Analytics and Solution Platform-Migrating Historical Raw Data to Cloud and Retrieving Data using File Access Service,ETLs

Submitted By

Patel Namrata J

16MCEN12

DEPARTMENT OF INFORMATION TECHNOLOGY

INSTITUTE OF TECHNOLOGY

NIRMA UNIVERSITY

AHMEDABAD-382481

MAY 2018

Analytics and Solution Platform-Migrating Historical Raw Data to Cloud and Retrieving Data using File Access Service,ETLs

Major Project

Submitted in partial fulfillment of the requirements

for the degree of

Master of Technology in Computer Science and Engineering (Networking Technologies)

Submitted By

Patel Namrata J

(16MCEN12)

Guided By

Prof. Sapan H. Mankad

DEPARTMENT OF INFORMATION TECHNOLOGY

INSTITUTE OF TECHNOLOGY

NIRMA UNIVERSITY

AHMEDABAD-382481

MAY 2018

Certificate

This is to certify that the Major Project entitled **"Analytics and Solution Platform-Migrating Historical Raw Data to Cloud and Retrieving Data using File Access Service,ETLs"** submitted by **Patel Namrata J (Roll No:16MCEN12)**, towards the partial fulfillment of the requirements for the award of degree of Master of Technology in Computer Science and Engineering (Networking Technologies) of Nirma University, Ahmedabad, is the record of work carried out by her under my supervision and guidance. In my opinion, the submitted work has reached a level required for being accepted for examination. The results embodied in this major project, to the best of my knowledge, haven't been submitted to any other university or institution for award of any degree or diploma.

Prof. Sapan H. Mankad
Guide & Assistant Professor,
IT Department,
Institute of Technology,
Nirma University, Ahmedabad.

Dr. Gaurang Raval
Associate Professor,
Coordinator M.Tech - CSE (NT)
Institute of Technology,
Nirma University, Ahmedabad

Dr. Madhuri Bhavsar
Professor and Head,
IT Department,
Institute of Technology,
Nirma University, Ahmedabad.

Dr. Alka Mahajan
Director,
Institute of Technology,
Nirma University, Ahmedabad

Statement of Originality

I, **Patel Namrata J**, **16MCEN12**, give undertaking that the Major Project entitled "**Analytics and Solution Platform-Migrating Historical Raw Data to Cloud and Retrieving Data using File Access Service,ETLs**" submitted by me, towards the partial fulfillment of the requirements for the degree of Master of Technology in **Computer Science and Engineering (Networking Technologies)** of Institute of Technology, Nirma University, Ahmedabad, contains no material that has been awarded for any degree or diploma in any university or school in any territory to the best of my knowledge. It is the original work carried out by me and I give assurance that no attempt of plagiarism has been made.It contains no material that is previously published or written, except where reference has been made. I understand that in the event of any similarity found subsequently with any published work or any dissertation work elsewhere; it will result in severe disciplinary action.

Signature of Student
Date:
Place:

Endorsed by
Prof. Sapan H. Mankad
(Signature of Guide)

Acknowledgements

First and foremost, sincere thanks to **Prof. Sapan H. Mankad**, Assistant Professor, Information Technology Department, Institute of Technology, Nirma University, Ahmedabad for his valuable guidance and continual encouragement throughout this work.

It gives me immense pleasure in expressing thanks and profound gratitude to **Mrs. Geetha Muddana**, Project Manager, Philips Innovation Campus, Bangalore.She gave me chance for working under ASP project.

I would like to thank to **Mr. Poovaramanan Kannan** , Architect, Philips Innovation Campus, Bangalore for his valuable guidance.He has given me workflow and pipeline to work on project work.

I would like to thank my Mentor **Mrs. Bhavya K**, Philips Innovation Campus, Bangalore for her valuable guidance and for driving me into ASP team.

It gives me an immense pleasure to thank **Dr. Madhuri Bhavsar**, Hon'ble Head of Information Technology Department, Institute of Technology, Nirma University, Ahmedabad for her kind support and providing basic infrastructure and healthy research environment.

A special thank you is expressed wholeheartedly to **Dr. Alka Mahajan**, Hon'ble Director, Institute of Technology, Nirma University, Ahmedabad for the unmentionable motivation she has extended throughout course of this work.

I would also thank the Institution, all faculty members of Computer Engineering Department, Nirma University, Ahmedabad for their special attention and suggestions towards the project work.

- Patel Namrata J
16MCEN12

Abstract

Analytics and Solution Platform (ASP) is designed to provide the analytical backbone for Imaging Customer Service(ImCS) solutions on the Health Suite Digital Platform(HSDP). The primary data sources include device logs and structured data from service systems.Device logs can be system name,system type,catalog number etc.The device logs are a mix of legacy logs and evolving logs which requires the Business Innovation unit(BIU) to share the domain expertise in designing parsers.Service systems which captures the service related data which also helps to generate detailed insights on the device serviceability. Applications are to be written on top of the Analytics Solutions Platform which utilizes the data for performing Diagnostics, Proactive, Predictive monitoring of the devices and utilization of the devices across the sites.

Abbreviations

ASP	Analytics and Solution Platform
ETL	Extract Transfer and Load
ImCS	Imaging Customer Service
HSDP	Health Suite Digital Platform
BIU	Business Innovation Unit
MR	Magnetic Resonance
CV	Cardiovascular
CT	Computed Tomography
US	Ultra Sound
ICAP	Imaging Clinical Application and platforms
BAM	Business Activity Monitoring
CRUD	Create Read Update Delete
S3	Simple Storage Service
IAM	Identity and Access Management
RADAR	Remote Application for Diagnostics Analysis and Reporting
AWS	Amazon Web Service
CSV	comma separated values
REST	Representational State Transfer
API	Application Programming Interface
URL	Uniform Resource Locator
LLD	Low-level design
HLD	High-level design

–

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 About the Organization

Philips was founded in the 1891 year by Frederik and Gerard Philips.First it was called as Royal Philips Electronics of Netherlands.It is a Dutch based technology company which was located in the city called Amsterdam. Philips was sub divided into following major areas like Philips Consumer Life Style i.e Philips Consumer Electronics and Philips Domestic Appliances and Personal Care,Lighting and Philips Healthcare [1].

Philips Electrical Co. Pvt Ltd situated in 1930 in India,Kolkata as outlet.Philips established in 1938 for manufacturing lamps and lights in Kolkata. Later it was renamed to Philops India Ltd.This is know as Philips Healthcare Innovation Centre in Pune.In bangalore, the philips software Centre established and renamed Philips Innovation Campus in 1996.After some years it was ranked 12th in India as most trusted brands according to the Brand Trust Report[1].

Philips Innovation Campus is healthcare based software company which handle and manage all the logs that are generated daily by medical devices.It ensures that to provide better and secure medical devices which are used to treat patient safely.In addition to that it develops various medical devices like Magnetic Resonance(MR) scan,Computed Tomography(CT) scan,Cardio Vascular(CV),Ultrasound,X-Ray etc.These help to take detailed image of patient and treat them with comfort.

1.2 Project Overview

ASP platform consist of three major frameworks:

- Data Ingestion Framework
- Data Processing Framework
- Data Access Framework

1.2.1 Data Ingestion Framework

Data Ingestion Framework provides the interface to ingest data from different sources. The purpose of the Data Ingestion Framework is to safely land the ingested data into the ASP Landing zone.Data can be device logs and structured data from service system as shown in figure1.1.Data are mix of legacy logs i.e old logs and evolving logs like logs for particular time period based on to requirement.API is used to collect that data and stored in landing zone.Landing zone is one kind of intermediate storage area in cloud foundary where Extract, Transform and Load(ETL) process is done.

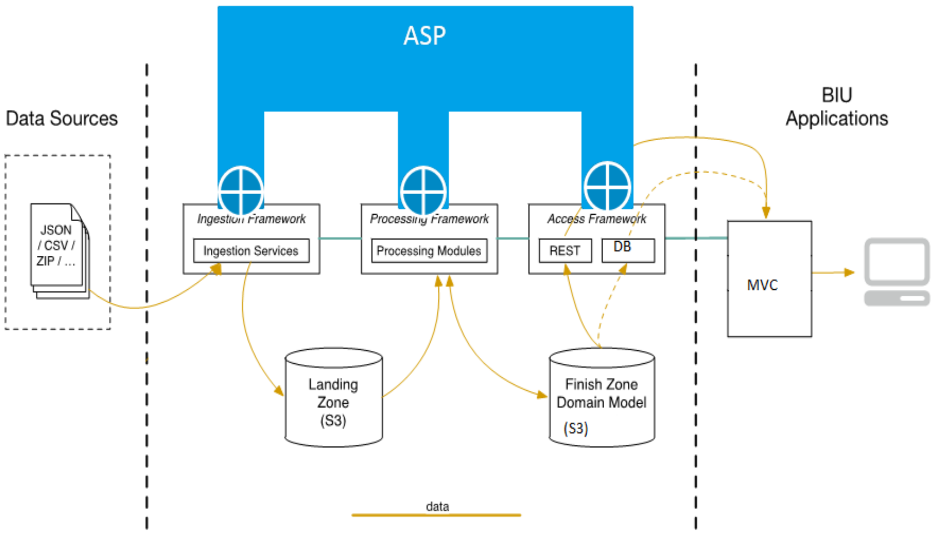


Figure 1.1: ASP

1.2.2 Data Processing Framework

Data Processing Framework provides a mechanism to orchestrate a workflow where the data processing components can be seamlessly plugged in. Data Processing Framework is responsible for initiating the execution of the workflow. Data Processing Framework also offers primitives to plug in Business rules and Analytical models. Components orchestrated in the workflow can access the Data Warehouse for persisting the results of

Data extraction and retrieving data for execution of Business rules and Models.

1.2.3 Data Access Framework

Data Access Framework offers an elegant and secure APIs to access the Domain data. Data Access is controlled at the level of domain models such that access to specific domain data sets are made available for the corresponding application components.Data Access Framework contains three services:

- ASP Domain Access Service :
  Provides the REST way of accessing the data stored in the ASP DWH by the Application layer and external clients following the OData Standard
- ASP BAM Domain Access Service :
  Responsible for performing any CRUD opeartions on the ASP Provenance datastore by the internal ASP services
- ASP File Access Service :
  Provides the REST way of accessing the device log files and the output files produced by the ETLs which are stored in S3 landing zone and the publish zone

1.3 Goals and Objectives

1.3.1 Historical Raw Data Migration

Nowadays big data migration to cloud is become one of the popular topic in any of the IT Organization as in company's data are generated day by day.For example,in health care industry everyday data are generated from medical devices like US in logs.To handle that data is one of the challenging task because data can be in structured ,semi-structured and unstructured format.In addition to that logs are increased exponentially so it can't be store in local system.

There are various tools available to migrate huge amount of data to cloud but they don't have capability to convert that data from unstructured to structured format.So if someone want to process that data for further use from S3 its very difficult to find that as they are not in proper format.

The purpose of Historical Raw Data Migration is to convert data into proper structure format and than migrate to amazon S3.In addition to that to reduce the loss of data and repetitive data it maintains different file states like In-Progress(IP),Complete(C) and Text(txt).

1.3.2 S3 File Access Service

The purpose of S3 File Access Service is to provide interface between Amazon S3 and the application for remote users to access required log files with IAM authentication and authorization.

Any user can access the log files with required credentials like S3 bucket name, shared key and api key. With these privilege users can perform any file operations such as upload, modify and delete. For these middle-ware application is required which controls the access to S3 log files.

Hence S3 File Access Service is REST web service which interact with S3 and perform the following operation:

- Authenticate and authorize the request
- Provide list of Pre-Signed URLs to download requested S3 Objects for read only purpose along with its expiration time, file metadata (file size and file last modified) and system metadata.

1.3.3 Extract Transfer and Load

Extract useful data from CSV files with headers and upload that data into amazon S3,RedShift and Postgres.

- Post ETL :Post ETL fetch PostTestName and Result from Description which start with POST.For this regular expression is used.Same data are uploaded to amazon S3 and if data is not present than it will export into Redshift so it handles deduplication of data.

- System ETL : Fetch Data from IMF notification,write into CSV file and upload that file to amazon S3.Data Mapping is done using batchid from IMF notification.At last check that data is presented in RedShift or not.If data is presented it will not export in RedShift otherwise it will export data in RedShift.So it handles duplication of data.

- System Sync ETL :Fetch batch id and modality from IMF notification based on to that information it will read data from RedShift as per modality(For example if modality is MR data should be fetch from ics_system_mr same for other modality like ics_system_cv,ics_system_ct,ics_system_dc,ics_system_us,ics_system_ icap).That data will write in CSV files with headers and uploaded to amazon S3.After that it checks if data uploaded or not if yes than insert that data into postgres ics_system table and also insert ETL name with insertdatetime in to public.delta_bookmark table.

1.4 Core Features

1.4.1 Historical Raw Data Migration

- Zip File Preparer
- Upload Ultra Sound Modality Zip To S3
- Quarantine Zip File Preparer
- Quarantine Upload US Modality Zip To S3
- House Keeping

1.4.2 S3 File Access Service

S3 File Access Service provides various features as following:

- The user is able to filter the request based on device metadata. Device metadata includes modality, catalognumber, serialnumber, software version, equipment number, system model, system type, file type (raw and published) and file name. Modality and file type are mandatory parameters.

- In order to educate the user regarding systems/modalities supported in ASP, the following 2 apis are exposed as part of the service to fetch system related information.
  - /systemmetadata returns list of catalognumbers with mapping to systemtype, systemmodel and the versions supported in ASP for the request modality.
  - /modality returns list of ASP supported modalities.
- The user is able to optionally filter the request based on workflowId which is the unique id to identify requests between RADAR and ASP.
- The request should be for specific data range, which is as received in ASP. This is a mandatory parameter
- Only reading of log files is allowed. In no scenario file write is allowed.
- URL for requested S3 object will expire within 60 minutes. This time is configurable.
- The Service api is able to provide presigned URLs for each object along with its expiration time, file metadata and system metadata which satisfies the request criteria.
- The service api provides pagination feature to accommodate larger data set.

1.4.3 ETLs

- Collect or pull data from different data sources
- Perform data cleaning and filtering on data
- Load data into warehouse or repository

1.5 Service Input

This section illustrates input parameters with purpose which are required for particular projects.

1. **Historical Raw Data Migration**

- –ZipFilePreparer $\langle$SourceLocation $\rangle\langle$BatchSize $\rangle\langle$StartDate $\rangle\langle$EndDate $\rangle$
  where date format : MM/DD/YYYY
- –UploadUSModalityZipToS3 $\langle$BatchSize $\rangle\langle$StartDate $\rangle\langle$EndDate $\rangle$
- –QuarantineZipFilePreparer $\langle$SourceLocation $\rangle\langle$BatchSize $\rangle\langle$StartDate $\rangle\langle$EndDate $\rangle$
- –QuarantineUploadUSModalityZipToS3 $\langle$BatchSize $\rangle\langle$StartDate $\rangle\langle$EndDate $\rangle$

2. **S3 File Access Service** The given table 1.1 also describes input parameters are mandatory or non-mandatory.The following parameters are not present in database but its used as internal purpose.

   - Date
   - File Type
   - File Name
   - Page Number
   - Page Size

3. **ETLs**

   - Amazon S3 browser location of IMF json file
     For Example : s3://bucket name/Dev/TestFiles/System/CT.json

1.6 Service Output

1. **Historical Raw Data Migration**

   - Zip File Preparer : Zip file of structured data at target location in local system for valid equipment number
   - Upload Ultra Sound Modality Zip To S3 : Uploaded structured data from local system to amazon S3 browser that is prepared byZip File Preparer job
   - Quarantine Zip File Preparer : Zip file of structured data at target location in local system for invalid equipment number

Input	Purpose	Mandatory/Non-mandatory
accessFromDate	From date in the format yyyy-mm-dd	Mandatory
accessToDate	To date in the format yyyy-mm-dd	Mandatory
catalogNumber	Device catalog number	Non-mandatory
equipment Number	Device equipment/sapsite number	Non-mandatory
fileType	FileType can be Raw (Landing zone files) or Published (Publish zone files)	Mandatory
filename	Name of the file required. Anything that ends with the given value will be returned.Filename can be .xml or .zip or .gz	Non-mandatory
modality	Device modality (MR, CV, CT, ICAP, DC, US)	Mandatory
pageNumber	User can request particular pageNumber	Non-mandatory(Default value-1)
pageSize	Records per page	Non-mandatory(Default value-10)
softwareVersion	Device software version for example: 8.1.17.2	Non-mandatory
systemModel	Device software model like FD20	Non-mandatory
systemType	Device/System type like Allura XPer	Non-mandatory
workflowId	Unique ID corresponding to RADAR	Non-mandatory

Table 1.1: Service Input

- Quarantine Upload US Modality Zip To S3 : Uploaded structured data from local system to amazon S3 browser that is prepared by Quarantine Zip File Preparer job
- House Keeping : Delete the stuctured files when it successfully uploaded to S3

2. **S3 File Access Service**

   This section depicts output parameters with purpose and sample as shown in table1.2.Sample output describes detailed information about device metadata,S3 object,presigned URL etc.In records parameter it will give all the following output parameters which describes in table1.2.

Output	**Purpose**	**Sample**
totalRecords	Total number of records matching the user input criteria	"totalRecords": "10"
pageNumber	Page number as requested by the user	"pageNumber": "7"
expectedPageSize	Page size as requested by the user	"expectedPageSize": "5"
records	Json array of S3 objects	records [ ]
s3Object	path of the S3 object	"PublishZone/logviewer/modality=CV /systemId=12345/year=2017/ month=10/date=23/logfilename.gz",
presignedUrl	Provide presignedUrls from where user can download requested logfiles	"https://bucketname.s3-eu-west-1.amazonaws.com/PublishZone/logviewer /modality/systemId/year/month/date /lofilename?AWSAccessKeyId&Expires &Signature"
deviceMetadata	System metadata namely the catalogNumber, modality, serialNumber, systemmodel, systemType, equipmentNumber, softwareVersion	"deviceMetadata": ["catalogNumber": "72345", "modality": "CV", "serialNumber": "109", "systemModel": "F10 u", "systemType": "Allura XPer", "systemuid": "12345", "equipmentNumber": "12345", "softwareVersion": "1.2.9.0"]
s3ObjectMeta data	It gives information about size of object and last modified date with time	"s3ObjectMetadata": ["size": "1400300", "lastModified": "2017-10-27 13:27:24"]
preSignedUrl Expiration	Presigned url expiration timestamp	"presignedUrlExpiration":"2017-10-28 13:45:51"
pageSize	Size of the page (number of records) returned	"pageSize":"2"

Table 1.2: Service Output

3. **ETLs**

   - Based on to batchid from IMF read comma separated values(CSV) files and write transformed CSV file to S3

Chapter 2

Literature Survey

Varieties of work has been done for big data migration to cloud.This section highlights the important points from referred papers.

1. The author describes how big data migration,storage of data and security become very popular topic nowadays.This paper specified the different technologies and way to migrate big data to cloud.

   He described migration can be done from hardware to cloud based infrastructure i.e Enterprise to cloud,mobile to cloud and cloud to cloud.

   By considered different risks like architecture ,network level,application level he did analysis and suggested different strategies[2].

2. The writer mainly focuses on different issues which are still present while migrating big data from worldwide to cloud.He specified that existing de facto approach is also not secure when it comes to privacy of data.

   He used MapReduce framework for migrating dynamically generated daily data from worldwide.He opt for main cost minimzing parameter while uploading data to cloud and proposed Online Lazy Migration(OLM) algorithm and Randomizex Fixed Horizon Control(RFHC) algorithm.These approach can be used while uploading any of data from worldwide at any time to cloud with less cost as compared to existing system.He compared both online algorithm and specified that OLM is able to achieves worst case ratio lower than 2.55 while considering real world setting. [3].

3. The author expressed how use of cloud for big data storage is increased day by day and big data can is categorized by its Velocity,Variety,Veracity and Volume.As data are not always in proper format it can be structured,unstructured and semi structured.Due to this reason its difficult to stored big data in distributed file system architectures.

   He specified that how HDFS and Hadoop by Apache are used for uploading big data to cloud which involves different challenges like scalability,flexibility and fault tolerance.The paper presented different methodologies for analysis and design of data migration to cloud[4].

4. The author mainly strengthens his paper towards security solutions while deploying software systems or any resources to cloud.This paper compared existing reserach done on cloud migration and based onto that framework and provides solution for legacy to cloud migration.Author has not used any of tool to automate data migration tasks and identifies that how architectural adaptiona and selft adaptive cloud system are required to migrate big data[5].

5. The author has considered different challenges and advantages by studying existing scenarios.In addition to that based on related work big data migration divided into three different strategies with different tasks respectively for migrating software from one environment to another i.e from local to cloud. [6].

6. The author told that in IT business how data migration from anywhere is increased day by day.He also specified that different challenges when data migration demand is increased from anywhere through remote office with cost effectively.He considered data transmission cost and other performance issues which are faced during data migration.Based on this analysis he gave some of methods like cloud-to-cloud and cloud-to-service which are effective as they have less number of steps and transmission time while migrating data. [7].

7. The author points out how data migration to cloud is increased day by and day most of organization are taking benefits.Moreover,in many companies data are generated day by day like log files in heath care industry.So mostly organization prefer to use cloud in which user can migrate data from local to cloud and one cloud to

another cloud with minimum time as time complexity is major issue while migrating data to cloud.In this paper author studied different open source cloud platform by considering all the issues related to performance and security he suggested cost effective solution i.e divide and squeeze.

He specified that huge data can be migrate through open source OpenStack with cost effective solution rather than to buy paid cloud provider[8].

8. The author described definition of data migration which is migration data between different storage devices and computers by using some tools or programs or scripts.Once data migration completed that particular organization has to verify that data by using data accuracy parameter.At last data cleaning process come into picture which remove usefully data.This paper presented different data migration strategies i.e Energy Efficient,Load Balancing,Fault Tolerant Migration techniques. [9].

9. The author justified some of security parameters and importance while migrating systems and also specified approach related to this.As demand for cloud is increased day by day as in organization data are produced everyday to handle that data and migrate to cloud it is one of the challenging task.In addition to that security and privacy are the major points when it comes to migrating some particular organization's data. [10].

10. The author mainly focuses on to classify the cloud migration by considering and evaluating existing approaches based on to that how to identify the optimization approach for same.The main optimization criteria are multi objective,tool based,model based and architecture based optimization.By comparing all four optimization solution he specified that multi optimization provides best solution but still it requires detailed level of future research work[11].

Chapter 3

System Requirements

This section provides number technologies and platform that this project will utilize for development

1. Server-side :
    - Java
2. Database:
    - PostgreSQL and Redshift
3. Platform :
    - Java 1.8.0_131
    - Apache Maven 3.5.0
    - Apache Tomcat 9.0
    - Postman
    - Eclipse Oxygen IDE
4. Infrastructure :
    - Amazon S3 Cloud

Chapter 4

Design Overview

The following diagram4.1 illustrates how S3 File Access Service fetch metadata from database based on to the request. After that it connects to S3 browser and request S3 object, S3 will send requested Objects. At last, S3 File Access Service provide list of URLs to client.

