Interfacing with the FERC Tariff System[1]

This document provides information on how to interpret the Report DB output generated from the FERC Tariff Field software, and discusses plans on designs for importing data into the FERC Tariff database. It is assumed the reader already has familiarity with the FERC Tariff Field software and the general operation of the FERC Tariff system.

1. Interpreting the FERC Tariff data

1.1 Introduction and Background

The FERC Tariff database system is designed as a distributed database. That is, there is a central database at FERC which acts as the central repository and authority for all tariff data. The FERC Tariff Field software is designed to 'download' and manage local subsets of that data (for each company and tariff). This approach allows for better system performance and reliability. Long system maintenance functions could occur at FERC, FERC servers could go down, Internet communications be interrupted, etc, and none of these would prevent users of the Field software from continuing to work with their data. There are some disadvantages of this approach as well, but in terms of the interface, the most problematic is accessing the data.

The database does not allow access from other systems apart from FERC-created software. Past FERC experience has shown that in other distributed database systems, allowing direct access leads to frequent data corruption and invalid, or inconsistent, data. The FERC Tariff system requires a significant amount of validation and checking, and reliance on data stability. Therefore, direct access to the FERC Tariff DB will not be permitted.

However, FERC understands the need for companies to perform various reports and/or interface the FERC Tariff data with other systems. This resulted in the creation of the “Report Database” - an output which the FERC Tariff Field software can generate. The data generated is a set of data that has no access restrictions whatsoever and that can be accessed via industry-standard database access mechanisms. The Report DB contains the information necessary to generate reports and provide data to interfaces with other systems.

[1] If printing this document, a color printer is suggested.

1.2 The Report Database

1.2.1 Summary Database Content

The report database is provided in a standard Visual Foxpro format. The data can be accessed via ODBC standards. The following provide a summary description of each table in the database (the complete detailed database structure is provided as an appendix to this document):

Table name	***Description***
ftr_s0_att_defs	Contains the 'template' list of filing attachments expected to be sent with each filing type.
ftr_s0_codes	Contains various codes and translations to text.
ftr_s0_company	Contains the list companies that file Tariffs with FERC.
ftr_s0_programs	Contains the list of FERC-defined programs, and the general industry to which it belongs.
ftr_s0_sub_defs	Contains the 'rules' for the different filing types that can be filed with FERC.
ftr_filing	Contains the actual filing data of a filing (e.g. point of contact, the filing type of the filing, the company, tariff to which the filing belongs, a unique filing ID for each filing of a tariff, etc).
ftr_filing_attach	Contains the actual attachments of a filing. Note that this attachment list is initially based on the FERC-defined attachments. However, additional items attachments can be added.
ftr_sect_effective	Contains the effective sections of a tariff. Any section that actually became effective will reside in this table. Sections that were rejected, or special purpose (e.g. withdrawals) will only show up in the ftr_sect_working table. Note that there will be 'duplicate' records of accepted sections because accepted records are not deleted from ftr_sect_working. However, in terms of querying, the section status field can be used as desired to show/ignore various records.
ftr_sect_working	Contains 'sections' of a tariff that are being sent with a filing. If these changes are accepted by FERC, the results are copied into the ftr_sect_effective table.
ftr_tariffs	Contains the name of the Tariffs, and the company to which they belong.

Note that tables which contain an '_s0_' (underscore, 's', zero, underscore) are 'system management' tables. These tables do not contain data which users can change. They provide various functions such as code value lookup, codes for enforcing system rules and validation, etc. In the other tables, the following fields are commonly seen:

Field Name	*Description*
respondent_id	This is the company “ID” as assigned in the tariff system. This ID can be used as a lookup into the ftr_s0_company table to determine the textual name of the company.
tariff_id	This is the tariff “ID” for each tariff. Note that this is unique to each company. So, to uniquely identify a specific tariff, the company ID and tariff ID need to be used together.

Next, by far, the most commonly used data will be the data in the *ftr_sect_effective*, and the *ftr_sect_working* tables. These tables contain the actual tariff data (organized in sections). There are additional key fields in these tables as follows:

Field Name	*Description*
section_id	The “ID” of the section in the specific tariff.
filing_id	The “Filing ID” where that specific section content was modified. Note: a filing ID of 1 (a baseline filing) is a special case. See below for more details.
option_letter	This is the option letter within the filing ID which a specific section content modificaiton. So, to uniquely identify a section 'change', the respondent_id, tariff_id, section_id, filing_id, and option_letter are needed.

Note that the *ftr_sect_effective* table contains only sections that have been made 'effective' at some point in time. The exception to this is that during initial tariff creation, while the tariff is non-baselined, the *ftr_sect_effective* table is used to store section data. The reason is for simpler system design and the fact that a baseline tariff request is a 'complete tariff' submission. During baselining, tariffs will either be accepted or rejected in total. Therefore, the *ftr_sect_working* table does not contain sections that are submitted during a baseline filing. So, when the Report DB is created, sections that are in a non-baselined status will be in the *ftr_sect_effective* table.

The *ftr_sect_working* table contains all other section modifications. To specifically identify a section modification, the respondent_id, tariff_id, section_id, filing_id, and option_letter fields are needed.

