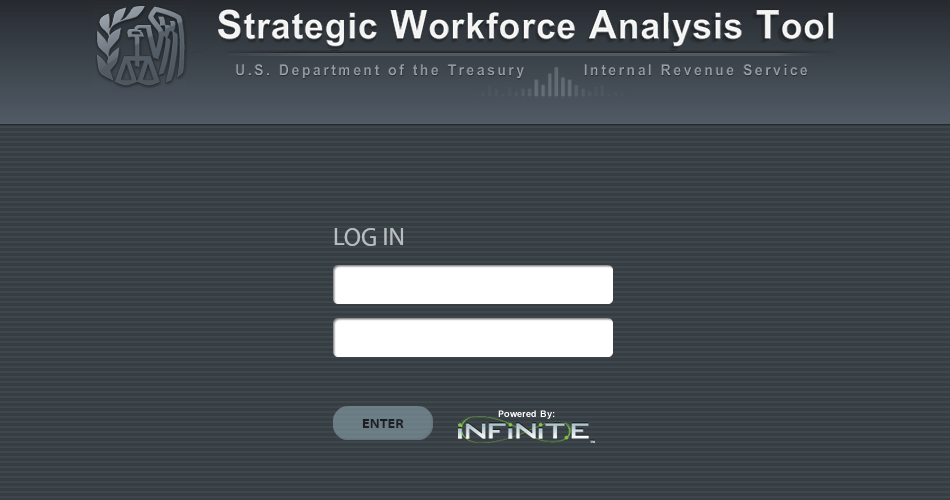
**Data Management and Import Documentation**

****

**Tuesday, April 17, 2012**

**Version 1.0**

**Overview**

The data upload process is managed by a set of server side scripts and data management processes. The data management processes are available to promote best practices to ensure data validity and accuracy. These are put into place to ensure the tool reports accurate information to the client. The rigor put into ensuring the accuracy of the data before import will help ensure the accuracy of the analytical results.

The server side tools that are part of the toolset allow the Strategic Workforce Analysis Toolkit (SWAT) administrator to manage the data environment and data upload process. The process is broken out into a set of small steps and tools for specific tasks. This provides the SWAT team with following capabilities.

* Backup the data environment
* Import new data into the environment
* Restore the data environment if necessary

We are anticipating that data will be added to the environment on a yearly basis and during that time testing will be performed to ensure the utmost accuracy and validity of the data and the tool accuracy.

This document assumes the user understands and is able to connect to the PostgreSQL server using the tools specified and has minimal set of data management experience. Those topics are not covered in this documentation.

**Types of Data**

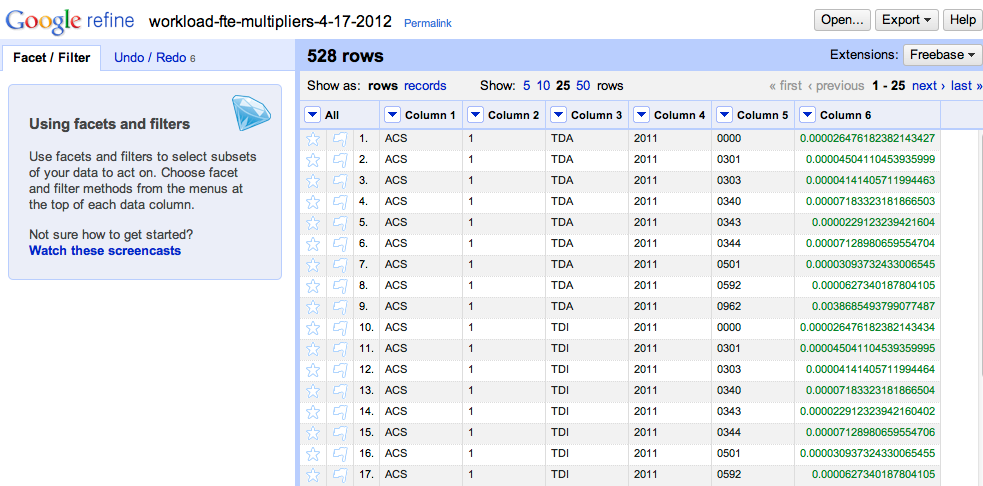
Two types of data can be managed through the SWAT. The first being the IRS workforce projections. These identify the number of workers the IRS anticipates measured over time. The second being the IRS workload projection that identifies the amount of work the IRS has to be able to perform over time. Each of these types of data is stored in various tables within the SWAT environment as noted below. The third important note is the *map* table which provide the default lever values and measures (these are not covered in the data import documentation will be managed separately)

