In [42]:
```
import pandas as pd
import numpy as np
import plotly.offline as plotly
import plotly.graph_objs as pgo


#from plotly.offline import download_plotlyjs, init_notebook_mode, plot,
#from plotly.graph_objs import Scatter, Figure, Layout
plotly.init_notebook_mode(connected=True)
```

In [43]:
```
# Load all Data -- only some columns
pfd = pd.read_csv('../PFDrill/Goal1_PFD_1of3.csv.gz', compression='gzip',
                  usecols=['CID', 'Chamber', 'CalendarYear', 'AssetSource
                  dtype={'Asset4Date':str})
# calendar years start from 0
pfd.CalendarYear += 2000
# some Chambers are lowercase
pfd.Chamber = pfd.Chamber.str.upper()

# Members of Congress
pfdc = pfd.loc[pfd.Chamber.isin(['H', 'S'])]

candidates = pd.read_csv('../PFDrill/Cands.csv.gz', compression='gzip', e
                    usecols=['CID', 'Firstname', 'Lastname', 'Gender'],
                        index_col='CID')
candidates['Name'] = candidates.Firstname + ' ' + candidates.Lastname
```

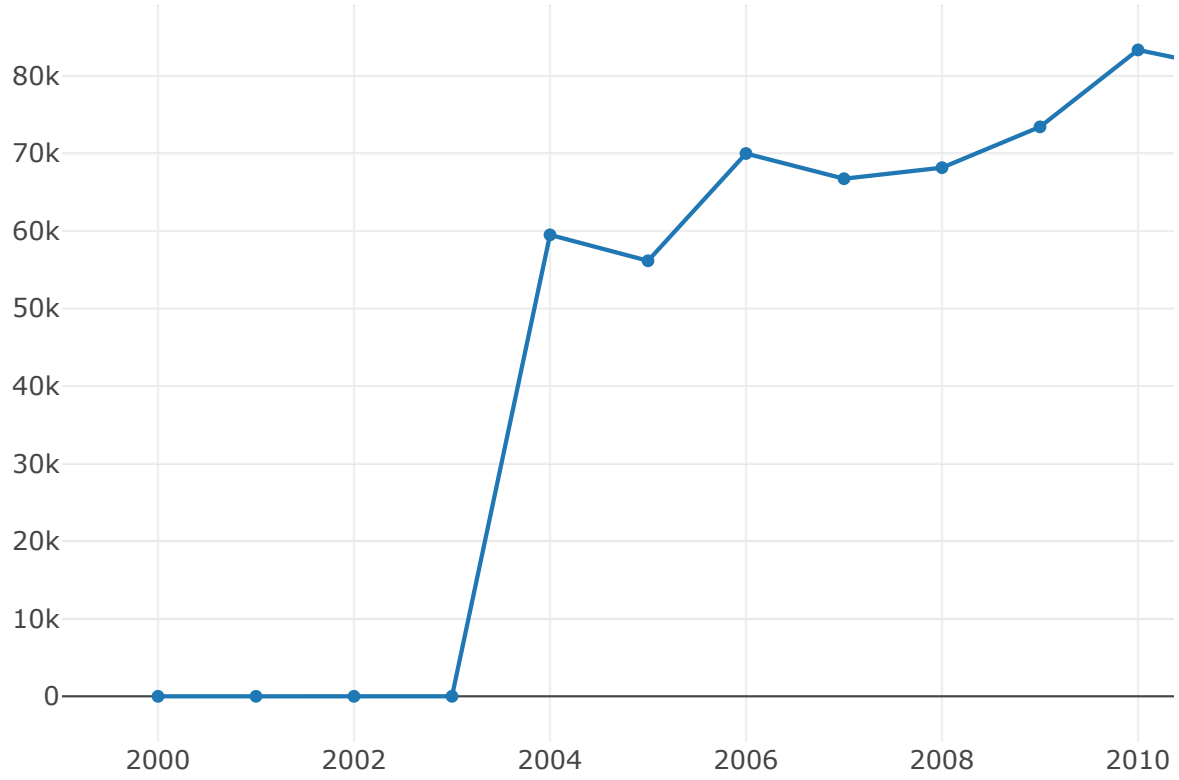
Totals

In [44]:
```
print('Total PFDs %d' % pfd.shape[0])
print('Total Members %d' % candidates.shape[0])
```

```
Total PFDs 796901
Total Members 35314
```

