Low Level Solution Design

|  |  |  |
| --- | --- | --- |
| Program | : | Human Resource Document Management System |
| Business Unit | : | Human Resource |
| Project Name | : | KOTAK HRDMS |

|  |  |
| --- | --- |
| Date Issued | 11-07-2019 |
| Version No. | 1.1 |
| Status | Draft |
| Prepared By | Bhavik Vora |

Table of Contents

[References 3](#_Toc14113782)

[Glossary 3](#_Toc14113783)

[1. Introduction 4](#_Toc14113784)

[1.1 Purpose 4](#_Toc14113785)

[1.2 Architectural representations 5](#_Toc14113786)

[1.2.1 Logical View 5](#_Toc14113787)

[2. Content Design 6](#_Toc14113788)

[2.1 Domain, Site, Virtual Server and Server Instance 6](#_Toc14113789)

[2.2 Object Store Design 7](#_Toc14113790)

[2.3 Security Policies Object store level permission is given to below AD groups only,In order to use HRDMS Objectstore 7](#_Toc14113791)

[2.4 Storage Policies 8](#_Toc14113792)

[2.5 Property Templates 9](#_Toc14113793)

[2.6 Document Classes 10](#_Toc14113794)

[2.7 Folder 12](#_Toc14113795)

[3. Master Data & Document Upload Applications 13](#_Toc14113796)

[3.1 Master data lookup 13](#_Toc14113797)

[3.1.1 Master Data mapping from HRMS 13](#_Toc14113798)

[3.1.2 User Role mapping from Kraman 13](#_Toc14113799)

[3.2 Document Upload 14](#_Toc14113800)

[3.2.1 CMIS Web service Configuration 15](#_Toc14113801)

[3.2.2 Configuration for DMS Document Upload Feature 16](#_Toc14113802)

[3.2.3 Bulk upload thick client utility 17](#_Toc14113803)

[4. Document Services 19](#_Toc14113804)

[4.1 Add Document 19](#_Toc14113805)

[4.1.1 Document addition from front-end application: DMS 19](#_Toc14113806)

[4.1.2 Document addition from Hire Pro application: Using CMIS Service 19](#_Toc14113807)

[4.1.3 Document addition from Hire Pro application: Bulk Upload Thick Client and CMIS Serivce 19](#_Toc14113808)

[4.2 Search Documents 20](#_Toc14113809)

[4.2.1 Basic Flow of Events 20](#_Toc14113810)

[4.2.2 Pre-conditions 21](#_Toc14113811)

[4.2.3 Kotak FileNet CMIS Search Interface 21](#_Toc14113812)

[4.3 IBM Content Navigator-viewer to view document 23](#_Toc14113813)

[4.4 Document Administrator 23](#_Toc14113814)

[4.4.1 Overview 23](#_Toc14113815)

[4.4.2 Navigator Desktop Configuration 23](#_Toc14113816)

[5. UPM System 24](#_Toc14113817)

[5.1 Overview 24](#_Toc14113818)

[5.2 Role Management 24](#_Toc14113819)

[5.2.1 Add Role 24](#_Toc14113820)

[5.2.2 Edit Role 25](#_Toc14113821)

[5.3 Manage Document Type 25](#_Toc14113822)

[5.4 Database Design for UPM 26](#_Toc14113823)

[6. Auditing & Retention Policy 27](#_Toc14113824)

[6.1 Auditing & Retention overview 27](#_Toc14113825)

[6.2 Configuring auditing 27](#_Toc14113826)

[6.3 Viewing audit information 27](#_Toc14113827)

[6.4 Managing the audit log 27](#_Toc14113828)

[6.5 Auditing & Retention in DMS 27](#_Toc14113829)

Document Control

**Revision History**

| **Version** | **Author** | **Issue Date** | **Summary of Changes** |
| --- | --- | --- | --- |
| 1.0 | Bhavik Vora | 11 Jul, 2019 |  |
| 1.1 | Bhavik Vora | 13 Jul, 2019 |  |
| 1.2 | Bhavik Vora | 15 Jul, 2019 |  |
|  |  |  |  |
|  |  |  |  |

References

| **Document Name** | **Additional Information** |
| --- | --- |
|  |  |
|  |  |

Glossary

| **Acronym/Term** | **Description** |
| --- | --- |
| ACE | Access Control Entries |
| ACL | Access Control List |
| CC | Cost Centre |
| CPE | Content Platform Engine |
| CTC | Cost to company |
| DMS | Document Management system |
| DOGJ | Date of group Joining |
| DOJ | Date of Joining |
| ERP | Enterprise Resource Planning |
| HR | Human Resource |
| ICN | IBM Content Navigator |
| LOB | Line of Business |
| LOC | Location |
| UPM | User Profile Management |

# Introduction

## Purpose

Purpose of this document is to provide the low level design details for Kotak DMS based on signed off SRS and HLD. This document elaborates the design aspect for each use case from HLD. Document provides details of configurations required for existing tool and design aspect of the customization. Intended audience for this document is technical team including developer and leads of the project. Technical people should refer to this document while developing the application. This document compromises of sample code snippet, configuration details and customization design details to achieve the business requirement.

This document detail out design aspect of every component used to build the DMS solution.