1.2.2 Tariff Structure

Traditionally, tariffs have been considered by most to be a 'document'. However, a document is very weak in terms of data integrity validation and content management

functions. FERC views a tariff as a database: a database that has text as its core content, but with vital 'metadata' assigned to the text content. Historically, primarily because there was nothing better at the time, tariff data was presented and exchanged in a document format. Therefore, there are key issues regarding the 'structure' of a tariff. That is, how to order and arrange how and where section data should appear when printed, displayed, etc. The system uses 2 main fields to perform structure functions:

Field Name	***Description***
parent_id	The “ID” of the parent section of the current section. A section that at the 'top level' of the structure has a parent ID of 0 (zero).
sect_num	This is numerical sequence for the section. E.g. 1.2, 2.3.4, 1.1.1.1.

The parent_id allows for constructing the hierarchy of sections, the sect_num is used to 'sort' the sections all belonging to the same parent_id.

1.2.3 Status Conditions

One of the most important parts of the whole tariff system is understanding status conditions. The conditions of sections closely match the descriptions found in FERC Regulations text (e.g. 'suspended', 'pending', 'accepted', etc). And there are other status conditions as required by system processing (e.g. 'work in progress'). These conditions are denoted by code values in the database. Note that the *ftr_s0_codes* table provides reference lookups for these codes. The field sect_status is on every record in the *ftr_sect_effective* and *ftr_sect_working* table. Changes in these codes determine what gets copied from *ftr_sect_working* to *ftr_sect_effective*. The currently defined status conditions can be found in the *ftr_s0_codes* table and it is recommended that those codes be carefully reviewed.

1.3 How to detect status changes

In a distributed database system, there needs to be a mechanism to detect and perform the appropriate synchronization. In the case of the FERC Tariff system, the primary means of synchronizing data are time tag values. Due to the nature of the data, a simple date/time field is sufficient (i.e. time recorded down to the second).

1.3.1 The Time-tag fields

The following are the time-tag fields that can be used for detecting data changes:

Table	*Field*	*Notes*
ftr_filing	fercdone	Date/time FERC performed action on the filing
	fielddone	Date/time the corresponding action was taken on the filing by the Field software
	ferc_response_date	The last date/time that FERC made some modification to the filing (or one of it's contained sections).
ftr_sect_working	fercdone	Date/time FERC performed action on section
	fielddone	Date/time the corresponding action was taken on the section by the Field software.

It is important to note that some filings do not contain sections (e.g. a tariff 'cancellation' filing). Therefore, in the simplest of terms, the ferc_response_date in the *ftr_filing* table may be the best time-tag to watch. This time-tag value gets updated whenever something in the filing data, or even in the contained section data, is changed by FERC.

The other time-tag values could then be used in conjunction with other, 3rd party, time-tag values to detect specific items that were changed.

1.3.2. Example use of monitoring status

One reason for wanting to monitor status would be to create an interface to the FERC Tariff system. The concept is the Report DB output could be monitored, and when specific changes are detected, special reports and other functions could be invoked. Additionally, reading the results in the Report DB would be a crucial item for systems that may try providing data back 'into' the FERC Tariff system.

For the sake of brevity, this example will assume Visual Foxpro is the RDBMS used by the external software system. Any RDBMS, or even simplistic database system, could be used – just translate the following to the appropriate terms for the DB system of choice.

1.3.2.1 Creating the interface tables

First, create a table in the external DB system. It is assumed the table is part of a database. The database could be an existing database, or a brand-new stand-alone database. The key consideration is how accessible is the interface table to the external software system. Here is a sample table creation:

```
CREATE TABLE IF_FT_FILINGS ;
        (respondent_id I, ;
        tariff_id I, ;
        filing_id I, ;
        ferc_response_date T, ;
        fercdone T, ;
        prev_response_date T, ;
        prev_fercdone T, ;
        ext_sys_filing_ID)


CREATE TABLE IF_FT_SECTIONS ;
        (respondent_id I, ;
        tariff_id I, ;
        filing_id I, ;
        option_letter C(1), ;
        sect_id I, ;
        fercdone T, ;
        prev_fercdone T, ;
        ext_sys_sect_ID)
```

Of special note are the fields prev_response_date, fercdone, ext_sys_filing_id, prev_fercdone, and ext_sys_sect_id. The reason these fields are important is that the external system will use these fields to detect when a change has been made to a section or a filing, and then be able to 'link' to the data in its own database to perform the desired functions.

Note that additional fields could be retrieved from the Report DB tables if desired. The above intends to only show the minimal necessary fields required to detect that a change in data had occurred. After a change was detected the other information (respondent_id, tariff_id, filing_id, etc) could be used to pull specific data out of the Report DB.

1.3.2.2 Populating the Interface tables for the first time

After the intial interface table creation, data needs to be populated. The following shows how to pull data from the Report DB only. Filling in the external system ID values would be a manual process, or some other process unique to the 3rd party system. The premise is the interface tables are populated with 'initial' data which can then be checked later.

First, the Report DB must be generated. This is a menu option inside the FERC

Tariff system. After that DB has been generated, status-tracking information can be extracted and placed into the interface tables (examples below.

```
*-- open up the interface tables
USE IF_FT_FILINGS IN 0
USE IF_FT_SECTIONS IN 0

*-- pull data out of Report DB, put into temporary cursors
SELECT respondent_id, tariff_id, filing_id, ferc_response_date, fercdone ;
        FROM ftr_filing ;
        INTO CURSOR tempfilingdata NOFILTER
SELECT respondent_id, tariff_id, filing_id, option_letter, sect_id, fercdone ;
        FROM ftr_sect_working ;
        INTO CURSOR tempsectdata

*-- add the data into the interface tables
SELECT IF_FT_FILINGS
APPEND FROM (DBF('tempfilingdata'))
SELECT IF_FT_SECTIONS
APPEND FROM (DBF('tempsectdata'))

*-- note that during the append, fields that don't exist in the incoming data will be filled
*-- with 'default' values. In this example, the default values for integer fields would be
*-- 0, and the default value for date/time would be an 'empty' date (note: this is not a
*-- .null. Value – it is a special VFP value). But the default values could be changed
*-- as desired when the interface tables are created.
```

At this point, the interface tables have the current 'snapshot' of when data in the Report DB was last changed. The next step would be to enter in the external ID links, if appropriate.