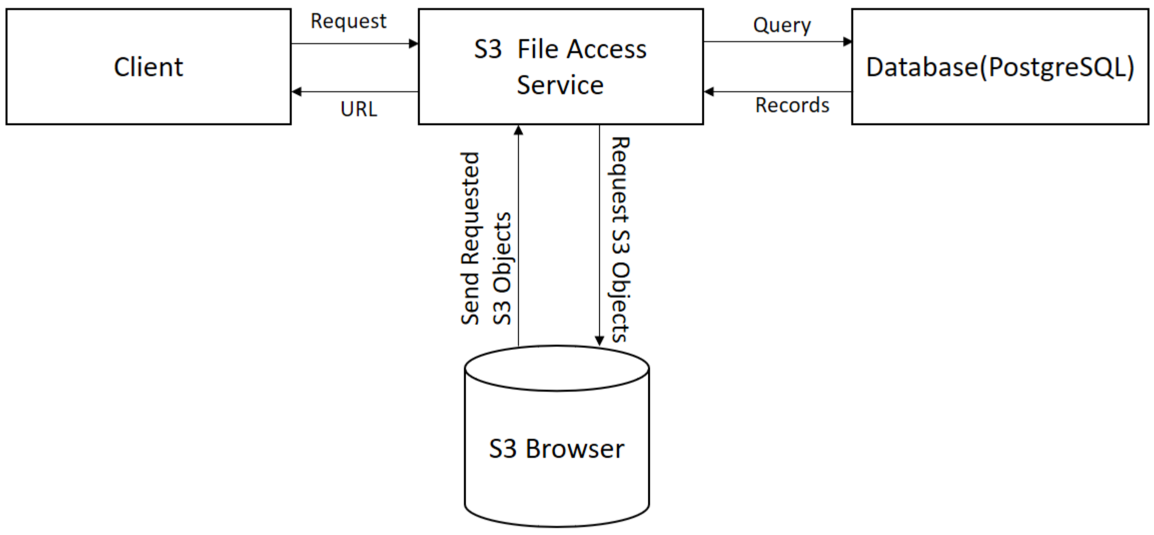


Figure 4.1: S3 file access service flow diagram

4.1 Low Level Design Diagram

Low-level design (LLD) is a component-level design process that follows a step-by-step refinement process. This process can be used for designing data structures, required software architecture, source code.

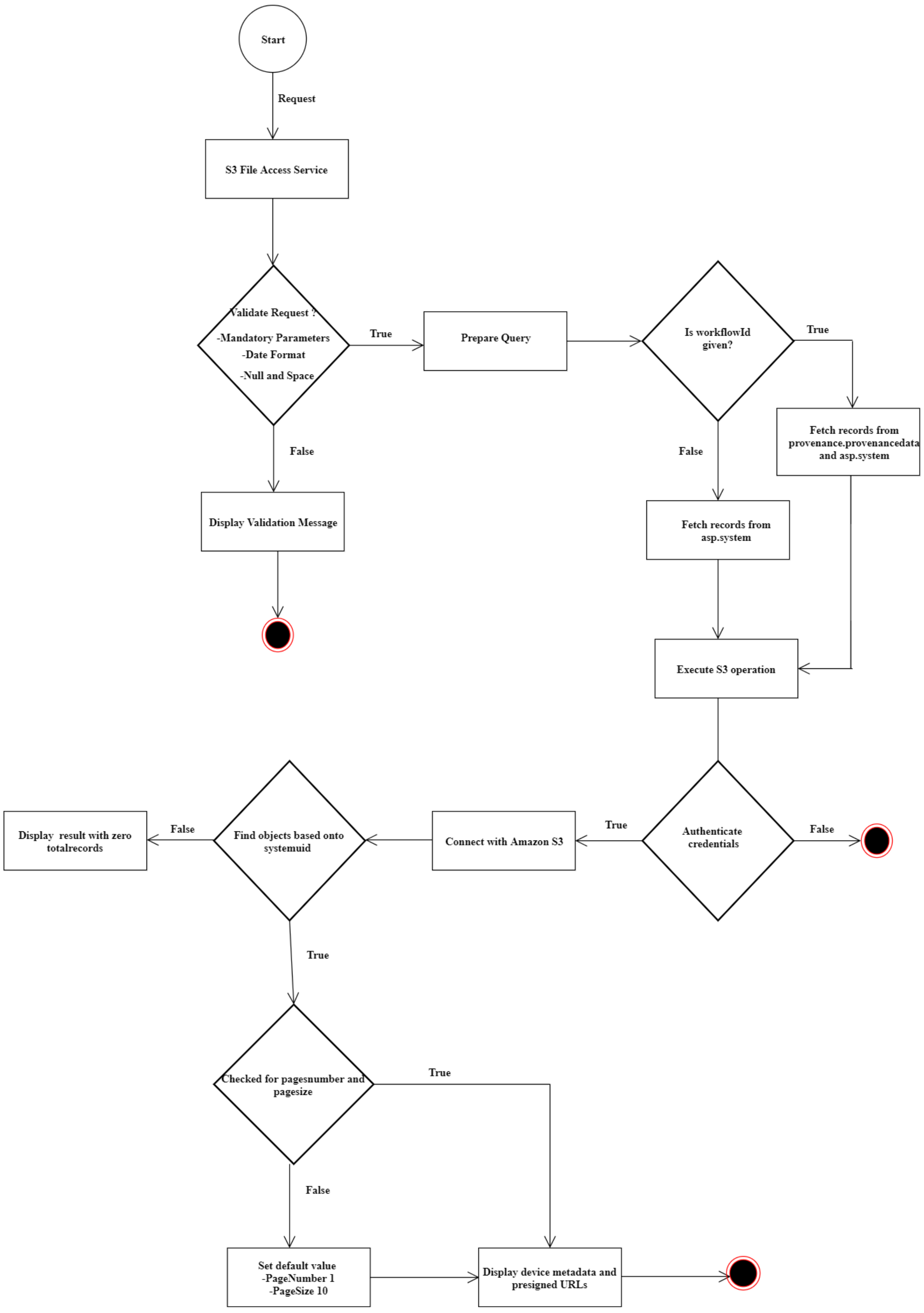


Figure 4.2: Low level design diagram

4.2 High Level Design Diagram

High-level design (HLD) explains the architecture that would be used for developing a software product. The architecture diagram provides an overview of an entire system, identifying the main components that would be developed for the product and their interfaces. The HLD uses possibly nontechnical to mildly technical terms that should be understandable to the administrators of the system. In contrast, low-level design further exposes the logical detailed design of each of these elements for programmers. File Access Service provides the REST way of accessing the device log files and the

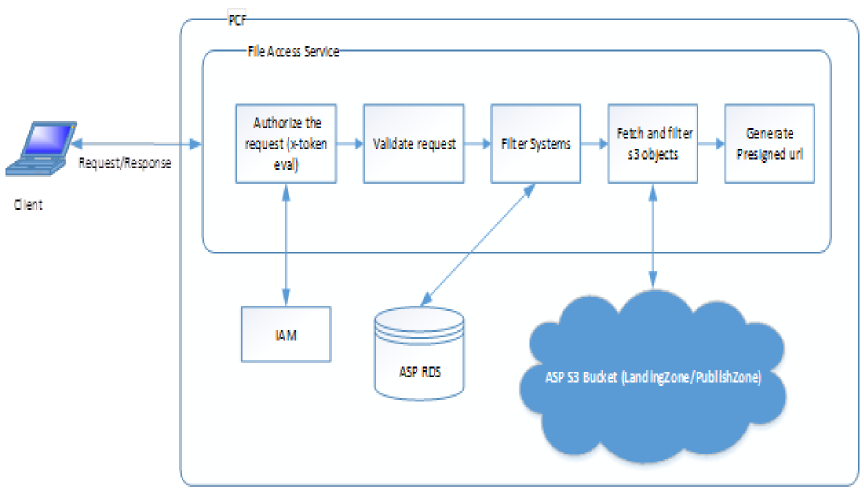


Figure 4.3: High level design diagram

output files produced by the ETLs which are stored in S3 landing zone and the publish zone. The service follows SOA architecture and scalable horizontally to handle the load.

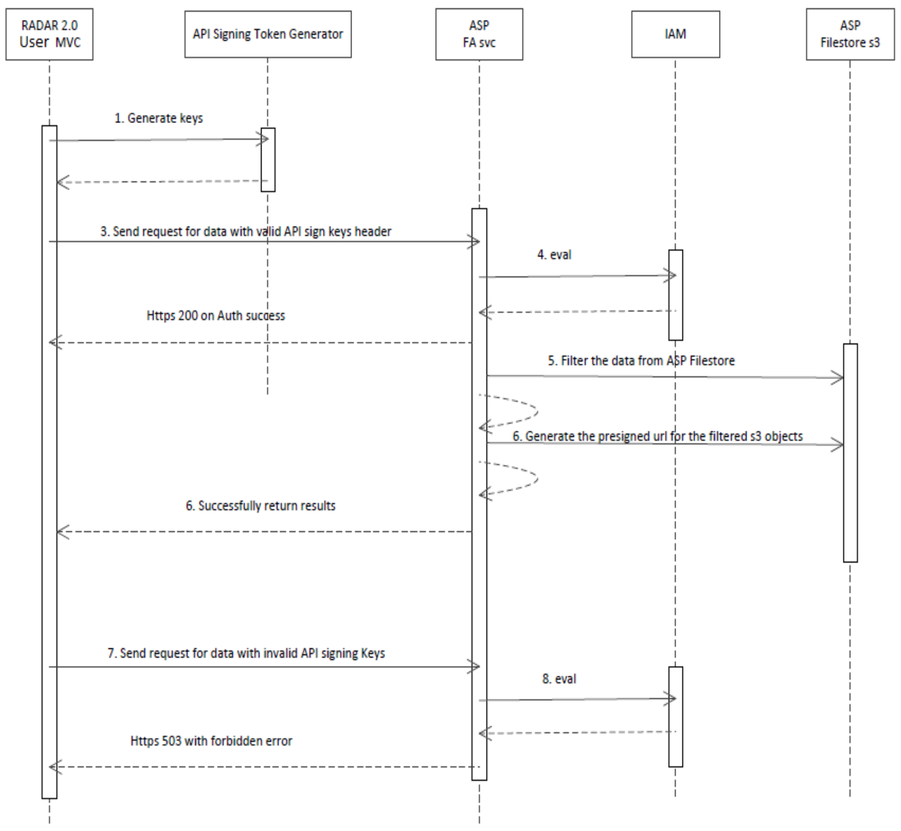
- The service provides data security via integration with IAM with Token based authentication. The x-token embedded in the header of the request is validated for authenticity.

- The successfully authorized request is then validated to satisfy the pre conditions to process the request further. This includes mandatory parameter validation and default value setting of missing mandatory parameters. The details of service input are as mentioned in the table 1.1.

- The request is then processed further to get the system metadata which is maintained in the ASP RDS system table satisfying the input criteria.

- For the systems filtered and for the input date range, file type, the s3 objects are retrieved.
- The retrieved s3 objects are then filtered accordingly for pagination criteria (page number and the page size).
- The result set thus formed is returned to the user in the form of json. The structure of the result is as mentioned in the table 1.2.

4.3 File Access Flow



RADAR 2.0 User MVC
API Signing Token Generator
ASP FA svc
IAM
ASP Filestore s3
1. Generate keys
3. Send request for data with valid API sign keys header
4. eval
Https 200 on Auth success
5. Filter the data from ASP Filestore
6. Generate the presigned url for the filtered s3 objects
6. Successfully return results
7. Send request for data with invalid API signing Keys
8. eval
Https 503 with forbidden error

Figure 4.4: File access flow diagram

4.4 Architecture Diagram

4.4.1 Unstructured Input Files

The following diagram 4.5 describes unstructured and improper format of data.It contains three types of file as follow:

- Type 1 : Files start with Alert
- Type 2 : Files start with Numbers
- Type 3 : Files start with US

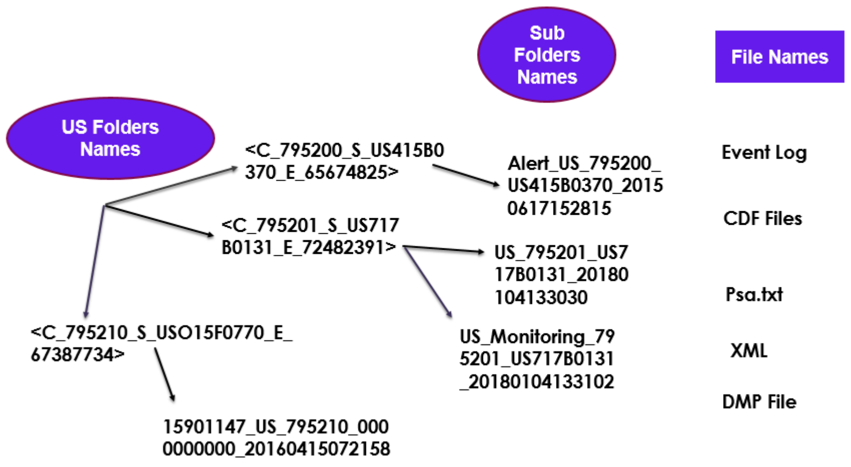

Sub Folders Names
File Names
US Folders Names
<C_795200_S_US415B0370_E_65674825>
Alert_US_795200_US415B0370_20150617152815
Event Log
CDF Files
<C_795201_S_US717B0131_E_72482391>
US_795201_US717B0131_20180104133030
Psa.txt
<C_795210_S_USO15F0770_E_67387734>
US_Monitoring_795201_US717B0131_20180104133102
XML
DMP File
15901147_US_795210_0000000000_20160415072158

Figure 4.5: US Source Folder Format: Unstructured and Improper format

4.4.2 Structured Files with destination folder

The following diagram 4.5 illustrates structured and proper format of data with destination.

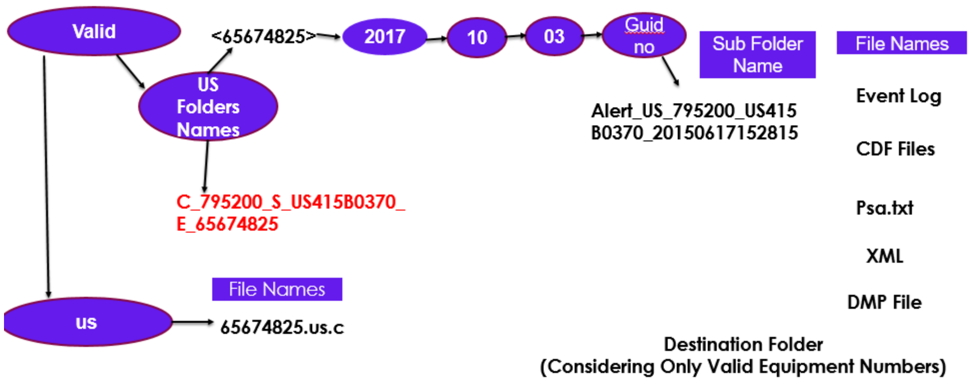


Figure 4.6: US Destination Folder: Structured Data and Proper format

4.4.3 Structured Data and Proper format(PsaLogs)

The following diagram 4.7 illustrates structured and proper format of data with destination.

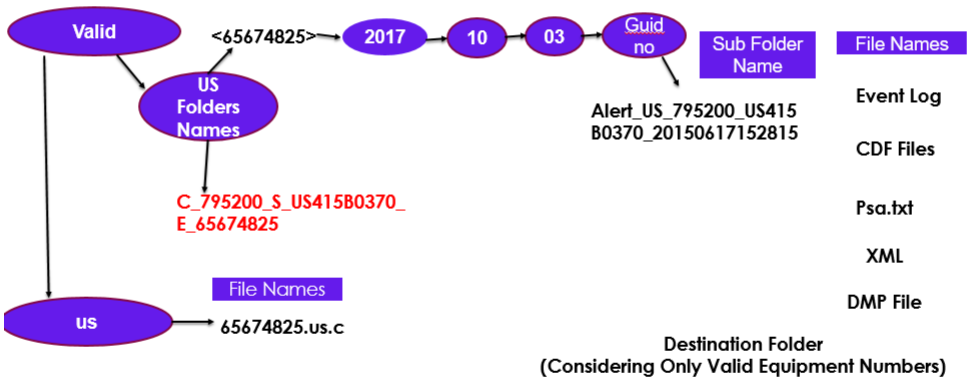


Figure 4.7: Structured and Proper format data(PsaLogs)

Chapter 5

Implementation

5.1 Ramp Up

5.1.1 Task I-Spring MVC and AngularJs

Task I is create form which include both technologies like Java Spring MVC and AngularJS.This web application contain following options:

- SignUP : New users must have to register themselves otherwise they are not able to login.
- Login : Only registered users can login.
- Photo Gallery : It contain three different images.
- Contact : It contains Contact details.

5.1.2 Task II-Session Management

Task II is about Session Management.It contains following three options:

- Login
- Logout
- Profile

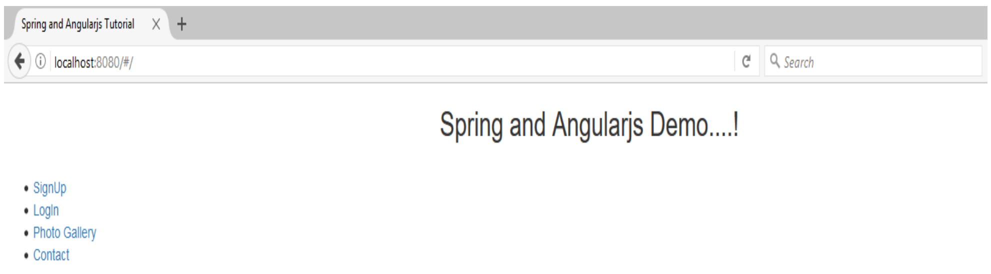


Figure 5.1: Spring and Angularjs Demo

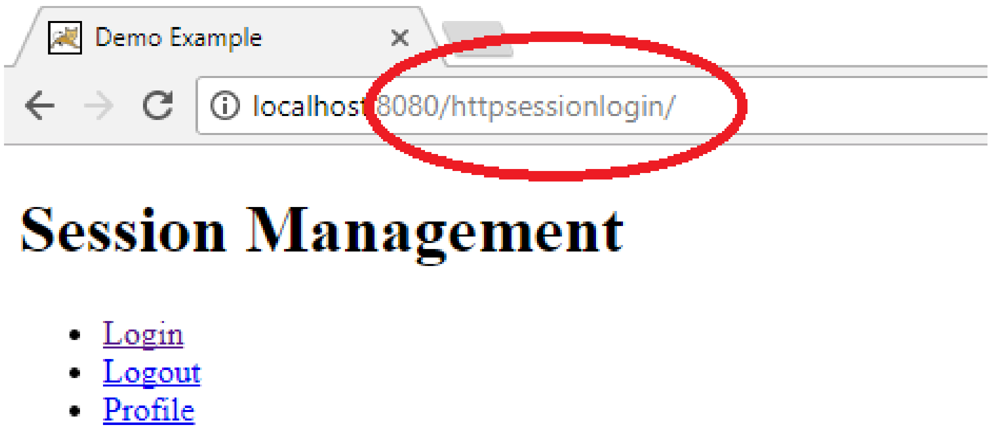


Figure 5.2: Session Management Demo

5.1.3 Task III-Registration Form

Task III is Registration Form which contains following features:

- Registration Form with required validation
- Add new user

- Fetch data from database
- Display data from database
- Insert data in database

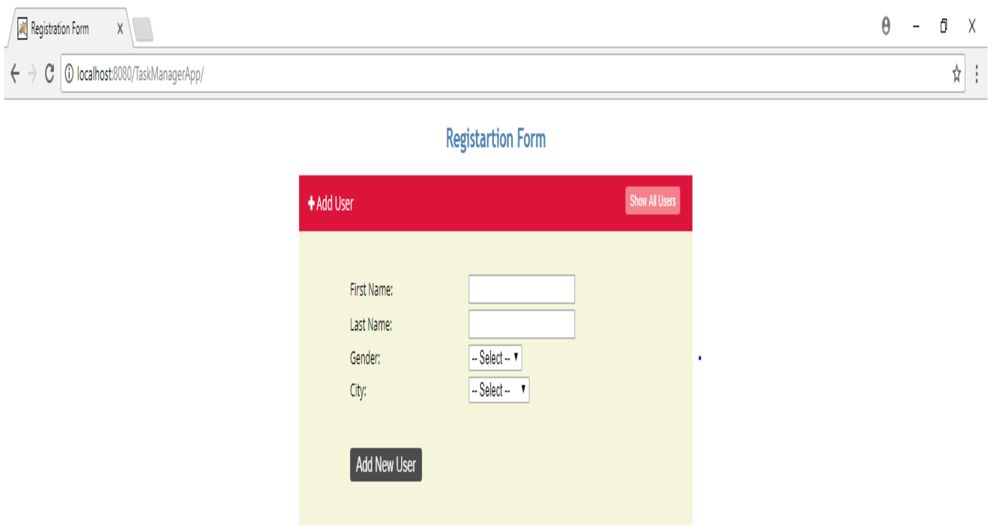


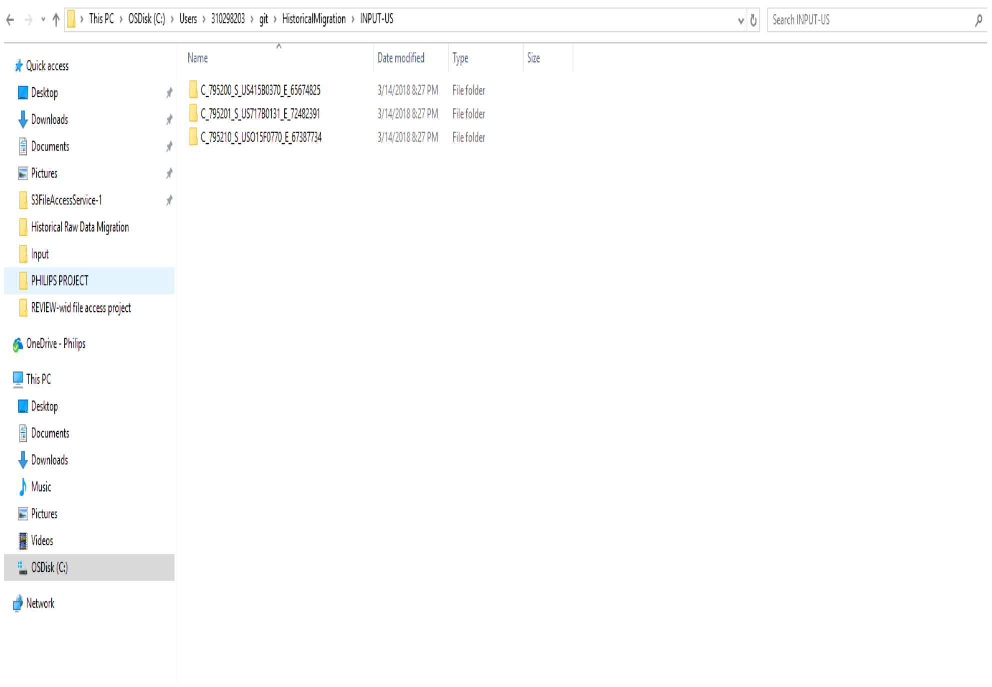
Figure 5.3: Registration Form

5.2 Historical Raw Data Migration

This section provides implementation of Historical Raw Data Migration. In figure 5.4 describes Unstructured input files stored into the local system. The first job ZipFilePrepare can be done using argument with source location,batch size and date range i.e shown in figure 5.5 Figure 5.6 and 5.7 depict that eclipse console output about all logs respective to the classes.It's running based onto the batch wise.

