




Algorithm Execution Client

Target release	Apollo v5.0
Story	<div><div> GSSPS-247 - Cloud Client - Analysis and Design IN PROGRESS</div><div> GSSPS-186 - Cloud Client - Implementation BACKLOG</div><div> GSSPS-197 - Cloud Client - Integration BACKLOG</div></div>
Document status	DRAFT
Document owner	Chandaluri, Siva N
Designer	N/A
Developers	Chandaluri, Siva N
QA	

1.1. Goals

- Algorithm Execution Client - Analysis, Design & Implementation

1.2. Background and strategic fit

- Algorithm Execution Client polls for incoming algorithm execution requests from AWS SQS and processes it. The client performs the following steps.
 - Reads the JSON payload from S3 object path and creates input payload file (input.csv)
 - Submits the R scripts for execution
 - Reads the output results (results.csv) and errors (errors.csv) and invokes the post processor/service api

1.3. Assumptions

- Execution client shall be load balanced to poll for the incoming requests and submit the jobs concurrently

1.4. Requirements

Following are the requirements for R algorithms to be executed against BDAA/Dx data source using Athena.

- Algorithm client shall create the input parameter (input.csv) file and make it available to R algorithm
- Athena and on-premise Reliability database connections would be written in a separate R script and the algorithms could source the connection script. R algorithms should not be hard-coding the connection strings/database credentials as part of algorithm logic/code.
- Algorithms shall loop thru all the algorithms defined using the group routine, process the data and output results for each algorithm.
- Algorithm shall output both flagged and healthy instrument data for each algorithm.
- Get available instruments in the last 24 hours, from all the logs that the algorithm is querying and identify healthy/flagged conditions for that data set.
- Algorithm execution results shall be written to results.csv file
- Algorithm execution errors shall be written to errors.csv file

Additional requirements/assumptions:

- Execution timeouts need to be defined for the algorithms?
- Additional libraries required for the Algorithm(s):
 - Checkpoint package install option shall be used to install the required packages, as needed

1.5. Analysis

Input file structure for R algorithms (input.csv):

Name	Type	Comments
ALGORITHM_NAME	VARCHAR2(200)	Algorithm Name, as entered in the algorithm definition

PHM_PATTERNS_SK	NUMBER	PHM_PATTERNS_SK from PHM_PATTERNS table
PARAMETER_NAME	VARCHAR2(500)	Algorithm Parameter names, as configured in the algorithm definition
PARAMETER_VALUE	VARCHAR2(4000)	Algorithm Parameter values, as configured in the algorithm definition

Note: Input file name available to the R algorithm would be input.csv

Output file structure (results.csv):

Name	Required?	Type	Comments
PHM_PATTERNS_SK	Yes	NUMBER	PHM_PATTERNS_SK (algorithm unique key), represents the algorithm
PL	Yes	VARCHAR2(5)	Product Line Code (Ex: 115, 116, 117, 205, 210, 214 etc..)
SN	Yes	VARCHAR2(30)	Instrument Serial Number
FLAG_DATE	Yes	VARCHAR2(20)	Instrument local Date/Time of when this instrument has been flagged. Value should be in YYYYMMDDHH24MISS format If instrument is marked as flagged (FLAG_YN =1) -> Flag Date - would be the max of instrument local date/time over a period of time that the algorithm ran against -> format: YYYYMMDDHH24MISS If instrument is marked as healthy (FLAG_YN=0) -> Flag Date -> would be the current day in YYYYMMDD format
CHART_DATA_VALUE	No	NUMBER(15,5)	Counter value. Required, If FLAG_YN is 1. Make sure to set DEVICE_VALUE to >= 1 if FLAG_YN = 1. Maps to DEVICE_VALUE in PHM_ALG_OUTPUT table Note: May not required to be populated by the algorithms. Deborah to confirm.
FLAG_YN	Yes	NUMBER(1)	Flag to identify if a instrument is healthy or unhealthy. Flagged -> 1; Healthy -> 0
IHN_LEVEL3_DESC	No	VARCHAR2(150)	1) Algorithm has single experience code assigned for a PL. -> If the results.csv file populate IHN_LEVEL3_DESC value then R processor takes it and persists as IHN_LEVEL3_DESC in PHM_ALG_OUTPUT table -> If the results.csv file does not populate IHN_LEVEL3_DESC then R processor picks up what is assigned to IHN_LEVEL3_DESC parameter in the algorithm definition and populates it in PHM_ALG_OUTPUT table 2) Algorithm has multiple experience codes assigned for a single PL -> In this case, results.csv should always populate IHN_LEVEL3_DESC and it should also match with LEVEL 3 description assigned in TCT. System will compare the TCT LEVEL 3 with IHN_LEVEL3_DESC column and pick up the right experience code. -> If the R output (results.csv) does not populate IHN_LEVEL3_DESC value then leave this field as blank in PHM_ALG_OUTPUT table.

Note: All of the above columns need to be present in the generated output results.csv file from the R algorithm

Error file structure (errors.csv):

Name	Required?	Type	Comments
PHM_PATTERNS_SK	Yes	NUMBER	PHM_PATTERNS_SK (algorithm unique key), represents the algorithm
ERROR_MESSAGE	Yes	VARCHAR2(4000)	Error message

Note: Execution errors shall be written to errors.csv file. All of the above columns need to be present in the generated errors.csv file from the R algorithm

1.6. Design

Include any mockups, diagrams or visual designs relating to these requirements.

1.7. Implementation

1.8. Verification/Testing

1.9. Questions

Below is a list of questions to be addressed as a result of this requirements document:

Question	Outcome

1.10. Not Doing