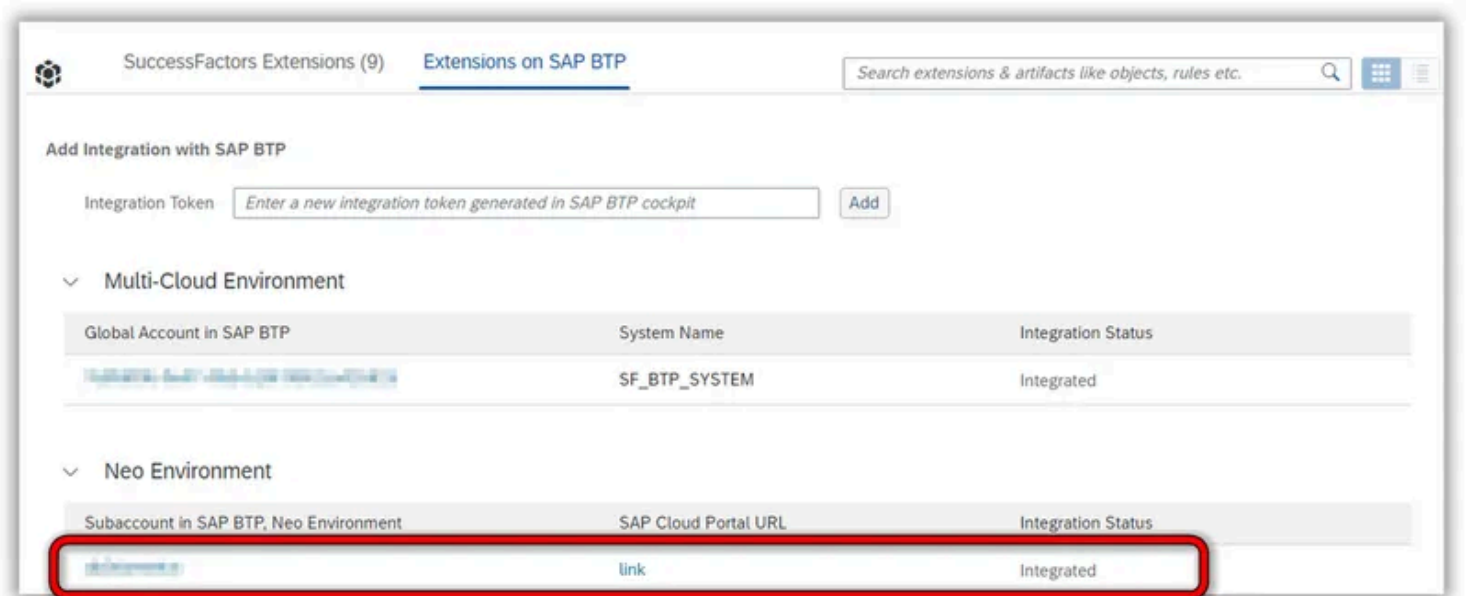


SuccessFactors Extensions (9) Extensions on SAP BTP Search extensions & artifacts like objects, rules etc.

Add Integration with SAP BTP

Integration Token Add

Once this token is added, it will show up in the **'Neo Environment'** Section



SuccessFactors Extensions (9) Extensions on SAP BTP Search extensions & artifacts like objects, rules etc.

Add Integration with SAP BTP

Integration Token Add

Multi-Cloud Environment

Global Account in SAP BTP	System Name	Integration Status
Subaccount in SAP BTP, Multi-Cloud Environment	SF_BTP_SYSTEM	Integrated

Neo Environment

Subaccount in SAP BTP, Neo Environment	SAP Cloud Portal URL	Integration Status
Subaccount in SAP BTP, Neo Environment	link	Integrated

A Trust has been created to BTP. To see this, Navigate to **BTP Cockpit → Security → Trust**

 **SAP BTP Neo Cockpit**

Overview

Applications

Services

Solutions

SAP HANA / SAP ASE

Monitoring

Connectivity

Security

Trust

Authorizations

OAuth

Repositories

Integration Tokens

Usage Analytics

Entitlements

Members

VersaFile, Inc. / SF_Integration_Dev

Subaccount: SF_Integration_Dev - Trust

Trust Management

Local Service Provider | Application Identity Provider | Custom Application Domains Settings | Basic Authentication

 **Manage Local Provider Settings**
for yukrury3mx

Configuration Type: Custom

Local Provider Name: *

Signing Key: *

Signing Certificate: *

Certificate Valid From: Wed Nov 09 22:24:42 UTC 2022

Certificate Valid To: Tue Nov 09 22:24:42 UTC 2032

Principal Propagation: Enabled

Force Authentication: Disabled

Edit

Get Metadata

455 / 534

docuflow Guides

Modified: 07/02/2026

5.3.2.1.1. BTP Sub Account Creation

This step only needs to be done if there is not already a sub account created for the BTP environment.

At the Global Account Level, navigate to **'Account Explorer'** and click the **'Create'** button and select **'Sub Account'**

NOTE: Select one of the SAP (NEO) Regions to run the integration

Create Subaccount

Display Name*

Subdomain ⓘ

Description

Region*

Parent*

Region options:

Singapore	cf-ap21
US East (VA)	cf-us21
US West (WA)	cf-us20
SAP	
Australia (Sydney)	neo-ap1
Brazil (São Paulo)	neo-br1
Canada (Toronto)	neo-ca1
China (Shanghai)	neo-cn1
Europe (Amsterdam)	neo-eu3
Europe (Frankfurt)	neo-eu2
Japan (Tokyo)	neo-jp1
KSA (Riyadh)	neo-sa1
UAE (Dubai)	neo-ae1
US West (Colorado Springs)	neo-us4

Account structure.

not global account or to an account.

Create **Cancel**

5.3.2.2. HTML5 Application Registrations

This section will demonstrate how install the HTML5 applications into the BTP Sub Account that has been integrated into the SuccessFactors system in the previous section.

SAP Web IDE Full-Stack Service

If not already enabled, SAP Web IDE Full-Stack Service must be enabled in the sub account.

To enable this, in the sub account navigate to 'Services' and then select the tile/option 'SAP Web IDE Full-Stack' and then click '**Enable**'.

Security Permissions

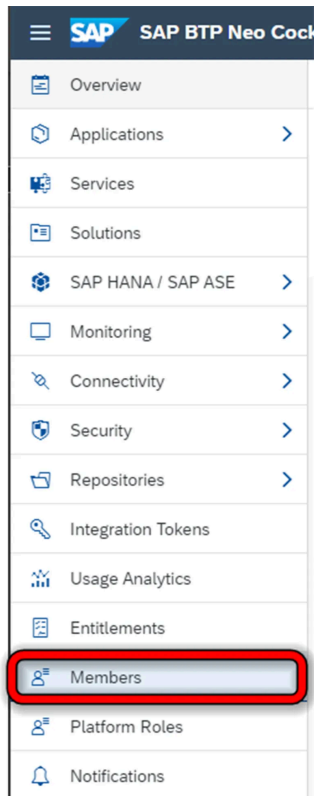
NOTE: This section will require the BTP Command Line Interface (CLI). If you do not have this installed please see the sub section '**BTP Command Line Interface**'

First, Permissions must be given to allow configuration of the BTP Sub Account.and Web Applications.

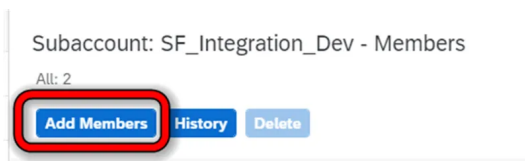
To Enable security for the Web Applications, navigate to **Security→Authorizations**

- Enter the S User ID and click 'Show Assignments'
- Click 'Assign' and then select
 - Subaccount 'SAP Web IDE xxx'
 - Application: di
 - Role: DiAdministrator
- Click 'Assign' and then select:
 - Subaccount 'SAP Web IDE xxx'
 - Application: featurediscovery
 - Role: DevXAdministrator

Next, Navigate to the '**Members**' section of the BTP Sub Account



In the Members section, click **'Add Members'**

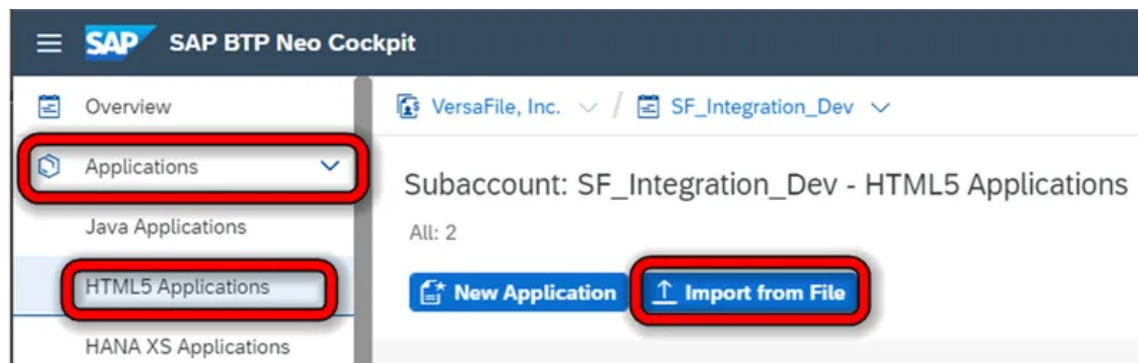


Enter the **User id** and select the permissions you want this user account to have the BTP Sub Account.

 A screenshot of the 'Add Members' dialog box. It contains a text input field for 'User IDs' with the placeholder text 'Sxxxxxxxxx'. To the right, under 'Assign roles:', there are five checkboxes, all of which are checked: 'Administrator (predefined role)', 'Developer (predefined role)', 'Support User (predefined role)', 'Application User Admin (predefined role)', and 'Cloud Connector Admin (predefined role)'. At the bottom, there is a text input field for 'Comment (optional, visible to all members):'. The dialog has 'Add Members' and 'Cancel' buttons at the bottom right.

Next, upload the HTML5 Applications to the BTP Sub Account. These applications are supplied in a .zip format.

Navigate to **Applications→HTML5 Applications** in the BTP Sub Account and click **'Import from File'**



In the dialog, select the '**sfmydocuments.zip**', enter the version number 1 and click 'Import'

Follow the same procedure as above for '**sfdocumentsearch.zip**'.

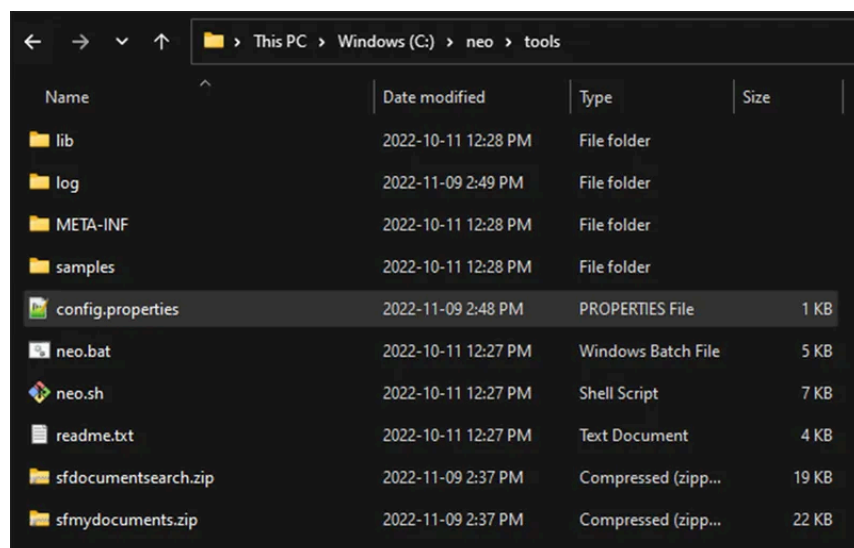
Once completed the applications should be listed in the HTML5 Applications screen.

Enable Application Access

NOTE: This next step required the BTP Command Line Interface (CLI). If you do not have this toolset installed, see the sub topic '**BTP Command Line Interface (CLI)**'

NOTE: The following examples are shown on windows but can be adapted for any other command line/shell

Open a Command Prompt/Shell and navigate to the '**Tools**' directory of the BTP Tools installation



Setup a properties file (e.g. **config.properties**) with the information that will be required for the connection. This will make the process easier.

NOTE: For a list of BTP Host values/locations: <https://help.sap.com/docs/btp/sap-btp-neo-environment/regions-and-hosts-available-for-neo-environment>

```
#####
# General settings - relevant for all commands #
#####

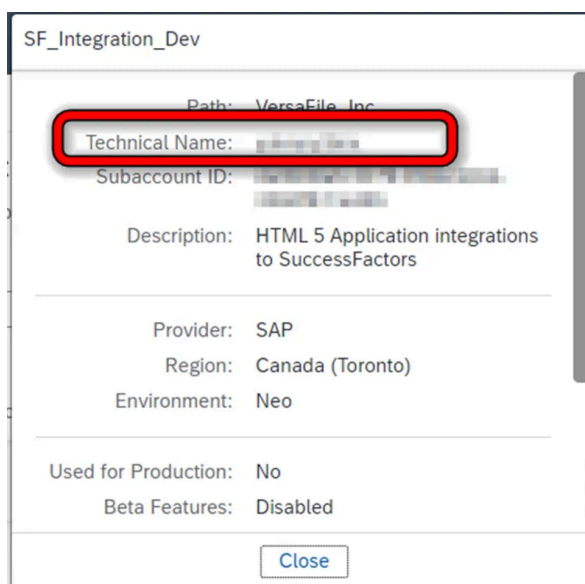
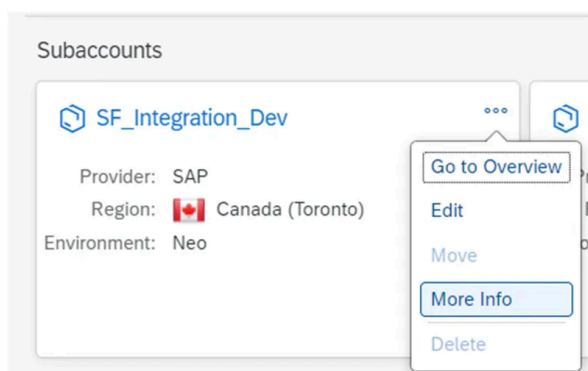
# Your account name
account=sfdocumentsearch

# Application name
application=sfdocumentsearch
application-type=html5

# User for login to hana.ondemand.com.
user=sapuser@hana.ondemand.com

# URL of the landscape admin server. Optional. Defaults to
hana.ondemand.com.
host=hana.ondemand.com
```

account = the ID of your BTP Sub Account



application = name of the application be registered. This process will be done twice, once for sfdocumentsearch and once for sfmydocuments. Note: use one of these values, run the command below, then alter this value and run the command again.

user = your SAP User Account. If you have a SAP Universal ID you will want to specify your S-ID, otherwise you can use the email associated with your S-ID

With the Command Prompt BTP CLI tools, execute the following command (use your properties file name)

neo.bat hcmcloud-enable-application-access config.properties

NOTE: if you are not using the properites file method, the full command would be:

```
neo hcmcloud-enable-application-access --application <my_application> --application-type java --account
<my_extension_subaccount> --user <user id> --host us1.hana.ondemand.com
```

Next, modify the config.properties file so that **application=sfmydocuments** to allow registration of the **sfmydocuments** application

```
#####
# General settings - relevant for all commands #
#####

# Your account name
account=sfmydocuments

# Application name
application=sfmydocuments
application-type=html5

# User for login to hana.ondemand.com.
user=sfmydocuments@hana.ondemand.com

# URL of the landscape admin server. Optional. Defaults to
hana.ondemand.com.
host=hana.ondemand.com
```

With the Command Prompt BTP CLI tools, execute the following command (use your properties file name)

neo.bat hcmcloud-enable-application-access config.properties

After registration example:

```

host      : https://ca1.hana.ondemand.com
account   : y[REDACTED]
application : sfmydocuments
application-type: html5
SDK version : 4.36.3
user      : [REDACTED]

Password for your user:
Enabling application access.....

Application access was enabled for application 'sfmydocuments' from subaccount 'y[REDACTED]':
Application Main URL      : https://sfmydocuments-[REDACTED].dispatcher.ca1.hana.ondemand.com
Service Provider Audience URL : https://ca1.hana.ondemand.com/[REDACTED]
Service Provider Logout URL  : https://authn.ca1.hana.ondemand.com/saml2/sp/slo/[REDACTED]

C:\Users\kyleh\Documents\SAP BTP\SAP NEO 2\tools>neo hcmcloud-enable-application-access vf2.properties

Neo Console Client

Executing [hcmcloud-enable-application-access] command with arguments:
host      : https://ca1.hana.ondemand.com
account   : [REDACTED]
application : sfdocumentsearch
application-type: html5
SDK version : 4.36.3
user      : S0024416632

Password for your user:
Enabling application access.....

Application access was enabled for application 'sfdocumentsearch' from subaccount '[REDACTED]':
Application Main URL      : https://sfdocumentsearch-[REDACTED].dispatcher.ca1.hana.ondemand.com
Service Provider Audience URL : https://ca1.hana.ondemand.com/[REDACTED]
Service Provider Logout URL  : https://authn.ca1.hana.ondemand.com/saml2/sp/slo/[REDACTED]