After completing that,figure 5.8 describes the stuctured data stored into local as the S3 uploder job is not executed.The path is userdirectory/date range/ICS/Device Data/US/SystemUid. As there are three types of input files for US medical device figure 5.9 ,5.10 and 5.11 shows output for each of input file.

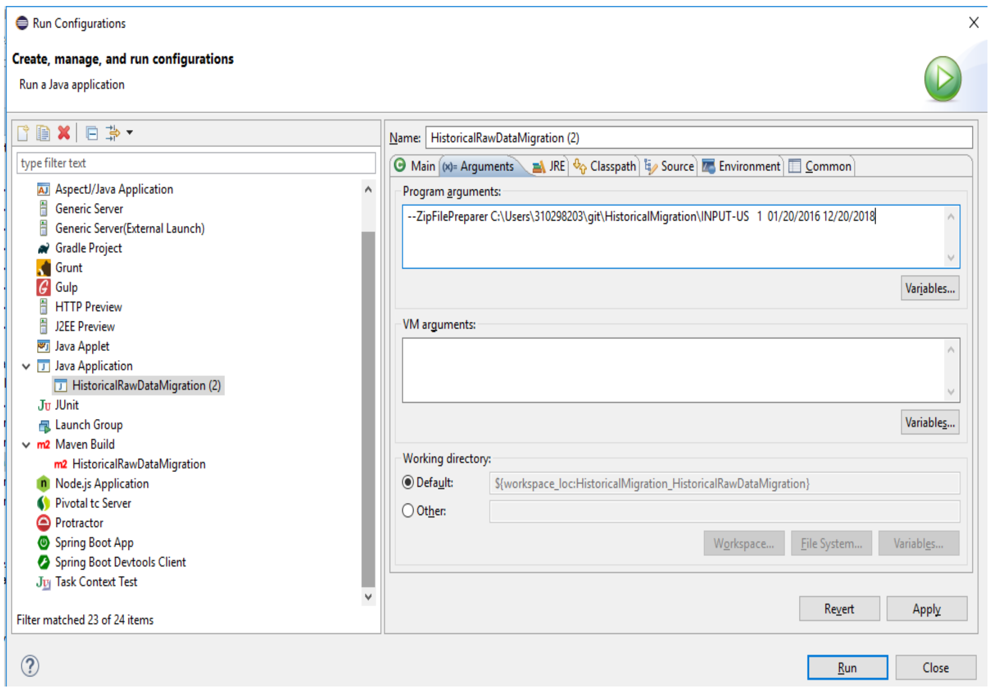
After completing ZipFilePrepare the file state change to Completed state i.e C which can be shown in figure 5.12 In figure 5.13 describes the argument for s3 uploader after Zipfilepreparer the argument consist of batch size and date range.



This PC › OSDisk (C:) › Users › 310298203 › git › HistoricalMigration › INPUT-US

Name	Date modified	Type	Size
C_795200_S_US415B0370_E_65674825	3/14/2018 8:27 PM	File folder	
C_795201_S_US717B0131_E_72482391	3/14/2018 8:27 PM	File folder	
C_795210_S_USO15F0770_E_67387734	3/14/2018 8:27 PM	File folder	

Figure 5.4: Unstructured Input Files



Run Configurations

Create, manage, and run configurations

Run a Java application

Name: HistoricalRawDataMigration (2)

Program arguments:

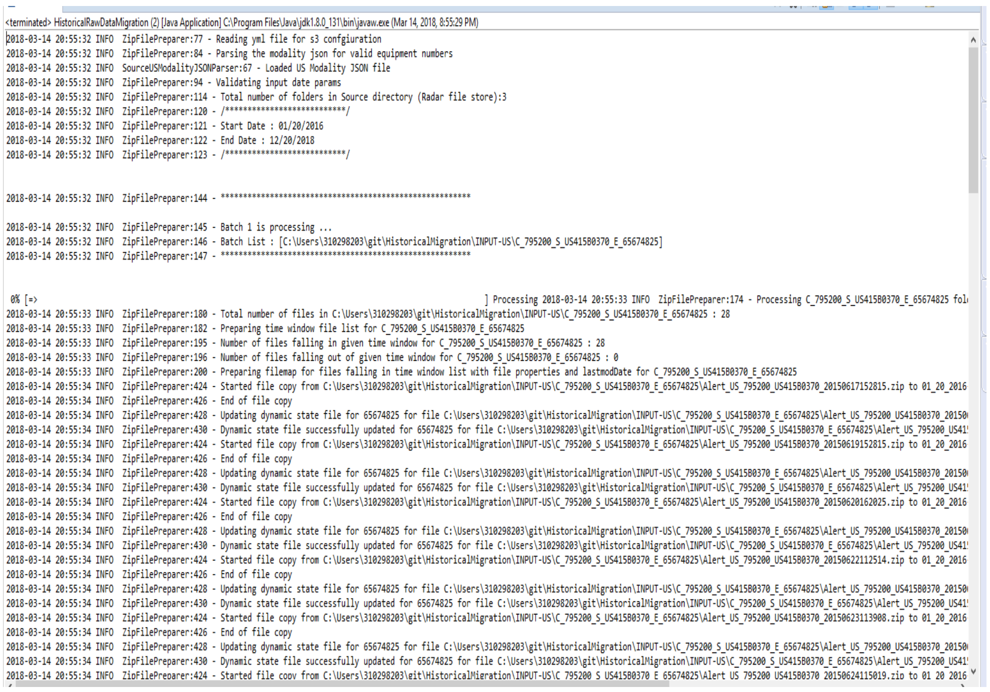
```
--ZipFilePreparer C:\Users\310298203\git\HistoricalMigration\INPUT-US  1  01/20/2016 12/20/2018
```

Working directory: [✓ Default:] ${workspace_loc:HistoricalMigration_HistoricalRawDataMigration}

Figure 5.5: ZipFilePreparer Argument

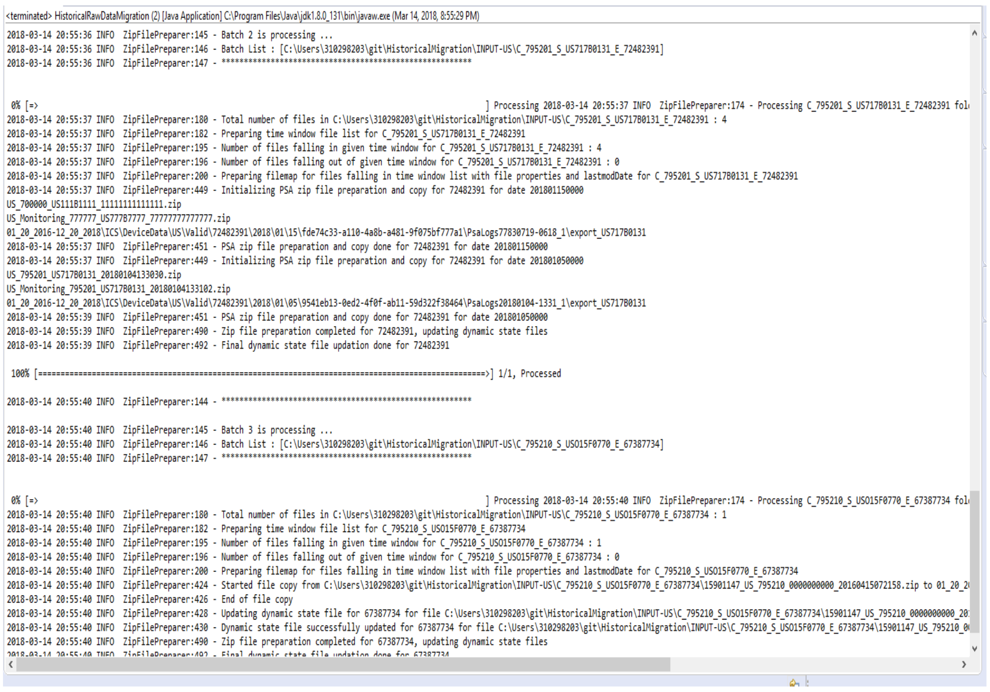
Figure 5.14 describes consloe output for S3 uploader job. After uploading all structured files to amzon S3 we can see that its now available in amazon S3 browser from figure 5.15

At last the file states change from Completed C to Text txt can be seen on figure 5.16 which indicate that files successfully uploaded to amazon S3 so next time if same file will



```
<terminated> HistoricalRawDataMigration (2) [Java Application] C:\Program Files\Java\jdk1.8.0_131\bin\javaw.exe (Mar 14, 2018, 8:55:29 PM)
2018-03-14 20:55:32 INFO  ZipFilePreparer:77 - Reading yml file for s3 confgiuration
2018-03-14 20:55:32 INFO  ZipFilePreparer:84 - Parsing the modality json for valid equipment numbers
2018-03-14 20:55:32 INFO  SourceUSModalityJSONParser:67 - Loaded US Modality JSON file
2018-03-14 20:55:32 INFO  ZipFilePreparer:94 - Validating input date params
2018-03-14 20:55:32 INFO  ZipFilePreparer:114 - Total number of folders in Source directory (Radar file store):3
2018-03-14 20:55:32 INFO  ZipFilePreparer:120 - /**************************/
2018-03-14 20:55:32 INFO  ZipFilePreparer:121 - Start Date : 01/20/2016
2018-03-14 20:55:32 INFO  ZipFilePreparer:122 - End Date : 12/20/2018
2018-03-14 20:55:32 INFO  ZipFilePreparer:123 - /**************************/


2018-03-14 20:55:32 INFO  ZipFilePreparer:144 - *********************************************************

2018-03-14 20:55:32 INFO  ZipFilePreparer:145 - Batch 1 is processing ...
2018-03-14 20:55:32 INFO  ZipFilePreparer:146 - Batch List : [C:\Users\310298203\git\HistoricalMigration\INPUT-US\C_795200_S_US415B0370_E_65674825]
2018-03-14 20:55:32 INFO  ZipFilePreparer:147 - *********************************************************


 0% [=>                                                                                  ] Processing 2018-03-14 20:55:33 INFO  ZipFilePreparer:174 - Processing C_795200_S_US415B0370_E_65674825 fol
2018-03-14 20:55:33 INFO  ZipFilePreparer:180 - Total number of files in C:\Users\310298203\git\HistoricalMigration\INPUT-US\C_795200_S_US415B0370_E_65674825 : 28
2018-03-14 20:55:33 INFO  ZipFilePreparer:182 - Preparing time window file list for C_795200_S_US415B0370_E_65674825
2018-03-14 20:55:33 INFO  ZipFilePreparer:195 - Number of files falling in given time window for C_795200_S_US415B0370_E_65674825 : 28
2018-03-14 20:55:33 INFO  ZipFilePreparer:196 - Number of files falling out of given time window for C_795200_S_US415B0370_E_65674825 : 0
2018-03-14 20:55:33 INFO  ZipFilePreparer:200 - Preparing filemap for files falling in time window list with file properties and lastmodDate for C_795200_S_US415B0370_E_65674825
2018-03-14 20:55:34 INFO  ZipFilePreparer:424 - Started file copy from C:\Users\310298203\git\HistoricalMigration\INPUT-US\C_795200_S_US415B0370_E_65674825\Alert_US_795200_US415B0370_20150617152815.zip to 01_20_2016
2018-03-14 20:55:34 INFO  ZipFilePreparer:426 - End of file copy
2018-03-14 20:55:34 INFO  ZipFilePreparer:428 - Updating dynamic state file for 65674825 for file C:\Users\310298203\git\HistoricalMigration\INPUT-US\C_795200_S_US415B0370_E_65674825\Alert_US_795200_US415B0370_2015
2018-03-14 20:55:34 INFO  ZipFilePreparer:430 - Dynamic state file successfully updated for 65674825 for file C:\Users\310298203\git\HistoricalMigration\INPUT-US\C_795200_S_US415B0370_E_65674825\Alert_US_795200_US41
2018-03-14 20:55:34 INFO  ZipFilePreparer:424 - Started file copy from C:\Users\310298203\git\HistoricalMigration\INPUT-US\C_795200_S_US415B0370_E_65674825\Alert_US_795200_US415B0370_20150619152815.zip to 01_20_2016
2018-03-14 20:55:34 INFO  ZipFilePreparer:426 - End of file copy
2018-03-14 20:55:34 INFO  ZipFilePreparer:428 - Updating dynamic state file for 65674825 for file C:\Users\310298203\git\HistoricalMigration\INPUT-US\C_795200_S_US415B0370_E_65674825\Alert_US_795200_US415B0370_2015
2018-03-14 20:55:34 INFO  ZipFilePreparer:430 - Dynamic state file successfully updated for 65674825 for file C:\Users\310298203\git\HistoricalMigration\INPUT-US\C_795200_S_US415B0370_E_65674825\Alert_US_795200_US41
2018-03-14 20:55:34 INFO  ZipFilePreparer:424 - Started file copy from C:\Users\310298203\git\HistoricalMigration\INPUT-US\C_795200_S_US415B0370_E_65674825\Alert_US_795200_US415B0370_20150620162025.zip to 01_20_2016
2018-03-14 20:55:34 INFO  ZipFilePreparer:426 - End of file copy
2018-03-14 20:55:34 INFO  ZipFilePreparer:428 - Updating dynamic state file for 65674825 for file C:\Users\310298203\git\HistoricalMigration\INPUT-US\C_795200_S_US415B0370_E_65674825\Alert_US_795200_US415B0370_2015
2018-03-14 20:55:34 INFO  ZipFilePreparer:430 - Dynamic state file successfully updated for 65674825 for file C:\Users\310298203\git\HistoricalMigration\INPUT-US\C_795200_S_US415B0370_E_65674825\Alert_US_795200_US41
2018-03-14 20:55:34 INFO  ZipFilePreparer:424 - Started file copy from C:\Users\310298203\git\HistoricalMigration\INPUT-US\C_795200_S_US415B0370_E_65674825\Alert_US_795200_US415B0370_20150622112514.zip to 01_20_2016
2018-03-14 20:55:34 INFO  ZipFilePreparer:426 - End of file copy
2018-03-14 20:55:34 INFO  ZipFilePreparer:428 - Updating dynamic state file for 65674825 for file C:\Users\310298203\git\HistoricalMigration\INPUT-US\C_795200_S_US415B0370_E_65674825\Alert_US_795200_US415B0370_2015
2018-03-14 20:55:34 INFO  ZipFilePreparer:430 - Dynamic state file successfully updated for 65674825 for file C:\Users\310298203\git\HistoricalMigration\INPUT-US\C_795200_S_US415B0370_E_65674825\Alert_US_795200_US41
2018-03-14 20:55:34 INFO  ZipFilePreparer:424 - Started file copy from C:\Users\310298203\git\HistoricalMigration\INPUT-US\C_795200_S_US415B0370_E_65674825\Alert_US_795200_US415B0370_20150623113908.zip to 01_20_2016
2018-03-14 20:55:34 INFO  ZipFilePreparer:426 - End of file copy
2018-03-14 20:55:34 INFO  ZipFilePreparer:428 - Updating dynamic state file for 65674825 for file C:\Users\310298203\git\HistoricalMigration\INPUT-US\C_795200_S_US415B0370_E_65674825\Alert_US_795200_US415B0370_2015
2018-03-14 20:55:34 INFO  ZipFilePreparer:430 - Dynamic state file successfully updated for 65674825 for file C:\Users\310298203\git\HistoricalMigration\INPUT-US\C_795200_S_US415B0370_E_65674825\Alert_US_795200_US41
2018-03-14 20:55:34 INFO  ZipFilePreparer:424 - Started file copy from C:\Users\310298203\git\HistoricalMigration\INPUT-US\C_795200_S_US415B0370_E_65674825\Alert_US_795200_US415B0370_20150624115019.zip to 01_20_2016
```

Figure 5.6: ZipFilePreparer Console Output for batch one

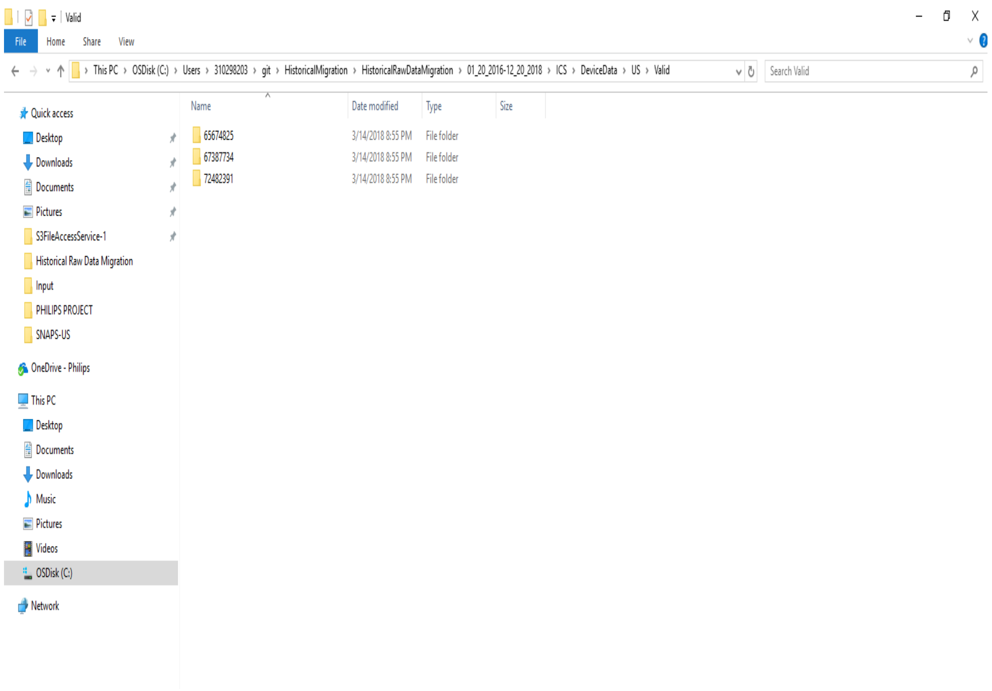


```
<terminated> HistoricalRawDataMigration (2) [Java Application] C:\Program Files\Java\jdk1.8.0_131\bin\javaw.exe (Mar 14, 2018, 8:55:29 PM)
2018-03-14 20:55:36 INFO  ZipFilePreparer:145 - Batch 2 is processing ...
2018-03-14 20:55:36 INFO  ZipFilePreparer:146 - Batch List : [C:\Users\310298203\git\HistoricalMigration\INPUT-US\C_795201_S_US717B0131_E_72482391]
2018-03-14 20:55:36 INFO  ZipFilePreparer:147 - *********************************************************


 0% [=>                                                                                  ] Processing 2018-03-14 20:55:37 INFO  ZipFilePreparer:174 - Processing C_795201_S_US717B0131_E_72482391 fol
2018-03-14 20:55:37 INFO  ZipFilePreparer:180 - Total number of files in C:\Users\310298203\git\HistoricalMigration\INPUT-US\C_795201_S_US717B0131_E_72482391 : 4
2018-03-14 20:55:37 INFO  ZipFilePreparer:182 - Preparing time window file list for C_795201_S_US717B0131_E_72482391
2018-03-14 20:55:37 INFO  ZipFilePreparer:195 - Number of files falling in given time window for C_795201_S_US717B0131_E_72482391 : 4
2018-03-14 20:55:37 INFO  ZipFilePreparer:196 - Number of files falling out of given time window for C_795201_S_US717B0131_E_72482391 : 0
2018-03-14 20:55:37 INFO  ZipFilePreparer:200 - Preparing filemap for files falling in time window list with file properties and lastmodDate for C_795201_S_US717B0131_E_72482391
2018-03-14 20:55:37 INFO  ZipFilePreparer:449 - Initializing PSA zip file preparation and copy for 72482391 for date 201801150000
US_700000_US111B1111_11111111111111.zip
US_Monitoring_777777_US777B7777_77777777777777.zip
01_20_2016-12_20_2018\ICS\DeviceData\US\Valid\72482391\2018\01\15\fde74c33-a110-4a8b-a481-9f075bf777a1\PsaLogs77830719-0618_1\export_US717B0131
2018-03-14 20:55:37 INFO  ZipFilePreparer:451 - PSA zip file preparation and copy done for 72482391 for date 201801150000
2018-03-14 20:55:37 INFO  ZipFilePreparer:449 - Initializing PSA zip file preparation and copy for 72482391 for date 201801050000
US_795201_US717B0131_20180104133030.zip
US_Monitoring_795201_US717B0131_20180104133102.zip
01_20_2016-12_20_2018\ICS\DeviceData\US\Valid\72482391\2018\01\05\9541eb13-0ed2-4f0f-ab11-59d322f38464\PsaLogs20180104-1331_1\export_US717B0131
2018-03-14 20:55:39 INFO  ZipFilePreparer:451 - PSA zip file preparation and copy done for 72482391 for date 201801050000
2018-03-14 20:55:39 INFO  ZipFilePreparer:490 - Zip file preparation completed for 72482391, updating dynamic state files
2018-03-14 20:55:39 INFO  ZipFilePreparer:492 - Final dynamic state file updation done for 72482391

 100% [=================================================================================================>] 1/1, Processed

2018-03-14 20:55:40 INFO  ZipFilePreparer:144 - *********************************************************

2018-03-14 20:55:40 INFO  ZipFilePreparer:145 - Batch 3 is processing ...
2018-03-14 20:55:40 INFO  ZipFilePreparer:146 - Batch List : [C:\Users\310298203\git\HistoricalMigration\INPUT-US\C_795210_S_US015F0770_E_67387734]
2018-03-14 20:55:40 INFO  ZipFilePreparer:147 - *********************************************************


 0% [=>                                                                                  ] Processing 2018-03-14 20:55:40 INFO  ZipFilePreparer:174 - Processing C_795210_S_US015F0770_E_67387734 fol
2018-03-14 20:55:40 INFO  ZipFilePreparer:180 - Total number of files in C:\Users\310298203\git\HistoricalMigration\INPUT-US\C_795210_S_US015F0770_E_67387734 : 1
2018-03-14 20:55:40 INFO  ZipFilePreparer:182 - Preparing time window file list for C_795210_S_US015F0770_E_67387734
2018-03-14 20:55:40 INFO  ZipFilePreparer:195 - Number of files falling in given time window for C_795210_S_US015F0770_E_67387734 : 1
2018-03-14 20:55:40 INFO  ZipFilePreparer:196 - Number of files falling out of given time window for C_795210_S_US015F0770_E_67387734 : 0
2018-03-14 20:55:40 INFO  ZipFilePreparer:200 - Preparing filemap for files falling in time window list with file properties and lastmodDate for C_795210_S_US015F0770_E_67387734
2018-03-14 20:55:40 INFO  ZipFilePreparer:424 - Started file copy from C:\Users\310298203\git\HistoricalMigration\INPUT-US\C_795210_S_US015F0770_E_67387734\15901147_US_795210_0000000000_20160415072158.zip to 01_20_2
2018-03-14 20:55:40 INFO  ZipFilePreparer:426 - End of file copy
2018-03-14 20:55:40 INFO  ZipFilePreparer:428 - Updating dynamic state file for 67387734 for file C:\Users\310298203\git\HistoricalMigration\INPUT-US\C_795210_S_US015F0770_E_67387734\15901147_US_795210_0000000000_20
2018-03-14 20:55:40 INFO  ZipFilePreparer:430 - Dynamic state file successfully updated for 67387734 for file C:\Users\310298203\git\HistoricalMigration\INPUT-US\C_795210_S_US015F0770_E_67387734\15901147_US_795210_0
2018-03-14 20:55:40 INFO  ZipFilePreparer:490 - Zip file preparation completed for 67387734, updating dynamic state files
2018-03-14 20:55:40 INFO  ZipFilePreparer:492 - Final dynamic state file updation done for 67387734
```

Figure 5.7: ZipFilePreparer Console Output for remaining batch

process than it will not uploaded to amazon S3 which reduce to upload repetative data in amazon S3.

