Emissions Inventory Import Test Plan

There are three pieces to the application, each of which requires detailed testing. In the first piece, a screen is presented where the user selects the “Inventory Year,” the “Content Type,” the “Regulatory Requirement,” the “Emissions Inventory Data File to Import,” the “Operation to Perform,” and any Attachments that are to be included. After the screen data is filled in, a button with text that matches the Operation (“Analyze” or “Import”) is presented. If the choice is to Analyze, no actual EIs will be created, but the validations described below will be run. The first piece of the code then takes the screen data, reads the CSV file, creates several objects to hold the data, and performs some basic validations on the data in the file. Assuming those validations pass, the second piece of code is given the objects from the first piece and creates the Emissions Inventory objects that can be stored in the DB by matching the imported data with pollutant definitions data. These EI objects are then returned to the import code which runs a third piece of code that performs detailed validations on the objects. If the Operation chosen was “Import,” the objects are then stored in the DB. Finally, any errors or warnings from the basic validations, the creation of the EI objects, and the detailed validations are made available to the user via a link on the original “Import Emissions Inventory Data” screen.  
  
The format of the CSV import file is provided in the attached CSV (Note that an actual import file will be a CSV file and will not contain the 1st header row):  
  


# Step 1: File Import and Basic Validations

The following CSV file may be suitable for testing the items in this step, although it may require some editing depending on facility and service catalog used for the tests.  
  


Step 1 begins by checking the validity of the Service Catalog associated with the “Inventory Year” and “Regulatory Requirement” selected on the screen.

Test 1: Go the 1st level tab “Emissions Inventories.” Go to the 2nd level tab “Service Catalog.” Expand the “Emission Inventories” node. For a given year (say, 2017), note the EI service catalogs for that year. These will be present in the list using the abbreviation patterns

Year\_ContentType\_RegulatoryRequirement.  
  
You may see, for example, the node 2017 AC TNTVP where the Year is 2017, the AC Content Type means “Annual Complete” and the TNTVP Regulatory Requirement means “Triennial Non-Title V Program.”

Note a Service Catalog you do **not** see, for example 2017\_WC\_UGRBNA (the 2017 Winter Complete Upper Green River Basin Non-Attainment Area). Go to the 2nd level tab “EI Data Import.” Try and import a file for the non-existent service catalog by picking the Year, Content Type, and Regulatory Requirement for the non-existent Service Catalog. The result should produce a message such as:  
  
**An error has occurred: No Service catalog found for : 2017 WC TNTVP. 0 rows processed.**There should be links to the log file and error file. The error file should contain an error similar to the one above.

Test 1 Notes and Issues:

Currently passes.

Test 2: Return to the 2nd level “Service Catalog” tab. Select a Service Catalog. There are three pollutant sections: “Criteria Air Pollutants/Other,” “Non-Chargeable Pollutants,” and “Data Import File Pollutants.” The 1st and 3rd sections determine the number of pollutant columns that must be present in the CSV import file. The number of expected pollutant columns in the CSV file beginning in column Y must be exactly 2 x (Number of “Criteria Air Pollutants/Other” pollutants + Number of “Data Import File Pollutants”). In each case, the 1st column for each pollutant will be the stack emissions for that pollutant and the 2nd column for each pollutant will be the fugitive emissions for that pollutant. For this test, the import file will not have enough pollutant data columns for all of the pollutants in the service catalog. For this test, the attached Step-1.csv file would need a service catalog with a total of 9 pollutants. If it were correct, it would have numeric data in columns Y through AP. Try and “Analyze” the file. The error file should contain an error like “Validation error in Row 2 with the following errors: Row does not have expected number of columns. 42 columns expected.” Repeat by adding too many columns of data in all rows.

#### Test 2 Notes and Issues:

Test 2 produces a warning rather than an error if there are too many columns. If one row has enough columns, but other rows do not, this is treated as a warning rather than an error.