1.3.2.3 Detecting a change in Report DB data

Now that the initial interface file has been created, the code to detect a change in status is quite simple.

```
*-- build a list of filings where some change has occurred
SELECT IFFILE.respondent_id, IFFILE.tariff_id, IFFILE.filing_id, ;
                IFFILE.ext_sys_filing_id, RPTFILE.ferc_response_date  ;
        FROM ftr_filing AS RPTFILE, if_ft_filings AS IFFILE ;
        WHERE (IFFILE.respondent_id = RPTFILE.respondent_id AND ;
                IFFILE.tariff_id = RPTFILE.tariff_id AND ;
                IFFILE.filing_id = RPTFILE.filing_id) AND ;
                (NOT EMPTY(RPTFILE.ferc_response_date AND ;
                IFFILE.prev_response_date < RPTFILE.ferc_response_date) ;
        INTO CURSOR tempchangedfilings
```

At this point, the cursor *tempchangefilings* contains the list of all filings in which some change had occurred since the last time the Interface table had been created/updated. At this point, the next logical step would be to step through the temporary cursor, and pull records out of the *ftr_sect_working* table that have changed. For example:

```
SELECT tempchangedfilings
SCAN
        SELECT AA.respondent_id, AA.tariff_id, AA.filing_id, AA.option_letter, ;
                AA.sect_id, AA.fercdone, BB.ext_sys_sect_id ;
            FROM ftr_sect_working AS AA, IF_FT_SECTIONS AS BB ;
            WHERE (AA.filing_id = tempchangedfilings.filing_id) AND ;
                (AA.respondent_id = BB.respondent_id AND ;
                AA.tariff_id = BB.tariff_id AND ;
                AA.option_letter = BB.option_letter AND ;
                AA.sect_id = BB.sect_id) AND ;
                (NOT EMPTY(AA.fercdone) AND ;
                BB.fercdone < AA.fercdone) ;
            INTO CURSOR tempchangedsections

        *-- Here is where the external system may want to do its processing. The above
        *-- creates a cursor (tempchangedsections) that has the needed identifiers to
        *-- pull data out of the Report DB (ftr_sect_working). That data could be pulled
        *-- 1 section at a time, to update other external system tables by use of the
        *-- ext_sys_sect_id.

        <<<external system processing, pull data from ftr_... tables and place into
        corresponding external system tables and fields>>>

        *-- At some point in the code here, the if_ft_sections table and the if_ft_filings
        *-- table should be updated. The prev_fercdone field (in if_ft_sections) should be
        *-- populated with the fercdone (in ftr_sect_working), and the
        *-- prev_response_date field (in if_ft_filings) should be updated with the value
        *-- from ferc_response_date (in ftr_filing). This will mean the next check will
        *-- not pick up those changed sections again (unless they had another update).

ENDSCAN
```

Note that the above is only 1 way to check for, and process, changes. For example, in pulling data from the Report DB, all the fields from the *ftr_*... tables could be retrieved. This would preclude the need to do an additional data pull when a change was detected. Also note that since most filings contain sections, so a jump directly to checking changes between *if_ft_sections* and *ftr_sect_working* could be done. But remember that not all filings have sections and things such as docket number are assigned at the filing level. The above example provides a sort of 'short-circuit' criteria. Overall, the number of filings will be much less than the number of sections contained therein. Therefore, checking the smaller data set first to find just the filings which have changed, could significantly improve performance over a 'brute force' check of all sections every time.

2. Providing data to the FERC Tariff system

The interface for providing data to be imported into the FERC Tariff system has be released as an initial implementation. The basic design is similar to Microsoft Word Import function previously implemented (e.g. A metadata file, and referenced other files).

The following is the structure of the meta data file, with brief explanations.

Here is a sample metadata file structure (standard .dbf format):

Field name	***Data Type/Size***	***Comment***
Rectype	N(3)	All data (filing, section, and attachment) is referred to in this single metadata file. Each record specifically references a single item (e.g. A specific section or file attachment). 1=filing level data (Filing title, POC info, type of filing, etc) (note: the metadata filing should only have a single record with this value) 2=section data (effective date, section number, section title, type of change, associated external file, etc). 3=attachment data (associated external file, comment, matching FERC template ID, etc)
filingtype	N(10)	The filing type ID (from *ftr_s0_sub_defs*) for this filing. This much match with the filing being imported 'into'.
poc_name	C(50)	Point of contact name for the filing (used only by rectypes of '1')
poc_phone	C(20)	POC phone (used only by rectypes of '1')
poc_email	C(100)	POC email (used only by rectypes of '1')
auth_name	C(50)	Authorizing person name (used only by rectypes of '1')
auth_title	C(50)	Title of authorizing person (used only by rectypes of '1')
auth_email	C(100)	Email of authorizing person (used only by rectypes of '1')
x_desc	C(200)	Multiple use (rectypes of '1', '2', and '3'). Title of Filing (rectype '1'), Section Title (rectype '2'), or attachment comment (rectype '3').
sec_id	N(10)	Section ID – the internal section ID of a section. This can be obtained from the Report DB data. For sections that are 'new' (e.g. adding a brand new section as part of the filing which doesn't exist in the tariff currently) – this would be a negative number (e.g. -1, -2, -3 ...). Used only by rectype '2'.
sec_num	C(25)	Section number. This can be obtained from the Report DB data. It's required format is n.n.n.n.n. (where n is a a number from 0 to 999). Used only by rectype '2'.
comp_num	C(40)	Company assigned section number. This can be obtained from the Report DB data. There are no format restrictions on this. Used only by rectype '2'.

