IFRS17 Data Ingestion Architect Guidelines

Version 1.0

Prepared by : Manash Nayak

01 Jun 2020

PCA IFRS17 Detail Azure Architect Design for Data & ETL Testing Environment

Document Control

Deliverable Title:	IFRS17 Detail Azure Architect Design		
Owner:	Manash Nayak	Deliverable ID:	SDLXXX
Deloitte Reviewer:	XXXX	Review Date:	
Prudential Reviewer:	William Chan	Review Date:	
Prudential Approver:		Sign-Off Date:	[Insert sign-off evidence here]

Purpose of this Document

This document serve as the detail Architectural document for the LBU Data ingestion mechanism for the IFRS17(ETL&DATA) project. This document contains the necessary technical details required for LBU's as referral on the overall technical architecture design for IFRS-17 data & ETL operation on the overall technology stacks .

Version Control

Date	Version	Version Development	Author
20-Jan-2020	V1.0	Draft	Manash Nayak
		Submitted version for review	

IFRS17 Data ingestion Architect

- Guiding principles for Data Movement,Data Stores and Data Flow in the Azure SQL Data Warehouse

  o Data solution Architecture Data Flow

  o Data Loading design decision

  o Interface specs & control file specs

  o Architect for Data Movement to Azure Data Lake (cloud)

  o Data File transmission

  o Data movement in different layers .

  o ETL Audit, Balance and Control Framework (ABC Framework)

  o Centralised Devops CI/CD & Control-M Approach
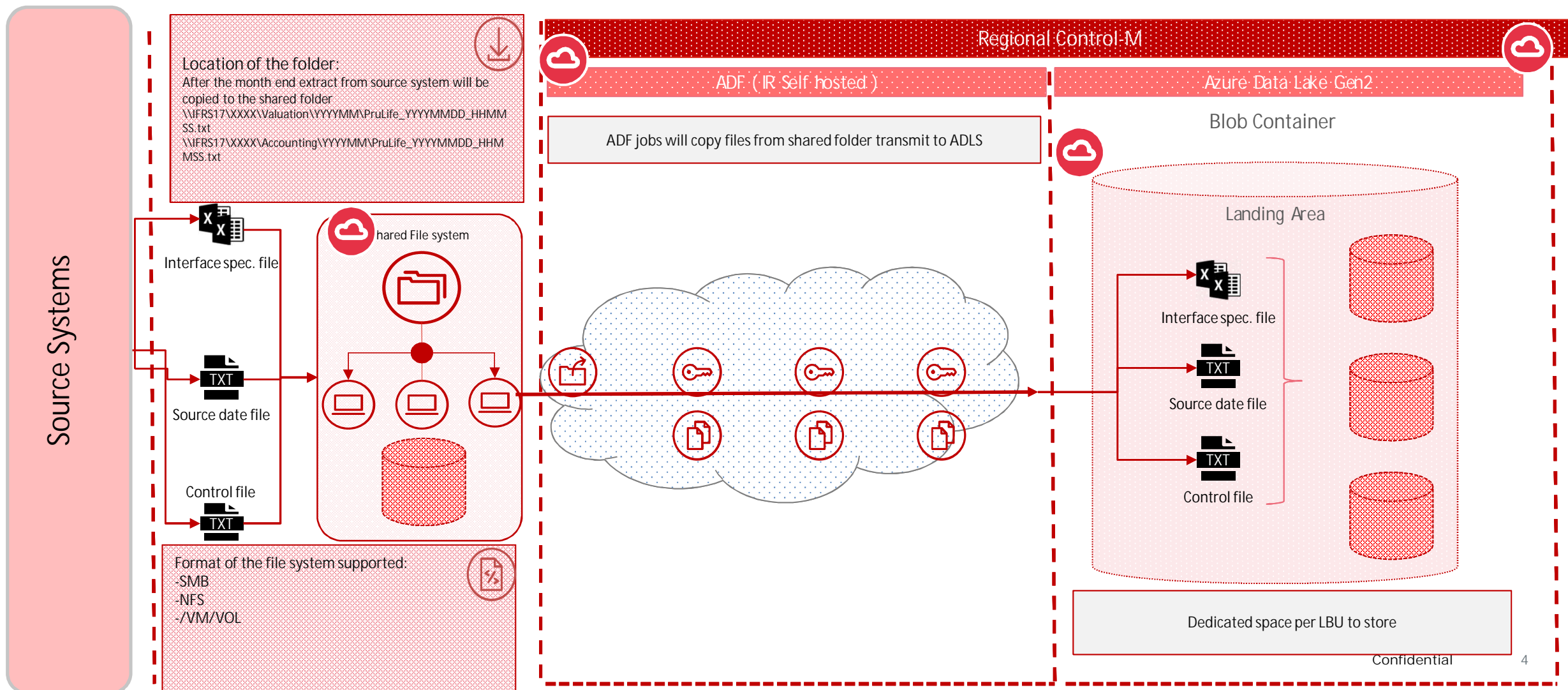
File transmission