**Observations**

Data provided by the IRS has been in the form of excel spreadsheets for both the workforce and workload data. In order to create a maintainable process the data import process will adhere to a *CSV* file format. In order to ensure the accuracy of the data before the upload process can begin it is imperative to ensure that an agreed data definition standard is put into place and a data review process is established before the physical upload process takes place.

The SWAT team has additionally created an Excel client macro that will aid in this process. This macro provides the ability to remove unnecessary characters and spaces errors that can occur when manually creating spreadsheets. This tool is meant to provide assistance in the data preparation process.

Other tools such as Google Refine <http://code.google.com/p/google-refine/> can be used for more powerful functions beyond excel for data preparation and cleanup.



**Data Preparation**

In preparing the data to be placed into the environment it is important to become familiar with the various PostgreSQL table names, commands, and the user interface of the tool itself. This document defines the data formats and structure of the information. There are also samples below that define the input formats for each type of data.

When preparing the data it is important to look for things such as **spaces**, **capitalization**, **column** **order**, etc. and correct before starting the import process.

Once this has been validated against the **table format** and data samples below the data is ready for the import process.

**Data Import Process and Tools**

**Tables**

**Workforce Tables**

* **tbl\_workforce** - Contains the IRS workforce projection information and associated metadata by year

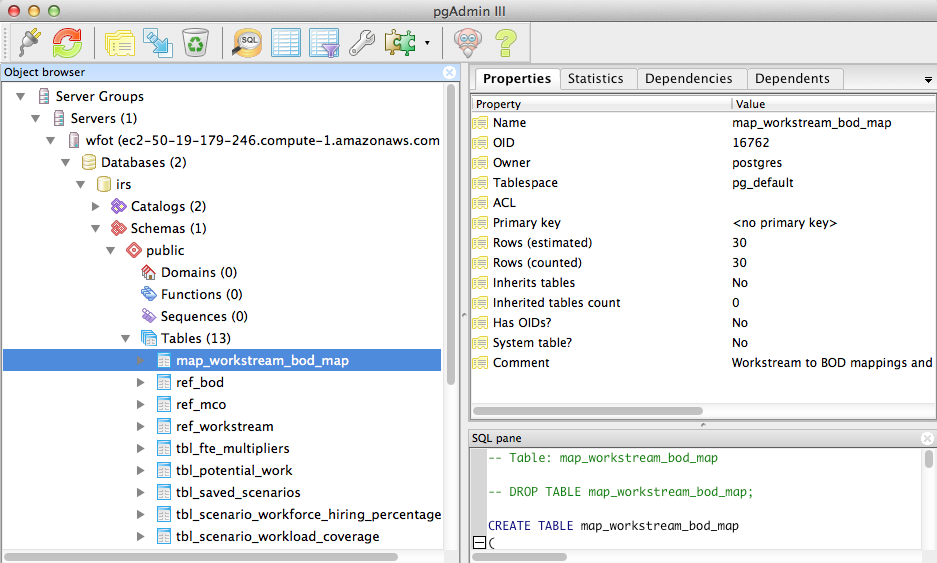
**Workload Tables**

* **tbl\_fte\_multipliers** - Contains the IRS workstreams and associated FTE multipliers by year
* **tbl\_potential\_work** - Contains the IRS workstreams and associated returns by year
* **tbl\_selection\_rates** - Contains the IRS workstreams and associated selection rates by year

**Tools**

All the tools specified below are server side tools that are necessary to maintain the data in platform. A system administrator that is familiar with command line access in Linux and simple Database Administration tasks is necessary.