Field name	*Data Type/Size*	*Comment*
opt_letter	C(1)	The option letter of the filing to which the section belongs. This is a letter from 'A' to 'Z'. If there is only 1 option in the filing, this should be set to 'A'. Used only by rectype '2'.
spec_type	C(1)	For 'special' types of sections. Currently only the value of 'C' is used – a 'C' denotes the section is the 'title' page (or 'cover' page) of the tariff. Used only by rectype '2'.
sec_parent	N(10)	The parent ID of the section. This can be obtained from the Report DB data. Note that this could be a negative number as well if subsections were added to sections in the filing that did not yet exist in the tariff. Used only by rectype '2'.
sec_chgtyp	C(15)	The type of change to the section. For most cases, this will be “REV SECT”. But for sections that are being newly added, it could be “REV ADD SECT” or “REV ADD SUBSECT”. Lastly, for a section 'cancellation', this could be set to “CANCEL SECT” Used only by rectype '2'.
eff_date	D	This is used by multiple records (rectype's '1' and '2'). For a rectype '1' record (a filing data record), this is used when the filing type is a tariff cancellation filing. This data is set to the date the tariff is to be cancelled. For a rectype of '2', this is the effective date (proposed) of section. If the section change type is a “CANCEL SECT”, this holds the proposed date of section cancellation.
x_linksub	N(10)	Filing number of a 'linked' filing. For example, a motion filing must be associated with a previous filing. This number is the filing number of that filing. For example, Motion, Compliance, Withdraw, etc type filings require a link to an earlier filing. This is for rectype '1'. Also, in section data, there are cases where a section is 'linked' to a specific section in another filing. This field is used in conjunction with the x_linksec and x_linkopt to identify the section referred to. This is for rectype '2'.
x_linksec	N(10)	The section ID of the section linked to (e.g. required for withdraw filings). This field is used in conjunction with the x_linksub and x_linkopt to identify the section referred to. This is used only for rectype '2' records.
x_linkopt	N(10)	The option letter in the filing of the section being linked to (e.g. required for withdraw filings). This field is used in conjunction with the x_linksub and x_linksec to identify the section referred to. This is used only for rectype '2' records.
apply_ord	N(10)	This is the system 'priority' order value. If multiple section changes are proposed to be effective on the same date, this value must be unique in order to determine which section would actually be effective. This is used only for rectype '2' records.

Field name	*Data Type/Size*	*Comment*
user_text	C(10)	This is a field used for miscellaneous user text entry. The system does not process this field in anyway, but users may find it useful to put in text to help them identify certain things about the specific section change (e.g. “Rev4”, etc). This is used only for rectype '2' records.
img_file	L	The system allows a 'file' to be entered with a section as opposed to an actual text (or wordprocessing) document. E.g. grandfathered contracts may be scanned into an image file and then that file placed in the system as a 'section image file' reference. This should be set to .T. (true) if the associated file (listed in x_filename) should be treated as an 'image' file. Note that Image files are meant to serve a specific purpose and are essentially ignored in regards to data analysis, printing, etc. The use of this should probably be limited to only grandfathered contract data. This is used only for rectype '2' records.
f_tmpltid	N(10)	This is the file attachment 'template' ID number. The values of these numbers can be obtained from the Report DB data (i.e. from *ftr_s0_att_defs*). Each filing type has a specific set of FERC-defined file attachments. In order to match up an import file with the right file attachment of the FERC 'template', this ID must match. Not that the comment (description) of file attachments cannot be changed for items that exist in the FERC template. This is used only for rectype '3' records.
f_privcat	C(1)	The privileged setting for the file attachment. This is used only for rectype '3' records. 'P' = public 'B' = public non-Internet 'M' = privileged 'X' = critical energy infrastructure
x_filename	C(80)	The name of the external file to read in for the data. This is used by rectypes '2' and '3'. Note that section records (rectype '2') must be in MS Word format (.doc files). Note: the filename is NOT fully path qualified (e.g. c:\temp\test\import\...) - it is expected the files will reside in the same folder where this metadata file is located. So only the filename and extension should be entered (e.g. 'sectrates1.doc', 'coverlet.txt', ...)
imp_result	C(40)	This is a text area where errors/problems during import will be written. This should be blank before performing the Import process. Those records that didn't pass the validation tests will have messages placed in this field.

Notes:

Some of these were mentioned above.

- All files should reside in the same folder. The metadata file, and every other file referenced by the metadata file.

- It is important to note the rectype field – it determines how the rest of the fields are used during the import process. E.g. if the field is set to '2', the other fields (for that record) are assumed to refer to section type data, the referenced file is a 'section', POC fields are ignored, etc.
- Order of the records is not important (e.g. the rectype '2's may occur prior to the '1' etc). But logically, things flow from filing, to sections of filing, to attachments of filing.
- Some fields serve multiple roles. E.g. the x_desc field holds the filing 'title' for rectype '1'. It holds the the 'section title' for retype '2's. And it holds the file 'comment' for rectype '3's. So, please take careful note of the comments about the metadata above.
- It is important to understand some key items about the section data in the metadata file. Section ID, parent ID, etc are critical in determining placement and structure of the tariff, while the section number (sec_num) is important for ordering/sorting/display purposes. Basically, the section ID (and parent ID) values should not be 'changed' from what exists in the Report DB database. An example below will explain how to add brand new sections into the tariff where the section ID information is not yet known (since it was not yet added into the database).