```

Register oData Connection

Finally, using the CLI register an oData connection (only one required for both applications)

Edit the **config.properties** file and comment out the application and:

Edit out the application application type properties:

```
#application=sfdocumentsearch
```

```
#application-type=html5
```

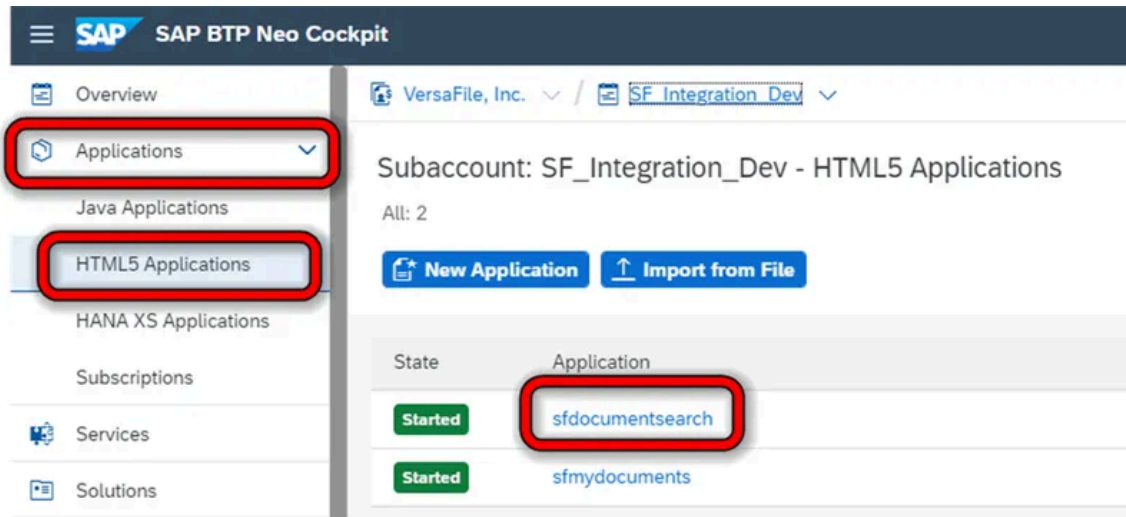
Add a new value for the name of the connection:

```
name=sfapps_odata
```

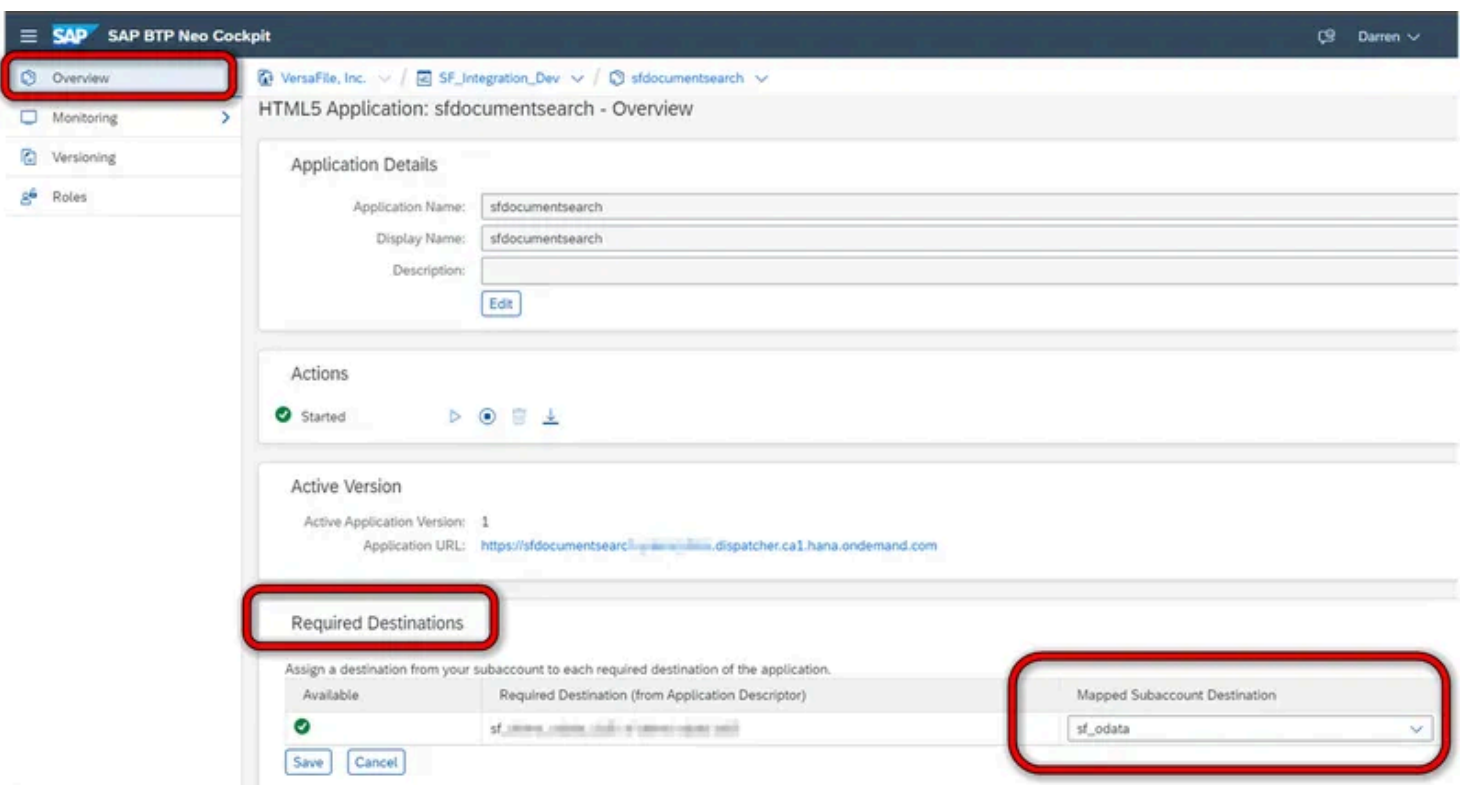
Neo.bat *hcmcloud-create-connection config.properties*

Set oData Connection in HTML5 Applications

For the next step, return to the **BTP Cockpit** and navigate to the **BTP Sub Account → Applications → HTML5 Applications** and click the sfdocumentsearch application



Under 'Required Destinations' set the 'Mapped Subaccount Destination'



Next, Click the 'Versioning' section, under 'History' section select the 'Versions' and set the 'Mapped Subaccount Destination'

SAP BTP Neo Cockpit

Overview Monitoring **Versioning** Roles

VersaFile, Inc. / SF_Integration_Dev / sfdocumentsearch

HTML5 Application: sfdocumentsearch - Versioning

Source Location

Git Repository URL: <https://git.ca1.hana.ondemand.com/> /sfdocumentsearch

[Edit Online](#)

History

[Commits](#) **[Versions](#)**

[Refresh](#) [Import from File](#)

Version	Active	Actions
1	✓	Edit Download

Required Destinations for Version "1"

Assign a destination from your subaccount to each required destination of the application.

Available	Required Destination (from Application Descriptor)	Mapped Subaccount Destination
✓	sf_odata	sf_odata

[Save](#) [Cancel](#)

Next, register the second HTML5 application sfmydocuments. Return to the **BTP Cockpit** and navigate to the **BTP Sub Account** → **Applications** → **HTML5 Applications** and click the **sfmydocuments** application

SAP BTP Neo Cockpit

Overview **Applications** Java Applications HANA XS Applications Subscriptions Services Solutions

VersaFile, Inc. / SF_Integration_Dev

Subaccount: SF_Integration_Dev - HTML5 Applications

All: 2

[New Application](#) [Import from File](#)

State	Application
Started	sfdocumentsearch
Started	sfmydocuments

Under '**Required Destinations**' set the '**Mapped Subaccount Destination**'

SAP BTP Neo Cockpit

Overview | Monitoring | Versioning | Roles

VersaFile, Inc. / SF_Integration_Dev / sfmydocuments

HTML5 Application: sfmydocuments - Overview

Application Details

Application Name: sfmydocuments
 Display Name: sfmydocuments
 Description:
 Edit

Actions

Started

Active Version

Active Application Version: 1
 Application URL: https://sfmydocuments.dispatcher.ca1.hana.ondemand.com

Required Destinations

Assign a destination from your subaccount to each required destination of the application.

Available	Required Destination (from Application Descriptor)	Mapped Subaccount Destination
✓	sf_odata	sf_odata

Edit

Next, Click the **'Versioning'** section, under **'History'** section select the **'Versions'** and set the **'Mapped Subaccount Destination'**

SAP BTP Neo Cockpit

Overview | Monitoring | Versioning | Roles

VersaFile, Inc. / SF_Integration_Dev / sfmydocuments

HTML5 Application: sfmydocuments - Versioning

Source Location

Git Repository URL: https://git.ca1.hana.ondemand.com/sfmydocuments
 Edit Online

History

Comm | **Versions**

Refresh | Import from File

Version	Active	Actions
1	✓	✕ ⬇

Required Destinations for Version "1"

Assign a destination from your subaccount to each required destination of the application.

Available	Required Destination (from Application Descriptor)	Mapped Subaccount Destination
✓	sf_odata	sf_odata

Edit

5.3.2.2.1. BTP Command Line Interface (CLI)

This section has information on how to install the BTP CLI if it does not already exist in your environment.

The CLI can be downloaded from: <https://tools.hana.ondemand.com/#cloud>

SAP BTP SDK for the Neo environment

The SAP BTP SDK for the Neo environment includes platform API JARs, a local runtime for testing, documentation, samples, and the Neo console client. These are the major versions of the SAP BTP SDK for the Neo environment:

- SAP BTP SDK for the Neo environment, version for Java Web Tomcat 8 - provides support for some of the standard Java EE APIs (Servlet, JSTL, EL)
- SAP BTP SDK for the Neo environment, version for Java Web Tomcat 9 - provides support for some of the standard Java EE APIs (Servlet, JSTL, EL)
- SAP BTP SDK for the Neo environment, version for Java EE 7 Web Profile TomEE 7 - provides support for Java EE 7 Web Profile APIs

Available SAP BTP SDKs for the Neo environment

SAP BTP SDK for the Neo environment	Comment	Version	File Size	Download	Show old versions
Java Web Tomcat 8		3.166.7	56.8 MB	neo-java-web-sdk-3.166.7.zip (sha1)	
Java Web Tomcat 9		4.39.6	59.1 MB	neo-java-web-sdk-4.39.6.zip (sha1)	
Java EE 7 Web Profile TomEE 7		1.137.8	93.9 MB	neo-javaee7-wp-sdk-1.137.8.zip (sha1)	

NOTE: The above tools will require a Java Runtime Environment (JRE). If you do not have one scroll down on the page to the SAP JVM section and download a version appropriate for your Operating System

SAP JVM

The SAP JVM is a prerequisite for local profiling with the SAP JVM Profiler. It is a standard compliant certified JDK, supplemented by additional supportability and developer features and extensive monitoring and tracing facilities. For more information, see the [SAP JVM documentation](#)

Available SAP JVMs

Operating System*	Architecture	Version	File Size	Download
Linux	x86_64	8.1.091	130.0 MB	sapjvm-8.1.091-linux-x64.zip (sha1)
Linux	x86_64	8.1.091	126.0 MB	sapjvm-8.1.091-linux-x64.rpm (sha1)
Linux	ppc64le	8.1.091	126.0 MB	sapjvm-8.1.091-linux-ppc64le.zip (sha1)
Linux	ppc64le	8.1.091	119.0 MB	sapjvm-8.1.091-linux-ppc64le.rpm (sha1)
Mac OS X	x86_64	8.1.091	131.0 MB	sapjvm-8.1.091-macosx-x64.zip (sha1)
Windows	x86_64	8.1.091	162.0 MB	sapjvm-8.1.091-windows-x64.zip (sha1)

*SAP JVM is supported for: SUSE Linux Enterprise Server 12 and 15, Redhat Enterprise Linux 7 and 8, Oracle Linux 7 and 8, Windows 10, Windows Server 2012, 2012 R2, 2016, 2019 and 2022, Mac OS X 10.14 (Mojave), 10.15 (Catalina), 11 (Big Sur) and 12 (Monterey)

Download the above tools and unpack to a system directory. Install the JRE if required.

If you just installed a JRE, the tools require 'JAVA_HOME' be defined. This will be the path where the JRE was downloaded to.

set JAVA_HOME=C:*your directory path* \JVM\sapjvm_8\jre

5.3.2.3. Extension Admin Permissions

The extension administrators will require permissions to the BTP environments to complete the Application setup in the SuccessFactors Extensions Center portlet.

- In the SuccessFactors Admin Center, navigate to the **Permissions Role Detail**
- Select the Administrative Role that is doing the solution configuration
- Navigate to the Permission setting '**Permission not requiring target**'
- Select '**Manage Extensions on SAP BTP**'
 - Allow '**Create Integration with SAP BTP**'

Add the Administrative user who will be accessing the Portal Configuration to the Extensions Administrators Group. In SuccessFactors:

- Navigate to '**Manage Permission Groups**'
- Select the group name: **Extension Administrators**
- Add the selected UserIds to the role

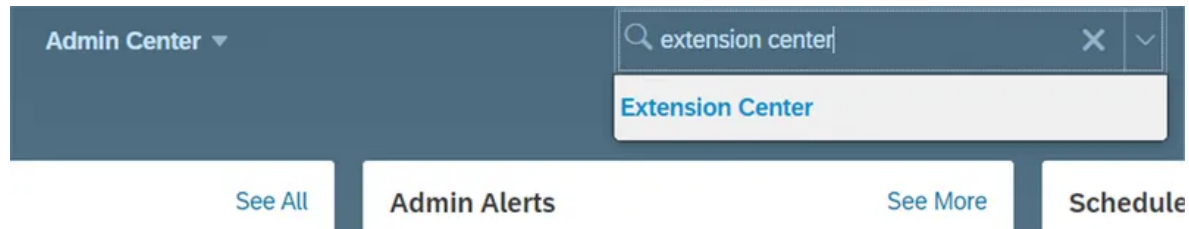
SAP Tech Note:

<https://help.sap.com/docs/btp/sap-business-technology-platform/register-sap-successfactors-system-in-global-account-in-sap-btp>

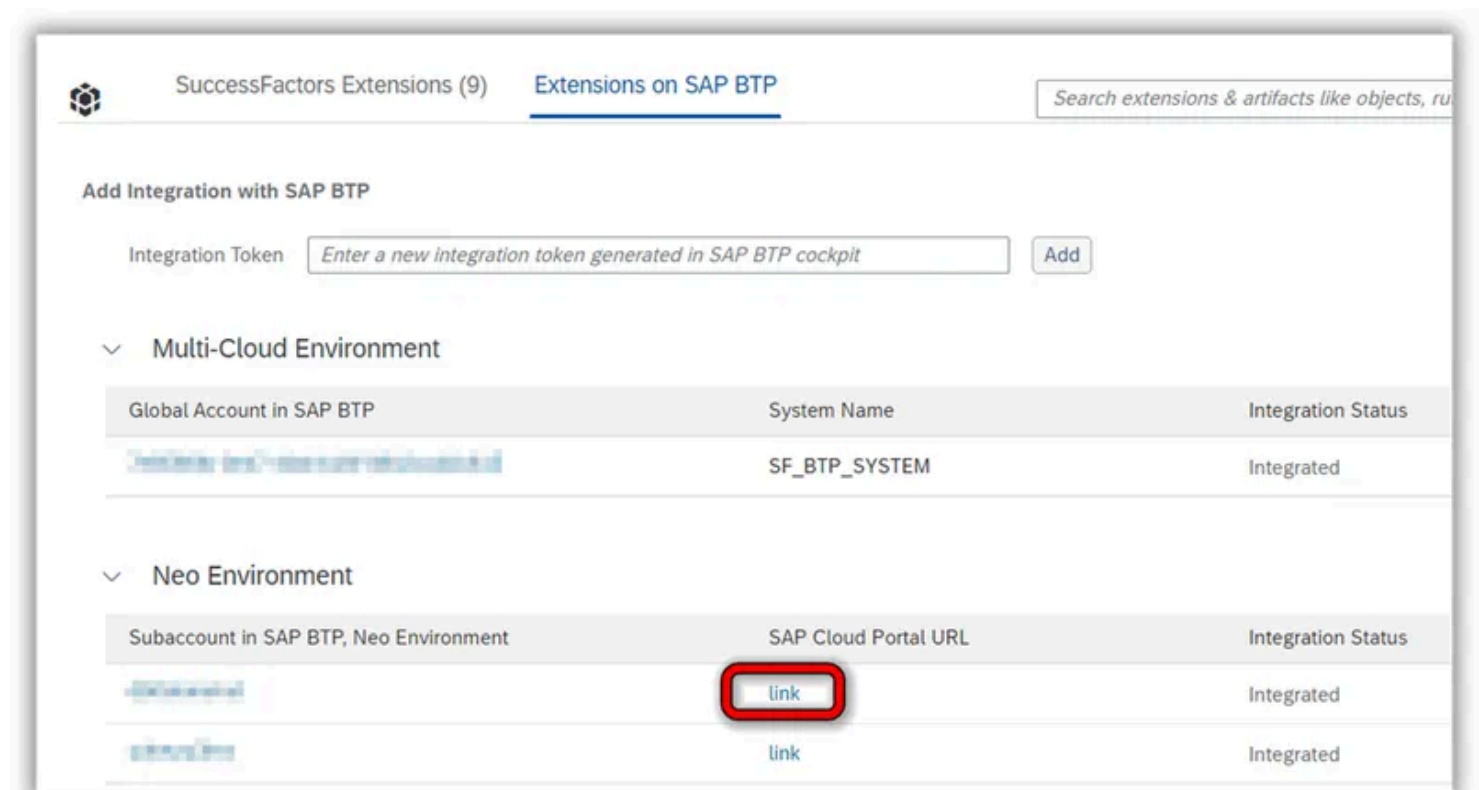
5.3.2.4. SuccessFactors App Setup

In this section, the HTML5 applications will be configured in SuccessFactors.

