FactSet Research_DB Parser

This is the manual for Div_RD FactSet parser program. The work is organized as follows:

Parsers and work environment

Environment

in order to let the parsers work, the folders should match the layout

```
.-- root folder
    |-- Div_RD_parser.pl
    |-- Div_RD_dsidmap.pl
    |-- Source_Data
    |   |-- Div_RD_(\w+).txt
    |-- Working_Folder
    |   |-- temp_output.csv (auto generated)
    |   |-- Mappingtable_
    |       |-- Parsermapping.csv
    |       |-- DSIDmapping.csv
    |   |-- Log
    |       |-- Div_RD_dsidmapping.log (auto generated)
    |       |-- Div_RD_parser.log (auto generated)
    |-- Results
    |   |-- Data_for_importer.csv (auto generated)
    |-- Docs (optional)
```

Parsers

There are two perl parsers in the root folder:

1. **Div_RD_parser.pl**
    - calculate aggregations and other statistical analysis on FactSet raw data files
    - generate temp_output files with fields "parserId, date, value", which will be the input of Div_RD_disdmap.pl
2. **Div_RD_disidmap.pl**
    - take temp_output generated from Div_RD_parser.pl as input, map each parserId

to its corresponding DSID in research database.

- generate final output file (Data_for_importer.csv), which has the format "DSID,Date,Value"

Parsers logic

The parsers' work flow can be summarized as below:[1]

```
>-- Call Div_RD_parser.pl
|   |
|   V-- Look up each Div_RD_(\w+).txt in ~/Source_Data folder
|   |
|   V-- parse each file and calculate the factor results
|   |
|   V-- open the mapping file "Parsermapping.csv" and map each factor
characteristic to a
|       parserID in the mapping file and use that as the id in output file
|   |
|   V-- generate temp_output.csv file in Working_Folder and append the parsed
results from
|       each file into that csv file.
|
V
>-- Call Div_RD_dsidmap.pl
|   |
|   v-- take the "temp_output.csv" generated by Div_RD_parser.pl as input
|   |
|   V-- mapping each parseID to its DSID using "DSIDmapping.csv" in
|       Working_Folder/Mappingtable
|   |
|   V-- generate "Data_for_importer.csv" in ~/Results folder, with each
|       parserID from temp_output.csv been replaced with DSID
|
|>> return 0 upon success, o/w return -1
```

Result files

Source file name format[2]

the file that comes out of FactSet named in a format of:

```
Div_RD_GroupID_SutdyName.txt
```

for example:

```
Div_RD_1_SummitNegEarning.txt
```

in the example above:

- **Div_RD**
  This is the universal title for Diversified team Research Database FactSet Download files. Any file start with `Div_RD` should be parsed and load into database
- **GroupID : 1**
  This groupid helps categorize a series of studies which belongs to the same project. also this grouid will also be used search for the correct parser for this file. Currently, `all the files have the same groupid are parsed by the same parser`
- **SutdyName : SummitNegEarning**
  In this case the studyname is `SummitNegEarning` . this study is a part of summit report project that runs monthly and this specific study is for NegEarning factor.
- **FactorName : NegEarnings**
  As introduced above, the FactorName is NegEarnings in this case and this is the only factor in this study. However, the number of factors in each study can definitely be more than 1.

Parsed Results Format

Each row in parsed results(temp_output.csv) follows the format:

```
GroupID_SutdyName_FactorName_Fractile_DataItems,Date,value
```

for example:

```
Div_RD_1_SummitNegEarning_NegEarning_1_ret, 11/01/2012, 0.013
```

in the example above:

GroupID_SutdyName_FactorName_Items is the `ParseID` of this specific record. It maps to a unique DSID in research database, where the Value part should be stored into.

- **GroupID : 1**
  This groupid helps categorize a series of studies which belongs to the same project. also this groupid will also be used search for the correct parser for this file. Currently, `all the files have the same groupid are parsed by the same parser`
- **SutdyName : SummitNegEarning**
  In this case the studyname is `SummitNegEarning` . this study is a part of summit report project that runs monthly and this specific study is for NegEarning factor.
- **FactorName : NegEarnings**
  As introduced above, the FactorName is NegEarnings in this case and this is the only factor in this study. However, the number of factors in each study can definitely be more than 1.
- **Fractile : 1**
  Fractile tells which fractile(1,2,3,4,5,etc) of this factor is presented in this record. In the example, we know that this row stores the data for factors 'Quartile 1' data.[3]
- **DataItems : ret**
  The DataItems stores what kind of calculation was performed on this factor. In this example, we know that the `return` was calculated. Items can also be stat data like `min,median,max,average etc` .
- **Date : 11/01/2012**
  The Date is in the format of %m/%d/%Y
- **Value : 0.013**
  This is the calculated value of the DataItem introduced above.

After calling DSIDmapping.pl, in "Data_for_importer.csv", the final output follows the format:

```
DSID,Date,value
12,11/01/2012,0.013
```

Later, this file will be used as input for the Research database importer.

---

1. log files are stored in ~/Working_Folder/Log/ folder. ↩
2. Items are separated by _ , when define studies, it should not contain any other _ sign in the name, otherwise it will mislead the parser.↩
3. User will not directly know the number of fractiles, This is only used for mapping to DSID. To check the fractile property, please go directly to research database UI and view the tags there.↩