General process of importing

Here are the steps to perform an import:

1) Create the set of import files (the metadata file and all other referenced files). This might be done manually or through the use of some software system.
2) Create a new filing (the "New" button on the Filing List Tab)
3) Select the newly created Filing in the Filing List tab
4) Right-click and from the pop-up menu, select "Import Filing Data"
5) Follow the prompts (e.g. navigate to the folder where the metadata file is located and select that file, etc)

During the import ETariff Field software will examine the metadata file and the associated files and validate their content against the FERC Tariff database and other importing rules. If all data is valid (e.g. Section/Filing Ids are correct, etc), the data is brought into the system. From that point on, the system would act as if the user had entered the imported data. That is, all validations prior to submission would still be performed, etc.

Discussion of adding new sections as part of a filing

In general, the section ID values should come from the Report DB data. That is, if section 1.1.3.3. is to be modified and put into a filing, the system section ID of that section needs to be determined (as well as it's parent ID). This information is found in the Report DB data (e.g. in the *ftr_sect_effective* table in the *sect_id* and *parent_id* fields).

But there will be cases where new sections need to be added as part of a filing. In this instance the section ID numbers will not be known. There are some key things to do to properly set up the metadata file correctly:

- first, is to make sure the text 'REV ADD SUBSECT' or 'REV ADD SECT' is in the *sec_chgtyp* field (either can be used)
- second, the *sec_id* field should be set to a negative number. Any number will do as long as it is unique in regards to other negative numbers (other new sections) that may be in the filing.
- third, the *sec_parent* field is set correctly. This includes the case where multiple levels of sections are being added (e.g. adding a new section, and then a new section under that, and then a new section under that).

For example, we'll assume that a new section 1.2.3. is going to be added. It never before existed in the tariff and it belongs under section 1.2. Additionally, sections 1.2.3.1. 1.2.3.2. are supposed to be added as well, and they belong under the new 1.2.3. Out of all these sections, only 1.2. currently exists in the tariff. Lets assume the section ID of 1.2. is 455. Here is a subset of the metadata showing how these should be set up:

rectype	***...***	***sec_id***	***sec_num***	***...***	***sec_parent***	***sec_chgtyp***	***...***
...	...	...	...	...	...	...	...
2	...	-1	1.2.3	...	455	REV ADD SECT	...
2	...	-2	1.2.3.1.	...	-1	REV ADD SECT	...
2	...	-3	1.2.3.2.	...	-1	REV ADD SECT	...
...	...	...	...	...	...	...	...

As can be seen in the above example, the 'unknown' section number of 1.2.3. (the "-1") is still appropriate to use for any new subsections being added under it. This is why the *sec_parent* field of 1.2.3.1. and 1.2.3.2. are set to "-1".

Available Utility function:

The ETariff software has a built-in utility function to assist users in understanding and creating import data. In the Tariff Filing window, under the Filing List tab, there is a pop-up menu for the list of filings shown. Right click on an item in the list and a context-sensitive menu appears. Note the functions: "Import Filing Data" and "Create Sample Import Data". The "Import Filing Data" is the function that begins the import process.

The “Create Sample Import Data” will take the currently selected filing and generate a corresponding metadata file and associated files (if there are sections and file attachments defined for the filing). So, the “Create Sample Import Data” can be used on the different filing types to see how the resulting metadata and files would appear.

Summary

The interface may expand and change as time goes on to help accommodate user and other software vendor requests, or to address updated FERC rules and requirements. Feedback with suggestions and questions will be incorporated as appropriate, and this document will continue to be updated as needed.

Appendix A

Summary Report DB descriptions and relationship diagram

Database Information (tables):

ftr_s0_programs
Contains the list of FERC-defined programs, and the general industry to which it belongs.

ftr_s0_company
Contains the list companies that file Tariffs with FERC.

ftr_tariffs
Contains the name of the Tariffs, and the company to which they belong.

ftr_s0_sub_defs
Contains the 'rules' for the different filing types that can be filed with FERC.

ftr_s0_att_defs
Contains the 'template' list of filing attachments expected to be sent with each filing type.

ftr_s0_codes
Contains various codes and translations to text.

ftr_filing
Contains the actual filing data of a filing (e.g. point of contact, the filing type of the filing, the company, tariff to which the filing belongs, a unique filing ID for each filing of a tariff, etc)