## Architectural representations

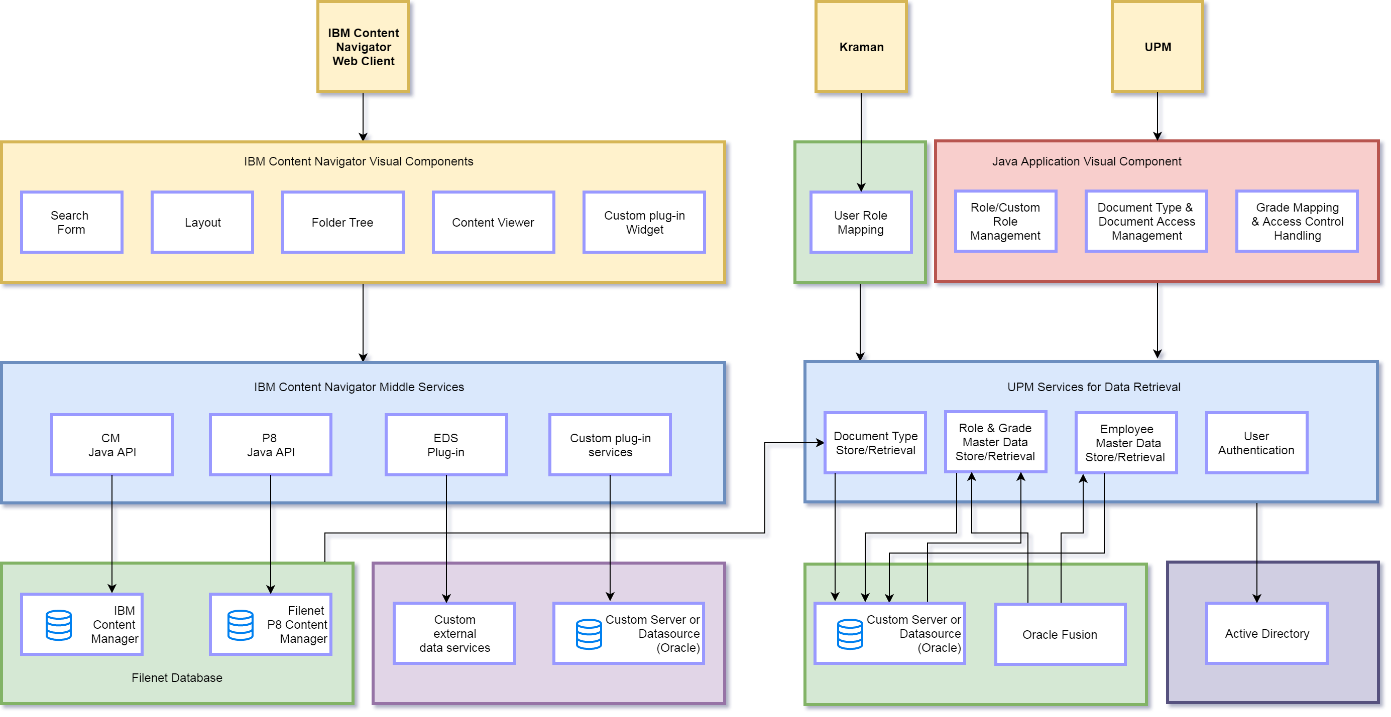
DMS is a group of components integrated tightly with each other. The below diagram is the detail view of FileNet P8 DMS components along with external applications used in Kotak DMS implementation

The front-end applications like Navigator, UPM and Bulk utility will invoke DMS functionality in their respective screens. Various DMS functionalities and integration with external applications are elaborated in details in the section **Content Design**

IBM Content Navigator(ICN) is the front end application provided to DMS Admin for handling document management functionality.

### Logical View

Figure 1: Logical View



# Content Design

This section provides design details of the FileNet content engine part. This section includes creating and configuring basic Filenet content components. Most of the activities in this section are one time activity over an environment.

## Domain, Site, Virtual Server and Server Instance

The FileNet P8 domain represents a logical grouping of physical resources (object store databases, full text index areas, file storage areas, and content cache areas) and Content Engine servers providing access to those resources. Each resource, and each Content Engine server, belongs to one and only one domain. A Content Engine server can access any resource in its domain, but cannot access any resource that lies outside of its domain.

A *site* represents a geographical location. All site resources are well connected via fast, reliable LAN. There is no functional limit to the number of sites that a single IBM FileNet P8 domain can contain.

A *virtual server* is the logical service point with which Content Engine clients interact. A virtual server can map to a single independent server instance or to a set of server instances. When a virtual server contains multiple server instances, client requests are load-balanced across the set of server instances through the J2EE application server’s clustering capabilities or through the use of a hardware load balancer that provides scalability and high availability.

A *server instance* is an individual J2EE application server instance. Multiple server instances (each running in their own JVM) can be hosted on a single physical server. Content Engine clients do not interact directly with a server instance. Logically, clients always go through a virtual server.

For Kotak DMS implementation, the FileNet P8 Domain will be a single domain for all Kotak FileNet users. There will be a Single site, single virtual server and single server instance within the domain. The FileNet P8 domain will be installed on the data centres of Kotak. FileNet P8 domain database server running on Oracle will be in the same P8 domain.

Table 1: FileNet Domain, Server and Site Details

|  |  |
| --- | --- |
| Component Name | Value |
| FileNet P8 Domain | KOTAKFNDomain |
| Site | Initial Site |
| Virtual Server | <Hostname>Node01 |
| Server | server1 |

## Object Store Design

An object store is a database repository for storing objects such as documents, folders and the metadata defining an object's classes and properties. Object Stores manages all documents and configuration data with regards to a P8 Content Engine. We recommend a single object store for all documents within the system for KOTAK.

The object store selection is purely based on KOTAK policy. Our recommendation is to have one object store for all documents with in KOTAK. The object store details are provided below

|  |  |
| --- | --- |
| **Object Store Details** | |
| Object Store Name | -HRDMS |

## Security Policies Object store level permission is given to below AD groups only,In order to use HRDMS Objectstore

|  |  |
| --- | --- |
| **Object Store Security** | |
| Use Object Store Permission | HR Users |
| Full Access Permission | HR Admins |
| Full Access with Modify System Properties | HR Super Admins |

## Storage Policies

A storage policy provides mapping to specific physical storage areas and is used to specify where content is stored for a class or object with content (for example, a document). Content Platform Engine supports the mapping of storage policies to one or more storage objects. Each storage policy can have one or multiple storage areas as its assigned content storage target.

