

---

July 2020

## The Fidelity® Family of Funds –PEGY Estimation

Doug Lyon

Fairfield University, lyon@docjava.com

Follow this and additional works at: <https://digitalcommons.fairfield.edu/ijcase>

---

### Recommended Citation

Lyon, Doug (2020) "The Fidelity® Family of Funds –PEGY Estimation," *International Journal of Computer and Systems Engineering*: Vol. 1 : Iss. 1 , Article 3.

Available at: <https://digitalcommons.fairfield.edu/ijcase/vol1/iss1/3>

This item has been accepted for inclusion in DigitalCommons@Fairfield by an authorized administrator of DigitalCommons@Fairfield. It is brought to you by DigitalCommons@Fairfield with permission from the rights-holder(s) and is protected by copyright and/or related rights. **You are free to use this item in any way that is permitted by the copyright and related rights legislation that applies to your use. For other uses, you need to obtain permission from the rights-holder(s) directly, unless additional rights are indicated by a Creative Commons license in the record and/or on the work itself.** For more information, please contact [digitalcommons@fairfield.edu](mailto:digitalcommons@fairfield.edu).



## The Fidelity® Family of Funds – PEGY Estimation

Deriving the PEGY ratio from the Fidelity® Fund metrics with Web Data Miners

by Douglas Lyon, Predrag Bokšić

### Abstract

The Fidelity® network of mutual funds has \$2T under management and gives investors access to over 10,000 funds. We describe a technique for screening some of these funds using the PEGY ratio. We are motivated by our employers' constraint on the purchase of funds outside of the Fidelity® family of funds and by the prohibition on the purchase of individual stocks in our retirement account. Our approach has led to a deep dive into data mining techniques and the creation of a data mining framework.

### 1. Problem Statement

Given a large number of publicly-traded funds, we seek to compute a screening metric called the PEGY ratio:

$$PEGY = \frac{\frac{Price}{EPS}}{[EPS_{growth} + Yield]}$$

The dividend yield is defined as the cash paid to the owners of a share (stock) over a period of one year divided by the current stock price, stated as a percentage.

The P/E ratios that use the expected (future) earnings per share are forward-looking. The current company earnings-per-share are also called the “trailing” earnings, denoted by the “EPS (ttm)”. These typically reflect the previous-year-to-date earnings [Fink] [Investopedia] [Lynch]. The PEG ratio does not take into account the dividend yield as the PEGY does [Smith] [Barnes]. Typically, PEG ratio includes the trailing



P/E ratio divided by the expected EPS growth for the next 5 years and funds often do not publish a complete list of the company stock tickers that they hold [Barnes].

## 2. Approach

Consider the N-Q form with the U.S. Securities and Exchange Commission (SEC.gov):

### Investments July 31, 2017 (Unaudited)

Showing Percentage of Net Assets

<b>Common Stocks - 97.9%</b>		
	<b>Shares</b>	<b>Value</b>
<b>CONSUMER DISCRETIONARY - 3.6%</b>		
<b>Auto Components - 0.2%</b>		
Magna International, Inc. Class A (sub. vtg.)	62,900	\$2,999,827
<b>Diversified Consumer Services - 0.0%</b>		
Park Lawn Corp.	24,850	378,705
<b>Hotels, Restaurants &amp; Leisure - 0.9%</b>		
Cara Operations Ltd.	138,900	2,612,557
Freshii, Inc. (a)	352,608	2,850,843
Paddy Power Betfair PLC (Ireland)	55,700	5,580,964
		11,044,364
<b>Media - 2.0%</b>		
Cogeco Communications, Inc.	125,700	8,781,608
Corus Entertainment, Inc. Class B (non-vtg.)	588,000	6,527,307
Quebecor, Inc. Class B (sub. vtg.)	312,400	10,827,194
		26,136,109
<b>Specialty Retail - 0.5%</b>		
AutoCanada, Inc. (a)	81,500	1,312,629
Sleep Country Canada Holdings, Inc.	162,600	4,946,796
		6,259,425
TOTAL CONSUMER DISCRETIONARY		46,818,430
<b>CONSUMER STAPLES - 7.2%</b>		
<b>Food &amp; Staples Retailing - 6.8%</b>		
Alimentation Couche-Tard, Inc. Class B (sub. vtg.)	705,900	33,473,277
George Weston Ltd.	301,700	26,350,201

By parsing this and other similar pages, we gain insight into fund composition and estimate the weight of a particular asset in the fund's portfolio.