- Architect for Data Movement to Azure Data Lake (cloud)

Azure Data Factory Self hosted-IR (Regional software) will be used to Copying of source system extracts from shared File-system to Azure Data Lake Blob container.
Source system program will generate a completion of file once the file transmission to Shared File system completes complete_batchname_batchdate.txt / complete_filename_batchdate.txt (ex : complete_PHKL_PRULIFE_GL_DDMMYYYYHH:MM:SS.txt) with status "Y")
Self hosted-IR (Regional software) will check the file completeness via a script injected to ADF pipe-line .
Once the above condition met the files will be Pull from shared FS to respective ADLS Blob container/folder .

Source Systems

Location of the folder:
After the month end extract from source system will be copied to the shared folder
\\IFRS17\XXXX\Valuation\YYYYMM\PruLife_YYYYMMDD_HHMMSS.txt
\\IFRS17\XXXX\Accounting\YYYYMM\PruLife_YYYYMMDD_HHMMSS.txt

Interface spec. file

Source date file

Control file

Shared File system

Format of the file system supported:
-SMB
-NFS
-/VM/VOL

Regional Control-M

ADF (IR Self hosted)

ADF jobs will copy files from shared folder transmit to ADLS

Azure Data Lake Gen2

Blob Container

Landing Area

Interface spec. file

Source date file

Control file

Dedicated space per LBU to store

Data solution Architecture and Layer wise Data Flow

- End to End Data Flow within Azure Data Lake and Azure SQL Data warehouse
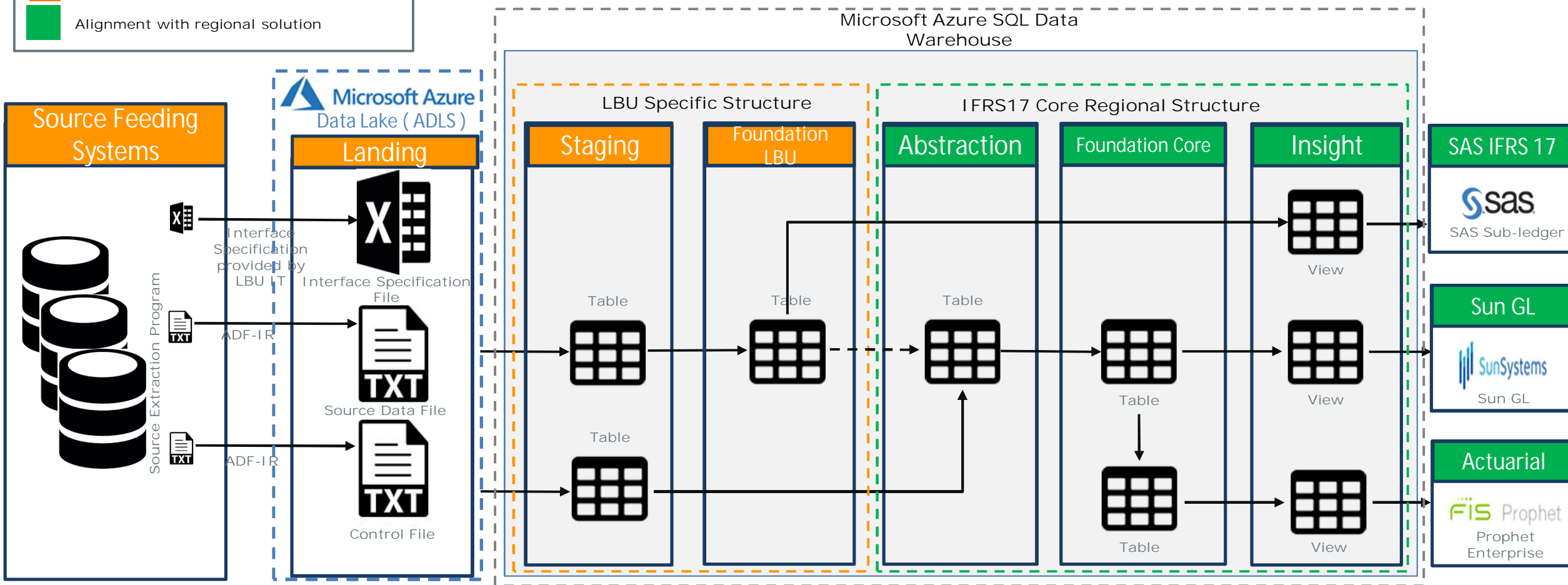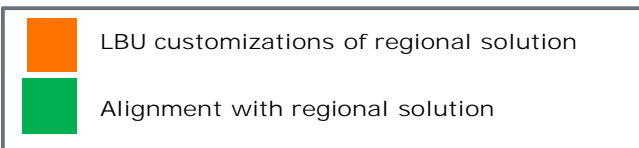
Control file spec