If you require access to the SWAT database environment a PostgreSQL client can be used to access. The free and open source pgAdmin III client offers this functionality and can be downloaded here. <http://www.pgadmin.org/download/>



**Backup**

After connecting to the PostgreSQL database environment using pgAdmin III you can drill into the "IRS" database and from there right click to perform a "Backup"

or

The PostgreSQL tool for database backup can be used directly on the database server via terminal to backup the data environment. This must be executed via the command line on the database server.

pg\_dump dbname > outfile

Below is the example command for the SWAT IRS database to backup the environment

pg\_dump irs > irs-swat-backup-01262012

For additional information on pg\_dump http://www.postgresql.org/docs/8.1/static/backup.html

**Restore**

The PostgreSQL tool for database restore will be used to backup the data environment. This must be executed via the command line on the database server. This command will perform a full restore of the IRS database from an existing database dump.

psql dbname < infile

Below is the example command for the SWAT IRS database to restore the environment.

psql irs < irs-swat-backup-01262012

**Import**

Imports can occur for various reasons and to different areas of the database as required.

Below are some example reasons why an import might be performed.

1. New year data needs to be added to the system for "Workforce" and "Workload"
2. A corruption in the existing data and a table needs to be cleaned up.
3. An error is noticed in a specific workstream and a correction needs to be made to the data.

Before you begin importing the data a good next step is backup the invididual tables you wish to make modifications to. This can be done using simple SQL commands to backup the table.

SELECT \* INTO tbl\_workforce\_bak FROM tbl\_workforce;

SELECT \* INTO tbl\_fte\_multipliers\_bak FROM tbl\_fte\_multipliers;

SELECT \* INTO tbl\_potential\_work\_bak FROM tbl\_potential\_work;

SELECT \* INTO tbl\_selection\_rates\_bak FROM tbl\_selection\_rates;

This creates a full backup of the table before you begin with your import process. Make sure you go in and drop these “bak” tables to cleanup the structure of the database after you are complete with your import tasks.

If data cleanup needs to occur in the event "2." occurs above. You must remove all existing data from the table and reload it. The removal can be accomplished by using the SQL "DELETE" command for the specific table and a "WHERE" clause can be added to filter by specific fields.

DELETE FROM tbl\_fte\_multipliers WHERE keyvalue in (x, x, x);

DELETE FROM tbl\_potential\_work WHERE keyvalue in (x, x, x);

DELETE FROM tbl\_selection\_rates WHERE keyvalue in (x, x, x);

to delete a single or multiple records or

delete from tbl\_fte\_multipliers

to delete all information contained in the table.

Now you should be ready to import your new data.

Any time new data needs to added or modified you must place the corresponding "CSV" file on the database server. Recommend creating a directory on "/" root called "data" or any place you wish to place your data files that you can access them on the server. Next you will want to call up pgAdmin III and open a "SQL" window. Here you can execute the "copy" command. The "copy" command expects that you specify the table name, data filename, and delimiters (example below).

copy tbl\_fte\_multipliers from '/data/12-fte-multiplier.csv' using delimiters

copy tbl\_potential\_work from '/data/12-potential-work.csv' using delimiters

copy tbl\_selection\_rates from '/data/12-selection-rates.csv' using delimiters

This assumes that the columns in the data file match those of the table name. See above for the table column names, order and data types.

The SQL language can be used to perform further advanced maintenance as required.

The examples below illustrate how to use the terminal tool "psql" as an alternative to pgAdmin III.

First you must access the "psql" application and the "irs" database by using the following.

psql irs

After you have accessed you can use the "copy" command to upload the additional data from a "csv" format. This assumes you have added the "CSV" file to a location on the server as specified above.

copy tbl\_fte\_multipliers from '/data/12-fte-multiplier.csv' using delimiters

This assumes you have placed the raw csv data onto the database server and it accessible on the file system. This also assumes that the data formats match in the csv and the database table. The data samples below identify how the csv should be formatted for database tables prior to import.

For additional information on **psql** http://www.postgresql.org/docs/8.1/static/app-psql.html and **copy** http://www.postgresql.org/docs/8.2/static/sql-copy.html

If additional manipulation is required **psql** or **pgAdmin III** can be utilized to perform more advanced manipulation of the data using **SQL**.

**Process**

When adding new data to environment ensure you follow these steps to help ensure data accuracy.