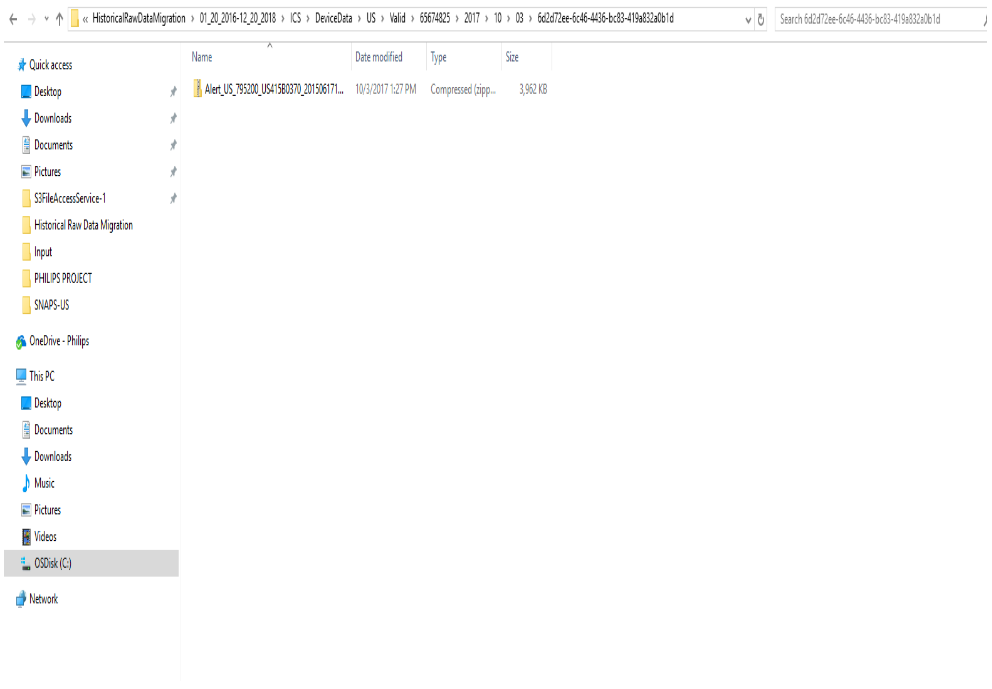
ZipFilePreparer process only valid data by mapping systemuid from json file .The valid systemuid data are fetched from data so for invalid systemuid QuarantineZip-FilePreparer job used.It works same as ZipFilePreparer exceppt that path is different in



This PC › OSDisk (C:) › Users › 310298203 › git › HistoricalMigration › HistoricalRawDataMigration › 01_20_2016-12_20_2018 › ICS › DeviceData › US › Valid

Name	Date modified	Type	Size
65674825	3/14/2018 8:55 PM	File folder	
67387734	3/14/2018 8:55 PM	File folder	
72482391	3/14/2018 8:55 PM	File folder	

Figure 5.8: Structure data in local directory



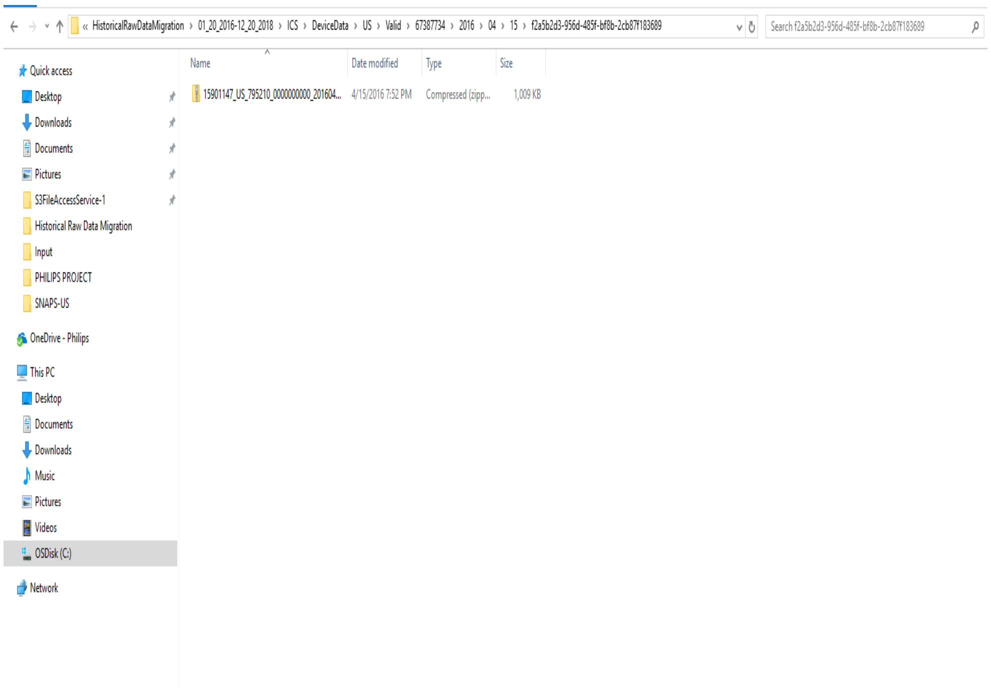
« HistoricalRawDataMigration › 01_20_2016-12_20_2018 › ICS › DeviceData › US › Valid › 65674825 › 2017 › 10 › 03 › 6d2d72ee-6c46-4436-bc83-419a832a0b1d

Name	Date modified	Type	Size
Alert_US_795200_US415B0370_201506171...	10/3/2017 1:27 PM	Compressed (zipp...	3,962 KB

Figure 5.9: Type 1 : Output for Alert File

amzon S3 to keep track of whether the data is valid or quarantine.The pathis userdirectory/date/ICS/Device Data/Quarantine/SystemUid.After completing this job file states are changed and execute quarantine S3 uploader so data can be seen in amazon S3 which describe by figure 5.20.

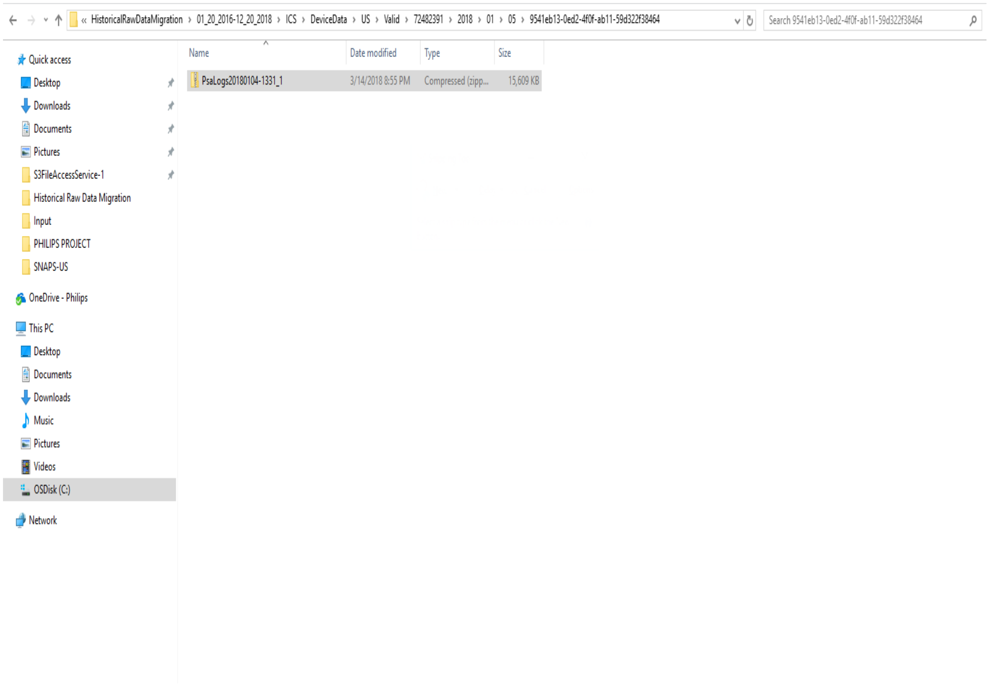
The last job is House Keeping,after data successfully uploaded to amazon S3 it will



« HistoricalRawDataMigration › 01_20_2016-12_20_2018 › ICS › DeviceData › US › Valid › 67387734 › 2016 › 04 › 15 › f2a5b2d3-956d-485f-bf8b-2cb87f183689

Name	Date modified	Type	Size
15901147_US_795210_0000000000_201604...	4/15/2016 7:52 PM	Compressed (zipp...	1,009 KB

Figure 5.10: Type 2 : Output for Number File

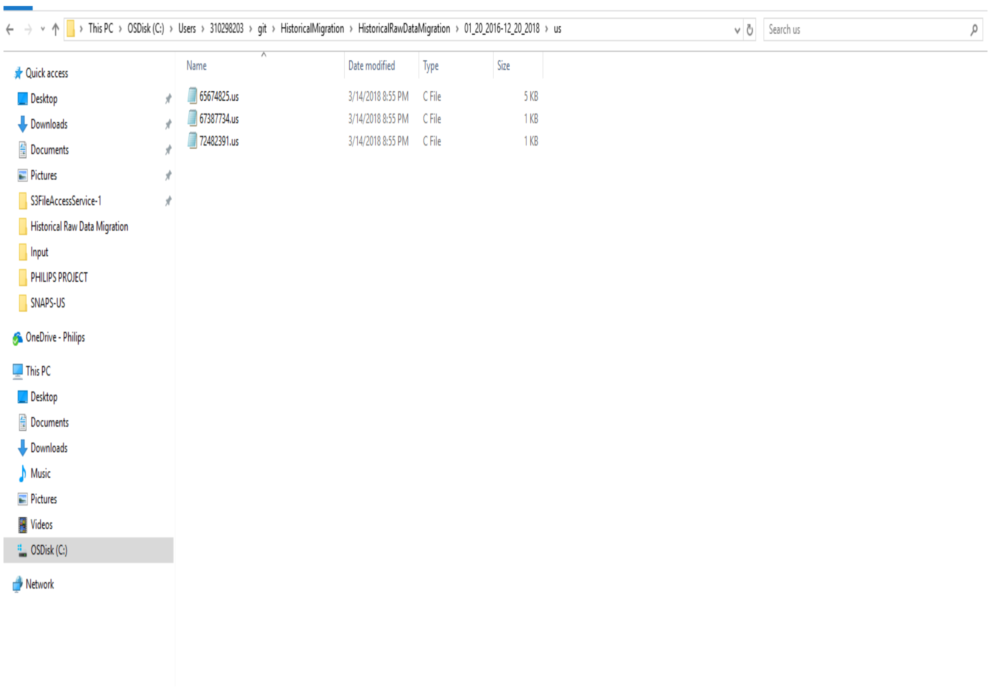


« HistoricalRawDataMigration › 01_20_2016-12_20_2018 › ICS › DeviceData › US › Valid › 72482391 › 2018 › 01 › 05 › 9541eb13-0ed2-4f0f-ab11-59d322f38464

Name	Date modified	Type	Size
PsaLogs20180104-1331_1	3/14/2018 8:55 PM	Compressed (zipp...	15,609 KB

Figure 5.11: Type 3 : Output for PsaLogs

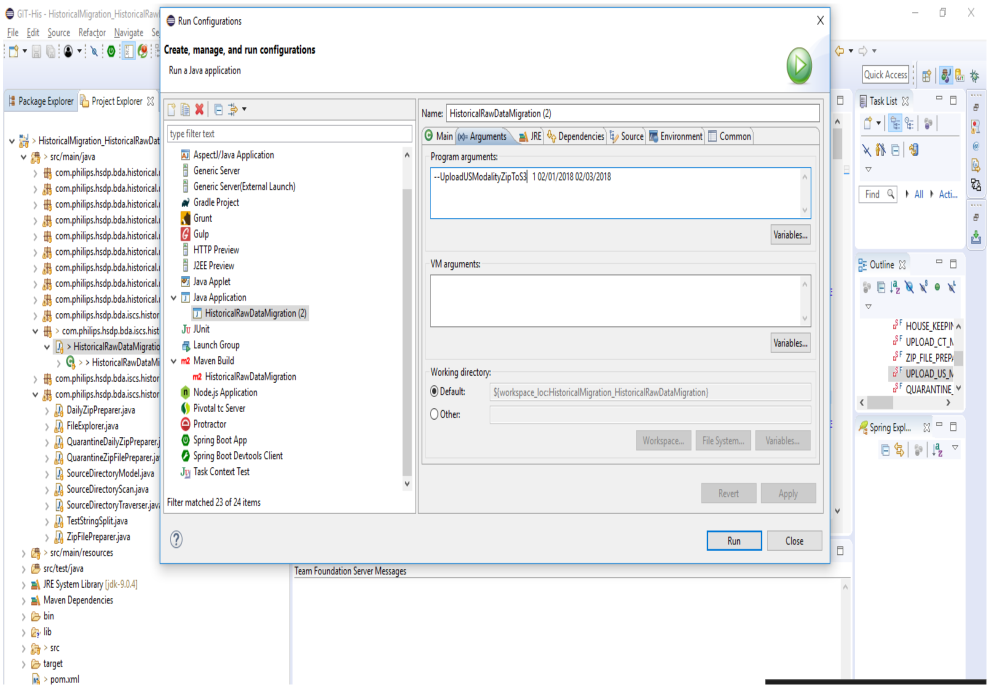
remove all the files from local system i.e data cleaning done by this job as shown in figure 5.21



This PC › OSDisk (C:) › Users › 310298203 › git › HistoricalMigration › HistoricalRawDataMigration › 01_20_2016-12_20_2018 › us

Name	Date modified	Type	Size
65674825.us	3/14/2018 8:55 PM	C File	5 KB
67387734.us	3/14/2018 8:55 PM	C File	1 KB
72482391.us	3/14/2018 8:55 PM	C File	1 KB

Figure 5.12: File State Changes after ZipFilePreparer



Run Configurations

Create, manage, and run configurations

Run a Java application

Name: HistoricalRawDataMigration (2)

Program arguments:

```
--UploadUSModalityZipToS3  1 02/01/2018 02/03/2018
```

Working directory:

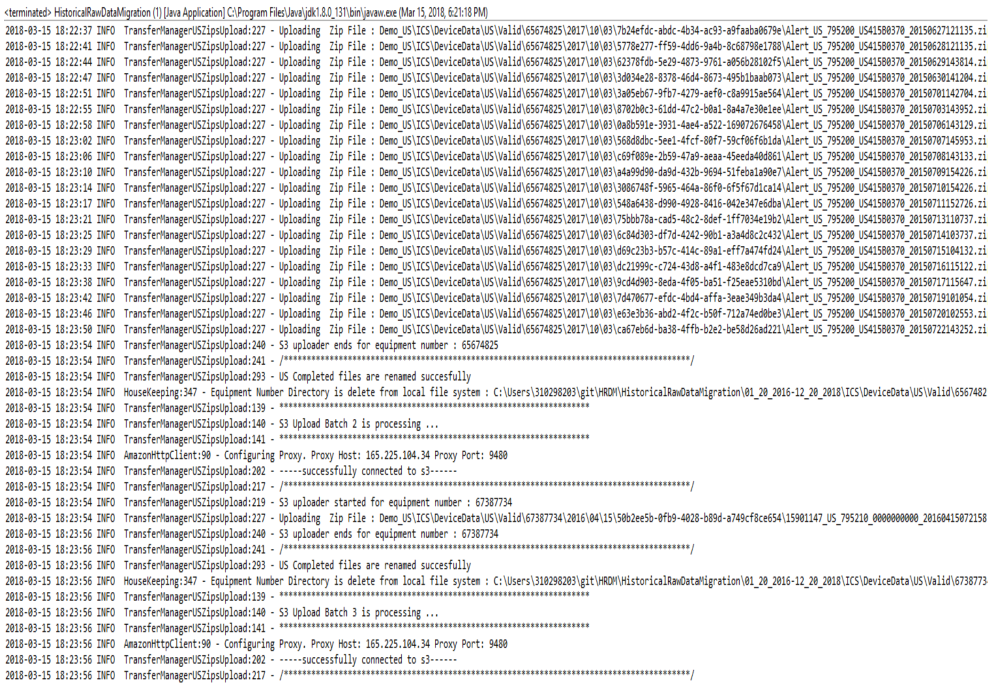
Default: ${workspace_loc:HistoricalMigration_HistoricalRawDataMigration}