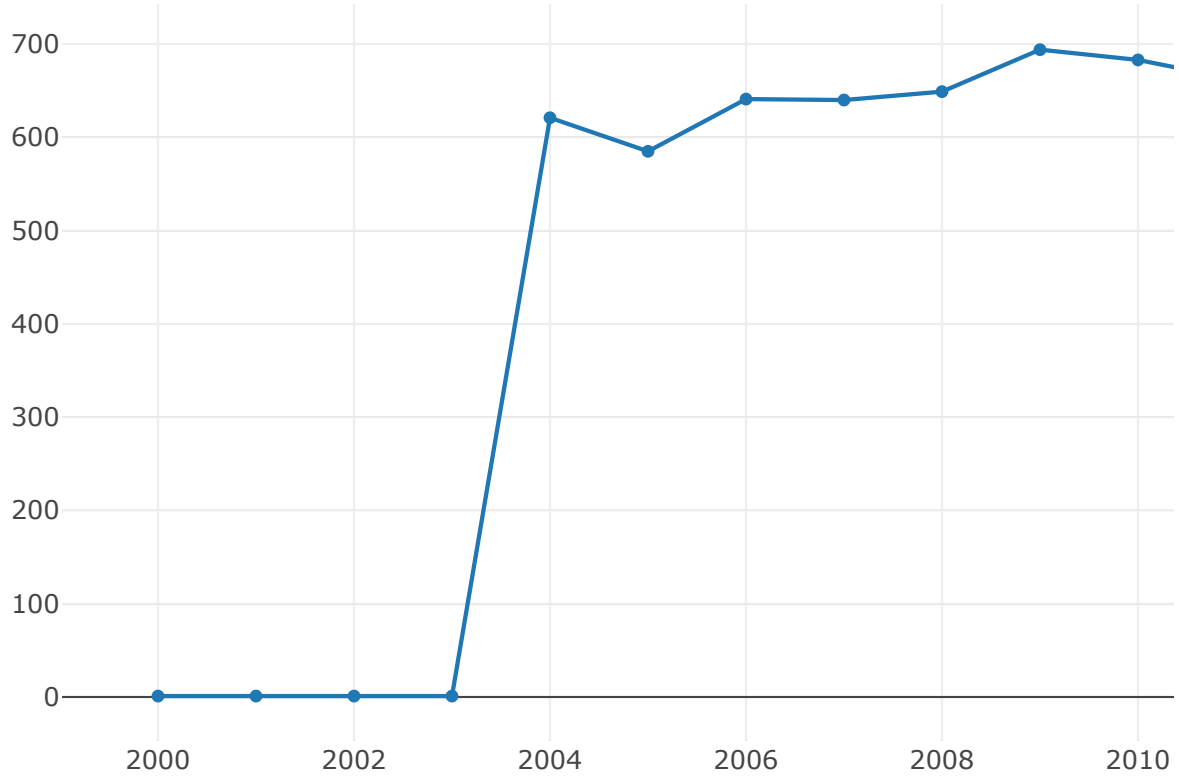
PFDs by year

In [45]:
```
d = pfd.groupby(['CalendarYear']).size()
plotly.iplot([{"x": d.index, "y": d}])
```