Log into SuccessFactors with an Administrative account and navigate to the **Extension Center**



Navigate to the '**Extensions on SAP BTP**' tab and click on the '**link**' under '**SAP Cloud Portal URL**'



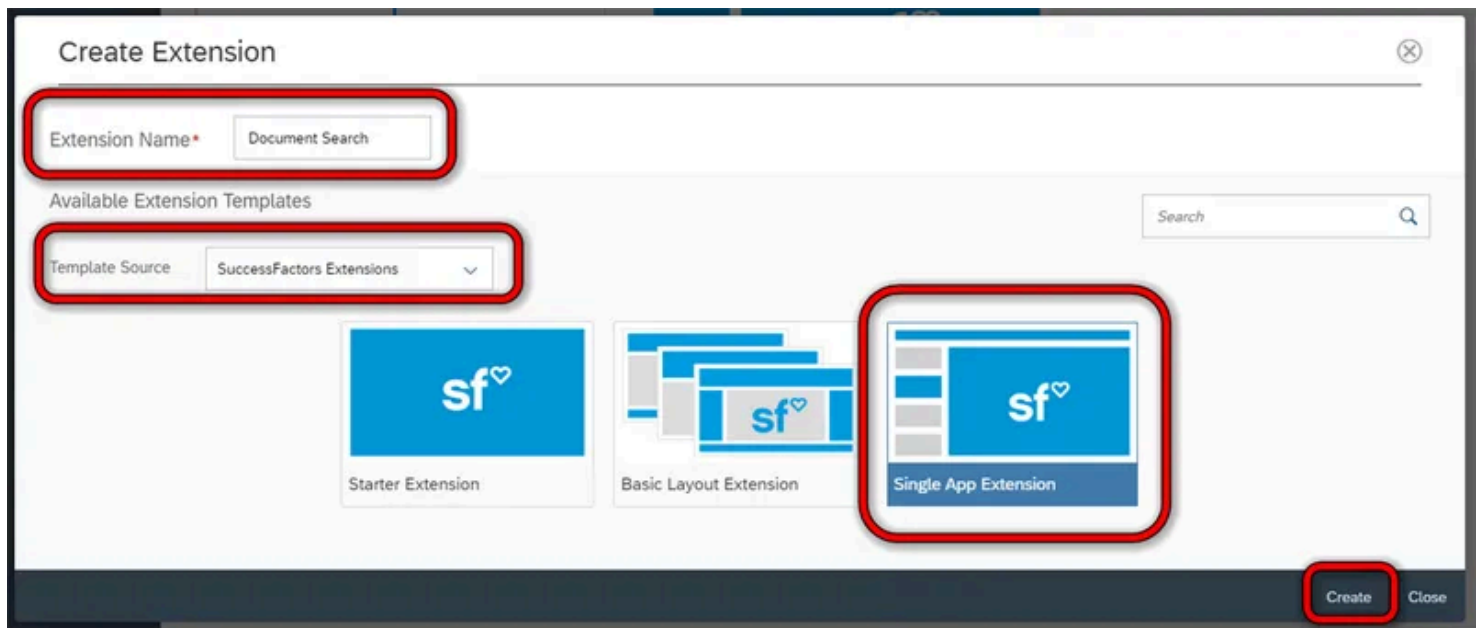
The SuccessFactors **Extension Directory** will open. Click on the giant plus button.



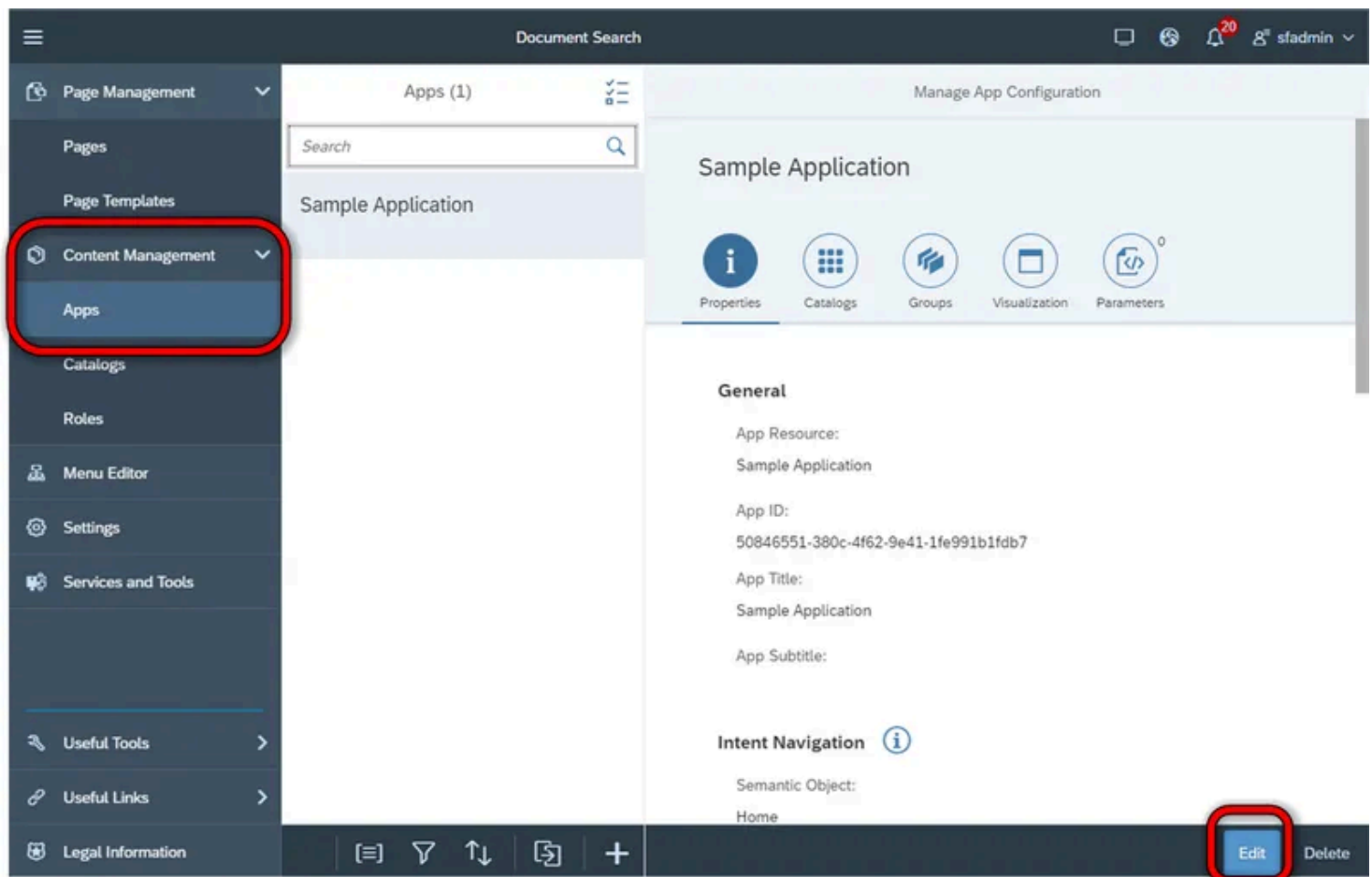
Set the following values for the new Extension and click the '**Create**' button

- Extension Name: **Document Search**

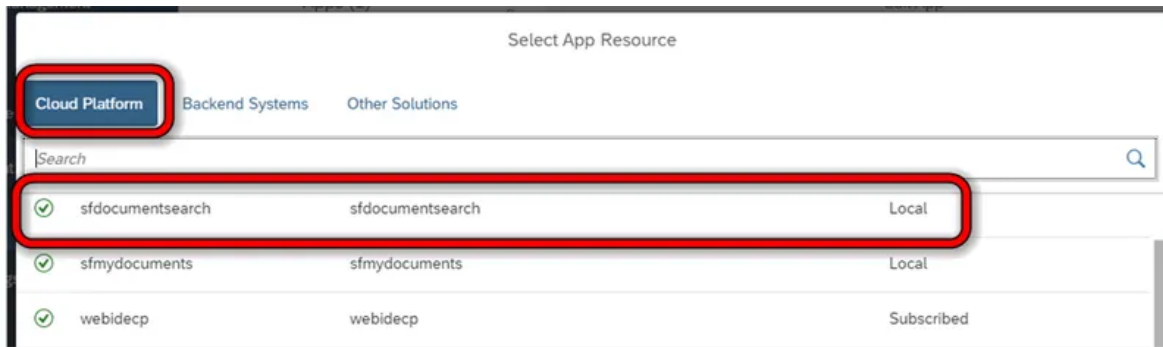
- Template Source: **SuccessFactors Extensions**
- Choose **Single App Extension**



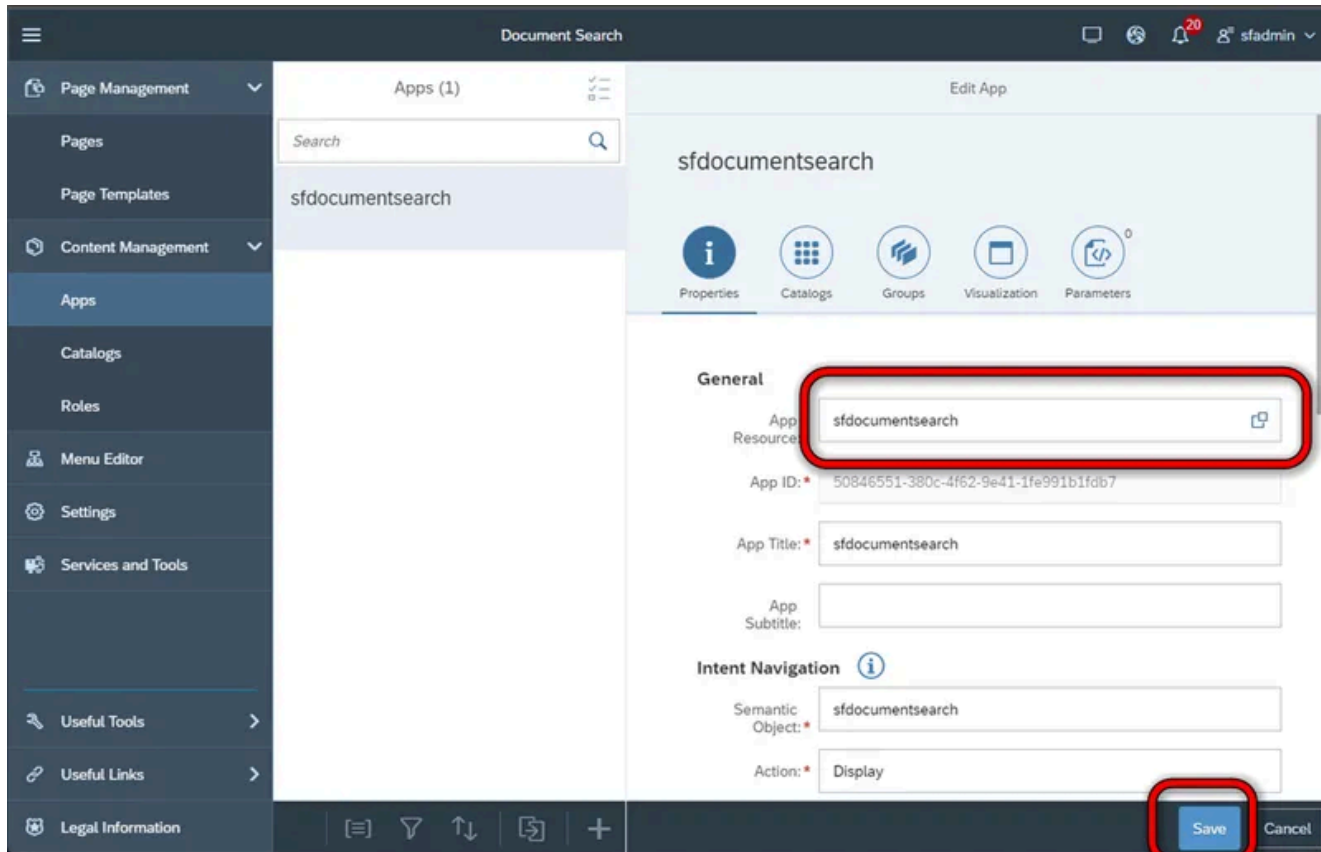
Once the Extension has been created, navigate to **Content Management** → **Apps** and click the **Edit** button



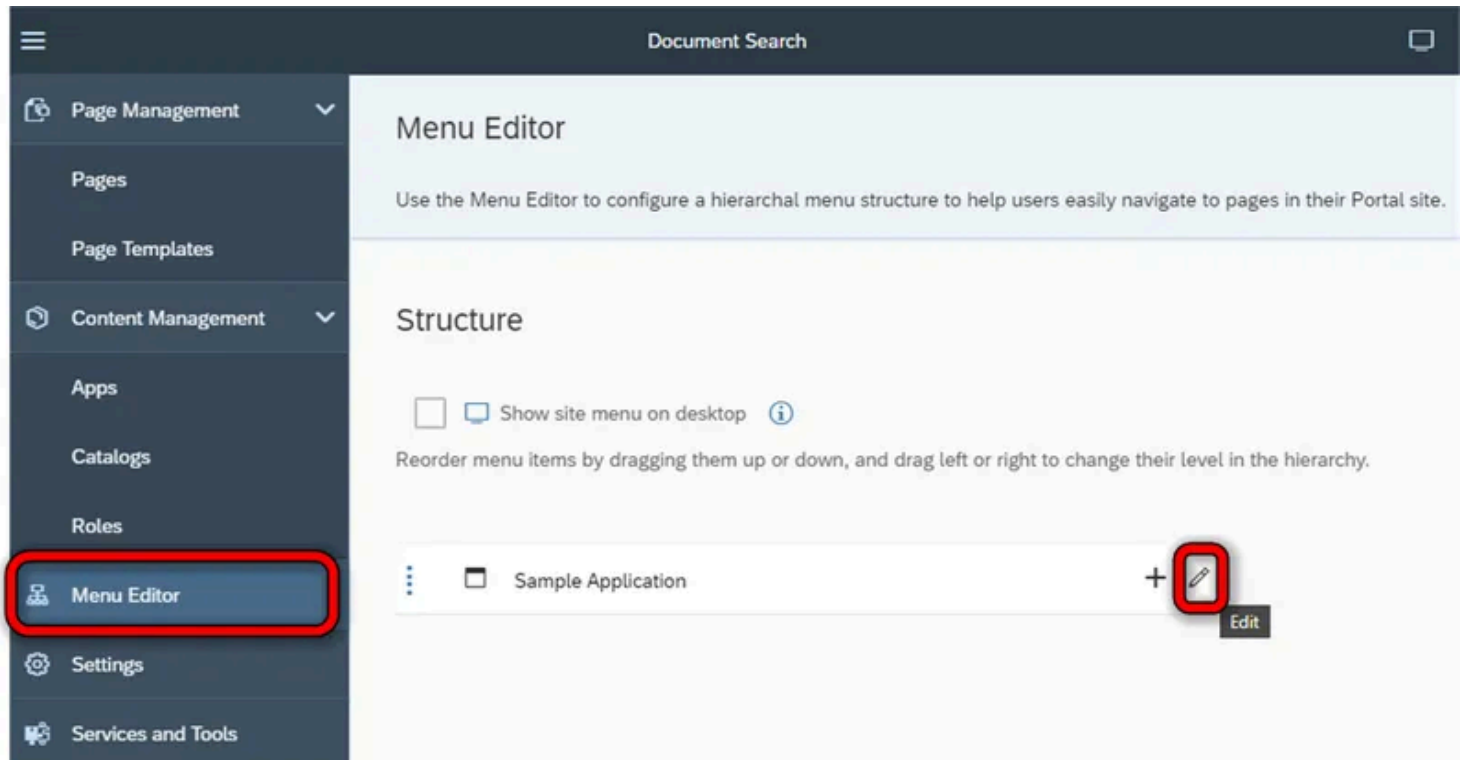
Choose the registered application from the BTP Platform (sfdocumentsearch)



