***EQUIFAX***

CMS Fusion

**INPUT PROCESS**

**DOCUMENTATION**

Table of Contents

[1. Input Tab 3](#_Toc508376828)

[1.1 View Input Process Status 3](#_Toc508376829)

[2. Import New file 4](#_Toc508376830)

[2.1 Name the process 4](#_Toc508376831)

[2.2 Specify the Input File 5](#_Toc508376832)

[2.3 Name the Output Table 7](#_Toc508376833)

[2.4 Select a Layout 7](#_Toc508376834)

[2.4.1 Layout Definition 7](#_Toc508376835)

[2.4.2 Import Layout 9](#_Toc508376836)

[2.4.3 Layout Search 11](#_Toc508376837)

[3. Run Data check 12](#_Toc508376838)

[4. Summary 13](#_Toc508376839)

**Revision History**

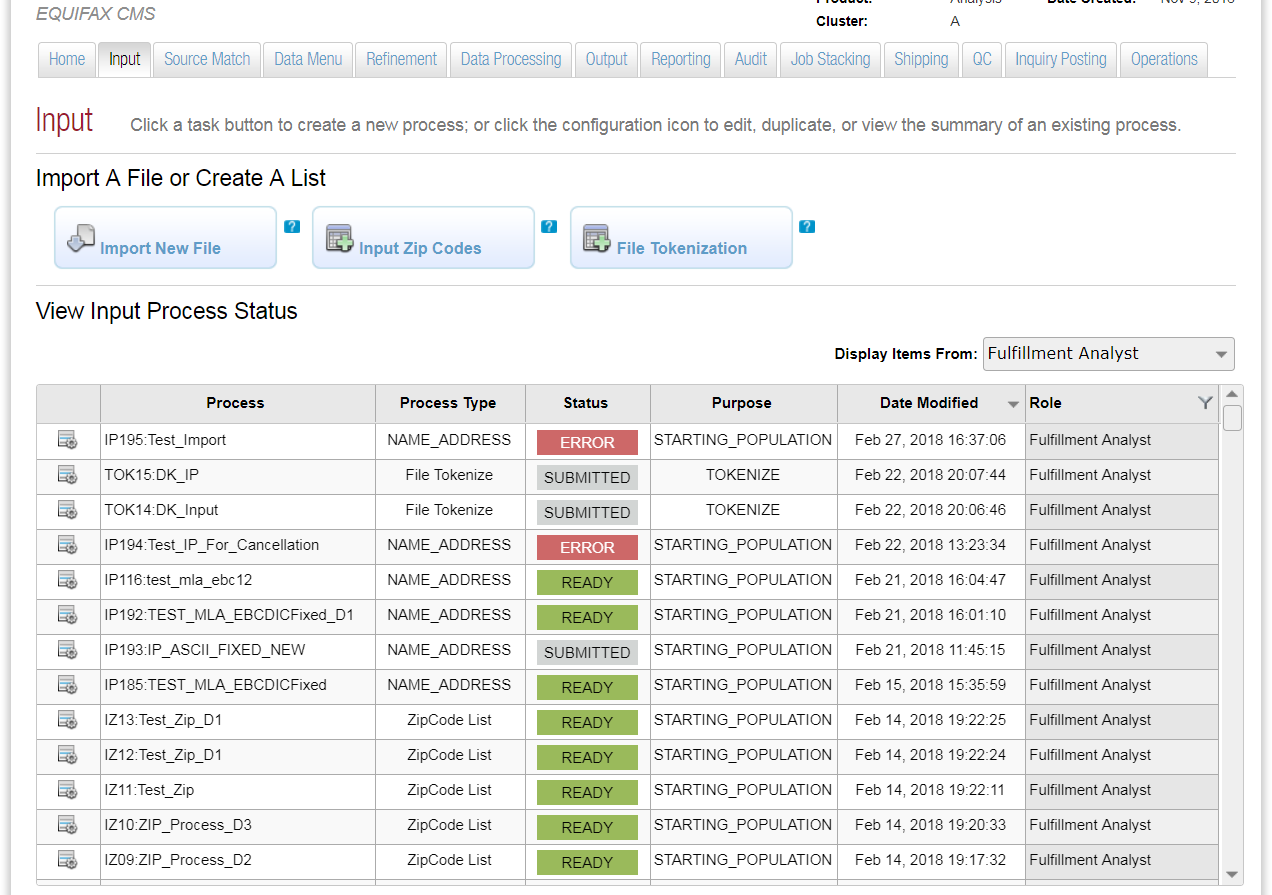
| **Revision** | **Author** | **Date** | **Status and Description** |
| --- | --- | --- | --- |
| 1.0 | UST Global | 11/01/2017 | Initial Draft |
| 1.1 | Equifax | 11/15/2017 | Review |
| 2.0 | UST Global | 11/17/2017 | Review Comments Incorporated |