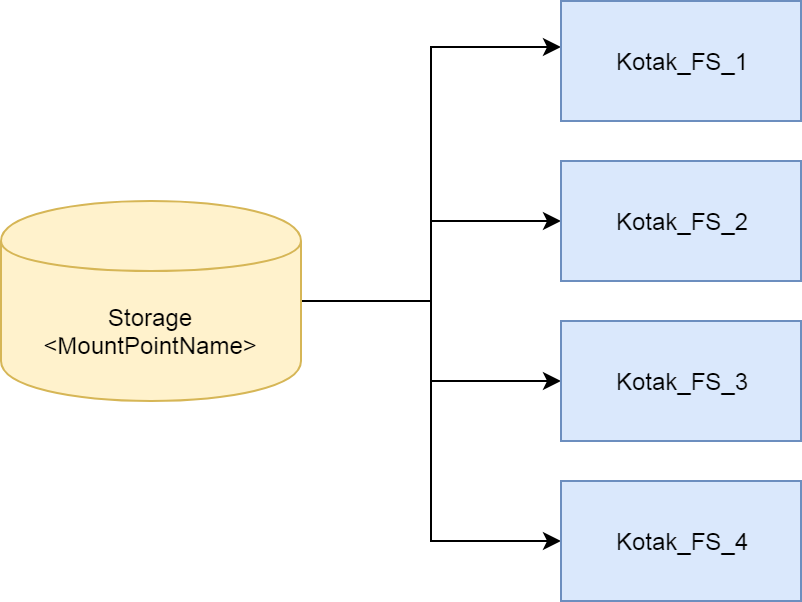
Storage farm should be created to increase the throughput by distributing the I/O load. Storage farm recommendations are as follows.

Multiple File Storage Area should be attached to one Storage Policy to perform Storage Farming and ease of maintenance activity.

Storage Area Name should be <MountPointName>/KOTAK\_FS\_1, <MountPointName>/KOTAK\_FS\_2, <MountPointName>/KOTAK\_FS\_3, <MountPointName>/KOTAK\_FS\_4 ...

These Storage Areas should be linked with one Storage Policy KOTAK\_SP\_1.

Figure 2: Storage Policy



This activity will be performed by DMS admin team with the help of storage and OS team on best approach basis.

## Property Templates

Property templates are used to index and search documents. Property templates are nothing but the attributes in the document. The following property templates will be defined in the FileNet P8 Content Engine. These property templates are going to be exposed on top level document class. It will be indirectly exposed to sub document class as well.

Table 2: Property Template Details

| **#** | **Property Symbolic Name** | **Property Name** | **Data Type** | **Cardinality** | **Description** |
| --- | --- | --- | --- | --- | --- |
| 1 | Employee Code | KDMS\_EmployeeCode | String | Single | Kotak Employee code |
| 2 | Document Status | KDMS\_DocumentStatus | String | Single | Document Status – it will be approved default |
| 3 | Active | KDMS\_IsActive | Boolean | Single | Indicates if document is active or not |
| 4 | Type of document upload | KDMS\_DocUploadType | String | Single | Upload type – it defines using which method doc is uploaded to DMS. |
| 5 | Classification | -Document Type | String | Single | Represents category of documents. |
| 6 | Document Type | -Document Sub Type | String | Single | Represents exact type of the document. E.g. PASSPORT,PAN. |
| 7 | Document Upload Date | KDMS\_DocumentUploadDate | DateTime | Single | Date when user uploaded the document from front end application. |
| 8 | Document Upload User | KDMS\_DocumentUploadUser | String | Single | UserID of the front end application user. |
| 9 | Document Unique ID | -Document Reference Number | String | Single | Unique Identification number associated with the document. |
| 10 | Document Comment | DocumentComment | String | Single | This field will be used to store the document comments provided by the front end user. |

## Document Classes

All Content Engine objects belong to a class. The class to which an object belongs determines its initial properties and behaviour. All objects of a class are identical in form and behaviour, but contain different values for their properties. Every Content Engine object has a property that describes the class to which it belongs, called the Class Description. Document Classes define the properties for identification of a document.

A parent *DMS\_HRDocuments* class will be created under Document class level. All common properties will be created under parent class. 7 child classes to be created under *DMS\_HRDocuments* class. Child classes will inherit parent class property. Following are the document classes to be created under root document class i.e. Document.

|  |  |
| --- | --- |
| Parent Class | Child Class |
| *DMS\_HRDocuments* | DMS\_ Compensation |
|  | DMS\_ DisciplinaryAction |
|  | DMS\_ ExitManagement |
|  | DMS\_JoiningDocuments |
|  | DMS\_Misc |
|  | DMS\_Nominations |
|  | DMS\_TestimonialVerification |

|  |  |  |
| --- | --- | --- |
| Document Class | Document Type | Document Sub Type |
| DMS\_ Compensation | Compensation | Additional NRA Letter |
| Appointment Letter |
| Group TRF Sheet |
| Offer Letter |
| Praise Documents |
| Salary Sip |
| Substitution Letter |
| Transfer Letter |
| Warning Letter Date |
| DMS\_ DisciplinaryAction | Disciplinary Action | KS Relieving Letter |
| Penalty Letter Date |
| DMS\_ ExitManagement | Exit Management | Full and Final Sheet |
| Recovery Letter |
| DMS\_JoiningDocuments | Joining Documents | AMFI Certificates |
| Basic Guidelines Documents |
| Commitment Letter |
| Compliance Form |
| Confirmation Letter |
| Employee Data Form |
| Employee Declaration Form |
| Interview Assessment Sheet |
| Medical Fitness Report |
| NCFM Certificates |
| PAN Card |
| Probation First Extension Letter |
| Probation Second Extension Letter |
| Resume |
| Social Media Code of Conduct |
| DMS\_Misc | Misc. | Employee Loan |
| Misc. Documents |
| DMS\_Nominations | Nominations | Declaration Form NEW FORM 11 |
| Gratuity Nomination Form |
| Insurance Form |
| PF Nomination Form |
| DMS\_TestimonialVerification | Verification Of Testimonial | Address Proof |
| Background Verification Form |
| Educational Certificates |
| Reference Check Report |
| Relieving Letter |
| Resignation Letter |
| Emergency Loan |
| Candidate Creation Form |
| Recruitment Requisition Form |
| Photo |
| Application for Identity Card |
| Self-Declaration |

**Document Class Security**



## Folder

Folders are logical repository used to group documents together. The documents could be from any document class. In our set up we propose to use the similar folder structure as the document class.

Two separate folders will be created for Entry template and Search temples for storing all the Entry Templates and Search templates.

The purpose of this folder provide the following benefits –

1. Easy to control access control list on basis of document category
2. Introduces ease of use while defining the search templates for business users

The folder structure is provided below.