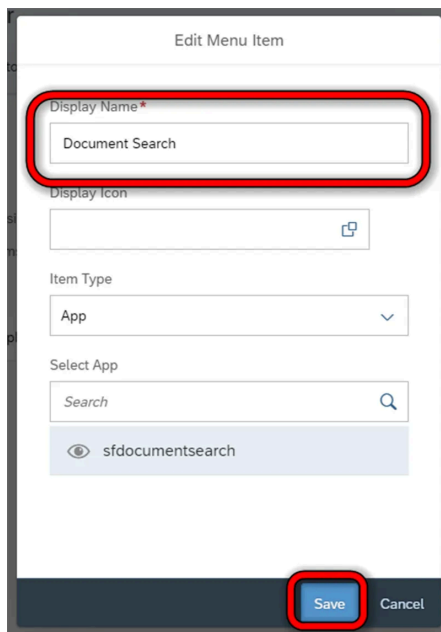
After selection the BTP Application click **SAVE**



Next, navigate to **'Menu Editor'** and click the **edit** pencil icon

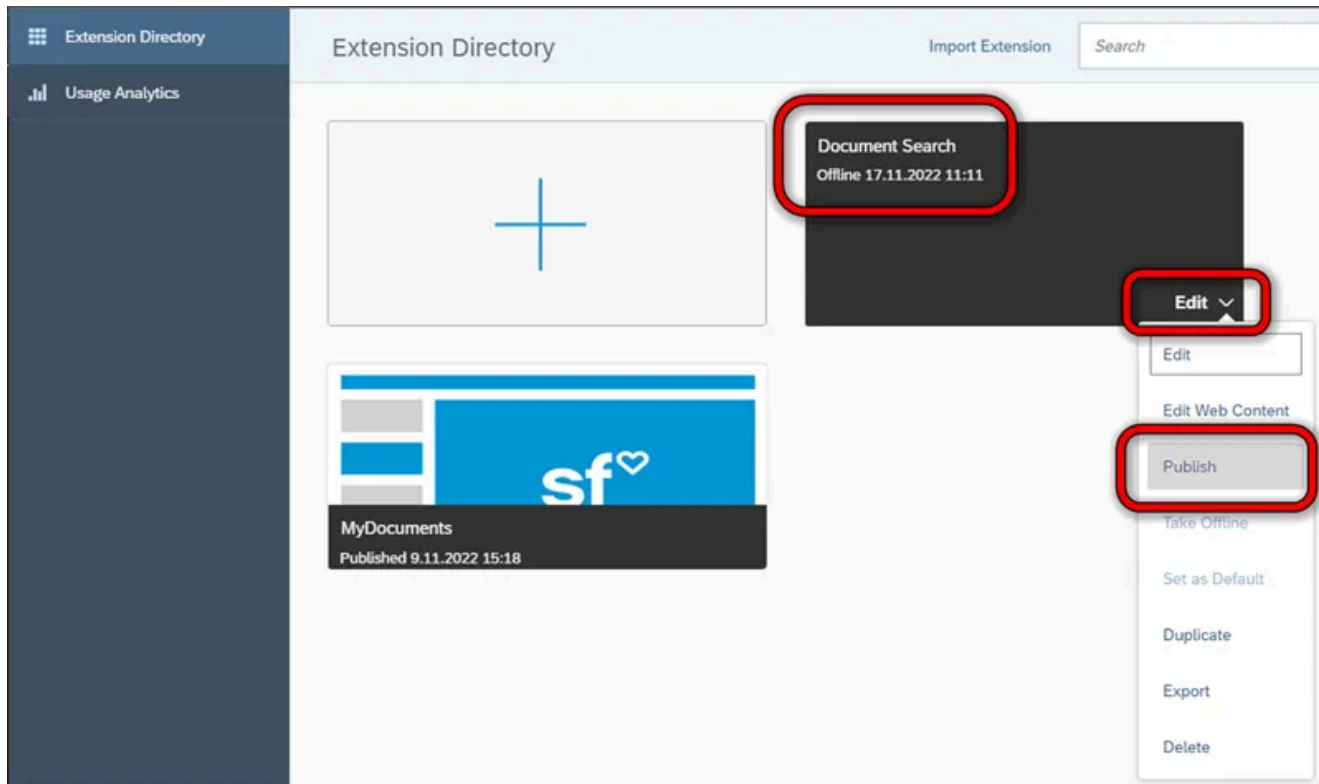


Set the display name to 'Document Search' and click the 'Save' button



After updating the menu display name, click the '**Save**' button in the lower right. You can now close the 'Menu Editor' browser tab and return to the '**Extension Directory**'

Select your new Extension, click the '**Edit**' button and select '**Publish**'



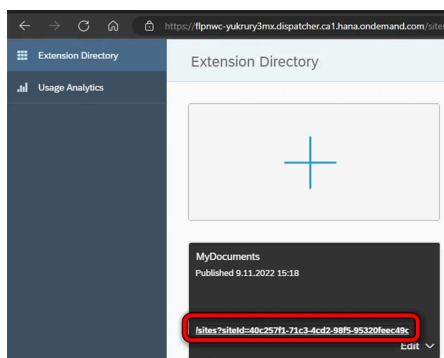
Custom Navigation Menu Setup

Next, register the Extension with the Custom Navigation in SuccessFactors

Navigate to the '**Configure Custom Navigation**' in SuccessFactors admin tools

Add a new line with the following values:

- Menu Item Label: Document Search
- ID: <next unused value>
- Link Type: External Link - New Tab/Window
- URL: Copy from the Extension Published



- Menu Location: <select where the menu should appear>
- Permissions: <select groups to see menu option>

NEXT Complete all of the Above Steps for the 'My Documents' application

5.3.3. SuccessFactors Configuration

This section describes the following configuration that must be completed in SuccessFactors.

5.3.3.1. API User

An API user is required for the docuflow for SuccessFactors middleware server.

Create a new user in SuccessFactors. This should be easily identifiable as an API user (i.e. apidflow)

Create a permission role for this user (i.e. apidflow_access)

The following permissions are required for the permission role:

Employee Data:

(Employee Profile)

- Country
- Email
- Employee Id
- Manager
- User ID
(User Information)
- Hire Date
(Employment Details)
- Employment Details MSS
- Hire Date

General User Permission:

- SFAPI User Login
 - User Login
- SAP System Configuration:
- View

Miscellaneous Permissions:

- Category Link Card 1 (view / edit)
- Category Link Card 3 (view / edit)
- Document Type Link Card 1 (view / edit)
- Document Type Link Card3 (view / edit)
- Employee Documents (view / view history / create / insert / correct / delete)
- Employee Documents.EmployeeDocumentChild (view / correct / create / adjust order / delete)
- HR Employee Documents (view / view history / create / insert / correct / delete)
- HR Employee Documents.HREmployeeDocumentChild (view / correct / create / adjust order / delete)
- Link Card Type 1 (View / Edit)
- Link Card Type 2 (View / Edit)
- Link Card Type 3 (View / Edit)
- Link Card Type 4 (View / Edit)
- cust_d4iFolderLink (View / Edit)
- SAP System Configuration (View)

Manage User

- User Account oData entity

Metadata Framework

- Admin access to MDF OData API

Manage Integration Tools

- oData API Attachment Export
- Allow Admin to Access OData API through Basic Authentication

Manage System Properties

- Picklist Management and Picklists Mappings Set Up

5.3.3.2. Create Pick List

The section will create the Pick List for document approval workflow status

Navigate to the Admin tool '**Pick List Center**'

Click '**New Pick List**'

In the Create Dialog, set the following values:

Name	Value	Notes
Code	LinkCard_DocStatus	
Name	Link Card Doc Status	
Status	Active	
Effective Start Date	<today's date>	
Display Order	Alpha	
Parent Picklist	<blank>	
Legacy Picklist ID	LinkCard_DocStatus	

Click the '**Save**' button

Create the following values:

Name	Value	Notes
External Code	1	
Label	InApproval	
Status	Active	
Non-unique External Code	1	

Name	Value	Notes
External Code	2	
Label	Rejected	
Status	Active	
Non-unique External Code	2	

Name	Value	Notes
External Code	3	
Label	Approved	
Status	Active	
Non-unique External Code	3	

5.3.3.3. Create Approval Workflow

The section will create the document approval workflow

Navigate to the Admin tool '**Manage Organization, Pay and Job Structures**'

Create New: Workflow

In the main workflow section, set the following values:

Name	Value	Notes
Workflow ID	d4i_WF	
Name	d4i_Approval	

In the section Step 1, set the following values:

Name	Value	Notes
Approver Type	Role	
Approver Role	Employee HR	
Context	Source	
Edit Transaction	Edit with Route Change	
No Approver Behaviour	Skip this Step	
Relationship to Approver	Employee	
Respect Permission	No	

5.3.3.4. SuccessFactors Objects

The following custom objects must be defined within the SuccessFactors administrative tools:

1. Category Link Card 1
2. Category Link Card 3
3. Document Type Link Card 1
4. Document Type Link Card 3
5. Link Card Type 1
6. Link Card Type 2
7. Link Card Type 3
8. Link Card Type 4
9. ESS Link Card Child
10. ESS Link Card Parent
11. HR Link Card Child
12. HR Link Card Parent
13. d4iFolderLink

NOTE: To access the administrative tools, navigate to '**Admin→Configure Object Definitions**'

5.3.3.4.1. Category Link Card 1

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	Category	
Effective Dating	None	
API Visibility	Editable	
Status	Active	
Label	Category Link Card 1	

In the '**Fields Section**', for '**external code**' select **details**

- In the details screen, set the value of '**Visibility**' to '**Not Visible**'
- NOTE: This prevents the UI from showing a count of the categories

In the '**Fields Section**', for '**external name**' select **details**

- In the details screen, set the value of '**Data Type**' to '**Translatable**'

Update the **Security Section** with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	<blank>	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	<blank>	

5.3.3.4.2. Category Link Card 3

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	Category_LinkCard3	
Effective Dating	None	
API Visibility	Editable	
Status	Active	
Label	Category Link Card 3	

In the '**Fields Section**', for '**external code**' select ***details***

- In the details screen, set the value of '**Visibility**' to '**Not Visible**'

In the '**Fields Section**', for '**external name**' select ***details***

- In the details screen, set the value of '**Data Type**' to '**Translatable**'

Update the **Security Section** with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	<blank>	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	<blank>	

5.3.3.4.3. Document Type Link Card 1

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	DocumentType_LinkCard1	
Effective Dating	None	
API Visibility	Editable	
MDF Version History	Complete History	
Status	Active	
Label	Document Type Link Card 1	

In the **Fields Section** for '**external code**' select **details**

- In the details screen, set the value of '**Visibility**' to '**Not Visible**'

In the '**Fields Section**', for '**external name**' select **details**

- In the details screen, set the value of '**Data Type**' to '**Translatable**'

In the **Associations** section set the following values:

Property	Value	Notes
Name	category	It will show as custom_category after its been created
Multiplicity	One to Many	
Destination Object	Category Link Card 1	Use the dropdown to find the custom object created for 'Category'
Type	Valid When	

Update the **Security** section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	<blank>	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	<blank>	

5.3.3.4.4. Document Type Link Card 3

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	DocumentType_LinkCard3	
Effective Dating	None	
API Visibility	Editable	
MDF Version History	Complete History	
Status	Active	
Label	Document Type Link Card 3	

In the **Fields Section** for '**external code**' select **details**

- In the details screen, set the value of '**Visibility**' to '**Not Visible**'

In the '**Fields Section**', for '**external name**' select **details**

- In the details screen, set the value of '**Data Type**' to '**Translatable**'

In the **Associations** section set the following values:

Property	Value	Notes
Name	CategoryLinkCard3	It will show as custom_CategoryLinkCard3 after its been created
Multiplicity	One to Many	
Destination Object	Category Link Card 3	Use the dropdown to find the custom object created for 'Category LinkCard3'
Type	Valid When	

Update the **Security** section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	<blank>	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	<blank>	

5.3.3.4.5. Link Card Type 1

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	d4i_LinkCard_1	
Effective Dating	None	
API Visibility	Editable	
Status	Active	
Label	Link Card Type 1	

Add the following fields to the d4i_LinkCard_1 object:

Name	Length	Data Type	More
user	100	user	Under details, set 'Required' to 'Yes'
Category	100	Generic Object	Valid Values Source: cust_Category_LinkCard3
DocumentType	100	Generic Object	Valid Values Source: cust_DocumentType_LinkCard3
Title	256	String	
UserID	38	String	
EmployeeName	256	String	
Country	128	String	
d4i_URL	256	String	
d4iUploadStatus	64	String	
CreatedDateTime	35	DateTime	
ownerId	38	String	
d4i_Status	64	String	

Update the Security section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	cust_user	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	mdfSystemEffectiveEndDate	

5.3.3.4.7. Link Card Type 3

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	d4i_LinkCard_3	
Effective Dating	None	
API Visibility	Editable	
Status	Active	
Label	Link Card Type 3	

Add the following fields to the d4i_LinkCard_3 object:

Name	Length	Data Type	More
user	100	user	Under details, set 'Required' to 'Yes'
Category	100	Generic Object	Valid Values Source: cust_Category_LinkCard3
DocumentType	100	Generic Object	Valid Values Source: cust_DocumentType_LinkCard3
Title	256	String	
UserID	38	String	
EmployeeName	256	String	
Country	128	String	
d4i_URL	256	String	
d4iUploadStatus	64	String	
CreatedDateTime	35	DateTime	
ownerId	38	String	
d4i_Status	64	String	

Update the Security section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	cust_user	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	mdfSystemEffectiveEndDate	

5.3.3.4.8. Link Card Type 4

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	d4i_LinkCard_4	
Effective Dating	None	
API Visibility	Editable	
Status	Active	
Label	Link Card Type 4	

Fields

Add the following fields to the d4i_LinkCard_4 object:

Name	Length	Data Type	More
user	100	user	Under details, set 'Required' to 'Yes'
Category	100	Generic Object	Valid Values Source: cust_Category_LinkCard3
DocumentType	100	Generic Object	Valid Values Source: cust_DocumentType_LinkCard3
Title	256	String	
UserID	38	String	
EmployeeName	256	String	
Country	128	String	
d4i_URL	256	String	
d4iUploadStatus	64	String	
CreatedDateTime	35	DateTime	
ownerId	38	String	
d4i_Status	64	String	

Security

Update the Security section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	cust_user	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	mdfSystemEffectiveEndDate	

5.3.3.4.9. ESS Link Card Child

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	d4i_LinkCard_ESS_Child	
Effective Dating	From Parent	
API Visibility	Editable	
Status	Active	
Label	Employee Document Child	
MDF Version History	Complete History	
Subject User Field		
Workflow Routing		
Pending Data	No	

Fields

In the '**Fields Section**', for '**externalName**' select **details**

- In the details screen, set the value of 'Visibility' to 'Not Visible'

Set the following fields on the cust_d4i_LinkCard_ESS_Child:

Name	Length	Data Type	More
externalCode	100	AutoNumber	This should exist, set type
Category	100	Generic Object	In Details , Set Valid Values Source to cust_category Set label to Document Category Set Required to Yes
DocumentType	100	Generic Object	In Details , Set Valid Values Source to cust_DocumentType_LinkCard1 Set Label to Document Type Also in Details, for Field Criteria, set 'Source Field Name' to cust_Category.mdfSystemInternalCode and 'Destination Field Name' to cust_Category Set Required to Yes
CollectionType	256	String	
d4iUploadStatus	38	Picklist	In Details , Set Valid Values Source to LinkCard_DocStatus
attachment	38	Attachment	Set Required to Yes

Security

Update the Security section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field		
CREATE Respects Target Criteria	No	
Base Date Field For Blocking		

Business Rule

The next step is to create a rule that will concatenate Category and Document Type into a single field that is used for data collection

Click on the plus next to 'Save Rules' to open '**Configure Business Rules**'

Select 'Basic'

Name	Value	Notes
Rule Name	LinkCard_CollectionType_Child	
Rule ID	LinkCard_CollectionType_Child	
Base Object	Employee Document Child	

In the **IF** statement section, select the '**Always True**' checkbox

Set the following Concatenation Rule:

Configure Business Rules

Search :Rule No Selection

LinkCard_CollectionType_Child (LinkCard_CollectionType_Child)

Scenario: Basic

Basic Information

Start Date 01/01/1900

Rule Type

Description

Parameters

Name	Object
Context	System Context
Employee Document Child	Employee Document Child

Variables

If

This rule is always true.
To add an expression please uncheck the Always True checkbox.