Test 3: Edit the csv file so that for two or more rows, the 1st three columns (Facility Id, EU Id, and Process Id) have the same values. Try and analyze the file. This should produce a warning: “Duplicate row found for combination: PRC002. Skipping row.”

Test 3 Notes and Issues:

Update (2/13/18) - Null pointer is fixed.

This currently dumps a null pointer exception to the console:  
  
2018-02-05 16:17:35,321 15115174 [ Thread-18] ERROR p.emissionsReport.EiDataImport - Unable to validateRow()

java.lang.NullPointerException

at us.wy.state.deq.impact.app.emissionsReport.EiDataImport.importEiData(EiDataImport.java:302)

at us.wy.state.deq.impact.app.emissionsReport.EiDataImport$RunImport.run(EiDataImport.java:205)  
  
  
Test 4: Edit the csv file so that the facility Id does not exist in the system. This should produce an error like: “Facility id, EU id and Process Id combination does not exist in IMPACT for Facility id: F111454, EU Id: ENG002, Process Id: PRC002.” Repeat for bad EU Id and bad Process Id.

Test 4 Notes and Issues:

Currently passes.

Test 5: We should probably test early in the process that EIs have been enabled for all facilities in the CSV. Go to the “Facility Detail” page for a facility in the CSV file. Select the “Enabled Emissions Inventories” link. Make sure that EI reporting is **not** enabled for the chosen facility and service catalog. Go back to “Emissions Inventories” -> “EI Data Import” and analyze the file. Since inventories have not been enabled, this should produce an error. (Re)Enable reporting for the facility. This error should not be reported for the facility.

Test 5 Notes and Issues:   
  
Update (2/13/18) – Check is implemented

This check has not been implemented.

# Step 2: Creation of the EI Objects

Test 6: This is a test to make sure the generated EI correctly handles the EUs associated with the facility. EUs associated with the facility can be handled in one of four ways:  
  
1) EU is present in the CSV file and was operational at some point during the reporting year. EU must be included in the generated EI.

2) EU is present in the CSV file but was not operational for the entire reporting year. This must be flagged as an error in the error file.

3) EU is not present in the CSV file and was operational at some point during the reporting year. The EU must be added to the EI but marked as “Did Not Operate.”

4) EU is not present in the CSV file and was not operational for the entire reporting year. These EUs must be ignored in the generated EI and in the errors file.

Begin by selecting a facility with a mix of operating and non-operating EUs. In the CSV file, create rows for some, but not all of the operating EUs. Add at least one row for a non-operating EU. Import the file and Analyze. Since case 2) above is included, the EU must be flagged as an error.

Remove the offending EU from the CSV file. Reimport the CSV. There should be no error. If the “Perform Data Import” option is chosen (and there are no other errors), all EUs in case 1) should be present in the generated EU, all EUs in case 3) will show up in the EI marked as “Did Not Operate,” and non-operational case 4) EUs will not be present.   
  
If the “Perform” option was chosen, the generated EI from the test can be located by selecting the “Emissions Inventory Search” tab and setting the “Year,” “Content Type,” and “Regulatory Requirement” to match the values of the Service Catalog used in the import. Set the “Reporting State” to “Approved” to further narrow the search.

Test 6 Notes and Issues:

Test 7: This test is to make sure that all processes associated with an EU that is present in the CSV file are included in the CSV file. Setup requires locating a facility with an operational EU that has two or more associated processes. One row with, for example, Facility Id Fxxx, EU Id EUyyy, and Process Id P001 should be in the CSV file while the 2nd row Facility Id Fxxx, EU Id EUyyy, and Process Id P002 is left out. “Analyze” should produce an error in the error file.

Test 7 Notes and Issues:

Test 8: For every pollutant present in the CSV, the code will attempt to locate one or more active fire rows associated with the EU Process’s SCC Code, material, report year, action, and the pollutant. If no fire row is found, a warning is issued. Otherwise, the factor numeric value will be set for the pollutant in the EI. This value can be set in one of two ways:  
  
1) Either directly by using the fire row’s factor, or

2) Using the formula present in the fire row.