**ROOT FOLDER**

**DMS HR Documents**

Compensation

Disciplinary Action

Exit Management

Joining Documents

Misc.

Nominations

Verification Of Testimonial

# Master Data & Document Upload Applications

This section provides details of UPM application design. This section will elaborates the design aspect of UPM application being developed to fulfil the business requirement captured in HLD.

## Master data lookup

This step is required to pull the master data stored in Kotak HRMS. UPM application are seeking data which will be provided by HRMS system and UPM application configure this data to its custom database.

### Master Data mapping from HRMS

Master data will get updated on daily bases in evening. A utility will fetch data from HRMS system. This utility will provide employee master data and role master data. UPM system utility will manipulate these data based on LOB-LOC-CC and will be stored to UPM Database.

Scheduler will be running every evening after HRMS system get updated. Kotak will provide the service required to pull the master data from HRMS.

Logs will be managed for each activity. If scheduler fails then it will re-initiate it again. On end of process a report will get generated for activity details.

* Every evening 7:30 PM scheduler will run. It will run on daily bases and perform its tasks.
* End of scheduler task logs will be generated.
* If scheduler is failed then it will be re-initiated after 15 min of failure automatically.
* This can be repeated for 3 times after that one email will be sent to admin so that they can check and track the issue.

Rollback strategy

* On failure of scheduler, all data dump will get rolled back to its previous state.
* No data will be committed till the time of successful data migration from HRMS Staging database to UPM database.

### User Role mapping from Kraman

Kraman is Kotak role and access allocation workflow system. This system generate user role mapping. Very first time UPM system will map all existing users and roles from Kraman system to UPM database.<TBD how to get this data?>

On update of any user role mapping Kraman will directly update that specific user role relation in UPM database. UPM will provide API to update this relation.

<Provide API name with param>

On creation of custom role UPM will push that role to Kraman system. If any user is assigned to this user it will get updated to UPM system using API.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Application Name | Master Lookup | Service Details | Input To Service | Response from the service |
| HRMS | Employee Master | <TBD> | <TBD> | <TBD> |
| HRMS | Role Master | <TBD> | <TBD> | <TBD> |
| Kraman | User Role Mapping | <TBD> | <TBD> | <TBD> |
| Kraman | Update User role mapping | <TBD> | <TBD> | <TBD> |

## Document Upload

UPM applications are designed to upload, process and store the document in FileNet repository. The document hierarchy describes the structure of the documents that your application is designed to process. There are three ways using which document will be uploaded to

1. **New employee on-boarding documents**

When new employee joins company, their details and documents are stored to Hire-pro system. Documents and meta-data from hire pro system will get uploaded manually by SMS team using DMS file upload service mechanism, this DMS File upload service uses CMIS service.

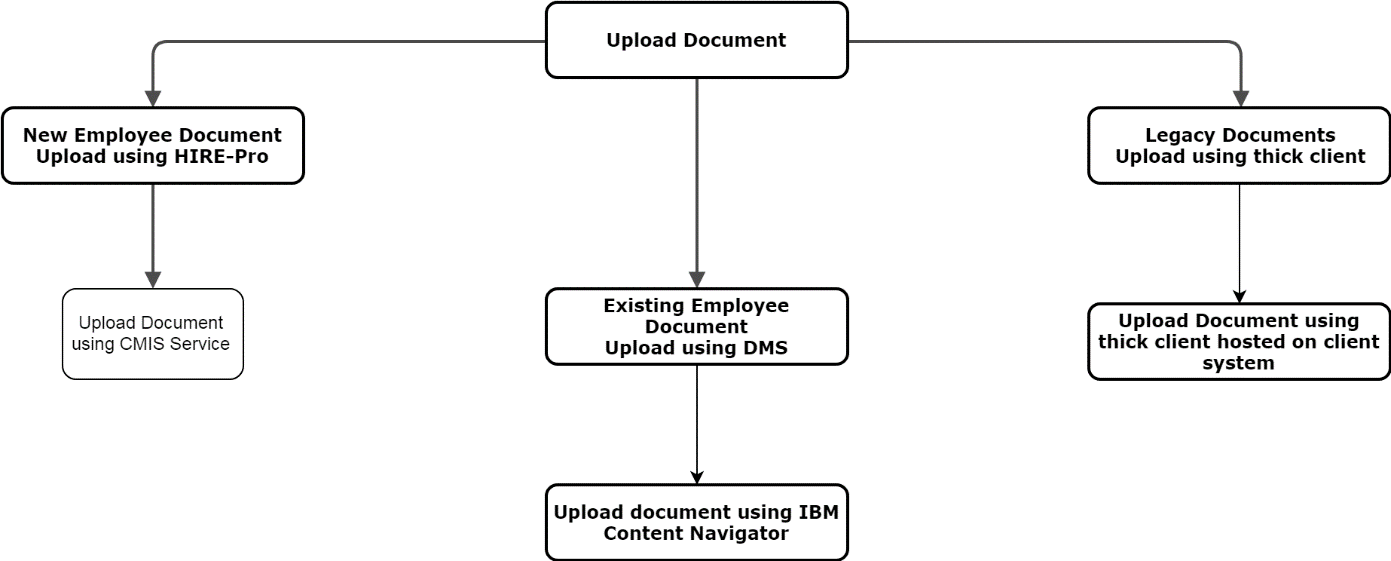
1. **Existing Employee Document**

Documents and meta-data from hire pro system will get uploaded manually using DMS file upload mechanism. Employee code will get verified at the time of document uploading. Using Entry template one can upload document to FileNet by utilising ICN functionality.

1. **Legacy Document**

Kotak Scanning vendor provides scanned documents along with its meta-data in csv format. There will be a thick client which will perform bulk document upload process by taking csv as a input and documents. During process of bulk upload it will track record of success and failure of document upload.

Figure 2 Below describes the Kotak DMS system document upload mechanism.



Kotak DMS solution will have 3 different applications as per the requirement received. **Table 3** below provides the application details.

