# State of Ohio
## Improper Payments (Fraud) Analytics

Technical Design Document

Date: 06/25/2021

**Ohio** | **Department of Job and Family Services**
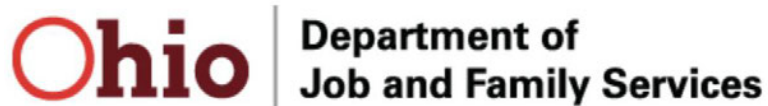
Authors: Sapan Bandi, Ward Loving

Prepared for: Ohio Department of Jobs and Family Services

*This document is intended to be a technical working document to help manage the delivery of a project and is provided for illustrative purposes only. The activities and goals serve as guidelines and additional detail and do not supersede any legal terms or conditions as defined in the customer's written contract with Google.*

PROFESSIONAL SERVICES

# Contents

# Background

The Ohio Department of Jobs and Family Services of Labor (JFS) is responsible for managing both Unemployment Assistance (UI) and Pandemic Unemployment Assistance (PUA) for the state. Over the last year the number of unemployment claims filed have dramatically increased for the state and unfortunately, a large number of the claims for assistance are fraudulent. PUA provides benefits to qualifying individuals who are otherwise able to work and available for work within the meaning of applicable state law, except that they are unemployed, partially unemployed, or unable or unavailable to work due to COVID-19 related reasons, as defined in the CARES Act and its extensions. Benefit payments under PUA are retroactive, for weeks of unemployment, partial employment, or inability to work due to COVID-19 reasons starting on or after January 27, 2020. The current system processes a total of 11.7 million records with 6.2 million requests for pandemic-related assistance.

JFS has partnered with Google and its implementation partner, SpringML, to analyze claimant data and develop a system to identify claimants based on suspicious patterns of data associated with a particular claimant or across the entire claimant population. The system uses a set of rules to identify conditions in the data that suggest invalid data and generates a score for each condition based on its severity. The scoring system is scalable and has been developed on the Google Cloud Platform. Once data has been loaded into the system, the improper payment solution can generate scores for JFS claimants in a matter of hours. Processed claimant data and scores are consumable through dashboards and file exports.

Google Cloud Platform offers flexibility and performance at scale for public-facing applications. Integrating Google Cloud infrastructure across an organization extends those benefits to all areas of an organization.. Coordination across teams and projects within a government entity is key to building the most successful platform. Provisioning user accounts, enforcing access control, auditing network security, and configuring chargebacks can all be handled gracefully if all the relevant teams are involved from the very start of the project. This document describes architecture and results of the larger improper payments project.

# High Level Architecture
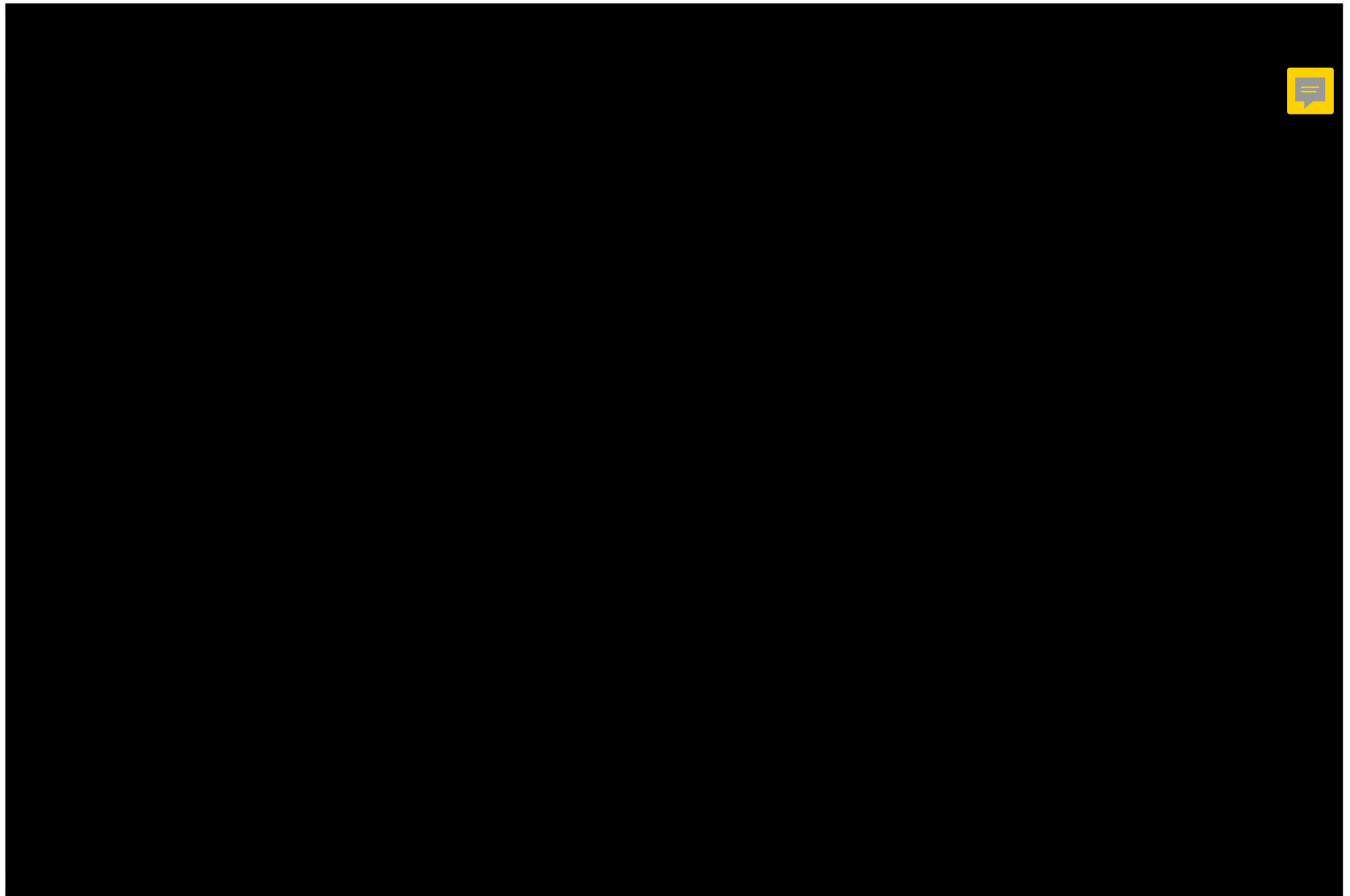


Legends

Phase 1 (Current State)

# Google Cloud Components

The Fraud Detection solution uses the following Google Cloud Platform Components:

| Component | Description |
| --- | --- |
| Google Cloud Storage | File Repository - Data Input. Audit extracts are placed in GCS on a weekly basis. |
| Google Compute Engine | Virtual compute resources are used to process fuzzy similarity logic for usernames and emails |
| Google Cloud DataFlow | Data Pipeline tool used to interface with IP Address evaluation services. |
| Google BigQuery | Processed scoring data is stored in date partitioned tables in BigQuery. SQL queries perform segmentation of claimant records. |

| Google Data Studio | Reports and visualizations for segmentation of claimants. |