This test will require a Service Catalog with several pollutants.

1) A pollutant with one matching fire row that has a factor numeric value,

2) A pollutant with one matching row that uses a formula,

3) A pollutant with multiple matching rows that use formulas, and

4) A pollutant with no matching fire row.

Variations of the following SQL can be used to locate combinations for each case. This statement will work for case 3:

SELECT \* FROM cm\_scc cmscc

INNER JOIN (SELECT COUNT(\*) AS cnt, action\_cd, material\_unit\_cd, emissions\_unit\_cd,

scc\_id, pollutant\_cd, material\_cd

FROM rp\_fire\_factor

WHERE (deprecated\_year IS NULL OR deprecated\_year > 2017)

AND factor IS NULL

AND formula IS NOT NULL

GROUP BY action\_cd, material\_unit\_cd, emissions\_unit\_cd, scc\_id,

pollutant\_cd, material\_cd

HAVING COUNT(\*) > 1) AS rpff

ON (rpff.scc\_id = cmscc.scc\_id)

INNER JOIN rp\_material\_def rpmd

ON (rpmd.material\_cd = rpff.material\_cd)

The scc\_id’s must match those of EU processes to be included in the CSV file. The matching pollutant for each case must be included in either the Service Catalog’s “Criteria Air Pollutants/Other” list or the “Data Import File Pollutants” list.  
  
The formula cases will also need to include any variables used by the fire formula in the CSV file. Locate the fire row(s) and examine the formula(s). The formula will include one or more variables such as “HCg.” Each variable must be entered in each row of the CSV file. The variable must be entered as pairs beginning in column N. The first column of the pair is the variable name (e.g. “HCg”) and the second is the variable value. The spreadsheet design can accommodate formulas with up to five variables (columns N-W). Tests 9 and 10 below will test for various formula and variable problems.

With the data set up in the Service Catalog (pollutants), the facility’s EU processes (scc codes), and CSV file (fire variables), run the “Perform Import.” None of the cases above should produce errors. Locate the new EI and check the values. These will be present in the “Uncontrolled Emissions Factor (Lbs/Throughput Units)” column of the process’s “Process Emissions” table. The factor will have been derived either directly from the appropriate fire row or indirectly using the fire row’s formula and fire variable values.

Test 8 Notes and Issues:

Test 9: Formula variables that are present in the CSV file, but unused in the EI should be dropped from the EI. Add a meaningless variable to the CSV, i.e. a variable unused in any pollutant’s fire row formula. The variable should not be present on the EI screen for the completed EI.  
  
  
Test 9 Notes and Issues:

Test 10: Remove a variable that **is** used in a formula from the CSV file. A missing variable should produce an error.

Test 10 Notes and Issues:

# Step 3: Detailed Validations and DB Save

The 1st set of validations performed are governed by the validation control mechanism. Select the 1st level “Admin” tab. Expand the “Facilities” node and select the “Validation Control” node. Note the column labeled “EI.” Each row marked as “Yes” in the EI column refers to an individual validation that can be run on the EI. Tests of these will involve locating facilities, EUs, processes, and other items that do not meet each of the individual validations. Each validation should be turned on (“Yes”), an EI import with data that fails the validation created, and the import run to be sure the data fails the validation. Then the validation is to be turned off (”No”) and the validation re-run. The failed validation should now be ignored. The validations can be set by clicking on the link in the “Category” and setting the “EI” dropdown value. Detailed looks at these validations follows.  
  
Test 11: The first set of validations to be tested are the basic facility validations.

1) facilBasic: Do all required fields in the facility detail screen have values?

2) facilLocation: Is the facility’s latitude and longitude within bounds?

3) facilAfs: Is the falitity’s AFS value present and formatted as a 10 digit length number?

4) facilOwner: Are the facilitity’s owner and company non-null?

