





Mount
Sinai



O **ORACLE**[®]
Consulting **TCM+**

OCS

MSHS OIC PaaS Migration – Integration Technical Design Document



Table of Contents

Document Control	5
Change Record	5
Reviewers	6
GLOSSARY	6
Purpose	8
Scope	8
Assumptions	8
Application Flow	9
HCM Common Outbound Framework (HFWK_OB).....	9
EDMCS Common Inbound Framework (EDMCS_IB).....	9
HCM Common Inbound Framework for PaaS Extension Applications (HFWK_IB)	9
HCM Common Inbound Framework for External/Vendor Integrations (HFWK_IB)	10
Approach	10
Integration	11
Architecture	11
Design Patterns	11
High Level Integration Design	12
HCM Common Outbound Framework (HFWK_OB).....	12
EDMCS Common Inbound Framework (EDMCS_IB).....	13
HCM Common Inbound Framework (HFWK_IB) – PaaS Applications	15
HCM Common Inbound Framework (HFWK_IB) – External/Vendor Integration	17
Bank Integrations	19
Detailed Integration Design	21
HCM Common Outbound Framework (HFWK_OB).....	21
EDMCS Common Inbound Framework (EDMCS_IB).....	22
HCM Common Inbound Framework (HFWK_IB) – PaaS Applications	23
HCM Common Inbound Framework (HFWK_IB) – External/Vendor Integration	24
Design Considerations	25
Extract Program Parameters	26
Outbound Data Format	26
Prerequisites/Functional Setup Requirements	26
Functional Test Scenarios	27



Technical Solution Overview & Components.....	27
Summary	27
Solution Components.....	30
OIC Components	30
Integrations	30
Connections.....	32
Lookups	33
Certificates	33
OCI Components	33
Object Storage/Buckets	33
Integration Logic	34
HCM Common Outbound Framework (HFWK_OB).....	34
HFWK_OB_LAUNCHER_SERVICE	35
HFWK_OB_EXTRACT_SERVICE.....	35
HFWK_OB_POLLING_SERVICE	35
HFWK_OB_DISPATCHER_SERVICE.....	36
HFWK_OB_NOTIFICATION_SERVICE	36
EDMCS Inbound Framework (EDMCS_IB)	36
EDMCS_IB_LAUNCHER_SERVICE	37
EDMCS_IB_PAAS_CNTRL_SERVICE	38
EDMCS_IB_PAAS_FILE_PROCESS_SERVICE.....	38
EDMCS_IB_PAAS_ERP_FILE_PROCESS_SERVICE.....	38
HCM Common Inbound Framework (HFWK_IB) – PaaS Applications	39
HFWK_IB_LAUNCHER_SERVICE	39
HFWK_IB_PAAS_TO_HCMCLOUD	40
HFWK_IB_ORCHESTRATE_LOAD_HDL_PROCESS	40
HCM Common Inbound Framework (HFWK_IB) – External Integrations	40
HFWK_IB_EXTERNAL_INTF_LAUNCHER_SERVICE	42
HFWK_IB_LIST_FILE_SERVICE	42
HFWK_IB_DISPATCHER_SERVICE	42
HFWK_IB_DECRYPTION_SERVICE	43
HFWK_IB_ENRICHMENT_SERVICE.....	43
HFWK_IB_CLOUDFILE_ORCHESTRATE_SERVICE	44
HFWK_IB_HDL_DATA_SET_STATUS_SCHEDULER.....	44
HFWK_IB_HDL_DATA_SET_STATUS_PROCESS	44
HFWK_IB_HDL_DATA_SET_STATUS_SUPP_PROCESS.....	45
Bank Integrations Outbound	45
HFWK_OB_PAYROLL_SERVICE	45
HFWK_OB_ENCRYPTION_DECRIPTION_SERVICE.....	45
HFWK_OB_POSITIVE_EFT_LAUNCHER	46
Fund Number Approvers Synchronization Process	46
HFWK_LOAD_FUND_APPROVERS_LAUNCHER.....	46
HFWK_LOAD_FUND_APPROVER_SERVICE	46



HFWK_LOAD_FUND_APPROVERS_NOTIFICATION_SERVICE	46
Error Handling & Logging	47
HFWK_DATASYNC_NOTIFICATION_SERVICE	47
HFWK_DATASYNC_LOGGING_SERVICE	47
HFWK_DATASYNC_DELETION_SERVICE	48
Other Integrations	48
HFWK_INTERFACETABLE_DATA_PURGE_SERVICE	48
HFWK_DATASYNC_UPDATE_SCHEDULE.....	48
Security.....	48
ATP DB Components	51
SaaS Components – BIP/HCM Extracts	53
User and Security	53
Open Closed Issues.....	54
Open Issues	54
Closed Issues.....	54
Appendix – A	55
OIC File Server Directory Structure – External Integrations	55
OIC File Server Directory Structure – Fund Number Approvers Sync Process.....	55
OIC File Server Directory Structure – Integrations for PaaS Applications	55
Appendix – B.....	56
Release Management - OIC Artifacts Deployment Automation.....	56
Appendix – C.....	56
Integration Testing Documents.....	56
PaaS OB Integrations Testing Document	56
PaaS Vendor IB Integrations Testing Document	56
PaaS IB integrations Testing Document	57
EDMCS IB Integrations Testing Document.....	57
Fund Approver Sync Process Testing Document	57
Bank Integrations Testing Document.....	57
Appendix – D	57
OIC Integration Job Schedules	58
Appendix – E	58
OIC Storage Bucket Configuration.....	59
Appendix – F	62
OIC File Server/SFTP Accounts Security Setups	63



Appendix – G	63
OIC Certificates Setup for PGP Keys	63
Appendix – H	63
Documentation	63

Document Control

Change Record

Version	Date	Author	Summary of Changes
1	29-Jul-2022	Venkat Ramakrishnan	Initial Draft
2	02-Aug-2022	Soumya Samantaray	Updates
3	05-Aug-2022	Chandan Bhomkar	Updates
4	11-Aug-2022	Venkat Ramakrishnan	More updates to the design, and architecture
5	16-Aug-2022	Kazi Rahaman	Updates
5	18-Aug-2022	Kallesh Basavarajappa	Updates
6	26-Aug-2022	Venkat Ramakrishnan	Reviews, and updates
7	30-Aug-2022	Venkat Ramakrishnan	Added bank integrations testing documents
8	26-Sep-2022	Venkat Ramakrishnan	Additional updates to documents in Appendix sections.
9	27-Sep-2022	Venkat Ramakrishnan	Added a new integration for EDMCS (198A10)



Version	Date	Author	Summary of Changes
10	29-Sep-2022	Venkat Ramakrishnan	Added configuration document for OIC Certificates
11	07-Oct-2022	Venkat Ramakrishnan	Updates for daylight/wintertime zone changes in meta data tables for OIC automated schedules using an integration.
12	18-Oct-2022	Venkat Ramakrishnan	Updates for testing documents. This is for the new EDMCS input screen to take the interface ids and for the bank integrations to accept the flow instance id using a text box.
13	20-Oct-2022	Venkat Ramakrishnan	Updates to the document after the new wrapper service was added to process the bank integrations. Also, changes made to EDMCS launcher service to accept the interface id as input parameter.

Reviewers

Reviewer	Date	Role / Responsibility
Venkat Ramakrishnan		Cloud Architect, OCS
Devesh Misra		Solution Architect, MSHS
Seung Yoo		Project Manager, MSHS
Sheetal Prasad		Chief Enterprise Architect, OCS
Vish Pattanashetty		Project Manager, OCS
John Metzger		Project Manager, Accenture

GLOSSARY



Abbreviation	Definition
HFWK	HCM Common OIC Framework
BIP	Business Intelligence Publisher
PaaS	Platform as a Service
SaaS	Software as a Service
UCM	Universal Content Management
OOTB	Out of the Box
OIC	Oracle Integration Cloud
JSON	Javascript Object Notation
ESS	Enterprise Scheduler Service
ATP	Autonomous Transaction Processing
OCI	Oracle Cloud Infrastructure
XREF	Cross Reference
HCM	Human Capital Management
MSHS	Mount Sinai Hospital Services
EDMCS	Enterprise Data Management Cloud Service
SOA	Service Oriented Architecture
TDD	Technical Design Document
VBCS	Visual Builder Cloud Service
IB	Inbound
OB	Outbound
JPMC	JP Morgan Chase Bank
HDL	HCM Data Loader
FTP	File Transfer Protocol
CI/CD	Continuous Integration, and Continuous Deployment
OCI	Oracle Cloud Infrastructure



Purpose

This document describes the integration architecture, and the technical design for the MSHS PaaS Integration framework using OIC. The document is broken down into following main sections.

1. Integration Architecture, and design for the MSHS OIC PaaS migration efforts. Following items are covered.
 - a. Common outbound framework to integrate:
 - i. Oracle HCM SaaS master, and application data into Oracle PaaS ATP layer.
 - b. Common inbound framework to integrate:
 - i. EDMCS SaaS master, and application data into Oracle PaaS ATP layer.
 - c. Common inbound framework to integrate:
 - i. MSHS PaaS custom Payroll extension applications with Oracle HCM Cloud.
 - ii. External vendor data such as employee, assignments, and payroll with Oracle HCM Cloud.
2. High level design flows for outbound, and inbound integrations.
3. Detailed flows for outbound, and inbound integrations.
4. Technical solution to implement the HCM inbound, outbound framework, and external vendor integrations with HCM.

Scope

The scope of the document is to cover the architecture, design, and implementation details for the MSHS OIC PaaS migration implementation with respect to integration of data with Oracle HCM cloud, Oracle EDMCS, and custom payroll extension applications. This document supports the delivery of a successful business transformation by making use of a generic design, and a reusable framework to synchronize data into to Oracle PaaS ATP Database, and to integrate HCM SaaS application with internal and external applications.

This document covers only the implementation details for the MSHS PaaS OIC framework. For some technical components that are reused in the OIC framework, appropriate TDDs that were created for the existing SOA framework should be referred. The implementation details will **not** be covered in this document for the following components as they are already covered in the TDDs for the existing SOA framework.

- The meta data design configurations, DB objects, and PL/SQL packages used by the OIC framework. Any minor tweaks to these components will be addressed in this document.
- The BI reports, and HCM extracts used by the OIC framework.
- Service accounts used for Oracle HCM, and EDMCS application integrations.

Following spread sheet contains the list of integrations for the OIC PaaS migration activities.

MSHS_Integ_Catalog.xlsx

Assumptions



- The OIC inbound, and outbound frameworks use the same ATP meta data design, and configurations used by the existing MSHS SOA framework.
- Minor tweaks are needed for object storage configuration, and meta data that is used by the SOA framework.
- OIC adapter configurations used for the integrations to connect to different applications must exist in OIC.
- Required application roles for the Oracle SaaS service account to run the BIP reports, and HCM extracts must be assigned.
- Object Storage/buckets setups are done in OCI for the MSHS PaaS OIC framework.
- All data objects such as staging, interface tables, PL/SQL packages, synonyms and sequences are available in ATP DB layer.
- Flow extracts, and BI reports needed for integration are configured in Oracle HCM Cloud.
- All functional configuration, and roles should be setup in Oracle HCM, and EDMCS cloud.

Application Flow

This section covers the application flow for four main OIC frameworks.

HCM Common Outbound Framework (HFWK_OB)

- Meta data is configured in ATP DB to run the HCM extract jobs, and BI reports at specified intervals.
- Staging, and transaction tables are defined in ATP DB layer.
- HCM extract is defined in Oracle HCM SaaS application to generate the report data for synchronization.
- HFWK_OB orchestrates a web service call to extract the data.
- The output of HCM extracts is sent to UCM.
- HFWK_OB checks the data extraction, downloads the file from UCM, and upload the data in object storage buckets.
- HFWK_OB framework orchestrates the DB stored procedure to load the data from object storage into staging tables, and to process into interface tables.
- HFWK_OB framework sends notifications to the email group for errors or success scenarios by reading the metadata.

EDMCS Common Inbound Framework (EDMCS_IB)

- Meta data is configured in ATP DB layer to extract data for EDMCS entities.
- EDMCS launcher integration is run at scheduled intervals to extract data from EDMCS.
- EDMCS integrations creates the control table record in PaaS ATP DB tables for the extract job that needs to be run.
- EDMCS polling service checks data extraction completion status, and updates control tables in ATP DB layer.
- EDMCS dispatcher integration is called after the data extraction is completed to retrieve the data from EDMCS, and to orchestrate DB stored procedures to load the data in interface tables.

HCM Common Inbound Framework for PaaS Extension Applications (HFWK_IB)



- Data is created by PaaS VBCS Payroll extension application in transaction tables.
- HFWK_IB framework polls for transaction tables, orchestrates the DB stored procedure to prepare the data file.
- DB stored procedure prepares the data files, and uploads to object storage.
- HFWK_IB reads the data from object storage and uploads the data in UCM. Initiates the HCM import job with appropriate parameters for the data load into HCM Cloud.
- HFWK_IB polls at frequent intervals for the data load and updates the status of load and import in PaaS ATP tables.

HCM Common Inbound Framework for External/Vendor Integrations (HFWK_IB)

- Files to be processed are pushed to the corresponding OIC file server directories used for various external integrations.
- HFWK_IB polls for the data files from OIC file server location.
- HFWK_IB decrypts the data as needed using the keys setup as certificates in OIC.
- Decrypted data is further enriched in the ATP DB layer as needed.
- HFWK_IB orchestrates DB stored procedures to prepare the data files in the ATP DB layer, and files are loaded to object storage.
- Files are read from object storage, zipped, and the import process is called by the OIC common inbound framework into HCM.
- Import status is updated by HFWK_IB in PaaS ATP DB tables after the load and import is completed.
- Notifications are sent to the appropriate email groups by HFWK_IB.

Approach

This section describes the integration patterns used for the OB, and IB framework.

1. Using HCM Extracts, BI Reports for PaaS OB Integrations:

- For PaaS Outbound Integrations (OBs), data extract jobs are configured in Oracle HCM Cloud, and run through flow action web service by the OIC common OB framework. using the asynchronous integration pattern.
- BI reports that are configured in Oracle HCM Cloud are used to check the status of HCM extract jobs for completion using the synchronous pattern.
- Data files from HCM extracts for PaaS OB integrations are sent to Oracle UCM, and subsequently loaded into object storage by OIC common OB framework.
- Data files are loaded into ATP DB interface tables by PL/SQL stored procedures from Object storage.

2. Using EDMCS REST Services for EDMCS OB Integrations:

- Applications, and Dimension data from EDMCS is extracted using the EDMCS REST services.
- Data files from for EDMCS IB integrations are loaded into object storage by OIC common EDMCS framework.
- Data is loaded by PL/SQL stored procedures from Object storage.