# Flags and Thresholds

| Name | flag name | threshold | weight |
|------|-----------|-----------|--------|
| ████████ | ███████ | ▮ | ■ |
| █████████ | ███████ | ▮ | ■ |
| ███████████ | ████████ | ██████ | ■ |
| █████████ | ██████ | ▮ | ■ |
| ███████ | ████ | ▮ | ■ |
| █████ | ██████ | ▮ | ■ |
| ██████████████ | ██████ | ▮ | ■ |
| ██████ | ████ | ▮ | ■ |
| █████████ | ██████ | ▮ | ■ |
| █████████ | ██████ | ▮ | ■ |
| ███████ | ████ | ▮ | ■ |
| █████████ | ███████ | ▮ | ■ |
| █████████████ | ███████ | ▮ | ■ |
| █████████████ | ███████ | ▮ | ■ |
| ███████████████ | ██████ | ▮ | ▮ |
| █████████ | ███████ | ▮ | ■ |
| █████████ | ██████████████ | ▮ | ■ |
| ██████████ | ████ | ████ | ■ |
| ██████████ | ████ | ████ | ■ |
| █████████████ ███ | ███████ | ▮ | ■ |
| █████ | ██████ | ▮ | ■ |
| ██████ | ███████ | ▮ | ■ |
| ██████████ | ███████ | ▮ | ■ |
| ██████████ | ██████ | ▮ | ■ |

| | | | |
|---|---|---|---|
| ███████████████ | ██████████ | ▪ | ▪ |
| ███████████████ | ████████ | ▪ | ▪ |
| ███████████████ | ███████████ | ▪ | ▪ |
| ███████████████ | ██████████ | ▪ | ▪ |
| ███████████████ | ████████████ | ▪ | ▪ |
| ███████████████ | ██████████ | ▪ | ▪ |
| ████████████ | ██████████ | ▪ | ▪ |
| ███████████ | ████████ | ▪ | ▪ |
| ██████ | ███████ | ▪ | ▪ |
| ██████ | ███████ | ▪ | ▪ |
| ███████████ | ███████ | ▪ | ▪ |
| ███████████████ | ██████████ | ▪ | ▪ |
| █████████ | ██████ | ████████ | ▪ |

## Solution Design

This Fraud Detection system uses a rules-based approach to identify fraudulent records. It seeks to mimic the process that an Unemployment Fraud Auditor would use to evaluate a claimant by looking for issues. It should be noted that this system does not have access to all the information an auditor can access. However, it does provide the ability to compare a particular claimant record to the rest of the claimant population across multiple dimensions. These comparisons are extremely helpful for the auditing staff in revealing patterns of misbehavior by bad actors in the system.

Only a small proportion of claimant records have no flags associated with their data. The records are on a continuum -- some records only have a few issues versus others that have many issues. These flags (or rules) are weighted in accordance with their severity and a score is generated for each rule and then aggregated for each claimant record. This application tracks the range of flags that are associated with each claimant and provides feedback to the auditors reviewing each record. Generally, the more flags a claimant has, the higher score they will receive. High scores are strongly correlated with fraud.

## Data Sources and Storage

████████████████████████████████████████

████████████████████████████████████████████████████

██████████████████████████████████
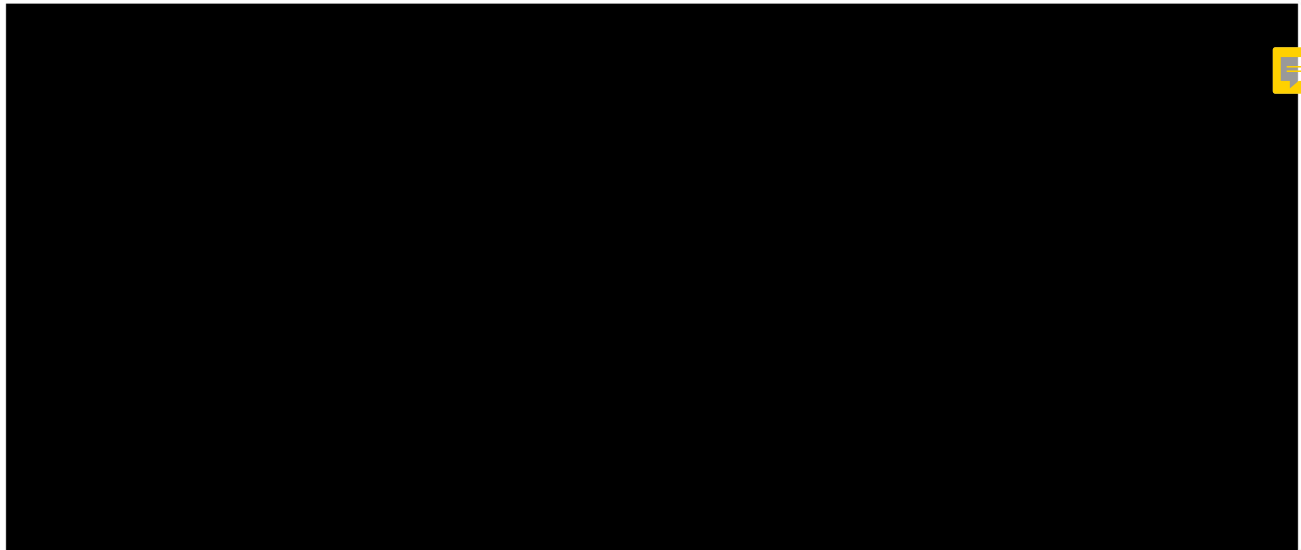
The two folders used primarily are incremental (for incremental data) and full-refresh (for full data)

Within incremental and full-refresh data folders, we receive files in a date folder where the date corresponds to the date of upload.

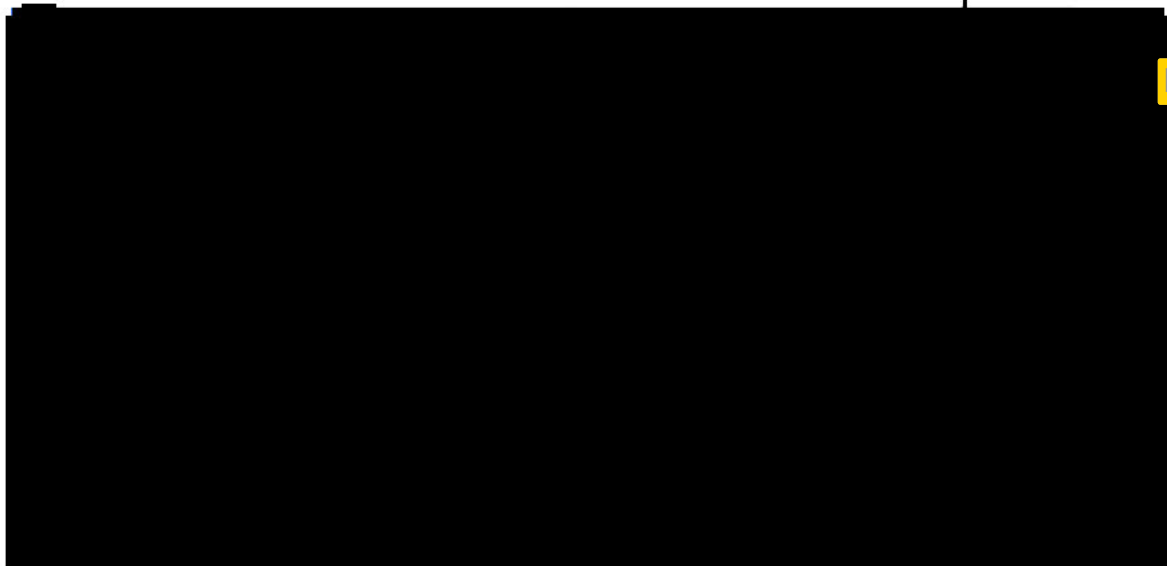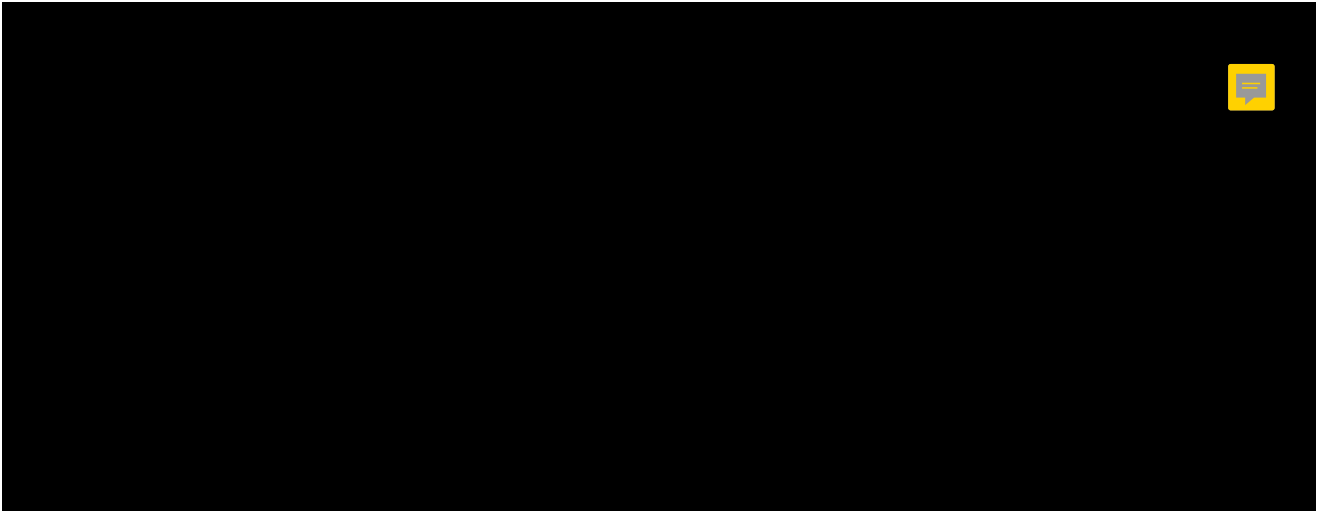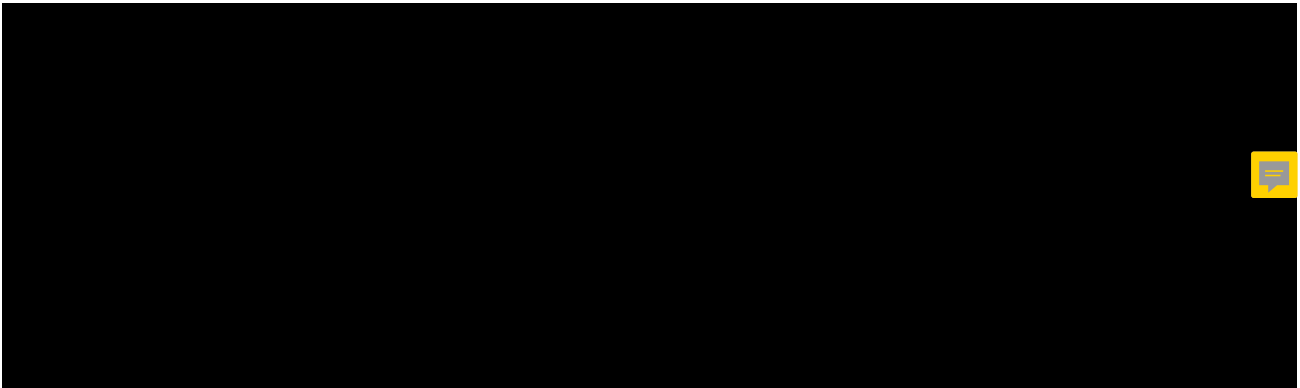## Data Ingestion and Loading

