**Scope of Work Document**

**203\_Orange\_CF\_ELASTIC\_SEARCH\_INTEGRATION**

**Orange Egypt**

**Prepared By: Mahmoud Shehab**

ASSET Technology Group

Version 1.1

February 16, 2022

Copyright© 2021 Asset Technology Group, All rights reserved.

This document contains information proprietary to Asset Technology Group services, methodologies, programs and products is to be treated as confidential and a trade secret of Asset Technology Group and is not to be used or disclosed except to recipient’s employees, officers, agents or contractors engaged in evaluating this document, and who are subject to appropriate written undertakings consistent with these confidentially and use restrictions. Asset Technology Group retains the intellectual property rights to these trade secrets. This document is protected under copyright laws as unpublished work of Asset Technology Group.

**Table of Contents**

[1 Scope of Work & Solution Description 3](#_Toc95735657)

[1. Introduction 3](#_Toc95735658)

[2. In Scope 3](#_Toc95735659)

[3. Out of Scope 3](#_Toc95735660)

[4. Assumptions 3](#_Toc95735661)

[5. Deliverables 3](#_Toc95735662)

[6. Prerequisites 3](#_Toc95735663)

[2 Detailed Solution Description 4](#_Toc95735664)

[1. Database Changes: 4](#_Toc95735665)

[2. New logging service: 4](#_Toc95735666)

[3. CF Changes 5](#_Toc95735667)

[3 Reference 5](#_Toc95735668)

# Scope of Work & Solution Description

## Introduction

* New feature will be added for allowing CF to integrate with external systems to handle Real-time troubleshooting with live tail.

## In Scope

The following activities are considered:

* + Analysis, design & development of the software components mentioned in the solution description.
  + Implementing features mentioned below in the detailed solution description.

## Out of Scope

* + Any data migration is considered out of scope.

## Assumptions

* + ASSET will deliver the proposed solution’s war file on testing storages at Orange.
  + The development (coding) of CR will take place at ASSET premises.

## Deliverables

* + War file containing proposed solution.
  + New Jar file for logging service component.

## Prerequisites

* + Access between CF nodes and new logging service nodes.

# Detailed Solution Description

The new feature will be added in CF for allowing it to integrate with Elastic search system, the solution will provide a high available and scalable service that should be responsible for integrating with elastic search system, also it will provide a dynamic configuration for controlling the required info that should be sent to the Elastic search system for each service.

## Database Changes:

Five tables will be added to handle our logic.

1. LK\_LOGGING\_PARAMETERS (ID, NAME)
   1. Predefined table that contains all applicable parameters that CF can send to the external system.
2. SERVICE\_LOGGING\_MAPPING (SERVICE\_ID, LOGGING\_PARAMTER\_ID, SPECIAL\_FORMAT, IS\_STATIC, DEFAULT\_VALUE, ORDER\_ID)
   1. SERVICE\_ID represent the required service id
   2. LOGGING\_PARAMTER\_ID represent the lookup id from LK\_LOGGING\_PARAMETERS
   3. SPECIAL\_FORMAT represent the special format like “Any Date format”.
   4. IS\_STATIC represent if the parameter is static.
   5. DEFAULT\_VALUE represent the default value in case of no value exists.
   6. ORDER\_ID represent the ordering of the parameter in the request body.
3. LOGGING\_NODES (NODE\_ID, NODE\_URL, MAX\_RATE)
   1. NODE\_ID represent the node id
   2. NODE\_URL represent the node URL i.e. “http://10.53.13.69:8004/clog/charging”.
   3. MAX\_RATE represent the max rate of the node that we shouldn’t exceed it.
4. SERVICE\_LOGGING\_NODES (SERVICE\_ID, NODE\_ID)
   1. Used for mapping each service with each node.
   2. Support many to many relationships between services and nodes.
5. INTERNAL\_LOGGING\_NODES (NODE\_ID, NODE\_URL)
   1. Contains the new service nodes that make CF load balanced on it.

The below file contains sample data for each table:

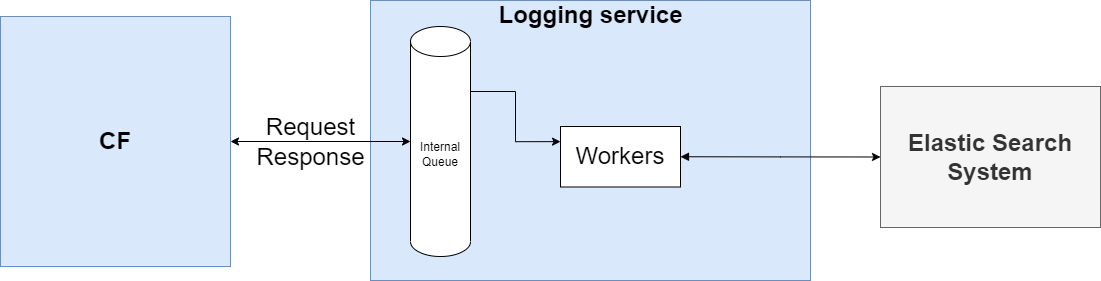


## New logging service:

This new component will be responsible for integrating with the external systems and send the real-time line logs with only http request with comma separated values body.

The new service will receive asynchronous requests from the CF, which means the CF will not wait until the request send to the external system, also the response will not affect to the current workflow for each service and shouldn’t impact the performance.

**Request workflow:**



After receiving request from CF the new service should push log line in an internal queue and return success message to the CF, internally there are many workers that should deque log lines and prepare each request based on the configuration of the service.

The below is a sample request that will be sent to Elastic search system:

* curl -POST 'http://10.53.13.69:8004/clog/charging '

-d '2000000001,2022-02-08 15:30:10.4567,test\_request,test\_response,200,30,HTTP 200 OK,1205557911,testID,null,null,null’

The technology that will be used in this service:

1. Java 8
2. Embedded tomcat.

## CF Changes

Make new changes in CF to integrate with the new logging service, CF will send the below post request internally to the new logging service:

* {

|  |
| --- |
| clientId=value, |
| clientName=value, |
| creationDate=value, |
| msisdn=value, |
| msisdnLastDigit=value, |
| otherError=value, |
| eCode=value, |
| moduleType=value, |
| serviceId=value, |
| packageId=value, |
| serviceClassId=value, |
| serverID=value, |
| cfTransactionId=value, |
| trackingId=value, |
| requestUrl=value, |
| serviceVersion=value, |
| totalRequestTime=value, |
| requestBody=value, |
| mainLoggingValue=value, |
| airStatus=value |

}

And the response should be as below:

{

|  |
| --- |
| “statusCode”=0, |
| “statusMessage”=success |

}

Note: CF will send requests to the new logging service in case of there was record in SERVICE\_LOGGING\_MAPPING table for this service.

# Reference