Filter matched 23 of 24 items

Figure 5.13: UploadUSModalityZipToS3

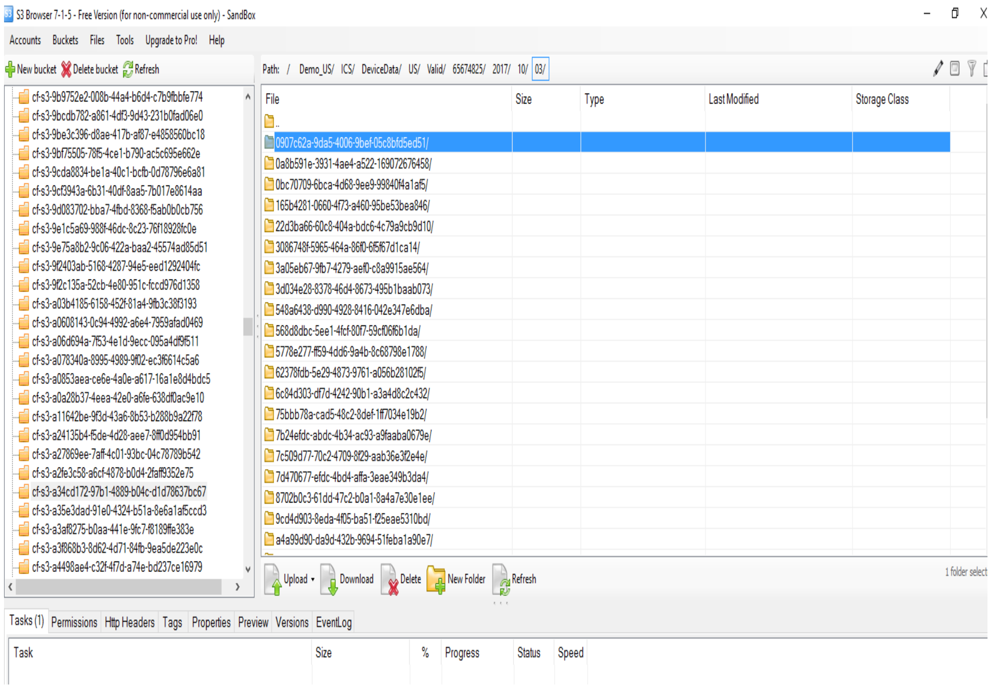
5.3 ETLs

This section provides how ETL extract transform and load that data in amazon S3.The following example shows that first it will find weather the description is start with Post or not if yes than it will take PostTestName and Result from Descriotion and and related



```
<terminated> HistoricalRawDataMigration (1) [Java Application] C:\Program Files\Java\jdk1.8.0_131\bin\javaw.exe (Mar 15, 2018, 6:21:18 PM)
2018-03-15 18:22:37 INFO  TransferManagerUSZipsUpload:227 - Uploading  Zip File : Demo_US\ICS\DeviceData\US\Valid\65674825\2017\10\03\7b24efdc-abdc-4b34-ac93-a9faaba0679e\Alert_US_795200_US415B0370_20150627121135.zi
2018-03-15 18:22:41 INFO  TransferManagerUSZipsUpload:227 - Uploading  Zip File : Demo_US\ICS\DeviceData\US\Valid\65674825\2017\10\03\5778e277-ff59-4dd6-9a4b-8c68798e1788\Alert_US_795200_US415B0370_20150628121135.zi
2018-03-15 18:22:44 INFO  TransferManagerUSZipsUpload:227 - Uploading  Zip File : Demo_US\ICS\DeviceData\US\Valid\65674825\2017\10\03\62378fdb-5e29-4873-9761-a056b28102f5\Alert_US_795200_US415B0370_20150629143814.zi
2018-03-15 18:22:47 INFO  TransferManagerUSZipsUpload:227 - Uploading  Zip File : Demo_US\ICS\DeviceData\US\Valid\65674825\2017\10\03\3d034e28-8378-46d4-8673-495b1baab073\Alert_US_795200_US415B0370_20150630141204.zi
2018-03-15 18:22:51 INFO  TransferManagerUSZipsUpload:227 - Uploading  Zip File : Demo_US\ICS\DeviceData\US\Valid\65674825\2017\10\03\3a05eb67-9fb7-4279-aef0-c8a9915ae564\Alert_US_795200_US415B0370_20150701142704.zi
2018-03-15 18:22:55 INFO  TransferManagerUSZipsUpload:227 - Uploading  Zip File : Demo_US\ICS\DeviceData\US\Valid\65674825\2017\10\03\8702b0c3-61dd-47c2-b0a1-8a4a7e30e1ee\Alert_US_795200_US415B0370_20150703143952.zi
2018-03-15 18:22:58 INFO  TransferManagerUSZipsUpload:227 - Uploading  Zip File : Demo_US\ICS\DeviceData\US\Valid\65674825\2017\10\03\0a8b591e-3931-4ae4-a522-169072676458\Alert_US_795200_US415B0370_20150706143129.zi
2018-03-15 18:23:02 INFO  TransferManagerUSZipsUpload:227 - Uploading  Zip File : Demo_US\ICS\DeviceData\US\Valid\65674825\2017\10\03\568d8dbc-5ee1-4fcf-80f7-59cf06f6b1da\Alert_US_795200_US415B0370_20150707145953.zi
2018-03-15 18:23:06 INFO  TransferManagerUSZipsUpload:227 - Uploading  Zip File : Demo_US\ICS\DeviceData\US\Valid\65674825\2017\10\03\c69f089e-2b59-47a9-aeaa-45eeda40d861\Alert_US_795200_US415B0370_20150708143133.zi
2018-03-15 18:23:10 INFO  TransferManagerUSZipsUpload:227 - Uploading  Zip File : Demo_US\ICS\DeviceData\US\Valid\65674825\2017\10\03\a4a99d90-da9d-432b-9694-51feba1a90e7\Alert_US_795200_US415B0370_20150709154226.zi
2018-03-15 18:23:14 INFO  TransferManagerUSZipsUpload:227 - Uploading  Zip File : Demo_US\ICS\DeviceData\US\Valid\65674825\2017\10\03\3086748f-5965-464a-86f0-6f5f67d1ca14\Alert_US_795200_US415B0370_20150710154226.zi
2018-03-15 18:23:17 INFO  TransferManagerUSZipsUpload:227 - Uploading  Zip File : Demo_US\ICS\DeviceData\US\Valid\65674825\2017\10\03\548a6438-d990-4928-8416-042e347e6dba\Alert_US_795200_US415B0370_20150711152726.zi
2018-03-15 18:23:21 INFO  TransferManagerUSZipsUpload:227 - Uploading  Zip File : Demo_US\ICS\DeviceData\US\Valid\65674825\2017\10\03\75bbb78a-cad5-48c2-8def-1ff7034e19b2\Alert_US_795200_US415B0370_20150713110737.zi
2018-03-15 18:23:25 INFO  TransferManagerUSZipsUpload:227 - Uploading  Zip File : Demo_US\ICS\DeviceData\US\Valid\65674825\2017\10\03\6c84d303-df7d-4242-90b1-a3a4d8c2c432\Alert_US_795200_US415B0370_20150714103737.zi
2018-03-15 18:23:29 INFO  TransferManagerUSZipsUpload:227 - Uploading  Zip File : Demo_US\ICS\DeviceData\US\Valid\65674825\2017\10\03\d69c23b3-b57c-414c-89a1-eff7a474fd24\Alert_US_795200_US415B0370_20150715104132.zi
2018-03-15 18:23:33 INFO  TransferManagerUSZipsUpload:227 - Uploading  Zip File : Demo_US\ICS\DeviceData\US\Valid\65674825\2017\10\03\dc21999c-c724-43d8-a4f1-483e8dcd7ca9\Alert_US_795200_US415B0370_20150716115122.zi
2018-03-15 18:23:38 INFO  TransferManagerUSZipsUpload:227 - Uploading  Zip File : Demo_US\ICS\DeviceData\US\Valid\65674825\2017\10\03\9cd4d903-8eda-4f05-ba51-f25eae5310bd\Alert_US_795200_US415B0370_20150717115647.zi
2018-03-15 18:23:42 INFO  TransferManagerUSZipsUpload:227 - Uploading  Zip File : Demo_US\ICS\DeviceData\US\Valid\65674825\2017\10\03\7d470677-efdc-4bd4-affa-3eae349b3da4\Alert_US_795200_US415B0370_20150719101054.zi
2018-03-15 18:23:46 INFO  TransferManagerUSZipsUpload:227 - Uploading  Zip File : Demo_US\ICS\DeviceData\US\Valid\65674825\2017\10\03\e63e3b36-abd2-4f2c-b50f-712a74ed0be3\Alert_US_795200_US415B0370_20150720102553.zi
2018-03-15 18:23:50 INFO  TransferManagerUSZipsUpload:227 - Uploading  Zip File : Demo_US\ICS\DeviceData\US\Valid\65674825\2017\10\03\ca67eb6d-ba38-4ffb-b2e2-be58d26ad221\Alert_US_795200_US415B0370_20150722143252.zi
2018-03-15 18:23:54 INFO  TransferManagerUSZipsUpload:240 - S3 uploader ends for equipment number : 65674825
2018-03-15 18:23:54 INFO  TransferManagerUSZipsUpload:241 - /****************************************************************************************/
2018-03-15 18:23:54 INFO  TransferManagerUSZipsUpload:293 - US Completed files are renamed succesfully
2018-03-15 18:23:54 INFO  HouseKeeping:347 - Equipment Number Directory is delete from local file system : C:\Users\310298203\git\HRDM\HistoricalRawDataMigration\01_20_2016-12_20_2018\ICS\DeviceData\US\Valid\6567482
2018-03-15 18:23:54 INFO  TransferManagerUSZipsUpload:139 - *******************************************************************
2018-03-15 18:23:54 INFO  TransferManagerUSZipsUpload:140 - S3 Upload Batch 2 is processing ...
2018-03-15 18:23:54 INFO  TransferManagerUSZipsUpload:141 - *******************************************************************
2018-03-15 18:23:54 INFO  AmazonHttpClient:90 - Configuring Proxy. Proxy Host: 165.225.104.34 Proxy Port: 9480
2018-03-15 18:23:54 INFO  TransferManagerUSZipsUpload:202 - -----successfully connected to s3------
2018-03-15 18:23:54 INFO  TransferManagerUSZipsUpload:217 - /****************************************************************************************/
2018-03-15 18:23:54 INFO  TransferManagerUSZipsUpload:219 - S3 uploader started for equipment number : 67387734
2018-03-15 18:23:54 INFO  TransferManagerUSZipsUpload:227 - Uploading  Zip File : Demo_US\ICS\DeviceData\US\Valid\67387734\2016\04\15\50b2ee5b-0fb9-4028-b89d-a749cf8ce654\15901147_US_795210_0000000000_20160415072158
2018-03-15 18:23:56 INFO  TransferManagerUSZipsUpload:240 - S3 uploader ends for equipment number : 67387734
2018-03-15 18:23:56 INFO  TransferManagerUSZipsUpload:241 - /****************************************************************************************/
2018-03-15 18:23:56 INFO  TransferManagerUSZipsUpload:293 - US Completed files are renamed succesfully
2018-03-15 18:23:56 INFO  HouseKeeping:347 - Equipment Number Directory is delete from local file system : C:\Users\310298203\git\HRDM\HistoricalRawDataMigration\01_20_2016-12_20_2018\ICS\DeviceData\US\Valid\6738773
2018-03-15 18:23:56 INFO  TransferManagerUSZipsUpload:139 - *******************************************************************
2018-03-15 18:23:56 INFO  TransferManagerUSZipsUpload:140 - S3 Upload Batch 3 is processing ...
2018-03-15 18:23:56 INFO  TransferManagerUSZipsUpload:141 - *******************************************************************
2018-03-15 18:23:56 INFO  AmazonHttpClient:90 - Configuring Proxy. Proxy Host: 165.225.104.34 Proxy Port: 9480
2018-03-15 18:23:56 INFO  TransferManagerUSZipsUpload:202 - -----successfully connected to s3------
2018-03-15 18:23:56 INFO  TransferManagerUSZipsUpload:217 - /****************************************************************************************/
```

Figure 5.14: Output for UploadUSModalityZipToS3



```
cf-s3-9b9752e2-008b-44a4-b6d4-c7b9fbbfe774
cf-s3-9bcdb782-a861-4df3-9d43-231b0fad06e0
cf-s3-9be3c396-d8ae-417b-af87-e4858560bc18
cf-s3-9bf75505-78f5-4ce1-b790-ac5c695e662e
cf-s3-9cda8834-be1a-40c1-bcfb-0d78796e6a81
cf-s3-9cf3943a-6b31-40df-8aa5-7b017e8614aa
cf-s3-9d083702-bba7-4fbd-8368-f5ab0b0cb756
cf-s3-9e1c5a69-988f-46dc-8c23-76f18928fc0e
cf-s3-9e75a8b2-9c06-422a-baa2-45574ad85d51
cf-s3-9f2403ab-5168-4287-94e5-eed1292404fc
cf-s3-9f2c135a-52cb-4e80-951c-fccd976d1358
cf-s3-a03b4185-6158-452f-81a4-9fb3c38f3193
cf-s3-a0608143-0c94-4992-a6e4-7959afad0469
cf-s3-a06d694a-7f53-4e1d-9ecc-095a4df9f511
cf-s3-a078340a-8995-4989-9f02-ec3f6614c5a6
cf-s3-a0853aea-ce6e-4a0e-a617-16a1e8d4bdc5
cf-s3-a0a28b37-4eea-42e0-a6fe-638df0ac9e10
cf-s3-a11642be-9f3d-43a6-8b53-b288b9a22f78
cf-s3-a24135b4-f5de-4d28-aee7-8ff0d954bb91
cf-s3-a27869ee-7aff-4c01-93bc-04c78789b542
cf-s3-a2fe3c58-a6cf-4878-b0d4-2faff9352e75
cf-s3-a34cd172-97b1-4889-b04c-d1d78637bc67
cf-s3-a35e3dad-91e0-4324-b51a-8e6a1af5ccd3
cf-s3-a3af8275-b0aa-441e-9fc7-f8189ffe383e
cf-s3-a3f868b3-8d62-4d71-84fb-9ea5de223e0c
cf-s3-a4498ae4-c32f-4f7d-a74e-bd237ce16979
```

```
Path:  /  Demo_US/  ICS/  DeviceData/  US/  Valid/  65674825/  2017/  10/  03/

File                                      Size   Type   Last Modified   Storage Class
..
0907c62a-9da5-4006-9bef-05c8bfd5ed51/
0a8b591e-3931-4ae4-a522-169072676458/
0bc70709-6bca-4d68-9ee9-99840f4a1af5/
165b4281-0660-4f73-a460-95be53bea846/
22d3ba66-60c8-404a-bdc6-4c79a9cb9d10/
3086748f-5965-464a-86f0-6f5f67d1ca14/
3a05eb67-9fb7-4279-aef0-c8a9915ae564/
3d034e28-8378-46d4-8673-495b1baab073/
548a6438-d990-4928-8416-042e347e6dba/
568d8dbc-5ee1-4fcf-80f7-59cf06f6b1da/
5778e277-ff59-4dd6-9a4b-8c68798e1788/
62378fdb-5e29-4873-9761-a056b28102f5/
6c84d303-df7d-4242-90b1-a3a4d8c2c432/
75bbb78a-cad5-48c2-8def-1ff7034e19b2/
7b24efdc-abdc-4b34-ac93-a9faaba0679e/
7c509d77-70c2-4709-8f29-aab36e3f2e4e/
7d470677-efdc-4bd4-affa-3eae349b3da4/
8702b0c3-61dd-47c2-b0a1-8a4a7e30e1ee/
9cd4d903-8eda-4f05-ba51-f25eae5310bd/
a4a99d90-da9d-432b-9694-51feba1a90e7/
```

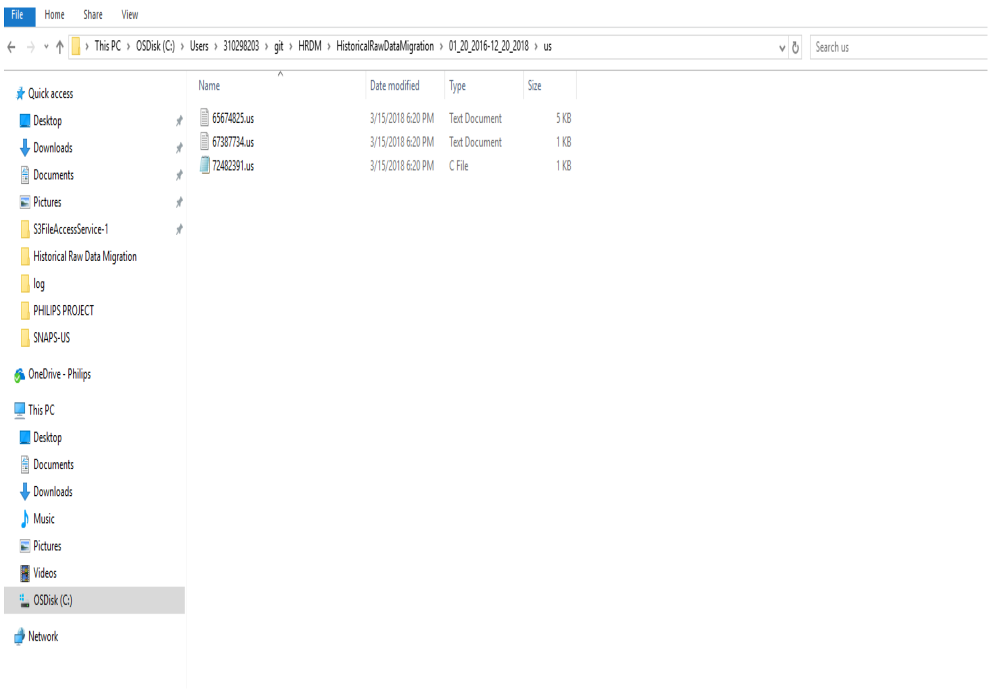
Figure 5.15: Uploaded Structured Data in Amazon S3

System metadata as following and export extracted data to Redshift in iX_cdf_posttest table:

Description: POST PDUPost Passed

PostTestName : PDUPost

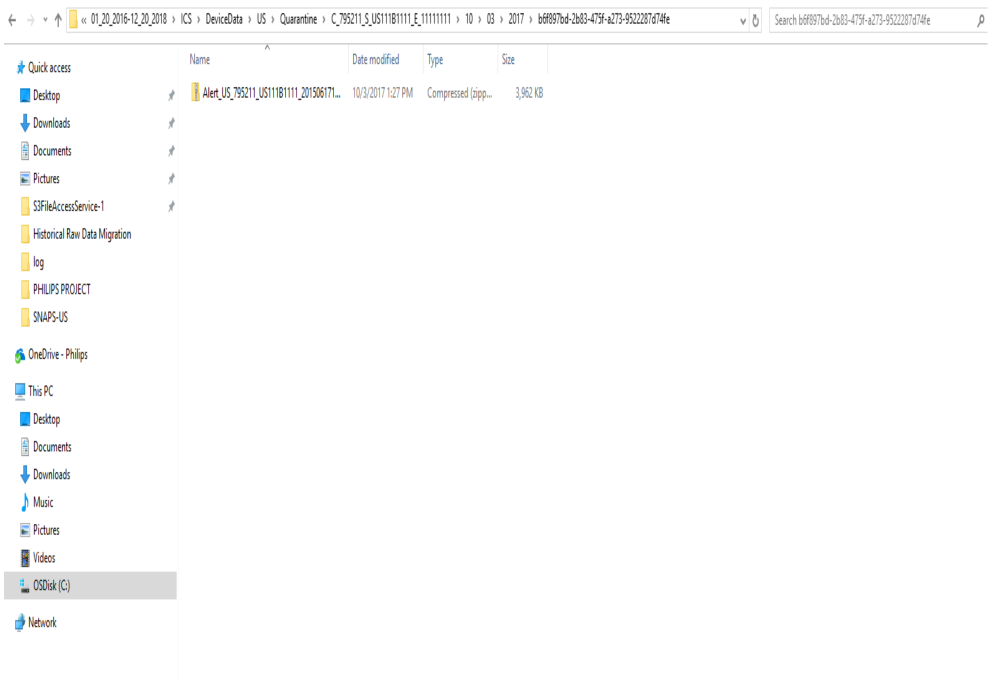
Result : Passed



This PC › OSDisk (C:) › Users › 310298203 › git › HRDM › HistoricalRawDataMigration › 01_20_2016-12_20_2018 › us

Name	Date modified	Type	Size
65674825.us	3/15/2018 6:20 PM	Text Document	5 KB
67387734.us	3/15/2018 6:20 PM	Text Document	1 KB
72482391.us	3/15/2018 6:20 PM	C File	1 KB

Figure 5.16: File State Changes after UploadZipToS3



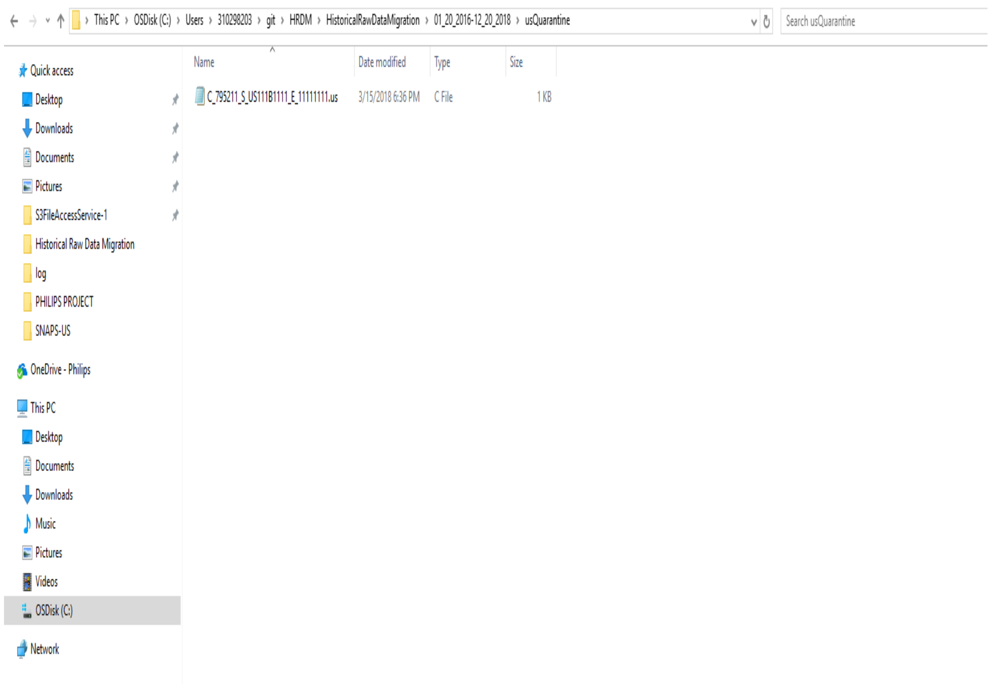
« 01_20_2016-12_20_2018 › ICS › DeviceData › US › Quarantine › C_795211_S_US111B1111_E_11111111 › 10 › 03 › 2017 › b6f897bd-2b83-475f-a273-9522287d74fe

Name	Date modified	Type	Size
Alert_US_795211_US111B1111_201506171...	10/3/2017 1:27 PM	Compressed (zipp...	3,962 KB

Figure 5.17: QuarantineZipFilePreparer

In addition,from addition Info it will take FRUName,FRUId,PostItemName,PostItemResult and related System metadata as follows and export extacted data to Redshift in iXR_cdf_postfru table:

Additional Info: involved **FRU's: PD.SCOMP.DCPS_24V_12V_5V code: 4598-**



Name	Date modified	Type	Size
C_795211_S_US111B1111_E_11111111.us	3/15/2018 6:36 PM	C File	1 KB

Figure 5.18: File State Changes After QuarantineZipFilePreparer

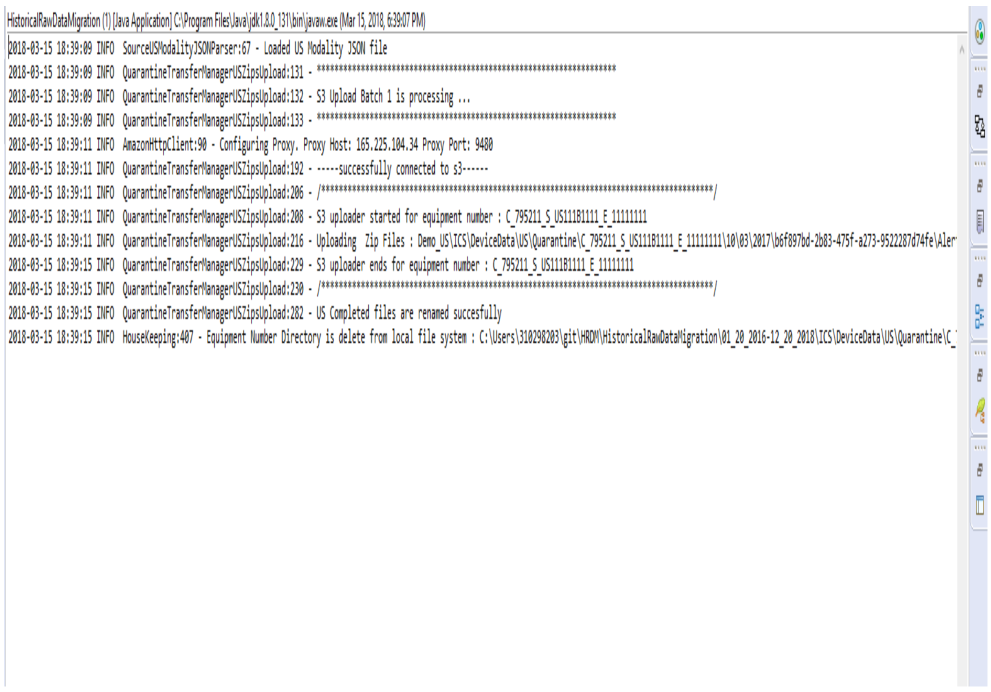
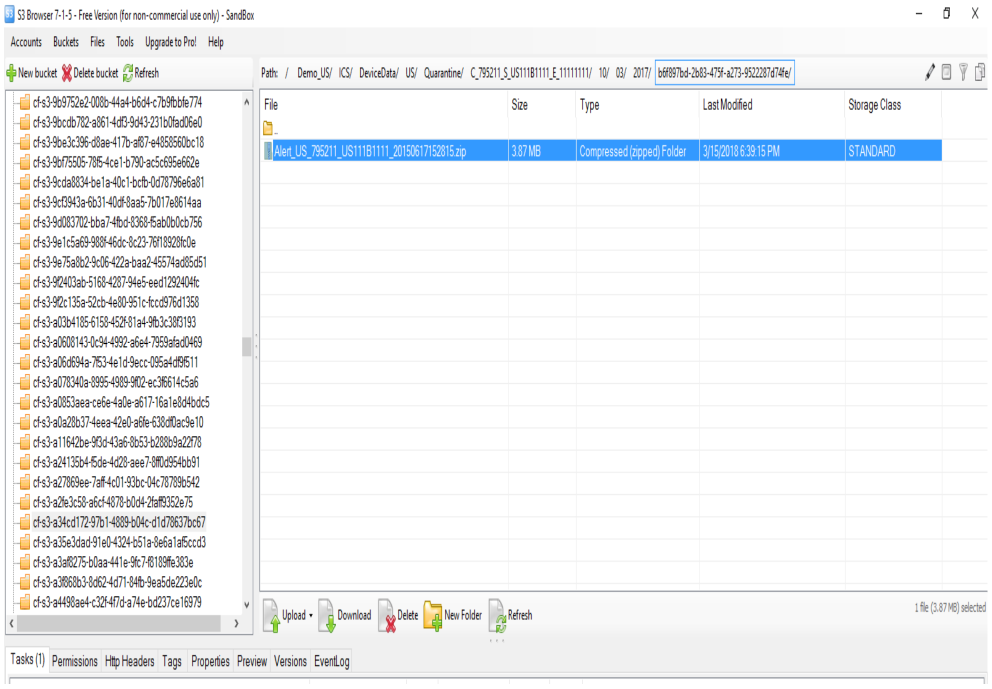

```
HistoricalRawDataMigration (1) [Java Application] C:\Program Files\Java\jdk1.8.0_131\bin\javaw.exe (Mar 15, 2018, 6:39:07 PM)
2018-03-15 18:39:09 INFO  SourceUSModalityJSONParser:67 - Loaded US Modality JSON file
2018-03-15 18:39:09 INFO  QuarantineTransferManagerUSZipsUpload:131 - ***************************************************************
2018-03-15 18:39:09 INFO  QuarantineTransferManagerUSZipsUpload:132 - S3 Upload Batch 1 is processing ...
2018-03-15 18:39:09 INFO  QuarantineTransferManagerUSZipsUpload:133 - ***************************************************************
2018-03-15 18:39:11 INFO  AmazonHttpClient:90 - Configuring Proxy. Proxy Host: 165.225.104.34 Proxy Port: 9480
2018-03-15 18:39:11 INFO  QuarantineTransferManagerUSZipsUpload:192 - -----successfully connected to s3------
2018-03-15 18:39:11 INFO  QuarantineTransferManagerUSZipsUpload:206 - /*********************************************************************************/
2018-03-15 18:39:11 INFO  QuarantineTransferManagerUSZipsUpload:208 - S3 uploader started for equipment number : C_795211_S_US111B1111_E_1111111
2018-03-15 18:39:11 INFO  QuarantineTransferManagerUSZipsUpload:216 - Uploading  Zip Files : Demo_US\ICS\DeviceData\US\Quarantine\C_795211_S_US111B1111_E_11111111\10\03\2017\b6f897bd-2b83-475f-a273-9522287d74fe\Aler
2018-03-15 18:39:15 INFO  QuarantineTransferManagerUSZipsUpload:229 - S3 uploader ends for equipment number : C_795211_S_US111B1111_E_1111111
2018-03-15 18:39:15 INFO  QuarantineTransferManagerUSZipsUpload:230 - /*********************************************************************************/
2018-03-15 18:39:15 INFO  QuarantineTransferManagerUSZipsUpload:282 - US Completed files are renamed succesfully
2018-03-15 18:39:15 INFO  HouseKeeping:407 - Equipment Number Directory is delete from local file system : C:\Users\310298203\git\HRDM\HistoricalRawDataMigration\01_20_2016-12_20_2018\ICS\DeviceData\US\Quarantine\C_
```