To ingest data from raw files and load into BigQuery, we have developed python scripts.

Log into Data Flow Notebooks and Open Jupyter Labs

**Incremental load:**

This python script uses following 2 configuration files:

- ████████████████████████████████████████████████████████
  █████████████████████████████████████████████
- ████████████████████████████████████████████████████████
  ██████████████████████████████████████████████████████
  █████████████████

Steps performed in Incremental:
- Download file from GCS bucket
- Unzip file
- Pre-process file applying some conversion and validation rules
- Load the file to BigQuery in incremental or temp tables in the "latest_data" dataset

- Dynamically generate merge statements and execute queries to update/insert the main the table joining with latest_data tables using the schema configuration mapping and primary keys mapping files as described above

We load the new data into the "latest_data" dataset as temporary tables. The merge queries then work on combining/merging the data in these temporary tables and the main tables in other datasets. Results are stored in main tables stored in one of the datasets

- ███████████████
- ███████████████████
- █████████████

**Full refresh load**:

Full refresh load is similar to Incremental load except that we do not have the last merge step. All the other 4 steps are the same in execution and inputs for these scripts is common form.

Steps performed in Full refresh:

███████████████████
█████████
█████████████
███████████████████████████████

These scripts take the date folder name as argument to the script when being called in runtime.

Screenshot for full refresh script execution:

Screenshot for incremental load:

# Data Processing Workflow

Queries to be executed manually after loading & merging process are as follows:

- Individual Claimant Combined: Idea is to bring things down to a claimant level for both OJI and UFACTS individually
  - OJI:
    - ███████████████████████ - table created & replaced
  - UFACTS:
    - ████████████████████████ - table created & replaced
- Individual Historical Analysis:
  - ███
    - ████████████████████
    - █████████████████████
    - ████████████████████
    - ███████████████
  - ██ ████████
    - ███████████████████████████
    - ████████████████████████████
    - ██████████████████████
    - ██████████████████

    - ████████
    - █████████████████████████
    - ████████████████████
    - ██████████████████████████████████
    - ███████████████████
  - ████████████████████████████
    - ███████████
    - ███████

# Data Processing and Flag Score Generation

██████████████████████████████████████████████

- █████████████████████
- ████████████████████████████████████████
- ███████████████████████████████████████
- ██████████████████████████