- Control file need to be send by LBU to match the completeness of file transmission as well as adopting Audit , Balance & Control
- The spec to follow below .

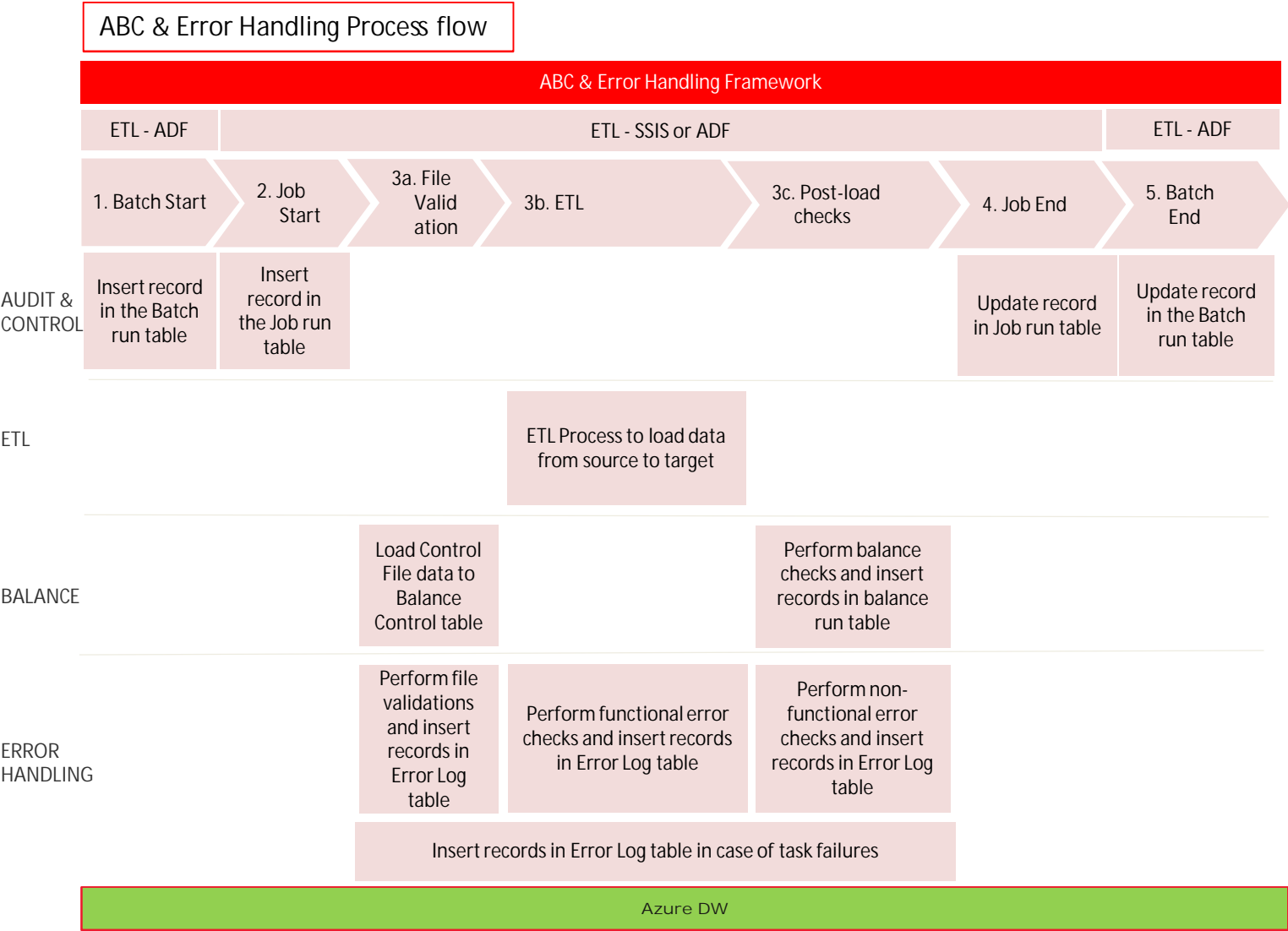| Field Name	Data Type	Length	Is Mandatary	Sample Value	Data Format	Descriptions
SRC_SYS_NAME	varchar	50	Y	PruLife	string	Name of the source system from input Data is coming for IFRS17 like Prulife, DMTM etc.
FILE_NAME	varchar	100	Y	PHKL_PRULIFE_GL_DDMMYYYYHH:MM:SS.txt	string	File Generation time . Name of the data file the interface control file corresponds to should be *.txt
BATCH_DT	Date		Y	02/05/2020	DD/MM/YYYY	Date when the batch is created.
SRC_FILE_GEN_DTTM	Datetime2		Y	02/05/2020 01:24:30	DD/MM/YYYY HH:MM:SS	Generation date & Time up to millisecond of the File
RECORD_COUNT	Bigint		Y	24566	99999999999999	Total record count in the data file
TOTAL_CR_AMT	decimal	28,6	N	optional (-12345.6789)	-9999999999.99999	Total CR Amount in the data file as per Data Model standard
TOTAL_DR_AMT	decimal	28,6	N	optional (12345.6789)	9999999999.99999	Total DR Amount in the data file
TOTAL_AMT	decimal	28,6	N	optional(12345.6789)	9999999999.99999	Total Amount in the data file
TRANSFER_MODE	varchar	10	Y	SYS or MAN	string	Mode indicate the type of data transfer (MAN/SYS)
FULL_SET_IND	varchar	1	Y	F or D	srting	F OR D 'F' for Full set , 'D' for Delta
Example (from PHKL) -						
PHKL_PRULIFE_GL_TRAN_BENEFIT_DDMMYYYYHH:MM:SS.txt,31/12/2019,04/02/2020 11:34:11,25,-000000010992921200,000000010992921200,0000065165231237125371,SYS,D						