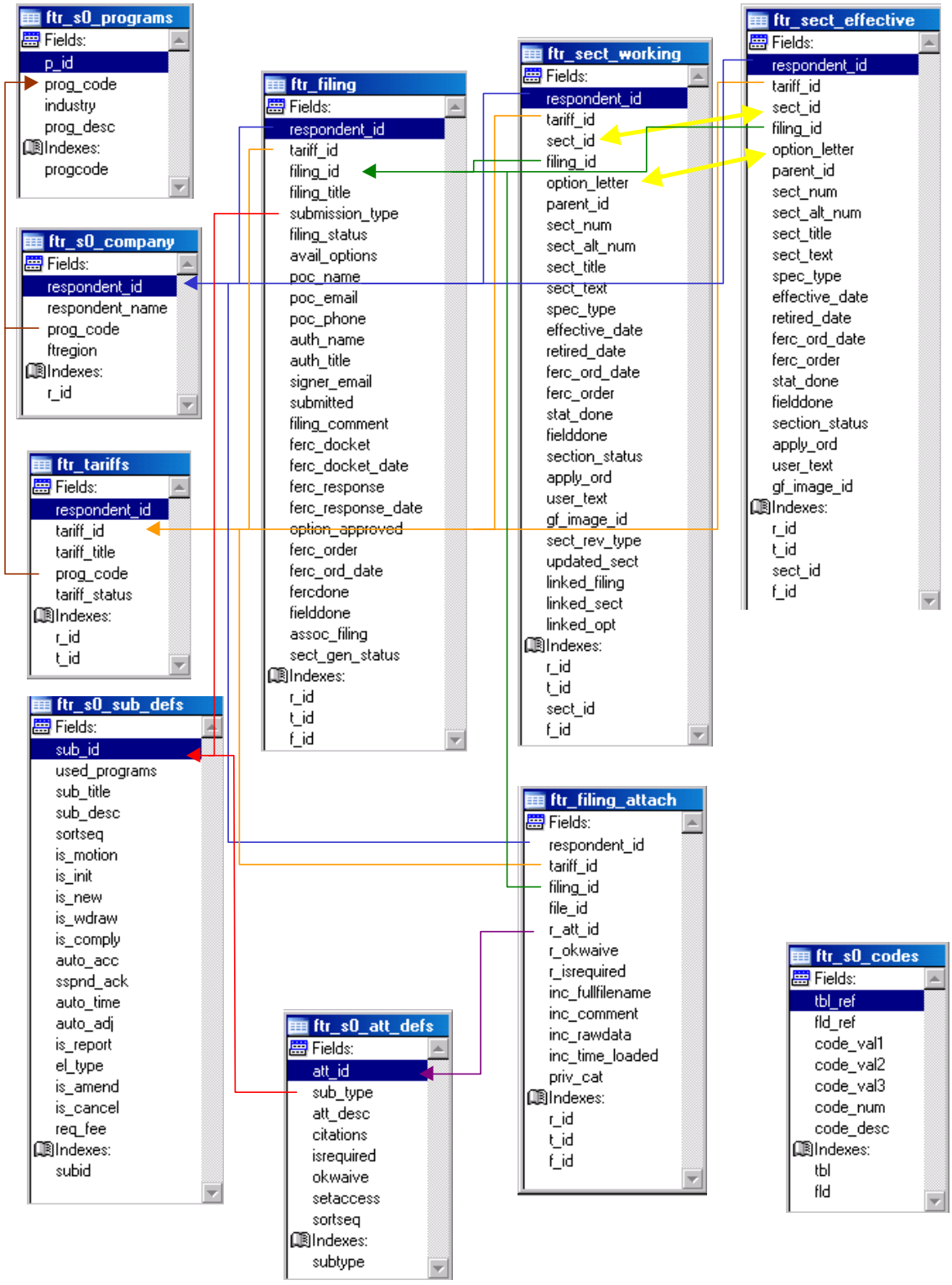
ftr_filing_attach
Contains the actual attachments of a filing. Note that this attachment list is initially based on the FERC-defined attachments. However, additional items attachments can be added.

ftr_sect_working
Contains 'sections' of a tariff that are being sent with a filing. If these changes are accepted by FERC, the results are copied into the ftr_sect_effective table.

ftr_sect_effective
Contains the effective sections of a tariff. Any section that actually became effective will reside in this table. Sections that were rejected, or special purpose (e.g. withdrawals) will only show up in the ftr_sect_working table. Note that there will be 'duplicate' records of accepted sections because accepted records are not deleted from ftr_sect_working. However, in terms of querying, the section status field can be used as desired to show/ignore various records.

Miscellaneous:

Note that the report DB does not have any specific restrictions (e.g. foreign keys defined, triggers, security, etc). This is intentional so as to allow other software or access data as desired. However, the diagram on the next page shows the general data relationships.


ftr_s0_programs
Fields:
p_id
prog_code
industry
prog_desc
Indexes:
progcode
ftr_s0_company
Fields:
respondent_id
respondent_name
prog_code
ftregion
Indexes:
r_id
ftr_tariffs
Fields:
respondent_id
tariff_id
tariff_title
prog_code
tariff_status
Indexes:
r_id
t_id
ftr_s0_sub_defs
Fields:
sub_id
used_programs
sub_title
sub_desc
sortseq
is_motion
is_init
is_new
is_wdraw
is_comply
auto_acc
sspnd_ack
auto_time
auto_adj
is_report
el_type
is_amend
is_cancel
req_fee
Indexes:
subid
ftr_filing
Fields:
respondent_id
tariff_id
filing_id
filing_title
submission_type
filing_status
avail_options
poc_name
poc_email
poc_phone
auth_name
auth_title
signer_email
submitted
filing_comment
ferc_docket
ferc_docket_date
ferc_response
ferc_response_date
option_approved
ferc_order
ferc_ord_date
fercdone
fielddone
assoc_filing
sect_gen_status
Indexes:
r_id
t_id
f_id
ftr_s0_att_defs
Fields:
att_id
sub_type
att_desc
citations
isrequired
okwaive
setaccess
sortseq
Indexes:
subtype
ftr_sect_working
Fields:
respondent_id
tariff_id
sect_id
filing_id
option_letter
parent_id
sect_num
sect_alt_num
sect_title
sect_text
spec_type
effective_date
retired_date
ferc_ord_date
ferc_order
stat_done
fielddone
section_status
apply_ord
user_text
gf_image_id
sect_rev_type
updated_sect
linked_filing
linked_sect
linked_opt
Indexes:
r_id
t_id
sect_id
f_id
ftr_filing_attach
Fields:
respondent_id
tariff_id
filing_id
file_id
r_att_id
r_okwaive
r_isrequired
inc_fullfilename
inc_comment
inc_rawdata
inc_time_loaded
priv_cat
Indexes:
r_id
t_id
f_id
ftr_sect_effective
Fields:
respondent_id
tariff_id
sect_id
filing_id
option_letter
parent_id
sect_num
sect_alt_num
sect_title
sect_text
spec_type
effective_date
retired_date
ferc_ord_date
ferc_order
stat_done
fielddone
section_status
apply_ord
user_text
gf_image_id
Indexes:
r_id
t_id
sect_id
f_id
ftr_s0_codes
Fields:
tbl_ref
fld_ref
code_val1
code_val2
code_val3
code_num
code_desc
Indexes:
tbl
fld