## Fuzzy Logic Processing

Fuzzy logic script can be found here:█████████████████

Custom metadata

Data location          US

15
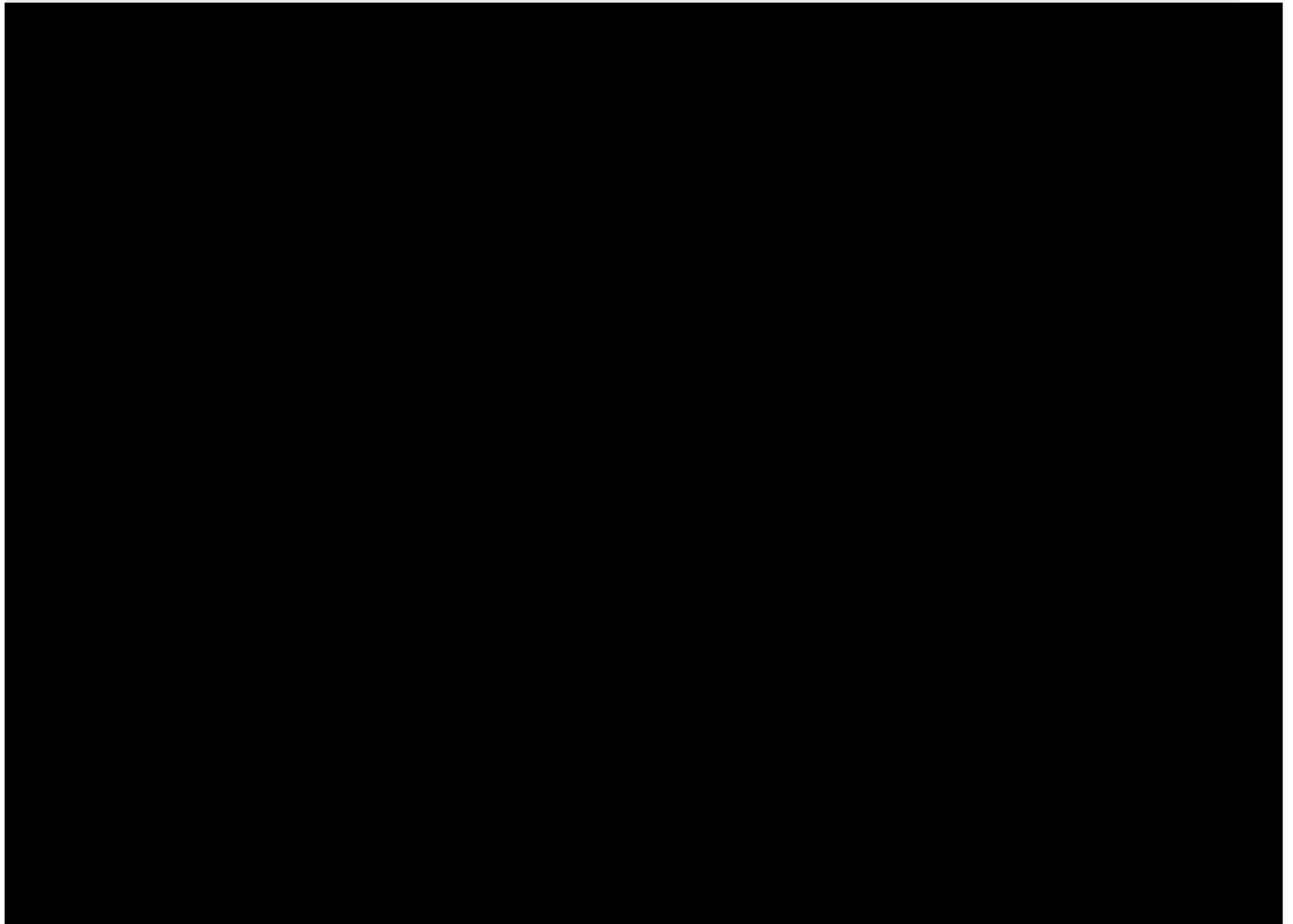
Flag generator or score computation

Step1:
Ensure all the preprocessing steps are complete as mentioned in ▓▓▓▓▓▓

Existing Data Sets

The following DataSets are managed and used in the Big Query with their usage descriptions

- ████████████████████████████████████
- ███████████████████████████████
- ███████████████████████████
- █████████████████████████████
- ████████████████████████████████
- █████████████████████████████████████
  ███████
- ███████████████████████████

## Data Studio Dashboards

███████████████████████████████████████████████

██████████████████████████████
█████████████████

| ██████ | ██████ | ██████ |
|--------|--------|--------|
| ██████ | ██████ | ████████████████████████████████████████<br>████████████████████████████<br>████████████████████████████<br>███████████████████████████<br>█████████████████████<br>██████████████ |
| █████ | █████████<br>████████ | █████████████████████████████████<br>██████████████████████████████<br>████████████████████████████<br>██████████████████████████<br>███████████████████████████<br>████████████████████████████████ |
| █████ | ██████████████ | ████████████████████████████████<br>███████████████████████████████<br>██████████████████████████████<br>██████████████████████. |
| ██████ | ███████████████ | █████████████████████████████████████ |

|  |  | ████████████████████████████████ |
|---|---|---|
|  | ████ | ████████████████████████████████ |
|  |  | ████████████████████████████████ |
|  |  | ████████████████ |
| ████ | ████████████ | ████████████████████████████████ |
|  |  | ████████████ |
| ████ | ██████████ | ████████████████████████████ |
|  | ██████████ | ████████████████████████████ |
|  |  | ████████████████████ |
| ████ | ██████████ | ████████████████████████████ |
|  | ████████████ | ████████████████████████████ |
|  | ██ | ████████████████████ |
| ████ | ████████████ | ████████████████████████████ |
|  |  | ████████████████████████████ |
|  |  | ████████████████████ |
| ████ | ██████████ | ████████████████████† ████████████ |
| ████████ | ████████ | ████████████████████████████ |
|  |  | ██████████ |

Summary

**22**
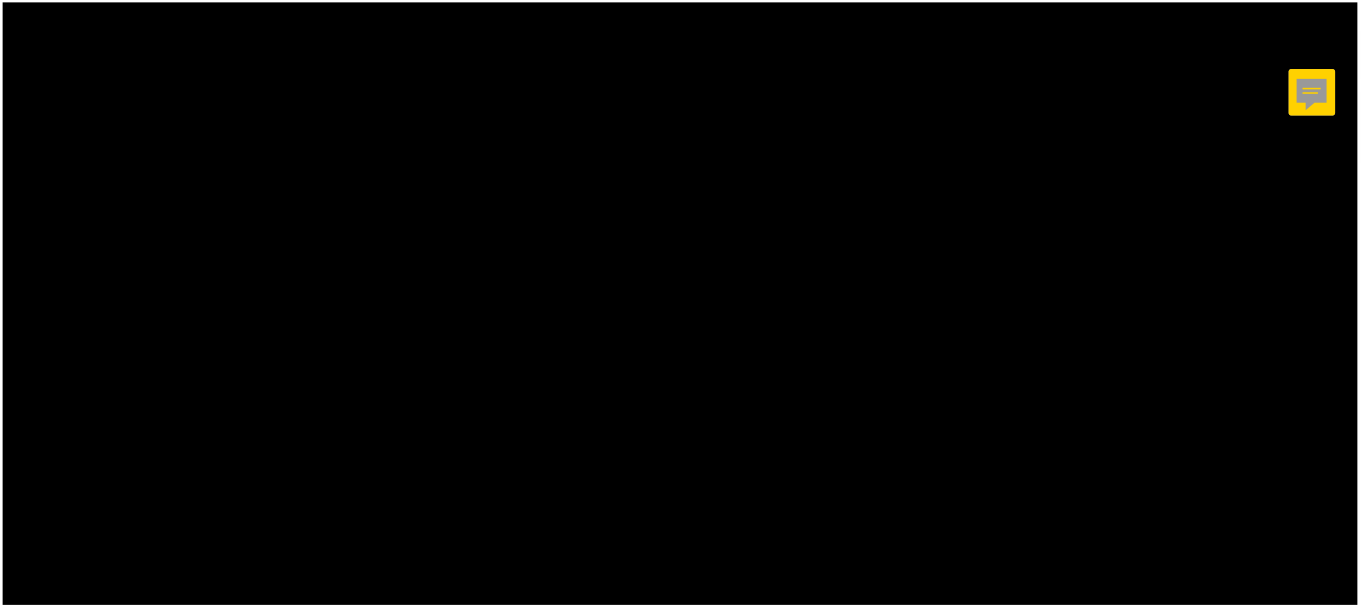
Below the fold of the summary view and many of the other pages are Claimant Level Details presented in table form.

# Fraud Analysis Dashboard Filters

The JFS Fraud Analysis dashboard contains a number of filters as outlined below:

## Program

Allows users to select claimants by program: OJI (Standard UI) or UFacts (PUA).

## Flag Bucket

The flag buckets are based on groupings of scores. The current groupings are:

Low: Score ████
Medium: Score ██████████████
High: ████████

## Flag Name

A drop down list of all the flags that are currently in place in the system.  By selecting an item you'll only return claimants that have been assigned that flag.

## Select Date Range

On all the pages except page 2 (Summary by AppDate) and page 7 (Total UI Claims Outstanding by AppDate), this control filters the records displayed by Last Payment Date for a particular claimant.  However, on page 2 and page 7, this control filters claimants by most recent claim Application Date instead.

## Total Payments

This controls filters records by Claimant Total Payments.  To see all claimants with payments simply enter "0" into the value line.  The greater than symbol (">") is provided by default but there are other options here as well for deeper analysis.