1. Validate new data against the **data elements** for data accuracy and format
2. **Backup** the existing data environment and **archive**
3. **Copy** the new data to the data environment in the proper location
4. Test and validate the system for accuracy

**Data Samples**

***For example purposes the column names are included in the raw data samples.***

***The column names should not be included in the CSV file that are used during data upload and should be removed prior to executing the copy command.***

**Workforce**

**Workforce** - tbl\_workforce

ageeocy,yoseocy,mandreteocy,retelig\_op,retelig\_early,state,begempcnt,predatt,projyr,grade,series,mco,endempcnt,keyvalue,id,exitbod,exitsched,exitws,exitgrade,exitseries,enterbod,entersched,enterws,entergrade,enterseries,bod,sched 68.741957563,24.832306639,0,1,0,MI,1,0.1747075923,2012,GS12,2210,1,0.8252924077,0,697809220,0,0,0,0,0,0,0,0,0,0,35,FT 69.741957563,25.832306639,0,1,0,MI,0.8252924077,0.1526147845,2013,GS12,2210,1,0.6726776233,0,697809220,0,0,0,0,0,0,0,0,0,0,35,FT 70.741957563,26.832306639,0,1,0,MI,0.6726776233,0.1319280188,2014,GS12,2210,1,0.5407496045,0,697809220,0,0,0,0,0,0,0,0,0,0,35,FT 71.741957563,27.832306639,0,1,0,MI,0.5407496045,0.1126848823,2015,GS12,2210,1,0.4280647222,0,697809220,0,0,0,0,0,0,0,0,0,0,35,FT 72.741957563,28.832306639,0,1,0,MI,0.2140323611,0.04747103,2016,GS12,2210,1,0.1665613311,0,697809220,0,0,0,0.4077440489,0,0,0,0,0,0,35,FT

* ageeocy – double precision
* yoseocy – double precision
* mandreteocy - numeric
* retelig\_op - \_ numeric
* retelig\_early - numeric
* state – character varying
* begempcnt – double precision
* predatt – double precision
* projyr – character varying
* grade – character varying
* series – character varying
* mco - numeric
* endempcnt – double precision
* keyvalue - numeric
* id - numeric
* exitbod – double precision
* exitsched – double precision
* exitws – double precision
* exitgrade – double precision
* exitseries – double precision
* enterbod – double precision
* entersched – double precision
* enterws – double precision
* entergrade – double precision
* enterseries – double precision
* bod - numeric
* sched – character varying

**Workload**

**Potential Work** - tbl\_potential\_work

workstream,keyvalue,worktype,year,returns

CI,19,CI,2011,3376861.25 CI,19,CI,2012,3334313 CI,19,CI,2013,3278922.75 CI,19,CI,2014,3220230 CI,19,CI,2015,3175364.5 ....

* workstream – character varying
* keyvalue – double precision
* worktype – character varying
* year – double precision
* returns – double precision

**FTE Multiplier** - tbl\_fte\_multipliers

workstream,keyvalue,worktype,year,series,multiplier

ACS,1,TDA,2013,0592,2.51808e-05 SC\_EXAM,3,IRTF278,2011,0512,0.000139437 SC\_EXAM,3,IRTF274,2018,0962,5.06666e-07 SC\_EXAM,3,IRTF271,2013,0000,5.25117e-05 FLDEXAM RA,12,BRTF1042,2017,0592,0.001123044 ....

* workstream – character varying
* keyvalue – double precision
* worktype – character varying
* year - integer
* series – character varying
* multiplier – double precision

**Selection Rate** - tbl\_selection\_rate

workstream,keyvalue,worktype,year,selection rate

FLDEXAM TCO,17,BRTF1041,2020,1.32466e-07 FLDEXAM TCO,17,BRTF10651,2020,1.30788e-06 FLDEXAM TCO,17,BRTF1120,2020,0 CI,19,CI,2011,1 ACS,1,TDA,2011,0.1132 ....

* workstream – character varying
* keyvalue – double precision
* worktype – character varying
* year - integer
* selection rate – double precision

**References**

* PostgreSQL Backup and Restore - <http://www.postgresql.org/docs/8.1/static/backup.html>
* PostgreSQL Copy - <http://www.postgresql.org/docs/8.2/static/sql-copy.html>
* pgAdmin III Copy - <http://www.pgadmin.org/docs/1.4/pg/sql-copy.html>
* pgAdmin III - <http://www.pgadmin.org/index.php>
* psql - <http://www.postgresql.org/docs/8.1/static/app-psql.html>
* Google Refine - <http://code.google.com/p/google-refine/>