Then

Concatenate string

Save your changes to complete the rule

Navigate back to the Admin tool Configure Object Definitions

Open the ESS Link Card object and take action→make correction

Go to the section 'Save Rule'

- Select the **LinkCard_CollectionType_Child** rule that was just created and save the object

5.3.3.4.10. ESS Link Card Parent

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	d4i_LinkCard_ESS_Parent	
Effective Dating	Basic	
API Visibility	Editable	
Status	Active	
Label	Employee Documents	
MDF Version History	Complete History	
Default Screen	Employee Document Parent	
Subject User Field	externalCode	
Workflow Routing	d4i_Approval	
Pending Data	Yes	

Fields

Set the following fields on the cust_d4i_LinkCard_ESS_Parent object:

Name	Length	Data Type	More
externalCode	100	user	This should exist, set to user type Set Required: True

Associations

In the Associations section, add the following values:

Name	Multiplicity	Destination Object	Type
cust_EmployeeDocumentChild	One to Many	Employee Document Child	Composite

Security

Update the Security section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	externalCode	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	mdfSystemEffectiveEndDate	

5.3.3.4.11. HR Link Card Child

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	d4i_LinkCard_HR_Child	
Effective Dating	From Parent	
API Visibility	Editable	
Status	Active	
Label	HR Employee Document Child	
MDF Version History	Complete History	
Subject User Field		
Workflow Routing		
Pending Data	No	

Fields

In the '**Fields Section**', for '**externalName**' select **details**

- In the details screen, set the value of 'Visibility' to 'Not Visible'

Set the following fields on the cust_d4i_LinkCard_HR_Child:

Name	Length	Data Type	More
externalCode	100	AutoNumber	This should exist, set type
Category	100	Generic Object	In Details , Set Valid Values Source to cust_Category_LinkCard3 Set label to Document Category (set localization) Set Required to Yes
Document Type	100	Generic Object	In Details , Set Valid Values Source to cust_DocumentType_LinkCard3 Set Label to Document Type Also in Details, for Field Criteria, set 'Source Field Name' to cust_CategoryLinkCard3.mdfSystemInternalCode and 'Destination Field Name' to cust_Category Set Required to Yes
CollectionType	256	String	
d4iUpload Status	38	Picklist	In Details , Set Valid Values Source to LinkCard_DocStatus
attachment	38	Attachment	Set Required to Yes

Security

Update the Security section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field		
CREATE Respects Target Criteria	No	
Base Date Field For Blocking		

Business Rule

The next step is to create a rule that will concatenate Category and Document Type into a single field that is used for data collection

Click on the plus next to 'Save Rules' to open '**Configure Business Rules**'

Select '**Basic**'

Name	Value	Notes
Rule Name	HR_LinkCard_CollectionType_Child	
Rule ID	HR LinkCard_CollectionType_Child	
Base Object	HR Employee Document Child	

In the **IF** statement section, select the '**Always True**' checkbox

Set the following Concatenation Rule:

If

*This rule is always true.
To add an expression please uncheck the Always True checkbox.*

Then

Concatenate()

Concatenate string

- Sub string: ESS Link Card.Category Value.externalName
- Sub string: Text -
- Sub string: ESS Link Card.Document Type Value.externalName

Save your changes to complete the rule

Navigate back to the Admin tool Configure Object Definitions

Open the ESS Link Card object and take action→make correction

Go to the section 'Save Rule'

- Select the **HR_LinkCard_CollectionType_Child** rule that was just created and save the object

5.3.3.4.12. HR Link Card Parent

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section:

Property	Value	Notes
Code	d4i_LinkCard_HR_Parent	
Effective Dating	Basic	
API Visibility	Editable	
Status	Active	
Label	HR Employee Documents	Localize
MDF Version History	Complete History	
Default Screen	HR Employee Documents	
Subject User Field	externalCode	
Workflow Routing	d4i_Approval	
Pending Data	Yes	

Fields

Set the following fields on the d4i_LinkCard_HR_Parent object:

Name	Length	Data Type	More
externalCode	100	user	This should exist, set to user type Required: True

Associations

In the Associations section, add the following values:

Name	Multiplicity	Destination Object	Type
HREmployeeDocumentChild	One to Many	HR Employee Document Child	Composite

Security

Update the Security section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	externalCode	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking	mdfSystemEffectiveEndDate	

5.3.3.4.13. d4iFolderLink

Navigate to 'Create New' → 'Object Definition and fill in the following values into the header section

Property	Value	Notes
Code	d4iFolderLink	
Effective Dating	None	
API Visibility	Editable	
Status	Active	
Label	d4iFolderLink	

Fields

Set the following fields on the cust_d4iFolderLink object:

Name	Length	Data Type	More
FolderID	64	String	
User	100	user	Required = True

Searchable Fields

Update the Security section with the following values:

Field	Status	Notes
cust_User	Active	
cust_FolderID	Active	

Security

Update the Security section with the following values:

Property	Value	Notes
Secured	Yes	
Permission Category	Miscellaneous Permissions	
RBP Subject User Field	cust_User	
CREATE Respects Target Criteria	No	
Base Date Field For Blocking		

5.3.3.5. Permission Provisioning

Navigate to the Admin tool '**Manage Permission Roles**'

Employee Self Service Permissions to Objects

Apply Document Upload to People Profile

- Select 'ESS' Role
- Select 'Permission...'
- Select 'Employee Views'
- Check 'Document Upload'

Apply Permissions to Objects

- Select 'ESS'
- Select 'Permission...'
- Select 'Miscellaneous Permissions'
- Set the following Values:

Name	Setting	Notes
Category Link Card 1	View	
Document Type Link Card1	View	
Employee Documents	View / Create	
Employee Documents.EmployeeDocument Child	View Current / Create	Field Overrides Collection Type Value : No Access d4iUploadStatus: No Access
Link Card Type 1	View	
Link Card Type 4	View	

Manager Self Service Permissions to Objects

- Select 'MSS'
- Select 'Permission...'
- Select 'Miscellaneous Permissions'
- Set the following Values:

Name	Setting	Notes
Category Link Card 3	View	
Document Type Link Card3	View	
Link Card Type 1	View	
Link Card Type 2	View	

HRBP Permissions to Objects

- Select 'HRBP Role'
- Select 'Permission...'
- Select 'Miscellaneous Permissions'
- Set the following Values:

Name	Setting	Notes
Category Link Card 1	View	
Category Link Card 3	View	
Document Type Link Card 1	View	
Document Type Link Card 3	View	
HR Employee Documents	View Current / View History / Create / Correct / Delete	
HR Employee Documents.HREmployeeDocumentChild	View / Correct / Create / Delete	Field Overrides Collection Type Value : No Access d4iUploadStatus: No Access
Link Card Type 1	View / Edit	
Link Card Type 2	View / Edit	
Link Card Type 3	View / Edit	
Link Card Type 4	View / Edit	

Client Administrator Permissions to Objects

- Select 'Client Administrators Role'
- Select 'Permission...'
- Select 'Miscellaneous Permissions'
- Set the following Values:

Name	Setting	Notes
Category Link Card 1	View	
Category Link Card 3	View	
Document Type Link Card 1	View	
Document Type Link Card 3	View	
HR Employee Documents	View Current / View History / Create / Correct / Delete	
HR Employee Documents.HREmployeeDocumentChild	View / Correct / Create / Delete	Field Overrides Collection Type Value : No Access d4iUploadStatus: No Access
Link Card Type 1	View / Edit	
Link Card Type 2	View / Edit	
Link Card Type 3	View / Edit	
Link Card Type 4	View / Edit	

5.3.3.6. Export Category and Document Type

Use these instructions to get a set of CSV files to use to import the Categories and Document types into the system

For Employee Document Upload Rules/Categories:

Go to **'Import and Export Data'**

For Categories:

- Select the action to perform: Download Template
- Select Generic Object: Category Link Card 1
- Include dependencies: No
- Click 'Download'

For Document Type LinkCard1:

- Select the action to perform: Download Template
- Select Generic Object: Document Type LinkCard 1
- Include dependencies: No
- Click 'Download'

For the Relationship between Category and Document Types for Link Card 1

- Select the action to perform: Download Template
- Select Generic Object: Document Type LinkCard 1 - Category
- Include dependencies: No
- Click 'Download'

For HR Document Upload Rules/Categories:

Go to **'Import and Export Data'**

For Categories:

- Select the action to perform: Download Template
- Select Generic Object: Category
- Include dependencies: No
- Click 'Download'

For Document Type LinkCard3:

- Select the action to perform: Download Template
- Select Generic Object: Document Type LinkCard 3
- Include dependencies: No
- Click 'Download'

For the Relationship between Category and Document Types for Link Card 3

- Select the action to perform: Download Template
- Select Generic Object: Document Type LinkCard 3 - Category

- Include dependencies: No
- Click 'Download'

5.3.3.7. Import Category and Document Types

Use these instructions to use a set of CSV files to use to import the Categories and Document types into the system

For Employee Document Upload Rules/Categories:

Go to '**Import and Export Data**'

For Categories:

- Select the action to perform: Import Data
- Select Generic Object: Category
- File: CSV with your Category Data
- Purge Type: Incremental Load
- Click 'Import'

To check the imported data:

Go to '**Manage Data**'

- Under '**Search**' select Category
- Under the selection dropdown, the values should be shown

Complete the above steps for:

- Category Link Card 1.csv
- Category Link Card 3.csv
- Document Type Link Card 1.csv
- Document Type Link Card 3.csv
- Document Type Link Card 1-category.csv
- Document Type Link Card3-CategoryLinkCard3.csv

5.3.3.8. UI Configuration

This step will place the 'Upload Documents' portal option for the Workflow approval in the Employee File UI interface

Configure UI Element

Navigate to the admin tool '**Manage Configuration UI**'

Employee Documents

Create **Employee Documents**

- Click on 'Create New'
- Select '**Employee Documents**' in the Select Base Object
- Set the ID to '**Employee Documents Parent**'
- Click SAVE

Complete the following configuration:

- Employee Documents Header Element : Title Type User Defined = Employee Documents: (NOTE Set Translations as well)
- externalCode : Visible = No
- externalName : Visible = No
- effectiveStartDate : Visible = No
- In the Child Documents Header Section
 - Document Header : Title Type User Defined = Document (NOTE Set Translations as well)
 - Visible Fields Count = 8
 - Set Order of Properties: Document Category, Document Type, attachment
 - Hide the other fields

HR Employee Documents

Create **HR Employee Documents**

- Click on 'Create New'
- Select '**HR Employee Documents**' in the Select Base Object
- Set the ID to '**HR Employee Documents Parent**'
- Click SAVE

Complete the following configuration:

- HR Employee Documents Header Element : Title Type User Defined = HR Employee Documents: (NOTE Set Translations as well)
- externalCode : Visible = No
- externalName : Visible = No
- effectiveStartDate : Visible = No
- In the Child Documents Header Section
 - Document Header : Title Type User Defined = Document (NOTE Set Translations as well)
 - Visible Fields Count = 8

- Set Order of Properties: Document Category, Document Type, attachment
 - Hide the other fields

Configure People Profile

Navigate to the admin tool '**Configure People Profile**'

Click on '**Add New Section**' to insert a new UI section into the Employee Profile Screen

- Name the Section ***Document Upload***
- Name the Sub-Section: ***Documents***

Click on the Left side area

- Drop a 'Live Profile MDF Information' block
- Set the **MDF Screen ID** to '***Employee Document Parent***'

Click on the Right side area

- Drop a 'Live Profile MDF Information' block
- Set the **MDF Screen ID** to '***HR Employee Document Parent***'
- Save

5.3.4. docuflow for Integrations (d4i) Middleware Configuration

The goal of this guide is to establish base installation, initial configuration, and connectivity testing to a point that docuflow d4i components can communicate with one another and allows configuration of the software.

The docuflow application consists of a core Transformation engine and a series of connectors. Each connector has one or more connections defined which provide the information required to connect to the end applications.

NOTE: docuflow d4i was previously known as TRex and there are still occasional references to this name in the product

5.3.4.1. Start & Stop Order

d4i can be started and stopped via command line or as services (if installed as service)

Start Order

1. connectors (CNN-* eg. CNN-FS) (*no dependency*)
2. datastore (DAO)
3. config manager (CFGMGR)
4. transform engine (NGIN)

Stop Order

1. connectors (CNN-* eg. CNN-FS)
2. config manager (CFGMGR)
3. transform engine (NGIN)
4. datastore (DAO)

5.3.4.2. docuflow Middleware Installation

5.3.4.2.1. Unpacking ZIP Package

The docuflow middleware is delivered as a simple compressed 'zip' package. To install docuflow, simply unpack the zip file to a location you would like on the filesystem. For example d:\d4i

docuflow requires a java runtime environment (JRE) to run. By default, docuflow will reference the JRE deployed into the JDK directory in the location docuflow was installed. For example, if docuflow was installed to d:\d4i the JRE should be located at d:\d4i\JRE.

Please see '**Minimum Requirements**' in the technical overview for supported JRE versions.

5.3.4.2.2. Initial Configuration

The base docuflow installation will come pre-configured with the core system, Filesystem, IBM FileNet and SuccessFactors connectors. The main task to complete for initial setup is the configuration of connections for these connectors.

The internal docuflow connectors and core components communicate through HTTPS/TLS encrypted connections. To enable these secure connections, x.509 certificates must be supplied. docuflow provides mechanisms to generate these certificates.

Options when generating certificates include:

- Configuration of which connector certificates to create
- Configuration of the server IP address/DNS entry when remote connectors are required
- Creation of passwords for certificates generated

Settings are configured in the file `<install location>\config\trex-certmgr-application.json`

The `trex-certmgr-application.json` file contains the following sections:

"defaults"

"defaults" : this section allows settings that will be used for all certificates generated. This can be overridden in the specifics of each individual certificate section

"master"

"master": This is the section for the master keystore for all configured connectors on the system. The default settings will be sufficient for a one FileNet installation.

For each connector, the value for **"disabled"** should be set to **'false'** to have a certificate generated.

If distributing a connector on a remote server, the **"san"** section can be used to define the remote server name and/or ip address.

An example snippet for the file system connector is shown below, including a "san" section.

```
{
  "disabled": false,
  "alias": "trex.cnn-trex-fs.server",
  "dn": {
    "commonName": "VersaFile TRex File System Connector"
  },
  "san": {
    "dns": ["fully.qualified.name"],
    "ip": ["127.0.0.1"]
  }
}
```

"keystores"

"keystores": The keystores section is where docuflow configuration for the p12 keystore certificates is done. Each connector must have an alias entry in the section "keystore": **"trex.cnn.keystore.p12"**. The **"storepass"** entry is where a password for the keystore can be configured.

The section that will need configuration is the trex.cnn.keystore.p12. This contains references to each of the connectors being used.

"truststores"

"truststores": The truststores section is where docuflow configuration for the p12 truststore certificates is done. These trusts allow the connectors and core docuflow components to exchange their certificates.

The **"storepass"** entry allows for configuration of the password for the truststore.

There are two sections that will need configuration:

- trex.ngin.truststore.p12 (allows connections to the core d4i transformation engine)
- trex.cfgmgr.truststore.p12 (allows connections to the d4i configuration UI engine)

Both of these sections contain references to each of the connectors being used. **Note:** this should match the list from the keystore section.