██████████

This control filters the results by Claimant Id allowing a user to retrieve information about a single claimant.
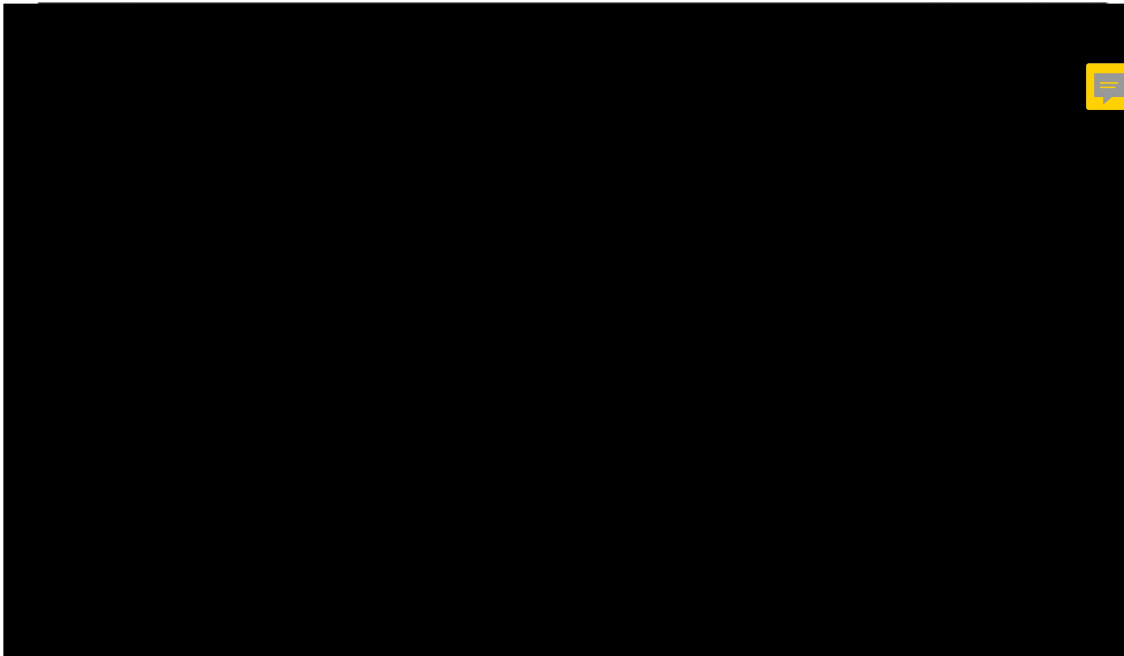
## Managing Data Sources

Data Studio Data Connections to the source BigQuery tables are managed through the "Manage added data sources"

For example, here are the data sources that are currently available to support Page 1 of JFS Fraud Analysis Dashboard:

If new fields have been added to the underlying BigQuery table (or fields have been removed or changed), clicking "RECONNECT" on the previous screen will adjust the data connection to pick up the adjustments.

## Linked Fields

Several of the dashboard pages are connected by hyperlinks. These hyperlinks are designated in the Dimension list of the Edit mode by the link icon.

By clicking the "link" part of the Dimension listing you can view the source field that is associated with this particular field label. In this case it is
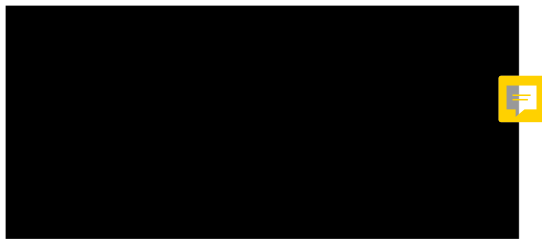
By navigating to ▮▮▮▮▮▮ in the available field list and clicking the link icon next to the field name, we can drill down to the underlying HYPERLINK function:

Clicking on this brings up the Formula Field interface:

The HYPERLINK function takes 2 parameters. The first is the target url. The second is the label of link. The CONCAT functions are used to string the parameters together.

Here is a complete formula link for navigating from the Summary View to the Claimant View using ███████ (in the main table data):

████████████████████████████████████████████████████████████████
██████████████████████████████████████████████████████████████████████
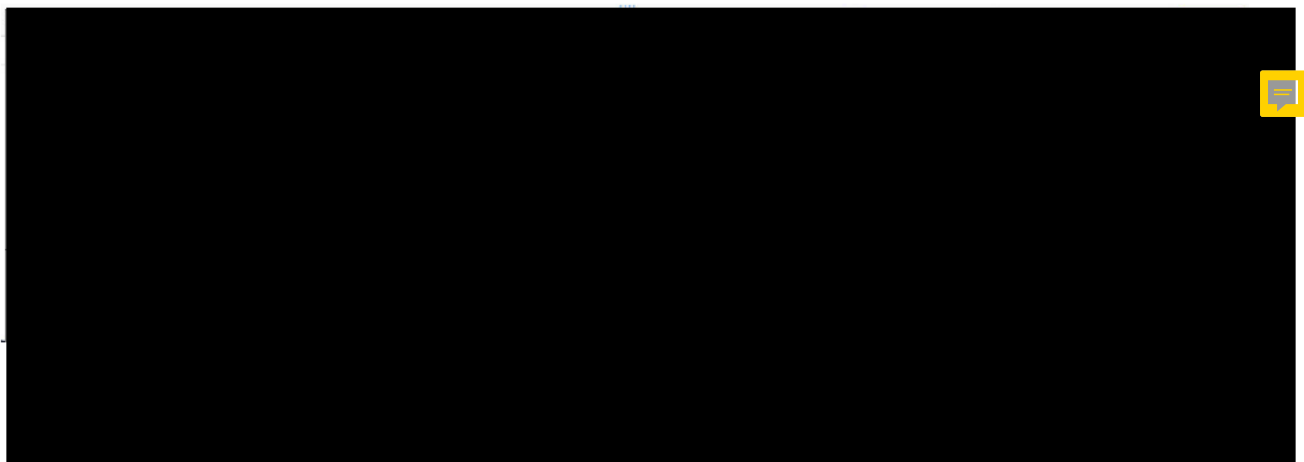██████████████████████████████████████████

██████████████████████████████████████████████████████████████████
████████████████████████████████████████

████████████████████████████████████████████
██████████████████████████████████████████████████████████████████████
████████████████████████████████

Once you have this target URL simply replace the dynamic parameter (████████████████ with variables and the CONCAT function.

# Infrastructure Configuration and Setup

## Project, Permission & network, Security, Authentication

████████████████████████████
████████████████████████████

## User Roles and Permissions

Created custom Role -> dev-springml-role will have following specific permissions

| | |
|---|---|
| ████████████ | |
| ████████████████ | |
| ████████████ | |
| ████████████ | |
| ████████████████ | |
| ████████████ | |

| Group/role Name | Function | Roles/Permissions |
|---|---|---|
| ████████████████████ | Editor Group for the Business users | JFS_ClaimsAnalytics_custom_editor_role<br>Storage Object Viewer |
| ██████████████████████v | Admin group to assist in resource procurement and configuration during Google/SpringML Engagement | Owner |
| ██████████████████████ | Developer Group to ingest and process data during Google/SpringML Engagement | dev_springml_custom_role<br>Editor<br>Service Account User<br>IAP-secured Tunnel User<br>Source Repository Reader<br>Source Repository Writer |
| ████████████████ | Custom Role for SpringML developers | BigQuery Admin<br>Storage Admin<br>Cloud Functions Admin<br>Logging Admin |

| | | DataFlow Admin<br>Instance Admin |
|---|---|---|
| ██████████████████████████████ | Business and workflow users with specific resource permissions | JFS_ClaimsAnalytics_custom_workflow_user<br>Service Account User<br>IAP-secured Tunnel User |
| ████████████████████████ | Admin users group with specific resource permissions | JFS_ClaimsAnalytics_custom_admin<br>Service Account User<br>IAP-secured Tunnel User |

# End Business and Technical Roles and Permissions Specs and configuration

This is proposed roles combination to be setup and configured by creating custom roles and assigning that custom roles to groups based on the GCP functional role or ability to perform necessary operations within project and executing or running the scripts

| Service/Feature Name | Admin Roles(custom) | User Workflow Roles(custom) |
|---|---|---|
| BigQuery | roles/bigquery.admin | ████████████<br>███████████████<br>████████████████████<br>██████████████ |
| Storage | roles/storage.admin | ██████████████<br>█████████████<br>████████████<br>██████████████ |
| Compute | roles/compute.admin | ████████████████████<br>██████████████████ |
| Service Account user | roles/iam.serviceAccountUser | ████████████████ |
| Dataflow | roles/dataflow.admin | ██████████████<br>████████████ |
| Cloud functions | roles/cloudfunctions.admin | ████████████████████<br>██████████████████ |
| Logging | roles/logging.admin | ██████████████<br>████████████████ |
| Stackdriver | roles/stackdriver.admin | ██████████████████████ |

| IAP for VM ssh | roles/iap.tunnelResourceAccessor | ███████████████████████ |
| Monitoring | roles/monitoring.admin | ███████████████████████ ███████████████ |
| Notebook | roles/notebooks.admin | ██████████████ ███████████████ |
| Cloud Source Repository | roles/source.admin | ███████████ ██████████ |
| Secret Manager | roles/secretmanager.admin | █████████████████ ███████████████████ |