Appendix B

Detailed report database structure

Table: *FTR_FILING*

Field	*Data Type*	*Size*		*Comment*
RESPONDENT_ID	I	4	0	System Company ID
TARIFF_ID	I	4	0	System ID for Tariff
FILING_ID	I	4	0	System generated ID of the filing
FILING_TITLE	C	80	0	The user-entered title of the filing
SUBMISSION_TYPE	I	4	0	From ft_s0_sub_types - sub_id
FILING_STATUS	N	10	0	The overall status code of the filing
AVAIL_OPTIONS	C	20	0	The option letters (sub sets) of a filing
POC_NAME	C	50	0	POC for filing
POC_EMAIL	C	100	0	POC email
POC_PHONE	C	20	0	POC phone
AUTH_NAME	C	50	0	Authorizing person
AUTH_TITLE	C	50	0	Authorizor title
SIGNER_EMAIL	C	100	0	Signatory email - must be eReg'd with FERC
SUBMITTED	T	8	0	Time submitted (filed) to FERC
FILING_COMMENT	M	4	0	Holding place for comments
FERC_DOCKET	C	80	0	FERC-assigned Docket
FERC_DOCKET_DATE	T	8	0	Time docket assigned
FERC_RESPONSE	C	15	0	General FERC response (text)
FERC_RESPONSE_DATE	T	8	0	Latest time of FERC change to anything in filing
OPTION_APPROVED	C	1	0	Option letter approved (only 1 opt of a filing can be accepted - others rejected)
FERC_ORDER	C	20	0	FERC-entered FERC Order
FERC_ORD_DATE	T	8	0	The date FERC finalized the decision on the filing (if the action was taken on filing in total)
FERCDONE	T	8	0	Latest time of FERC change to anything in filing
FIELDDONE	T	8	0	time when status was applied in Field Software
ASSOC_FILING	I	4	0	The linked filing ID (if applicable)
SECT_GEN_STATUS	C	2	0	summary status of sections - ' ' sections not acted on, 'P' partial, 'C' complete, 'S' suspends exist

Table: *FTR_FILING_ATTACH*

Field	Data Type	Size		Comment
RESPONDENT_ID	I	4	0	System Company ID
TARIFF_ID	I	4	0	System ID for Tariff
FILING_ID	I	4	0	System generated ID of a filing
FILE_ID	I	4	0	Sys unique file ID
R_ATT_ID	I	4	0	Related attach ID from s0_att_defs table will be 0 for those added by user
R_OKWAIVE	L	1	0	associated flag from att_defs if OK to waive
R_ISREQUIRED	L	1	0	from associated s0_att_defs - if attachment is required
INC_FULLFILENAME	C	150	0	path-qualified orig filename
INC_COMMENT	C	150	0	Desc/title of file - can be entered by user in some cases
INC_RAWDATA	M	4	0	The 'file image'
INC_TIME_LOADED	T	8	0	Time file was 'loaded' into database
PRIV_CAT	C	1	0	Security level (public, non-Internet...)

Table: *FTR_S0_ATT_DEFS*

Field	Data Type	Size		Comment
ATT_ID	I	4	0	system generated ID
SUB_TYPE	I	4	0	The filing type (ID) to which this attachment belongs
ATT_DESC	M	4	0	Description of the attachment
CITATIONS	C	50	0	Related regulation citations
ISREQUIRED	L	1	0	.T. if an attachment file is required for this item.
OKWAIVE	L	1	0	.T. if a waiver request can be entered as the 'file' for this item
SETACCESS	C	1	0	The required access level of the attachment.
SORTSEQ	C	20	0	Optional sort sequence for ordering during display

Table: *FTR_S0_CODES*

Field	Data Type	Size		Comment
TBL_REF	C	32	0	reference to table where code is used
FLD_REF	C	32	0	reference to field name where code is used
CODE_VAL1	C	32	0	Character code value
CODE_VAL2	C	32	0	Optional second code value
CODE_VAL3	C	10	0	Optional 3rd code value
CODE_NUM	I	4	0	Integer code value
CODE_DESC	C	100	0	description of code meaning

Table: *FTR_S0_COMPANY*

Field	Data Type	Size		Comment
RESPONDENT_ID	I	4	0	Sys ID of company
RESPONDENT_NAME	C	80	0	Company name
PROG_CODE	C	1	0	From ft_s0_programs
FTREGION	C	7	0	FERC assigned Region. Used with prog_code - email notifications