3. Using OIC HCM adapter to load, and import the data in Oracle HCM Cloud for IB integrations:

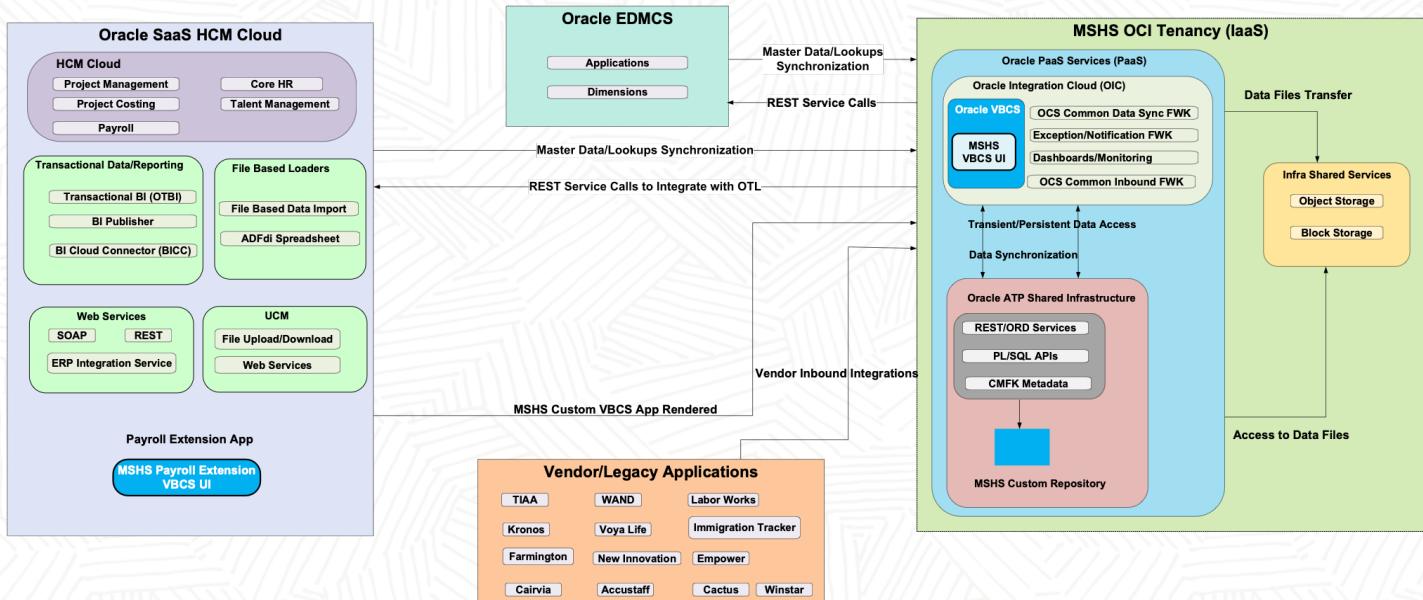
- Data files are generated using stored procedures in ATP DB layer and uploaded to object storage.
- HCM adapter is used to load the data from object storage into UCM, and to import the data into Oracle HCM Cloud asynchronously using import jobs.
- Stored procedure will update the status of import into ATP DB tables by running the BI report synchronously.

Integration

Architecture

Following diagram illustrates the Integration architecture diagram.

MSHS Technology Architecture



Design Patterns

Following are the three integrations patterns illustrated in the following diagrams.

- **OIC Common Outbound Framework (HFWK_OB)** - Asynchronous design pattern to synchronize data in Oracle PaaS ATP DB layer from Oracle HCM Cloud SaaS application.
- **EDMCS Common Inbound Framework (IB)** – Asynchronous design pattern to synchronize data in Oracle PaaS ATP DB layer from Oracle EDMCS SaaS application.



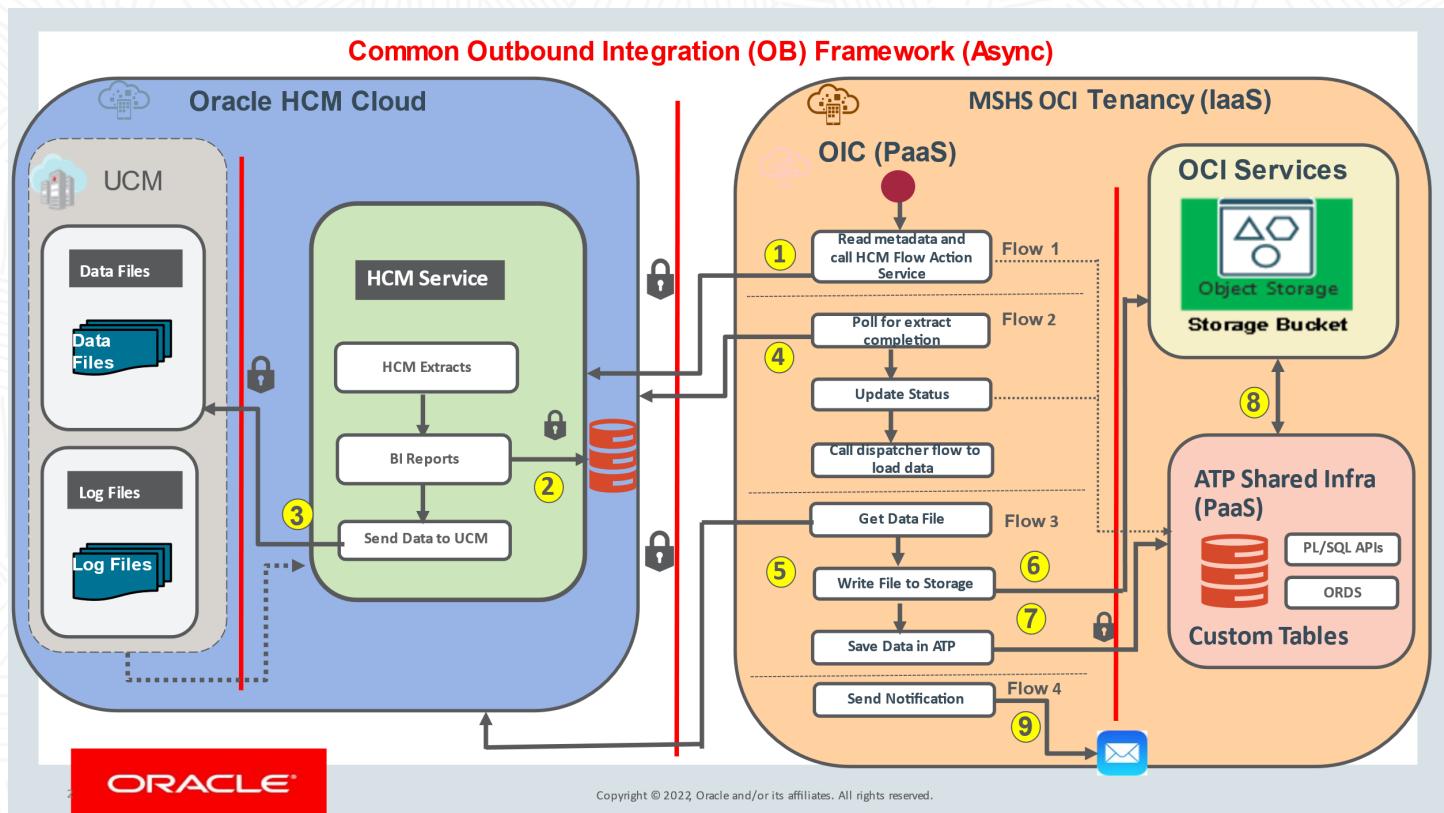
Mount Sinai

- **OIC Common Inbound Framework (HFWK_IB)** – Asynchronous design pattern to load and import data in Oracle HCM SaaS application from Oracle PaaS ATP DB layer. This design pattern is used to integrate the following two use cases.
 - Integrate MSHS Custom PaaS VBCS Payroll extensions with Oracle HCM Cloud SaaS application through Oracle ATP DB layer.
 - Integrate External/Vendor employee, and payroll data with Oracle HCM Cloud SaaS application through Oracle ATP DB layer.

High Level Integration Design

This section illustrates the high-level integration design used for the OIC PaaS migration activities.

HCM Common Outbound Framework (HFWK_OB)





Mount Sinai

Common Outbound Integration (OB) Framework (Async) - Steps

- ① Launcher service in OIC reads meta data and calls OIC service to submit HCM extracts
- ② HCM extract/BI Report is submitted in SaaS to extract data
- ③ Extracted data is sent to Oracle UCM
- ④ Polling service in OIC checks for data extraction status at defined intervals
- ⑤ Dispatcher service in OIC gets the extracted data files from UCM
- ⑥ Data files are uploaded to Object storage in OCI by the dispatcher service
- ⑦ OIC orchestrates the DB stored procedure to process the data from object storage
- ⑧ Data is interfaced in Oracle PaaS custom tables by the DB stored procedure from object storage
- ⑨ OIC sends the notification by reading the meta data for email groups

ORACLE®

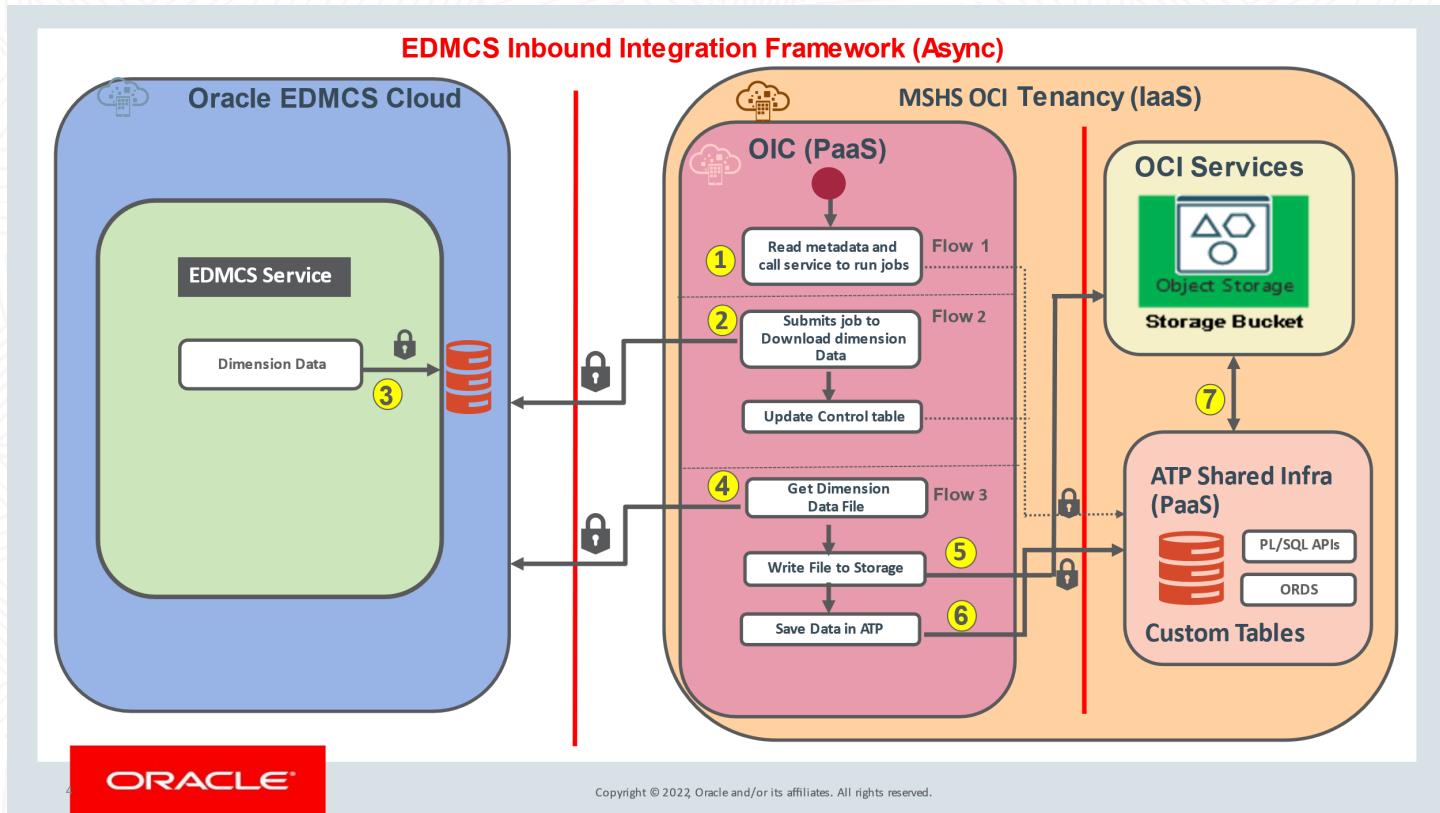
© 2020 TCS, OCS, & ManpowerGroup Confidential 8/15/2022

Copyright © 2022, Oracle and/or its affiliates. All rights reserved.

3

3

EDMCS Common Inbound Framework (EDMCS_IB)





EDMCS Inbound Integration Framework (Async) - Steps

- ① Launcher service in OIC reads meta data and calls the EDMCS control service in OIC
- ② EDMCS control service submits the Dimension job to extract data in EDMCS
- ③ EDMCS job extracts the data in EDMCS
- ④ EDMCS file process service in OIC checks for data extraction status at defined intervals and gets the data files once the extraction is completed
- ⑤ EDMCS file process service writes the data files to object storage
- ⑥ EDMCS file process service calls the DB stored procedure to process the data from object storage
- ⑦ Data is interfaced in Oracle PaaS custom tables by the DB stored procedure from object storage



© 2020 TCS, OCS, & ManpowerGroup Confidential 8/26/2022

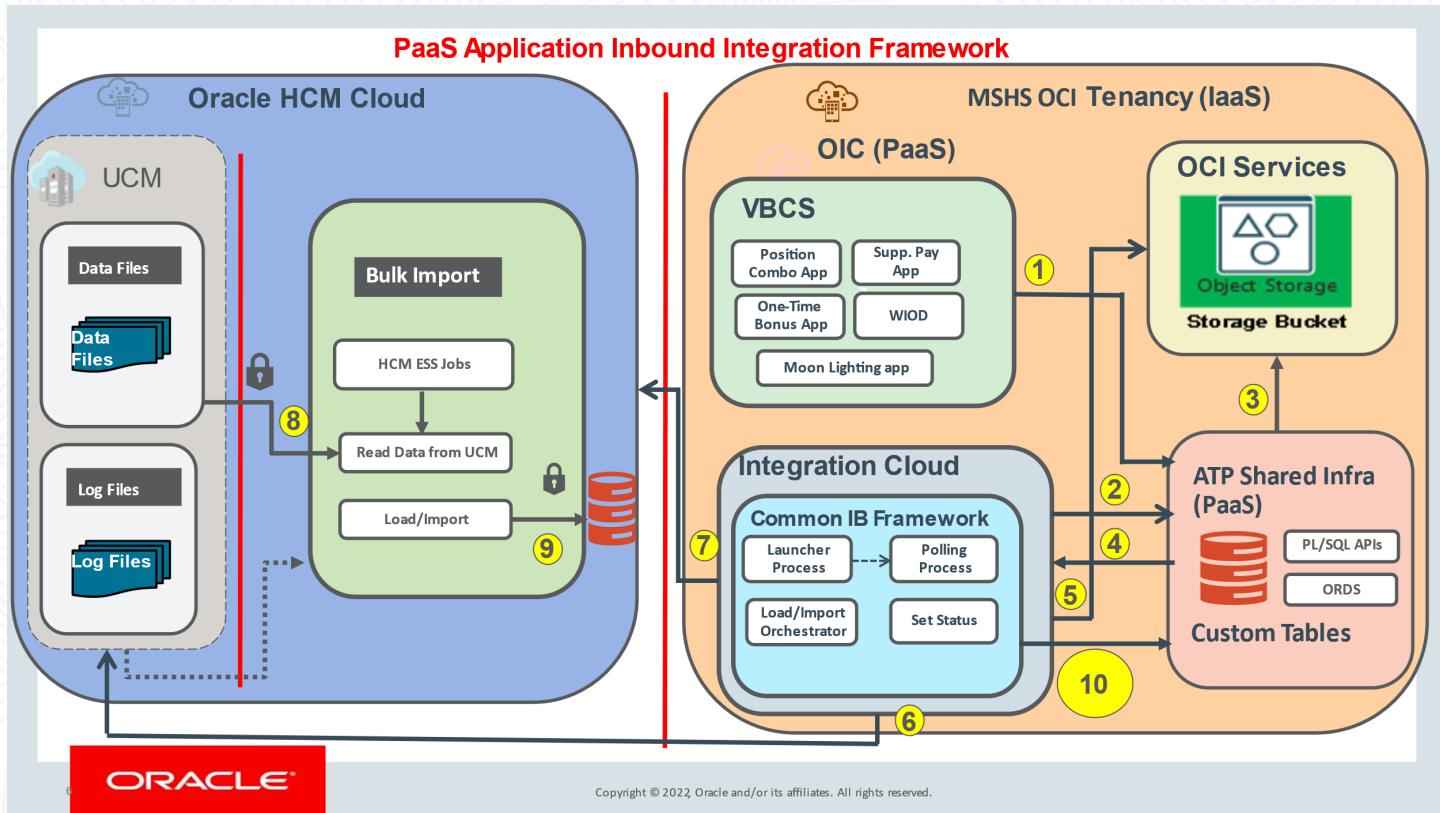
Copyright © 2022, Oracle and/or its affiliates. All rights reserved.