5) facilEnvCont: Does the facility have an environmental contact?

6) facilBillCont: Does the facility have a billing contact? This is also tested separately if the EI is considered billable, in which case there must be an active billing contact for the reporting year.

7) facilOnSiteCont: Does the facility have an on site contact?

8) facilOfficialCont: Does the facility have a responsible official contact?

9) facilNaics: Does the facility have at least one NAICS code that is not deprecated?

10) facilAPI: Does the facility have an API that is not deprecated?

11) facilFedRules: Federal rules and regs are valid (Part 60 NSPSPART, Part 61 NESHAP, and Part 63 NESHAP Subparts).

12) facilOpStat: Is the facility operating status valid?

13) facilNSRBillCont: Does the facility have an NSR billing contact?

Test 11 Notes and Issues:

Test 12: This set of validations checks each emission unit.

1) euBasic: Checks emission unit type, operating status, company description, company ID, and AQD description (Internal).

2) euAdditional: This provides specific checks based on the emission unit type.

3) euSerialTracking: Serial number tracking data is validated for EU types: Boiler, Crushing/Screening/Handling, and Calciner/Kiln/Dryer/Smelter/Foundary Furnace.

4) euEngineTracking: Serial number tracking data is validated for EU type: Engine.

5) euProcesses: Ensure that the EU has process(es).

6) euPte: Ensure EU potential emissions are valid.

7) euOpStat: EU Operating Status and various EU dates are consistent (if Shutdown there is shutdown date. If operating has Commence Operation date, etc.).  
  
  
Test 12 Notes and Issues:

Test 13: Check the emissions report itself.

Required variables:

* hoursPer[Day|Week|Year] within range
* 0 < winterThroughputPct <= 100
* 0 < springThroughputPct <= 100
* 0 < summerThroughputPct <= 100
* 0 < fallThroughputPct < =100
* Sum of seasonal throughputsPct == 100.
* firstHalfHrsOfOperationPct
* secondHalfHrsOfOperationPct
* if there is a trade secret, there must be a justification.
* Is there a conflict between 0/non-0 throughput and non-0/0 actual hours?
* Actual Hours cannot be less than the hours of uncontrolled operation for each pollutant.

Fire Variables:

Test 13 Notes and Issues:

Test 14: For each Fire Variable, check the allowed min and max values in the rp\_fire\_variable\_names\_def table to be sure that all Fire Variable’s used in the spreadsheet are within range.  
  
  
Test 14 Notes and Issues:

Test 15: Ensure that all fields are correctly stored in the database.