Table: *FTR_S0_PROGRAMS*

Field	Data Type	Size		Comment
P_ID	I	4	0	The program ID (system generated)
PROG_CODE	C	1	0	1-Character code for the FERC program
INDUSTRY	I	4	0	1-Electric, 2-Gas, 3-Oil
PROG_DESC	C	50	0	Description of the Program

Table: *FTR_S0_SUB_DEFS*

Field	Data Type	Size		Comment
SUB_ID	I	4	0	System generated ID for the filing type
USED_PROGRAMS	C	10	0	Programs that use the filing
SUB_TITLE	C	150	0	The title of the filing type
SUB_DESC	M	4	0	Longer description of the filing type
SORTSEQ	C	20	0	option sort sequence field for ordering during display
IS_MOTION	L	1	0	.T. if a motion type filing
IS_INIT	L	1	0	.T. if an initial baseline filing
IS_NEW	L	1	0	.T. if a 'new' baseline filing
IS_WDRAW	L	1	0	.T. if a withdrawal type filing
IS_COMPLY	L	1	0	.T. if a compliance type filing
AUTO_ACC	L	1	0	.T. if filing automatically goes into conditional acceptance
SSPND_ACK	L	1	0	.T. if user must specify whether or not suspended sections automatically go into effect
AUTO_TIME	I	4	0	the time out value (days) before status automatically changes
AUTO_ADJ	I	4	0	
IS_REPORT	L	1	0	.T. if report type filing
EL_TYPE	I	4	0	ELibrary type for filing (FERC use)
IS_AMEND	L	1	0	.T. if an amendment type filing
IS_CANCEL	L	1	0	.T. if a cancellation type filing
REQ_FEE	L	1	0	.T. if a filing fee is required for the filing

Table: *FTR_SECT_EFFECTIVE*

Field	Data Type	Size		Comment
RESPONDENT_ID	I	4	0	System Company ID
TARIFF_ID	I	4	0	System ID for Tariff
SECT_ID	I	4	0	System section ID
FILING_ID	I	4	0	The system generated ID of a filing
OPTION_LETTER	C	1	0	The option letter in a filing
PARENT_ID	I	4	0	Tariff structure - 'parent' section ID
SECT_NUM	C	25	0	User-entered section number
SECT_ALT_NUM	C	40	0	User-entered alternative section number
SECT_TITLE	M	4	0	User-entered section title
SECT_TEXT	M	4	0	Binary 'file image' of section text (Tx Text OCX format)
SPEC_TYPE	C	1	0	blank - normal section, 'C' - cover page
EFFECTIVE_DATE	D	8	0	User-entered effective date
RETIRED_DATE	D	8	0	Date when superceded (by section change)
FERC_ORD_DATE	T	8	0	The date FERC finalized their decision on the section modification
FERC_ORDER	C	20	0	FERC-entered FERC Order
STAT_DONE	T	8	0	time when status was last updated by FERC
FIELDDONE	T	8	0	time when status was applied in Field Software
SECTION_STATUS	N	10	0	The status code of this specific section record (see lookup codes)
APPLY_ORD	I	4	0	order to apply sect changes - sects may be made effective on same date
USER_TEXT	C	10	0	Free-form text for additional identification use by users.
GF_IMAGE_ID	I	4	0	The system ID of a linked 'image' file.

Table: *FTR_SECT_WORKING*

Field	Data Type	Size		Comment
RESPONDENT_ID	I	4	0	System Company ID
TARIFF_ID	I	4	0	System ID for Tariff
SECT_ID	I	4	0	System section ID
FILING_ID	I	4	0	The system-generated ID of a filing
OPTION_LETTER	C	1	0	option letter (filing sub-set)
PARENT_ID	I	4	0	Tariff structure - 'parent' section ID
SECT_NUM	C	25	0	User-entered section number
SECT_ALT_NUM	C	40	0	User-entered alternative section number
SECT_TITLE	M	4	0	User-entered section title

SECT_TEXT	M	4	0	Binary 'file image' of section text (Tx Text OCX format)
SPEC_TYPE	C	1	0	blank - normal section, 'C' - cover page
EFFECTIVE_DATE	D	8	0	User-entered effective date
RETIRED_DATE	D	8	0	Date when superceded (by section change)
FERC_ORD_DATE	T	8	0	The date FERC finalized the decision on the section
FERC_ORDER	C	20	0	FERC-entered FERC Order
STAT_DONE	T	8	0	time when status was last updated by FERC
FIELDDONE	T	8	0	time when status was applied in Field Software
SECTION_STATUS	N	10	0	The status code of the section
APPLY_ORD	I	4	0	order to apply sect changes - sects may be made effective on same date
USER_TEXT	C	10	0	User-entered text (misc descriptive?). Not used by software
GF_IMAGE_ID	I	4	0	The system ID of a linked 'image' file.
SECT_REV_TYPE	C	15	0	Sys - desc - general filing type
UPDATED_SECT	T	8	0	set when approved sect has been noted in FT_APPROVED- blank otherwise
LINKED_FILING	I	4	0	Filing ID that is linked to this filing
LINKED_SECT	I	4	0	Sys sect ID of linked sect (if applicable)
LINKED_OPT	C	1	0	Linked option letter of linked sect (if applicable)

Table: *FTR_TARIFFS*

Field	***Data Type***	***Size***		***Comment***
RESPONDENT_ID	I	4	0	Sys Company ID
TARIFF_ID	I	4	0	Sys Tariff ID
TARIFF_TITLE	C	120	0	User-entered Tariff Title
PROG_CODE	C	1	0	FERC-specified Program Code (company program)
TARIFF_STATUS	N	10	0	Tariff status (e.g. Code for non-baselined, baselined,...)