## Enabled Project Level API

- ██████████████
- ████████████████████████
- ████████████████████████
- ███████████████
- █████████████
- █████████████
- ██████████████████
- █████████████
- ████████████

██████████████
█████████████████
█████████████
█████████████
████████████
██████████████
███████████████
██████████
██████████████

## Networking and security

████████████████████████████
- ████████████████████████████████████████
████████████████████████████
- ████████████████
- ███████████████████████
- ███████████████████████
- ██████████████████

- ██████████████████████████

| VM Name | Zone | Machine_TYPE | Internal_IP |
|---|---|---|---|
| ██████████████ | ██████ | ████████████ | ████████ |
| ██████████████████ | █████ | ██████████ | ████████ |
| ████████████████ | ██████ | ████████ | ████████ |
| ████████████████████ | ██████ | ███████████████ | ███████ |
|  |  |  |  |

## Service Account Setup and configuration

| Service Account | Purpose | Project | IAM Roles |
|---|---|---|---|

| | | | |
|---|---|---|---|
| ███████████████████ ██████████ | ████ ████ ██████ | ██████████████ | ████████████ ████████████ ██████████████ ██████████ █████████ █████████████ |
| ████████████████████████ | ██████████ ██████████ █████ | ████████████ | ████████ |
| ███████████████████████ | █████ ██████ ██████████ ████ | ████████████ ██ | ████████ |
| ████████████████████ | ████ ██████████ | █████████ ██ | ████████████████ ████████ |
| █████████████████████ | ████ ██████████ | █████████ ██ | ████████ |

## VI. Cloud Repositories

Setup and configure cloud repositories within GCP environment on the test project with following naming convention

## VII. VPC service control setup and configuration

- Service perimeter
- Org Policy Changes to support cloud functions,dataflow
- Org view permission
- List of users who will be using the systems
    - Users, groups
    - Service account
    - List of Public IP range for the access to the environment
- DNS and static configuration for GCP API communication
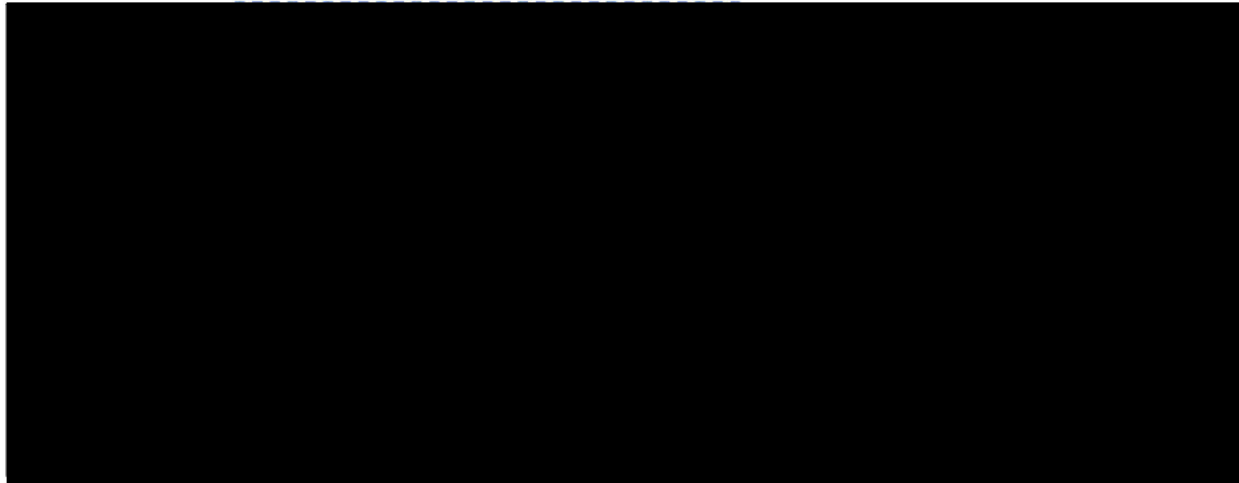    - Details to follow

# VIII. Authentication

- Will use ███████████ to authenticate for GCP project authentication and authorization

## VPC service control

VPC Service Controls provides these benefits by enabling you to define security policies that prevent access to Google-managed services outside of a trusted perimeter, blocking access to data from untrusted locations and mitigating data exfiltration risks.

With this release of VPC Service Controls, you are able to:

- ████████████████████████████████
- ██████████████████████████████████████
- ███████████████████████████

███████████████████████████████████████████████
███████████████████████████████████████████████
███████████████████████████████████████████████
███████████████████████████████████████████████
███████████████████████████████████████████████
███████████████████████████████████████████████

For TWC following will be configured, provision and setup for the VPC service controls

1) VPC network with 1 subnets in us-central1 region
2) Define and create a service perimeter that allows free communication within the perimeter but, by default, blocks all communication across the perimeter.
   a) If you want to use the gcloud command-line tool or the Access Context Manager APIs to create your service perimeters, create an access policy. Note: You do not have to manually create an access policy if you are using the Cloud Console to manage VPC Service Controls. An access policy will be created for your Organization automatically.
   b) Define and create default basic level of access policy for user members, IP networks from the
3) Private Google Access enabled for VM to access the Google API/service via private IP address space having no public external IP address.
4) Setting up the Private connectivity to Google APIs and services by defining the static route and configuring the DNS zone.

5) Grant access from outside a service perimeter using access levels. Access Level configuration and setup with user member, public IP address range and device attributes
6) All Service accounts will be allowed to access resources and services from outside to the service perimeter boundary.

## Roles/Permissions

The following curated IAM roles provide the necessary permissions to view or configure service perimeters and access levels using the gcloud command-line tool:

- ███████████████████████████████████
- ██████████████████████████████
- ███████████████████████████████

Additionally, to let your users manage VPC Service Controls using the Google Cloud Console, the following role is required

- ████████████████████████████████████

## VPC service Perimeter configuration scripts

████████

████████████

███████████████████

██████████████

██████████████████

███████████████

███████

████████████████████████████████

████████████

█████████

█████████

███████████████

█

███████████████████████████████████

  ████████████████████████████████

  ███████████████████

  ████████

█

████████████████████████████████████████████

   ████████████████████████████████████████████

    ████████████████████████

    ████████████

███████████████████

████████████████████████████████████████

█████████████

███████ ████████████████████████

████████████████████████████████████████████████████████████████

████████

  ████████████████████████████████████████████████████████████

  ███████████████████████████████████████████████

  ███████████████████████████████████████████

  ████████████████████████████████████████

  ████████████████████████████████████████

  ██████████████████████████

██████████████████

██████████████████

█████████████████

██████████████████

█████████████████

████████████████████████████████████████████████████

#███████████████████████

████████████████████████████████████████████████████████████████████████
████████

██████████████████████████████

███████████████████████████████████████

 ████████████████████████████████████

 █████████████████████████████████████████

 ████████████████

 ██████████████

████████████████████████

███████████████████████████████████████████

 ████████████████████████████████████████

 █████████████

Hello all:

Attached below is our weekly status report. Good progress this week on data loading and analysis as we prepare for our initial review next week.