|  |  |  |  |
| --- | --- | --- | --- |
| **RP\_EMISSIONS\_RPT** | | | |
| **COLUMN** | **NULLABLE?** | **VALUE** | **COMMENTS** |
| EMISSIONS\_RPT\_ID | N | Auto generated |  |
| EIS\_STATUS\_CD | N | 3ur | 3ur = Submitted |
| PRIMARY\_ADDR | Y | NULL |  |
| REPORT\_MODIFIED | Y | NULL |  |
| FP\_ID | N | Supplied by the csv parser | Set to fp\_id of the current facility inventory |
| RPT\_RECEIVED\_STATUS\_CD | N | 10aa | 10aa = Approved |
| FEE\_ID | Y | Supplied by the csv parser | Set to the first half fee id from the associated service catalog |
| COMPANION\_REPORT\_FK | Y | NULL |  |
| PREV\_OWNER\_FORWARDING\_ADDR | Y | NULL |  |
| NEW\_OWNER\_ADDR | Y | NULL |  |
| BILLING\_ADDR | Y | NULL |  |
| RPT\_RECEIVED\_STATUS\_DT | Y | NULL |  |
| TOTAL\_EMISSIONS | Y | Sum total of emissions of all pollutants – minus emissions from pollutants that are part of a pollutant set that is chargeable as a set | See updateTotalEmissions method in the EmissionsReportBO class |
| REPORT\_YEAR | N | Supplied by the csv parser |  |
| REPORT\_APPROVED\_STATUS\_DT | Y | Today’s date |  |
| RECEIVED\_DATE | Y | Today’s date |  |
| TRANSFER\_DATE | Y | NULL |  |
| NEW\_OWNER | N | N | Based on the value for other approved reports in the system |
| SHUTDOWN\_DATE | Y | NULL |  |
| FACILITY\_NM | Y | NULL |  |
| PROVIDE\_BOTH\_YEARS | Y | N | Based on the value for other approved reports in the system |
| LEGACY\_FLAG | Y | N |  |
| LAST\_MODIFIED | N | 1 | Defaulted to 1 by DB |
| AUTO\_GENERATED | N | Y | For CSV reports this flag should be set to Y |
| REVISION\_REASON | Y | NULL |  |
| EI\_ID | Y | Computed |  |
| TOTAL\_REPORTED\_EMISSIONS | Y | Sum total of emissions of all non-chargeable pollutants | See updateTotalEmissions method in the EmissionsReportBO class |
| FEE\_ID2 | Y | Supplied by the csv parser | Set to the second half fee id from the associated service catalog |
| VALIDATED\_FLAG | N | Y | All reports created from the CSV are validated |
| INVOICE\_AMOUNT | Y | NULL |  |
| INVOICE\_DATE | Y | NULL |  |
| PAYMENT\_RECEIVED\_DATE | Y | NULL |  |
| SUBMITTED\_BY\_USER | Y | Set to the user id of the user running the import |  |
| SUBMITTED\_BY\_CONTACT | Y | NULL |  |

Test 15, RP\_EMISSIONS\_RPT Notes and Issues:

|  |  |  |  |
| --- | --- | --- | --- |
| **RP\_REPORT\_EU** | | | |
| **COLUMN** | **NULLABLE?** | **VALUE** | **COMMENTS** |
| EMISSIONS\_RPT\_ID | N | Auto generated |  |
| CORR\_ID | N | Supplied by the csv parser | CORR\_EPA\_EMU\_ID of the emissions unit |
| EXEMPT\_BY\_EG71 | N | N |  |
| LAST\_MODIFIED | N | 1 | Defaulted to 1 by DB |
| ZERO\_EMISSIONS | N | Y | Set to Y when the EU is not included in the csv file |
| N | Set to N when the EU is included in the csv file |
| FORCE\_DETAILED\_REPORTING | N | Y | Set to Y when the EU is included in the csv file |
| N | Set to N when the EU is not included in the csv file |

Test 15, RP\_REPORT\_EU Notes and Issues:

|  |  |  |  |
| --- | --- | --- | --- |
| **RP\_REPORT\_PERIOD\_XREF** | | | |
| **COLUMN** | **NULLABLE?** | **VALUE** | **COMMENTS** |
| EMISSION\_PERIOD\_ID | N | Auto generated |  |
| EMISSIONS\_RPT\_ID | N | Auto generated |  |
| CORR\_ID | Y | Supplied by the csv parser | CORR\_EPA\_EMU\_ID of the emissions unit |
| REPORT\_EU\_GROUP\_ID | Y | NULL |  |
| LAST\_MODIFIED | N | 1 | Defaulted to 1 by DB |

Test 15, RP\_REPORT\_PERIOD\_XREF Notes and Issues:

|  |  |  |  |
| --- | --- | --- | --- |
| **RP\_EMISSION\_PERIOD** | | | |
| **COLUMN** | **NULLABLE?** | **VALUE** | **COMMENTS** |
| EMISSION\_PERIOD\_ID | N | Auto generated |  |
| SCC\_ID | Y | Supplied by the csv parser |  |
| FORMULA\_VARIABLES | Y | NULL |  |
| WINTER\_THROUGHPUT\_PCT | Y | Supplied by the csv parser |  |
| SPRING\_THROUGHPUT\_PCT | Y | Supplied by the csv parser |  |
| SUMMER\_THROUGHPUT\_PCT | Y | Supplied by the csv parser |  |
| FALL\_THROUGHPUT\_PCT | Y | Supplied by the csv parser |  |
| DAYS\_PER\_WEEK | Y | Supplied by the csv parser |  |
| WEEKS\_PER\_YEAR | Y | Supplied by the csv parser |  |
| HOURS\_PER\_DAY | Y | Supplied by the csv parser |  |
| HOURS\_PER\_YEAR | Y | Supplied by the csv parser |  |
| SCHED\_TS | N | N |  |
| SCHED\_TS\_JUST | Y | NULL |  |
| LAST\_MODIFIED | N | 1 | Defaulted to 1 by DB |
| FIRST\_HALF\_HOURS\_OF\_OPERATION\_PCT | N | 50 |  |
| SECOND\_HALF\_HOURS\_OF\_OPERATION\_PCT | N | 50 |  |

Test 15, RP\_EMISSION\_PERIOD Notes and Issues:

|  |  |  |  |
| --- | --- | --- | --- |
| **RP\_EMISSIONS\_RPT\_SC\_EMISSIONS\_REPORT\_XREF** | | | |
| **COLUMN** | **NULLABLE?** | **VALUE** | **COMMENTS** |
| EMISSIONS\_RPT\_ID | N | Auto generated |  |
| SC\_EMISSIONS\_REPORT\_ID | N | Supplied by the csv parser | Check with AQD if import process need to support selecting multiple regulatory requirements? |

Test 15, RP\_EMISSIONS\_RPT\_SC\_EMISSIONS\_REPORT\_XREF Notes and Issues:

|  |  |  |  |
| --- | --- | --- | --- |
| **RP\_MATERIAL\_ACTIONS** | | | |
| **COLUMN** | **NULLABLE?** | **VALUE** | **COMMENTS** |
| EMISSION\_PERIOD\_ID | N | Auto generated |  |
| MATERIAL\_CD | N | Supplied by the csv parser |  |
| MATERIAL\_UNIT\_CD | N | Value is derived from the RP\_FIRE\_FACTOR table for the given SCC\_ID and MATERIAL\_CD |  |
| ACTION\_CD | N | Value is derived from the RP\_FIRE\_FACTOR table for the given SCC\_ID and MATERIAL\_CD |  |
| THROUGHPUT | Y | Supplied by the csv parser |  |
| MATERIAL\_TS | N | N |  |
| MATERIAL\_TS\_JUST | Y | NULL |  |
| THROUGHPUT\_TS | N | N |  |
| THROUGHPUT\_TS\_JUST | Y | NULL |  |
| LAST\_MODIFIED | N | 1 | Defaulted to 1 by DB |

Test 15, RP\_MATERIAL\_ACTIONS Notes and Issues:

|  |  |  |  |
| --- | --- | --- | --- |
| **RP\_PERIOD\_VARIABLES** | | | |
| **COLUMN** | **NULLABLE?** | **VALUE** | **COMMENTS** |
| EMISSION\_PERIOD\_ID | N | Auto generated |  |
| VARIABLE | N | Supplied by the csv parser |  |
| VALUE | Y | Supplied by the csv parser |  |
| VALUE\_TS | Y | N |  |
| VALUE\_TS\_JUST | Y | NULL |  |
| LAST\_MODIFIED | N | 1 | Defaulted to 1 by DB |