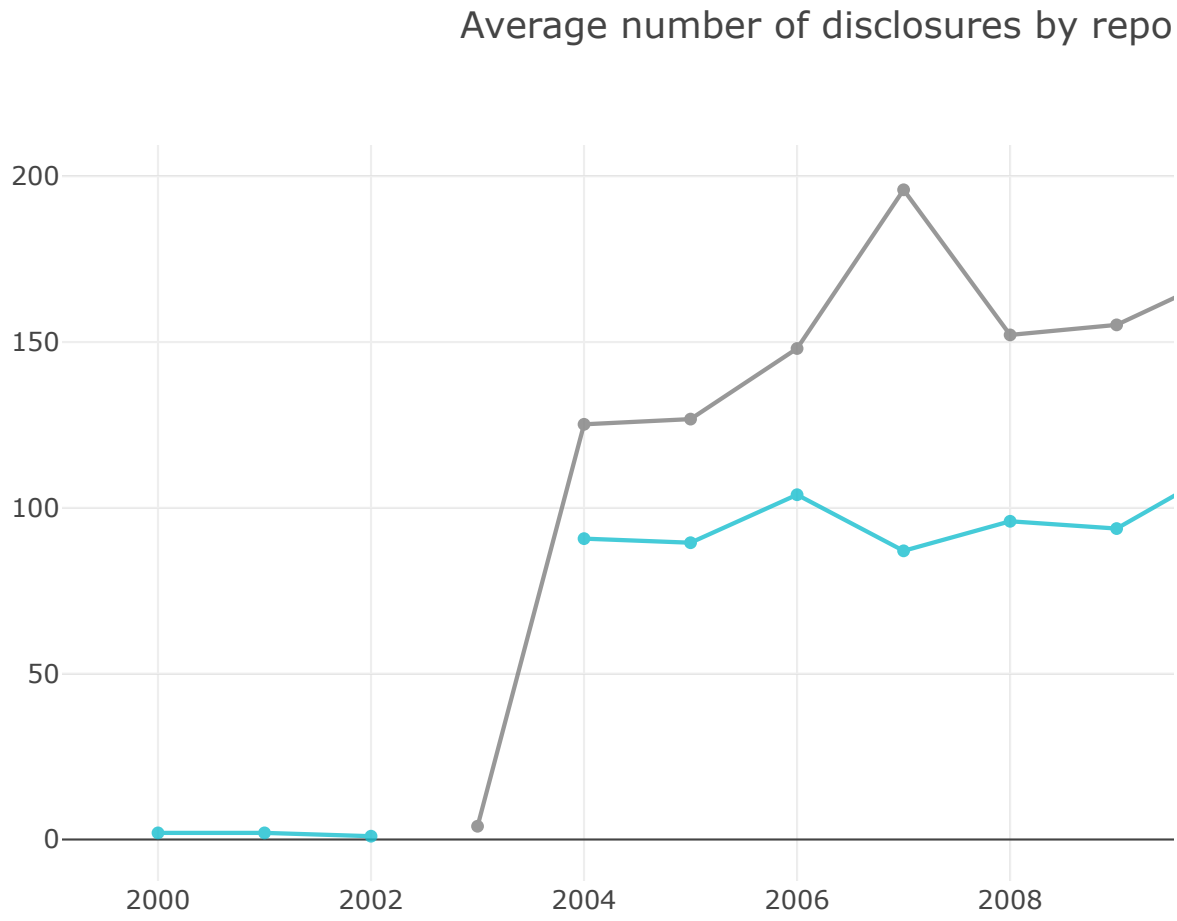
Number of people reporting by year

In [46]:
```
d = pfd.groupby('CalendarYear').CID.nunique()
plotly.iplot([{"x": d.index, "y": d}])
```