Table 3: Kotak DMS Applications

| Sr. No. | LOB | Application Name | Description | Accepted File Type |
| --- | --- | --- | --- | --- |
| 1 | New Documents | CMIS web-service consumed by Hire Pro | CMIS web-service will be exposed to upload document on FileNet system. | 1. TIFF 2. PDF |
| 2 | Existing Documents | FileNet DMS | IBM Content Navigator application will be used to upload document to FileNet directly. | 1. TIFF 2. PDF |
| 3 | Legacy Documents | Bulk Utility thick client | This application will be used to upload bulk document by providing CSV on client system. | 1. TIFF 2. PDF |

### CMIS Web service Configuration

This section will provide the information on creating a CMIS Web service and configuring. CMIS basic configuration consisting of following tasks,

1. Create connection URL <connection url>
2. Set list of properties coming as metadata from hirepro
3. Set document Path
4. Create document using document creation permission.

**Table 4** below provides the application Object store, Connection url and authenticated user for development environment. Details will change environment to environment and will be mentioned in release management guide accordingly.

Table 4: Application Configuration

|  |  |
| --- | --- |
|  | Details |
| Object Store Name | <OBJECT STORE> |
| Connection Name | <URL> |
| Authenticated USER | <ALLOWED User> |

### Configuration for DMS Document Upload Feature

Configure DMS Document Upload Feature. Deploy custom DMS feature.

As Kotak will have many documents as unstructured documents, DMS applications for Kotak will be created with “DMS Document” entry template. Once application is deployed you can find the application in login features list.

<Add fig for displaying feature>

Kotak <Oracle> DB is identified as database for DMS. Administrator, Content Engine database should point out to <Oracle> database already existing in Kotak environment.

#### Configure Object Store & Storage Farm

Administrator will configure object store on existing Kotak Environment. Object store name <Object store> will be used for storing document, properties and retrieving documents.

<Fig to create Object Store>

Configure Storage farm for newly created object store. Apply basic security on Storage Farm and object store.

#### Create Property templates

Property are used to tag document uploaded to DMS. Property templates are going to be mapped with document classes. These property templates will be used to retrieve documents from DMS.

Property creation wizard need to be launch to create property template. Please check below wizard to create property,

<Fig to create property>

#### Create Document Classes

Once application is created, document class hierarchy needs to be created. Go to newly created Object store and create the document classes. Create document classes using Add Document class wizard. Refer to following fig.

<Fig for adding document class>

#### Create Entry Template

Once document classes are created document entry template need to be created. While uploading document need to open add document content item dialog. This requires document entry template which allows users to upload document using it. Refer to below fig to add entry template.

<Fig for adding entry template>

#### Validate Employee Code

Once document entry template is created, employee code field need to validate. On document upload wizard user need to add employee code. This employee code must need to be available in HRMS system. So validate employee code service will verify it and allow user to upload document if employee code is valid.

#### Configure Feature with Navigator Repository

With Navigator, users can upload, classify, and retrieve documents. The viewer can be split off to a separate window and displayed on a separate monitor to improve productivity and ease of use.

Create following navigator desktop for Kotak DMS project

Desktop Name – **HRDMS**

Configure KDMS Feature in features section

### Bulk upload thick client utility

With Kotak DMS system providing system which enables bulk upload document utility. Bulk upload utility allows users to upload existing Kotak documents to Kotak FileNet repository.

Thick client utility will be installed on authorised users system. This is executable desktop based utility communicating with FileNet using CMIS service to upload documents and meta-data. Documents uploaded via this utility are approved documents.

<Fig of thick client>

Bulk upload utility takes CSV file as input. CSV file will have following details,

* Document Name
* Employee code
* Document Classification
* Document Class Name
* Document full path

Other information will be captured automatically

* Document creator
* Document creation date-time
* Document Status (Default value - Approved)

Bulk utility will generate report for its success and failure document uploading scenarios. On completion of utility process it will provide report of it. This report will be having same information as available in CSV along with new Columns

* Status (Success - Fail)
* Comments (If fail then reason of failure will be there)

<Add csv document format>

#### Validate Employee code

Bulk upload utility captures each record of csv and identifies its employee code. This employee code must need to be available in HRMS system. So validate employee code service will verify it and utility to upload document if employee code is valid.

#### Configure thick client to upload document

Bulk utility will communicate with FileNet via CMIS services to upload document and meta-data. Connection URL will be configured with bulk utility. Bulk utility will take connection URL at initialization time and it can be configured by admin user if need to be changed later on.

<fig for Connection url config>

# Document Services

This section provides the details of design aspect for Kotak DMS use cases captured in HLD. This section covers development configuration aspect of each use case.

FileNet Content Engine will store contents associated with all Kotak Line of business. These documents can be related to New employee, existing employee etc. Users can Add, View, search documents to DMS via front end application Navigator.

IBM Content Navigator (ICN) is the front end application provided to DMS Admin for handling document management functionality

Various DMS Integration touch points are elaborated in section below.

## Add Document

The Add Document functionality will be triggered by front-end applications. The below front-end applications will trigger the add document function to upload the document into DMS.

1. DMS
2. Bulk Utility Thick client
3. CMIS Service for Hire Pro

### Document addition from front-end application: DMS

User will perform the following actions from Front End application to add a document into DMS:

1. User will login to the front-end application (IBM Content Navigator) and go to KDMS desktop where DMS feature is configured.
2. User will select DMS feature and go to upload document section.
3. User will provide meta-data: Employee Code and click on upload document button.
4. User will prompt with Document upload dialog which is prefilled with Employee code
5. User will browse the destination folder to select corresponding file.
6. User will press Add button to add the document in DMS.
7. User will see uploaded document to documents list.

<Fig DMS system>

### Document addition from Hire Pro application: Using CMIS Service

IBM Content Collector for File System will be used to add documents in to FileNet DMS. IIB will place the Content File and respective metadata File (in XML format) in a file share to be picked by ICC task route Engine.

For each ETLI Line of Business, separate Task routes will be created. Each task route will be monitoring different source Folder for the Content and Meta Data File.

<TBD Add service url and params>

### Document addition from Hire Pro application: Bulk Upload Thick Client and CMIS Serivce

Kotak scanning vendor provides CSV which will contain meta-data Employee code, document title, document type, document full path.