5

HCM Common Inbound Framework (HFWK_IB) – PaaS Applications



**Mount
Sinai**





Mount Sinai

PaaS Application Inbound Integration Framework - Steps

- ① VBCS apps create payroll transactions in PaaS DB tables
- ② OIC common IB framework polls for DB transactions at frequent intervals and calls DB procedure to create input files
- ③ DB stored procedure uploads the input files in object storage in OCI
- ④ Input file names are retrieved by the OIC service
- ⑤ OIC service reads the files from object storage and packs the file to import
- ⑥ Data files are uploaded to UCM by the OIC service
- ⑦ OIC service calls the HDL import process
- ⑧ Data is read in UCM by the HDL service
- ⑨ Data is interfaced in HCM product tables
- ⑩ OIC updates the import status in PaaS tables

ORACLE®

© 2020 TCS, OCS, & ManpowerGroup Confidential 8/15/2022

Copyright © 2022, Oracle and/or its affiliates. All rights reserved.

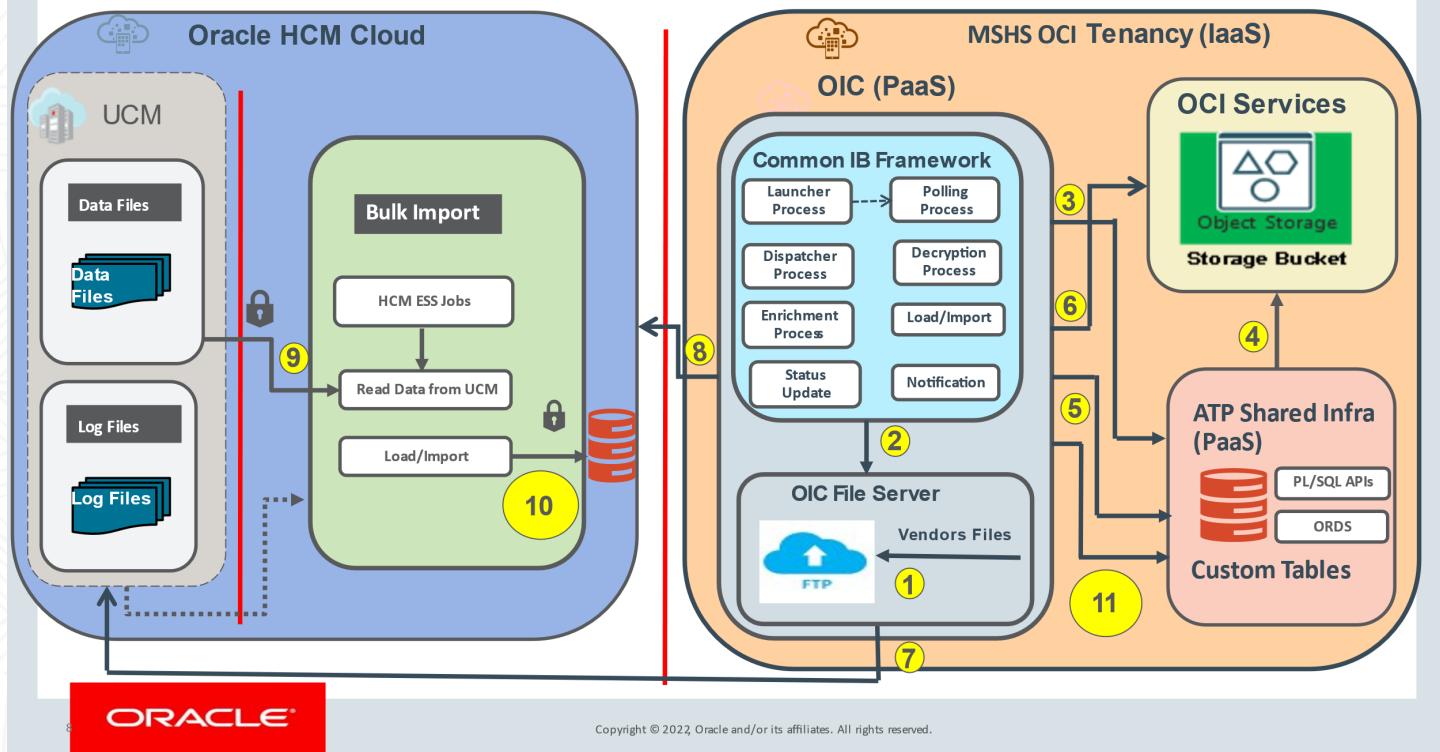
7

HCM Common Inbound Framework (HFWK_IB) – External/Vendor Integration



**Mount
Sinai**

External Vendor IB Integration Framework





Mount Sinai

PaaS Application Inbound Integration Framework - Steps

- ① Vendor places input files in OIC file server
- ② OIC common IB framework polls for input files from OIC file server
- ③ OIC service decrypts the data files, and calls DB procedure to transform, and enrich the data
- ④ DB stored procedure uploads the input files in object storage in OCI
- ⑤ Input file names are retrieved by the OIC service
- ⑥ OIC service reads the files from object storage and packs the file to import
- ⑦ Data files are uploaded to UCM by the OIC service
- ⑧ OIC service calls the HDL import process
- ⑨ Input Data is read in UCM by the HDL service
- ⑩ Data is interfaced in HCM product tables
- ⑪ Import status is updated in DB tables and notification is sent to the email groups

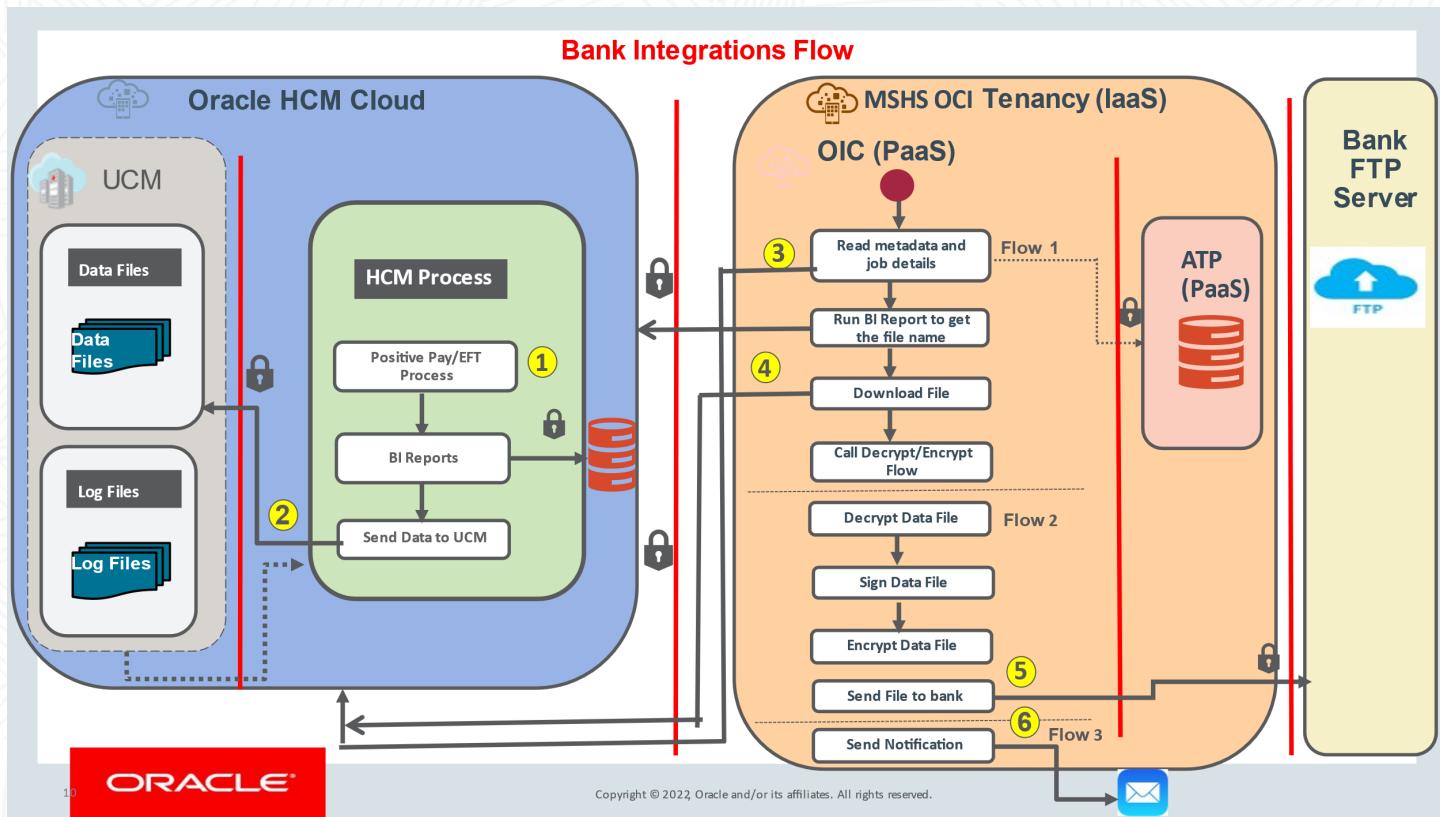


© 2020 TCS, OCS, & ManpowerGroup Confidential 8/15/2022

Copyright © 2022, Oracle and/or its affiliates. All rights reserved.

9

Bank Integrations





**Mount
Sinai**

Bank Integrations Flow - Steps

- 1** HCM business processes are run for Positive Pay, and EFT Payments to generate data files
- 2** Encrypted data files are sent to Oracle UCM
- 3** OIC will get the details of the encrypted files to be downloaded from UCM
- 4** OIC downloads the encrypted file from UCM
- 5** Data file is decrypted first by OIC
- 6** OIC signs the data files using the certificates
- 7** OIC encrypts the data files using the certificates
- 8** OIC transfers encrypted files to Bank's FTP server location
- 9** OIC sends notification to the configured email group

ORACLE®

©2020 TCS, OCS, & ManpowerGroup Confidential 8/29/2022

Copyright © 2022, Oracle and/or its affiliates. All rights reserved.

11

Detailed Integration Design

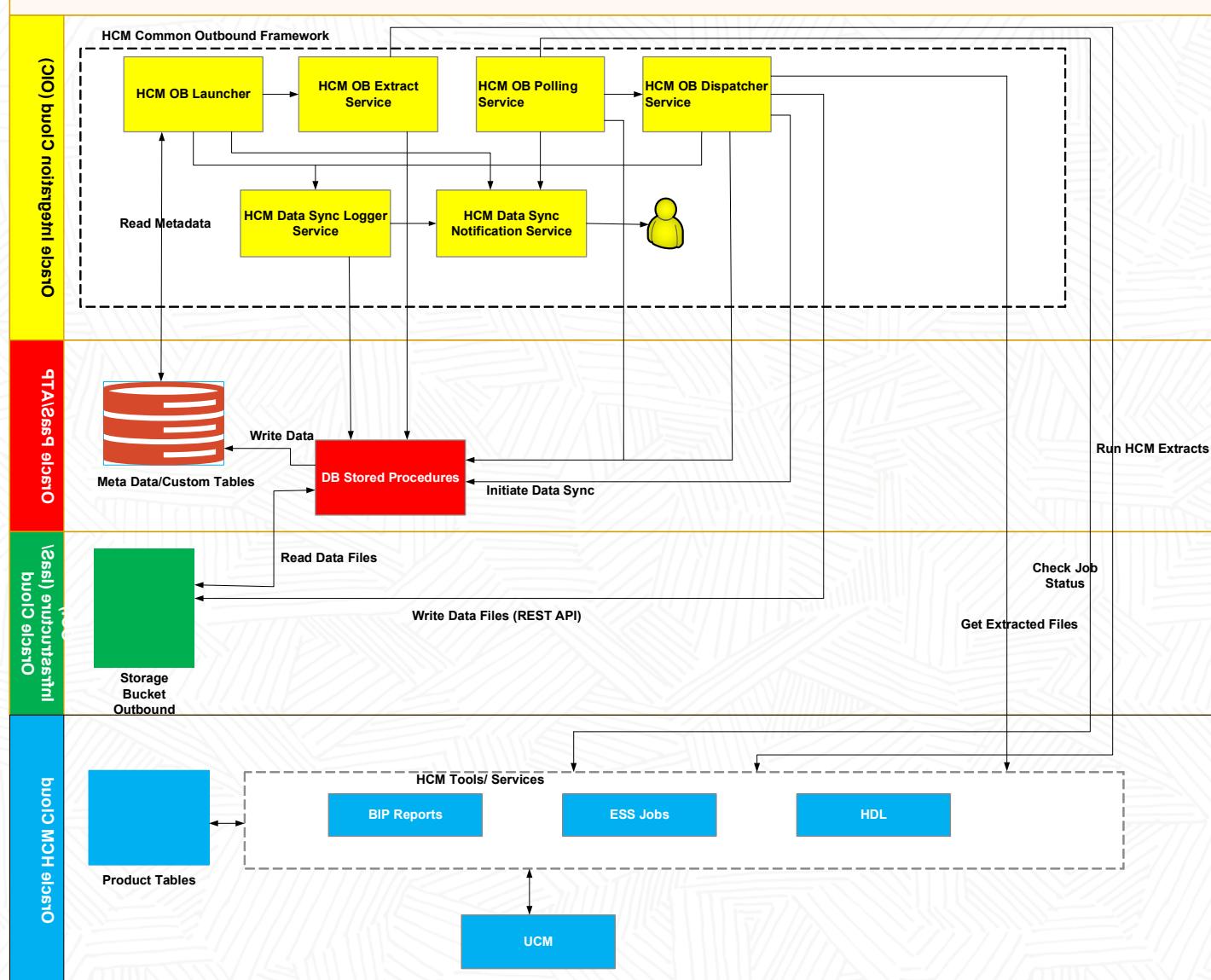
Following diagrams illustrate the detailed design for the following integrations framework.