An example of this section is shown below, with 5 connectors being setup:

```
"truststores": [
  {
    "keystore": "trex.dao.truststore.p12",
    "storepass": "<password>",
    "aliases": [ "trex.cfgmgr.server", "trex.ngin.server" ]
  },
  {
    "keystore": "trex.ngin.truststore.p12",
    "storepass": "<password>",
    "aliases": [
      "trex.dao.server",
      "trex.cfgmgr.server",
      "trex.cnn-trex-fs.server",
      "trex.cnn-cm8.server",
      "trex.cnn-elst.server",
      "trex.cnn-trex-util.server",
      "trex.cnn-ibm-p8ce.server"
    ]
  },
  {
    "keystore": "trex.cfgmgr.truststore.p12",
    "storepass": "<password>",
    "aliases": [
      "trex.cfgmgr.client",
      "trex.dao.server",
      "trex.ngin.server",
      "trex.cnn-trex-fs.server",
      "trex.cnn-cm8.server",
      "trex.cnn-elst.server",
      "trex.cnn-trex-util.server",
      "trex.cnn-ibm-p8ce.server"
    ]
  },
  {
    "keystore": "trex.cnn.truststore.p12",
    "storepass": "<password>",
    "aliases": [ "trex.cfgmgr.server", "trex.ngin.server" ]
  }
],
```

"exports"

"exports": The exports section configures the exported certificates for the system. Of note is the **"privatekeys"** section. This will generate a .p12 certificate that can be imported into a web browser to allow connectivity to the docuflow configuration service. Set a storepass that will be used. NOTE: Here you can configure a separate certificate/password for different environments (DEV vs PROD etc).

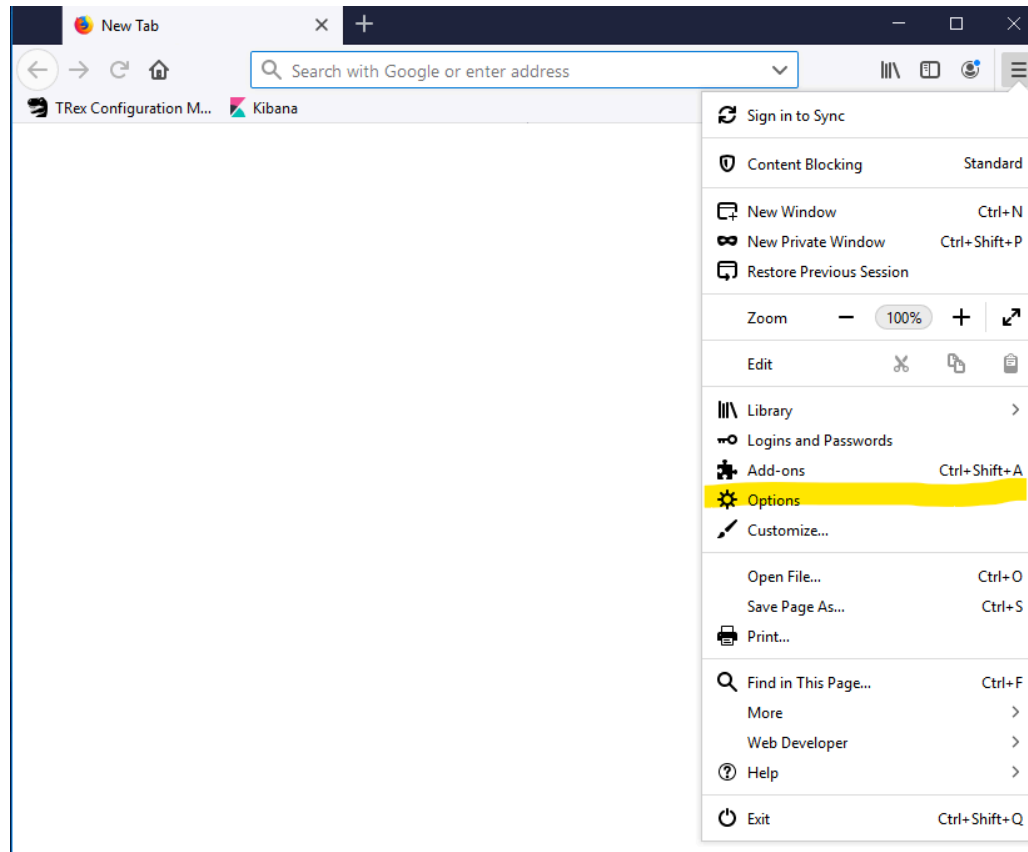
start-cert-mgr.ps1

Once configuration is complete, the powershell script '**<install location>start-cert-mgr.ps1**' should be run. This script will start a java application that will generate (or re-generate) the application certificates.

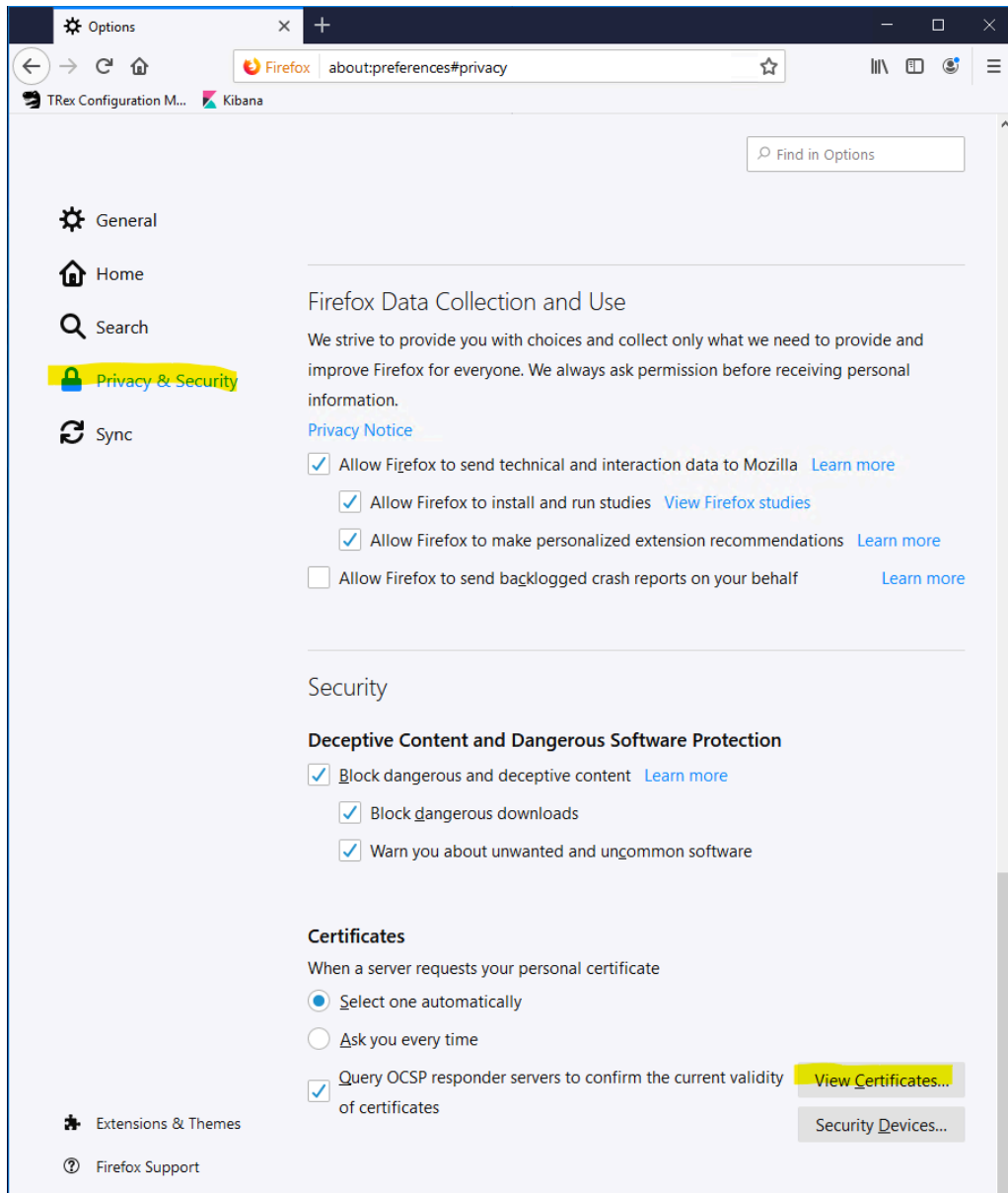
5.3.4.2.3. Setting up Firefox Certificate

The docuflow configuration service requires a HTTPS secured connection in order to communicate with a web browser. To enable this connection, a client certificate must be imported into the browser. This procedure explains how to install the certificate into Mozilla Firefox.

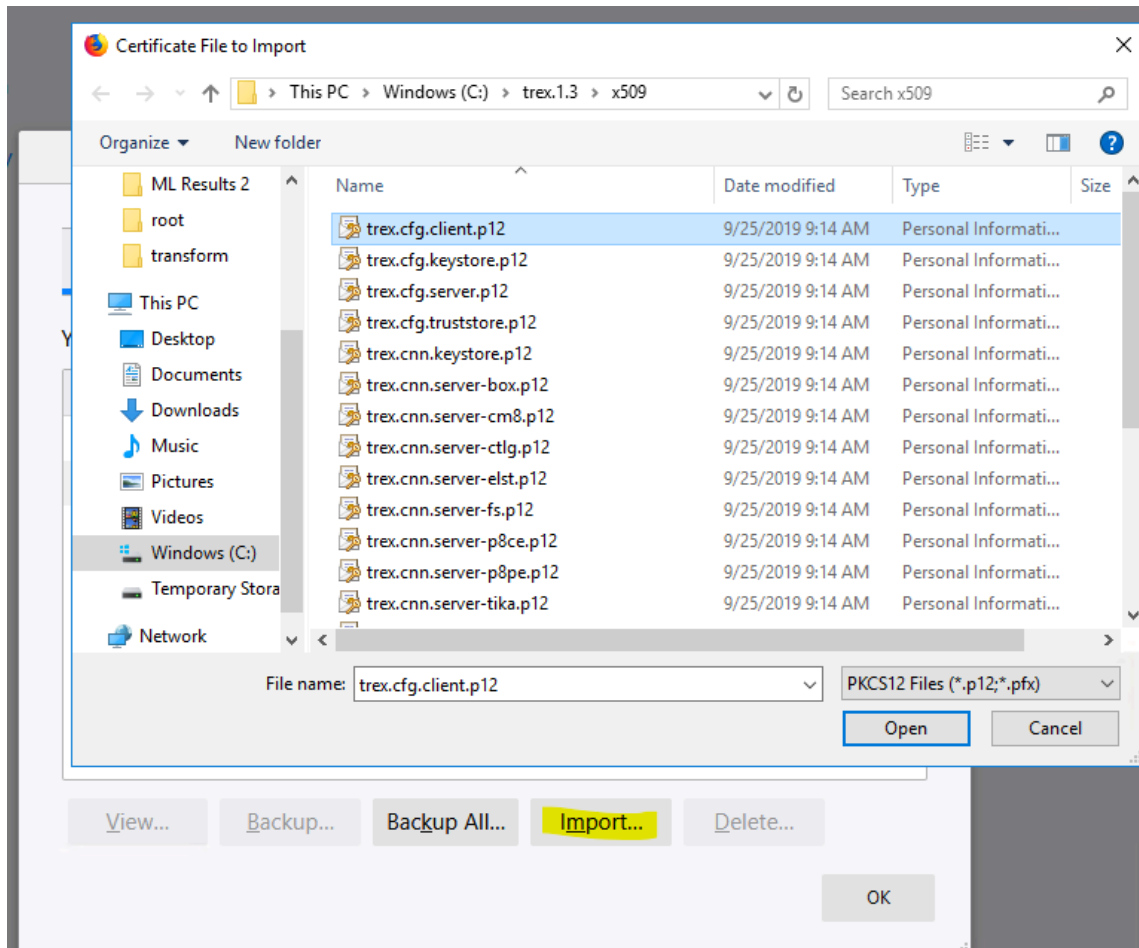
Open the Options section of Firefox



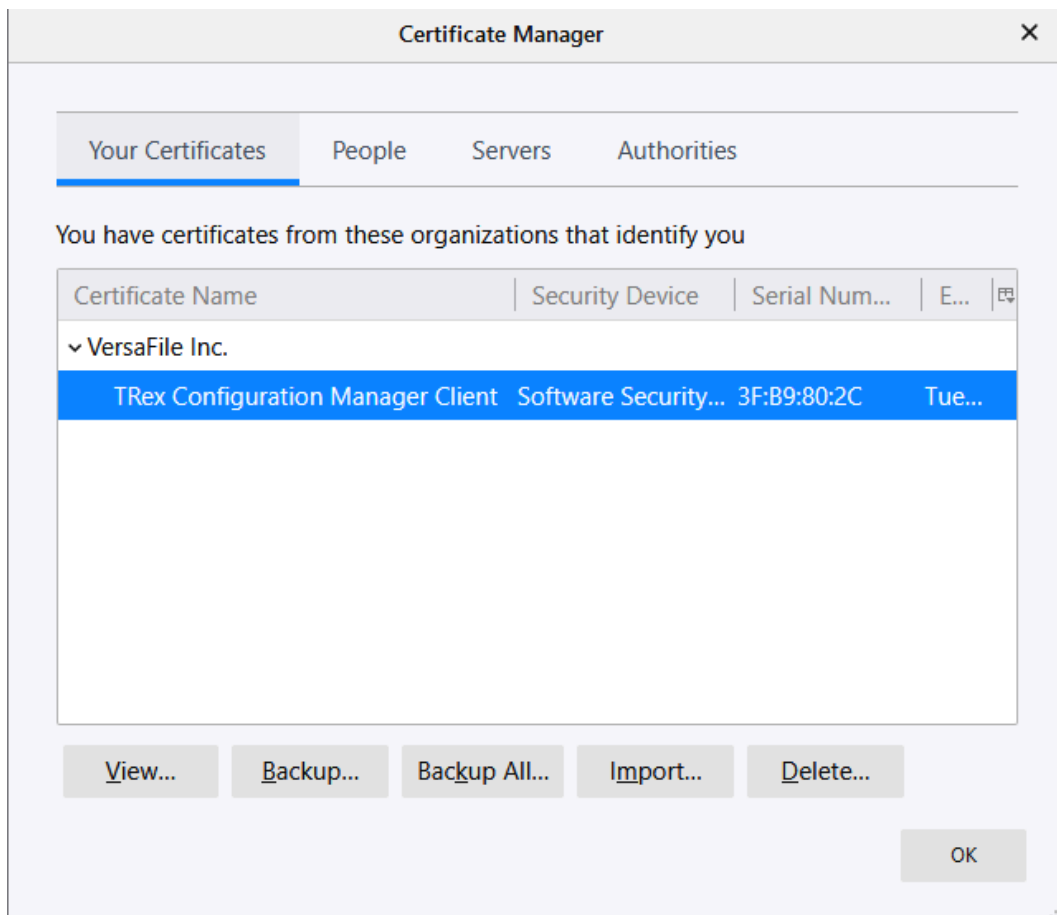
Select '**Privacy & Security**' section, scroll to the bottom and select View Certificates and then click '**Import**'



Select the 'exports' directory where you have created the x509 certificates (typically a directory called x509 in the root docuflow folder) and then select '***trex.cfgmgr.client.p12***' then supply the password that was configured in the '***trex-certmgr-application.json***' file.



You should see something similar to the following



5.3.4.2.4. Connecting to docuflow Configuration

Once all of the configuration steps are completed and the client certificate has been imported into your browser, Launch the windows power shell, go to File **<installocation>** , then run the powershell script **.\start-trex.ps1**.

If the file **<installocation>\start-trex.ps1** cannot be loaded, then the file **<installocation>\start-trex.ps1** is not digitally signed. And you cannot run this script on the current system. You will need to change the Execution policy to run these PowerShell scripts.

To display the execution policies for each scope in the order of precedence, use **"Get-ExecutionPolicy -List"**. To see the effective execution policy for your PowerShell session, use **"Get-ExecutionPolicy"** with no parameters.

The effective execution policy is determined by order of precedence as follows:

- MachinePolicy: Set by a Group Policy for all users of the computer.
- UserPolicy: Set by a Group Policy for the current user of the computer.
- Process: Affects only the current PowerShell session.
- CurrentUser: Affects only the current user.
- LocalMachine: Default scope that affects all users of the computer.

To change and set a new execution policy for the local computer for instance:

Use **Set-ExecutionPolicy** to specify the RemoteSigned policy. Then enter the Scope parameter to specify the default scope value, LocalMachine.