Data Loading Design Decision :

- Step-1 : The Data loading from Landing(ADLS Blob storage) -> Staging Layer(Data Warehouse) . This includes the ability to data retention in Azure DW staging layer .

- Step-2 : Data will be loaded from Staging -> Abstraction layer .

- To accomplish above Azure Data Factory (ADF) is the recommended solution .

- Each LBU will build the ADF pipe-line of their own and below design applies to ID , TH , PH

- Staging schema and data structure ETL-1,2,4 (ID , TH , PH) to be prepared by Regional Data team
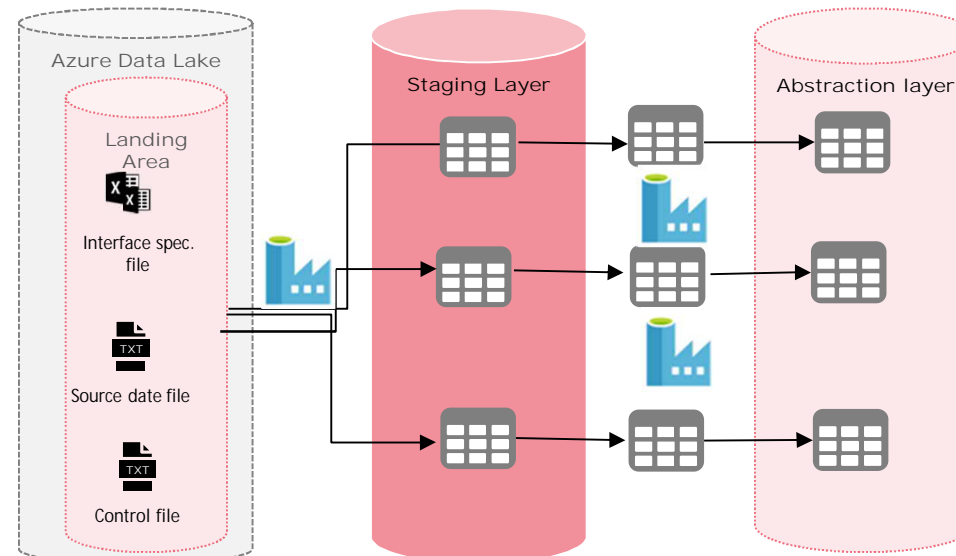
Audit Balance Control (ABC) & Error Handling Framework – LBU roll out approach