Regards,
--
Ward Loving | 214.450.4391
Technical Project Manager | SpringML



Blog: Highlighting SpringML's COVID-19 Solutions

**CAUTION:** This is an external email and may not be safe. If the email looks suspicious, please do not click links or open attachments and forward the email to csc@ohio.gov or click the Phish Alert Button if available.

| From: | Ward Loving |
|---|---|
| To: | Wilson, Amy; Allen, Audrey; Lauterbach, Eric; Turner, Jason; Skinner, John; Furmanova, Vladislava; Brent Evans; Marshall Jung; Mike Lynn; Asim Sheriff; Jermyn Davis |
| Subject: | Weekly Status Report - 4.19.2021 |
| Date: | Monday, April 26, 2021 7:27:42 PM |

Hello all:

Attached below is our weekly status report. Thanks for your attendance at our initial analysis and dashboard review today. We'll be continuing to add flags and deepen our analysis over the week or so and will have another review session scheduled for next week.

Regards,

--
Ward Loving | 214.450.4391
Technical Project Manager | SpringML



Blog: The Second Mile : How to strategically conquer the vaccine distribution challenge

**CAUTION:** This is an external email and may not be safe. If the email looks suspicious, please do not click links or open attachments and forward the email to csc@ohio.gov or click the Phish Alert Button if available.

| From: | Ward Loving |
|---|---|
| To: | Wilson, Amy; Allen, Audrey; Lauterbach, Eric; Turner, Jason; Skinner, John; Furmanova, Vladislava; Brent Evans; Marshall Jung; Mike Lynn; Asim Sheriff; Jermyn Davis |
| Subject: | Weekly Status Report - 4.26.2021 |
| Date: | Monday, May 3, 2021 6:27:10 PM |

Hello all:

Attached below is our weekly status report. The SpringML team is heads-down preparing for our review session scheduled later this week. I thought we had constructive discussions today as well about how to incorporate Google's scoring into the existing resolution process. We're making progress!

Thanks,
--
Ward Loving | 214.450.4391
Technical Project Manager | SpringML



Blog: The Second Mile : How to strategically conquer the vaccine distribution challenge

**CAUTION:** This is an external email and may not be safe. If the email looks suspicious, please do not click links or open attachments and forward the email to csc@ohio.gov or click the Phish Alert Button if available.

| From: | Ward Loving |
|---|---|
| To: | Wilson, Amy; Allen, Audrey; Lauterbach, Eric; Turner, Jason; Skinner, John; Furmanova, Vladislava; Brent Evans; Marshall Jung; Mike Lynn; Asim Sheriff; Jermyn Davis |
| Subject: | Weekly Status Report - 5.3.2021 |
| Date: | Saturday, May 8, 2021 12:24:43 AM |

Hello all:

Attached below is our weekly status report. Thanks for everyone's time and attention at our Sprint 1 Review today. Another busy week coming up with additional flags and integration of Google/SpringML scoring with the current backlog buckets.

Thanks,

--
Ward Loving | 214.450.4391
Technical Project Manager | SpringML



 Blog: The Second Mile : How to strategically conquer the vaccine distribution challenge

| | |
|---|---|
| **From:** | Ward Loving |
| **To:** | Wilson, Amy; Allen, Audrey; eric.lauterbach@jfs.ohio.gov; Turner, Jason; Skinner, John; Furmanova, Vladislava; Brent Evans; Marshall Jung; Mike Lynn; Asim Sheriff; Jermyn Davis |
| **Subject:** | Re: Weekly Status Report - 5.10.2021 |
| **Date:** | Wednesday, May 19, 2021 4:48:22 PM |

Let's try one more time with the status report attached...8-)

On Wed, May 19, 2021 at 2:46 PM Ward Loving <ward.loving@springml.com> wrote:

Hello all:

Attached below is our weekly status report. Thanks for everyone's attendance at our Sprint 2 Review and Score Comparison session yesterday. We're moving into documentation and handover mode as we wrap our initial scope of work.

Thanks,

--
Ward Loving | 214.450.4391
Technical Project Manager | SpringML



Blog: Solving the Master Data Management challenge

--
Ward Loving | 214-450-4391
Technical Program Manager | springML

| From: | Ward Loving |
|---|---|
| To: | Furmanova, Vladislava |
| Cc: | Wilson, Amy; Allen, Audrey; Turner, Jason; Skinner, John; Brent Evans; Marshall Jung; Mike Lynn; Asim Sheriff; Jermyn Davis |
| Subject: | Re: Weekly Status Report - 5.3.2021 - Need clarification, impact date and statement |
| Date: | Monday, May 10, 2021 11:53:08 AM |
| Attachments: | image003.jpg |
| | image004.jpg |

Hi Vlada:

I've updated the status slide for last week to include the following risk/impact statement:

> The risk here is that the quality of our analysis will be impaired if we don't get access to the data by 5/17. The SpringML team needs a little time to ingest data and analyze to provide feedback before the end of the engagement.

On Mon, May 10, 2021 at 8:28 AM Vlada.Furmanova@jfs.ohio.gov <Vlada.Furmanova@jfs.ohio.gov> wrote:

> Ward,
>
> I do think that the risk/issue is accurate as described in the attached. I have provided information below. Can you please update the risk/issue to reflect OJI v. PUA and please add an impact statement, i.e., what the impact will and how severe it will be if this data is not received by a certain date.
>
> - Google/SpringML made four data requests:
>   1. OJI
>      a. Passwords - Legacy system; the passwords are not able to be exported – cannot provide.
>      b. Security question/answer – Can provide. Data is set is too large to format without timing out. Met with Google/SpringML on Friday, 5/7 to clarify need and determine how to optimally provide this large set of data. Decision at the 5/7 meeting was that JFS would send sample of 1k records. No 'need by' date was provided.
>   2. uFacts
>      a. Passwords – Can provide. In process. Deloitte had a problem with the extraction which caused a delay. A fix was implemented and a new extraction was initiated.
>      b. Security question/answer (Does not exist; system does not request this data from claimants) – cannot provide.
> - This data is not stored in the same location as all of the other data previously provided.
> - Since this data is very sensitive from a security standpoint, it took some time to receive approval (given that the same request was previously denied to the U.S. DOL).
> - Once approved, the teams (separate teams for OJI and uFacts) have worked to export and transfer the data to the Google cloud. Given that the data sets are very large, both teams have faced challenges in providing the data.

**44**

- Google/SpringML will receive the data as quickly as the uFacts and OJI teams are able to successfully export it from the source and transfer it to the Google cloud. Ultimately, the data that is available to be sent (OJI = security answers, uFacts = passwords), will be provided.

If Google/SpringML views this request as a risk/issue, please provide the impact date and an impact statement so that we can understand the impact.

Thanks,

Vlada

---

**From:** Ward Loving <ward.loving@springml.com>
**Sent:** Saturday, May 8, 2021 12:24 AM
**To:** Wilson, Amy <Amy.Wilson@jfs.ohio.gov>; Allen, Audrey <Audrey.Allen@jfs.ohio.gov>; Lauterbach, Eric <Eric.Lauterbach@jfs.ohio.gov>; Turner, Jason <Jason.Turner@jfs.ohio.gov>; Skinner, John <John.Skinner@jfs.ohio.gov>; Furmanova, Vladislava <Vlada.Furmanova@jfs.ohio.gov>; Brent Evans <brentevans@google.com>; Marshall Jung <marshalljung@google.com>; Mike Lynn <mjlynn@google.com>; Asim Sheriff <asim.sheriff@springml.com>; Jermyn Davis <jermyn@google.com>
**Subject:** Weekly Status Report - 5.3.2021

Hello all:

Attached below is our weekly status report. Thanks for everyone's time and attention at our Sprint 1 Review today. Another busy week coming up with additional flags and integration of Google/SpringML scoring with the current backlog buckets.

Thanks,

--

Ward Loving | 214.450.4391
Technical Project Manager | SpringML

Image removed by sender.