Bulk utility client will be installed on user environment. Connection URL will be configured in this thick client. Below is csv file format for bulk upload.

<Add csv file>

Bulk utility will be generating reports on end of process. In output report csv file two columns status and comment will be added. If an error occurs during processing in the main task route, the affected object will be marked as failed in status column along with reason in comment column. File which is failed to upload then again it can re-initiate to upload file by resolving issue based on comments.

<Fig bulk upload utility>

User will perform the following actions from Bulk Utility application to add a document into FileNet:

1. Admin will configure Connection URL.
2. User will login to Bulk upload utility. Authentication will be managed with Active directory.
3. On successful login user can see connection url and object store in read only mode.
4. User will see browse button to select input csv file.
5. User click on process button and it will start import document from define path in CSV and upload it to Filenet along with meta-data.
6. File upload in FileNet will happened with CMIS service.
7. End of process csv report will be generated.

## Search Documents

The search Document functionality will be handled by the IBM Content Navigator DMS feature.

### Basic Flow of Events

User will perform the following action to search document(s) in DMS

1. The user will select the search criteria for the document search i.e.
   * Employee Code
   * LOB-LOC-CC / Responsibility name
   * DOJ/DOGJ
   * Last Date
   * Status
2. User will press the search button in the DMS applications
3. Search Request will trigger along with
   * Logged in user id
   * Logged in user role
   * Search criteria
4. Custom filter will get called to get document.
5. Depends on user role document will retrieve. Role is mapped with Grade Level, Access Control, and Document type.
6. Search filter will send list of requested document along with the document Id and version series Id for each document as response to the DMS application.
7. Based on user access control, logged in user can View, Update, Print, and Download.
8. Print and download restrictions are manually managed where as other restrictions are used managed using FileNet
9. Documents restricted with Grade level and HR Exceptions will not be retrieved based on user role.

### Pre-conditions

The search criteria will be selected by the user.

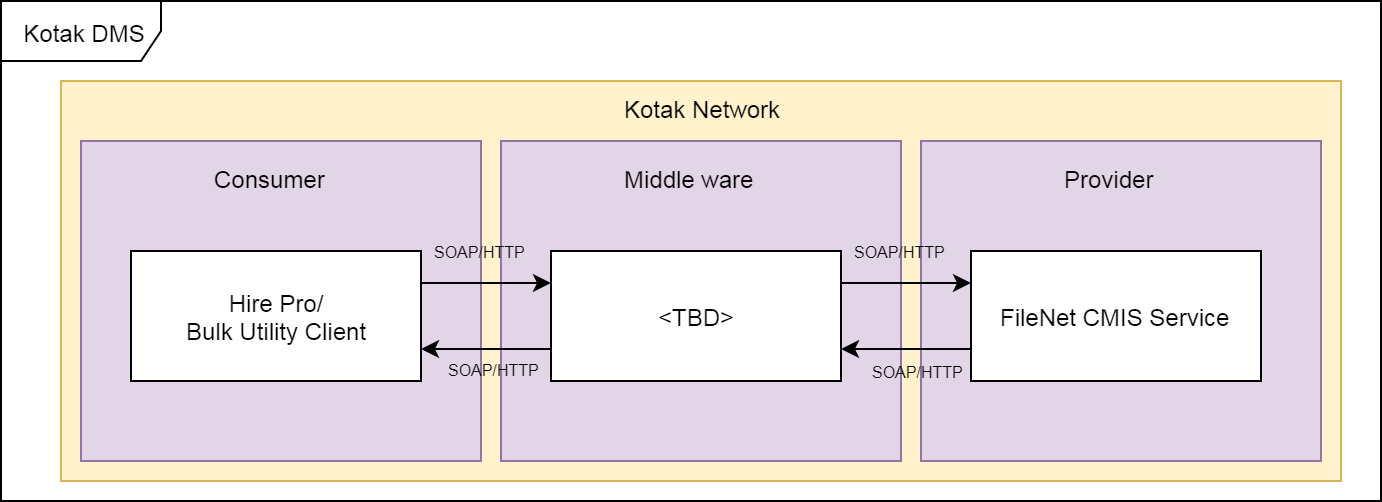
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Employee Number | LOB&LOL&CC | DOJ | DOGJ | Last Date | Active Status | Valid Search |
| Enter |  |  |  |  |  | **Yes** |
|  | **Enter** |  |  |  |  | **Yes** |
|  |  | **Enter** |  |  |  | **No** |
|  |  |  | **Enter** |  |  | **No** |
|  |  |  |  | **Enter** |  | **No** |
|  |  |  |  |  | **Enter** | **No** |
|  | **Enter** | **Enter** |  |  |  | **Yes** |
|  | **Enter** |  | **Enter** |  |  | **Yes** |
|  | **Enter** |  |  | **Enter** |  | **Yes** |
|  | **Enter** |  |  |  | **Enter** | **Yes** |
|  |  |  |  |  |  |  |

### Kotak FileNet CMIS Search Interface

Content Management Interoperability Services (CMIS) standard is a uniform means for applications to work with content repositories. CMIS will be installed and configured in IBM Content Navigator Application in Kotak FileNet environment.

#### CMIS Component Diagram

Figure 2: CMIS Component Diagram



#### Service name: query

This service is called to search document(s) from repository (FileNet DMS) based on the search Criteria. This web service is configured for ***HTTP basic authentication***.

Parameters passed are:

1. repositoryId
   1. repository name– Always pass **KOTAKOS**
2. statement
   1. statement– Typically the document search Criteria to be passed to retrieve documents

#### WSDL URL:

**FileNet:**

**http://<FileNet CMIS Host>:<port>/fncmis/wsdl**

**Operation Name –** query

**User name:** <FN\_UserID>

**Password:** <FN\_Password>

**Domain: <FN\_DomainName>**

#### Error Handling with SOAP Fault

If an error occurs during processing, the response to the CMIS SOAP request message are handled using a specialized envelope known as a Fault Envelope. FileNet CMIS service constructs a SOAP Fault and sends back to the sender of the SOAP message.