Set-ExecutionPolicy -ExecutionPolicy RemoteSigned -Scope LocalMachine

For more information on changing the execution policy, please visit <https://go.microsoft.com/fwlink/?LinkID=135170>.

After changing the execution policy, run **D:\docuflow> .\start-trex.ps1** A number of PowerShell windows will open up. Type "R" and run all scripts.

Once all scripts run, you can connect to the TRex Configuration Service UI. <https://<servername/localhost>:8443>

5.3.4.2.5. SSL Certificates Modifications

docuflow requires SSL certificates to secure communications amongst applications within the platform. docuflow does provide the ability to create its own self-signed certificates. Certificates are generated through the Certificate Manager application.

The following are execution guidelines for managing the certificates for docuflow .

Regenerate and Configure Certificates

After completing file modifications, such as those noted below, the following steps are required to regenerate and configure the new certificates for use with the docuflow platform.

1. Stop all docuflow applications
2. Rename (or delete) the <docuflow INSTALL LOCATION>\x509 directory
e.g. x509.old
3. Start the Certificate Manager application
 1. <docuflow INSTALL LOCATION>\start-cert-mgr.ps1
 2. Upon successful completion, exit/stop the Certificate Manager application
4. Start docuflow
5. Close and open a new browser instance
6. Import the newly generated certificates; follow the existing steps [here](#).
 1. Note: it not required, but recommended, to remove the previously configured certificates:
 1. Personal > VersaFile Configuration Manager Client
 2. Trusted Root Certification Authorities > VersaFile TRex Root CA

Modify the Expiry Length

- The expiry length for certificates is located within the <docuflow INSTALL LOCATION>\config\trex-certmgr-application.json file
 - within the defaults.validity object
 - default value 3650 (units in days; 10 years)
- To modify the expiry length of all certificates, edit the trex-certmgr-application.json file
 1. Modify the defaults.validity value
e.g. for 1 year expiry length

```
"validity": 365
```
 2. Save the file edit
 3. Execute the [Regenerate and Configure Certificates](#) steps, noted above.

Modify a Password

- Each application with the docuflow platform is created with its own truststore and keystore, to permit or restrict communication amongst applications.
- Passwords exist for each application's truststore and keystore. These passwords exist in:

- <docuflow INSTALL LOCATION>\config\trex-certmgr-application.json
- One of the following:
 - <docuflow INSTALL LOCATION>\config\application-cfgmgr.yaml
 - <docuflow INSTALL LOCATION>\config\application-dao.yaml
 - <docuflow INSTALL LOCATION>\config\application-ngin.yaml
 - <docuflow INSTALL LOCATION>\config\application-cnn.yaml
 - each connector has its own respective yaml config file; however, the application-cnn.yaml is inherited in each case as well
- A password configuration also exists for the client certificate, which exists in <docuflow INSTALL LOCATION>\config\trex-certmgr-application.json. Specifically:
 - exports.privatekeys.storepass object
- To modify a password, edit the trex-certmgr-application.json file and edit the respective yaml file, as required.
 - For example, to modify the password for both Configuration Manager certificates and the client certificate:

1. trex-certmgr-application.json edits:

1. Navigate to keystores.keystore : "trex.cfgmgr.keystore.p12"

2. Edit the storepass object, e.g.

```
{
  "keystore": "trex.cfgmgr.keystore.p12",
  "storepass": "newPassword1",
  "aliases": ["trex.cfgmgr.server"]
},
```

3. Navigate to truststores.truststore : "trex.cfgmgr.truststore.p12"

Edit the storepass object, e.g.

```
{
  "keystore": "trex.cfgmgr.truststore.p12",
  "storepass": "newPassword2",
  "aliases": [
    "trex.cfgmgr.client",
    ...
    "trex.cnn-trex-websvc.server"
  ]
},
```

4. Navigate to exports.privatekeys

Edit the storepass object, e.g.

```
"privatekeys": [{
  "alias": "trex.cfgmgr.client",
  "file": "exports/trex.cfgmgr.client.p12",
  "storepass": "newPassword3"
}]
```

2. application-cfgmgr.yaml edits

1. Navigate to and edit the server.ssl object

2. Edit the key-store-password and trust-store-password values, e.g.

```
server:
  port: 8443
```

```

ssl:
  key-store: ./x509/trex.cfgmgr.keystore.p12
  key-store-password: newPassword1
  key-alias: trex.cfgmgr.server
  trust-store: ./x509/trex.cfgmgr.truststore.p12
  trust-store-password: newPassword2

```

3. Execute the [Regenerate and Configure Certificates](#) steps, noted above.

◦ Password creation guidelines:

- It is recommended not to use, but if using the \ or " characters, these must be escaped with the \ character within the json and yaml files, e.g.
 - not escaped: *this"isaPwd*
 - escaped: *this|"is||aPwd*
- Although not used by default, it is recommended to surround the passwords in the yaml file with double quotes. e.g.

```

server:
  port: 8443
  ssl:
    key-store: ./x509/trex.cfgmgr.keystore.p12
    key-store-password: "newPassword1"
    key-alias: trex.cfgmgr.server
    trust-store: ./x509/trex.cfgmgr.truststore.p12
    trust-store-password: "newPassword2"

```

Modify Access to Configuration Manager

- By default, Configuration Manager is configured as accessible locally through a browser through `http://localhost:8443`
- An IP or DNS configuration can be added and included in the certificate creation.

1. Edit the `trex-certmgr-application.json` file
2. Navigate to the `master.certificates` object
3. Edit the `"alias": "trex.cfgmgr.server"` object

1. For a new DNS entry, add in a new entry in the `"dns"` array. For example, to add a DNS config with a name of *trex.cfgmgrtest.com* with an IP of *188.77.166.88*

```

{
  "alias": "trex.cfgmgr.server",
  "dn": {
    "commonName": "VersaFile Configuration Manager Application"
  },
  "san": {
    "dns": ["localhost", "trex.cfgmgrtest.com"],
    "ip": ["127.0.0.1", "188.77.166.88"]
  }
},

```

Note: an IP can be added on its own, without a DNS entry

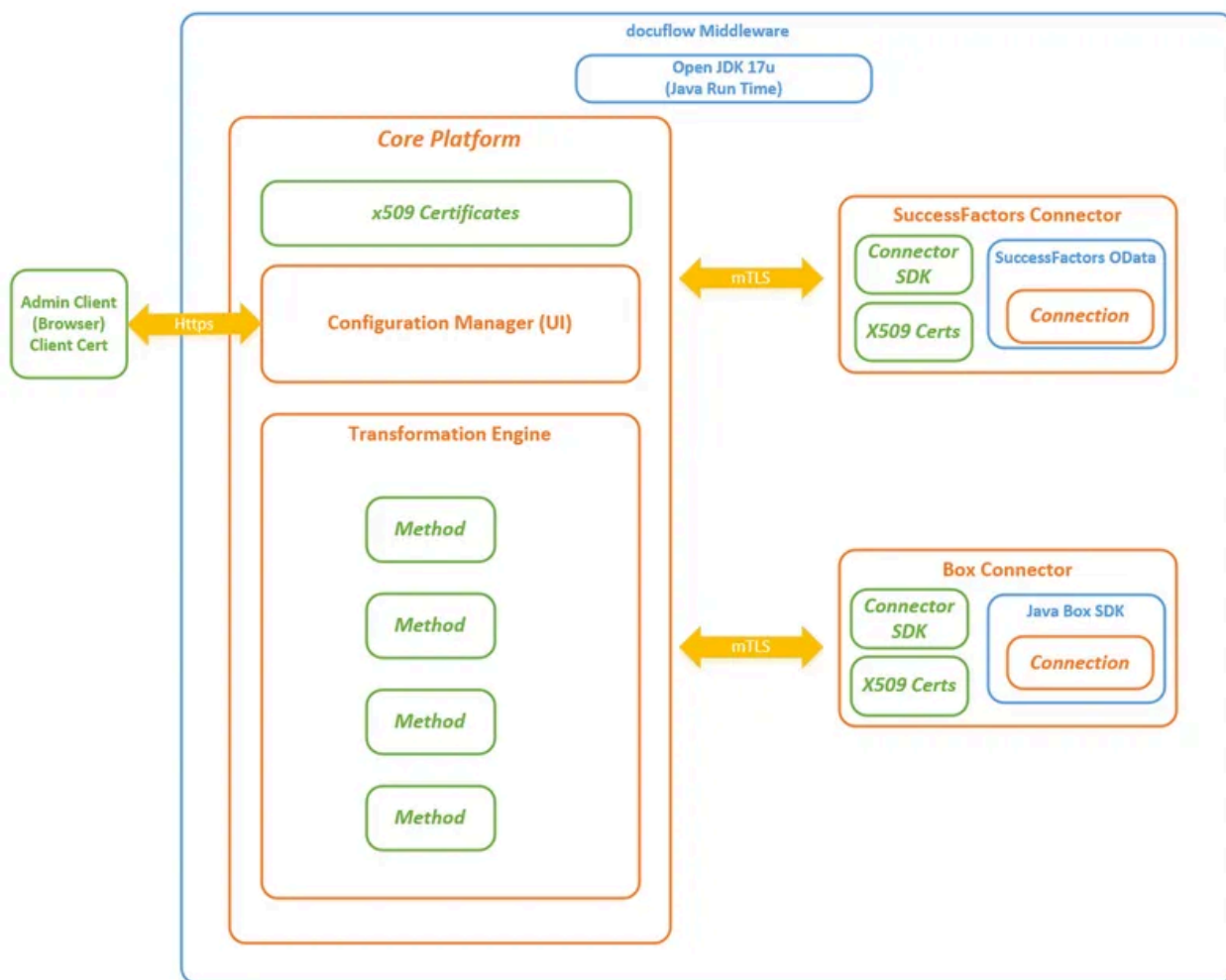
4. Execute the [Regenerate and Configure Certificates](#) steps, noted above.

5.3.4.3. docuflow Base Configuration

The Configuration of docuflow falls into four key components:

1. Engines
2. Connectors
3. Connections
4. Transformations

For docuflow for SuccessFactors and IBM FileNet, the following architecture and connectors are used:



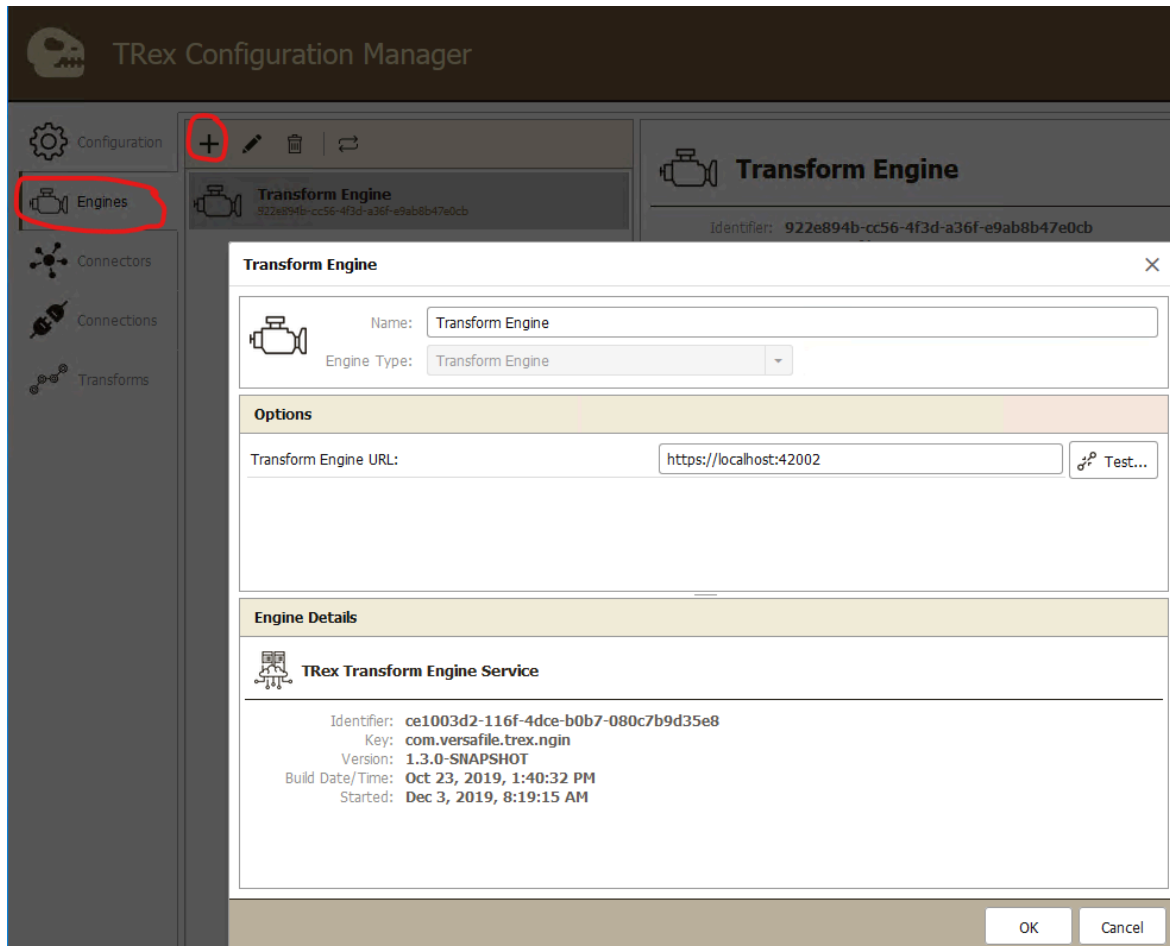
5.3.4.3.1. Engines

To add the main Transformation Engine to docuflow, select the 'Engines' tab on the right, then click on the '+' button

Fill in the URL using HTTPS protocol to the Transformation Engine service. By default this will be on the local machine on port 42002.

For example: `https://localhost:42002`

Click on the 'Test' button and after a successful connection, the Engine Details will be shown.



5.3.4.3.2. Connectors

Each connector is configured by linking

Select the 'Connectors' tab, then click the '+' button

Enter the URL to the connector endpoint using HTTPS and providing the port

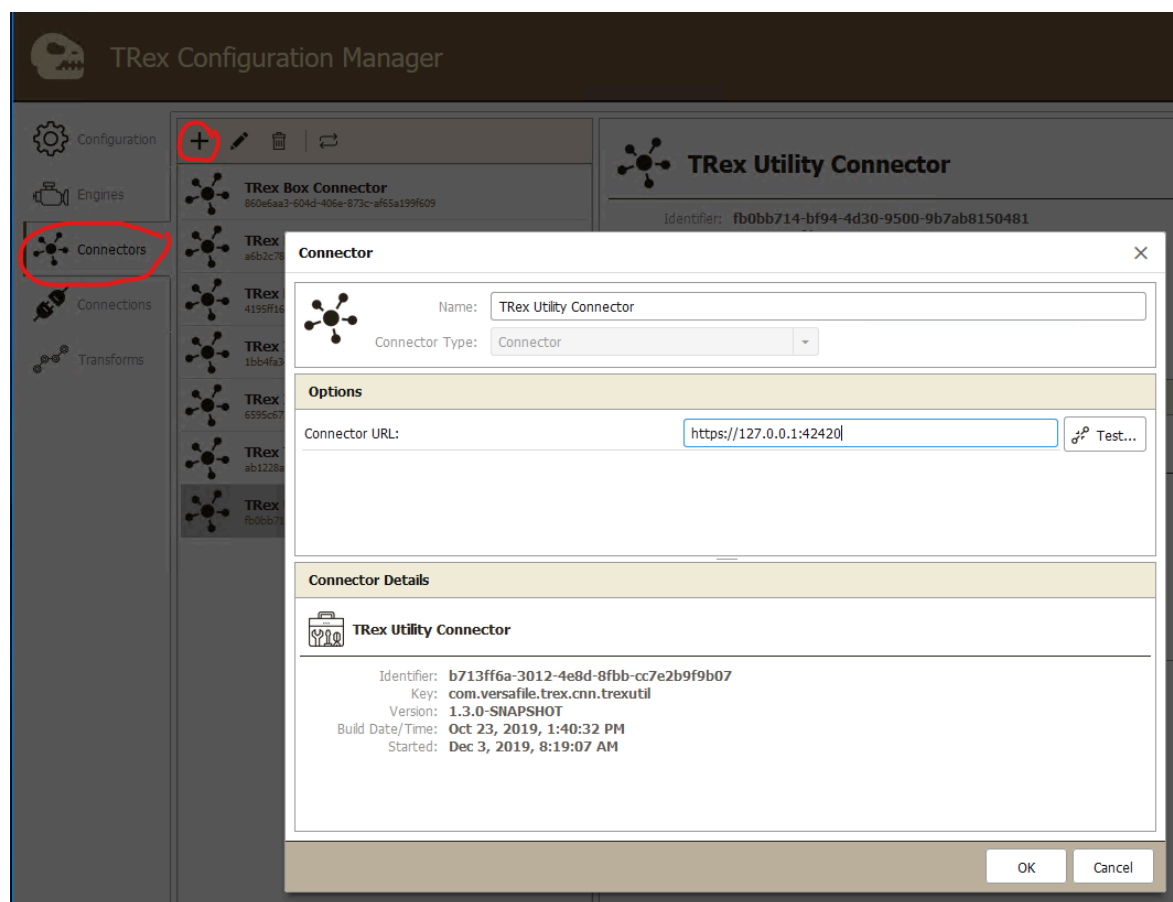
In default installations localhost or loopback addresses can be used. In advanced configurations where connectors are deployed on remote servers, the remote machine address/name should be used:

```
https://127.0.0.1:42420
```

```
https://localhost:42420
```

```
https://<remoteconnectorserver>:42420
```

Note: that 127.0.0.1 is a loopback ip4 address - the same as using localhost. This indicates the connector is running on the same machine as the TRex core services.