HCM Common Outbound Framework (HFWK_OB)



Mount Sinai

HCM Common Outbound Framework (HFWK_OB)

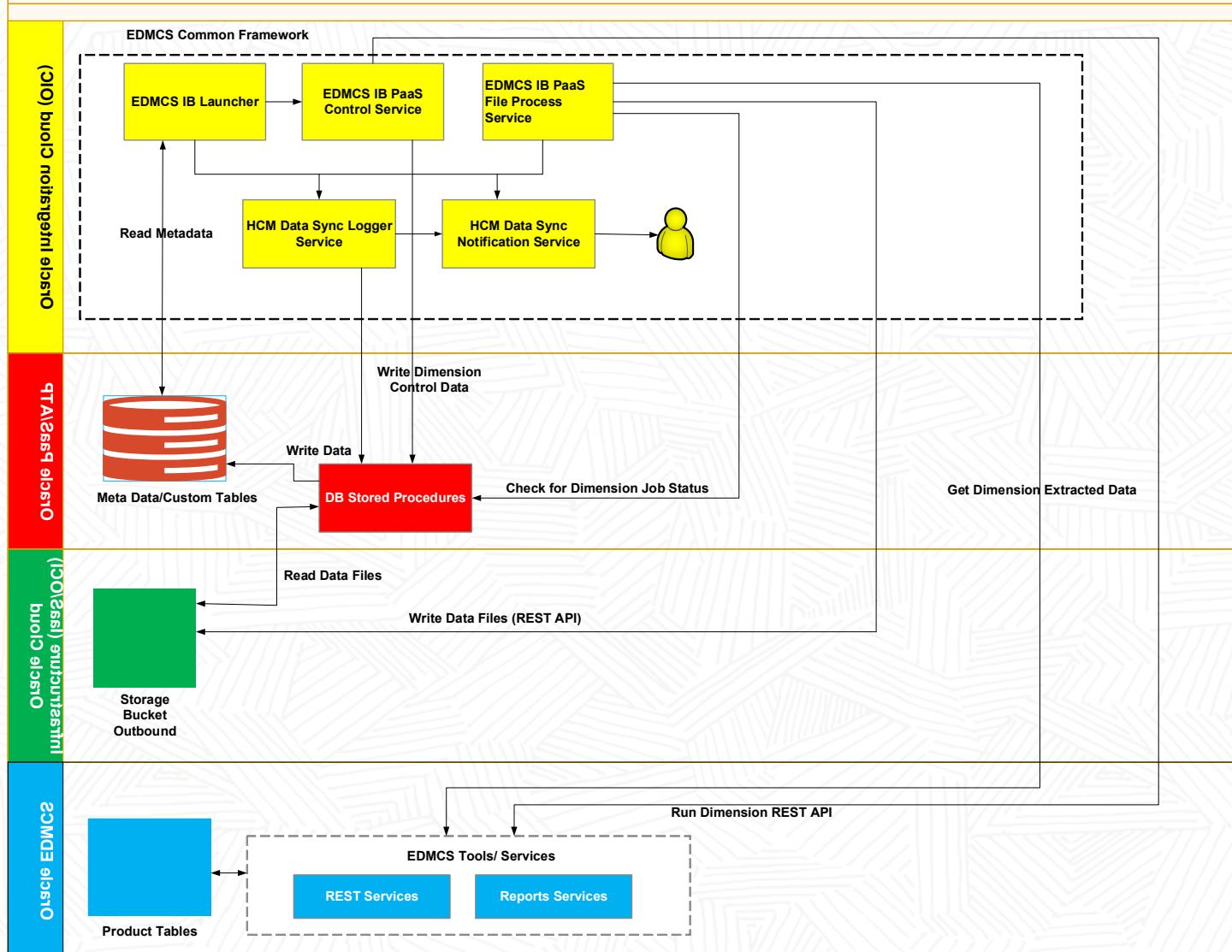


EDMCS Common Inbound Framework (EDMCS_IB)



Mount Sinai

EDMCS Inbound Common Framework

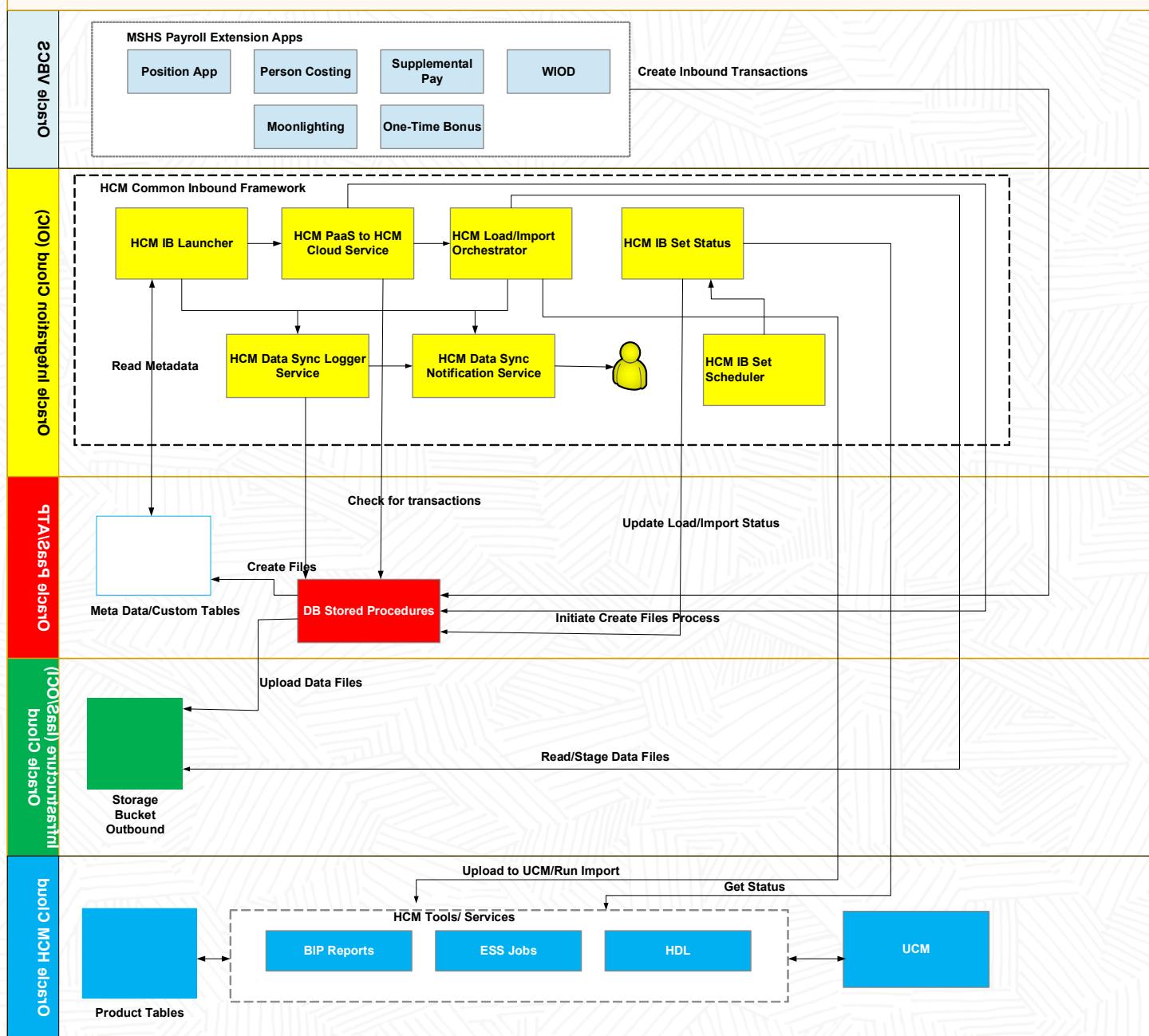


HCM Common Inbound Framework (HFWK_IB) – PaaS Applications



Mount Sinai

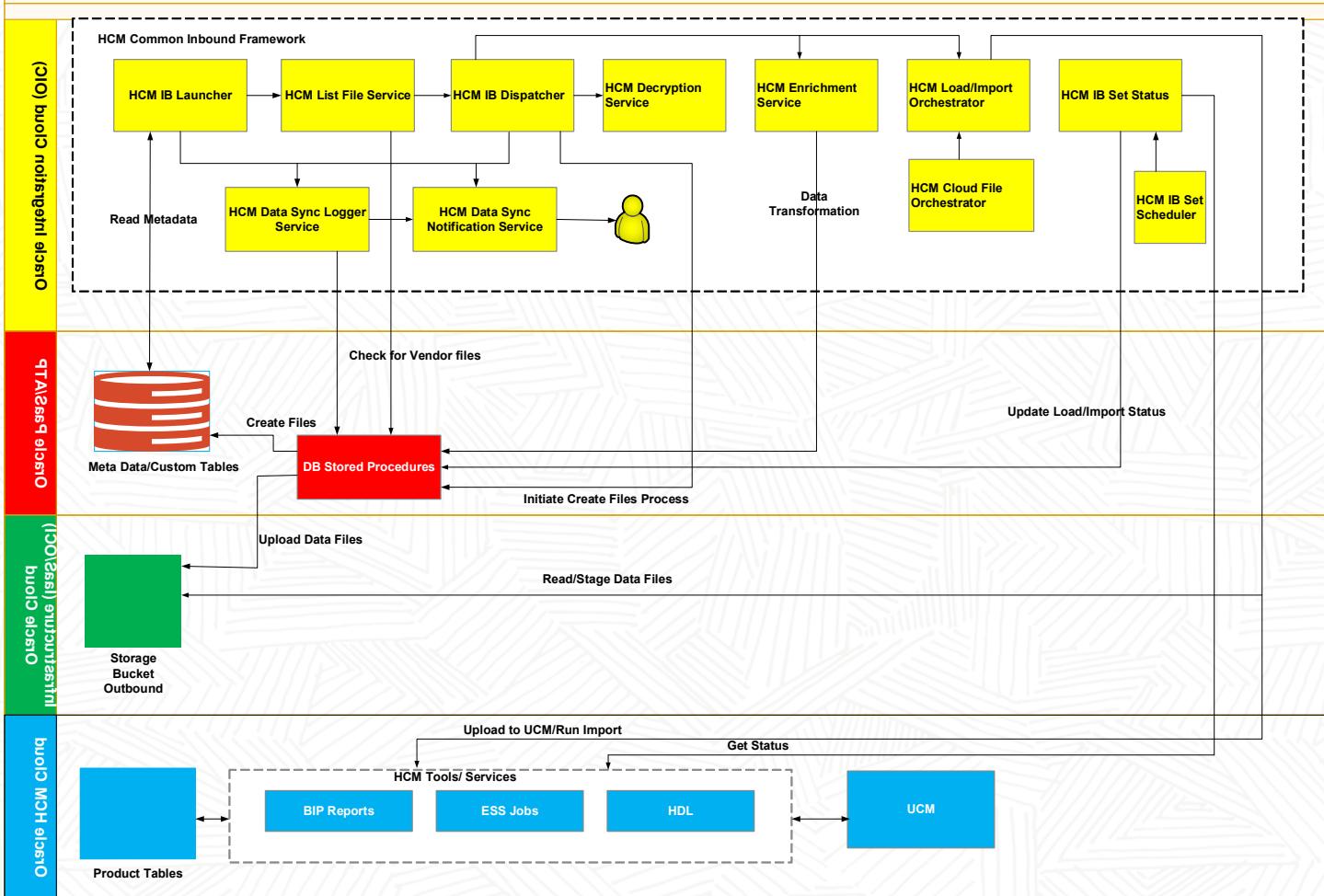
HCM Common Inbound Framework (HFWK_IB) – PaaS Extension Apps Integration



HCM Common Inbound Framework (HFWK_IB) – External/Vendor Integration



HCM Common Inbound Framework (HFWK_IB) – Vendor Integration



Design Considerations



Application Security	<ul style="list-style-type: none"> SaaS service account will be used to configure HCM adapter connections in OIC. OCI user tokens and private keys will be used to configure object storage connection in OIC to read/write files. REST Adapter connection in OIC will be used to connect to EDMCS using a service account Service account for OIC file server will be used to configure SFTP adapter connections in OIC for file transfer. OIC agent is used to connect to the database hosted in the private network.
Transmission Method	<ul style="list-style-type: none"> UCM to ATP DB via OIC Object Storage to ATP DB via OIC
Incremental/Full Data	<ul style="list-style-type: none"> Outbound: Full Load. Data is always truncated before each load in ATP DB tables.
File Naming Convention	
Scheduling Requirement	<ul style="list-style-type: none"> Same ESS schedules will be followed in OIC like SOA implementation.
Record Delimiter	<ul style="list-style-type: none"> CSV

Extract Program Parameters

Parameter values will be defined in the Metadata tables in ATP DB and is the same used for the SOA implementation. Please refer to appropriate documents created for those efforts.

Outbound Data Format

The outbound files that are created by the synchronization framework use the same mechanism that is used in the SOA implementation. Please refer to appropriate documents created for those efforts.

Prerequisites/Functional Setup Requirements

1. OCI Bucket needs to be set up in the OCI layer.
2. Service account with appropriate roles need to integrate the data in HCM should be created in HCM SaaS application. The OIC framework uses the same service account used for the SOA implementation.
3. Wallet for the ATP DB need to be downloaded and configured in OIC.
4. OIC agent is installed in the VM instance in OCI that facilitates the connection to Oracle ATP DB.



5. UCM, and other security related setups are needed in HCM for file transfer. These setups are maintained the same that is used in the SOA implementation.
6. OIC file server account to be configured to allow transfer of files in OIC file server.

Functional Test Scenarios

Scenario	Key Objective	Expected Outcome
1. FULL Load of data from Oracle HCM Cloud and EDMCS applications	Data Sync to ATP DB layer	All data for the respective entity should be synced to ATP layer
2. Exceptions	Notify the user group and update respective DB tables	The user group should be notified, and DB tables should be updated with relevant details.
3. Inbound Processing customer, and payroll data into HCM from Custom Payroll extensions, and external/vendor applications	Data is integrated in Oracle HCM Cloud	All data should be loaded/imported successfully in Oracle HCM Cloud.

Technical Solution Overview & Components

Summary

OIC integration frameworks consist of the following technical components.

- ATP DB
 - Holds the metadata configuration used by the data synchronization process to execute the data extract jobs, and to synchronize in ATP DB tables at specified intervals.
 - Custom tables in ATP DB layer to maintain status and details for each scheduled/ad-hoc run.
 - Holds the output transaction tables for the data from the HCM Cloud, and EDMCS applications which are extracted using ESS jobs, BIP, HCM Extracts, and REST APIs.
 - Holds the custom tables, and DB objects used to process the data into Oracle HCM Cloud application.
- PL/SQL Packages
 - APIs to read the data files from OCI object storage, and to process the data in ATP DB transaction tables.