The SOAP fault mechanism returns specific information about the error, including a predefined code, a description, and the address of the SOAP processor that generated the fault

Sub-elements of Fault:

The SOAP Fault has the following sub elements

|  |  |
| --- | --- |
| Sub-element | Description |
| <faultCode> | It is a text code used to indicate a class of errors. See the next Table for a listing of predefined fault codes. |
| <faultString> | It is a text message explaining the error. |
| <detail> | It is an element used to carry application-specific error messages. The detail element can contain child elements called detail entries. |

SOAP Fault Codes

The **faultCode** values defined below will be used in the faultcode element while describing faults.

| Error | Description |
| --- | --- |
| soapenv:Server | There was a problem with the CMIS server, so the message could not proceed |
| soapenv:Client | The message was incorrectly formed or contained incorrect information |
| soapenv:VersionMismatch | Found an invalid namespace for the SOAP Envelope element. |
| soapenv:MustUnderstand | An immediate child element of the Header element, with the mustUnderstand attribute set to "1", was not understood. |

Attached the sample SOAP error message for reference.



## IBM Content Navigator-viewer to view document

 IBM Content Navigator viewer for viewing the documents which are stored in the FileNet DMS. The supported image file types for IBM Content Navigator Viewer are as below.

1. **TIFF 6.0**
2. **BMP**
3. **GIF, JPEG** and **JPG, PNG**
4. **PDF**

 If user wants to view image file types other than the mentioned in the above list, users will be prompted to download the image in to the local desktop to view.

For applications that choose to use the IBM Content Navigator viewer to view documents, the supported way to instantiate the viewer is to obtain a user session through SSO solution

## Document Administrator

### Overview

DMS admin UI will be provided to DMS admins for document management.

1. Add
2. Search,
3. View
4. Browse

### Navigator Desktop Configuration

Navigator UI will be provided to DMS admin for performing administration activity. A navigator desktop will be created for DMS Admin use. Authorized DMS admin will have access to this desktop. Following are the details for creating the desktop for DMS admin.

1. Repository Name: **HRDMS**
2. Desktop Name: **KOTAKADMIN**
3. Display Features to Select: **Home, DMS, Browse and Search**
4. Desktop URL = **http://<Server Name>:<Port>/navigator/?desktop=KOTAKADMIN**

Following groups will have access on the -KOTAKADMIN desktop.

1. <TBD>HR ADMINS

# UPM System

## Overview

This section will provide information about UPM System, where users are managed. User Profile Management system is responsible for mapping users with role, accessibility, document restriction and document types.

Kotak HR users are having different access control on different level of grades. This grades and mapping of documents need to be configurable with each role. When any user request for access to document retrieval this role mechanism identifies which user will be having access on documents to what level.

UPM system gets data dump from HRMS system and manipulates those data for UPM use. UPM system also interact with Kotak Kraman System to get user and roles mapping. This mapping is stored to UPM system.

Flow of master data mapping to UPM system

* Every evening time (Exact Time) HRMS system dump data to Stagging server of Kotak, UPM will retrieve this data.Need to mention trigger for this scheduler or fix time
* UPM system access Employee Master data and HR-Employee mapped Master data.
* UPM system manipulates data and Map Employee and HR employee data with respect to Responsibility name.

## Role Management

Roles are used to manage HRMS users document access restriction, document orations mapping and document type mapping. Roles are mapped with Document types/subtypes, Grades, Operation permissions, Status (Active/in-Active) and end date of role.

Flow of defining and mapping roles,

* 1. Authorised user will login to UPM system
  2. User selects Role Management
  3. List of available roles will be displayed in Table which are already added to system.
  4. User can add new Role to UPM system by clicking on Add Role button.
  5. User can edit existing role by clicking on edit action in list of Roles table

<Fig of Role Screen>

### Add Role

Steps to add roles to UPM,

* 1. On click of Add role button, role add screen appears along with option to map role with Grade, permissions, status, document types/subtypes.
  2. Select Role from role dropdown.
  3. Existing roles are getting populated from Master data table
  4. If user selects Other Role then ask user to add Custom Role name which is not available in System, user can not add same name role which is already exist in system.
  5. Select Grade for which role is allowed to access employee document. On retrieval of document system will check that logged in user role is having access to specific Grade document or not.
  6. Select Document type/subtype. This will indicate that role can only access selected document type/subtype of documents.
  7. Role is mapped with access permission, user will select checkbox of permissions.
  8. Role need to have end date, so after that date that role will not be active. If there is no end date specified that means this role will be having long duration active state.
  9. Select role status and save it.

On performing this process role will be list out in Role table. All necessary information will be displayed in Role table.

<Fig to add role>

### Edit Role

Roles created are listed in role table. When there is required to change any mapping done with roles then user will be having provision to edit and update roles details.

Steps to edit role in UPM,

1. User select role which need to edit by clicking on action icon of edit in role table
2. Role edit screen will appear
3. User can edit all mapping here
4. Save role after editing details

<Fig to edit role>

## Manage Document Type

Document types used to classify documents entered to DMS. When documents are created documents types are mapped with it. UPM provides document type management. Document types are created in UPM system to FileNet system.