Our system automatically finds the fund composition in the form of a list of companies denoted by their names and stock tickers. In that regard, the public documents at the SEC.gov such as the N-Q form represent a starting point.

In some cases, the holding is not publicly traded and thus cannot contribute to the PEGY computation [Investopedia].

The N-Q form has no ticker symbols, only company names; requiring matching of company names with ticker symbols. Our system matches N-Q company names with ticker symbols in order to calculate the PEGY for the portfolio companies. Prorating PEGY by holding percentage. The NAts (the Net Assets able to be assessed) is our metric of certainty. If the PEGY can be determined, the NAts increases. In some cases, we will not be able to find the holding in the publicly traded US markets. In those cases, the holding does contribute to our estimate of the PEGY ratio [Two Cents].

Our mining uses the Burkhard-Keller string matching algorithm and calculates the Levenshtein distance from the exact word match. However, when we match “chemed corp” and “Cheetah Mobile Inc.”. The safest value is a distance less than or equal to 3. The company name matching in this form represents the weakest point of the entire mining project. Even when tuned to a “safe distance value”, the name matcher remains questionable.

CIK numbers are given to companies in the SEC EDGAR database at <https://www.sec.gov/Archives/edgar/cik-lookup-data.txt> [SEC]. We also resolve name ambiguity by using Yahoo Search Suggestions [Yahoo]. A list of company names, tickers, and CIK numbers can be downloaded from this address [Rank and Filed]. Whereas, the EDGAR database offers an exported list of company names and CIK numbers in the CIK-lookup database [CIK]. The file contains 720634 lines in a “:” delimited file. Comparatively, the cik\_ticker.csv file from the rankandfiled.com [Rank and Filed] website contains 13738 entries. The file also contains the name of the public stock exchange at which the stock ticker (symbol) is registered, a vital detail whose absence might lead to a few mismatches.



A significantly reduced list of company stock tickers can be obtained from The Wall Street Journal at AMEX.csv [AMEX], NYSE.csv [NYSE], and Nasdaq.csv [Nasdaq]. For reference, a list of OTC stocks can be examined at otcmarkets.com [OTC Markets].

Once we compute the PEGY ratio, we complete the picture of the fund's portfolio.

The question of how we obtain the financial metric data about the company stocks such as the P/E ratio, EPS for the next 5 years, Dividend yield, PEG, Price will be addressed in the following section [Lynch] [Lyon A] [Lyon B].

### 3. Downloading the Financial Metrics

This section shows how we mine the PEGY value using *Zacks.com*. As a preliminary step, we first mine stock metrics from "Financial Visualizations" website, *finviz.com* [Barnes]. The Java class dedicated to this task is found in

```
pb/finance/finviz/QueryCompanyTickersAndStatsFromFinviz.java.
```

The *finviz* query URL for stock metrics web scraping is:

<https://finviz.com/screener/ashx?v=150&r=1&c=0,1,2,7,9,14,20,48,65>

This URL returns a list of 7458 company stocks with properties, in the following order:

No	Ticker	Company	Trailing P/E	PEG	Dividend	EPS next 5Y	Beta	Price
----	--------	---------	--------------	-----	----------	-------------	------	-------

The  $r=1$  in the URL means that the URL will open the first page. There may be hundreds of pages. Other parameters denoted with  $c=0,1,2,\dots$  represent the selected columns of the data table at the *finviz.com* website. The *finviz.com* pages are parsed with the Jsoup library,

*"org.jsoup:jsoup."*



From the resulting data table, produce the *pegMiners.csv* file with data in the format:

Ticker | Company | Trailing P/E | PEG | Dividend | EPS next 5Y | Beta | Price | PEGY

The CSV parsing (reading and writing) is performed by the Univocity library,

```
com.univocity:univocity-parsers [Univocity].
```

The CSV parser settings used include:

```
parserSettings.getFormat().setDelimiter('|');  
parserSettings.setHeaderExtractionEnabled(false);  
parserSettings.setNullValue("-");
```

The *pegMiner.csv* file can be reused for the fund research while the stock prices last. The sample of the file follows below.

```
A|Agilent Technologies,  
Inc.|26.32|2.428044280442804|0.89|10.84|1.38|67.26|2.243819  
266837170  
  
AA|Alcoa Corporations|52.02|-28.9|0|-1.80||44.37|-28.9  
  
AABA|Altaba Inc.|2.46|-2.963855421686747|0|-  
0.83|1.92|69.33|-2.963855421686747  
  
...
```

The numbers such as *P/E*, *PEG*, and *Dividend Yield* are always positive or undefined. The *EPS* for the next 5 years can be positive, negative, or undefined, which, in turn, impacts the *PEGY* values.



*Finviz.com* also allows  $EPS\_5y > 0$  in combination with  $PEG = NaN$ , in some cases. We hypothesize that the analysts are willing to assert the earnings-per-share growth for the next 5 years, even in cases that lack other data. The fund scan at *Zacks.com* is completed by reading the `pegMiner.csv` file using:

```
pb/finance/zacks/com/QueryFundTickersFromZacksMinimal.java
```

We calculate the fund PEGY as the weight sum of the PEGY values of the included company stocks. A typical fund page on the *Zacks.com* website tells us the weight (percentage) for each holding. The list of all Fidelity® funds can be obtained in a CSV file from

<https://fundsresearch.fidelity.com/mutual-funds/fidelity-funds-daily-pricing-yields/download>.

The header format is:

```
Fund Name,Symbol,NAV($),Last Dividend,Dividend Date,NAV Change($),30-Day Yield(%),7-  
Day Yield(%),7-Day Effective Yield(%),Morningstar Investment Category,Morningstar Overall  
Rating,# of Funds Rated,Date
```

The fund name is given inside quotation marks. The file loading procedure has been placed in the method `getSymbolMap` in

```
pb.finance.zacks.com.QueryFundTickersFromZacksMinimal#getSym  
bolMap
```

Each line in the file can be processed with the `Pattern/Matcher` classes by using:

```
Pattern pattern = Pattern.compile("^\"(.*)\"", ([A-Z]{5}), "");  
...  
Matcher matcher = pattern.matcher(line);  
if (matcher.find()) {  
    String fundName = matcher.group(1);  
    String symbol = matcher.group(2);  
    ...  
}
```



The `pegMiner.csv` file is loaded in:

`pb.finance.zacks.com.Query.FundTickersFromZacksMinimal#readFilePegMinerDataFromUrl`.

This method reads a local `pegMiner.csv` file if present from the local directory, for testing purposes. Otherwise, the default choice is to query the address:

[https://www.bondbutcher.com/comput\\_1/peg/fidelity-web/pegMiner.csv](https://www.bondbutcher.com/comput_1/peg/fidelity-web/pegMiner.csv). The corresponding web folder is `/var/www/html/output_1/peg/fidelity-web`.

On a server with very little memory, where 256Mb of free memory is not guaranteed at all times, the `QueryFundTickersFromZacksMinimal` class may run out of memory. The OOM killer, a process by Linux kernel that hunts for memory-demanding applications could kill the web scraping program. If the free memory falls below 40Mb-50Mb, the application is in danger of shutting down. The class was optimized for memory consumption and the first library to be removed was Gson, from: `com.google.code.gson:gson`, along with the program's JSON cache features.

If we used the list of all known companies and their CIK numbers, the file is 701,850 lines. The file location is <https://www.sec.gov/Archives/edgar/cik-lookup-data.txt>. Such a large file could be read sequentially using: <https://codereview.stackexchange.com/questions/44021/fast-way-of-searching-for-a-string-in-a-text-file>. In fact, by upscaling the project to read and write large files, we would need to generate the search indexes and use a different approach, which by itself is memory demanding. See the sequential reading example in `pb.finance.pegMiner.SequentialFileReader`.

The base URL for Zacks.com scraping comes in the form of

<https://www.zacks.com/funds/mutual-fund-quote/SYMBOL/holding>. In this page, we find the list of holdings (i.e. company stocks included in the fund) according to the Zacks.com research.



The data table contains a weight column of a particular interest. The weight is a percentage that expresses a prorated portion of the portfolio.

The page can be parsed manually by noting that the data table contained within the HTML page is preloaded in its entirety via JavaScript variable `document.table_data = [ [ ... ] ];`

The [Yahoo Finance](#) web portal was the centerpiece of mining for ages. You can find examples of making the “stock queries” in all programming languages.

The large data mining operations are unfavorably perceived by the Yahoo servers and the frequent user visitations are obstructed. Typically a HTTP 503 error will stall the mining process completely and require a resting period in hours.

The data-mining program may recover quickly from an error such as a *java.net.ProtocolException*. The program recovers from this error and continues. A random number of such errors makes the acquired data patchy.

Long stock ticker lists and OTC stock ticker queries produce response failures from the Yahoo Finance. These errors ruin the reputation of the originating IP address of the mining program computer. Another possible error the *404 error*, is the logical equivalent of a file not found exception.

Alternatives to Yahoo include [Morningstar.com](#) and [FinViz](#). [FinViz](#) serves a large amount of data that enables computation of the PEGY value. You can see some of the basic stock details of about 7586 companies at <https://finviz.com/screener.ashx?v=150&r=1&c=0,1,2,7,9,14,20,48,65>.



The [Zacks](#) website collects details of several Fidelity mutual funds and their included assets. This website displays the company stock weights in the given fund [Zacks]. Zacks has 207 but does not cover 202. Several stock tickers are listed numbers that have no clear interpretation. We speculate that these represent the references to the international stocks in some cases, but the website lacks documentation on that.

By combining the [SEC.gov](#) and [FinViz](#) data, we can complete the PEGY ratio sums for the available company stocks under the fund management, thus avoiding having to go to the [SEC.gov](#) web site. This approach is more straightforward than the analysis of the N-Q form on the [SEC.gov](#) website.

A few samples below include a data table that stems from the [SEC.gov](#) mining, followed by a data table obtained through [Zacks](#). The columns denote PEGY, NATs, Sharpe Ratio for a 3-year period, 5-year period, 10-year period, Beta for a 3-year period, Ticker, Name, and 401k qualified status.

PEGY	NAts	SR3	SR5	SR10	Beta3	Ticker	Name	401k qualified
0.94	0.80	0.64	0.21	3.0	1.01	FSAVX	Fidelity Select Automotive Portfolio	Yes
0.98	0.86	0.85	0.49	0.84	1.24	FSRBX	Fidelity Select Banking Portfolio	Yes
0.98	0.46	0.42	0.03	5.0	0.76	FIVLX	Fidelity International Value Fund	Yes
1.00	0.20	0.77	0.76	2.0	0.93	FTBFX	Fidelity Total Bond Fund	Yes
1.05	0.79	0.97	0.35	3.0	1.03	FSLBX	Fidelity Select Brokerage and Investment Management Portfolio	Yes
1.12	0.10	N/A	N/A	N/A	NaN	FNSOX	Fidelity Short-Term Bond Index Fund	No
1.16	0.53	NaN	NaN	NaN	0.81	FDMLX	Fidelity Series Intrinsic Opportunities Fund	No

PEGY	NAts	SR3	SR5	SR10	Beta3	Ticker	Name	401k qualified
0.904	76.84	1.22	0.63	3	1.12	FSVLX	Fidelity Select Consumer Finance Portfolio	Yes
0.991	81.88	1.04	0.77	2	0.94	FEXPX	Fidelity Export and Multinational Fund	No
1.017	88.588	0.9	0.85	2	1.15	FSHCX	Fidelity Select Health Care Services Portfolio	Yes



1.019	82.231	1.08	0.74	2	0.92	FDGFX	Fidelity Dividend Growth Fund	No
1.029	27.6	1.22	1.07	3	0.33	FRIFX	Fidelity Real Estate Income Fund	Yes
1.03	65.357	0.83	0.55	NaN	0.81	FGILX	Fidelity Global Equity Income Fund	Yes
1.059	80.69	1.47	0.92	1	1.24	FSDAX	Fidelity Select Defense and Aerospace Portfolio	Yes
1.064	66.897	0.81	0.62	3	0.89	FBCVX	Fidelity Blue Chip Value Fund	Yes
1.079	4.752	0.91	0.21	4	0.72	FEMEX	Fidelity Emerging Europe, Middle East, Africa (EMEA) Fund	Yes
1.094	51.185	0.77	0.17	5	0.86	FICDX	Fidelity Canada Fund	Yes
1.118	93.681	0.64	0.21	3	1.01	FSAVX	Fidelity Select Automotive Portfolio	Yes
1.13	85.341	1.01	0.77	2	0.91	FSPCX	Fidelity Select Insurance Portfolio	Yes
1.138	91.591	0.95	0.75	2	1.29	FSAIX	Fidelity Select Air Transportation Portfolio	Yes
1.142	78.14	0.94	0.62	NaN	0.88	FNKIX	Fidelity Series Value Discovery Fund	No

## 4. Java Libraries

Java libraries that we used include Jsoup, Freemarker, and Univocity Parsers. The Jsoup enables the selection of HTML table elements by means of `cssQuery`. The Univocity Parsers are used to read and write CSV files with headers and embedded comments. The FreeMarker library is used to generate HTML. The list of libraries by Maven coordinates follows:

```
com.google.code.gson:gson:2.8.2
com.opencsv:opencsv:4.1
com.univocity:univocity-parsers:2.5.9
commons-io:commons-io:2.6
commons-lang:commons-lang:2.6
commons-logging:commons-logging:1.2
freemarker:freemarker:2.3.8
org.apache.httpcomponents:httpclient:4.5.5
org.apache.httpcomponents:httpcore:4.4.9
org.jsoup:jsoup:1.11.2
```

## 5. Summary

We described a series of technologies that were able to mine financial data in order compute a simple financial metric (the PEGY ratio). The computation was simple enough, but obtaining the data was anything but simple, or easy. Our approach to automation has taken us on a journey through a series of data mining frameworks that had a varying levels of effectiveness. In the future, we think that the downloading and computing of financial metrics will be simplified by the trail that we have blazed here. One thing we found remarkable is the number of companies in funds (LIKE FSRBX, the select banking



fund from Fidelity) that had earnings losses and negative growth rates. These companies turn the holding funds into value traps which would have deceptively low PEGY ratios, if the negative earnings were allowed to impact the average PEGY. Our system takes hours to complete in its run, but we need only run it once a day; which we do, as a cron job. You may view the updated output of our system at: [https://www.bondbutcher.com/comput\\_1/peg/fidelity-web/index2.html](https://www.bondbutcher.com/comput_1/peg/fidelity-web/index2.html)

In retrospect, the most efficient data mining approach downloaded all the metrics for the publicly traded stocks at once, from *finviz*. Then the holdings, with tickers and percentages, are available for each fund from *zacks*. Individual stock queries were just too time-consuming and raised denial of service attack flags that triggered IP address blocking (something we try to avoid).

## 6. References

[AMEX] AMEX <http://online.wsj.com/public/resources/documents/AMEX.csv>

[Barnes] “PEG Ratio Nails Down Value Stocks” by Ryan Barnes, <https://www.investopedia.com/articles/analyst/043002.asp>. Last accessed 3/19/18.

[CIK] CIK Lookup Database <https://www.sec.gov/Archives/edgar/cik-lookup-data.txt>.

[Estrada] “Adjusting P/E ratios by growth and risk: the PERG ratio” by Javier Estrada <https://pdfs.semanticscholar.org/e7bd/cd6bedcfa527740ee2cdc7c70843ff948a15.pdf>

[Fidelity®] Fidelity® funds, <https://www.fidelity.com/mutual-funds/all-mutual-funds/overview>.

[FinViz] Financial Visualizations, <https://finviz.com/>.

[Fink] “The PEGY Ratio” by Jim Fink <https://www.investingdaily.com/11091/the-pegy-ratio>

[GAA Accounting] <http://www.gaaaccounting.com/earnings-per-share-as-a-measure-of-financial-performance-do-we-interpret-it-appropriately>

[Investopedia] <https://www.investopedia.com/ask/answers/buy-over-the-counter-stock>

[Investopedia] <https://www.investopedia.com/terms/d/dilutedeps.asp>

[Investopedia] <https://www.investopedia.com/terms/p/pegratio.asp>

[Lynch] “One up on wall Street” by Peter Lynch, Simon & Shuster, 2<sup>nd</sup> Edition, 2000.



[Lyon A] Lyon, Douglas. (2008). Mining Edgar Tender Offers. Journal of Object Technology. 7. 17-31. 10.5381/jot.2008.7.7.c2. <http://www.docjava.com/pub/document/jot/v7n7.pdf>

[Lyon B] Lyon, Douglas. (2008). Multi-threaded data mining of EDGAR CIKs (Central Index Keys) from ticker symbols. 1 - 7. 10.1109/IPDPS.2008.4536453. <http://www.docjava.com/pub/document/jot/pdcof.pdf>

[Morningstar] <http://www.morningstar.co.uk/uk/news/105269/the-limitations-of-earnings-per-share.aspx>

[Nasdaq] Nasdaq, <http://online.wsj.com/public/resources/documents/Nasdaq.csv>.

[NYSE] NYSE, <http://online.wsj.com/public/resources/documents/NYSE.csv>.

[OTC Markets] <https://www.otcmarkets.com/>

[Rank and Filed] <http://rankandfiled.com/>

[Rank and Filed] [http://rankandfiled.com/static/export/cik\\_ticker.csv](http://rankandfiled.com/static/export/cik_ticker.csv)

[SEC] <https://www.sec.gov/Archives/edgar/cik-lookup-data.txt>

[Smith] “How to Find P/E and PEG Ratios”, by Thomas Smith, Last accessed 12/12/17, <https://www.investopedia.com/articles/fundamental-analysis/09/price->

[Two Cents] <https://twocents.lifehacker.com/how-to-see-whats-actually-inside-your-mutual-fund-1823329898>

[Univocity] Univocity Library, <https://github.com/uniVocity/univocity-parsers>.

[Yahoo] Yahoo Search Suggestions, [https://www.yahoo.com/?err=404&err\\_url=https%3A%2F%2Ffinance.yahoo.com%2F\\_finance\\_doubledown%2Fapi%2Fresource%2Fsearchassist%3BsearchTerm%3D](https://www.yahoo.com/?err=404&err_url=https%3A%2F%2Ffinance.yahoo.com%2F_finance_doubledown%2Fapi%2Fresource%2Fsearchassist%3BsearchTerm%3D)

[Zacks] Zacks, <https://www.zacks.com/>

## Author information

Douglas A. Lyon (M'89-SM'00) received the Ph.D., M.S. and B.S. degrees in computer and systems engineering from Rensselaer Polytechnic Institute (1991, 1985 and 1983). Dr. Lyon is Professor of Electrical and Biomedical Engineering at Fairfield University. He has worked at AT&T Bell Laboratories at Murray Hill, NJ and the Jet Propulsion Laboratory at the California



Institute of Technology, Pasadena, CA, a senior member of the IEEE and President of DocJava, Inc., a consulting firm in Connecticut. Dr. Lyon has authored or co-authored four books and over 50 journal publications. Email: [lyon@docjava.com](mailto:lyon@docjava.com). Web: <http://www.DocJava.com>.

Predrag L. Bokšić, professor of physics (PMF Novi Sad, Serbia in 2008). He is a Java programming enthusiast with particular interests in computer graphics, chaos theory, and AI. He works in the localization and web services testing areas.

dd information about yourself here.