Figure 5.19: QuarantineUploadUSModalityZipToS3

003-8758.1 PD.ASSY.FRONTPANEL kit code: 4598-005-0223.1 PD.SCOMP.1 PUPS_BATTERY - UPS batterypack code: 4598-003-8757.1 PD.SBB.PRS_PPB - Pulse Power Bus relay code: 4598-003-8745.1 PD.SCOMP.TVSS - set of surge surpressors code: 4598-003-8740.1 PD.SCOMP.1PUPS_CTRL - UPS controller code: 4598-003-8756.1 PD .SCOMP.PCM_DC24V - PCM powersupply code: 4598-003-8747.1 PD. SCOMP.PROPIO_


S3 Browser 7-1-5 - Free Version (for non-commercial use only) - SandBox

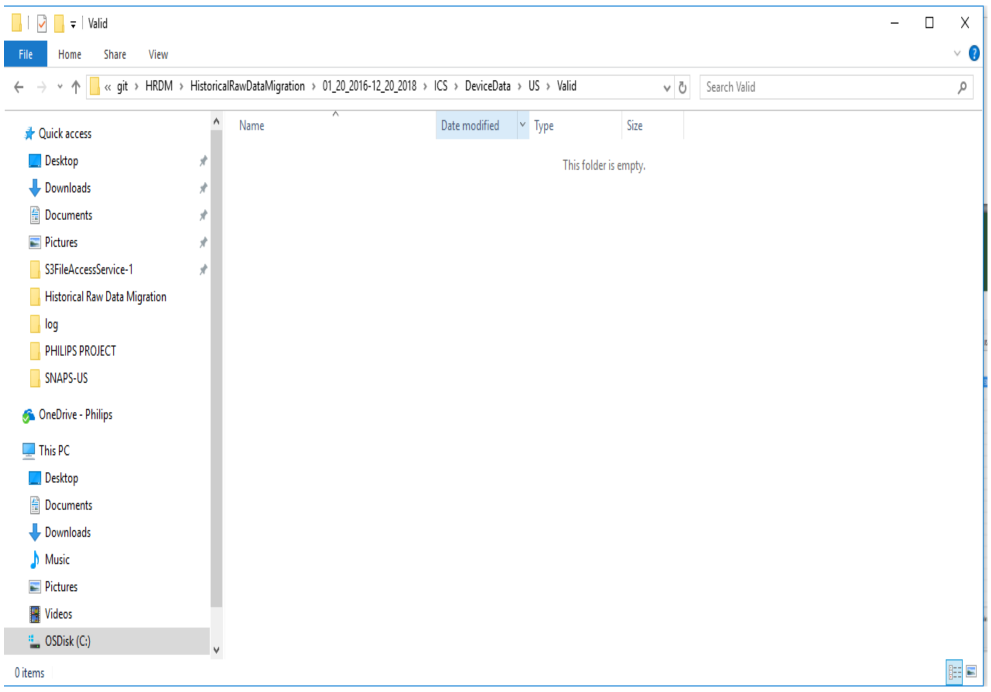
Path: / Demo_US/ ICS/ DeviceData/ US/ Quarantine/ C_795211_S_US111B1111_E_11111111/ 10/ 03/ 2017/ b6f897bd-2b83-475f-a273-9522287d74fe/

```
cf-s3-9b9752e2-008b-44a4-b6d4-c7b9fbbfe774
cf-s3-9bcdb782-a861-4df3-9d43-231b0fad06e0
cf-s3-9be3c396-d8ae-417b-af87-e4858560bc18
cf-s3-9bf75505-78f5-4ce1-b790-ac5c695e662e
cf-s3-9cda8834-be1a-40c1-bcfb-0d78796e6a81
cf-s3-9cf3943a-6b31-40df-8aa5-7b017e8614aa
cf-s3-9d083702-bba7-4fbd-8368-f5ab0b0cb756
cf-s3-9e1c5a69-988f-46dc-8c23-76f18928fc0e
cf-s3-9e75a8b2-9c06-422a-baa2-45574ad85d51
cf-s3-9f2403ab-5168-4287-94e5-eed1292404fc
cf-s3-9f2c135a-52cb-4e80-951c-fccd976d1358
cf-s3-a03b4185-6158-452f-81a4-9fb3c38f3193
cf-s3-a0608143-0c94-4992-a6e4-7959afad0469
cf-s3-a06d694a-7f53-4e1d-9ecc-095a4df9f511
cf-s3-a078340a-8995-4989-9f02-ec3f6614c5a6
cf-s3-a0853aea-ce6e-4a0e-a617-16a1e8d4bdc5
cf-s3-a0a28b37-4eea-42e0-a6fe-638df0ac9e10
cf-s3-a11642be-9f3d-43a6-8b53-b288b9a22f78
cf-s3-a24135b4-f5de-4d28-aee7-8ff0d954bb91
cf-s3-a27869ee-7aff-4c01-93bc-04c78789b542
cf-s3-a2fe3c58-a6cf-4878-b0d4-2faff9352e75
cf-s3-a34cd172-97b1-4889-b04c-d1d78637bc67
cf-s3-a35e3dad-91e0-4324-b51a-8e6a1af5ccd3
cf-s3-a3af8275-b0aa-441e-9fc7-f8189ffe383e
cf-s3-a3f868b3-8d62-4d71-84fb-9ea5de223e0c
cf-s3-a4498ae4-c32f-4f7d-a74e-bd237ce16979
```

File	Size	Type	Last Modified	Storage Class
..				
Alert_US_795211_US111B1111_20150617152815.zip	3.87 MB	Compressed (zipped) Folder	3/15/2018 6:39:15 PM	STANDARD

1 file (3.87 MB) selected

Figure 5.20: Uploaded Quarantine Structured Data in Amazon S3


Valid

« git › HRDM › HistoricalRawDataMigration › 01_20_2016-12_20_2018 › ICS › DeviceData › US › Valid

Name	Date modified	Type	Size

This folder is empty.

0 items

Figure 5.21: House Keeping

term - IO relay board code: 4598-003-8751.1 PD.ASSY.BASE_TOPWORKS kit code: 4598-005-0226.1 PD.SBB.PRS_PB15KVA - Power Bus relay code: 4598-003-8744.1 PD.SBB .MDS - Mains Disconnect Switch code: 4598-003-8739.1 PD.COMP.FUSES code: 4598-005-4598.1 PD.ASSY.BASE_FAN UNIT kit code: 4598-005-0846.1 PD.ASSY.PDM kit code: 4598-005-0225.1 PD.ASSY.REARPANEL kit code: 4598-005-0224.1 PD.SCOMP.PCM _uSD -card code: 4598-003-8749.1 **POSTItem: PDU SelfTest Done/Passed** started:06 /12/2017 7:28:32 AM ended:06/12/2017 7:28:34 AM TRUE PDUPost

FRUName: PD.SCOMP.DCPS_24V_12V_5V
FRUId: 4598-003-8758.1
PostItemName: PDU SelfTest
PostItemResult: Passed

5.4 S3 File Access Service-Input and Output Samples

As per mention in **??** Date,Modality and FileType are mandatory parameters.

5.4.1 Input Request Sample with Pagination and Published File Type

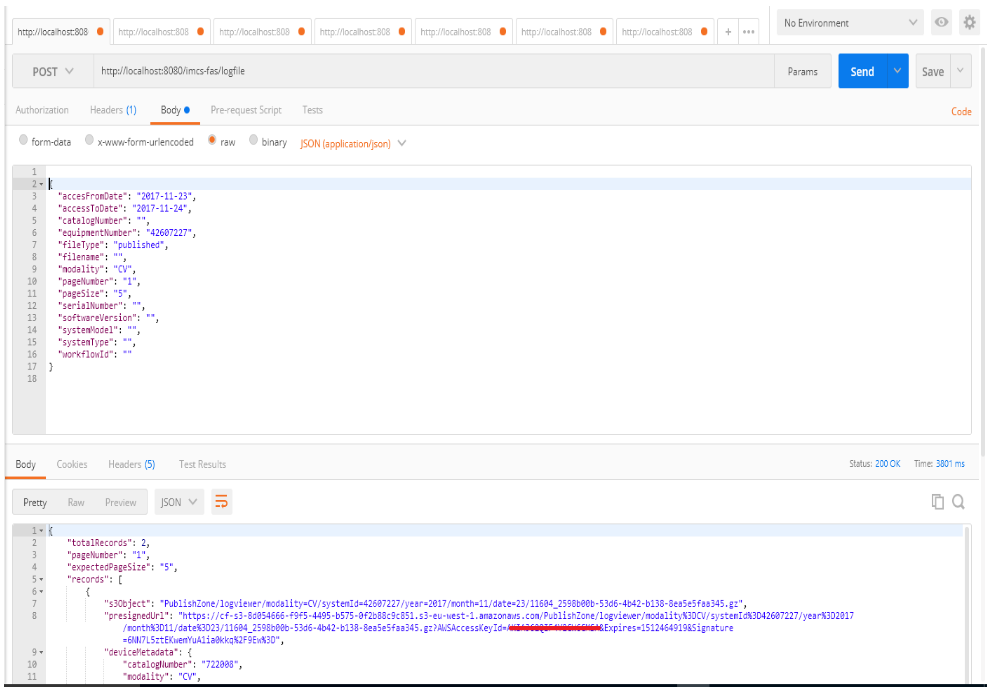
[
"accesFromDate": "2017-10-20",
"accessToDate": "2017-10-23",
"catalogNumber": "",
"equipmentNumber": "12345",
"fileType": "published",
"filename": "",
"modality": "CV",
"pageNumber": "1",
"pageSize": "5",
"serialNumber": "",
"softwareVersion": "",
"systemModel": "",
"systemType": "",
"workflowId": ""
]

As per request user will get records from database,presigned URLs which satisfied above request.In addition to that user will directly get speified page number and page size.

Output

```
[
"totalRecords": 1,
"pageNumber": "1",
"expectedPageSize": "5",
"records": [
[
"s3Object":
"PublishZone/logviewer/modality=CV/systemId=12345/year=2017/month=10/date=23/
logfilename.gz",
"presignedUrl": "https://bucketname.s3-eu-west-1.amazonaws.com/PublishZone/logviewer/
modality%3DCV/systemId%3D12345/year%3D2017/month%3D10/date%3D23/lofilename.gz/
AWSAccessKeyId=1901120290290&Expires=1511856951&Signature=ueijfijf19283%3D",
"deviceMetadata":
[
"catalogNumber": "72345",
"modality": "CV",
"serialNumber": "109",
"systemModel": "F10 u",
"systemType": "Allura XPer",
"systemuid": "12345",
"equipmentNumber": "12345",
"softwareVersion": "1.2.9.0",
]
"s3ObjectMetadata":
[
"size": "1400300",
"lastModified": "2017-10-27 13:27:24",
],
"presignedUrlExpiration": "2017-10-28 13:45:51",
]
```

],
"pageSize": 1
]



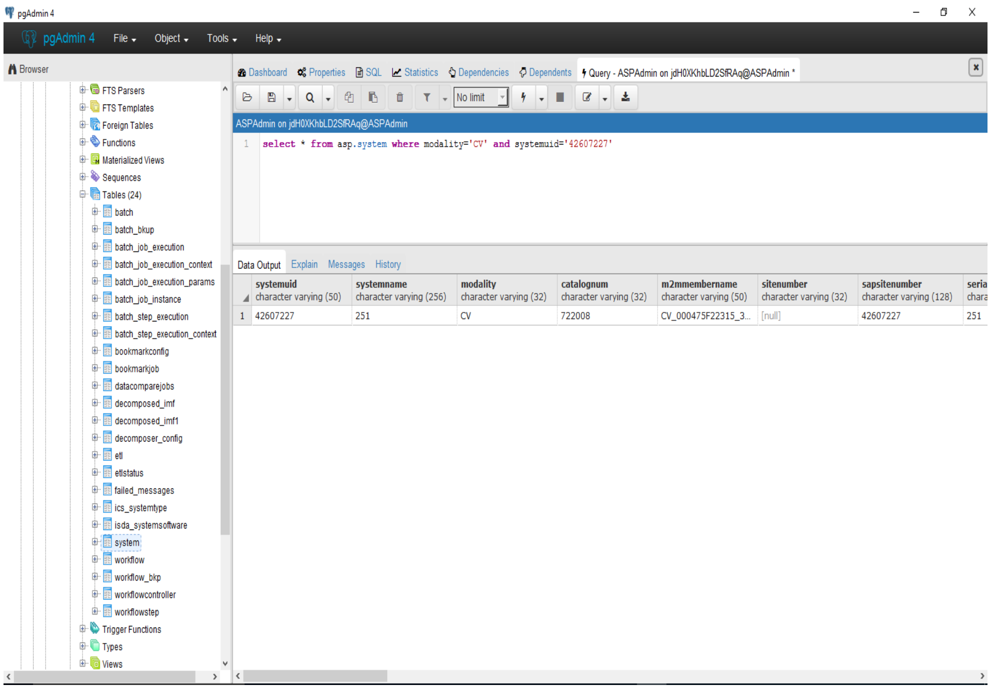
POST http://localhost:8080/imcs-fas/logfile

```

{
  "accesFromDate": "2017-11-23",
  "accessToDate": "2017-11-24",
  "catalogNumber": "",
  "equipmentNumber": "42607227",
  "fileType": "published",
  "filename": "",
  "modality": "CV",
  "pageNumber": "1",
  "pageSize": "5",
  "serialNumber": "",
  "softwareVersion": "",
  "systemModel": "",
  "systemType": "",
  "workflowId": ""
}

```

Status: 200 OK Time: 3801 ms

```
{
    "totalRecords": 2,
    "pageNumber": "1",
    "expectedPageSize": "5",
    "records": [
        {
            "s3Object": "PublishZone/logviewer/modality=CV/systemId=42607227/year=2017/month=11/date=23/11604_2598b00b-53d6-4b42-b138-8ea5e5faa345.gz",
            "presignedUrl": "https://cf-s3-8d054666-f9f5-4495-b575-0f2b88c9c851.s3-eu-west-1.amazonaws.com/PublishZone/logviewer/modality%3DCV/systemId%3D42607227/year%3D2017
              /month%3D11/date%3D23/11604_2598b00b-53d6-4b42-b138-8ea5e5faa345.gz?AWSAccessKeyId=[redacted]&Expires=1512464919&Signature
              =6NN7L5ztEKwemYuA1ia0kkq%2F9Ew%3D",
            "deviceMetadata": {
                "catalogNumber": "722008",
                "modality": "CV",
```

Figure 5.22: Input request sample with pagination

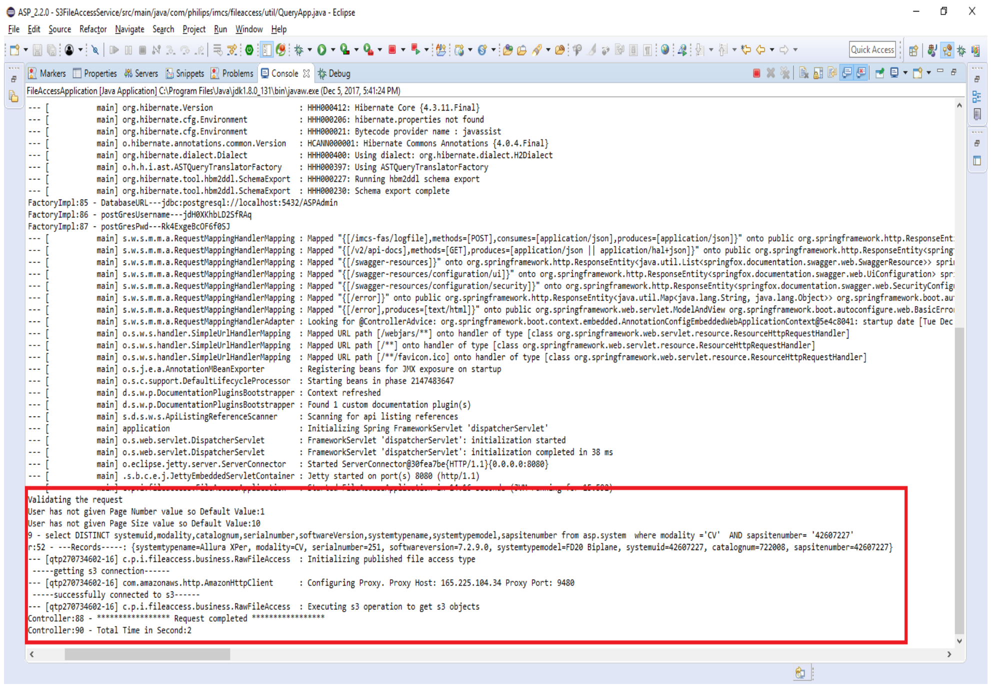


ASPAdmin on jdH0XKhbLD2SfRAq@ASPAdmin

```sql
select * from asp.system where modality='CV' and systemuid='42607227'
```

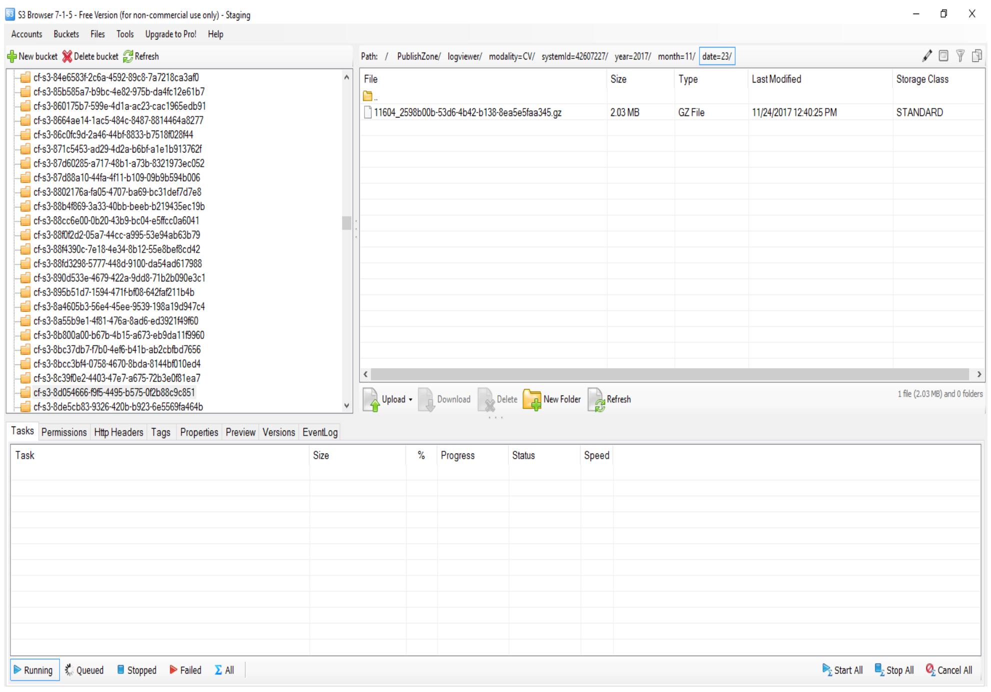
	systemuid character varying (50)	systemname character varying (256)	modality character varying (32)	catalognum character varying (32)	m2mmembername character varying (50)	sitenumber character varying (32)	sapsitenumber character varying (128)	seria chara
1	42607227	251	CV	722008	CV_000475F22315_3...	[null]	42607227	251