- Orchestrated by OIC.
- OCI Object Storage
 - Storage buckets to hold the data files from Oracle HCM, and EDMCS SaaS applications, and a layer for file transfer.
 - Generated output file will be placed in respective folders at the Object Storage location.
 - OIC orchestrates the calls to transfer files from storage buckets to PaaS ATP DB.
- UCM – Universal Content Management
 - Output files will be placed in UCM that are generated by BI reports, and HCM extracts.
- Oracle Integration Cloud
 - Common Outbound Integration Framework is built to synchronize data from Oracle HCM Cloud, and EDMCS applications to PaaS.
 - Orchestrate ESS, HCM Extract jobs, and REST APIs at scheduled intervals to extract data from Oracle HCM Cloud, and EDMCS applications.
 - Orchestrate to move generated files from UCM to Object Storage on completion of extract jobs.
 - Orchestrate ATP DB APIs to process the data into ATP DB transaction tables.
 - Check ESS Job status in Oracle HCM Cloud, and EDMCS applications for Job completion (Success/Warning/Error).
 - Send email notifications to configured email groups on processing status (Success/Error).
 - Common Inbound Integration Framework is built to load/import custom data into HCM.
 - Polls for transactions created by custom payroll VBCS extension applications and prepares data files for load and import of data into HCM.
 - Polls for files in OIC file server queued by the external vendors, and to process into Oracle HCM cloud.
 - Decrypt inbound data in OIC as needed.
 - Orchestrate to perform additional enrichment/transformation in the ATP DB layer.
 - Orchestrate data loads into Oracle UCM.
 - Orchestrate import jobs to import data into Oracle HCM Cloud from UCM.
 - Perform error handling and send email notifications to configured email groups on processing status (Success/Error).
- Oracle HCM Cloud
 - Data models are defined for BIP reports.
 - BIP reports are exposed as ESS jobs.
 - BIP reports are used in HCM extracts.
 - HCM extracts will be configured to extract data from HCM cloud.
- OIC File Server



Mount Sinai

- This is an in-built FTP server in OIC used for file transfers.
- Incoming files are processed into HCM placed by vendors in OIC file server.
- Error and output files are written to appropriate directories by integrations framework.



Solution Components

OIC Components

Integrations

#	Name	Description
1	HFWK_OB_LAUNCHER_SERVICE	HCM Outbound Launcher Service to initiate the PaaS OB data synchronization process based on metadata configurations
2	HFWK_OB_HCM_EXTRACT_SERVICE	HCM OB service to run extracts.
3	HFWK_OB_POLLING_SERVICE	HCM Service to check status for PaaS OB data extractions.
4	HFWK_OB_DISPATCHER_SERVICE	HCM Service to orchestrate the load of data into PaaS ATP tables for Outbound integrations
5	HFWK_OB_NOTIFICATION_SERVICE	Common Outbound notification service utilized by all integrations
6	HFWK_DATASYNC_NOTIFICATION_SERVICE	Common Outbound notification service utilized by all integrations
7	HFWK_DATASYNC_LOGGING_SERVICE	Common logging service used by all integrations
8	EDMCS_IB_LAUNCHER_SERVICE	Inbound launcher Service to initiate the EDMCS data synchronization processes based on metadata configurations.
9	EDMCS_IB_PAAS_CNTRL_SERVICE	EDMCS service to create control table transaction for EDMCS jobs
10	EDMCS_IB_PAAS_FILE_PROCESS_SERVICE	Orchestrates the EDMCS data load into PaaS ATP tables for EDMCS entities
11	HFWK_IB_LAUNCHER_SERVICE	Inbound launcher service to process data into HCM Cloud for PaaS/VBCS custom extension payroll applications.
12	HFWK_IB_PAAS_TO_HCMCLOUD	Polls for inbound transactions to be interfaced into HCM Cloud. These transactions are created by PaaS/VBCS custom extension payroll applications.



13	HFWK_IB_ORCHESTRATE_LOAD_HDL_PROC_ESS	Common service to orchestrates load and import of data into HCM Cloud
14	HFWK_IB_HDL_DATA_SET_STATUS_SCHEDULER	Scheduler service to run the process for load/import HCM job status.
15	HFWK_IB_HDL_DATA_SET_STATUS_PROCESS	Updates the load/import job status in PaaS ATP tables for inbound integrations
16	HFWK_IB_HDL_DATA_SET_STATUS_SUPP_PROCESS	Updates the load/import job status in PaaS ATP tables for some of the integrations used for PaaS/VBCS custom payroll applications.
17	HFWK_IB_EXTERNAL_INTF_LAUNCHER_SERVICE	Launcher service to initiate the load/import of data into Oracle HCM for external vendor integrations.
18	HFWK_IB_LIST_FILE_SERVICE	Service to look for incoming vendor files and to process them
19	HFWK_IB_DISPATCHER_SERVICE	Service to orchestrate the data decryption, enrichment, and HCM load/import services.
20	HFWK_IB_DECRYPTION_SERVICE	Decrypt the incoming vendor data files
21	HFWK_IB_ENRICHMENT_SERVICE	Service to orchestrate the additional data enrichment logic based on meta data configurations for external vendor integrations
22	HFWK_IB_CLOUDFILE_ORCHESTRATE_SERVICE	Orchestrator service to check the data enrichment status and to orchestrate the HCM load/import service. This is used only for integrations that need additional data enrichment
23	HFWK_LOAD_FUND_APPROVERS_LAUNCHER	Launcher service to run the sync process to integrate fund number approvers data into PaaS ATP tables
24	HFWK_LOAD_FUND_APPROVER_SERVICE	Service to orchestrate the load of fund number approvers data into PaaS ATP tables.
25	HFWK_LOAD_FUND_APPROVERS_NOTIFICATION_SERVICE	Notification service to email user groups for fund number approver sync process.
26	HFWK_OB_PAYROLL_SERVICE	Service used to send the data to banks from HCM
27	HFWK_OB_ENCRYPTION_DECRIPTION_SERVICE	Service used to encrypt or decrypt data files for bank integrations
28	HFWK_INTERF_DATA_PURGE_SERVICE	Service to purge the data in PaaS ATP tables
29	HFWK_DATASYNC_DELETION_SERVICE	Service to delete the files from object storage and OIC file server



30	EDMCS_IB_PAAS_ERP_FILE_PROCESS_SERV ICE	Used to populate data for EDMCS multi segment map entity (198A10)
----	--	---

Connections

#	Name	Description
1	HFWK_OIC_FILE_SERVER	Used by integrations to connect to OIC file server for transfer of files
2	HFWK_HCM_CLOUD_CONNECTION	Integrations use this to connect to HCM cloud
3	HFWK_BIP_Connection	SOAP connection to run BIP reports
4	HFWK_SOAP_GENERIC_CONNECTION	SOAP connection to run UCM related services
5	HFWK_SOAP_CONNECTION	SOAP connection to run HCM flow action service
6	HFWK_REST_TRIGGER	REST service trigger used by integrations
7	HFWK_OB_ATP_CONNECTION	ATP DB connection used by outbound integrations
8	HFWK_ATP_CONNECTION	ATP DB connection used by all integrations except outbound integrations
9	HFWK_OBJ_STORAGE_CONN	Object storage connection used by integrations to read/write files
10	HFWK_OIC_REST_CONN	REST Connection trigger and invoke
11	EDMCS_REST_CONN	Used by EDMCS integrations to connect to EDMCS application
12	HFWK_OIC_BANK_FILE_SERVER	Used for the bank integrations (INT_OB_49 and INT_OB_50)
13	JPMC_SFTP_SERVER_CONN	This is used for JPMC bank integration to place the Pay positive and EFT payment files.
14	TDBANK_BI_SFTP_SERVER_CONN	This is used for TD bank integration to place the Pay positive and EFT payment files.
15	TDBANK_SL_SFTP_SERVER_CONN	This is used for TD bank integration to place the Pay positive and EFT payment files.
16	HFWK_TIAA_FTP_CONNECTION	Used to place the Outbound report files in the MSHS internal FTP server for TIAA



Lookups

#	Name	Description
1	HFWK_OIC_FILE_SERVER_LKP	Lookup to manage OIC file server directories
2	HFWK_NOTIF_LKP	Lookup for email notifications
3	HFWK_INTF_METADATA	Lookups to manage interface metadata
4	HFWK_OBJ_STORAGE_LKP	Lookups to manage object storage buckets and namespaces
5	HFWK_PGP_LKP	Lookups to manage PGP keys for external vendor integrations.

Certificates

1	MSHS_ALL_INTERNAL_IB_PRIVATE_KEY	Certificate used for most of the external vendor IB integrations
2	MSHS_VOYALIFE_PRIVATE_KEY	Used for Voyalife related IB integrations
3	MSHS_KRONOS_PRIVATE_KEY	Used for Kronos related IB integrations
4	MSHS_WAND_PRIVATE_KEY	Used for Wand related IB integrations
5	MSHS_TIAA_PRIVATE_KEY	Used for TIAA integrations

OCI Components

This section describes the Oracle Cloud Infrastructure components such as object storage/buckets used for file transfers by all integrations.

Object Storage/Buckets

Bucket Name	Description
OIC_DEV2_OUTBOUND	Buckets used for outbound integrations in PaaS DEV2 environment



OIC_DEV2_INBOUND

Buckets used for inbound integrations in PaaS DEV2 environment

OIC_DEV4_OUTBOUND

Buckets used for outbound integrations in PaaS DEV4 environment

OIC_DEV4_INBOUND

Buckets used for inbound integrations in PaaS DEV4 environment

OIC_PROD_OUTBOUND

Buckets used for outbound integrations in PaaS PROD environment

OIC_PROD_INBOUND

Buckets used for inbound integrations in PaaS PROD environment

Integration Logic

This section describes the flow logic for integrations used in the OIC PaaS migration project implementation.

HCM Common Outbound Framework (HFWK_OB)

This section describes the integrations that are used for the HCM common outbound framework. These integrations are used to synchronize the data from HCM Cloud into Oracle PaaS ATP DB tables. In addition to the synchronization, some of the outbound common integrations are used to generate data files for TIAA outbound integrations that are send to the MSHS internal FTP server. The services work based on the metadata configurations. Currently, it supports the data synchronization for the following outbound integrations.

Integration Id	Description
INT_OB_PAAS_1	PaaS HCM Extension Work Structure
INT_OB_PAAS_2	PaaS HCM Extension Demographic
INT_OB_PAAS_3	PaaS HCM Extension Payroll
INT_OB_PAAS_10	PaaS HCM Extension Payroll HDL
INT_OB_19_EMP_MSBISLW	TIAA CREF Census and Remittance Report for Employee (MSBISLW)
INT_OB_19_EMP_MSH	TIAA CREF Census and Remittance Report for Employee (MSH)

**Note:**

When a new outbound data synchronization from HCM Cloud application into PaaS ATP DB tables is needed in the future, following objects need to be created or configured first.

- DB objects such as staging tables, interface tables, sequences, indexes, PL/SQL packages, object types, and synonyms should exist in ATP DB layer.
- BI Reports and/or HCM extracts should be created.
- Metadata configuration needs to be defined for the new interface.

Once these objects are created, common HCM outbound framework can be used to orchestrate the data synchronization.

HFWK_OB_LAUNCHER_SERVICE

- It is a scheduled integration, which will be scheduled as per the time defined in the metadata tables.
- The following query is used to fetch integration details.
 - `SELECT intf_id, source,target,job_name,extract_type,active,INTF_SCHEDULE_TIME FROM MSHS_SOAINST_USER.CMFK_intf_METADATA where active='Y' and integration_mode='OUTBOUND' and source='Cloud HCM' and target in ('PaaS','TIAA')`
- If the INTF_SCHEDULE_TIME column in metadata table matches with OIC current time and day, specific matching criteria interface id will be passed to HFWK_OB_HCM_EXTRACT_SERVICE integration. The value for INTF_SCHEDULE_TIME column of cmfk_intf_metadata table must be in UTC format, since OIC server time is in UTC time zone.
- By using the schedule parameter INTERFACE_ID_ID, it can be run on demand basis also for a single interface id. FROM_DATE and THROUGH_DATE can be passed manually for TIAA integrations

HFWK_OB_EXTRACT_SERVICE

- This integration submits the HCM extract process.
- It gets the metadata details and creates a transaction in cmfk_intf_master_tbl.
- Integration submits the HCM extract and updates the master table with flow instance id.
- This flow is called by the OB launcher service.

HFWK_OB_POLLING_SERVICE

- This integration polls data from the master table using the procedure INTERFACE_POOL_MASTER. It returns details of all flows for which the HCM extracts are running.
- It checks the status of the HCM extract.



- If the Extract is completed, HFWK_OB_DISPATCHER_SERVICE is called.
- If the Extract is not completed, it checks the max wait time using CHECK_TIME_OUT procedure and if the time out is exceeded, the integration throws, and error and the master tables is updated with Error status as well.
- If the time out is not exceeded, the integration updates the master table with running status.

HFWK_OB_DISPATCHER_SERVICE

- This integration gets the metadata details and BI report details from ATP and runs the BI report to get the list of UCM ids for respective HCM extracts.
- After getting the report response output, it is decoded and read through the stage activity to get the UCM ids.
- All these documents are downloaded from UCM and uploaded to object storage.
- Then the integration calls CMFK_INTF_MASTER_PKG.INSERT_DATA_SYNC and calls CMFK_INTF_MASTER_PKG.UPDATE_DATA_SYNC_MASTER to insert and update transaction details.
- Then the integration calls MSHS_HCM.Utility_INTERFACE.START_CLOUD_DATA_SYNC to start the data sync process.
- In case of any errors, this integration deletes all files uploaded to Object storage and updates the cmfk_intf_master_tbl using UPDATE_DATA_SYNC_MASTER procedure with the status ERROR.

HFWK_OB_NOTIFICATION_SERVICE

- This is a scheduled integration which polls for transactions using CMFK_INTF_MASTER_PKG.INTERFACE_POOL_NOTIFY procedure.
- If the interface is a TIAA interface, it deletes the file from the object storage.
- Then this integration calls HFWK_DATASYNC_NOTIFICATION_SERVICE to send success/error notification depending on the status of the data sync process.

Caveat

For the automated schedules to work in OIC for PaaS OB integrations, the meta data table in ATP DB must be updated for the outbound interface in **UTC** time zone to match the schedule in OIC (**Example:** 00:00 AM EST in OIC -> 04:00 AM UTC in ATP DB). Otherwise, the automated schedules will not work. For more information, refer to the document which describes the steps to configure the schedules in **Appendix D**.

EDMCS Inbound Framework (EDMCS_IB)



Mount Sinai

This section describes the integrations that are used in the common EDMCS IB framework. These are used to synchronize data from EDMCS cloud application into PaaS ATP DB tables. The integrations work based on the metadata configurations. Currently, it supports the data synchronization for the following EDMCS inbound entities.

Integration Id	Description
INT_IB_EDMCS_1	EDMCS GEAC HCM Fund Number
INT_IB_EDMCS_2	EDMCS GEAC HCM Entity
INT_IB_EDMCS_3	EDMCS GEAC HCM Net Asset Class
INT_IB_EDMCS_4	EDMCS GEAC HCM Cost Center
INT_IB_EDMCS_5	EDMCS GEAC HCM Functional Class
INT_IB_EDMCS_6	EDMCS GEAC HCM Fund Type
INT_IB_EDMCS_7	EDMCS GEAC HCM Site
INT_IB_198A10	EDMCS GEAC Multi Segment Data to Stage

Note:

When a new inbound data synchronization from EDMCS application into PaaS ATP DB tables is needed in the future, following objects need to be created or configured first.

- DB objects such as staging tables, interface tables, sequences, indexes, PL/SQL packages, object types, and synonyms should exist in ATP DB layer.
- REST endpoints in EDMCS should be configured to extract data.
- Metadata configuration needs to be defined for the new interface.

Once these objects are created, common EDMCS inbound framework can be used to orchestrate the data synchronization.