- ABC & Error Handling framework for IFRS17 is a set of Data model, SQL Stored procedures created which are mostly ETL tool agnostic and upon integration with ETL process helps us to maintain job execution statistics, rejects & error handling records

ABC & Error Handling Process flow

ABC & Error Handling Framework

| ETL - ADF | ETL - SSIS or ADF | ETL - ADF |

1. Batch Start → 2. Job Start → 3a. File Validation → 3b. ETL → 3c. Post-load checks → 4. Job End → 5. Batch End

AUDIT & CONTROL
- Insert record in the Batch run table
- Insert record in the Job run table
- Update record in Job run table
- Update record in the Batch run table

ETL
- ETL Process to load data from source to target

BALANCE
- Load Control File data to Balance Control table
- Perform balance checks and insert records in balance run table

ERROR HANDLING
- Perform file validations and insert records in Error Log table
- Perform functional error checks and insert records in Error Log table
- Perform non-functional error checks and insert records in Error Log table

Insert records in Error Log table in case of task failures

Azure DW

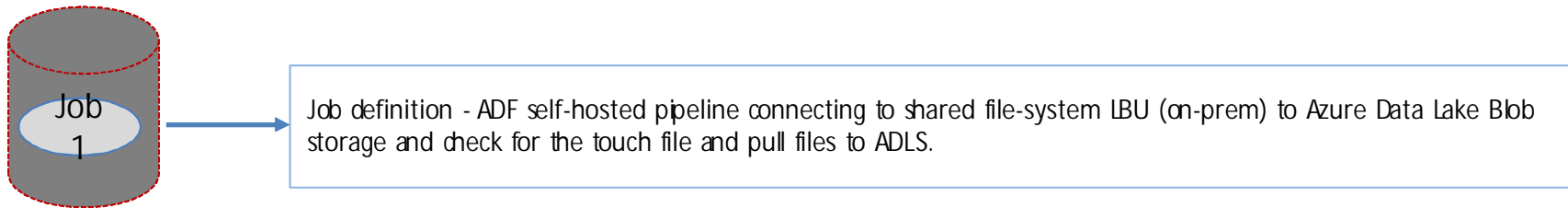|	ABC Framework implementation	
	Framework Database Objects	Framework Integration
Regional	Reusable components built using Azure DW	Reusable components built using SSIS & ADF
LBUs not using SSIS	No customizations needed for Azure DW	Customize Reusable components using ADF**

**Assumption - use ADF only

Guidelines Data Movement to Azure Data Lake (cloud)

- Azure Data Factory Self hosted-IR (Regional software) will be used to Copying of source system extracts from shared File-system to Azure Data Lake Blob container .
- Network teams on LBU & Regional work together to setup communication between on-prem to cloud .
- LBU's to work on shared area to copy extracts from source systems .
- LBU's responsibilities to create ADF pipe-lines copying files .
- LBU's responsibility to create control M job definitions .

Job 1

Job definition - ADF self-hosted pipeline connecting to shared file-system LBU (on-prem) to Azure Data Lake Blob storage and check for the touch file and pull files to ADLS.