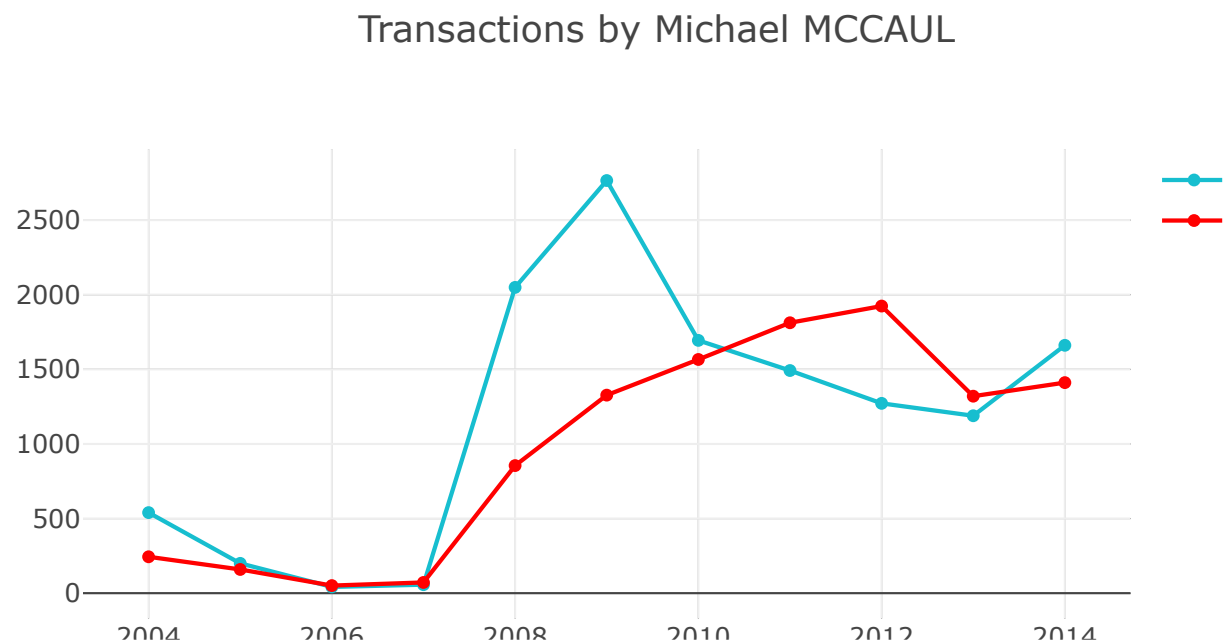


In [47]:
```
#TODO: there's definitely a better way to do this
# count number of reporting members
df1 = pfdc.groupby(['CalendarYear', 'Chamber']).CID.nunique().unstack()
df1.columns = ['House', 'Senate']
# count total numbers by year and chamber
df2 = pfdc.groupby(['CalendarYear', 'Chamber']).size().unstack()
df2.columns = ['House', 'Senate']
df2.House /= df1.House
df2.Senate /= df1.Senate

h = pgo.Scatter(x=df2.index, y=df2.House, name = "House", line = dict(col
                opacity = 0.8)
s = pgo.Scatter(x=df2.index, y=df2.Senate, name = "Senate", line = dict(c
                opacity = 0.8)
```

```
layout = dict(title = "Average number of disclosures by reporting member"

fig = dict(data=[h,s], layout=layout)
plotly.iplot(fig, filename = "")
```



In [48]:
```
buy_sell = pfdc[pfdc.TransactionType.isin(['Purchased', 'Sold'])]
N = 15
topN = buy_sell\
    .groupby(['CID']).size()\
    .sort_values(ascending=False)\
    .head(N)\
    .index
df = buy_sell.loc[buy_sell.CID.isin(topN)]\
    .groupby(['CID', 'CalendarYear', 'TransactionType']).size()

def member_buy_sell_graph(i):
    d = buy_sell.loc[buy_sell.CID == topN[i]]\
        .groupby(['CalendarYear', 'TransactionType'])\
        .size()\
        .unstack()
    d = pd.DataFrame(d.to_records())
    d.set_index(d.CalendarYear)
    b = pgo.Scatter(x=d.CalendarYear, y=d.Purchased, name = "Purchased",
    s = pgo.Scatter(x=d.CalendarYear, y=d.Sold, name = "Sold", line = dic
    name = candidates[candidates.index == topN[i]].iloc[0].Name
    layout = pgo.Layout(title = "Transactions by " + name , autosize=True
        width=700,
        height=400)
    fig = dict(data=[b,s], layout=layout)
    plotly.iplot(fig, filename = "test")

for i in range(N):
    member_buy_sell_graph(i)
```



In [49]:
```
names =  candidates.loc[topN].Name
#names

import ipywidgets as widgets
from ipywidgets import interactive

import qgrid

def show_buy_sells(name):
    cid = names.index[names == name]
    d = buy_sell[buy_sell.CID == cid[0]]
    d = d.drop(['CID', 'Chamber'], axis=1).reset_index()
    grid_options = {'forceFitColumns': True, 'editable': False,
                    'enableAddRow': 0, 'autoEdit': False}
    r = qgrid.show_grid(d, show_toolbar=True, grid_options={'forceFitColu
                    'enableAddRow': False, 'autoEdit': False})
    return display(r)

w = widgets.Dropdown(options=names.tolist())
widgets.interactive(show_buy_sells, name=w)
#topNdf = pd.DataFrame({'Name' : names})
```

name Michael MCCAUL

Add Row | Remove Row

index	CalendarYear	AssetSource

13944	2010	LLM Family Investments II Ab…
13945	2010	LLM Family Investments II Ab…
13946	2010	LLM Family Investments II Ab…

In [ ]:

In [ ]: