edgar_xbrl

September 29, 2021

1 Working with filing data from the SEC's EDGAR service

[1]:
```
import warnings
warnings.filterwarnings('ignore')
```

[2]:
```
%matplotlib inline

from pathlib import Path
from datetime import date
import json
from io import BytesIO
from zipfile import ZipFile, BadZipFile
from tqdm import tqdm
import requests

import pandas_datareader.data as web
import pandas as pd

from pprint import pprint

import seaborn as sns
import matplotlib.pyplot as plt
import matplotlib.ticker as mticker
```

[3]:
```
sns.set_style('whitegrid')
```

[4]:
```
# store data in this directory since we won't use it in other chapters
data_path = Path('data') # perhaps set to external harddrive to accomodate␣
 ↪large amount of data
if not data_path.exists():
    data_path.mkdir()
```

1.1 Download FS & Notes Data

The following code downloads and extracts all historical filings contained in the Financial Statement and Notes (FSN) datasets from Q1/2014 through Q3/2020.

The SEC has moved to a monthly cadence after Q3/2020; feel free to extend the code by creating the correpsonding file names (see linked website) and download those as well.

Downloads over 40GB of data!

[5]:

```
SEC_URL = 'https://www.sec.gov/'
FSN_PATH = 'files/dera/data/financial-statement-and-notes-data-sets/'
```

[6]:

```
filing_periods = [(d.year, d.quarter) for d in pd.date_range('2014',␣
 ↪'2020-09-30', freq='Q')]
filing_periods
```

[6]:

```
[(2014, 1),
 (2014, 2),
 (2014, 3),
 (2014, 4),
 (2015, 1),
 (2015, 2),
 (2015, 3),
 (2015, 4),
 (2016, 1),
 (2016, 2),
 (2016, 3),
 (2016, 4),
 (2017, 1),
 (2017, 2),
 (2017, 3),
 (2017, 4),
 (2018, 1),
 (2018, 2),
 (2018, 3),
 (2018, 4),
 (2019, 1),
 (2019, 2),
 (2019, 3),
 (2019, 4),
 (2020, 1),
 (2020, 2),
 (2020, 3)]
```

[7]:

```
for yr, qtr in tqdm(filing_periods):
    # set (and create) directory
    path = data_path / f'{yr}_{qtr}' / 'source'
    if not path.exists():
        path.mkdir(parents=True)

    # define url and get file
    filing = f'{yr}q{qtr}_notes.zip'
    url = SEC_URL + FSN_PATH + filing
    response = requests.get(url).content

    # decompress and save
    try:
        with ZipFile(BytesIO(response)) as zip_file:
            for file in zip_file.namelist():
                local_file = path / file
                if local_file.exists():
                    continue
                with local_file.open('wb') as output:
                    for line in zip_file.open(file).readlines():
                        output.write(line)
    except BadZipFile:
        print(f'\nBad zip file: {yr} {qtr}\n')
        continue
```

```
100%|      | 27/27 [20:04<00:00, 44.61s/it]
```

1.2 Save to parquet

The data is fairly large and to enable faster access than the original text files permit, it is better to convert the text files to binary, columnar parquet format (see Section ‘Efficient data storage with pandas’ in chapter 2 for a performance comparison of various data-storage options compatible with pandas DataFrames):

> Some fo the `txt.tsv` source files contain a small number of faulty lines; the code below drops those lines but indicates the line numbers where you can find the errors if you would like to investigate further.

[21]:
```
for f in tqdm(sorted(list(data_path.glob('**/*.tsv')))):
    # set (and create) directory
    parquet_path = f.parent.parent / 'parquet'
    if not parquet_path.exists():
        parquet_path.mkdir(parents=True)

    # write content to .parquet
    file_name = f.stem  + '.parquet'
    if not (parquet_path / file_name).exists():
        try:
            df = pd.read_csv(f, sep='\t', encoding='latin1', low_memory=False,␣
 ↪error_bad_lines=False)
            df.to_parquet(parquet_path / file_name)
        except Exception as e:
            print(e, ' | ', f)
        # optional: uncomment to delete original .tsv
#         else:
```

```
            # f.unlink
```

```
  3%|          | 7/216 [00:27<11:21,  3.26s/it]b'Skipping line 153605: expected
20 fields, saw 22\nSkipping line 153606: expected 20 fields, saw 21\nSkipping
line 316753: expected 20 fields, saw 23\nSkipping line 324401: expected 20
fields, saw 28\nSkipping line 469248: expected 20 fields, saw 22\nSkipping line
509941: expected 20 fields, saw 22\n'
  7%|          | 15/216 [00:57<09:00,  2.69s/it]b'Skipping line 97778: expected
20 fields, saw 21\nSkipping line 101028: expected 20 fields, saw 23\nSkipping
line 154935: expected 20 fields, saw 28\nSkipping line 162785: expected 20
fields, saw 23\nSkipping line 167447: expected 20 fields, saw 23\nSkipping line
193011: expected 20 fields, saw 21\nSkipping line 193012: expected 20 fields,
saw 21\nSkipping line 193013: expected 20 fields, saw 21\nSkipping line 193014:
expected 20 fields, saw 21\nSkipping line 370048: expected 20 fields, saw
21\nSkipping line 370049: expected 20 fields, saw 21\nSkipping line 370050:
expected 20 fields, saw 21\n'
 11%|          | 23/216 [01:24<08:06,  2.52s/it]

No columns to parse from file  |  data/2014_3/source/txt.tsv

 47%|        | 101/216 [01:59<02:10,  1.13s/it]b'Skipping line 838521:
expected 9 fields, saw 10\n'
 51%|        | 110/216 [02:30<04:04,  2.31s/it]b'Skipping line 480899:
expected 9 fields, saw 10\n'
 58%|       | 125/216 [03:21<05:11,  3.42s/it]b'Skipping line 91798: expected
9 fields, saw 10\n'
 81%|      | 174/216 [06:38<02:14,  3.20s/it]b'Skipping line 69252: expected
9 fields, saw 10\nSkipping line 582206: expected 9 fields, saw 10\nSkipping line
607590: expected 9 fields, saw 10\n'
 84%|      | 181/216 [07:08<02:25,  4.17s/it]b'Skipping line 374579:
expected 9 fields, saw 10\n'
 92%|      | 198/216 [08:15<01:08,  3.83s/it]b'Skipping line 754776:
expected 9 fields, saw 10\n'
 95%|      | 206/216 [08:50<00:33,  3.30s/it]b'Skipping line 530291:
expected 9 fields, saw 10\n'
 99%|      | 214/216 [09:21<00:05,  2.95s/it]b'Skipping line 602232:
expected 9 fields, saw 10\n'
100%|      | 216/216 [09:30<00:00,  2.64s/it]
```

1.3 Metadata json

[22]:
```
file = data_path / '2018_3' / 'source' / '2018q3_notes-metadata.json'
with file.open() as f:
    data = json.load(f)

pprint(data)
```

```
{'@context': 'http://www.w3.org/ns/csvw',
'dialect': {'delimiter': '\t', 'header': True, 'headerRowCount': 1},
'tables': [{'tableSchema': {'aboutUrl': 'readme.htm',
                            'columns': [{'datatype': {'base': 'string',
                                                      'maxLength': 20,
                                                      'minLength': 20},
                                         'dc:description': 'Accession Number. '
                                                           'The 20-character '
                                                           'string formed '
                                                           'from the 18-digit '
                                                           'number assigned '
                                                           'by the Commission '
                                                           'to each EDGAR '
                                                           'submission.',
                                         'name': 'adsh',
                                         'required': 'true',
                                         'titles': ['Accession Number']},
                                        {'datatype': {'base': 'decimal',
                                                      'maxLength': 10,
                                                      'minInclusive': 0},
                                         'dc:description': 'Central Index Key '
                                                           '(CIK). Ten digit '
                                                           'number assigned '
                                                           'by the Commission '
                                                           'to each '
                                                           'registrant that '
                                                           'submits filings.',
                                         'name': 'cik',
                                         'titles': ['Central Index Key']},
                                        {'datatype': {'base': 'string',
                                                      'maxLength': 150},
                                         'dc:description': 'Name of '
                                                           'registrant. This '
                                                           'corresponds to '
                                                           'the name of the '
                                                           'legal entity as '
                                                           'recorded in EDGAR '
                                                           'as of the filing '
                                                           'date.',
                                         'name': 'name',
                                         'titles': ['Registrant']},
                                        {'datatype': {'base': 'string',
                                                      'maxLength': 4},
                                         'dc:description': 'Standard '
                                                           'Industrial '
                                                           'Classification '
                                                           '(SIC). Four digit '
                                                           'code assigned by '
                                                           'the Commission as '
```

```
                   'of the filing '
                   'date, indicating '
                   "the registrant's "
                   'type of business.',
 'name': 'sic',
 'titles': ['Standard Industrial '
            'Classification Code']},
{'datatype': {'base': 'string',
              'maxLength': 2,
              'minLength': 2},
 'dc:description': 'The ISO 3166-1 '
                   'country of the '
                   "registrant's "
                   'business address.',
 'name': 'countryba',
 'titles': ['Business Address Country',
            'Country (B)']},
{'datatype': {'base': 'string',
              'maxLength': 2,
              'minLength': 2},
 'dc:description': 'The state or '
                   'province of the '
                   "registrant's "
                   'business address, '
                   'if field '
                   'countryba is US '
                   'or CA.',
 'name': 'stprba',
 'titles': ['Business Address State '
            'or Province',
            'State (B)']},
{'datatype': {'base': 'string',
              'maxLength': 30},
 'dc:description': 'The city of the '
                   "registrant's "
                   'business address.',
 'name': 'cityba',
 'titles': ['Business Address City',
            'City (B)']},
{'datatype': {'base': 'string',
              'maxLength': 10},
 'dc:description': 'The zip code of '
                   "the registrant's "
                   'business address.',
 'name': 'zipba',
 'titles': ['Business Address Zip or '
            'Postal Code',
            'Zip (B)']},
{'datatype': {'base': 'string',
              'maxLength': 40},
 'dc:description': 'The first line of '
                   'the street of the '
                   "registrant's "
                   'business address.',
 'name': 'bas1',
 'titles': ['Business Address Street '
            '1',
            'Street1 (B)']},
{'datatype': {'base': 'string',
              'maxLength': 40},
 'dc:description': 'The second line '
                   'of the street of '
                   "the registrant's "
                   'business address.',
 'name': 'bas2',
 'titles': ['Business Address Street '
            '2',
            'Street2 (B)']},
{'datatype': {'base': 'string',
              'maxLength': 20},
 'dc:description': 'The phone number '
                   'of the '
                   "registrant's "
                   'business address.',
 'name': 'baph',
 'titles': ['Business Address Phone',
            'Phone (B)']},
{'datatype': {'base': 'string',
              'maxLength': 2,
              'minLength': 2},
 'dc:description': 'The ISO 3166-1 '
                   'country of the '
                   "registrant's "
                   'mailing address.',
 'name': 'countryma',
 'titles': ['Mailing Address Country',
            'Country (M)']},
{'datatype': {'base': 'string',
              'maxLength': 2,
              'minLength': 2},
 'dc:description': 'The state or '
                   'province of the '
                   "registrant's "
                   'mailing address, '
                   'if field '
                   'countryma is US '
                   'or CA.',
 'name': 'stprma',
 'titles': ['Mailing Address State or '
            'Province',
            'State (M)']},
{'datatype': {'base': 'string',
              'maxLength': 30},
 'dc:description': 'The city of the '
                   "registrant's "
                   'mailing address.',
 'name': 'cityma',
 'titles': ['Mailing Address City',
            'City (M)']},
{'datatype': {'base': 'string',
              'maxLength': 12},
 'dc:description': 'The zip code of '
                   "the registrant's "
                   'mailing address.',
 'name': 'zipma',
 'titles': ['Mailing Address Zip or '
            'Postal Code',
            'Zip (M)']},
{'datatype': {'base': 'string',
              'maxLength': 40},
 'dc:description': 'The first line of '
                   'the street of the '
                   "registrant's "
                   'mailing address.',
 'name': 'mas1',
 'titles': ['Mailing Address Street1',
            'Street1 (M)']},
{'datatype': {'base': 'string',
              'maxLength': 40},
 'dc:description': 'The second line '
                   'of the street of '
                   "the registrant's "
                   'mailing address.',
 'name': 'mas2',
 'titles': ['Mailing Address Street2',
            'Street1 (M)']},
{'datatype': {'base': 'string',
              'maxLength': 2,
              'minLength': 2},
 'dc:description': 'The country of '
                   'incorporation for '
                   'the registrant.',
 'name': 'countryinc',
 'titles': ['Country of Incorporation',
            'Incorporation Country']},
{'datatype': {'base': 'string',
              'maxLength': 2,
              'minLength': 2},
 'dc:description': 'The state or '
                   'province of '
                   'incorporation for '
                   'the registrant, '
                   'if countryinc is '
                   'US or CA, '
                   'otherwise NULL.',
 'name': 'stprinc',
 'titles': ['State or Province of '
            'Incorporation',
            'Incorporation State']},
{'datatype': {'base': 'string',
              'maxLength': 9},
 'dc:description': 'Employee '
                   'Identification '
                   'Number, 9 digit '
                   'identification '
                   'number assigned '
                   'by the Internal '
                   'Revenue Service '
                   'to business '
                   'entities '
                   'operating in the '
                   'United States.',
 'name': 'ein',
 'titles': ['EIN',
            'Employee Identification '
            'Number']},
{'datatype': {'base': 'string',
              'maxLength': 150},
 'dc:description': 'Most recent '
                   'former name of '
                   'the registrant, '
                   'if any.',
 'name': 'former',
 'titles': ['Former Name']},
{'datatype': {'base': 'string',
              'maxLength': 8,
              'minLength': 8},
 'dc:description': 'Date of change '
                   'from the former '
                   'name, if any.',
 'name': 'changed',
 'titles': ['Date of Name Change']},
                                         {'datatype': {'base': 'string',
                                                       'maxLength': 5},
                                          'dc:description': 'Filer status with '
                                                            'the Commission at '
                                                            'the time of '
                                                            'submission: '
                                                            '1-LAF=Large '
                                                            'Accelerated, '
                                                            '2-ACC=Accelerated,
'
                                                            '3-SRA=Smaller '
                                                            'Reporting '
                                                            'Accelerated, '
                                                            '4-NON=Non-
Accelerated, '
                                                            '5-SML=Smaller '
                                                            'Reporting Filer, '
                                                            'NULL=not '
                                                            'assigned.',
                                          'name': 'afs',
                                          'titles': ['Status',
                                                     'Accelerated Filer '
                                                     'Status']},
                                         {'datatype': {'base': 'decimal',
                                                       'maxInclusive': 1,
                                                       'minInclusive': 0},
                                          'dc:description': 'Well Known '
                                                            'Seasoned Issuer '
                                                            '(WKSI). An issuer '
                                                            'that meets '
                                                            'specific '
                                                            'Commission '
                                                            'requirements at '
                                                            'some point during '
                                                            'a 60-day period '
                                                            'preceding the '
                                                            'date the issuer '
                                                            'satisfies its '
                                                            'obligation to '
                                                            'update its shelf '
                                                            'registration '
                                                            'statement.',
                                          'name': 'wksi',
                                          'titles': ['Well-known Seasoned '
                                                     'Issuer']},
                                         {'datatype': {'base': 'string',
                                                       'maxLength': 4},
                                          'dc:description': 'Fiscal Year End '
```

```
                   'Date, rounded to '
                   'nearest '
                   'month-end.',
 'name': 'fye',
 'titles': ['FY End Date']},
{'datatype': {'base': 'string',
              'maxLength': 20},
 'dc:description': 'The submission '
                   'type of the '
                   "registrant's "
                   'filing.',
 'name': 'form',
 'titles': ['Submission Type',
            'Filing Type',
            'EDGAR Form Type']},
{'datatype': {'base': 'string',
              'maxLength': 8,
              'minLength': 8},
 'dc:description': 'Balance Sheet '
                   'Date.',
 'name': 'period',
 'titles': ['Report Period',
            'Date of Balance Sheet, '
            'rounded to nearest '
            'month-end']},
{'datatype': {'base': 'string',
              'maxLength': 4,
              'minLength': 4},
 'dc:description': 'Fiscal Year Focus '
                   '(as defined in '
                   'EFM Ch. 6).',
 'name': 'fy',
 'titles': ['Fiscal Year']},
{'datatype': {'base': 'string',
              'maxLength': 2,
              'minLength': 2},
 'dc:description': 'Fiscal Period '
                   'Focus (as defined '
                   'in EFM Ch. 6) '
                   'within Fiscal '
                   'Year. The 10-Q '
                   'for the 1st, 2nd '
                   'and 3rd quarters '
                   'would have a '
                   'fiscal period '
                   'focus of Q1, Q2 '
                   '(or H1), and Q3 '
                   '(or M9) '
                   'respectively, and '
                   'a 10-K would have '
                   'a fiscal period '
                   'focus of FY.',
 'name': 'fp',
 'titles': ['Fiscal Period']},
{'datatype': {'base': 'string',
              'maxLength': 8},
 'dc:description': 'The date of the '
                   "registrant's "
                   'filing with the '
                   'Commission.',
 'name': 'filed',
 'titles': ['Date Filed']},
{'datatype': {'base': 'date',
              'format': 'YYYYMMDD '
                        'HH:MM:SS.S'},
 'dc:description': 'The acceptance '
                   'date and time of '
                   "the registrant's "
                   'filing with the '
                   'Commission. '
                   'Filings accepted '
                   'after 5:30pm EST '
                   'are considered '
                   'filed on the '
                   'following '
                   'business day.',
 'name': 'accepted',
 'titles': ['Acceptance Datetime']},
{'datatype': {'base': 'decimal',
              'maxInclusive': 255,
              'minInclusive': 0},
 'dc:description': 'Previous Report.  '
                   'TRUE indicates '
                   'that the '
                   'submission '
                   'information was '
                   'subsequently '
                   'amended prior to '
                   'the end cutoff '
                   'date of the data '
                   'set.',
 'name': 'prevrpt',
 'required': 'true',
 'titles': ['Previous Report Flag',
            'Subsequently Amended '
            'Flag']},
{'datatype': {'base': 'decimal',
              'maxInclusive': 255,
              'minInclusive': 0},
 'dc:description': 'TRUE indicates '
                   'that the XBRL '
                   'submission '
                   'contains '
                   'quantitative '
                   'disclosures '
                   'within the '
                   'footnotes and '
                   'schedules at the '
                   'required detail '
                   'level (e.g., each '
                   'amount).',
 'name': 'detail',
 'required': 'true',
 'titles': ['Detail Tagged']},
{'datatype': {'base': 'string',
              'maxLength': 32},
 'dc:description': 'The name of the '
                   'submitted XBRL '
                   'Instance Document '
                   '(EX-101.INS) type '
                   'data file. The '
                   'name often begins '
                   'with the company '
                   'ticker symbol.',
 'name': 'instance',
 'titles': ['Instance Filename']},
{'datatype': {'base': 'decimal',
              'maxInclusive': 32767,
              'minInclusive': 0},
 'dc:description': 'Number of Central '
                   'Index Keys (CIK) '
                   'of registrants '
                   '(i.e., business '
                   'units) included '
                   'in the '
                   'consolidating '
                   "entity's "
                   'submitted filing.',
 'name': 'nciks',
 'required': 'true',
 'titles': ['Number of '
            'Coregistrants']},
{'datatype': {'base': 'string',
              'maxLength': 120},
 'dc:description': 'Additional CIKs '
                   'of co-registrants '
                   'included in a '
                   'consolidating '
                   "entity's EDGAR "
                   'submission, '
                   'separated by '
                   'spaces. If there '
                   'are no other '
                   'co-registrants '
                   '(i.e., nciks = '
                   '1), the value of '
                   'aciks is NULL. '
                   'For a very small '
                   'number of filers, '
                   'the list of '
                   'co-registrants is '
                   'too long to fit '
                   'in the field. '
                   'Where this is the '
                   'case, PARTIAL '
                   'will appear at '
                   'the end of the '
                   'list indicating '
                   'that not all '
                   "co-registrants' "
                   'CIKs are included '
                   'in the field; '
                   'users should '
                   'refer to the '
                   'complete '
                   'submission file '
                   'for all CIK '
                   'information.',
 'name': 'aciks',
 'titles': ['Additional Coregistrant '
            'CIKs']},
{'datatype': {'base': 'decimal'},
 'dc:description': 'Public float, in '
                   'USD, if provided '
                   'in this '
                   'submission.',
 'name': 'pubfloatusd',
 'titles': ['Public Float']},
{'datatype': {'base': 'string',
              'maxLength': 8},
 'dc:description': 'Date on which the '
                   'public float was '
```

```
                                                'measured by the '
                                                'filer.',
                              'name': 'floatdate',
                              'titles': ['Public Float Measurement '
                                         'Date']},
                             {'datatype': {'base': 'string',
                                           'maxLength': 255},
                              'dc:description': 'If the public '
                                                'float value was '
                                                'computed by '
                                                'summing across '
                                                'several tagged '
                                                'values, this '
                                                'indicates the '
                                                'nature of the '
                                                'summation.',
                              'name': 'floataxis',
                              'titles': ['Public Float Axis']},
                             {'datatype': {'base': 'decimal',
                                           'maxInclusive': 255,
                                           'minInclusive': 0},
                              'dc:description': 'If the public '
                                                'float was '
                                                'computed, the '
                                                'number of terms '
                                                'in the summation.',
                              'name': 'floatmems',
                              'titles': ['Public Float Members']}],
                 'primaryKey': 'adsh'},
 'url': 'sub.tsv'},
{'tableSchema': {'aboutUrl': 'readme.htm',
                 'columns': [{'datatype': {'base': 'string',
                                           'maxLength': 256},
                              'dc:description': 'The unique '
                                                'identifier (name) '
                                                'for a tag in a '
                                                'specific taxonomy '
                                                'release.',
                              'name': 'tag',
                              'required': 'true',
                              'titles': ['Localname']},
                             {'datatype': {'base': 'string',
                                           'maxLength': 20},
                              'dc:description': 'For a standard '
                                                'tag, an '
                                                'identifier for '
                                                'the taxonomy; '
                                                'otherwise the '
```

```
                   'accession number '
                   'where the tag was '
                   'defined.',
 'name': 'version',
 'required': 'true',
 'titles': ['Namespace', 'Taxonomy']},
{'datatype': {'base': 'decimal',
              'maxInclusive': 1,
              'minInclusive': 0},
 'dc:description': '1 if tag is '
                   'custom '
                   '(version=adsh), 0 '
                   'if it is '
                   'standard. Note: '
                   'This flag is '
                   'technically '
                   'redundant with '
                   'the version and '
                   'adsh fields.',
 'name': 'custom',
 'required': 'true',
 'titles': []},
{'datatype': {'base': 'decimal',
              'maxInclusive': 1,
              'minInclusive': 0},
 'dc:description': '1 if the tag is '
                   'not used to '
                   'represent a '
                   'numeric fact.',
 'name': 'abstract',
 'required': 'true',
 'titles': []},
{'datatype': {'base': 'string',
             'maxLength': 20},
 'dc:description': 'If abstract=1, '
                   'then NULL, '
                   'otherwise the '
                   'data type (e.g., '
                   'monetary) for the '
                   'tag.',
 'name': 'datatype',
 'titles': []},
{'datatype': {'base': 'string',
              'maxLength': 1},
 'dc:description': 'If abstract=1, '
                   'then NULL; '
                   'otherwise, I if '
                   'the value is a '
                   'point in time, or '
                   'D if the value is '
                   'a duration.',
 'name': 'iord',
 'titles': ['Instant or Duration']},
{'datatype': {'base': 'string',
              'maxLength': 1},
 'dc:description': 'If datatype = '
                   'monetary, then '
                   "the tag's natural "
                   'accounting '
                   'balance from the '
                   'perspective of '
                   'the balance sheet '
                   'or income '
                   'statement (debit '
                   'or credit); if '
                   'not defined, then '
                   'NULL.',
 'name': 'crdr',
 'titles': ['Credit or Debit']},
{'datatype': {'base': 'string',
              'maxLength': 512},
 'dc:description': 'If a standard '
                   'tag, then the '
                   'label text '
                   'provided by the '
                   'taxonomy, '
                   'otherwise the '
                   'text provided by '
                   'the filer. A tag '
                   'which had neither '
                   'would have a NULL '
                   'value here.',
 'name': 'tlabel',
 'titles': ['Label']},
{'datatype': {'base': 'string'},
 'dc:description': 'The detailed '
                   'definition for '
                   'the tag. If a '
                   'standard tag, '
                   'then the text '
                   'provided by the '
                   'taxonomy, '
                   'otherwise the '
                   'text assigned by '
                   'the filer. Some '
                   'tags have '
```

```
                                                'neither, in which '
                                                'case this field '
                                                'is NULL.',
                              'name': 'doc',
                              'titles': ['Documentation']}],
                 'primaryKey': ['tag', 'version']},
 'url': 'tag.tsv'},
{'tableSchema': {'aboutUrl': 'readme.htm',
                 'columns': [{'datatype': {'base': 'string',
                                           'maxLength': 34},
                              'dc:description': 'MD5 hash of the '
                                                'segments field '
                                                'text. Although '
                                                'MD5 is unsuitable '
                                                'for cryptographic '
                                                'use, it is used '
                                                'here merely to '
                                                'limit the size of '
                                                'the primary key.',
                              'name': 'dimh',
                              'required': 'true',
                              'titles': ['Dimension Hash']},
                             {'datatype': {'base': 'string',
                                           'maxLength': 1024},
                              'dc:description': 'Concatenation of '
                                                'tag names '
                                                'representing the '
                                                'axis and members '
                                                'appearing in the '
                                                'XBRL segments. '
                                                'Tag names have '
                                                'their first '
                                                'characters '
                                                '"Statement", last '
                                                '4 characters '
                                                '"Axis", and last '
                                                '6 characters '
                                                '"Member" or '
                                                '"Domain" '
                                                'truncated where '
                                                'they appear. '
                                                'Namespaces and '
                                                'prefixes are '
                                                'ignored because '
                                                'EDGAR validation '
                                                'guarantees that '
                                                'the local-names '
                                                'are unique with a '
```

```
                                                            'submission. Each '
                                                            'dimension is '
                                                            'represented as '
                                                            'the pair '
                                                            '"{axis}={member};"
'
                                                            'and the axes '
                                                            'concatenated in '
                                                            'lexical order. '
                                                            'Example: '
'"LegalEntity=Xyz;Scenario=Restated;" '
                                                            'represents the '
                                                            'XBRL segment with '
                                                            'dimension '
                                                            'LegalEntityAxis '
                                                            'and member '
                                                            'XyzMember, '
                                                            'dimension '
'StatementScenarioAxis '
                                                            'and member '
                                                            'RestatedMember.',
                                          'name': 'segments',
                                          'titles': []},
                                         {'datatype': {'base': 'decimal',
                                                       'maxInclusive': 1,
                                                       'minInclusive': 0},
                                          'dc:description': 'TRUE if the '
                                                            'segments field '
                                                            'would have been '
                                                            'longer than 1024 '
                                                            'characters had it '
                                                            'not been '
                                                            'truncated, else '
                                                            'FALSE.',
                                          'name': 'segt',
                                          'required': 'true',
                                          'titles': ['Segments Truncated']}],
                             'primaryKey': 'dimh'},
             'url': 'dim.tsv'},
            {'tableSchema': {'aboutUrl': 'readme.htm',
                             'columns': [{'datatype': {'base': 'string',
                                                       'maxLength': 20,
                                                       'minLength': 20},
                                          'dc:description': 'Accession Number. '
                                                            'The 20-character '
                                                            'string formed '
                                                            'from the 18-digit '
                                                            'number assigned '
```

```
                   'by the Commission '
                   'to each EDGAR '
                   'submission.',
 'name': 'adsh',
 'required': 'true',
 'titles': ['Accession Number']},
{'datatype': {'base': 'string',
              'maxLength': 255},
 'dc:description': 'The unique '
                   'identifier (name) '
                   'for a tag in a '
                   'specific taxonomy '
                   'release.',
 'name': 'tag',
 'required': 'true',
 'titles': ['Localname']},
{'datatype': {'base': 'string',
              'maxLength': 20},
 'dc:description': 'For a standard '
                   'tag, an '
                   'identifier for '
                   'the taxonomy; '
                   'otherwise the '
                   'accession number '
                   'where the tag was '
                   'defined.',
 'name': 'version',
 'required': 'true',
 'titles': ['Namespace']},
{'datatype': {'base': 'string',
              'maxLength': 8,
              'minLength': 8},
 'dc:description': 'The end date for '
                   'the data value, '
                   'rounded to the '
                   'nearest month '
                   'end.',
 'name': 'ddate',
 'required': 'true',
 'titles': ['Data Date']},
{'datatype': {'base': 'decimal',
              'minInclusive': 0},
 'dc:description': 'The count of the '
                   'number of '
                   'quarters '
                   'represented by '
                   'the data value, '
                   'rounded to the '
                   'nearest whole '
                   'number. "0" '
                   'indicates it is a '
                   'point-in-time '
                   'value.',
 'name': 'qtrs',
 'required': 'true',
 'titles': ['Quarters']},
{'datatype': {'base': 'string',
              'maxLength': 50},
 'dc:description': 'The unit of '
                   'measure for the '
                   'value.',
 'name': 'uom',
 'required': 'true',
 'titles': ['Unit of Measure']},
{'datatype': {'base': 'string',
              'maxLength': 34},
 'dc:description': 'The 32-byte '
                   'hexadecimal key '
                   'for the '
                   'dimensional '
                   'information in '
                   'the DIM data set.',
 'name': 'dimh',
 'titles': ['Dimension Hash']},
{'datatype': {'base': 'decimal',
              'maxInclusive': 32767,
              'minInclusive': 0},
 'dc:description': 'A positive '
                   'integer to '
                   'distinguish '
                   'different '
                   'reported facts '
                   'that otherwise '
                   'would have the '
                   'same primary key. '
                   'For most '
                   'purposes, data '
                   'with iprx greater '
                   'than 1 are not '
                   'needed. The '
                   'priority for the '
                   'fact based on '
                   'higher precision, '
                   'closeness of the '
                   'end date to a '
                   'month end, and '
                   'closeness of the '
                   'duration to a '
                   'multiple of three '
                   'months. See '
                   'fields dcml, durp '
                   'and datp below.',
 'name': 'iprx',
 'titles': ['Fact Preference']},
{'datatype': {'base': 'decimal'},
 'dc:description': 'The value. This '
                   'is not scaled, it '
                   'is as found in '
                   'the Interactive '
                   'Data file, but is '
                   'rounded to four '
                   'digits to the '
                   'right of the '
                   'decimal point.',
 'name': 'value',
 'titles': []},
{'datatype': {'base': 'string',
              'maxLength': 512},
 'dc:description': 'The plain text of '
                   'any superscripted '
                   'footnotes on the '
                   'value, if any, as '
                   'shown on the '
                   'statement page, '
                   'truncated to 512 '
                   'characters.',
 'name': 'footnote',
 'titles': ['Footnote Text']},
{'datatype': {'base': 'decimal',
              'minInclusive': 0},
 'dc:description': 'Number of bytes '
                   'in the plain text '
                   'of the footnote '
                   'prior to '
                   'truncation; zero '
                   'if no footnote.',
 'name': 'footlen',
 'required': 'true',
 'titles': ['Footnote Length']},
{'datatype': {'base': 'decimal',
              'minInclusive': 0},
 'dc:description': 'Small integer '
                   'representing the '
                   'number of '
                   'dimensions. Note '
                   'that this value '
                   'is a function of '
                   'the dimension '
                   'segments.',
 'name': 'dimn',
 'required': 'true',
 'titles': ['Number of Dimensions']},
{'datatype': {'base': 'string',
              'maxLength': 256},
 'dc:description': 'If specified, '
                   'indicates a '
                   'specific '
                   'co-registrant, '
                   'the parent '
                   'company, or other '
                   'entity (e.g., '
                   'guarantor). NULL '
                   'indicates the '
                   'consolidated '
                   'entity. Note that '
                   'this value is a '
                   'function of the '
                   'dimension '
                   'segments.',
 'name': 'coreg',
 'titles': ['Coregistrant']},
{'datatype': {'base': 'decimal'},
 'dc:description': 'The difference '
                   'between the '
                   'reported fact '
                   'duration and the '
                   'quarter duration '
                   '(qtrs), expressed '
                   'as a fraction of '
                   '1. For example, a '
                   'fact with '
                   'duration of 120 '
                   'days rounded to a '
                   '91-day quarter '
                   'has a durp value '
                   'of 29/91 = '
                   '+0.3187.',
 'name': 'durp',
 'titles': ['Duration Preference']},
{'datatype': {'base': 'decimal'},
 'dc:description': 'The date '
                   'proximity in '
```

```
                                                            'number of days '
                                                            'between end date '
                                                            'reported and '
                                                            'month-end rounded '
                                                            'date.',
                                          'name': 'datp',
                                          'titles': ['Date Preference']},
                                         {'datatype': {'base': 'decimal',
                                                       'maxInclusive': 32767,
                                                       'minInclusive': -32768},
                                          'dc:description': 'The value of the '
                                                            'fact "decimals" '
                                                            'attribute, with '
                                                            'INF represented '
                                                            'by 32767.',
                                          'name': 'dcml',
                                          'titles': ['Decimals']}],
                             'foreignKeys': [{'columnReference': 'adsh',
                                              'reference': {'columnReference':
'adsh',
                                                            'resource':
'sub.tsv'}},
                                             {'columnReference': 'dimh',
                                              'reference': {'columnReference':
'dimh',
                                                            'resource':
'https://wwww.sec.gov/files2018q3.zip#path=dim.tsv'}},
                                             {'columnReference': ['tag',
                                                                  'version'],
                                              'reference': {'columnReference':
['tag',
'version'],
                                                            'resource':
'https://wwww.sec.gov/files2018q3.zip#path=tag.tsv'}}],
                             'primaryKey': ['adsh',
                                            'tag',
                                            'version',
                                            'ddate',
                                            'qtrs',
                                            'uom',
                                            'dimh',
                                            'iprx']},
             'url': 'num.tsv'},
            {'tableSchema': {'aboutUrl': 'readme.htm',
                             'columns': [{'datatype': {'base': 'string',
                                                       'maxLength': 20,
                                                       'minLength': 20},
                                          'dc:description': 'Accession Number. '
```

```
                   'The 20-character '
                   'string formed '
                   'from the 18-digit '
                   'number assigned '
                   'by the Commission '
                   'to each EDGAR '
                   'submission.',
 'name': 'adsh',
 'required': 'true',
 'titles': ['Accession number']},
{'datatype': {'base': 'string',
              'maxLength': 255},
 'dc:description': 'The unique '
                   'identifier (name) '
                   'for a tag in a '
                   'specific taxonomy '
                   'release.',
 'name': 'tag',
 'required': 'true',
 'titles': ['Localname']},
{'datatype': {'base': 'string',
              'maxLength': 20},
 'dc:description': 'For a standard '
                   'tag, an '
                   'identifier for '
                   'the taxonomy; '
                   'otherwise the '
                   'accession number '
                   'where the tag was '
                   'defined. For '
                   'example, '
                   '"invest/2013" '
                   'indicates that '
                   'the tag is '
                   'defined in the '
                   '2013 INVEST '
                   'taxonomy.',
 'name': 'version',
 'required': 'true',
 'titles': ['Namespace', 'Taxonomy']},
{'datatype': {'base': 'string',
              'maxLength': 8,
              'minLength': 8},
 'dc:description': 'The end date for '
                   'the data value, '
                   'rounded to the '
                   'nearest month '
                   'end.',
 'name': 'ddate',
 'required': 'true',
 'titles': ['Data Date']},
{'datatype': {'base': 'decimal',
              'minInclusive': 0},
 'dc:description': 'The count of the '
                   'number of '
                   'quarters '
                   'represented by '
                   'the data value, '
                   'rounded to the '
                   'nearest whole '
                   'number. A point '
                   'in time value is '
                   'represented by 0.',
 'name': 'qtrs',
 'required': 'true',
 'titles': ['Quarters']},
{'datatype': {'base': 'decimal',
              'maxInclusive': 32767,
              'minInclusive': -32768},
 'dc:description': 'A positive '
                   'integer to '
                   'distinguish '
                   'different '
                   'reported facts '
                   'that otherwise '
                   'would have the '
                   'same primary key. '
                   'For most '
                   'purposes, data '
                   'with iprx greater '
                   'than 1 are not '
                   'needed. The '
                   'priority for the '
                   'fact based on '
                   'higher precision, '
                   'closeness of the '
                   'end date to a '
                   'month end, and '
                   'closeness of the '
                   'duration to a '
                   'multiple of three '
                   'months. See '
                   'fields dcml, durp '
                   'and datp below.',
 'name': 'iprx',
 'titles': ['Fact Preference',
            'Preferred Fact Sort '
            'Key']},
{'datatype': {'base': 'string',
              'maxLength': 5},
 'dc:description': 'The ISO language '
                   'code of the fact '
                   'content.',
 'name': 'lang',
 'titles': ['Language']},
{'datatype': {'base': 'decimal',
              'maxInclusive': 32767,
              'minInclusive': -32768},
 'dc:description': 'The value of the '
                   'fact "xml:lang" '
                   'attribute, en-US '
                   'represented by '
                   '32767, other "en" '
                   'dialects having '
                   'lower values, and '
                   'other languages '
                   'lower still.',
 'name': 'dcml',
 'titles': ['Language Preference',
            'Language Sort Key']},
{'datatype': {'base': 'decimal'},
 'dc:description': 'The difference '
                   'between the '
                   'reported fact '
                   'duration and the '
                   'quarter duration '
                   '(qtrs), expressed '
                   'as a fraction of '
                   '1. For example, a '
                   'fact with '
                   'duration of 120 '
                   'days rounded to a '
                   '91-day quarter '
                   'has a durp value '
                   'of 29/91 = '
                   '+0.3187.',
 'name': 'durp',
 'titles': ['Duration Preference']},
{'datatype': {'base': 'decimal'},
 'dc:description': 'The date '
                   'proximity in '
                   'number of days '
                   'between end date '
                   'reported and '
                   'month-end rounded '
                   'date.',
 'name': 'datp',
 'titles': ['Date Preference']},
{'datatype': {'base': 'string',
              'maxLength': 34},
 'dc:description': 'The 32-byte '
                   'hexadecimal key '
                   'for the '
                   'dimensional '
                   'information in '
                   'the DIM data set.',
 'name': 'dimh',
 'titles': ['Dimension Hash']},
{'datatype': {'base': 'decimal',
              'minInclusive': 0},
 'dc:description': 'Small integer '
                   'representing the '
                   'number of '
                   'dimensions, '
                   'useful for '
                   'sorting. Note '
                   'that this value '
                   'is function of '
                   'the dimension '
                   'segments.',
 'name': 'dimn',
 'required': 'true',
 'titles': ['Number of Dimensions']},
{'datatype': {'base': 'string',
              'maxLength': 256},
 'dc:description': 'If specified, '
                   'indicates a '
                   'specific '
                   'co-registrant, '
                   'the parent '
                   'company, or other '
                   'entity (e.g., '
                   'guarantor). NULL '
                   'indicates the '
                   'consolidated '
                   'entity. Note that '
                   'this value is a '
                   'function of the '
                   'dimension '
                   'segments.',
 'name': 'coreg',
 'titles': ['Coregistrant']},
{'datatype': {'base': 'decimal',
              'maxInclusive': 1,
              'minInclusive': 0},
 'dc:description': 'Flag indicating '
                   'whether the value '
                   'has had tags '
                   'removed.',
 'name': 'escaped',
 'required': 'true',
 'titles': []},
{'datatype': {'base': 'decimal',
              'minInclusive': 0},
 'dc:description': 'Number of bytes '
                   'in the original, '
                   'unprocessed '
                   'value. Zero '
                   'indicates a NULL '
                   'value.',
 'name': 'srclen',
 'required': 'true',
 'titles': ['Source Length']},
{'datatype': {'base': 'decimal',
              'minInclusive': 0},
 'dc:description': 'The original '
                   'length of the '
                   'whitespace '
                   'normalized value, '
                   'which may have '
                   'been greater than '
                   '8192.',
 'name': 'txtlen',
 'required': 'true',
 'titles': ['Text Length']},
{'datatype': {'base': 'string',
              'maxLength': 512},
 'dc:description': 'The plain text of '
                   'any superscripted '
                   'footnotes on the '
                   'value, as shown '
                   'on the page, '
                   'truncated to 512 '
                   'characters, or if '
                   'there is no '
                   'footnote, then '
                   'this field will '
                   'be blank.',
 'name': 'footnote',
 'titles': ['Footnote Text']},
{'datatype': {'base': 'decimal',
              'minInclusive': 0},
 'dc:description': 'Number of bytes '
                   'in the plain text '
                   'of the footnote '
                   'prior to '
                   'truncation.',
 'name': 'footlen',
 'required': 'true',
 'titles': ['Footnote Length']},
{'datatype': {'base': 'string',
              'maxLength': 255},
 'dc:description': 'The value of the '
                   'contextRef '
                   'attribute in the '
                   'source XBRL '
                   'document, which '
                   'can be used to '
                   'recover the '
                   'original HTML '
                   'tagging if '
                   'desired.',
 'name': 'context',
 'titles': ['Context Ref']},
{'datatype': {'base': 'string'},
 'dc:description': 'The value, with '
                   'all whitespace '
                   'normalized, that '
                   'is, all sequences '
                   'of line feeds, '
                   'carriage returns, '
                   'tabs, '
                   'non-breaking '
                   'spaces, and '
                   'spaces having '
                   'been collapsed to '
                   'a single space, '
                   'and no leading or '
                   'trailing spaces. '
                   'Escaped XML that '
                   'appears in EDGAR '
                   '"Text Block" tags '
                   'is processed to '
                   'remove all '
                   'mark-up '
                   '(comments, '
                   'processing '
                   'instructions, '
```

```
                                                            'elements, '
                                                            'attributes). The '
                                                            'resulting text is '
                                                            'not intended for '
                                                            'end user display '
                                                            'but only for text '
                                                            'analysis '
                                                            'applications.',
                                          'name': 'value',
                                          'titles': []}],
                             'foreignKeys': [{'columnReference': 'adsh',
                                              'reference': {'columnReference':
'adsh',
                                                            'resource':
'sub.tsv'}},
                                             {'columnReference': 'dimh',
                                              'reference': {'columnReference':
'dimh',
                                                            'resource':
'https://wwww.sec.gov/files2018q3.zip#path=dim.tsv'}},
                                             {'columnReference': ['tag',
                                                                  'version'],
                                              'reference': {'columnReference':
['tag',
'version'],
                                                            'resource':
'https://wwww.sec.gov/files2018q3.zip#path=tag.tsv'}}],
                             'primaryKey': ['adsh',
                                            'tag',
                                            'version',
                                            'ddate',
                                            'qtrs',
                                            'dimh',
                                            'iprx']},
             'url': 'txt.tsv'},
            {'tableSchema': {'aboutUrl': 'readme.htm',
                             'columns': [{'datatype': {'base': 'string',
                                                       'maxLength': 20,
                                                       'minLength': 20},
                                          'dc:description': 'Accession Number. '
                                                            'The 20-character '
                                                            'string formed '
                                                            'from the 18-digit '
                                                            'number assigned '
                                                            'by the Commission '
                                                            'to each EDGAR '
                                                            'submission.',
                                          'name': 'adsh',
 'required': 'true',
 'titles': ['Accession Number']},
{'datatype': {'base': 'decimal',
              'minInclusive': 0},
 'dc:description': 'Represents the '
                   'report grouping. '
                   'The numeric value '
                   'refers to the "R '
                   'file" as computed '
                   'by the renderer '
                   'and posted on the '
                   'EDGAR website. '
                   'Note that in some '
                   'situations the '
                   'numbers skip.',
 'name': 'report',
 'required': 'true',
 'titles': ['Report Number']},
{'datatype': {'base': 'string',
              'maxLength': 1},
 'dc:description': 'The type of '
                   'interactive data '
                   'file rendered on '
                   'the EDGAR '
                   'website, H = .htm '
                   'file, X = .xml '
                   'file.',
 'name': 'rfile',
 'required': 'true',
 'titles': ['Report File Type']},
{'datatype': {'base': 'string',
              'maxLength': 2},
 'dc:description': 'If available, one '
                   'of the menu '
                   'categories as '
                   'computed by the '
                   'renderer: '
                   'C=Cover, '
                   'S=Statements, '
                   'N=Notes, '
                   'P=Policies, '
                   'T=Tables, '
                   'D=Details, '
                   'O=Other, and '
                   'U=Uncategorized.',
 'name': 'menucat',
 'titles': ['Menu Category']},
{'datatype': {'base': 'string',
              'maxLength': 512},
 'dc:description': 'The portion of '
                   'the long name '
                   'used in the '
                   'renderer menu.',
 'name': 'shortname',
 'titles': ['Short Name']},
{'datatype': {'base': 'string',
              'maxLength': 512},
 'dc:description': 'The '
                   'space-normalized '
                   'text of the XBRL '
                   'link "definition" '
                   'element content.',
 'name': 'longname',
 'titles': ['Long Name']},
{'datatype': {'base': 'string',
              'maxLength': 255},
 'dc:description': 'The XBRL '
                   '"roleuri" of the '
                   'role.',
 'name': 'roleuri',
 'titles': ['Role URI']},
{'datatype': {'base': 'string',
              'maxLength': 255},
 'dc:description': 'The XBRL roleuri '
                   'of a role for '
                   'which this role '
                   'has a matching '
                   'shortname prefix '
                   'and a higher '
                   'level menu '
                   'category, as '
                   'computed by the '
                   'renderer.',
 'name': 'parentroleuri',
 'titles': ['Parent Role URI']},
{'datatype': {'base': 'decimal',
              'minInclusive': 0},
 'dc:description': 'The value of the '
                   'report field for '
                   'the role where '
                   'roleuri equals '
                   'this '
                   'parentroleuri.',
 'name': 'parentreport',
 'titles': ['Parent Report']},
{'datatype': {'base': 'decimal',
                                                       'maxInclusive': 32767,
                                                       'minInclusive': 0},
                                          'dc:description': 'The highest '
                                                            'ancestor report '
                                                            'reachable by '
                                                            'following '
                                                            'parentreport '
                                                            'relationships. A '
                                                            'note (menucat = '
                                                            'N) is its own '
                                                            'ultimate parent.',
                                          'name': 'ultparentrpt',
                                          'titles': ['Ultimate Parent']}],
                             'foreignKeys': [{'columnReference': 'adsh',
                                              'reference': {'columnReference':
'adsh',
                                                            'resource':
'sub.tsv'}}],
                             'primaryKey': ['adsh', 'report']},
             'url': 'ren.tsv'},
            {'tableSchema': {'aboutUrl': 'readme.htm',
                             'columns': [{'datatype': {'base': 'string',
                                                       'maxLength': 20,
                                                       'minLength': 20},
                                          'dc:description': 'Accession Number. '
                                                            'The 20-character '
                                                            'string formed '
                                                            'from the 18-digit '
                                                            'number assigned '
                                                            'by the Commission '
                                                            'to each EDGAR '
                                                            'submission.',
                                          'name': 'adsh',
                                          'required': 'true',
                                          'titles': ['Accession Number']},
                                         {'datatype': {'base': 'decimal',
                                                       'minInclusive': 0},
                                          'dc:description': 'Represents the '
                                                            'report grouping. '
                                                            'The numeric value '
                                                            'refers to the "R '
                                                            'file" as computed '
                                                            'by the renderer '
                                                            'and posted on the '
                                                            'EDGAR website. '
                                                            'Note that in some '
                                                            'situations the '
                                                            'numbers skip.',
 'name': 'report',
 'required': 'true',
 'titles': []},
{'datatype': {'base': 'decimal',
              'minInclusive': 0},
 'dc:description': 'Represents the '
                   "tag's "
                   'presentation line '
                   'order for a given '
                   'report. Together '
                   'with the '
                   'statement and '
                   'report field, '
                   'presentation '
                   'location, order '
                   'and grouping can '
                   'be derived.',
 'name': 'line',
 'required': 'true',
 'titles': []},
{'datatype': {'base': 'string',
              'maxLength': 2},
 'dc:description': 'The financial '
                   'statement '
                   'location to which '
                   'the value of the '
                   '"report" field '
                   'pertains.',
 'name': 'stmt',
 'titles': ['Statement']},
{'datatype': {'base': 'decimal',
              'maxInclusive': 1,
              'minInclusive': 0},
 'dc:description': '1 indicates that '
                   'the value was '
                   'presented '
                   '"parenthetically" '
                   'instead of in '
                   'fields within the '
                   'financial '
                   'statements. For '
                   'example: '
                   'Receivables (net '
                   'of allowance for '
                   'bad debts of USD '
                   '200 in 2012) USD '
                   '700',
 'name': 'inpth',
 'required': 'true',
 'titles': ['Parenthentical']},
{'datatype': {'base': 'string',
              'maxLength': 256},
 'dc:description': 'The tag chosen by '
                   'the filer for '
                   'this line item.',
 'name': 'tag',
 'required': 'true',
 'titles': ['Localname']},
{'datatype': {'base': 'string',
              'maxLength': 20},
 'dc:description': 'The taxonomy '
                   'identifier if the '
                   'tag is a standard '
                   'tag, otherwise '
                   'adsh.',
 'name': 'version',
 'required': 'true',
 'titles': ['Namespace', 'Taxonomy']},
{'datatype': {'base': 'string',
              'maxLength': 50},
 'dc:description': 'The XBRL link '
                   '"role" of the '
                   'preferred label, '
                   'using only the '
                   'portion of the '
                   'role URI after '
                   'the last "/".',
 'name': 'prole',
 'titles': ['Preferred Role']},
{'datatype': {'base': 'string',
              'maxLength': 512},
 'dc:description': 'The text '
                   'presented on the '
                   'line item, also '
                   'known as a '
                   '"preferred" '
                   'label.',
 'name': 'plabel',
 'titles': ['Label']},
{'datatype': {'base': 'decimal',
              'maxInclusive': 1,
              'minInclusive': 0},
 'dc:description': 'Flag to indicate '
                   'whether the prole '
                   'is treated as '
                   'negating by the '
```

```
                                                            'renderer.',
                                          'name': 'negating',
                                          'required': 'true',
                                          'titles': []}],
                             'foreignKeys': [{'columnReference': ['adsh',
                                                                  'report'],
                                              'reference': {'columnReference':
['adsh',
'report'],
                                                            'resource':
'ren.tsv'}},
                                             {'columnReference': ['tag',
                                                                  'version'],
                                              'reference': {'columnReference':
['tag',
'version'],
                                                            'resource':
'tag.tsv'}}],
                             'primaryKey': ['adsh', 'report', 'line']},
             'url': 'pre.tsv'},
            {'tableSchema': {'aboutUrl': 'readme.htm',
                             'columns': [{'datatype': {'base': 'string',
                                                       'maxLength': 20,
                                                       'minLength': 20},
                                          'dc:description': 'Accession Number. '
                                                            'The 20-character '
                                                            'string formed '
                                                            'from the 18-digit '
                                                            'number assigned '
                                                            'by the Commission '
                                                            'to each EDGAR '
                                                            'submission.',
                                          'name': 'adsh',
                                          'required': 'true',
                                          'titles': ['Accession Number']},
                                         {'datatype': {'base': 'decimal',
                                                       'maxInclusive': 255,
                                                       'minInclusive': 0},
                                          'dc:description': 'Sequential number '
                                                            'for grouping arcs '
                                                            'in a submission.',
                                          'name': 'grp',
                                          'required': 'true',
                                          'titles': ['Group']},
                                         {'datatype': {'base': 'decimal',
                                                       'minInclusive': 255},
                                          'dc:description': 'Sequential number '
                                                            'for arcs within a '
```

```
                               'group in a '
                               'submission.',
             'name': 'arc',
             'required': 'true',
             'titles': []},
            {'datatype': {'base': 'decimal',
                          'maxInclusive': 1,
                          'minInclusive': 0},
             'dc:description': 'Indicates a '
                               'weight of -1 '
                               '(TRUE if the arc '
                               'is negative), but '
                               'typically +1 '
                               '(FALSE).',
             'name': 'negative',
             'required': 'true',
             'titles': ['Negative Weight']},
            {'datatype': {'base': 'string',
                          'maxLength': 256},
             'dc:description': 'The tag for the '
                               'parent of the arc',
             'name': 'ptag',
             'required': 'true',
             'titles': ['Parent Tag']},
            {'datatype': {'base': 'string',
                          'maxLength': 20},
             'dc:description': 'The version of '
                               'the tag for the '
                               'parent of the arc',
             'name': 'pversion',
             'required': 'true',
             'titles': ['Parent Namespace']},
            {'datatype': {'base': 'string',
                          'maxLength': 255},
             'dc:description': 'The tag for the '
                               'child of the arc',
             'name': 'ctag',
             'required': 'true',
             'titles': ['Child Tag']},
            {'datatype': {'base': 'string',
                          'maxLength': 20},
             'dc:description': 'The version of '
                               'the tag for the '
                               'child of the arc',
             'name': 'cversion',
             'required': 'true',
             'titles': ['Child Namespace']}],
'foreignKeys': [{'columnReference': 'adsh',
                                              'reference': {'columnReference':
'adsh',
                                                            'resource':
'sub.tsv'}},
                                             {'columnReference': ['ptag',
                                                                  'pversion'],
                                              'reference': {'columnReference':
['tag',
'version'],
                                                            'resource':
'tag.tsv'}},
                                             {'columnReference': ['ctag',
                                                                  'cversion'],
                                              'reference': {'columnReference':
['tag',
'version'],
                                                            'resource':
'tag.tsv'}}],
                             'primaryKey': ['adsh', 'grp', 'arc']},
             'url': 'cal.tsv'}]}
```

1.4 Data Organization

For each quarter, the FSN data is organized into eight file sets that contain information about submissions, numbers, taxonomy tags, presentation, and more. Each dataset consists of rows and fields and is provided as a tab-delimited text file:

File	Dataset	Description
SUB	Submission	Identifies each XBRL submission by company, form, date, etc
TAG	Tag	Defines and explains each taxonomy tag
DIM	Dimension	Adds detail to numeric and plain text data
NUM	Numeric	One row for each distinct data point in filing
TXT	Plain Text	Contains all non-numeric XBRL fields
REN	Rendering	Information for rendering on SEC website
PRE	Presentation	Detail on tag and number presentation in primary statements
CAL	Calculation	Shows arithmetic relationships among tags

1.5 Submission Data

The latest submission file contains around 6,500 entries.

[23]:
```
sub = pd.read_parquet(data_path / '2018_3' / 'parquet' / 'sub.parquet')
sub.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6493 entries, 0 to 6492
Data columns (total 40 columns):
 #   Column       Non-Null Count  Dtype
```

```
---  ------       --------------  -----
 0   adsh         6493 non-null   object
 1   cik          6493 non-null   int64
 2   name         6493 non-null   object
 3   sic          6492 non-null   float64
 4   countryba    6482 non-null   object
 5   stprba       5900 non-null   object
 6   cityba       6482 non-null   object
 7   zipba        6478 non-null   object
 8   bas1         6482 non-null   object
 9   bas2         2804 non-null   object
 10  baph         6482 non-null   object
 11  countryma    6448 non-null   object
 12  stprma       5906 non-null   object
 13  cityma       6448 non-null   object
 14  zipma        6447 non-null   object
 15  mas1         6448 non-null   object
 16  mas2         2761 non-null   object
 17  countryinc   5936 non-null   object
 18  stprinc      5632 non-null   object
 19  ein          6492 non-null   float64
 20  former       3618 non-null   object
 21  changed      3618 non-null   float64
 22  afs          6480 non-null   object
 23  wksi         6493 non-null   int64
 24  fye          6490 non-null   float64
 25  form         6493 non-null   object
 26  period       6493 non-null   int64
 27  fy           6375 non-null   float64
 28  fp           6358 non-null   object
 29  filed        6493 non-null   int64
 30  accepted     6493 non-null   object
 31  prevrpt      6493 non-null   int64
 32  detail       6493 non-null   int64
 33  instance     6493 non-null   object
 34  nciks        6493 non-null   int64
 35  aciks        130 non-null    object
 36  pubfloatusd  639 non-null    float64
 37  floatdate    640 non-null    float64
 38  floataxis    3 non-null      object
 39  floatmems    4 non-null      float64
dtypes: float64(8), int64(7), object(25)
memory usage: 2.0+ MB
```

1.5.1 Get AAPL submission

The submission dataset contains the unique identifiers required to retrieve the filings: the Central Index Key (CIK) and the Accession Number (adsh). The following shows some of the information about Apple's 2018Q1 10-Q filing:

[24]:
```
name = 'APPLE INC'
apple = sub[sub.name == name].T.dropna().squeeze()
key_cols = ['name', 'adsh', 'cik', 'name', 'sic', 'countryba', 'stprba',
            'cityba', 'zipba', 'bas1', 'form', 'period', 'fy', 'fp', 'filed']
apple.loc[key_cols]
```

[24]:
```
name                     APPLE INC
adsh          0000320193-18-000100
cik                         320193
name                     APPLE INC
sic                         3571.0
countryba                       US
stprba                          CA
cityba                   CUPERTINO
zipba                        95014
bas1           ONE APPLE PARK WAY
form                          10-Q
period                    20180630
fy                          2018.0
fp                              Q3
filed                     20180801
Name: 1101, dtype: object
```

1.6 Build AAPL fundamentals dataset

Using the central index key, we can identify all historical quarterly filings available for Apple, and combine this information to obtain 26 Forms 10-Q and nine annual Forms 10-K.

1.6.1 Get filings

[25]:
```
aapl_subs = pd.DataFrame()
for sub in data_path.glob('**/sub.parquet'):
    sub = pd.read_parquet(sub)
    aapl_sub = sub[(sub.cik.astype(int) == apple.cik) & (sub.form.isin(['10-Q',␣
 ↪'10-K']))]
    aapl_subs = pd.concat([aapl_subs, aapl_sub])
```

We find 15 quarterly 10-Q and 4 annual 10-K reports:

[26]:
```
aapl_subs.form.value_counts()
```

[26]:
```
10-Q    21
10-K     6
Name: form, dtype: int64
```

1.6.2 Get numerical filing data

With the Accession Number for each filing, we can now rely on the taxonomies to select the appropriate XBRL tags (listed in the TAG file) from the NUM and TXT files to obtain the numerical or textual/footnote data points of interest.

First, let's extract all numerical data available from the 19 Apple filings:

[27]:
```
aapl_nums = pd.DataFrame()
for num in data_path.glob('**/num.parquet'):
    num = pd.read_parquet(num).drop('dimh', axis=1)
    aapl_num = num[num.adsh.isin(aapl_subs.adsh)]
    print(len(aapl_num))
    aapl_nums = pd.concat([aapl_nums, aapl_num])
aapl_nums.ddate = pd.to_datetime(aapl_nums.ddate, format='%Y%m%d')
aapl_nums.to_parquet(data_path / 'aapl_nums.parquet')
```

```
923
1176
905
1028
707
961
919
1224
1035
1277
952
793
738
1364
751
784
1017
1001
937
942
1039
1271
805
755
682
1345
951
```

In total, the nine years of filing history provide us with over 18,000 numerical values for AAPL.

[28]:
```
aapl_nums.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
Int64Index: 26282 entries, 3654248 to 3294696
Data columns (total 15 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   adsh      26282 non-null  object
 1   tag       26282 non-null  object
 2   version   26282 non-null  object
 3   ddate     26282 non-null  datetime64[ns]
 4   qtrs      26282 non-null  int64
 5   uom       26282 non-null  object
 6   iprx      26282 non-null  int64
 7   value     26218 non-null  float64
 8   footnote  68 non-null     object
 9   footlen   26282 non-null  int64
 10  dimn      26282 non-null  int64
 11  coreg     0 non-null      object
 12  durp      26282 non-null  float64
 13  datp      26282 non-null  float64
 14  dcml      26282 non-null  int64
dtypes: datetime64[ns](1), float64(3), int64(5), object(6)
memory usage: 3.2+ MB
```

1.7 Create P/E Ratio from EPS and stock price data

We can select a useful field, such as Earnings per Diluted Share (EPS), that we can combine with market data to calculate the popular Price/Earnings (P/E) valuation ratio.

[29]:

```python
stock_split = 7
split_date = pd.to_datetime('20140604')
split_date
```

[29]:

```
Timestamp('2014-06-04 00:00:00')
```

We do need to take into account, however, that Apple split its stock 7:1 on June 4, 2014, and Adjusted Earnings per Share before the split to make earnings comparable, as illustrated in the following code block:

[30]:

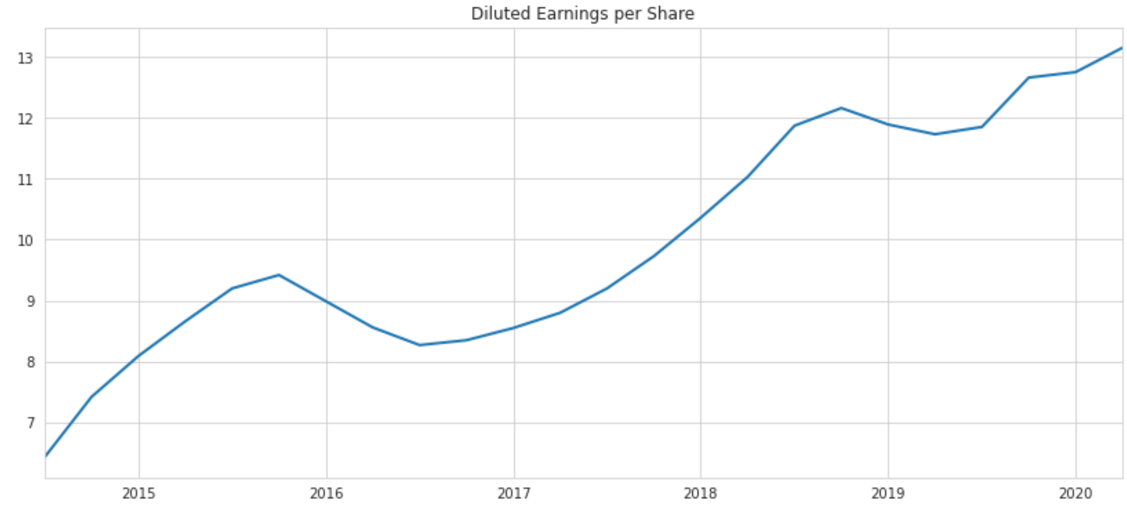
```python
# Filter by tag; keep only values measuring 1 quarter
eps = aapl_nums[(aapl_nums.tag == 'EarningsPerShareDiluted')
                & (aapl_nums.qtrs == 1)].drop('tag', axis=1)

# Keep only most recent data point from each filing
eps = eps.groupby('adsh').apply(lambda x: x.nlargest(n=1, columns=['ddate']))

# Adjust earnings prior to stock split downward
eps.loc[eps.ddate < split_date,'value'] = eps.loc[eps.ddate < split_date,␣
 ↪'value'].div(7)
eps = eps[['ddate', 'value']].set_index('ddate').squeeze().sort_index()
eps = eps.rolling(4,min_periods=4).sum().dropna()
```

[31]:

```
eps.plot(lw=2, figsize=(14, 6), title='Diluted Earnings per Share')
plt.xlabel('')
plt.savefig('diluted eps', dps=300);
```



[32]:

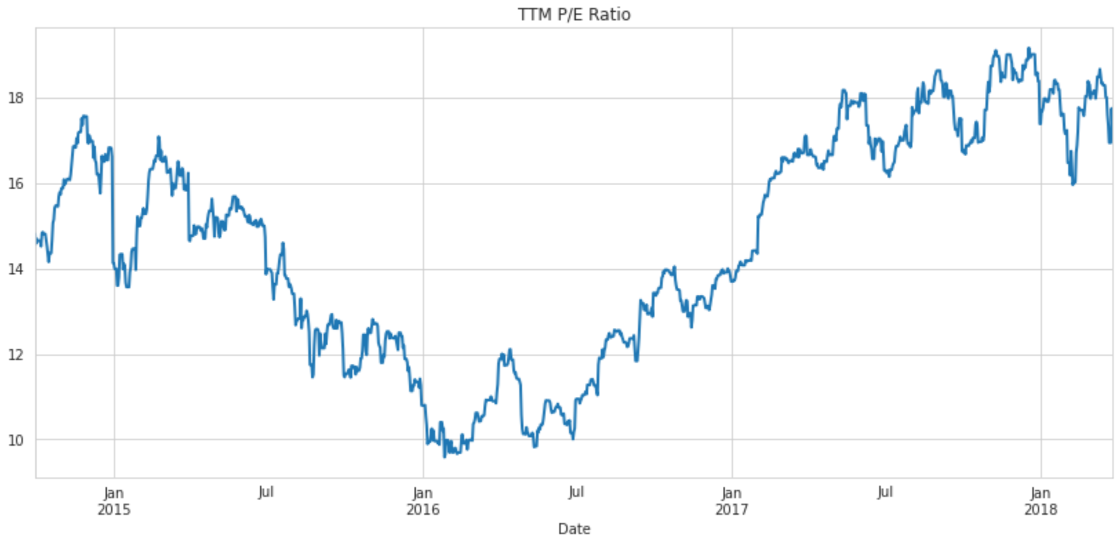
```
symbol = 'AAPL.US'

aapl_stock = (web.
              DataReader(symbol, 'quandl', start=eps.index.min())
              .resample('D')
              .last()
             .loc['2014':eps.index.max()])
aapl_stock.info()
```

```
<class 'pandas.core.frame.DataFrame'>
DatetimeIndex: 1275 entries, 2014-09-30 to 2018-03-27
Freq: D
Data columns (total 12 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   Open        877 non-null    float64
 1   High        877 non-null    float64
 2   Low         877 non-null    float64
 3   Close       877 non-null    float64
 4   Volume      877 non-null    float64
 5   ExDividend  877 non-null    float64
 6   SplitRatio  877 non-null    float64
 7   AdjOpen     877 non-null    float64
 8   AdjHigh     877 non-null    float64
 9   AdjLow       877 non-null    float64
 10  AdjClose     877 non-null    float64
 11  AdjVolume    877 non-null    float64
dtypes: float64(12)
memory usage: 129.5 KB
```

[33]:

```
pe = aapl_stock.AdjClose.to_frame('price').join(eps.to_frame('eps'))
pe = pe.fillna(method='ffill').dropna()
pe['P/E Ratio'] = pe.price.div(pe.eps)
pe['P/E Ratio'].plot(lw=2, figsize=(14, 6), title='TTM P/E Ratio');
```



[34]:

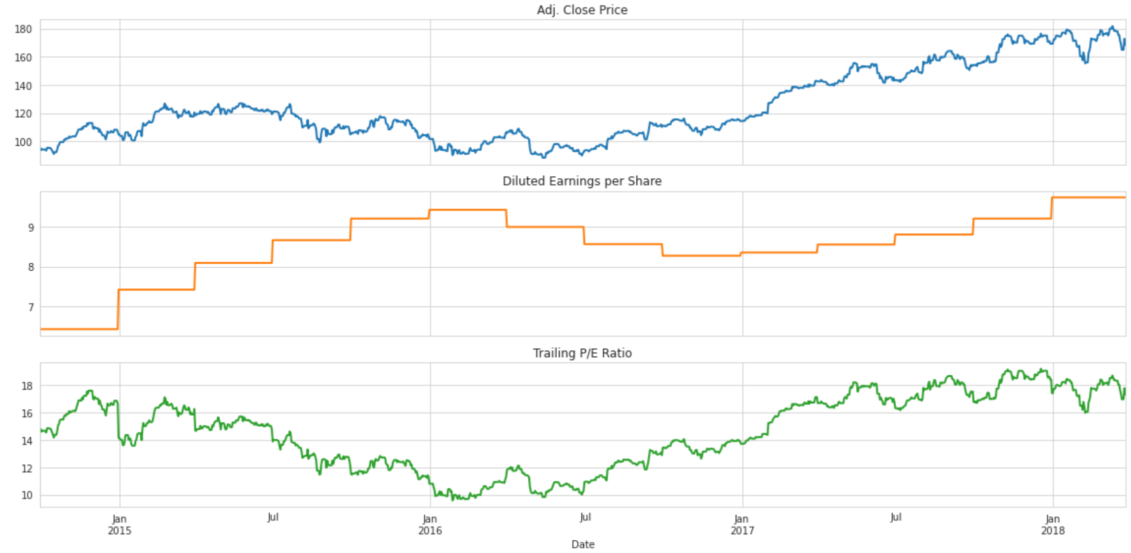
```
pe.info()
```

```
<class 'pandas.core.frame.DataFrame'>
DatetimeIndex: 1275 entries, 2014-09-30 to 2018-03-27
Freq: D
Data columns (total 3 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   price      1275 non-null   float64
 1   eps        1275 non-null   float64
 2   P/E Ratio  1275 non-null   float64
dtypes: float64(3)
memory usage: 39.8 KB
```

[35]:

```
axes = pe.plot(subplots=True, figsize=(16,8), legend=False, lw=2)
axes[0].set_title('Adj. Close Price')
axes[1].set_title('Diluted Earnings per Share')
axes[2].set_title('Trailing P/E Ratio')
```

```
plt.tight_layout();
```



1.8 Explore Additional Fields

The field `tag` references values defined in the taxonomy:

[36]:
```
aapl_nums.tag.value_counts()
```

[36]:
```
CashAndCashEquivalentsAtCarryingValue
810
DebtInstrumentInterestRateEffectivePercentage
782
OperatingIncomeLoss
650
AvailableForSaleSecuritiesNoncurrent
560
AvailableForSaleSecuritiesCurrent
560
                                                …
DeferredTaxAssetsGoodwillAndIntangibleAssets
1
ShareBasedCompensationArrangementsByShareBasedPaymentAwardOptionsGrantsInPeriodW
eightedAverageExercisePrice          1
TaxCutsAndJobsActOf2017MeasurementPeriodAdjustmentIncomeTaxExpenseBenefit
1
TaxCutsAndJobsActOf2017IncompleteAccountingTransitionTaxForAccumulatedForeignEar
ningsProvisionalLiability            1
ShareBasedCompensationArrangementsByShareBasedPaymentAwardOptionsExercisesInPeri
odWeightedAverageExercisePrice       1
```

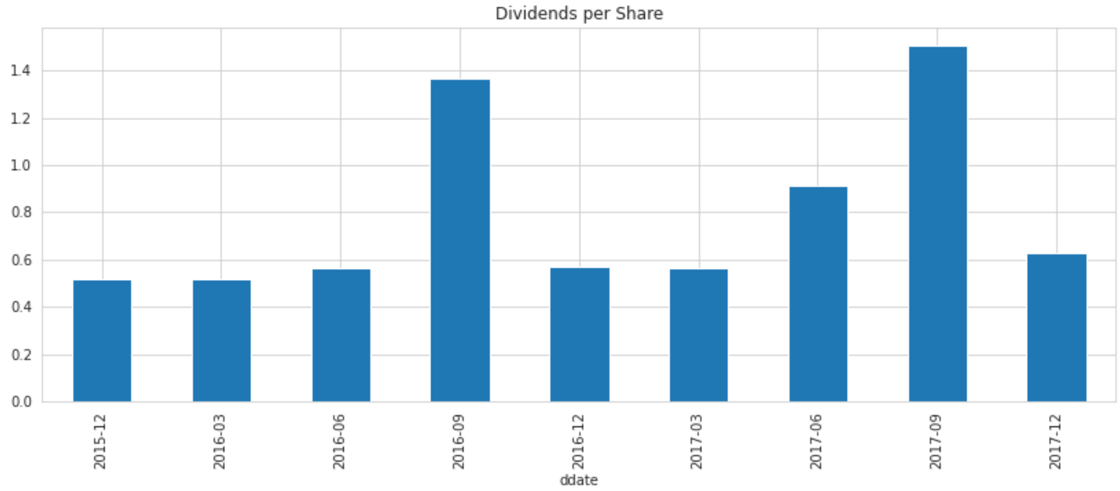
```
Name: tag, Length: 545, dtype: int64
```

We can select values of interest and track their value or use them as inputs to compute fundamental metrics like the Dividend/Share ratio.

1.8.1 Dividends per Share

```
[37]: fields = ['EarningsPerShareDiluted',
                'PaymentsOfDividendsCommonStock',
                'WeightedAverageNumberOfDilutedSharesOutstanding',
                'OperatingIncomeLoss',
                'NetIncomeLoss',
                'GrossProfit']
```

```
[38]: dividends = (aapl_nums
                   .loc[aapl_nums.tag == 'PaymentsOfDividendsCommonStock', ['ddate',␣
       ↪'value']]
                   .groupby('ddate')
                   .mean())
      shares = (aapl_nums
                .loc[aapl_nums.tag ==␣
       ↪'WeightedAverageNumberOfDilutedSharesOutstanding', ['ddate', 'value']]
                .drop_duplicates()
                .groupby('ddate')
                .mean())
      df = dividends.div(shares).dropna()
      ax = df.plot.bar(figsize=(14, 5), title='Dividends per Share', legend=False)
      ax.xaxis.set_major_formatter(mticker.FixedFormatter(df.index.strftime('%Y-%m')))
```



1.9 Bonus: Textual Information

```
[39]: txt = pd.read_parquet(data_path / '2016_2' / 'parquet' /  'txt.parquet')
```

AAPL's adsh is not avaialble in the txt file but you can obtain notes from the financial statements here:

```
[40]: txt.head()
```

```
[40]:
                     adsh                        tag   version     ddate  qtrs  \
0  0000799288-16-000122              AmendmentFlag  dei/2014  20160131     4
1  0000799288-16-000122   CurrentFiscalYearEndDate  dei/2014  20160131     4
2  0000799288-16-000122  DocumentFiscalPeriodFocus  dei/2014  20160131     4
3  0000799288-16-000122    DocumentFiscalYearFocus  dei/2014  20160131     4
4  0000799288-16-000122      DocumentPeriodEndDate  dei/2014  20160131     4

   iprx   lang   dcml      durp  datp        dimh  dimn coreg  escaped  \
0     0  en-US  32767  0.021918   2.0  0x00000000     0  None        0
1     0  en-US  32767  0.021918   2.0  0x00000000     0  None        0
2     0  en-US  32767  0.021918   2.0  0x00000000     0  None        0
3     0  en-US  32767  0.021918   2.0  0x00000000     0  None        0
4     0  en-US  32767  0.021918   2.0  0x00000000     0  None        0

   srclen  txtlen footnote  footlen      context       value
0       5       5     None        0  FD2015Q4YTD       false
1       7       7     None        0  FD2015Q4YTD     --01-29
2       2       2     None        0  FD2015Q4YTD          FY
3       4       4     None        0  FD2015Q4YTD        2015
4      10      10     None        0  FD2015Q4YTD  2016-01-29
```