Blog: The Second Mile : How to strategically conquer the vaccine distribution challenge

**CAUTION:** This is an external email and may not be safe. If the email looks suspicious,

--
Ward Loving | 214-450-4391
Technical Program Manager | springML

| **From:** | Ward Loving |
|---|---|
| **To:** | Wilson, Amy; Allen, Audrey; eric.lauterbach@jfs.ohio.gov; Turner, Jason; Skinner, John; Furmanova, Vladislava; Brent Evans; Marshall Jung; Mike Lynn; Asim Sheriff; Jermyn Davis |
| **Subject:** | Weekly Status Report - 6.5.2021 |
| **Date:** | Wednesday, June 16, 2021 11:26:36 AM |

Hello al:

We're putting our final touches on the dashboards, wrapping up our technical documentation, and doing our handover sessions this week as this phase of the project wraps us. Attached is our status report for the week.

--
Ward Loving |
Technical Project Manager | SpringML



 Blog: Solving the Master Data Management challenge

| **From:** | Ward Loving |
| **To:** | Wilson, Amy; Allen, Audrey; eric.lauterbach@jfs.ohio.gov; Turner, Jason; Skinner, John; Furmanova, Vladislava; Brent Evans; Marshall Jung; Mike Lynn; Asim Sheriff; Jermyn Davis |
| **Subject:** | Weekly Status Report - 6.13.2021 |
| **Date:** | Tuesday, June 22, 2021 1:34:50 PM |

Hello all:

We're wrapping up our deliverables for the current phase. JFS has successfully loaded data into the Google/SpringML solution and executed the scoring scripts. I'll be sending out a link to our finalized Technical Design Document shortly. Thanks to Amy and Carl for the feedback last week -- we found several issues which we were able to resolve.

This week we're knocking down the last issue with the Application Date filter and helping to load some scoring files so that the correct JFS scores can be included in the Google solution.

Thanks,

--
Ward Loving | 214.450.4391
Technical Project Manager | SpringML



 Blog: The Race for Vaccination: How SpringML Helps Accelerate State Vaccine Rollouts

| From: | ward.loving |
| --- | --- |
| To: | Skinner, John |
| Cc: | Wilson, Amy; Furmanova, Vladislava; Jermyn Davis; Asim Sheriff; Brent Evans |
| Subject: | Re: comparison of data |
| Date: | Thursday, May 13, 2021 11:38:58 PM |
| Attachments: | Fraud System Factors - 5.13.2021.xlsx |

Hi John:

I've completed the grid as you've requested. I've gone ahead and entered the flags that will be in the system by Tuesday.

On Thu, May 13, 2021 at 5:01 PM Ward Loving <ward.loving@springml.com> wrote:

Hi John,

Thanks for the detail about the current OH algorithm. I'll fill out the flag comparison worksheet. We do track the number of flags we assign in our scoring table so that information is available for analysis. Amy mentioned that you will be sending some additional scoring our way for the complete claimant population. I do think it will be interesting to see how the scores/flags line up.

Thanks,

On Thu, May 13, 2021 at 1:12 PM John.Skinner@jfs.ohio.gov <John.Skinner@jfs.ohio.gov> wrote:

Ward,

I will send you data we have for the fields shown below, can you do an analysis/comparison of how we align on fraud scores by claimant between your solution and our data?

We would also like to update the spreadsheet to see what factors are covered in each solution.

Score is near the bottom. The data we have provided you already includes only scores for claimants with pending issues. This export will include every claimant with a fraud score > 1.0.

We want to look for the opportunities to have the items that your solution has as definitions leveraged in our processes and explore the differences in weighting and its impact on overall score.
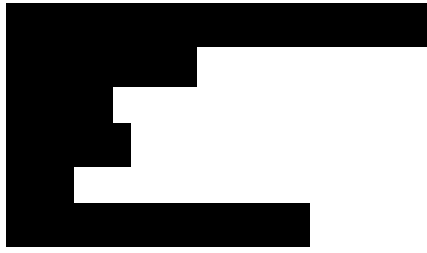
Do you have a non-weighted score that would help with the comparison?

**49**

--
Ward Loving | 214-450-4391
Technical Program Manager | springML

**CAUTION:** This is an external email and may not be safe. If the email looks suspicious, please do not click links or open attachments and forward the email to csc@ohio.gov or click the Phish Alert Button if available.

| Factor/Feature | Google |
|---|---|
| **Current Duplicate Checks** | |
| Duplicate Mailing Address Count | |
| Duplicate Physical Address Count | |
| Duplicate Phone Number Count | |
| Duplicate Bank Account Count | |
| Duplicate Email Exact Count | |
| Duplicate Fuzzy Email Count | |
| Duplicate IP Address Count | |
| Duplicate DMV Count | |
| **Historical Duplicate Checks:** | |
| Duplicate Historical Mailing Address Count | |
| Duplicate Historical Physical Address Count | |
| Duplicate Historical Phone Number Count | |
| Duplicate Historical Bank Account Count | |
| Duplicate Historical Email Exact Count | |
| Duplicate Historical Fuzzy Email Count | |
| Duplicate Historical IP Address Count | |
| Duplicate Historical DMV Count | |
| **Historical Data Change Counts:** | |
| Historical Count of Name Changes | |
| Historical Count of Email Changes | |
| Historical Count of DMV Changes | |
| Historical Count of Mailing Address Changes | |
| Historical Count of Physical Address Changes | |
| Historical Count of Phone Changes | |
| Historical Count of Bank Changes | |
| **Additional Fraud Checks:** | |
| Count of Claimants Changing Age from Over70/Under16 to Under70/Over16 | |
| Age over 70 | |
| Under 16 | |
| Matched to BPC Suspicious IP Addresse Watch List | |
| Matched to BPC Suspicious Email Domain Watch List | |
| Matched to BPC Suspicious Bank Account Watch List | |
| Claimants with SSN in both PUA and OJI | |
| Claimants with SSN in OJI with BPC Denied Issue (PUA CHECK ONLY) | |
| Claimants with Mismatching Mailing and Physical Address | |
| **Known Good Values (excludes factor from Fraud Dashboard for the claimant)** | |
| IP Address | |
| email address | |
| Phone number | |
| Address | |
| **External responses factored into Fraud Score** | |
| Ohio Department of Health | |
| Ohio Department of Medicaid | |
| Ohio Department of Corrections | |
| NASWA Integrity Data Hub | |

| | |
|---|---|
| State of Ohio Employee Data | |
| **Visualizations** | |
| Fraud Dashboard | |
| Claimant History | |
| DRC cross match detail | |
| ODH cross match detail | |
| OAKS cross match detail | |
| ODM cross match detail | |
| OUIO at a Glance | |
| Operations data - pendng issues detail - PUA | |
| Operations data - Supervisor queue - PUA | |
| **In process for scoring** | |
| Multi-factor analysis fraud score weighting | |
| External responses factored into Fraud Score | |
| **Pending** | |

BMV Data Match - need data source

Other features under discussion

| State |
|-------|
|  |
| X |
| X |
| X |
| X |
| X |
| X |
| X |
| X |
| X |
|  |
| X |
| X |
| X |
| X |
| X |
| X |
| X |
|  |
| X |
| X |
| X |
| X |
| X |
| X |
| X |
|  |
| X |
| X |
| X |
| X |
| X |
| X |
| X |
| X |
| X |
|  |
| X |
| X |
| X |
| X |
|  |
| X |
| X |
| X |
| X |

| |
|---|
| x |
| |
| x |
| x |
| x |
| x |
| x |
| x |
| x |
| in process |
| in process |
| |
| o |
| o |
| |

o

o

Hello all:

Attaching our weekly status report for the week. Various dashboard updates are in the works as we load the latest and greatest (incremental) data and develop our merge process.

Thanks,

--
Ward Loving | 214.450.4391
Technical Project Manager | SpringML

Blog: Solving the Master Data Management challenge

Hello all:

Attaching our weekly status report for the week. All the anticipated flags have been implemented in the system at this point. We're making final updates to our dashboards and documentation for our handover sessions this week.

Thanks,

--
Ward Loving | 214.450.4391
Technical Project Manager | SpringML

Blog: Solving the Master Data Management challenge

**CAUTION:** This is an external email and may not be safe. If the email looks suspicious, please do not click links or open attachments and forward the email to csc@ohio.gov or click the Phish Alert Button if available.