File transmission completeness in Azure Data Lake Service

Job 2 will be scheduled on Azure control M via scripting to confirm the completeness of file transmission

Job 2

scripting will be scheduled on ADLS to check the completeness of transmitted files
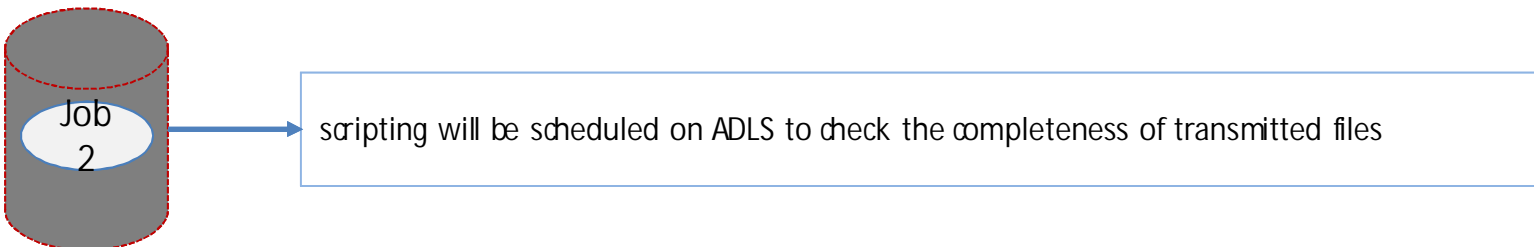
Data Loading mechanism

Data Loading from Landing(ADLS) to staging will be done via ADF .
Job 3 will be scheduled on Azure control M via scripting to load data .

Data Loading from staging to Abstraction will be done via ADF .
Job 4 will be scheduled on Azure control M via scripting to load data from staging to Abstraction

Build ADF pipe-line and control M job definition will be LBU responsibilities

```
Job 3  ────────────►  Data Loading Job Landing -> staging
```
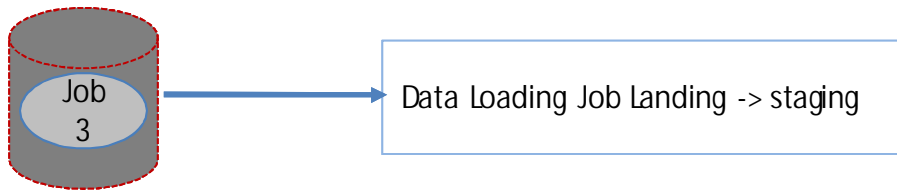
Job dependency :