Figure 5.23: Data in postgreSQL


ASP_2.2.0 - S3FileAccessService/src/main/java/com/philips/imcs/fileaccess/util/QueryApp.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Markers Properties Servers Snippets Problems Console Debug

```
FileAccessApplication [Java Application] C:\Program Files\Java\jdk1.8.0_131\bin\javaw.exe (Dec 5, 2017, 5:41:24 PM)
--- [           main] org.hibernate.Version                    : HHH000412: Hibernate Core {4.3.11.Final}
--- [           main] org.hibernate.cfg.Environment            : HHH000206: hibernate.properties not found
--- [           main] org.hibernate.cfg.Environment            : HHH000021: Bytecode provider name : javassist
--- [           main] o.hibernate.annotations.common.Version   : HCANN000001: Hibernate Commons Annotations {4.0.4.Final}
--- [           main] org.hibernate.dialect.Dialect            : HHH000400: Using dialect: org.hibernate.dialect.H2Dialect
--- [           main] o.h.h.i.ast.ASTQueryTranslatorFactory    : HHH000397: Using ASTQueryTranslatorFactory
--- [           main] org.hibernate.tool.hbm2ddl.SchemaExport  : HHH000227: Running hbm2ddl schema export
--- [           main] org.hibernate.tool.hbm2ddl.SchemaExport  : HHH000230: Schema export complete
FactoryImpl:85 - DatabaseURL---jdbc:postgresql://localhost:5432/ASPAdmin
FactoryImpl:86 - postGresUsername---jdH0XKhbLD2SfRAq
FactoryImpl:87 - postGresPwd---Rk4ExgeBcOF6f0SJ
--- [           main] s.w.s.m.m.a.RequestMappingHandlerMapping : Mapped "{[/imcs-fas/logfile],methods=[POST],consumes=[application/json],produces=[application/json]}" onto public org.springframework.http.ResponseEnt
--- [           main] s.w.s.m.m.a.RequestMappingHandlerMapping : Mapped "{[/v2/api-docs],methods=[GET],produces=[application/json || application/hal+json]}" onto public org.springframework.http.ResponseEntity<spring
--- [           main] s.w.s.m.m.a.RequestMappingHandlerMapping : Mapped "{[/swagger-resources]}" onto org.springframework.http.ResponseEntity<java.util.List<springfox.documentation.swagger.web.SwaggerResource>> spri
--- [           main] s.w.s.m.m.a.RequestMappingHandlerMapping : Mapped "{[/swagger-resources/configuration/ui]}" onto org.springframework.http.ResponseEntity<springfox.documentation.swagger.web.UiConfiguration> spr
--- [           main] s.w.s.m.m.a.RequestMappingHandlerMapping : Mapped "{[/swagger-resources/configuration/security]}" onto org.springframework.http.ResponseEntity<springfox.documentation.swagger.web.SecurityConfig
--- [           main] s.w.s.m.m.a.RequestMappingHandlerMapping : Mapped "{[/error]}" onto public org.springframework.http.ResponseEntity<java.util.Map<java.lang.String, java.lang.Object>> org.springframework.boot.au
--- [           main] s.w.s.m.m.a.RequestMappingHandlerMapping : Mapped "{[/error],produces=[text/html]}" onto public org.springframework.web.servlet.ModelAndView org.springframework.boot.autoconfigure.web.BasicErro
--- [           main] s.w.s.m.m.a.RequestMappingHandlerAdapter : Looking for @ControllerAdvice: org.springframework.boot.context.embedded.AnnotationConfigEmbeddedWebApplicationContext@5e4c8041: startup date [Tue Dec
--- [           main] o.s.w.s.handler.SimpleUrlHandlerMapping  : Mapped URL path [/webjars/**] onto handler of type [class org.springframework.web.servlet.resource.ResourceHttpRequestHandler]
--- [           main] o.s.w.s.handler.SimpleUrlHandlerMapping  : Mapped URL path [/**] onto handler of type [class org.springframework.web.servlet.resource.ResourceHttpRequestHandler]
--- [           main] o.s.w.s.handler.SimpleUrlHandlerMapping  : Mapped URL path [/**/favicon.ico] onto handler of type [class org.springframework.web.servlet.resource.ResourceHttpRequestHandler]
--- [           main] o.s.j.e.a.AnnotationMBeanExporter        : Registering beans for JMX exposure on startup
--- [           main] o.s.c.support.DefaultLifecycleProcessor  : Starting beans in phase 2147483647
--- [           main] d.s.w.p.DocumentationPluginsBootstrapper : Context refreshed
--- [           main] d.s.w.p.DocumentationPluginsBootstrapper : Found 1 custom documentation plugin(s)
--- [           main] s.d.s.w.s.ApiListingReferenceScanner     : Scanning for api listing references
--- [           main] application                              : Initializing Spring FrameworkServlet 'dispatcherServlet'
--- [           main] o.s.web.servlet.DispatcherServlet        : FrameworkServlet 'dispatcherServlet': initialization started
--- [           main] o.s.web.servlet.DispatcherServlet        : FrameworkServlet 'dispatcherServlet': initialization completed in 38 ms
--- [           main] o.eclipse.jetty.server.ServerConnector   : Started ServerConnector@30fea7be{HTTP/1.1}{0.0.0.0:8080}
--- [           main] .s.b.c.e.j.JettyEmbeddedServletContainer : Jetty started on port(s) 8080 (http/1.1)
--- [           main] c.p.i.fileaccess.FileAccessApplication   : Started FileAccessApplication in 14.16 seconds (JVM running for 15.502)
Validating the request
User has not given Page Number value so Default Value:1
User has not given Page Size value so Default Value:10
9 - select DISTINCT systemuid,modality,catalognum,serialnumber,softwareVersion,systemtypename,systemtypemodel,sapsitenumber from asp.system  where modality ='CV'  AND sapsitenumber= '42607227'
r:52 - ---Records-----: {systemtypename=Allura XPer, modality=CV, serialnumber=251, softwareversion=7.2.9.0, systemtypemodel=FD20 Biplane, systemuid=42607227, catalognum=722008, sapsitenumber=42607227}
--- [qtp270734602-16] c.p.i.fileaccess.business.RawFileAccess  : Initializing published file access type
 -----getting s3 connection------
--- [qtp270734602-16] com.amazonaws.http.AmazonHttpClient      : Configuring Proxy. Proxy Host: 165.225.104.34 Proxy Port: 9480
 -----successfully connected to s3------
--- [qtp270734602-16] c.p.i.fileaccess.business.RawFileAccess  : Executing s3 operation to get s3 objects
Controller:88 - **************** Request completed ****************
Controller:90 - Total Time in Second:2
```

Figure 5.24: Display record from database


S3 Browser 7-1-5 - Free Version (for non-commercial use only) - Staging
Accounts Buckets Files Tools Upgrade to Pro! Help
New bucket Delete bucket Refresh
Upload Download Delete New Folder Refresh
Tasks Permissions Http Headers Tags Properties Preview Versions EventLog
Running Queued Stopped Failed All
Start All Stop All Cancel All

```
cf-s3-84e6583f-2c6a-4592-89c8-7a7218ca3af0
cf-s3-85b585a7-b9bc-4e82-975b-da4fc12e61b7
cf-s3-860175b7-599e-4d1a-ac23-cac1965edb91
cf-s3-8664ae14-1ac5-484c-8487-8814464a8277
cf-s3-86c0fc9d-2a46-44bf-8833-b7518f028f44
cf-s3-871c5453-ad29-4d2a-b6bf-a1e1b913762f
cf-s3-87d60285-a717-48b1-a73b-8321973ec052
cf-s3-87d88a10-44fa-4f11-b109-09b9b594b006
cf-s3-8802176a-fa05-4707-ba69-bc31def7d7e8
cf-s3-88b4f869-3a33-40bb-beeb-b219435ec19b
cf-s3-88cc6e00-0b20-43b9-bc04-e5ffcc0a6041
cf-s3-88f0f2d2-05a7-44cc-a995-53e94ab63b79
cf-s3-88f4390c-7e18-4e34-8b12-55e8bef8cd42
cf-s3-88fd3298-5777-448d-9100-da54ad617988
cf-s3-890d533e-4679-422a-9dd8-71b2b090e3c1
cf-s3-895b51d7-1594-471f-bf08-642faf211b4b
cf-s3-8a4605b3-56e4-45ee-9539-198a19d947c4
cf-s3-8a55b9e1-4f81-476a-8ad6-ed3921f49f60
cf-s3-8b800a00-b67b-4b15-a673-eb9da11f9960
cf-s3-8bc37db7-f7b0-4ef6-b41b-ab2cbfbd7656
cf-s3-8bcc3bf4-0758-4670-8bda-8144bf010ed4
cf-s3-8c39f0e2-4403-47e7-a675-72b3e0f81ea7
cf-s3-8d054666-f9f5-4495-b575-0f2b88c9c851
cf-s3-8de5cb83-9326-420b-b923-6e5569fa464b
```

Path: / PublishZone/ logviewer/ modality=CV/ systemId=42607227/ year=2017/ month=11/ date=23/

File	Size	Type	Last Modified	Storage Class
..				
11604_2598b00b-53d6-4b42-b138-8ea5e5faa345.gz	2.03 MB	GZ File	11/24/2017 12:40:25 PM	STANDARD

1 file (2.03 MB) and 0 folders

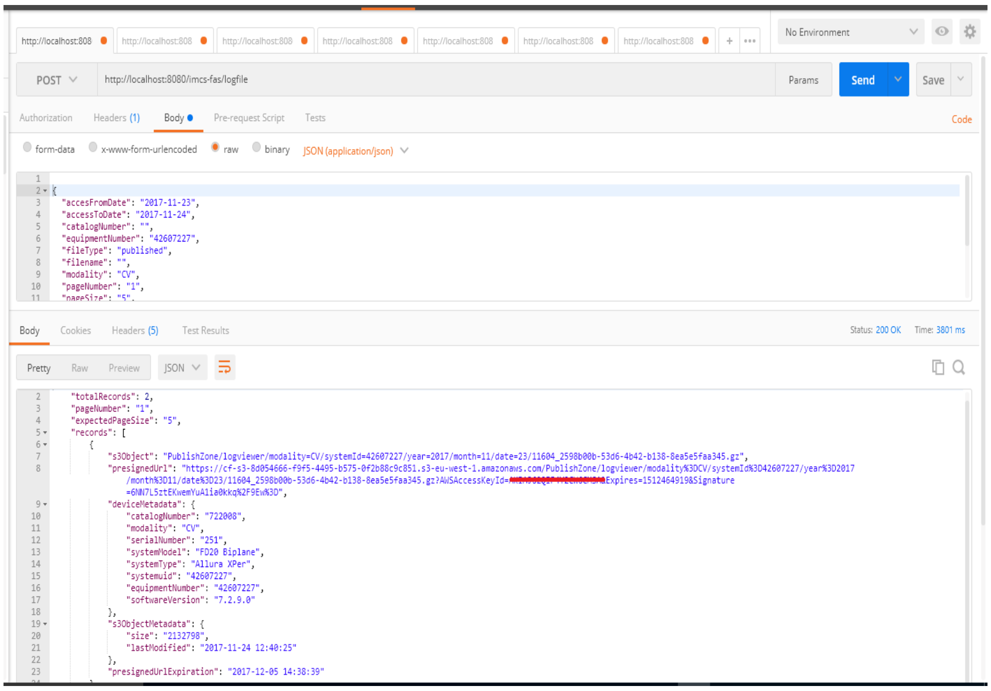
Task	Size	%	Progress	Status	Speed

Figure 5.25: Amazon s3 with bucketname and file path

5.4.2 Input Request Sample without Pagination and Raw File Type

[

"accesFromDate": "2017-05-20",



POST http://localhost:8080/imcs-fas/logfile

```
{
  "accesFromDate": "2017-11-23",
  "accessToDate": "2017-11-24",
  "catalogNumber": "",
  "equipmentNumber": "42607227",
  "fileType": "published",
  "filename": "",
  "modality": "CV",
  "pageNumber": "1",
  "pageSize": "5",
```

```
    "totalRecords": 2,
    "pageNumber": "1",
    "expectedPageSize": "5",
    "records": [
        {
            "s3Object": "PublishZone/logviewer/modality=CV/systemId=42607227/year=2017/month=11/date=23/11604_2598b00b-53d6-4b42-b138-8ea5e5faa345.gz",
            "presignedUrl": "https://cf-s3-8d054666-f9f5-4495-b575-0f2b88c9c851.s3-eu-west-1.amazonaws.com/PublishZone/logviewer/modality%3DCV/systemId%3D42607227/year%3D2017
                /month%3D11/date%3D23/11604_2598b00b-53d6-4b42-b138-8ea5e5faa345.gz?AWSAccessKeyId=[redacted]Expires=1512464919&Signature
                =6NN7L5ztEKwemYuA1ia0kkq%2F9Ew%3D",
            "deviceMetadata": {
                "catalogNumber": "722008",
                "modality": "CV",
                "serialNumber": "251",
                "systemModel": "FD20 Biplane",
                "systemType": "Allura XPer",
                "systemuid": "42607227",
                "equipmentNumber": "42607227",
                "softwareVersion": "7.2.9.0"
            },
            "s3ObjectMetadata": {
                "size": "2132798",
                "lastModified": "2017-11-24 12:40:25"
            },
            "presignedUrlExpiration": "2017-12-05 14:38:39"
```

Figure 5.26: Output sample with pagination

"accessToDate": "2017-05-23",

"catalogNumber": "",

"equipmentNumber": "",

"fileType": "raw",

"filename": "",

"modality": "MR",

"pageNumber": ",

"pageSize": "",

"serialNumber": "23456",

"softwareVersion": "",

"systemModel": "",

"systemType": "",

"workflowId": ""

]

Here if user will not give any input for pagination(page number and page size).S3 File Access Service consider default values of page number and page size 1 and 10 respectively as mention in table 1.1.

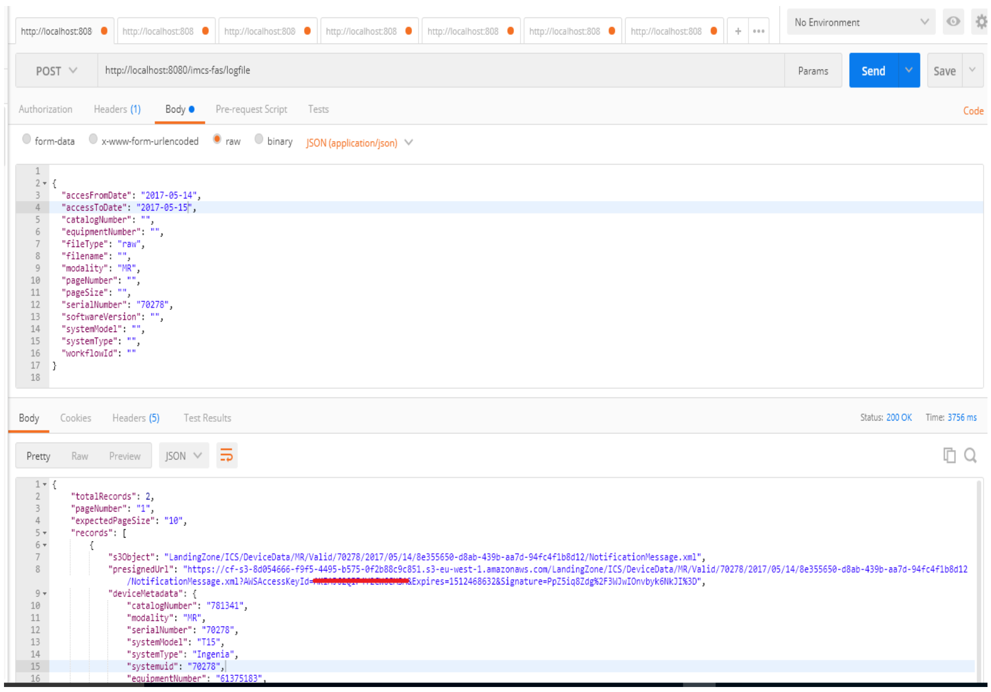
Output

[
"totalRecords": 2,
"pageNumber": "1",
"expectedPageSize": "10",
"records": [
[
"s3Object":
"LandingZone/ICS/DeviceData/MR/Valid/23456/2017/05/13/foldernameinS3bucketname/
logfilename.xml",
"presignedUrl":"https://bucketname.s3-eu-west-1.amazonaws.com/LandingZone/ICS/DeviceData
/MR/Valid/23456/2017/05/13/foldername/logfilename.xml/?AWSAccessKeyId=1901120290290
&Expires=1512018595&Signatur= ueijfijf192djnckjfdn83%3D",
"deviceMetadata":
[
"catalogNumber": "909909",
"modality": "MR",
"serialNumber": "23456",
"systemModel": "T15",
"systemType": "Achieva",
"systemuid": "23456",
"equipmentNumber": "PHC-1921",
"softwareVersion": "5.3.0.1"
],
"s3ObjectMetadata":
[
"size": "32469",
"lastModified": "2017-05-13 08:09:52"
],
"presignedUrlExpiration": "2017-11-30 10:39:55"
],
[
"s3Object":

”LandingZone/ICS/DeviceData/MR/Valid/23456/2017/05/13/foldernameinS3bucket/
logfilename.zip”,
”presignedUrl”:
”https://bucketname.s3-eu-west-1.amazonaws.com/LandingZone/ICS/DeviceData
/MR/Valid/23456/2017/05/13/foldernameinS3bucket/logfilename.zip?
AWSAccessKeyId=1901120290290&Expires=1512018595&Signature=djnvfjidvjldfvkf%3D”,
”deviceMetadata”:
[
”catalogNumber”: ”909909”,
”modality”: ”MR”,
”serialNumber”: ”23456”,
”systemModel”: ”T15”,
”systemType”: ”Achieva”,
”systemuid”: ”23456”,
”equipmentNumber”: ”PHC-1921”,
”softwareVersion”: ”5.3.0.1”
],
”s3ObjectMetadata”:
[
”size”: ”22286521”,
”lastModified”: ”2017-05-13 08:09:52”
],
”presignedUrlExpiration”: ”2017-11-30 10:39:55”
]
],
”pageSize”: 2
]

5.4.3 Input Request Sample with workflowId

If is able to access only LOD files by giving workflowid. [
”accesFromDate”: ”2017-10-20”,



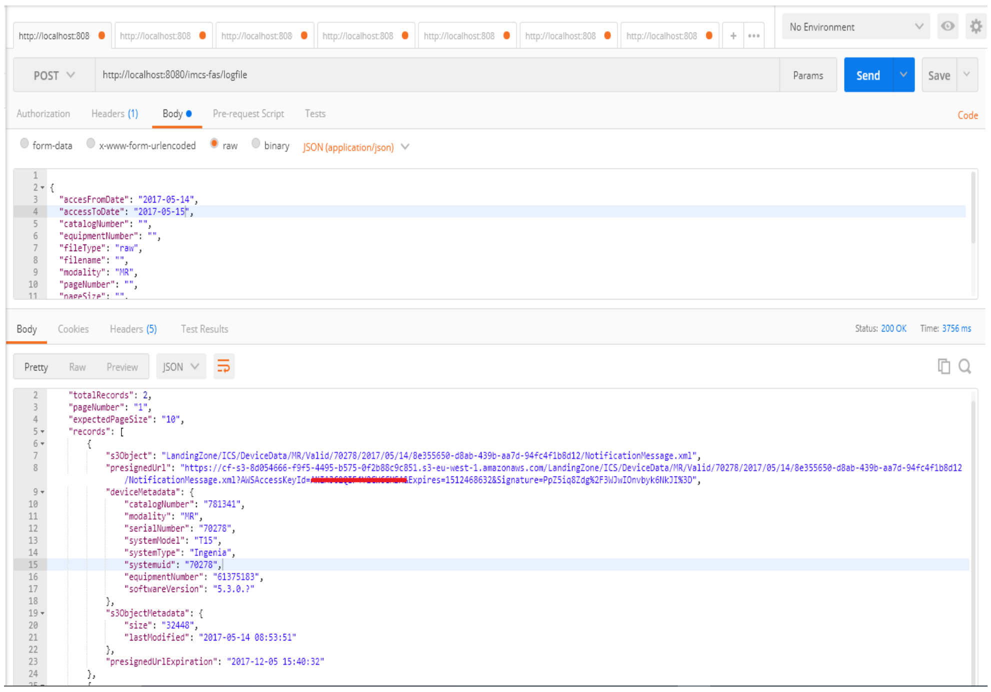
```
http://localhost:8080/imcs-fas/logfile
```

```
{
  "accesFromDate": "2017-05-14",
  "accessToDate": "2017-05-15",
  "catalogNumber": "",
  "equipmentNumber": "",
  "fileType": "raw",
  "filename": "",
  "modality": "MR",
  "pageNumber": "",
  "pageSize": "",
  "serialNumber": "70278",
  "softwareVersion": "",
  "systemModel": "",
  "systemType": "",
  "workflowId": ""
}
```

Status: 200 OK Time: 3756 ms

```
{
    "totalRecords": 2,
    "pageNumber": "1",
    "expectedPageSize": "10",
    "records": [
        {
            "s3Object": "LandingZone/ICS/DeviceData/MR/Valid/70278/2017/05/14/8e355650-d8ab-439b-aa7d-94fc4f1b8d12/NotificationMessage.xml",
            "presignedUrl": "https://cf-s3-8d054666-f9f5-4495-b575-0f2b88c9c851.s3-eu-west-1.amazonaws.com/LandingZone/ICS/DeviceData/MR/Valid/70278/2017/05/14/8e355650-d8ab-439b-aa7d-94fc4f1b8d12
                /NotificationMessage.xml?AWSAccessKeyId=[redacted]&Expires=1512468632&Signature=PpZ5iq8Zdg%2F3WJwIOnvbyk6NkJI%3D",
            "deviceMetadata": {
                "catalogNumber": "781341",
                "modality": "MR",
                "serialNumber": "70278",
                "systemModel": "T15",
                "systemType": "Ingenia",
                "systemuid": "70278",
                "equipmentNumber": "61375183",
```

Figure 5.27: Input request sample without pagination



```
http://localhost:8080/imcs-fas/logfile
```

```
{
  "accesFromDate": "2017-05-14",
  "accessToDate": "2017-05-15",
  "catalogNumber": "",
  "equipmentNumber": "",
  "fileType": "raw",
  "filename": "",
  "modality": "MR",
  "pageNumber": "",
  "pageSize": "",
```

Status: 200 OK Time: 3756 ms

```
    "totalRecords": 2,
    "pageNumber": "1",
    "expectedPageSize": "10",
    "records": [
        {
            "s3Object": "LandingZone/ICS/DeviceData/MR/Valid/70278/2017/05/14/8e355650-d8ab-439b-aa7d-94fc4f1b8d12/NotificationMessage.xml",
            "presignedUrl": "https://cf-s3-8d054666-f9f5-4495-b575-0f2b88c9c851.s3-eu-west-1.amazonaws.com/LandingZone/ICS/DeviceData/MR/Valid/70278/2017/05/14/8e355650-d8ab-439b-aa7d-94fc4f1b8d12
                /NotificationMessage.xml?AWSAccessKeyId=[redacted]&Expires=1512468632&Signature=PpZ5iq8Zdg%2F3WJwIOnvbyk6NkJI%3D",
            "deviceMetadata": {
                "catalogNumber": "781341",
                "modality": "MR",
                "serialNumber": "70278",
                "systemModel": "T15",
                "systemType": "Ingenia",
                "systemuid": "70278",
                "equipmentNumber": "61375183",
                "softwareVersion": "5.3.0.?"
            },
            "s3ObjectMetadata": {
                "size": "32448",
                "lastModified": "2017-05-14 08:53:51"
            },
            "presignedUrlExpiration": "2017-12-05 15:40:32"
        },
```