<Fig to Manage Document type>

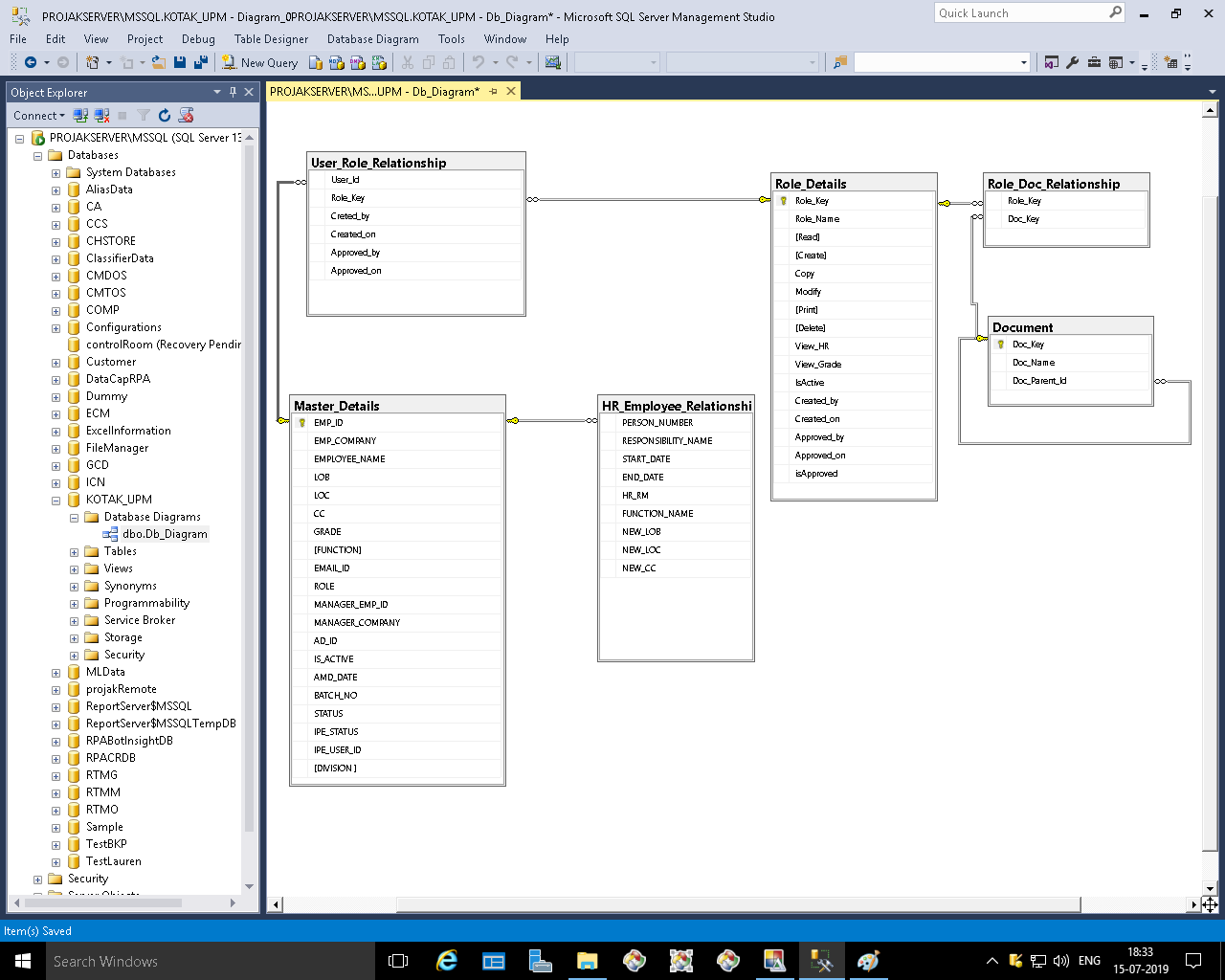
Flow of defining Document Type

1. Authorised user will login to UPM system
2. User select Document Type Management
3. List of Documents types and sub types are listed in table.
4. To create new Document type user can click on Add Document type button
5. On click of add document type user will navigate to add document type page
6. User write document type name
7. User selects any document type if newly adding document need to be set as child of that document type
8. User can’t add document type which is already available
9. Select document type status and save it

<Fig to add document type>

On creation of document type/subtype service will be called to create document type in FileNet system. On success of document type creation in FileNet that document type will displayed in document type list table.

## Database Design for UPM



# Auditing & Retention Policy

## [Auditing & Retention overview](https://www.ibm.com/support/knowledgecenter/SSNW2F_5.1.0/com.ibm.p8.ce.admin.doc/audit/al_overview.htm?view=kc)

Content Engine auditing is accomplished by recording custom or system events that occur on an object. When an audited event occurs, Content Engine creates audit entries that are stored in an audit log in the object store database.

You can implement data protection and long-term retention by setting retention policies and holds on documents. Retention policies and holds are important components of an overall information lifecycle governance program.

Retention controls deletion by setting a date and time before which the object cannot be deleted. Although you can set and modify the date on each object individually, more typically retention defaults according to a policy expressed at the class level.

## [Configuring auditing](https://www.ibm.com/support/knowledgecenter/SSNW2F_5.1.0/com.ibm.p8.ce.admin.doc/audit/al_config.htm?view=kc)

To audit changes to objects in an object store, you must enable auditing on the object store, configure audit options for each class you want to audit, and optionally configure auditing for individual properties.

## [Viewing audit information](https://www.ibm.com/support/knowledgecenter/SSNW2F_5.1.0/com.ibm.p8.ce.admin.doc/audit/al_view.htm?view=kc)

You can view the audit entries for an object by viewing the object properties or by querying the audit log. From the audit log, you can view the audit entries, export the audit entries to XML for reporting and other purposes, and administer the audit log.

## [Managing the audit log](https://www.ibm.com/support/knowledgecenter/SSNW2F_5.1.0/com.ibm.p8.ce.admin.doc/audit/al_manage.htm?view=kc)

By default, audit entries remain in the audit log even if the audited object is deleted. Use audit disposition to control the size of your audit log when you enable auditing for prolonged periods or if you save copies of the original or modified objects to the audit log. Configuring audit disposition involves creating and enabling one or more audit disposition policies for each object store, creating an auditing configuration for the domain or site, and managing unused bookmarks.

## [Auditing](https://www.ibm.com/support/knowledgecenter/SSNW2F_5.1.0/com.ibm.p8.ce.admin.doc/audit/al_manage.htm?view=kc) & Retention in DMS

Enable auditing for every documents uploaded via CMIS service or Bulk utility. On upload of document via bulk upload manage event handler to create document, it will get logged to track uploaded document.

Enable auditing for DMS upload mechanism which will be using inbuilt document creation process and track down for its success and failure scenario.

To modify the retention setting of an object, you must first modify the retention permission. To change the retention setting of an existing object:

1. In the administration console, access the retention settings for object.
   1. In the domain navigation pane, select the object store.
   2. In the object store navigation pane, select Browse > Root Folder.
   3. Click the Root Folder or a folder under the Root Folder. Open a folder if you are accessing a document or custom object.
2. Click the object and select the Retention tab. Set retention for the object. The existing retention period can only be extended. You cannot set the retention date to an earlier date than the current setting or to no retention.