EDMCS_IB_LAUNCHER_SERVICE

- It is a scheduled integration and runs at specified intervals. Takes the input parameter for interface id to process the specific EDMCS integration as well. Interface id must be entered as maintained in metadata tables and no validation is done.
- Below query is used to fetch the interface entity details.
 - SELECT ENTITY,INTF_SCHEDULE_TIME,intf_id,DESCRIPTION,SOURCE,TARGET FROM cmfk_intf_metadata WHERE module='HCM' and source='EDMCS' and active='Y' and intf_id = #intf_id order by intf_id
- If the INTF_SCHEDULE_TIME value in metadata table matches with OIC schedule time, then that specific matching criteria entity will be passed to EDMCS_IB_PAAS_CNTRL_SERVICE integration. The value for INTF_SCHEDULE_TIME column of cmfk_intf_metadata table must be in UTC format, since OIC server time is in UTC time zone.



EDMCS_IB_PAAS_CNTRL_SERVICE

- It is app driven integration, which accepts JSON message as input.
- This integration will be triggered by EDMCS_IB_LAUNCHER_SERVICE by passing Entity Number as input. Sample Input payload – { "IntegrationEntityName" : "EDMCS_GEAC_HCM_FUND_NUMBER" }
- Integration id for IntegrationEntityName is retrieved from DVM.
- DB call is made to get Application Name (Ex: Oracle Cloud GL), Dimension Name (Ex: Fund Name), and Mapping Name for that entity by passing Integration id.
- REST Service is called to get all Dimensions from EDMCS.
- Application Name and Dimension Name retrieved from DB are matched with REST service response from EDMCS to get the Dimension id.
- REST service to run dimension job with Dimension Id and File Name as parameters is called by the integration.
- Once job is completed, Job Id is stored in control table for further processing.

EDMCS_IB_PAAS_FILE_PROCESS_SERVICE

- It is a scheduled integration which runs at defined intervals.
- This is a polling service which will poll the CMFK_HCM_EDMCS_FILE_TBL table using msbs_hcm_edmcs_intf_pkg.edmcs_job_status_pool_list procedure.
- If the record is found, then the metadata details are fetched for that EDMCS entity using cmfk_utility_intf_pkg.get_intf_metadata procedure and further process begins from below steps, else the integration exits.
 - Calls EDMCS REST API to get the job status.
 - If the job status is completed successfully, then REST API is called to get the file, else error email notification is triggered, and error message is inserted in the logging table.
 - If the get file API returns success return code, then the file is uploaded to the object storage, else error email notification is triggered, and error message is inserted in the logging table.
 - After uploading the EDMCS entity file in object storage, msbs_hcm_edmcs_intf_pkg.cloud_copy_data procedure is called, which copies the data from the file uploaded to the object storage and insert in to the respective entity staging and interface table. If the procedure returns any error, then error email notification is triggered, and error message is inserted in the logging table
 - On successful call of the cloud_copy_data procedure, msbs_hcm_edmcs_intf_pkg.update_out_edmcs_file procedure is called which updates the job status column of CMFK_HCM_EDMCS_FILE_TBL to P.

EDMCS_IB_PAAS_ERP_FILE_PROCESS_SERVICE

- This is a scheduled integration, and it works like the EDMCS_IB_PAAS_FILE_PROCESS_SERVICE.
- It is only used to process the data for interface INT_IB_198A10.
- The process uses a different stored procedure and polling table as opposed to the tables and stored procedure used for other EDMCS integrations.



- This also uses another REST service that is different from the service that is used for other EDMCS integrations.

Caveat

For the automated schedules to work in OIC for EDMCS integrations, the meta data table in ATP DB must be updated for the EDMCS interface in **UTC** time zone to match the schedule in OIC (**Example:** 07:00 AM EST in OIC -> 11:00 AM UTC in ATP DB). Otherwise, the automated schedules will not work. For more information, refer to the document which describes the steps to configure the schedules in **Appendix D**.

HCM Common Inbound Framework (HFWK_IB) – PaaS Applications

This section describes the integrations that are used in the common HCM IB framework for the PaaS VBCS custom payroll extension applications. These are used to load and import customer and payroll data from PaaS ATP DB layer into HCM Cloud application. The data is created by the PaaS VBCS custom payroll extension applications in PaaS ATP DB tables. The integrations work based on the metadata configurations. Currently, it supports the following PaaS inbound integrations.

Integration Id	Description
INT_IB_PAAS_1	PaaS HCM Position Request Inbound
INT_IB_PAAS_2	PaaS HCM Labor Costing Inbound
INT_IB_PAAS_3	PaaS HCM Supplemental Pay Inbound
INT_IB_PAAS_4	PaaS HCM WIOD Inbound
INT_IB_PAAS_5	PaaS HCM Moonlighting Inbound
INT_IB_PAAS_6	PaaS HCM One Time Bonus Inbound

Note:

When a new inbound integration to load and import into HCM Cloud application from PaaS ATP DB tables is needed in the future, following objects need to be created or configured first.

- DB objects such as transaction tables, interface tables, sequences, indexes, PL/SQL packages, object types, and synonyms should exist in ATP DB layer.
- BI Reports should be configured for import jobs, parameter, and status of jobs.
- Metadata configuration needs to be defined for the new interface.

Once these objects are created, common HCM inbound framework can be used to orchestrate the load and import of data.

HFWK_IB_LAUNCHER_SERVICE



- The service is a scheduled integration, which uses an ATP DB connection to fetch the details of integrations that are supposed to be processed (Example: INT_IB_PAAS_1, INT_IB_PAAS_2) from CMFK_INTF_METADATA based on filter criteria mentioned below.
- Following query is used to get the metadata details.
 - select * from CMFK_INTF_METADATA t where t.active='Y' and t.integration_mode='INBOUND' and t.SOURCE ='PaaS' and t.TARGET = 'HCM Cloud';
- Once the query fetches the result, the integration will call the integration HFWK_IB_PAAS_TO_HCMCLOUD to process transactions for INT_IB_PAAS_1 up until INT_IB_PAAS_6.
- The integration is scheduled to run at defined intervals.

HFWK_IB_PAAS_TO_HCMCLOUD

- Is an app driven integration that will input as JSON payload. The integration is called from the above launcher integration. HFWK_IB_PAAS_TO_HCMCLOUD integration acts as a polling service which will read the data from the cmfk_in_list_file_tbl using the procedure INTERFACE_POOL_LIST to check if there are any transactions which are in the column “Status” as N. If the procedure (INTERFACE_POOL_LIST) finds a transaction for the provided input, then the integration will proceed with the flow further. Else it will end the flow when there are no transactions to process. This is an asynchronous service.
- On receiving a transaction in the cmfk_in_list_file_tbl table – the integration will proceed to get the metadata details from GET_INTF_METADATA_ALL procedure call.
- The service calls a common PL/SQL wrapper packaged procedure to create files. If there is any error, the integration will call common logging service which also acts as common error handler and update the cmfk_in_list_file_tbl table for the error details that is returned from the CREATE FILE procedure.
- If create file procedure returns success status, the integration calls HFWK_IB_ORCHESTRATE_LOAD_HDL_PROCESS integration which will load and import the data to HCM cloud application.

HFWK_IB_ORCHESTRATE_LOAD_HDL_PROCESS

- Is an app driven integration that acts as a common service to load and import data to HCM cloud. This service is used by all the integrations that are pushing the data to HCM cloud application. This service is an asynchronous service.
- The integration calls stored procedure which will return the import job parameters, and other details for every integration that is processing the data.
- This service uses the HCM cloud connection to process the data files that are placed in the object storage by the dispatcher service. These files are created by another stored procedure.
- The status of load and import will be updated later by the set status service.
- The service is used for vendor integrations also to load and import the data.

HCM Common Inbound Framework (HFWK_IB) – External Integrations



Mount Sinai

This section describes the integrations that are used for the HCM common inbound framework for external vendor integrations. These integrations are used to load and import vendor data into HCM Cloud application. The integrations work based on the metadata configurations. Currently, it supports the load and import of data for the following external vendor integrations.

Integration Id	Description
INT_IB_1	AccuStaff Time Entry for MSQ Payroll Inbound
INT_IB_2	LaborWorks Time Entry Payroll Inbound
INT_IB_3	Clairvia to Cloud Payroll Inbound Interface
INT_IB_10	Cactus (Voluntary Faculty) for MSHS HCM Inbound
INT_IB_13	Kronos Payroll Inbound
INT_IB_14	Laborworks Absence for MSH MSSM HCM Inbound
INT_IB_15	Empower Student Census for MSSM HCM Inbound
INT_IB_16	New Innovation Create Emp Record for MSHS HCM Inbound
INT_IB_21	Windstar Inbound Interface
INT_IB_28	Acustaff Absence for MSQ Nursing HCM Inbound (Absence Entry)
INT_IB_29	Laborworks Absence Usage for MSH MSSM HCM Inbound
INT_IB_362	Acustaff Absence for MSQ Nursing HCM Inbound (Balance)
INT_IB_364	WAND Temporary Workers to Cloud HCM Inbound
INT_IB_430	Kronos Person Accrual Adjustment Inbound
INT_IB_45	TIAA CREF Contribution for MSHS Payroll Inbound
INT_IB_45BT	TIAA CREF Contribution for MSHS Payroll Inbound
INT_IB_53	Farmington Payroll Inbound Interface
INT_IB_54	Voya Life Deductions (Payroll) Interface to Cloud Payroll
INT_IB_69	Regulatory and Compliance for foreign nations for MSHS HCM Inbound
INT_IB_28K	KronosLi Absence Usage HCM Inbound (Absence Entry)
INT_IB_362K	KronosLi Absence Balance HCM Inbound (Balance)
INT_IB_113	Kronos Payroll Inbound-Direct HDL

Note:

When a new inbound integration to load and import into HCM Cloud application for external vendor integration is needed in the future, following objects need to be created or configured first.



Mount Sinai

- DB objects such as transaction tables, interface tables, sequences, indexes, PL/SQL packages, object types, and synonyms should exist in ATP DB layer.
- BI Reports should be configured for import jobs, parameter, and status of jobs.
- Metadata configuration needs to be defined for the new interface.
- OIC file server structure to hold the input, output, error, and other directories for inbound file processing.

Once these objects are created, common HCM inbound framework can be used to orchestrate the data load and import.

HFWK_IB_EXTERNAL_INTF_LAUNCHER_SERVICE

- It is a scheduled integration, which will be scheduled as per the time and day of in EST time zone.
- Below query is used to fetch the external integration details.
 - ```
SELECT INTF_ID,DESCRIPTION,ENTITY,
 SOURCE,TARGET,INTF_SCHEDULE_TIME,INTF_SCHEDULE_DAY ,active FROM cmfk_intf_metadata
 WHERE MODULE='HCM' and TARGET='Cloud HCM' and SOURCE_PROTOCOL ='File' and active='Y'
 order by intf_id
```
- If the INTF\_SCHEDULE\_TIME and INTF\_SCHEDULE\_DAY values match with OIC current time and day, specific matching criteria interface id will be passed to HFWK\_IB\_LIST\_FILE\_SERVICE integration. The value for INTF\_SCHEDULE\_TIME column of cmfk\_intf\_metadata table must be in UTC format since OIC server time is in UTC time zone.
- By using the schedule parameter P\_INTERFACE\_ID, it can be run on demand basis also for a single interface id.

## **HFWK\_IB\_LIST\_FILE\_SERVICE**

- It is an app driven integration called from HFWK\_IB\_EXTERNAL\_INTF\_LAUNCHER\_SERVICE integration.
- For the given interface id, OIC filer server source path is fetched from the HFWK\_OIC\_FILE\_SERVER\_LKP.
- Files are then pulled from OIC filer server. if there is no source file present, an email notification is sent to the email groups.
- If the file is present in the source file path and the extract type is HCM\_HDL\_MULTI\_PAIR\_TRANSFORM, then cmfk\_utility\_intf\_pkg.create\_list\_file\_pair\_pub procedure is called. Otherwise, cmfk\_utility\_intf\_pkg.create\_list\_file\_pair procedure is called.
- If there is no error received from the procedure call, HFWK\_IB\_DISPATCHER\_SERVICE integration is called. Otherwise, error email notification is sent, and source file is moved to respective interface error folder.

## **HFWK\_IB\_DISPATCHER\_SERVICE**

- This service is called from the List File Service with appropriate parameters such as interface\_id, stage\_flow\_id, and total number of files.
- Dispatcher service receives the count of files from the List file service.



- Parent flow id is maintained throughout the flow in all IB integrations and is used for tracking. It is derived by concatenating the instance id with file count.
- It gets the data by calling the INTERFACE\_POOL\_LIST procedure in CMFK.Utility.INTF\_PKG. If the procedure returns START, the file processing starts in the dispatcher. Otherwise, all source files are moved to ERROR and the flow stops after updating the transaction table.
- If the procedure returns START, then we get the metadata and enrichment details from ATP and update the list file table.
- The source file is then copied to the backup location and decrypted using the decryption service.
- The decrypted file is uploaded to the object storage and the source file is deleted from the source location.
- If the Enrichment is required for the interface, Load\_Cloud\_Data procedure is called followed by the Enrichment service.
- If data enrichment is not required, HDL Loader service is called, and ListFile Status is updated.

## **HFWK\_IB\_DECRYPTION\_SERVICE**

- Is an app driven integration that acts as a common service to decrypt the incoming files.
- It takes the encrypted input file name, location, and interface id returning the decrypted file name, status, and the location of decrypted file.
- This is a common service used by all integrations that require file decryption.
- To decrypt the files, OIC certificates are used which are pre-configured with private keys.
- Conditional logic with switch action is used to determine whether the interface needs file decryption.
- The output of the decrypted file is saved in the OIC file server. The output location is retrieved from the OIC lookup table.

## **HFWK\_IB\_ENRICHMENT\_SERVICE**

- It is an app driven integration, gets called from the HFWK\_IB\_DISPATCHER\_SERVICE if the enrichment\_flag is enabled in meta data table.
- Takes the interface\_id, flow\_id, file\_id, and source\_file\_name as inputs.
- Common wrapper procedure named hfwk\_utility\_intf\_pkg.wrapper\_get\_enrichment, is called which in turn calls the interface specific package to get the enrichment flag and job details.
- Until we get the enrichment flag as Y from above step, BIP report is called for every 1000 records in a while loop.



# Mount Sinai

- The BIP report output is being passed to the put\_enrichment procedure, which updates the BIP data in the interface specific tables.
- Once the enrichment of all the data is completed, hfwk\_utility\_intf\_pkg.wrapper\_create\_file procedure is called, which in turn calls the specific package to create HDL files in object storage.

## **HFWK\_IB\_CLOUDFILE\_ORCHESTRATE\_SERVICE**

- It is a scheduled integration run at specified intervals. This service is called only for integrations that need data enrichment. It is an intermediate service to call the HDL load and import service after the data enrichment is complete. This service is not applicable for integrations that do not need data enrichment.
- This integration acts as a polling service which will poll the CMFK\_IN\_CLOUD\_FILE\_TBL table using the cmfk\_utility\_intf\_pkg.interface\_pool\_cloud procedure. If the procedure returns the action value as START, the metadata information is fetched from the CMFK\_INTF\_METADATA table using cmfk\_utility\_intf\_pkg.get\_intf\_metadata\_all procedure.
- To get the cloud file details, cmfk\_utility\_intf\_pkg .get\_list\_file\_det procedure is called. If multiple cloud files are found, they are separated and passed as multiple file names to the HFWK\_IB\_ORCHESTRATE\_LOAD\_HDL\_PROCESS integration. Otherwise, file name for a single cloud file is passed.