**Updates**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Date** | **Story** | **Requirement** | **Section** | **Modified by** |
| 26/02/2018 | Sprint58(CF2-2351) | Handling EOL character while file import | 2.4.1 Layout Definition | Varnika Priydarshini |
| 28/02/2018 | Sprint58(CF2-1116) | Validation of MLA Transaction Id | [2.4.2 Import Layout](#MLATransactionID) | Varnika Priydarshini |
| 6/03/2018 | 1. Sprint59([CF2-2574](https://jira.equifax.com/browse/CF2-2574)) | Support for selecting a File Tokenization process | [2.2 Specify the input file](#Tokenization)  [4. Summary](#summary) | Varnika Priydarshini |
| 9/03/2018 | Sprint60(CF2-2375) | Source File Size Exceeds Fusions Capacities | [2.2 Specify the Input file Record length](#RecordSize) | Varnika Priydarshini |
| 13/03/2018 | Sprint60(CF2-3229) | Record Length Validation of Tokenized File | [2.2 Specify the Input file Validation](#validation) | Varnika Priydarshini |
|  |  |  |  | Varnika Priydarshini |

# 1. Input Tab

Customers will be providing the details of the files for getting processed within the fusion for getting the credit details. These files will be added in a ‘nas’ location, which is a server space where these files are stored. These files will be imported to Fusion app and processing is done to get data from Data Pump. Import Zips is to create a list of individual zips to Data Pump job.



## 1.1 View Input Process Status

While clicking on the input tab, it will list the set of Input processes created for the project and its details

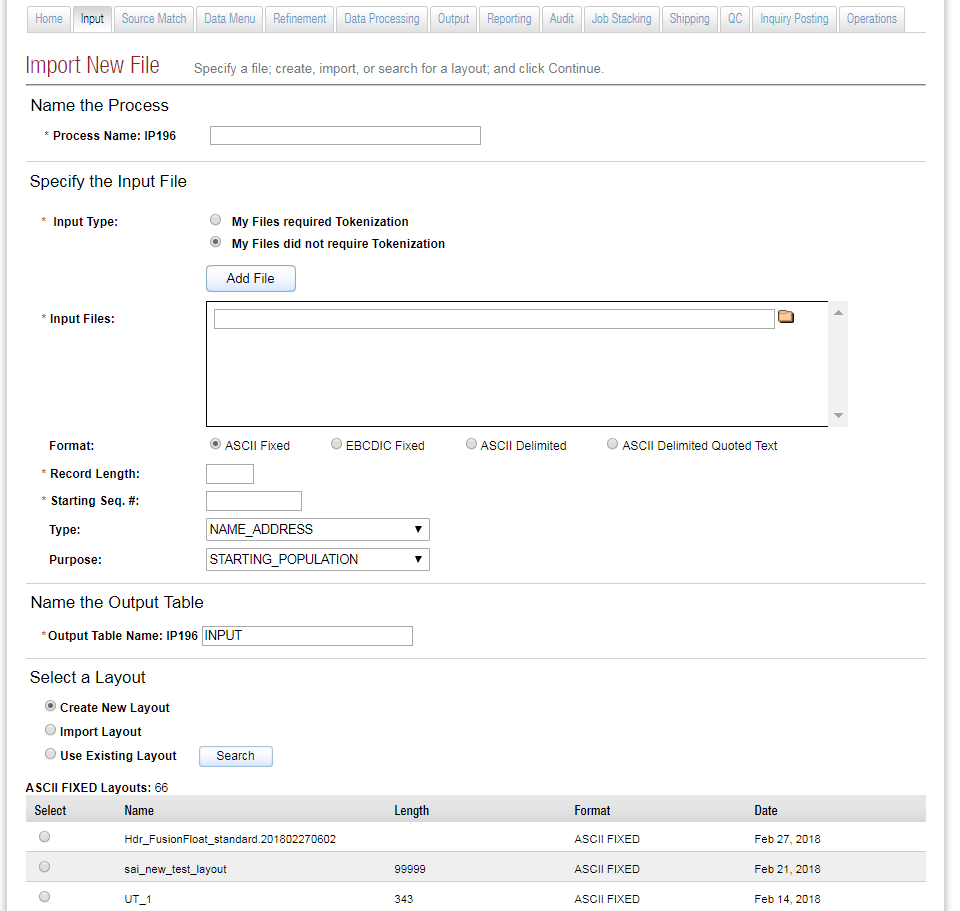
* Process – The Process name used for the process.
* Process Type – Type of process.
* Status – The status of the process created.
* Role – Represents different roles in the application. – Fulfilment Analyst, Programmers.