Figure 5.28: Output sample without pagination first record

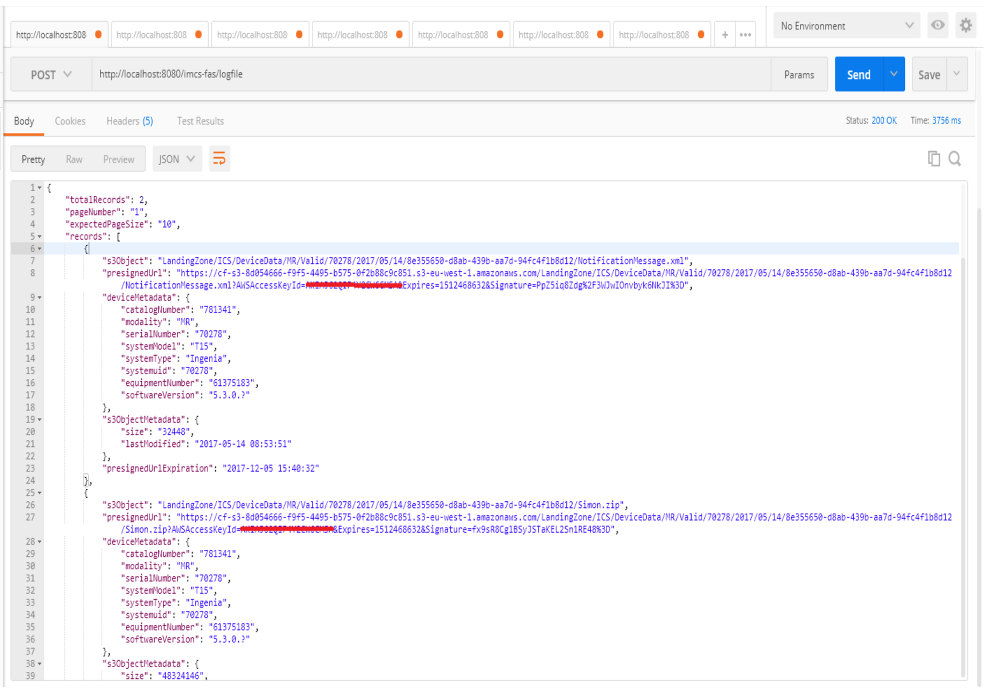
"accessToDate": "2017-10-23",

"catalogNumber": "",

"equipmentNumber": "1000",

"fileType": "published",

"filename": "",



```
{
    "totalRecords": 2,
    "pageNumber": "1",
    "expectedPageSize": "10",
    "records": [
        {
            "s3Object": "LandingZone/ICS/DeviceData/MR/Valid/70278/2017/05/14/8e355650-d8ab-439b-aa7d-94fc4f1b8d12/NotificationMessage.xml",
            "presignedUrl": "https://cf-s3-8d054666-f9f5-4495-b575-0f2b88c9c851.s3-eu-west-1.amazonaws.com/LandingZone/ICS/DeviceData/MR/Valid/70278/2017/05/14/8e355650-d8ab-439b-aa7d-94fc4f1b8d12
                /NotificationMessage.xml?AWSAccessKeyId=[redacted]Expires=1512468632&Signature=PpZ5iq8Zdg%2F3WJwIOnvbyk6NkJI%3D",
            "deviceMetadata": {
                "catalogNumber": "781341",
                "modality": "MR",
                "serialNumber": "70278",
                "systemModel": "T15",
                "systemType": "Ingenia",
                "systemuid": "70278",
                "equipmentNumber": "61375183",
                "softwareVersion": "5.3.0.?"
            },
            "s3ObjectMetadata": {
                "size": "32448",
                "lastModified": "2017-05-14 08:53:51"
            },
            "presignedUrlExpiration": "2017-12-05 15:40:32"
        },
        {
            "s3Object": "LandingZone/ICS/DeviceData/MR/Valid/70278/2017/05/14/8e355650-d8ab-439b-aa7d-94fc4f1b8d12/Simon.zip",
            "presignedUrl": "https://cf-s3-8d054666-f9f5-4495-b575-0f2b88c9c851.s3-eu-west-1.amazonaws.com/LandingZone/ICS/DeviceData/MR/Valid/70278/2017/05/14/8e355650-d8ab-439b-aa7d-94fc4f1b8d12
                /Simon.zip?AWSAccessKeyId=[redacted]&Expires=1512468632&Signature=fx9sR8Cg1BSyJSTaKEL2Sn1RE48%3D",
            "deviceMetadata": {
                "catalogNumber": "781341",
                "modality": "MR",
                "serialNumber": "70278",
                "systemModel": "T15",
                "systemType": "Ingenia",
                "systemuid": "70278",
                "equipmentNumber": "61375183",
                "softwareVersion": "5.3.0.?"
            },
            "s3ObjectMetadata": {
                "size": "48324146",
```

Figure 5.29: Output sample without pagination

”modality”: ”CV”,

”pageNumber”: ”1”,

”pageSize”: ”5”,

”serialNumber”: ””,

”softwareVersion”: ””,

”systemModel”: ””,

”systemType”: ””,

”workflowId”: ”B9FKLO90”

]

Output

[

”totalRecords”: 3,

”pageNumber”: ”1”,

”expectedPageSize”: ”5”,

”records”: [

[

”s3Object”:

”PublishZone/logviewer/modality=CV/systemId=1098/year=2017/month=11/date=23/

logfilename.gz”,
”presignedUrl”:
”https://S3bucketname.s3-eu-west-1.amazonaws.com/PublishZone/logviewer/modality%3DCV/
systemId%3D1098/year%3D2017/month%3D11/date%3D23/ logfilename.gz
?AWSAccessKeyId=1901120290290&Expires=1512019283&Signature=faGEDusUuF%3D”,
”deviceMetadata”:
[
”catalogNumber”: ”720909”,
”modality”: ”CV”,
”serialNumber”: ”123”,
”systemModel”: ”FD10 C”,
”systemType”: ”Allura XPer”,
”systemuid”: ”1098”,
”equipmentNumber”: ”1098”,
”softwareVersion”: ”8.2.1.9”
],
”s3ObjectMetadata”:
[
”size”: ”1400300”,
”lastModified”: ”2017-11-27 13:27:24”
],
”presignedUrlExpiration”: ”2017-11-30 10:51:23”
],
[
”s3Object”:
”PublishZone/logviewer/modality=CV/systemId=1098/year=2017/month=11/ date=22
/1162711808a60asujhdj.gz”,
”presignedUrl”:
”https://S3bucketname.s3-eu-west-1.amazonaws.com/PublishZone/
logviewer/modality%3DCV/systemId%3D1098/year%3D2017/month%3D11/date%3D22/
logfilename.gz?
AWSAccessKeyId=1901120290290&Expires=1512019283

&Signature=aFtDly4GeZBxk76%3D",
"deviceMetadata":
[
"catalogNumber": "720909",
"modality": "CV",
"serialNumber": "123",
"systemModel": "FD10 C",
"systemType": "Allura XPer",
"systemuid": "1098",
"equipmentNumber": "1098",
"softwareVersion": "8.1.17.2"
],
"s3ObjectMetadata":
[
"size": "1400300",
"lastModified": "2017-11-27 13:27:24"
],
"presignedUrlExpiration": "2017-11-30 10:51:23"
],
[
"s3Object":
"PublishZone/logviewer/modality=CV/systemId=59236/year=2017/month=11/
date=23/logfilename.gz",
"presignedUrl":
"https://S3buketname.s3-eu-west-1.amazonaws.com/PublishZone/logviewer/modality%3DCV
/systemId%3D59236/year%3D2017/month%3D11/date%3D23/
logfilename.gz?AWSAccessKeyId=1901120290290
&Expires=1512019283&Signature=ZAsxnsnkKXjkjkDsuv%2BXf%3D
"deviceMetadata":
[
"catalogNumber": "72298",
"modality": "CV",

"serialNumber": "165",

"systemModel": "FD20",

"systemType": "Allura XPer",

"systemuid": "59236",

"equipmentNumber": "59236",

"softwareVersion": "8.1.17.2"

],

"s3ObjectMetadata":

[

"size": "1400300",

"lastModified": "2017-11-27 13:27:24"

],

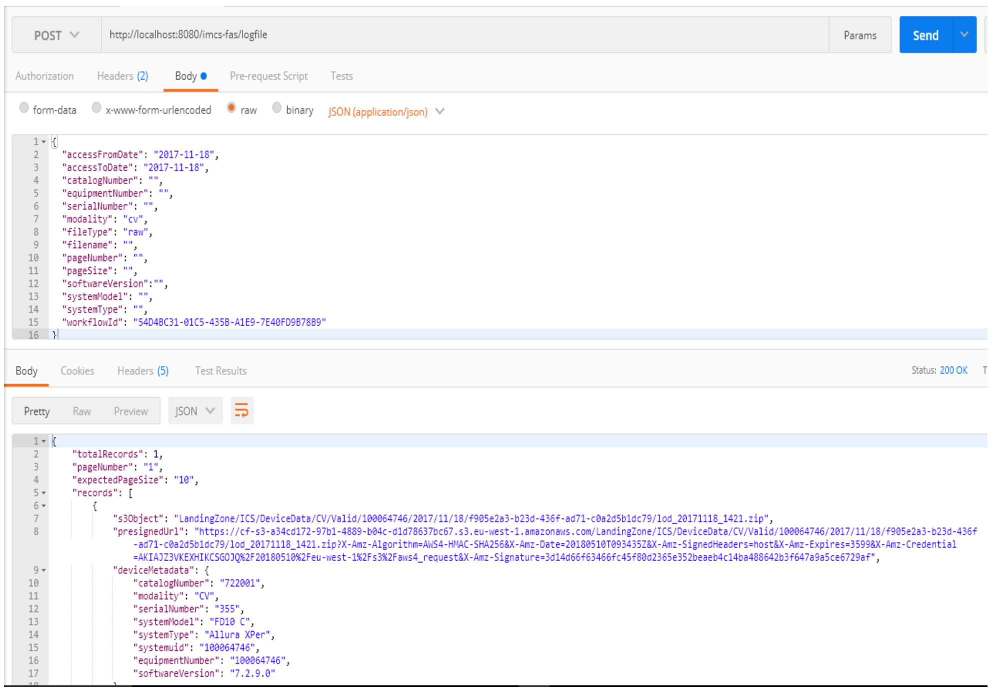
"presignedUrlExpiration": "2017-11-30 10:51:23"

]

],

"pageSize": 3

]



```
http://localhost:8080/imcs-fas/logfile
```

```
{
  "accessFromDate": "2017-11-18",
  "accessToDate": "2017-11-18",
  "catalogNumber": "",
  "equipmentNumber": "",
  "serialNumber": "",
  "modality": "cv",
  "fileType": "raw",
  "filename": "",
  "pageNumber": "",
  "pageSize": "",
  "softwareVersion":"",
  "systemModel": "",
  "systemType": "",
  "workflowId": "54D4BC31-01C5-435B-A1E9-7E40FD9B78B9"
}
```

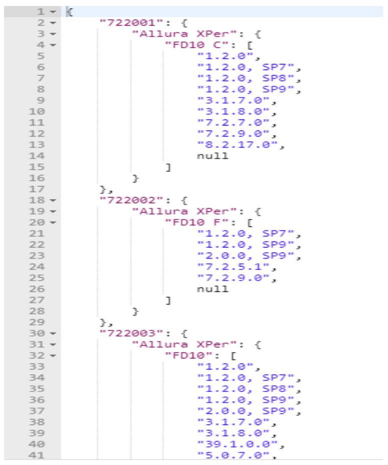
Status: 200 OK

```
{
    "totalRecords": 1,
    "pageNumber": "1",
    "expectedPageSize": "10",
    "records": [
        {
            "s3Object": "LandingZone/ICS/DeviceData/CV/Valid/100064746/2017/11/18/f905e2a3-b23d-436f-ad71-c0a2d5b1dc79/lod_20171118_1421.zip",
            "presignedUrl": "https://cf-s3-a34cd172-97b1-4889-b04c-d1d78637bc67.s3.eu-west-1.amazonaws.com/LandingZone/ICS/DeviceData/CV/Valid/100064746/2017/11/18/f905e2a3-b23d-436f
                -ad71-c0a2d5b1dc79/lod_20171118_1421.zip?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Date=20180510T093435Z&X-Amz-SignedHeaders=host&X-Amz-Expires=3599&X-Amz-Credential
                =AKIAJZ3VKEXHIKCSGOJQ%2F20180510%2Feu-west-1%2Fs3%2Faws4_request&X-Amz-Signature=3d14d66f63466fc45f80d2365e352beaeb4c14ba488642b3f647a9a5ce6729af",
            "deviceMetadata": {
                "catalogNumber": "722001",
                "modality": "CV",
                "serialNumber": "355",
                "systemModel": "FD10 C",
                "systemType": "Allura XPer",
                "systemuid": "100064746",
                "equipmentNumber": "100064746",
                "softwareVersion": "7.2.9.0"
```

Figure 5.30: Output for LOD file

5.4.4 System Metada API

Provides list of catalognumbers to systemtype, systemmodel and the versions mapping supported in asp. The api accepts modality as the request parameter in the query string.
Sample url : https://url/imcs-fas/systemmetadata?modality=cv
Figure 5.31 describes sample response for above api.



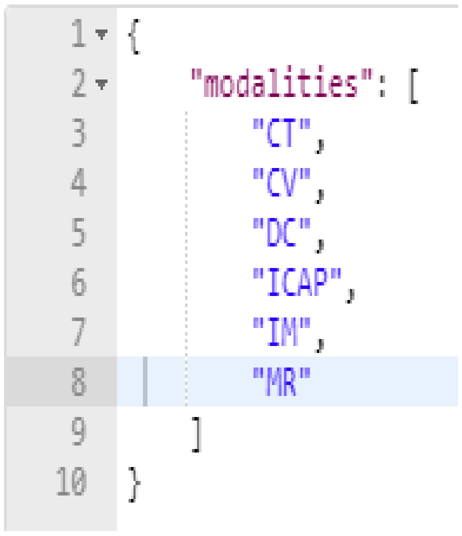
```
{
    "722001": {
        "Allura XPer": {
            "FD10 C": [
                "1.2.0",
                "1.2.0, SP7",
                "1.2.0, SP8",
                "1.2.0, SP9",
                "3.1.7.0",
                "3.1.8.0",
                "7.2.7.0",
                "7.2.9.0",
                "8.2.17.0",
                null
            ]
        }
    },
    "722002": {
        "Allura XPer": {
            "FD10 F": [
                "1.2.0, SP7",
                "1.2.0, SP9",
                "2.0.0, SP9",
                "7.2.5.1",
                "7.2.9.0",
                null
            ]
        }
    },
    "722003": {
        "Allura XPer": {
            "FD10": [
                "1.2.0",
                "1.2.0, SP7",
                "1.2.0, SP8",
                "1.2.0, SP9",
                "2.0.0, SP9",
                "3.1.7.0",
                "3.1.8.0",
                "39.1.0.0",
                "5.0.7.0",
```

Figure 5.31: Output for System Metadata

5.4.5 Modality API

Provides list of modalities supported in ASP Sample url : https://url/imcs-fas/modality

Figure 5.32 describes sample response for above api.



```
{
    "modalities": [
        "CT",
        "CV",
        "DC",
        "ICAP",
        "IM",
        "MR"
    ]
}
```

Figure 5.32: Output for Modality API

Chapter 6

Testing

Testing is one of the major component in software life cycle as it provides the accuracy and quality of any of the product and tool.After this step only product can be deliver to consumers.Based onto the test cases and specification verification team can identify the which are the passed and failed scenarios for that particular product.Tester also check that till which point tool can able to handle huge data while migrating to amazon S3.The product has to work as per consumer requirements with particular flow and respective input and output.These are the following things which should be satisfied during verification process.

- The product should be work as per consumers requirements.It should consider both positive and negative scenarios.
- It should pass all the required test cases which are specified in test cases.
- If any bug reported, Development team has to fixed that without breaking other features.
- Deliver product in time line after verification
- Verified for all possible test scenarios for better outcome

6.1 Continuous Integration

Continuous Integration is a high level practice of combining all the code that has been developed by all teams with a minimal shared repository. Its main purpose is to combine with automated units. With this concept it gave rise to new concept of runtime build servers, which leads to running unit test in a sequence of steps or after each commit. It is the processing of having repository with existing changes or new changes , we can even compare the code of previous existing ones . There will be commit doing by developer , when taking build it should not break or no errors has to be raised during build, if it comes then there is no chance in correcting them. The build can be stable and freezed . So that if any new check in was happened if it failed we can revert it to back. One of the popular open source automated build project is Jenkins repository. It follows some features and principles.

1. Code Repository: As multiple developers are working on one project everyone have to give code for review before check in.So next time anyone want to work on that same project they can able to get latest code of project.The code should be check in at proper location and no one can delete that as proper permission are given to users.
2. Build Automate:By using single line command projects can be build.That Automating the build consists of continuous integration with deployment into production.
3. Check the Build : After automating build check in Jenkins UI whether build is successful or not if not than check Jenkins console and make it successful.If yes than check whether latest changes are reflected or not and working fine as per developers have developed.
4. Each commit Each Build: Commit each and every code so that other user can work on latest build only.
5. Deployment Automation: Develop scripts which run after each of the successful build.

6.2 Different types of Testing

6.2.1 Sanity Testing

This is the very important type of testing which can be done after build gets successful.It can be carried out after some of few changes are done in build.Its aim is to check whether functionalities of product are working or not.If it fails then reject the build so that time and cost can be saved.

6.2.2 Unit Testing

Unit testing is the testing of each module by developers.Developers checked for each of the input whether product is able to produce valid output or not.It is related to particular one module testing.The unit testing should be done in such a way that it should not affect the result of others modules.

6.2.3 Smoke Testing

Smoke testing is performed after build is successful to ensure that all the features of product are working fine or not.It is performed before regression and functional testing.The main purpose of this testing is to test major and core functionalities of product and reject the product if something broke down so that installation software and manual testing can not be done.

6.2.4 Component Integration Testing

It tests if any of dependancy components are failed or not.It should work as per requirements and design specification of project.

6.2.5 Regression Testing

This testing checks whether fixing bug is breaking other thing of product or not. As sometimes it might be possible that by fixing some issues it will produce other bugs in project so tester should make sure that scenarios befor any of bug fixing also working properly.

6.2.6 Functional Testing

As mentioned in name it willl test for all the functionalities of product which are specified in software requiement specification and detailed design documents. It works on following things:

- Typo error of any input parameters
- It should accept only valid input
- It should validate input through different validation
- It should give proper output without any failure

6.2.7 System Testing

It tests for entire flow of product whether its working fine or not.It tests all the functionalities with required configuration to get the expected results.

6.2.8 Test Cases

Historical Raw Data Migration

Testcase id	Objective	Result
TC-001	Verify that appropriate error codes returned on trying to run ZipFilePreparer with invalid date and source location	Pass
TC-002	Verify that only valid systemuid(equipment number) which are present in json file should be converted to structured format using ZipFilePreparer job	Pass
TC-003	Verify that data are converted to proper format with proper path structure as S3 without losing any internal files	Pass
TC-004	Verify that the file states should be in Completed state after successfully completing ZipFilePrepare job	Pass
TC-005	Verify that the file states should be in Completed state after successfully completing ZipFilePrepare job	Pass
TC-006	Verify that structured data should be uploaded to S3 without losing any files at proper location using UploadUSModalityToS3	Pass
TC-007	Verify that file state should be changed from Completed to Text after successfully data are uploaded to S3	Pass
TC-008	Verify that only invalid systemuid(equipment number) which are not present in json file should be converted to structured format using QuarantineZipFilePreparer	Pass
TC-009	Verify that structured data should be uploaded to S3 without losing any files at proper location using QuarantineUploadUSModalityToS3	Pass
TC-010	Verify that file state should be changed from Completed to Text after QuarantineUploadUSModalityToS3	Pass
TC-011	Verify that after successfully completing Uploader job data should be deleted from local system	Pass

Table 6.1: Test Case for Historical Raw Data Migration

File Access Service

Testcase id	Objective	Result
TC-001	Verify that user shall be able to access with IAM token raw files in a date range for a system from S3	Pass
TC-002	Verify that user shall be able to access published files (drools output) in a date range for a system from S3	Pass
TC-003	Verify that appropriate error codes returned on trying to get presigned url details for raw or published files with madatory information missing as part of request	Pass
TC-004	Verify that user shall be able to access raw or published files for a system from S3 based on file type	Pass
TC-005	Verify that appropriate error codes returned on trying to get presigned url details for raw or published files with incorrect date value	Pass
TC-006	Verify that appropriate error codes returned on trying to get presigned url details for raw or published files with incorrect authentication token or with token details	Pass
TC-007	Verify that user shall be able to access raw and published files (drools output) with given workflowid for a system from S3	Pass

Table 6.2: Test Case for File Access Service

Chapter 7

Conclusion

The aim of Historical Raw Data Migration is to migrate the huge amount of data from local system to amazon S3 cloud as in health care organization data are generated day day like log files for different medical devices.The data are not in proper format that might be structured , semi-structured and unstructured so with help of this tool user can easily migrate big data without any other intervention.As security and lose of data is major concern nowadays,it handled using different states like In-progress,Completed and Text which ensure that files are migrated successfully.

The main focus of this project is to access log files of device data from amazon s3 browser.In addition,user can filter the particular log file as per requirement.As per that request file access service provide URL and remote user can easily download that requested log files.The service provides device metadata for more information about requested log files.The service is not able to generate records for more than 2000 records ,so pagination is also provided to accommodate large result-set.

The ETLs are used when one want to extract only useful data from large amount of data by using some mapping strategy.The extracted data can be transformed and export to Redshift or Postgres.At last the data are uploaded to amazon S3.The data are in comma separated format so it should be read using any of the parser in such way that it can extract useful data.

7.1 Future Scope

The future work of Historical Raw Data Migration project is to increase the security and privacy by including X-token and identity access management with front end.As its back end project front end can be develop so that by one click any authenticated user can migrate data.In addition to that as its migrating huge data time complexity is one of the major concern so that can be optimize.

The File Access Service is working for two file types i.e raw and published.In future that can be extended for other file types as per that log files store in amazon S3.In addition,user can able to access log files which are related to ASP and RADAR(workflowId) it can be extended for other remote users also.

Bibliography

[1] https://en.wikipedia.org/wiki/Philips

[2] D. G. Pradeepini and A. S. MAnekar, "Opportunity and Challenges for Migrating Big Data Analytics in Cloud," *IOP Conference Series: Materials Science and Engineering*, 2017.

[3] L. Wu, C;, L. Zhang, Z, Guo, C. C, L. M, and FCM, "Move big data to the cloud: an online cost-minimizing approach," *IEEE Journal on Aelected Areas in Communications*, vol. v. 31 n. 12, p. 2710-2721.

[4] M.Sravanthi, K.Preethi, M.Anusha, and D. Jangala, "Recent Issues and Challenges on Big Data in Cloud Computing," *IJCST*, vol. Vol. 6,Issue 2,Apring-June.

[5] C. Pahl, A. Ahmad, and P. Jamshidi, "Cloud Migration Research:A Systematic Review," *IEEE Transactions on Cloud Computing*, 2015.

[6] J.-F. Z. J.-T. Zhou, "Strategies and Methods for Cloud Migration," *International Journal of Automationa and Computing*, vol. 11(2),143-152 DOI: 10.1007/s11633-014-0776-7, 2014.

[7] S. Thakur and P. Pant, "Data Migration Across The Clouds," *International Journal of Soft Computing and Engineering(IJSCE)*, vol. ISSN: 2231-2307,Volume-3,Issue-2,May, 2013.

[8] U. SNDT, R. Vaidya, and P. S. Pundkar, "LArge Data migration within CLoud Environments using Compressiona and Encryption Technique," *International Journal of Innovative and Emerging Researchh in Engineering*, vol. Volume 2,Issue 2, 2015.

[9] M. Dhore and A. J.Mungole, “Techniques of Data Migration in Cloud Migration,” *IOSR Journal of Computer Engineering(IOSR-JCE)*, vol. e-ISSN: 2278-0661,p-ISSN: 2278-8727 – 36-38.

[10] A. Saxena and V. S. Kushwah, “A Security Approach for Data Migration in Cloud Migration,” *International Journal of Scientific and Research Publications*, vol. Volume 3,Issue 5, May 2013.

[11] D. N. Jawai, A. Abdelmabound, A. Elsafi, and I. Ghani, “A Comarative Evalution of Cloud Migration Optimization Approaches:S Systematic Literature Review,” *Journal of Theoretical and Applied Information Technology*, 2015.