## **HFWK\_IB\_HDL\_DATA\_SET\_STATUS\_SCHEDULER**

- This is a scheduled integration and runs at specified intervals.
- The service calls the load import status service which in turn updates the status of load and import jobs in PaaS ATP tables.
- It is primarily used for inbound integrations. This service calls two different services depending on the interface ids.
- Calls the service named HFWK\_IB\_HDL\_DATA\_SET\_STATUS\_PROCESS if the interface is INT\_IB\_PAAS\_1, INT\_IB\_PAAS\_2, and all the vendor integrations. Otherwise, it calls HFWK\_IB\_HDL\_DATA\_SET\_STATUS\_SUPP\_PROCESS service for INT\_IB\_PAAS\_3, INT\_IB\_PAAS\_4, INT\_IB\_PAAS\_5, and INT\_IB\_PAAS\_6 integrations.
- The integration processes the data for the records in CMFK\_INTF\_MASTER\_TBL table with status as N.

## **HFWK\_IB\_HDL\_DATA\_SET\_STATUS\_PROCESS**

- Is an app driven integration, which will update the status of the HCM load and import process through various procedures, reports and HCM cloud services for all the interfaces expect INT\_IB\_PAAS\_3, 4, 5 and 6.
- This integration acts as a polling service which will poll the CMFK\_INTF\_MASTER\_TBL table using procedure CMFK\_INTF\_MASTER\_PKG.INTERFACE\_POOL\_MASTER. The service will read the data from CMFK\_INTF\_MASTER\_TBL to process the data.



# Mount Sinai

- The integration will retrieve the status of HCM load and import data job requests by running BI reports and updates the status in CMFK\_INTF\_MASTER\_TBL.
- This integration will also send a detailed report on the load and import status via the notification service to the email groups defined in meta data.

## **HFWK\_IB\_HDL\_DATA\_SET\_STATUS\_SUPP\_PROCESS**

- Is an app driven integration which will update the status of the HCM load and import process through stored procedures for INT\_IB\_PAAS\_3, INT\_IB\_PAAS\_4, INT\_IB\_PAAS\_5, and INT\_IB\_PAAS\_6 integrations.
- This integration acts as a polling service which will poll the CMFK\_INTF\_MASTER\_TBL table using CMFK\_INTF\_MASTER\_PKG .INTERFACE\_POOL\_MASTER\_SUPP procedure.
- The integration will extract the status of HCM load and import data job request by running BI reports and updates the status in CMFK\_INTF\_MASTER\_TBL table.
- This integration will also send a detailed report on the load and import status via the notification service to the email groups defined in meta data.

## **Bank Integrations Outbound**

This section describes the common services used for Bank Integrations.

| Integration Id | Description                                   |
|----------------|-----------------------------------------------|
| INT_IB_49      | Bank Positive Pay for MSHS Payroll Outbound   |
| INT_IB_50      | Bank Direct Deposit for MSHS Payroll Outbound |

Following two integrations are used to send payroll related data files to JPMC, and TD banks.

## **HFWK\_OB\_PAYROLL\_SERVICE**

- It is an app driven integration.
- Calls the stored procedures to retrieve the meta data for the above interfaces, and the job details.
- Run the BI report to get the UCM ID of the Positive Pay and Direct deposit files generated by the HCM SaaS business process.
- Files are downloaded from UCM by passing UCM ID.
- HFWK\_OB\_ENCRYPTION\_DECRYPTION\_SERVICE will be called to decrypt, sign, and encrypt the data files.
- Private key configured in OIC is used to decrypt the file. The bank public keys are used to encrypt the file, and then signing is done using the MSHS private key.

## **HFWK\_OB\_ENCRYPTION\_DECRYPTION\_SERVICE**



- This integration takes the input file that needs to be decrypted, encrypted, and signed.
- File downloaded from UCM is decrypted using the Private Key configured in OIC.
- Depending on the interface id, files are decrypted, encrypted, and signed. For TD bank, the file is not signed.
- Encrypted file will be transferred to bank's FTP server location.

## **HFWK\_OB\_POSITIVE\_EFT\_LAUNCHER**

- This integration is the wrapper service to submit the integrations for Positive pay and EFT payments.
- It is created to avoid the end users from providing the Positive Pay or EFT payment job names.
- Check boxes are provided as inputs to select the Positive pay or EFT payments file processing.
- Flow instance id to get the UCM file is one of the inputs for this service which must be entered. This is case sensitive, and input must be entered as is that was generated in HCM process. Also, it won't take care of extra spaces in the string.
- Calls the Positive Pay and/or EFT payments jobs to send the files to the bank. It essentially calls the HFWK\_OB\_PAYROLL\_SERVICE.

## **Fund Number Approvers Synchronization Process**

This section describes the Fund Number Approvers sync process. The process takes a .txt file queued in OIC file server, parses the file, performs additional data enrichment, and loads the data in PaaS ATP DB table. The synchronization is done using the following integrations.

## **HFWK\_LOAD\_FUND\_APPROVERS\_LAUNCHER**

- This is a scheduled integration which runs at specified intervals.
- Polls for the incoming file in the input directory for Fund Number approvers sync in OIC file server.
- If a file is not found with the current date, error is thrown, and notification is sent to the email groups. Otherwise, HFWK\_LOAD\_FUND\_APPROVER service is called to process the file.

## **HFWK\_LOAD\_FUND\_APPROVER\_SERVICE**

- It is an app driven orchestration.
- Input file received is downloaded to a staging location in OIC.
- The semicolon delimited approver emails data in the original input file is broken into multiple lines. The resulting enriched data file is written to the OUTPUT directory in OIC file server.
- The expanded file with additional data enrichment is parsed and a stored procedure is called to orchestrate the data load in PaaS ATP DB table.
- Service calls a stored procedure to insert a control table record.
- The file with enriched data is moved to the backup location in OIC file server.

## **HFWK\_LOAD\_FUND\_APPROVERS\_NOTIFICATION\_SERVICE**



- It is a scheduled orchestration and run at specified intervals.
- Service calls a stored procedure to get the load status.
- If the load is completed successfully, control table record is updated to Y for status, and a notification is sent to the user groups. Otherwise, a failure notification is sent.

## Error Handling & Logging

- In case of any errors, the tracking/logging tables will be updated, and notification will be sent to the configured email group. Same logger tables that are used for SOA implementation is used by the OIC framework. Please refer to appropriate documents created for those efforts.
- Data will be synchronized up in the next run.
- User can manually submit the integration by passing the appropriate parameters as needed for the integration.
- All integrations use the common logging, and notification services.
- Errors are handled using scope level activity and global handler flow in OIC. If errors are not caught inside the scope level activity, it gets caught in the global handler.
- Log messages are written into activity tables in PaaS ATP DB layer using a stored procedure. These activity details for different stages of processing are then queried by dashboard VBCS application.

## HFWK\_DATASYNC\_NOTIFICATION\_SERVICE

- This service is used to send emails to the user groups that is configured in metadata.
- If the INTF\_ID is passed from the calling service, this integration gets the Email distribution list from ATP using CMFK\_NOTIFICATION\_DTL\_PKG.PROCESS\_MAIN\_PRC procedure.
- If the procedure does not return the distribution list, Email To will be populated from the Lookup.
- If the Calling service is not passing the INTF\_ID, it must pass the value for EMAIL\_TO.
- Attachments will be prepared only if the file content and name is present in the request payload.
- At the end, Email will be sent depending in the NOTIF\_TYPE value. S-Success, E-Error, W-Warning, and VBCS.

## HFWK\_DATASYNC\_LOGGING\_SERVICE

- It is an app driven orchestration, and a common service used by all integrations.
- Orchestrates a stored procedure to create log messages in the activity details table during various stage of processing.
- This also sends the notification using HFWK\_DATASYNC\_NOTIFICATION\_SERVICE if the Notification request flag parameter is passed a value.



## **HFWK\_DATASYNC\_DELETION\_SERVICE**

- This service is used to delete a file from object storage or OIC file server.
- Request payload- { "bucketName" : "OIC\_DEV4\_INBOUND", "objectName" : "Element.dat", "fileName" : "Element.dat", "filePath" : "/MSHS\_OIC\_ROOT/MSHS\_OIC\_INBOUND/ARCHIVE" }
- If bucket name and object name are passed, it will delete from the object storage.
- If the file name and file path are passed as parameters, it will delete from OIC file server.

## **Other Integrations**

### **HFWK\_INTERFACETABLE\_DATA\_PURGE\_SERVICE**

- This integration purges the interface tables such as CMFK\_INTF\_MASTER\_TBL, CMFK\_INTF\_ACTIVITY\_DETAILS tables for the given interface id.
- Should be run cautiously and as needed only.

### **HFWK\_DATASYNC\_UPDATE\_SCHEDULE**

- This will update the time schedule in meta data tables before the daylight time/Wintertime change kicks in. **Example:** During Winter-time zone period, the time difference between UTC and EST is 5 hours. Similarly, in Daylight time zone period, the time difference between UTC and EST is 4 hours. To handle this, we need a process that will update the schedule time for all interfaces that use the automated OIC schedules.
- It is run automatically once during the **second Sunday in March**, and **first Sunday in November**.
- For all OBs and IB that are automated to run using OIC schedules, the **intf\_schedule\_time** column in **cmfk\_intf\_metadata** table is updated twice a year to match the UTC time. Otherwise, these automated schedules won't run properly. **Example:** If an OB integration is run at **7 AM EST** Daily using automated schedule, the UTC time will be **11 AM UTC** in **cmfk\_intf\_metadata** table during daylight time zone period, and **12 PM UTC** in **cmfk\_intf\_metadata** table during wintertime zone period.

## **Security**

This section describes the service accounts used by integrations, OIC artifacts deployment automation, and vendors for external integrations.

Following table shows the service account used for the integrations.

| Name                  | Description                                                          | Type                    |
|-----------------------|----------------------------------------------------------------------|-------------------------|
| MSHS_OIC_FS_SERV_ACCT | Common service account used by all integrations to read/write files. | OIC File Server Account |



# Mount Sinai

Following table shows the service account used to automate the deployment of OIC objects such as integrations, connections, and lookups using Oracle Visual Builder Studio/Git tool for CI/CD.

| Name                  | Description                                                                                                          | Type                    |
|-----------------------|----------------------------------------------------------------------------------------------------------------------|-------------------------|
| MSHS_OIC_SERVICE_USER | Common service account used to automate the deployment of OIC objects to higher environment from a lower environment | OIC File Server Account |

Following table shows the service accounts used by the vendors for external integrations to place the inbound files in OIC file server for processing into HCM Cloud application.

| Name               | Description                                                                                                                                 | Type                    |
|--------------------|---------------------------------------------------------------------------------------------------------------------------------------------|-------------------------|
| mshs_tiaa_ib       | Account used by vendor to place files for TIAA related integrations. Only those folders appropriate to TIAA will be accessible.             | OIC File Server Account |
| mshs_kronos_ib     | Account used by vendor to place files for Kronos related integrations. Only those folders appropriate to Kronos will be accessible.         | OIC File Server Account |
| mshs_kronosli_ib   | Account used by vendor to place files for Kronos related integrations. Only those folders appropriate to Kronos will be accessible.         | OIC File Server Account |
| mshs_farmington_ib | Account used by vendor to place files for Farmington related integrations. Only those folders appropriate to Farmington will be accessible. | OIC File Server Account |
| mshs_wand_ib       | Account used by vendor to place files for WAND related integrations. Only those folders appropriate to WAND will be accessible.             | OIC File Server Account |
| mshs_voyalife_ib   | Account used by vendor to place files for Voyalife related integrations. Only those folders appropriate to Voyalife will be accessible.     | OIC File Server Account |



# Mount Sinai

|                   |                                                                                                                                                                                       |
|-------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| mshs_newinnov_ib  | Account used by vendor to place files for OIC File Server Account Newinnovation related integrations. Only those folders appropriate to NewInnovation will be accessible.             |
| mshs_lbrworks_ib  | Account used by vendor to place files for OIC File Server Account Laborworks related integrations. Only those folders appropriate to Laborworks will be accessible.                   |
| mshs_immitrac_ib  | Account used by vendor to place files for OIC File Server Account Immigration Tracker related integrations. Only those folders appropriate to Immigration Tracker will be accessible. |
| mshs_empower_ib   | Account used by vendor to place files for OIC File Server Account Empower related integrations. Only those folders appropriate to Empower will be accessible.                         |
| mshs_clairvia_ib  | Account used by vendor to place files for OIC File Server Account Clairvia related integrations. Only those folders appropriate to Clairvia will be accessible.                       |
| mshs_accustaff_ib | Account used by vendor to place files for OIC File Server Account Accustaff related integrations. Only those folders appropriate to Accustaff will be accessible.                     |
| mshs_cactus_ib    | Account used by vendor to place files for OIC File Server Account Cactus related integrations. Only those folders appropriate to Cactus will be accessible.                           |
| mshs_windstar_ib  | Account used by vendor to place files for OIC File Server Account Windstar related integrations. Only those folders appropriate to Windstar will be accessible.                       |

Following is the OCI service account used to connect to object storage/buckets.

| Name            | Description                                                                | Type                 |
|-----------------|----------------------------------------------------------------------------|----------------------|
| mshsnonproduser | Service account used by integrations to connect to object storage from OIC | OCI Service Account. |



## ATP DB Components

### Schema Name

MSHS\_SOAINST\_USER

### Description

Hosts all Metadata tables

Following is the table for database objects that have been changed for the OIC PaaS migration project. Other DB objects remain the same that are used in SOA implementation. Refer to the appropriate documents created for those efforts.