Job dependencies will be defined on Control M JSON file

```
Job1  ◄────  Job2  ◄────  Job3  ◄────  Job4
```
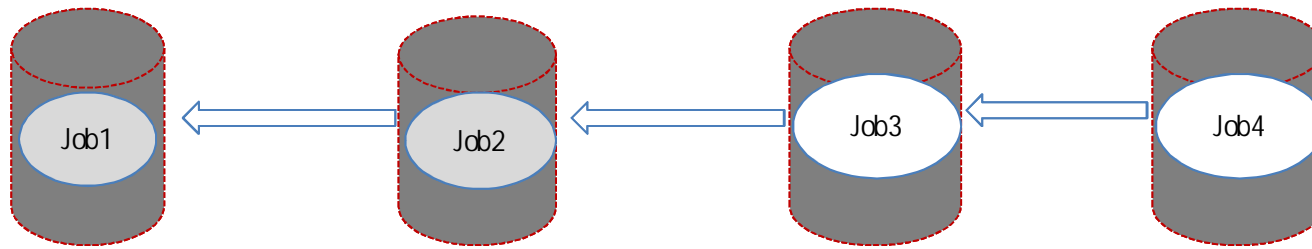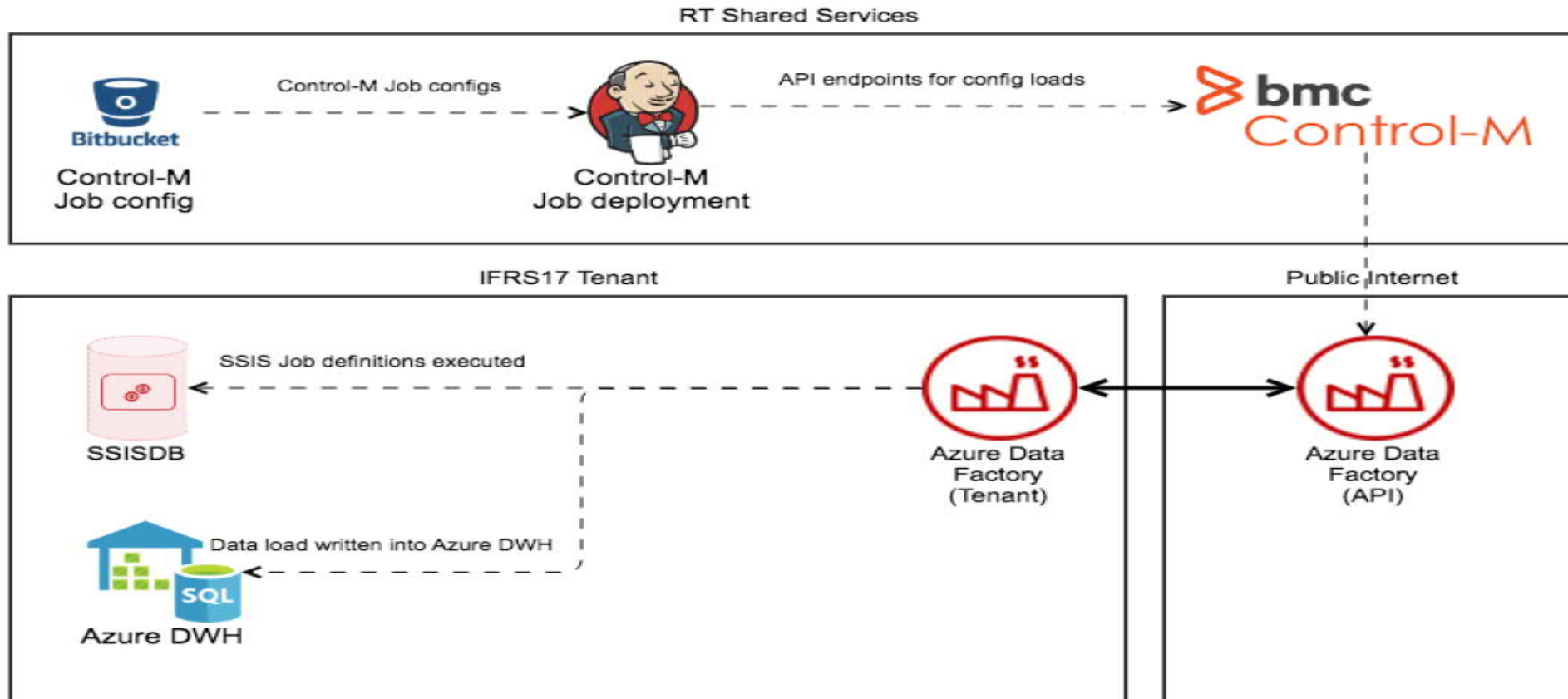
Centralised Devops CI/CD & Control-M Approach

ADF pipe-lines will be checked into Bit-bucket repo .
Control M JSON files will be checked into Bit-bucket repo .
Scripts will be checked into Bit-bucket repo .

Repos will be provided to all LBU's to check-in the code .

Centralized Jenkin pipe-line will take care of deployment and releases .

RT Shared Services

Bitbucket
Control-M Job config

Control-M Job configs

Control-M Job deployment

API endpoints for config loads

bmc Control-M

IFRS17 Tenant

Public Internet

SSIS Job definitions executed

SSISDB

Azure Data Factory (Tenant)

Azure Data Factory (API)

Data load written into Azure DWH

SQL
Azure DWH

Responsibility Matrix

Task	Responsible	Comment
Communication	William	William and Manash to communicate
Data Extraction Source systems	LBU	LBU team to work
Preparing Shared File-system	LBU	LBU team to work with LBU infra team
Network connectivity between Shared File-system and Azure cloud	LBU Network team and Regional Network team	Regional Infra team will help
Staging Layer Data structure	Regional team	Data Architect to prepare
Data loading Landing(ADLS) -> Staging Layer (DW)	LBU ETL team	Regional ETL team will show case sample
Data loading from Staging(DW) -> Abstraction layer(DW)	LBU ETL team	Regional ETL team will show case sample
Audit ,control Framework	LBU ETL team	Regional ETL team will customize framework wrapper to adopt ADF and show case to LBU .
Balanace & Error handling	LBU ETL team	Process and control team to provide requirement to LBU's
Control M & Devops process	LBU ETL team	Regional team will help showcasing the POC .