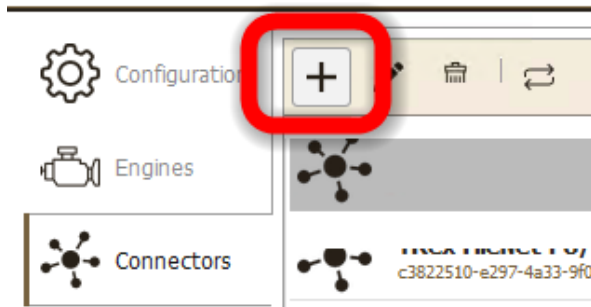
The default ports for the connectors assigned in the TRex configuration are listed in the table below:

Trex Utilities	42420
Filesystem	42421
IBM FileNet P8 Content Engine	42422
Apache Tika	42423
IBM Watson Cloud Services	42424
Elasticsearch	42425
IBM FileNet P8 Process Engine	42426
Box Cloud Content Management	42427
IBM Content Manager CM8	42429
IBM Forms Server	42430
Microsoft SharePoint	42431
Generic Web Services	42432

5.3.4.3.2.1. IBM FileNet

To Create a new Box Connector in the docuflow Configuration UI:

1. Click on the 'Connectors' Tab
2. Click on the + icon
3. Enter the Connector URL (**https://localhost:42426**)



Adding a Connector

Connector Description

The Box connector provides connectivity to the IBM FileNet Content Platform Engine (CPE) for document storage, retrieval, and management.

IBM FileNet is an Enterprise Content Management (ECM) platform, and all currently supported versions of IBM FileNet P8 CPE are compatible with this connector. (??? for validation)

Methods Included:

File Operations

- Delete File
- Fetch File
- File Iterator
- Update File
- Upload File

Folder Operations

- Create Folder
- Delete Folder
- Folder Iterator
- Update Folder

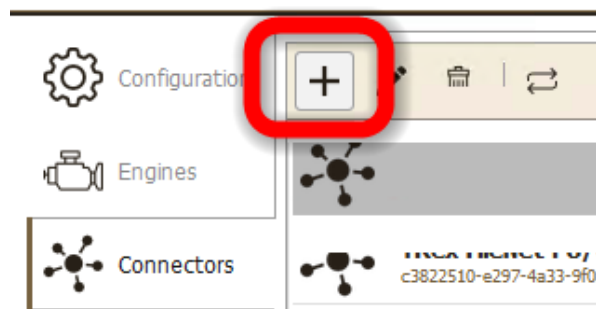
Search

- Metadata Query
- Search

5.3.4.3.2.2. Filesystem

To Create a new Filesystem Connector in the docuflow Configuration UI:

1. Click on the 'Connectors' Tab
2. Click on the + icon
3. Enter the Connector URL (**https://localhost:42421**)



Adding a Connector

Connector Description

The filesystem connector allows for connections to any file locations that are accessible from the server d4i is installed on.

The Filesystem connector takes an argument on its mount point to connect to. Only this location and folders/files below will be accessible.

This is configured in file `./config/application-cnn-fs.yaml`. By default, it will start looking for a folder called 'root' within the installation location of TRex. To change this, update the 'mount-point' variable.

```
plugin-args:
  mount-point: ./root
```

This can be changed to a local drive location such as C: or D: or D:\mystuff. Additionally, remote shares can be access using UNC paths. For example: \\servername\share

NOTE: If you are mounting a remote filesystem share, the Filesystem service should be configured to log in with an account that has access/priviledges to the remote share.

Methods Included:

CSV File Operations

- CSV Row Iterator
- CSV Write Row

File Conversion

- Convert to PDF

File Operations

- Copy File
- Delete File
- File Digest

- [File Iterator](#)
- [Move File](#)
- [Write Content To File](#)

Folder Operations

- [Delete Folder](#)
- [Folder Iterator](#)

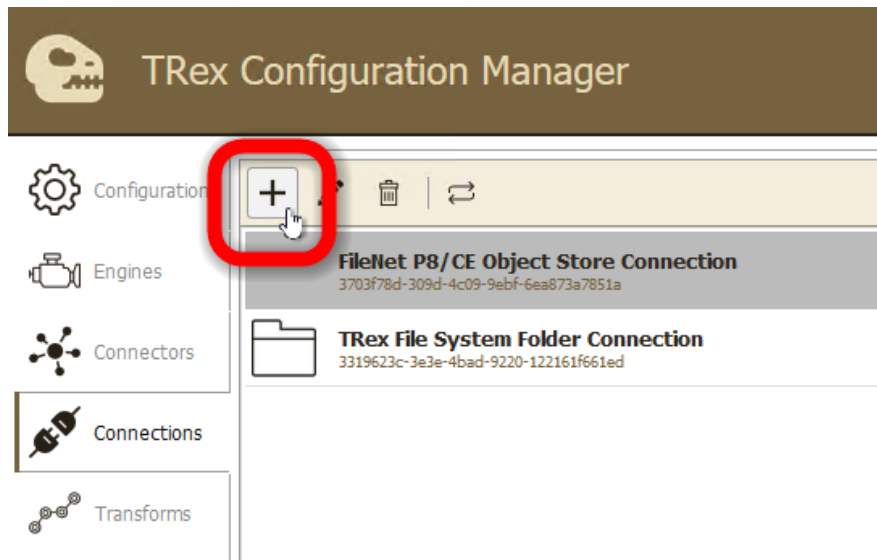
XML File Operations

- [XML Node Iterator](#)

5.3.4.3.3. Connections

Once Connectors have been created, the Connections can be created.

To create a new connection, select the '**Connections**' Tab and then the + (add...) button.



Each connection will require connection specific information and is detailed in the following sections.

5.3.4.3.3.1. IBM FileNet

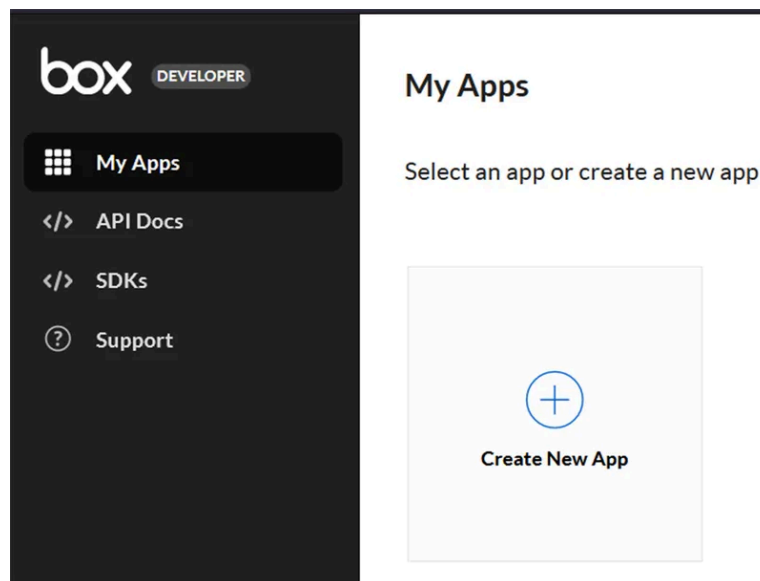
Configuring an IBM FileNet Connection

(TO BE UPDATED FOR FILENET)

Go to your Box developer console:

- <https://app.box.com/developers/console>

Under MyApps, click on the 'Create New App' button



Under 'Lets Get Started' - Select **Custom App**

Under 'Authentication Method' - Select **Server Authentication (with JWT)** and click Next

Custom App ✕

Select an authentication method and enter a name for your app. You can change the name later in General Settings.

Authentication Method

☒ **Server Authentication (with JWT)**

- Best for building integrations or apps with external collaborators - [Learn More](#)
- The app authenticates using a JSON Web Token. Requires a public/private key-pair for added security.

☐ **User Authentication (OAuth 2.0)**

- Best for building mobile or web apps - [Learn More](#)
- Users must authenticate with their Box login.

☐ **Server Authentication (Client Credentials Grant)**

- Best for scripting and back office integrations - [Learn More](#)
- The app authenticates using Client ID and Client Secret.

App Name

Enter an app name

By clicking 'Create App', you agree to the terms of the [Box Developer Agreement](#) and the [Box Privacy Policy](#).

Cancel Create App

Name your application e.g. '**d4iConnector**' and click 'Create App'

Click 'View Your App' and navigate to the '**Configuration**' tab

Under **Application Scopes** select the following options:

- Read all files and folders stored in Box
- Write all files and folders stored in Box
- Manage users (allows App users to be created)
 - Note: if you want to script out and create app users without docuflow, this setting is not require
 - Also: This setting can be used for the application to create the app users, and once the process is complete, it can be removed

Under **Advanced Features** select the following options

- Make API calls using the as-user header (allows scaling across app users)
- Generate user access tokens

Under **Add and Manage Public Keys** select 'Generate a Public/Private Keypair' and Save the json file somewhere secure. This JSON will be required for the Box Connection configuration. ***This private key cannot be regenerated and must be stored safely and securely.***

Under **App Settings** select the 'Download JSON' - this JSON file is required for the Box connector

After completing these steps go to the 'App Authorization' and click 'Submit for Authorization'

Return to the Admin console, go to 'Apps' → Custom Apps

- Click 'Authorize New App' and then enter the Client ID from the previous step
- Click the Authorize button

Creating a d4i Box Connection

There are two types of connections supported for connecting to Box: Enterprise and AppUser. When creating a new Box connection this can be selected in the 'connection type' dropdown.

Box Enterprise Connection

The Enterprise connection uses the default service account for connecting to Box.

The following information is required when creating a new Enterprise connection:

- **Enterprise ID:** This can be found in the Box Admin under the **Account and Billing** section
- **Client ID:** This can be found in the **OAuth 2.0 Credentials** section of the Box Enterprise Application configured earlier
- **Client Secret:** This can be found in the **OAuth 2.0 Credentials** section of the Box Enterprise Application configured earlier
- **Public Key ID:** This can be found in the **Add and Manage Public Keys** section of the Box Enterprise Application configured earlier
- **Private Key:** This is the Private Key that corresponds to the Public key that was registered with the Box Enterprise Application. Must be entered properly formatted with multiple lines (not only one long line).
- **Passphrase:** This is the passphrase for the Key registered with the Box Enterprise Application.

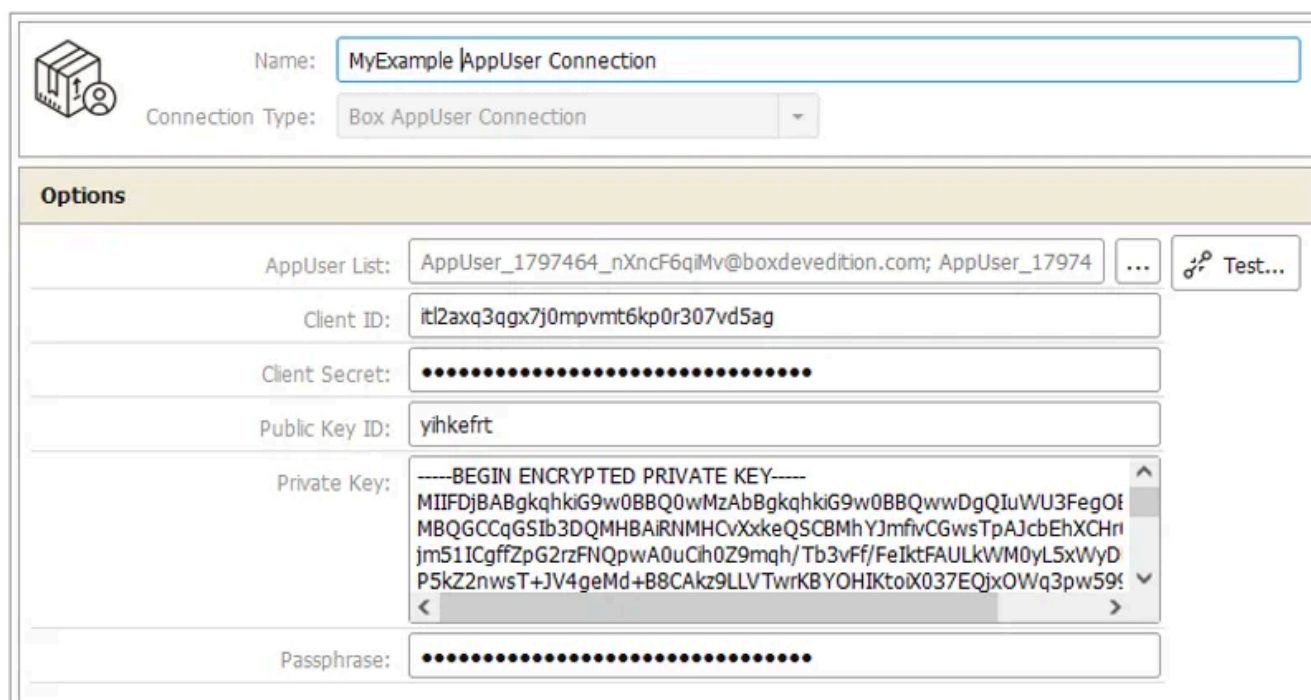
Box AppUser Connection

The AppUser connection uses a defined set of AppUsers associated with the Box Enterprise Application to connect to Box. The d4i Box connector will round robin between the provided App Users configured in the connector.

The following information is required when creating a new AppUser connection:

- **AppUser List:** A list of Box App users that have been previously created under the Box Enterprise Application
- **Enterprise ID:** This can be found in the Box Admin under the **Account and Billing** section
- **Client ID:** This can be found in the **OAuth 2.0 Credentials** section of the Box Enterprise Application configured earlier
- **Client Secret:** This can be found in the **OAuth 2.0 Credentials** section of the Box Enterprise Application configured earlier
- **Public Key ID:** This can be found in the **Add and Manage Public Keys** section of the Box Enterprise Application configured earlier
- **Private Key:** This is the Private Key that corresponds to the Public key that was registered with the Box Enterprise Application. Must be entered properly formatted with multiple lines (not only one long line).
- **Passphrase:** This is the passphrase for the Key registered with the Box Enterprise Application.

Once a connection is created, a connection test can be run to check if the configuration is valid.



The screenshot shows a web form for configuring a 'Box AppUser Connection'. At the top, there is a 'Name' field containing 'MyExample AppUser Connection' and a 'Connection Type' dropdown menu set to 'Box AppUser Connection'. Below this is an 'Options' section with several fields: 'AppUser List' (containing 'AppUser_1797464_nXncF6qIMv@boxdevedition.com; AppUser_17974' and a 'Test...' button), 'Client ID' (containing 'itl2axq3qgx7j0mpvmt6kp0r307vd5ag'), 'Client Secret' (masked with dots), 'Public Key ID' (containing 'yihkefrt'), 'Private Key' (containing a multi-line encrypted key starting with '-----BEGIN ENCRYPTED PRIVATE KEY-----'), and 'Passphrase' (masked with dots). A scrollbar is visible on the right side of the Private Key field.

5.3.4.3.3.2. Filesystem

The filesystem connection allows for the setting of a subfolder root that is relative to where the connector is running.

For example, if the connector service was connected to the root of D:\ then the value entered would be a subdirectory of D:\. Use the default ./ to allow access to all subdirectory locations.



Name:

TRex File System Folder Connection

Connection Type:

TRex File System Folder Connection

Options

Root Path:

./

 Test...

Output Entity

 Folder Entity

Attributes

Breadcrumb:

WTI5dExuWmxjbk5oWm1sc1pTNTBjbVY0TG10dWJpNW1jdz09fl

Folder Name:

root

Folder Path:

.\