Test 15, RP\_PERIOD\_VARIABLES Notes and Issues:

|  |  |  |  |
| --- | --- | --- | --- |
| **RP\_EMISSIONS** | | | |
| **COLUMN** | **NULLABLE?** | **VALUE** | **COMMENTS** |
| EMISSION\_PERIOD\_ID | N | Auto generated |  |
| EMISSION\_UNIT\_NUMERATOR | Y | Should be set to TON per AQD |  |
| FACTOR\_UNIT\_NUMERATOR | Y | NULL |  |
| FACTOR\_UNIT\_DENOMINATOR | Y | NULL |  |
| POLLUTANT\_CD | N | Supplied by the csv parser |  |
| FIRE\_ID | Y | Value is derived from the RP\_FIRE\_FACTOR table for the given SCC\_ID, POLLUTANT\_CD, and MATERIAL\_CD |  |
| EMISSION\_CALC\_METHOD\_CD | Y | Supplied by the csv parser |  |
| MATERIAL\_CD | Y | NULL |  |
| FACTOR\_NUMERIC\_VALUE | Y | In the corresponding FIRE row, if factor is not NULL, then set to the factor value in the FIRE row else set to the value derived from evaluating the formula in the FIRE row. |  |
| FUGITIVE\_EMISSIONS | Y | Supplied by the csv parser |  |
| STACK\_EMISSIONS | Y | Supplied by the csv parser |  |
| ANNUAL\_ADJUSTMENT | Y | NULL | Corresponds to uncontrolled hours |
| EXPLANATION | Y | NULL |  |
| FACTOR\_TS | N | N |  |
| FACTOR\_TS\_JUST | Y | NULL |  |
| EXPLAIN\_TS | N | N |  |
| EXPLAIN\_TS\_JUST | Y | NULL |  |
| LAST\_MODIFIED | N | 1 | Defaulted to 1 by DB |
| TIME\_BASED\_FACTOR\_NUMERIC\_VALUE | Y | NULL |  |
| FACTOR\_NUMERIC\_VALUE\_OVERRIDE | Y | N |  |

Test 15, RP\_EMISSIONS Notes and Issues:

|  |  |  |  |
| --- | --- | --- | --- |
| **RP\_REPORT\_POLLUTANT\_TOTALS** | | | |
| **COLUMN** | **NULLABLE?** | **VALUE** | **COMMENTS** |
| EMISSIONS\_RPT\_ID | N | Auto generated |  |
| POLLUTANT\_CD | N | Supplied by the csv parser |  |
| TOTAL\_EMISSIONS | Y | Sum total of emissions across all reported emissions units for a given chargeable pollutant | Rows are inserted into this table when an emissions inventory report is submitted. |
| LAST\_MODIFIED | N | 1 | Defaulted to 1 by DB |

Test 15, RP\_REPORT\_POLLUTANT\_TOTALS Notes and Issues:

Test 16: Create an EI from scratch using the same year, content type, pollutant values, etc as the import. Compare values in the tables above and note any discrepancies.

Test 16 Notes and Issues:

Test 17: Create an EI from scratch in release and compare it to the results from Test 16 above.

Test 17 Notes and Issues:

Test 18: Attachments should be tested by adding at least two attachments in the initial import screen. One attachment must be a trade secret document and the other a normal document. When selecting the trade secret document type, the justification field is required. Once the documents have been selected and uploaded, import a CSV file with at least two separate facilities. Once the import is complete, search for the new EIs, open each in turn, and make sure all documents are present in the EI and can be downloaded.  
  
Test 18 Notes and Issues:

Test 19: Upon creating a new Emissions Inventory through import, previous EIs for the same Year and Content type must be set to “Prior/Invalid” status.

Test 19 Notes and Issues:

Test 20: All validations in csv parser (this falls under step 1)