| DB Object Name                  | Type    | Purpose                                                                                                 | Comments                                     |
|---------------------------------|---------|---------------------------------------------------------------------------------------------------------|----------------------------------------------|
| HFWK.Utility_INTF_PKG           | Package | Wrapper package to call the existing PL/SQL packages for data enrichment and creating files for import. | New Package used in OIC implementation       |
| CMFK_INTF_METADATA              | Table   | Metadata configurations                                                                                 | Few columns are added for OIC implementation |
| MSHS_HCM_ALL_POSITIONS_STG_N5   | Index   | To improve Performance                                                                                  |                                              |
| MSHS_HCM_ALL_GRADES_STG_N5      | Index   | To improve Performance                                                                                  |                                              |
| MSHS_HCM_ALL_LOCATIONS_STG_N5   | Index   | To improve Performance                                                                                  |                                              |
| MSHS_HCM_JOB_VALID_GRADE_STG_N5 | Index   | To improve Performance                                                                                  |                                              |
| MSHS_HCM_ALL_JOBS_STG_N5        | Index   | To improve Performance                                                                                  |                                              |
| MSHS_HCM_ALL_ORG_STG_N5         | Index   | To improve Performance                                                                                  |                                              |
| MSHS_HCM_GRADE_LADDER_STG_N5    | Index   | To improve Performance                                                                                  |                                              |
| MSHS_HCM_ORG_HIER_STG_N5        | Index   | To improve Performance                                                                                  |                                              |
| MSHS_PAY_DEPT_COST_STG_N5       | Index   | To improve Performance                                                                                  |                                              |



|                                    |       |                        |  |
|------------------------------------|-------|------------------------|--|
| MSHS_HCM_GRADE_RATE_VAL_STG_N5     | Index | To improve Performance |  |
| MSHS_HCM_RATE_STG_N5               | Index | To improve Performance |  |
| MSHS_HCM_POSITION_HIERARCHY_STG_N5 | Index | To improve Performance |  |
| MSHS_HCM_SALARY_BASIS_STG_N5       | Index | To improve Performance |  |
| MSHS_HCM_SETID_STG_N5              | Index | To improve Performance |  |
| MSHS_HCM_USER_ROLES_STG_N5         | Index | To improve Performance |  |
| MSHS_HCM_AOR_SECURITY_STG_N5       | Index | To improve Performance |  |
| MSHS_HCM_PERSON_STG_N4             | Index | To improve Performance |  |
| MSHS_HCM_PERSON_ASSIGNMENT_STG_N4  | Index | To improve Performance |  |
| MSHS_HCM_CMP_SALARY_STG_N4         | Index | To improve Performance |  |
| MSHS_HCM_COST_INFO_STG_N4          | Index | To improve Performance |  |
| MSHS_HCM_EDMCS_CC_N3               | Index | To improve Performance |  |
| MSHS_HCM_ELEMENT_ENTRIES_N4        | Index | To improve Performance |  |
| MSHS_HCM_ELEMENT_MP_N4             | Index | To improve Performance |  |
| MSHS_HCM_GEOGRAPHIES_N4            | Index | To improve Performance |  |
| MSHS_HCM_EE_HDL_STG_N1             | Index | To improve Performance |  |
| MSHS_HCM_EEV_HDL_STG_N1            | Index | To improve Performance |  |
| MSHS_HCM_CA_HDL_STG_N1             | Index | To improve Performance |  |



|                                        |         |                                          |                  |
|----------------------------------------|---------|------------------------------------------|------------------|
| MSHS_HCM_CAA_HDL_STG_N1                | Index   | To improve Performance                   |                  |
| MSHS_HCM_EDMCS_ENTITY_STG_N3           | Index   | To improve Performance                   |                  |
| MSHS_HCM_EDMCS_NETASSET_CLASS_STG_N3   | Index   | To improve Performance                   |                  |
| MSHS_HCM_EDMCS_COST_CENTER_STG_N3      | Index   | To improve Performance                   |                  |
| MSHS_HCM_EDMCS_FUNCTIONAL_CLASS_STG_N3 | Index   | To improve Performance                   |                  |
| MSHS_HCM_EDMCS_FUND_TYPE_STG_N3        | Index   | To improve Performance                   |                  |
| MSHS_HCM_EDMCS_SITE_STG_N3             | Index   | To improve Performance                   |                  |
| MSHS_HCM.Utility.INTF.PKG              | Package | Minor change for performance improvement | Existing Package |
| MSHS_HCM.Payroll.Utility.PKG           | Package | Minor change for performance improvement | Existing Package |
| MSHS_HCM.EDMCS.INTF.PKG                | Package | Minor change for performance improvement | Existing Package |

## SaaS Components – BIP/HCM Extracts

### Extraction Logic

OIC integration framework uses the same HCM Extracts and BIP reports that are used in SOA Framework implementation. The details should be referred in TDDs created for the SOA Framework implementation.

### Data Models for BIP Report

OIC integration framework uses the same data models for BIP reports that are used in SOA Framework implementation. The details should be referred in TDDs created for the SOA Framework implementation.

## User and Security



| User                              | System                           | Roles                                                       |
|-----------------------------------|----------------------------------|-------------------------------------------------------------|
| MSHS_COREHR_INT_USER              | HCM Cloud                        | Same roles are used as in the SOA Framework implementation. |
| ejis.Service_edmcs                | EDMCS                            | This should be covered in another document                  |
| Service_Sinaicloud@mountsinai.org | HCM Cloud (PROD Service Account) | Same roles are used as in the SOA Framework implementation. |

## Open Closed Issues

### Open Issues

| ID | Issue | Resolution | Responsibility | Target Date | Impact Date |
|----|-------|------------|----------------|-------------|-------------|
| 1  |       |            |                |             |             |
|    |       |            |                |             |             |

### Closed Issues

| ID | Issue | Resolution | Responsibility | Target Date | Impact Date |
|----|-------|------------|----------------|-------------|-------------|
|    |       |            |                |             |             |
|    |       |            |                |             |             |



# Appendix – A

## OIC File Server Directory Structure – External Integrations

Following is the spread sheet used for the OIC file server directory structures to process the incoming vendor files. Vendors place the files in designated directory for the integration to poll for data files, and to process them in HCM cloud.



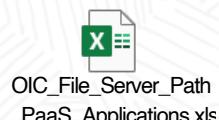
## OIC File Server Directory Structure – Fund Number Approvers Sync Process

Following is the spread sheet used for the OIC file server directory structures to process the incoming fund number approvers file.



## OIC File Server Directory Structure – Integrations for PaaS Applications

Following is the spread sheet used for the OIC file server directory structures to store the output files from Load and import process for the PaaS application integrations.





# Appendix – B

## Release Management - OIC Artifacts Deployment Automation

Oracle Visual Builder Studio tool is the code repository, and release management/version control tool for all custom builds created for the OIC PaaS migration activities. Git tool is integrated within the Oracle VB studio which is used to promote the builds to an OIC environment. Git tool for Windows is used to push/pull the files from local machine to Git Repository.

OIC objects such as integrations, connections, and lookups are deployed to higher environments using shell scripts in a Git job in Oracle VB studio. For example, the shell scripts export the builds from a development OIC environment, archives the files in Git repository, and promotes the builds from Git to a higher OIC environment. Connections, and integrations are automatically activated using the shell script. Also, the schedule for OIC jobs are automatically started.

Following document has screen shots for how the OIC software releases are managed using automation.



MSHS\_Visual\_Builder  
\_Studio\_OIC\_Git\_Repo

# Appendix – C

## Integration Testing Documents

This section describes the steps to test the integrations on an ad hoc basis.

### PaaS OB Integrations Testing Document



HCM\_PAAS\_OB\_Testing\_Steps.docx

### PaaS Vendor IB Integrations Testing Document



MSHS\_EXT\_INTEGRATION\_TESTING.docx

## PaaS IB integrations Testing Document

HCM\_IB\_PAAS\_TESTING\_STEPS.docx

## EDMCS IB Integrations Testing Document

MSHS\_EDMCS\_IB\_Testing\_Steps.docx

## Fund Approver Sync Process Testing Document

MSHS\_FUND\_NUMBERS\_USER\_ROLES\_SYNC.docx

Load\_Fund\_Approval\_Monitoring.docx

## Bank Integrations Testing Document

MSHS\_BankIntegrations\_Testing\_Modified.docx

# Appendix – D



**Mount  
Sinai**

## OIC Integration Job Schedules



MSHS\_OIC\_HCM\_IB\_  
OB\_EDMCS\_IB\_Sched

Following documents describe the scheduler configuration steps in OIC for HCM – PaaS, EDMCS – PaaS, and Vendor Inbound integrations. These integrations synchronize the master data into ATP tables from HCM and EDMCS applications and process inbound data into HCM from vendor applications which are run using automated schedules in OIC.



MSHS\_HCM\_OB\_Paa  
S\_Scheduler\_Configur



MSHS\_EDMCS INTF  
Scheduler\_Configurati



MSHS VEDNOR INTF  
SCHEDULING.docx

## Appendix – E



Mount  
Sinai

## OIC Storage Bucket Configuration

The screenshot shows a web browser window with the URL [https://www.oracle.com/cloud/sign-in.html?redirect\\_uri=https%3A%2F%2Fcloud.oracle.com%2F](https://www.oracle.com/cloud/sign-in.html?redirect_uri=https%3A%2F%2Fcloud.oracle.com%2F). The Oracle Cloud logo is at the top left, followed by the text "OCI". A navigation bar below includes links for Services, Solutions, Why OCI, Customers, Pricing, Learn, Developers, Support, and Marketplace. On the right, there are search, filter, and sign-in buttons. The main content area displays two promotional cards. The first card, titled "Launch a pre-built automated CI/CD pipeline for Kubernetes", features a circular icon with a steering wheel and text about simplifying and automating software development cycles. It has "Deploy now" and "View architecture" buttons. The second card, titled "Red Bull Racing Workshop", shows a Formula 1 race car and text about learning predictions with analytics and machine learning models. The desktop taskbar at the bottom shows various application icons.



Cloud Infrastructure

## Single Sign-On (SSO)

We have detected that your tenancy has been federated to another Identity Provider.

Select your Identity Provider below.

ejisdev4

ejisdev4

ejisdev5

ejisdev11

oracleidentitycloudservice

ejisdev2

[Terms of Use](#) | [Privacy](#) | [Cookie Preferences](#)

COPYRIGHT © 2016-2022, ORACLE AND/OR ITS AFFILIATES. ALL RIGHTS RESERVED.

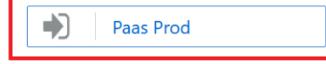
Password

[Sign In](#)

Need help signing in? [Click here](#)

[Cookie Preferences](#)

Or sign in with




# Mount Sinai



Sign in

If you're a Hospital user Sign-in using  
MSNYUHEALTH\UserName  
OR  
with your Email Address,  
first.last@m Mountsinai.org

If you're a School user Sign-in using

X ORACLE Cloud Cloud Classic > Search resources, services, documentation, and Marketplace US East (Ashburn) ▾

Storage

Block Storage File Storage Related services

- Block Volumes File Systems Data Transfer
- Block Volume Backups Mount Targets Limits, Quotas and Usage
- Block Volume Replicas
- Volume Groups Object Storage & Archive Storage Help
- Volume Group Backups Buckets
- Volume Group Replicas
- Backup Policies

Object Storage & Archive Storage

Buckets

Getting Started on OCI

Block Storage

File Storage

Object Storage

Archive Storage

Storage Gateways

Service Limits

Copyright © 2022, Oracle and/or its affiliates. All rights reserved.



# Mount Sinai

List Scope

Compartment  mshs (root)/MSHSNONPRODDB

Tag filters [add](#) | [clear](#)  
no tag filters applied

| Name                          | Default Storage Tier | Visibility | Created                         |  |
|-------------------------------|----------------------|------------|---------------------------------|--|
| <a href="#">ADBW-TEST</a>     | Standard             | Private    | Tue, Sep 15, 2020, 17:20:29 UTC |  |
| <a href="#">DEV2_INBOUND</a>  | Standard             | Private    | Mon, Aug 10, 2020, 18:38:54 UTC |  |
| <a href="#">DEV2_OUTBOUND</a> | Standard             | Private    | Mon, Aug 10, 2020, 18:39:10 UTC |  |
| <a href="#">DEV3_INBOUND</a>  | Standard             | Private    | Mon, Aug 10, 2020, 18:39:20 UTC |  |
| <a href="#">DEV3_OUTBOUND</a> | Standard             | Private    | Mon, Aug 10, 2020, 18:39:29 UTC |  |
| <a href="#">DEV4_INBOUND</a>  | Standard             | Private    | Thu, Mar 26, 2020, 10:45:26 UTC |  |

[Terms of Use and Privacy](#) [Cookie Preferences](#)

Copyright © 2022, Oracle and/or its affiliates. All rights reserved.



|                                   |          |         |                                 |  |
|-----------------------------------|----------|---------|---------------------------------|--|
| <a href="#">DEV5_OUTBOUND</a>     | Standard | Private | Sun, Jan 19, 2020, 12:08:57 UTC |  |
| <a href="#">MSHSNONPRODBUCKET</a> | Standard | Private | Wed, Dec 18, 2019, 16:48:19 UTC |  |
| <a href="#">OIC_CFWK_BUCKET</a>   | Standard | Private | Thu, Apr 28, 2022, 12:39:17 UTC |  |
| <a href="#">OIC_DEV2_INBOUND</a>  | Standard | Private | Thu, Jun 16, 2022, 15:23:30 UTC |  |
| <a href="#">OIC_DEV2_OUTBOUND</a> | Standard | Private | Thu, Jun 16, 2022, 15:23:19 UTC |  |
| <a href="#">OIC_DEV4_INBOUND</a>  | Standard | Private | Wed, May 4, 2022, 19:39:05 UTC  |  |
| <a href="#">OIC_DEV4_OUTBOUND</a> | Standard | Private | Wed, May 4, 2022, 21:39:49 UTC  |  |

[Terms of Use and Privacy](#) [Cookie Preferences](#)

Copyright © 2022, Oracle and/or its affiliates. All rights reserved.

# Appendix – F



**Mount  
Sinai**

## OIC File Server/SFTP Accounts Security Setups

Attached document guides the steps to configure security for the SFTP/OIC File server accounts used for vendor integrations.



MSHS\_OIC\_FileServer\_Sec\_Config\_Setups.doc

## Appendix – G

### OIC Certificates Setup for PGP Keys

Following document shows the steps to configure certificates in OIC



MSHS\_OIC\_Certificates\_Setup.docx

## Appendix – H

### Documentation

#### HCM Flow Action Web Service Documentation

<https://docs.oracle.com/en/cloud/saas/human-resources/22c/oeswh/flowactions-d19961e1218.html#flowactions>

#### UCM Service Documentation:



[https://docs.oracle.com/cd/E21764\\_01/doc.1111/e10807/c12\\_web\\_services.htm#CSSDK816](https://docs.oracle.com/cd/E21764_01/doc.1111/e10807/c12_web_services.htm#CSSDK816)

[https://<Cloud\\_host\\_name>:<Cloud\\_port>/idcws/GenericSoapPort?WSDL](https://<Cloud_host_name>:<Cloud_port>/idcws/GenericSoapPort?WSDL)

#### **BI Publisher Report Services:**

[https://<cloud\\_host\\_name>:<cloud\\_port>/xmlpserver/services/ExternalReportWSService?wsdl](https://<cloud_host_name>:<cloud_port>/xmlpserver/services/ExternalReportWSService?wsdl)

#### **OIC HCM Adapter Documentation**

<https://docs.oracle.com/en/cloud/paas/integration-cloud/hcm-adapter/oracle-hcm-cloud-adapter-capabilities.html>

#### **Oracle Object Storage Documentation**

[https://docs.oracle.com/en-us/iaas/Content/Object/Tasks/managingbuckets.htm#Managing\\_Buckets](https://docs.oracle.com/en-us/iaas/Content/Object/Tasks/managingbuckets.htm#Managing_Buckets)

#### **OIC User Guide**

<https://docs.oracle.com/en/cloud/paas/integration-cloud/integrations-user/index.html>

#### **OIC File Server Documentation**

<https://docs.oracle.com/en/cloud/paas/integration-cloud/file-server/file-server.html>

#### **OIC REST Adapter Documentation**

<https://docs.oracle.com/en/cloud/paas/integration-cloud/rest-adapter/index.html>

#### **OIC SOAP Adapter Documentation**

<https://docs.oracle.com/en/cloud/paas/integration-cloud/soap-adapter/oracle-soap-adapter-capabilities.html>

#### **OIC DB Adapter Documentation**

<https://docs.oracle.com/en/cloud/paas/integration-cloud/database-adapter/oracle-database-adapter-capabilities.html>

#### **Tortoise Git Documentation**

<https://tortoisegit.org/download/>

#### **OIC REST API Documentation**



<https://docs.oracle.com/en/cloud/paaS/integration-cloud/rest-api/index.html>