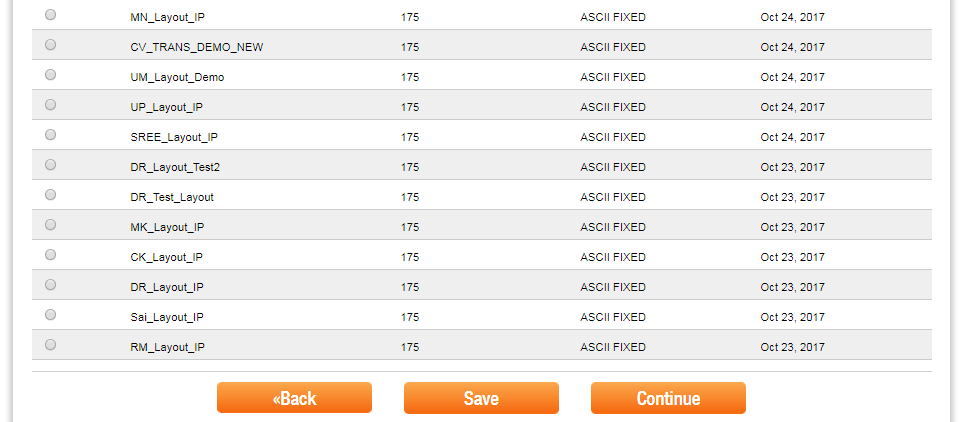
These will be the role of users who created the process.

* Display Item from - Filters the list of input files based on the role, who created the Input process.
* Edit – If need to edit any option provided in the process before submitting.
* Duplicate – Provides an option to duplicate an existing process.
* View Summary – There is a high level view of what all options are selected for the process.

# 2. Import New file

This Wizard helps to add the options to import a file.





## 2.1 Name the process

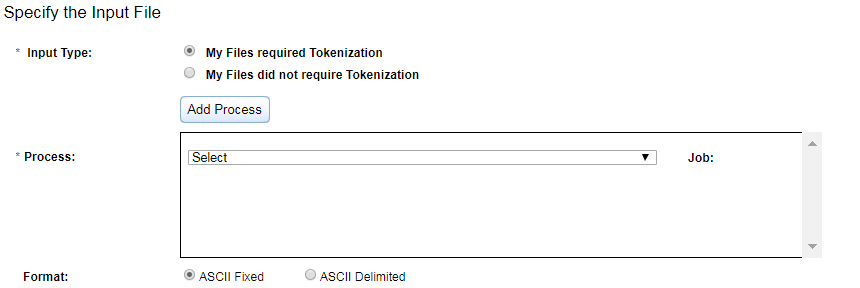
**Process Name:** A process name has a Process Id which is an Auto Generated Id. This will prefix the Process Name that is provided. In the screenshot above Process Id is IP33 and Process Name is TRANS\_DEMO\_IP. So the process Name would be IP33: TRANS\_DEMO\_IP.

## 2.2 Specify the Input File

**Input Type:** File Import process provides a choice to use files that had to be tokenized or those that did not. User can tokenize the input files imported by selecting My Files need tokenization button.

**My Files required Tokenization:** The process being loaded is from the last successful execution of the tokenization process.

In case of using tokenization process inputs and the file from the last run instance does not exist, an error message is displayed as: "Error: The output file from the last run instance of process <tokenization process name> no longer exists. Please make another selection or rerun <tokenization process name>.

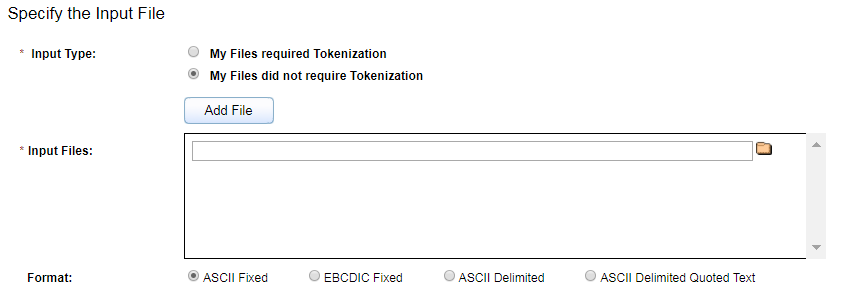


**Validation:**

* Tokenization is applicable only for ASCII Format.
* Tokenization process cannot be not be duplicated, otherwise is shows an alert **“Please select unique process”**.
* At least one Input process must be selected to continue.

**My files did not require Tokenization:**

**Input Files:** Import the input file from the ‘nas’ location, which is a server location. We can import multiple files of same format as input for a process. Files can be viewed using a file viewer tool, which will chunk the large files so that each chunk can be viewed separately.



**Sample Input file:** */nas/fusion/workspace/qa/R3A4762/OP43\_DM97\_SM\_FIELDS\_STD\_12374/OP43\_DM97\_SM\_FIELDS\_STD/I26288\_OT\_COMBINEFILE/ALL\_REC.fixed*

**Format:** This represents the format of the file that is provided as the input. The file formats can be ASCII, ASCII Delimited, EBCDIC and ASCII Delimited Quoted Text.

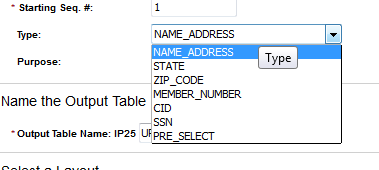
* EBCDIC is the mainframe file format.
* ASCII Delimited format will contain the inputs separated by a delimiter (E.g. Comma, Tab, Space, Colon, Semicolon and Pipe).
* ASCII Delimited Quoted Text format will contain the inputs separated by a delimiter, but there can be other special characters in between the fields. For e.g. Name can have some special characters. (E.g. Comma, Tab, Space, Colon, Semicolon, Pipe).

**Record Length:** The maximum record size = 999,999. When the record size entered exceeds 9999 and user clicks Preview button on the layout configuration page a message is displayed: "The Preview feature is unavailable for files with a record length > 9,999". Also, when the user clicks on the file link on the File Import process summary this message is displayed. "FILE CONTENTS CANNOT BE DISPLAYED FOR FILES WITH A RECORD LENGTH > 9,999."

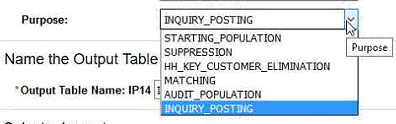
What happens at the other end is that when the maximum amount of data that can be displayed has been exceeded. The data in the display are truncated to the first 9,999 bytes of each record. User must use the layout grid to manually define fields that extend beyond this point. The Preview option on this page and on the Summary page will be unavailable."

**Sequence Number:** It is a unique number which identifies the record. This will continue while taking the record from the second file. So the sequence number of the second file will be the continuation of the sequence number from the first file.

**Type:** Contains the information about the input file. It would specify what the input file contains. So the input file can contain Name and address or CID or SSN etc. This is used to tag or let the user know what the input file contains.



**Purpose:** This field provides information on what the file is used for.



## 2.3 Name the Output Table

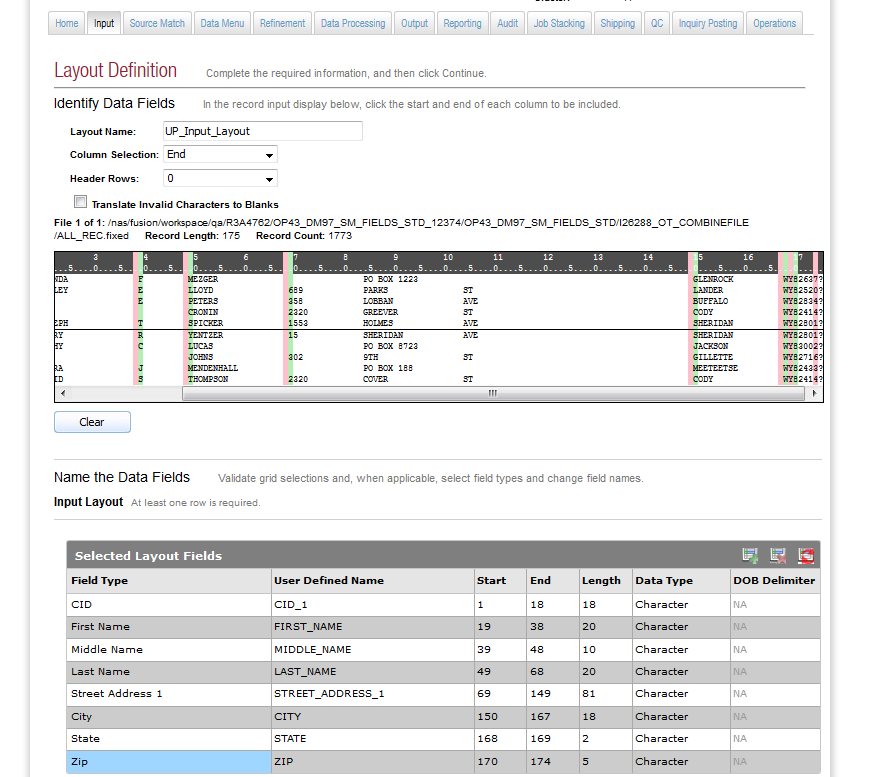
Logical name of table in GreenPlum where the Input fields are stored.

## Select a Layout

* + 1. Create layout – creates a new layout.
    2. Import Layout – Can import a custom layout and it must be in csv.
    3. Use Existing – It will list the layout used within a project. We can search for a layout in other project and use that in the current project. It’s worth noting that a layout is linked to the project in which it’s used. Therefore, if you use a layout from another project, a copy of that layout will be made and linked to the current project.

### Layout Definition

Here we can define a layout for the input file.



**The Column Selection and Header Rows:** The Column selection controls how the UI will behave. When you have the Column Selection set to Start, each mark made in the preview indicates the starting position of a field (column). When set to End, each mark indicates the ending position of a field (column). When set to Sequential Columns, each mark denotes the end of the field to its left, and the next field begins in the first byte to its right. Header Rows is used to indicate whether or not the file contains header records that should be omitted when the file is imported. Yes, there is a preview of the file which displays the first and last five records.

For EBCDIC files, the file contents are shown with the display value (character) on top of the corresponding hex value. ASCII files only show the display value. These fields will specify how the new layout should be. There is a Doc view which represents the first five and last five records of the file. With the selection in the Column Selection, we can specify the start and end of the columns. It also contains the Hex representation of the data.

**Name the Data Fields:** This would specify the field types and names for the inputs in the Database. We can specify the data type start and end etc. for the Fields that needs to be inserted into the database. GreenPlum will store the data in ASCII format.

The Generate Positions Table button inserts a row into the Layout Fields table for each field defined in the file preview.

**Grid View:** EOL and characters other than ASCII will be displayed in the layout Grid as?. If the record inserted contains a LF or CR , then layout record algorithm = length of record – 1. If record contains LF and CR, then layout record algorithm = length of record - 2. These algorithms will exclude the EOL character(s) from the layout preview.

**Layout Preview:** In case a user selects a field containing EOL characters or field exceeds length of EOL character an error message will display at the time of saving or continue saying ……….**"<Field Name> cannot contains EOL characters".**

**Add Column:** Add a constant into Green Plum Database. Add a value to every record.

Example: If more than one file, we can tag from which file the data come from. In the below, all data from ALL\_REC.fixed is having an additional field in ‘UP\_INPUT’, than mentioned in the fields, and its value will be A. ‘Tag’ is the field name in the Database.

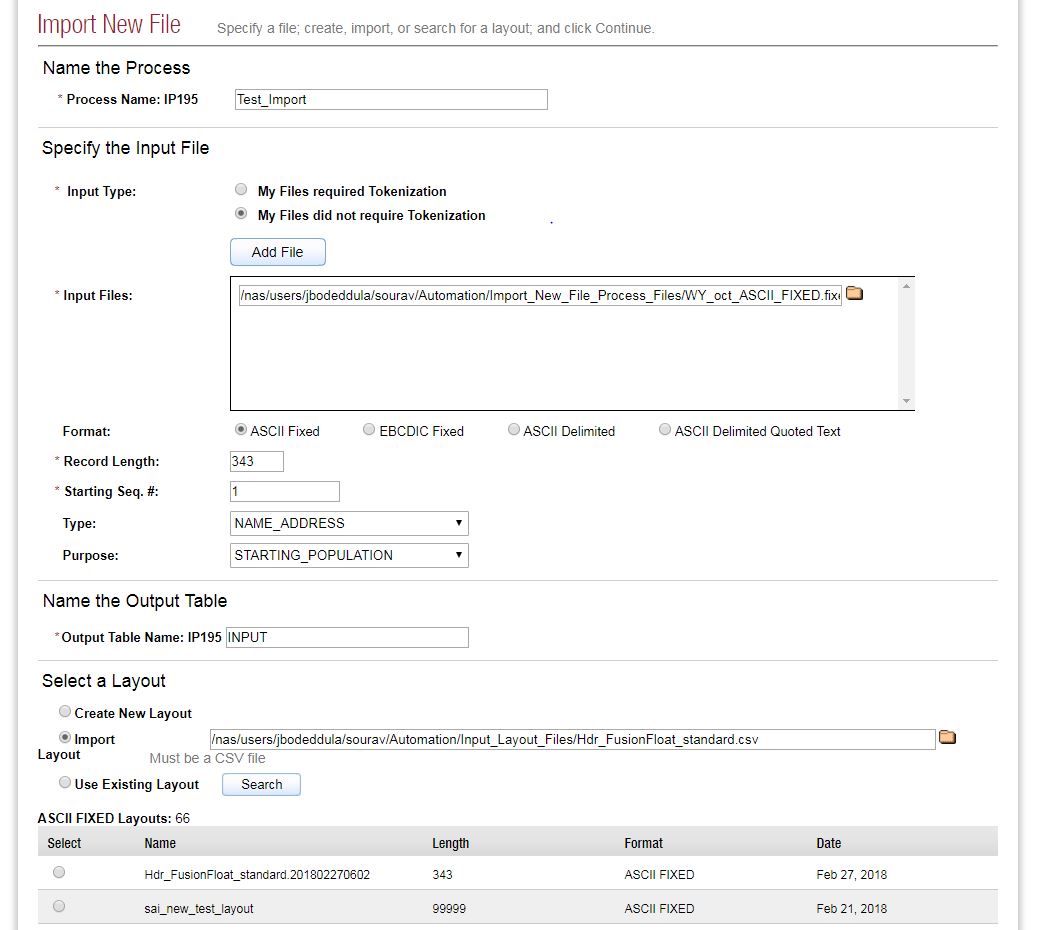


### Import Layout

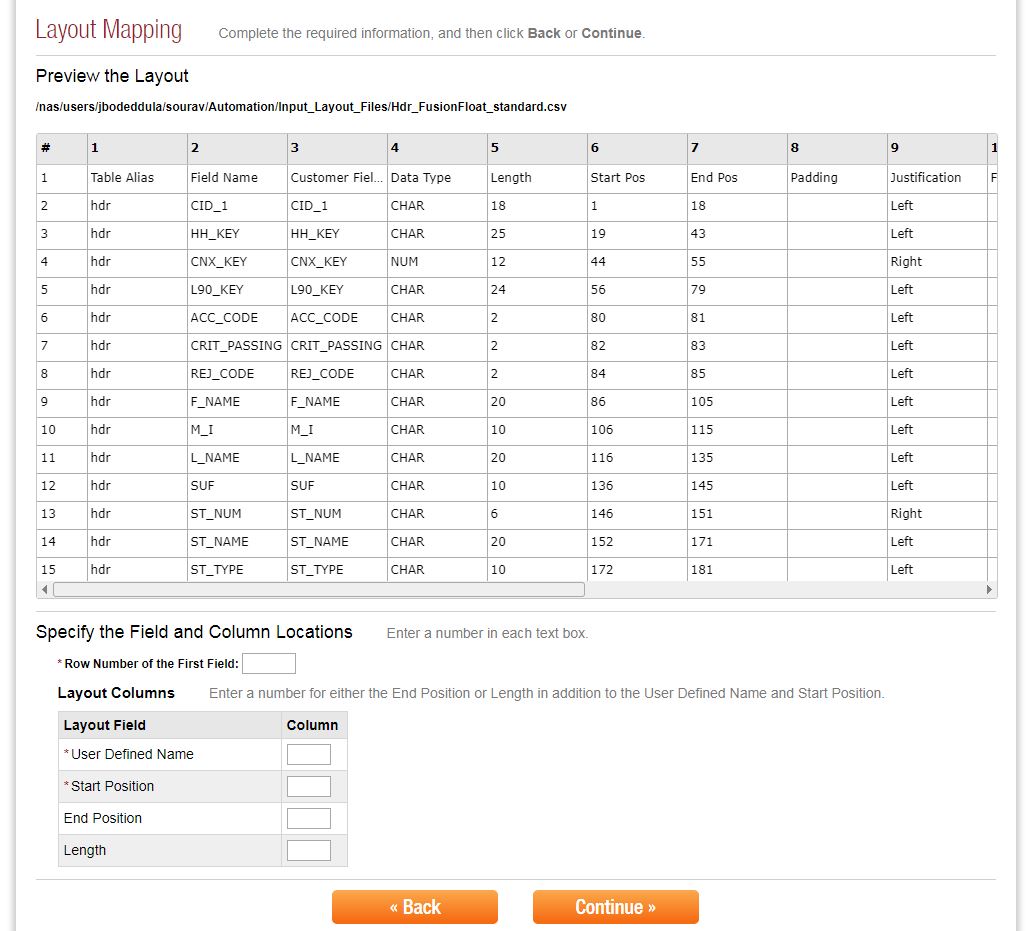
In order to import a layout one must provide the path of the file. The file format must be CSV having a list of record specifying name, length, format, date. For every file imported these validations must perform.

1. Every record imported must contain the same distinct Job #, which is the first 22 bytes of the MLA Transaction ID, across all files imported. Job will fail, if more than one distinct job # appears in the MLA Transaction ID field.
2. MLA Transaction ID must have a length of 40 bytes.
3. Data type validation (Only Char data type is allowed for MLA Transaction ID) for the entire file format.

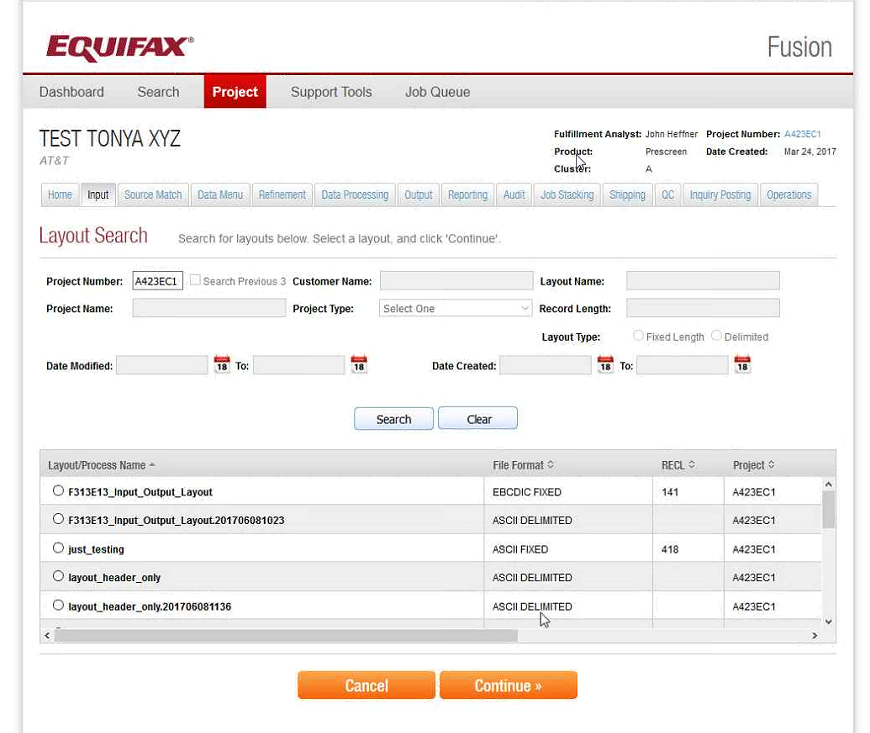
Continue tab navigates to Layout Mapping.



**Layout Mapping:** Preview of the layout of CSV file is displayed. Specifying the fields and layout column details and continuing will create a layout Definition of the file imported.



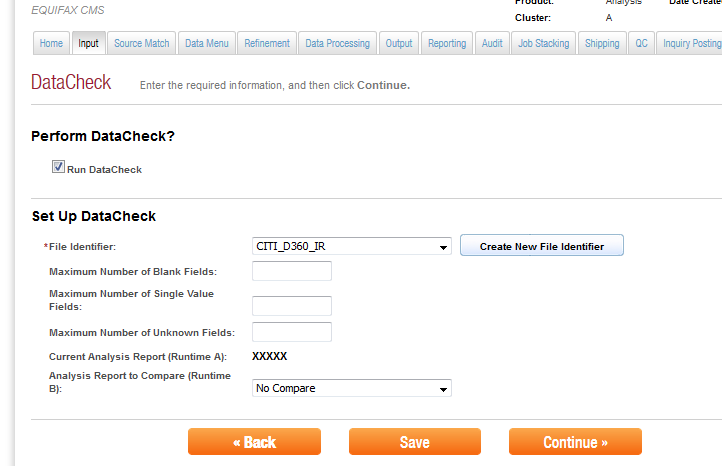
### Layout Search



**Continue:** By selecting a layout and clicking Continue, a copy of that layout is made and linked to the current project. Only a reference to the layout would be stored with the process configuration.

# 3. Run Data check

Run Data Check is used for field validations against input fields. It will compare the datatype of fields with the datatype in the layout selected. It can analyze, if every record is blank or it contains any unprintable characters. File Identifier is an option to create a unique identifier name to the generated file.



# 4. Summary

This page shows the summary of the process we have created. It will show the layout and other options that are selected for the process. Once we click Submit button, no change can be made to the process. The output of the process would be saved in Green Plum database.