|  |  |  |
| --- | --- | --- |
| **Column number in csv** | **Validation Message** | **Notes** |
| 1 | Facility id is empty | Facility id is empty for the row in csv |
| 1 | Facility Fxxxxxx does not have reporting enabled for selected year and content type | For given facility Fxxxxxx, no reporting has been enabled |
| 2 | EU id is empty | EU id is empty for the row in csv |
| 3 | Process id is empty | Process id is empty for the row in csv |
| 4 | Maximum Hours Per Day is empty |  |
| 4 | Maximum Hours Per Day is not a valid value | Value provided is not numeric |
| 4 | Maximum Hours Per Day X is not in range. Expected range is 0 to 24 |  |
| 5 | Maximum Days Per Week is empty |  |
| 5 | Maximum Days Per Week is not a valid value | Value provided is not numeric |
| 5 | Maximum Days Per Week X is not in range. Expected range is 0 to 7 |  |
| 6 | Maximum Weeks Per Year is empty |  |
| 6 | Maximum Weeks Per Year is not a valid value | Value provided is not numeric |
| 6 | Maximum Weeks Per Year X is not in range. Expected range is 0 to 52 |  |
| 7 | Actual Hours is empty |  |
| 7 | Actual Hours is not a valid value | Value provided is not numeric |
| 7 | Actual Hours YYYY is not in range. Expected range is 0 to ZZZZ |  |
| 8 | Winter percentage is empty |  |
| 8 | Invalid value for Winter percentage | Value provided is not numeric |
| 8 | Winter percentage is a negative value |  |
| 9 | Spring percentage is empty |  |
| 9 | Invalid value for Spring percentage | Value provided is not numeric |
| 9 | Spring percentage is a negative value |  |
| 10 | Summer percentage is empty |  |
| 10 | Invalid value for Summer percentage | Value provided is not numeric |
| 10 | Summer percentage is a negative value |  |
| 11 | Fall percentage is empty |  |
| 11 | Invalid value for Fall percentage | Value provided is not numeric |
| 11 | Fall percentage is a negative value |  |
| 8,9,10,11 | Winter, Spring, Summer & Fall percent total must equal and not exceed 100% |  |
| 12 | Material is empty |  |
| 12 | Material x not found in definition list or is Inactive | Material was not found in Material Definition list or is deprecated or doesn’t match a material for Scc code |
| 13 | Throughput is empty |  |
| 13 | Throughput is a negative value |  |
| 13 | Throughput is not a valid value | Value provided is not numeric |
| 14 .. 23 | Fire variable x not found in definition list | Variable name was not found in Fire Variables Definition list |
| 14 .. 23 | Fire Variable X does not have a valid value | Not a number or scientific value |
| 14 .. 23 | Fire Variable X available, but value is empty | Csv has entry for variable name, but no value for it |
| 14 .. 23 | Fire variable value is negative for Fire variable: S |  |
| 14 .. 23 | Duplicate Fire Variable S found | Row has the same fire variable listed more than once |
| 24 | Calculation Method is empty |  |
| 25 .. n | Actual Hours and Throughput are zero, but pollutants have values |  |
| 25 .. n | Stack Emission value is blank for CAP Pollutant: X | CAP pollutant value cannot be null |
| 25 .. n | Fugitive Emission value is blank for CAP Pollutant: X | CAP pollutant value cannot be null |
| 25 .. n | Stack Emission is a negative value for Pollutant: X | Emission cannot be negative |
| 25 .. n | Fugitive Emission is a negative value for Pollutant: X | Emission cannot be negative |
| 25 .. n | Stack Emission value is invalid for Pollutant: X | Emission value is not valid (not numeric) |
| 25 .. n | Fugitive Emission value is invalid for Pollutant: X | Emission value is not valid (not numeric) |

Test 20 Notes and Issues:

Test 21: For any EI created through EI import feature, verify the following:

* New flag added to EI header. **Generated from Imported File : Yes**
* For any process(row) in the csv file – If the Calculation method starts with ‘E’ or ‘e’, set the calculation method for that process in the EI to ‘*Emissions’*, else set to *‘AQD Generated’*.

Test 21 Notes and Issues:

Test 22: Cloning an imported EI. When an EI created through data import is cloned, verify the following:

* New flag added to EI header. **Generated from Imported File : No**
* For any process(row) in the csv file – If the Calculation method in the EI was *‘AQD Generated’, they should be set to ‘Emissions’ in the cloned EI*.

Test 22 Notes and Issues:

# Final Notes and Issues: