Sumitha_John_Allen_P1 (/github/TheRealAllenZ/Sumitha_John_Allen_P1/tree/ceedd5d6083ad44bf2e9585704bff9d1d8bff9ef)
/ ALLEN (/github/TheRealAllenZ/Sumitha_John_Allen_P1/tree/ceedd5d6083ad44bf2e9585704bff9d1d8bff9ef/ALLEN)
/ FILES (/github/TheRealAllenZ/Sumitha_John_Allen_P1/tree/ceedd5d6083ad44bf2e9585704bff9d1d8bff9ef/ALLEN/FILES)

In [51]:
```
import os
import csv
from pathlib import Path
import pandas as pd
import plotly.express as px
pd.set_option("display.max_rows", None, "display.max_columns", None)

from bokeh.plotting import figure, output_file, show
import numpy as np
#from bokeh.transform import linear_cmap
#from bokeh.util.hex import hexbin
import holoviews as hv
hv.extension('bokeh')
import hvplot.pandas


from bokeh.resources import INLINE
import bokeh.io

bokeh.io.output_notebook(INLINE)

%matplotlib inline
```

(https://bokeh.org) BokehJS 2.2.3 successfully loaded.

CLOSING PRICES - <<<*PERIOD = SHORT TERM*>>>

* To see the <u>IMMEDIATE</u> impact of the FB boycott, we've selected a period of
* 3 weeks from June 24th (26th was the 1st day of announcement) through to July 17th
* As a Snapshot in TIME

In [52]:
```
path = Path('closing_prices2020.csv')
close2020= pd.read_csv(path, delimiter=',')
close2020.set_index('Unnamed: 0',inplace=True)
close2020.head()
```

Out[52]:

Unnamed: 0	HMC	BBY	BLNK	ANTM	KO	CTVA	CVS	LEVI	SAM	EPC	ETSY	EXC	F	FOSL	FMS	GME	CL	HP	SJM	EPC.1
2020-06-01	26.37	77.50	1.7800	290.08	46.98	26.80	65.64	13.81	569.26	30.03	78.59	38.84	5.870	3.12	42.60	4.14	72.73	20.27	115.35	30.03
2020-06-02	26.48	80.55	1.7800	294.25	46.89	28.01	66.86	14.28	568.34	29.88	81.84	38.94	5.900	3.44	43.35	4.18	72.43	21.79	115.18	29.88
2020-06-03	27.00	82.23	1.8899	289.10	47.90	28.48	66.87	15.34	555.93	30.13	80.61	39.91	6.190	3.81	43.32	4.45	71.65	22.69	114.39	30.13
2020-06-04	26.93	79.92	2.1674	282.09	47.92	30.18	66.52	15.44	529.92	29.78	80.25	38.98	6.570	4.37	43.25	4.47	70.66	24.99	109.09	29.78
2020-06-05	27.98	82.53	2.2000	287.69	49.15	30.69	68.37	16.06	522.33	31.74	78.76	40.36	7.345	5.24	43.95	4.14	72.41	26.83	107.43	31.74

In [53]:
```
snapshot = close2020.loc['2020-06-24':'2020-07-16']
```

In [54]:
```
snapshot_pctchange = snapshot.pct_change().dropna()
snapshot_pctchange.head()
```

Out[54]:

Unnamed: 0	HMC	BBY	BLNK	ANTM	KO	CTVA	CVS	LEVI	SAM	EPC	ETSY	EXC	F	
2020-06-25	-0.001521	0.010737	0.074764	0.014227	0.005822	0.004987	0.012814	-0.005947	0.005881	0.026135	0.032080	-0.029198	0.013445	-0.0
2020-06-26	-0.035048	0.000354	0.104436	-0.020160	-0.030053	-0.010687	-0.008903	-0.054180	0.056428	-0.020777	0.012141	-0.019486	-0.019900	-0.0
2020-06-29	0.009870	0.018761	0.326700	0.014199	0.018132	0.008488	0.015760	0.085106	-0.027851	0.050992	0.002048	0.036578	0.018613	0.0
2020-06-30	-0.001173	0.010424	0.420825	0.014309	0.006312	0.024484	0.007913	0.009804	-0.020273	0.014653	0.034054	0.008058	0.011628	0.0
2020-07-01	-0.007045	0.001032	-0.151567	0.021636	0.003808	0.008962	-0.006389	0.023152	0.034960	-0.030809	0.045751	0.029768	-0.019704	-0.0

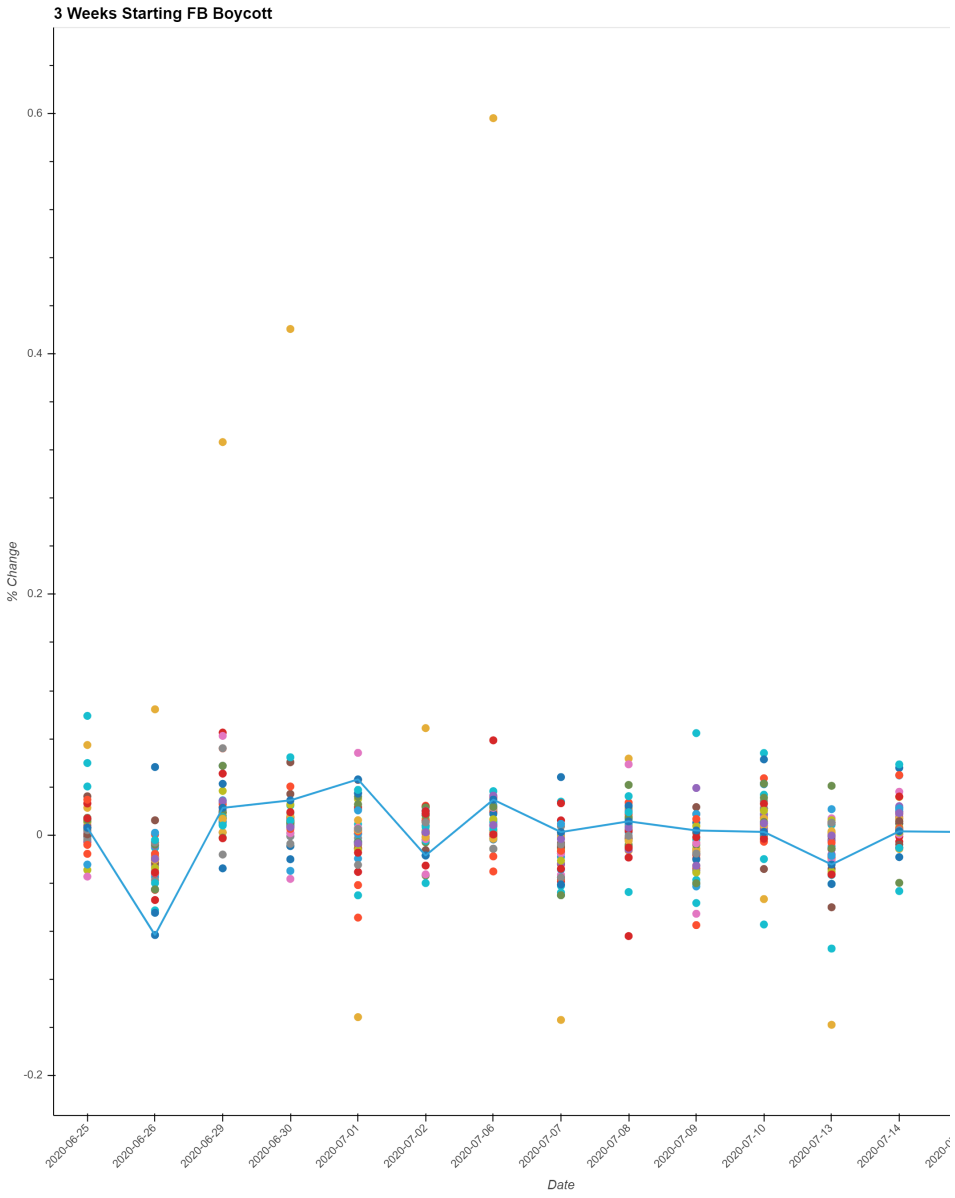
In [107]:
```
plot1 = snapshot_pctchange.hvplot.scatter(size=50, rot = 45, legend=True, title = '3 Weeks Starting FB Boycott', xlabel = 'Date', yl
```

In [108]:
```
snapshot_pctchange_fb = snapshot_pctchange.loc[:,'FB']
```

In [109]:
```
plot2 = snapshot_pctchange_fb.hvplot( hover=True, width = 1200, height=1200)
```

In [110]:
```
plot3= plot1 * plot2
plot3
```

Out[110]:



We can start to see <u>trends**</u> when we break down our FB closing prices compared to each sector**

in the list of participating boycott companies

CLOSING PRICES - <<<*PERIOD* = *SHORT TERM*>>>

<u>*** BY SECTOR***</u>

<u>CONSUMER CYCLICAL SECTOR</u>

- Higher Growth Consumer Sector
- Higher Volatility
- Associated with convenience items, cars, clothing, fashion, travel & entertainment

In [59]:
```
consumer_cyclical_2020_period_immediate=close2020.loc['2020-06-24':'2020-07-16',('CZR', 'SIG', 'SIX', 'LEVI', 'HMC', 'GME', 'FOSL',
```

```
C:\Users\allen\.conda\envs\pyvizenv\lib\site-packages\pandas\core\indexing.py:868: FutureWarning:
Passing list-likes to .loc or [] with any missing label will raise
KeyError in the future, you can use .reindex() as an alternative.

See the documentation here:
https://pandas.pydata.org/pandas-docs/stable/indexing.html#deprecate-loc-reindex-listlike
  return self._getitem_lowerdim(tup)
```

In [60]:
```
snapshot2 = consumer_cyclical_2020_period_immediate
```

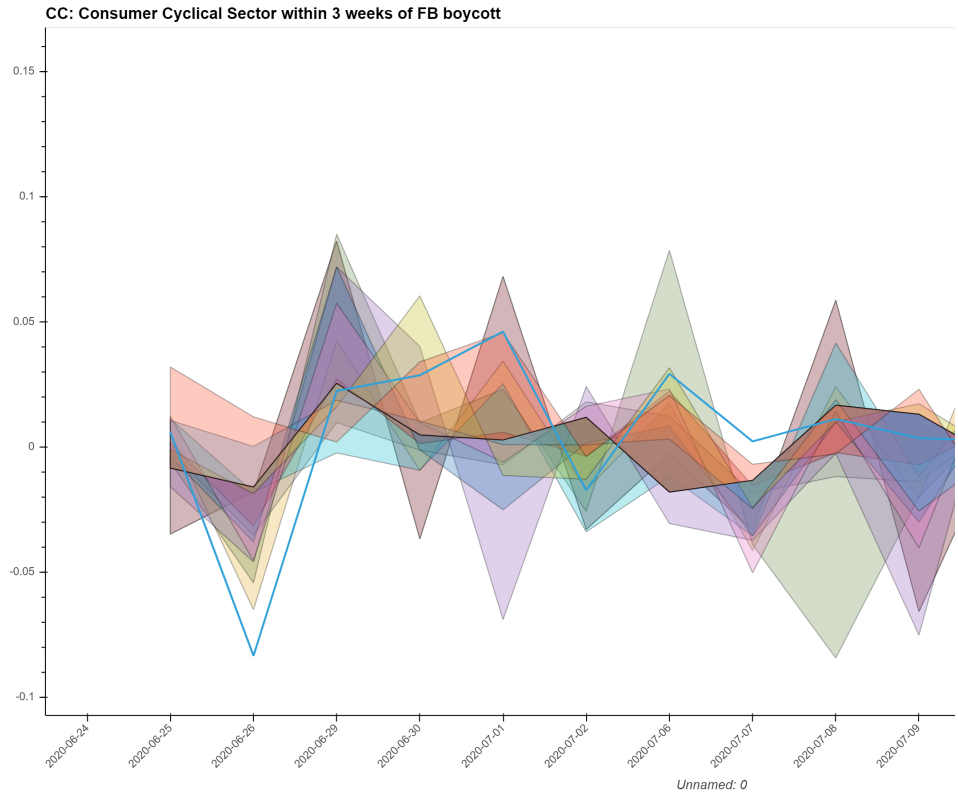
In [61]:
```
snapshot2_pctchg = snapshot2.pct_change()
```

In [62]:
```
snap2graph = snapshot2_pctchg.hvplot(kind='area',title='CC: Consumer Cyclical Sector within 3 weeks of FB boycott',hover_cols='all',
```

```
WARNING:param.main: sort option not found for area plot; similar options include: ['sort_date']
```

In [63]:
```
# CONSUMER CYCLICAL - plotted again FB's Trend over the same time period

snap2graph.opts(height=800, width = 1500) * plot2.opts(height=800, width=1500)
```

Out[63]:



CONSUMER DEFENSIVE SECTOR

- Steady Growth Consumer Sector
- Lower Volatility
- Associated with groceries, food, alcohol, and bulk commodities

In [64]:
```
consumer_defensive_2020_period_immediate=close2020.loc['2020-06-24':'2020-07-16',('TGT', 'PEP', 'UL','CL','SAM','SJM','TAP','EPC','
```

In [65]:
```
snapshot3 = consumer_defensive_2020_period_immediate
```

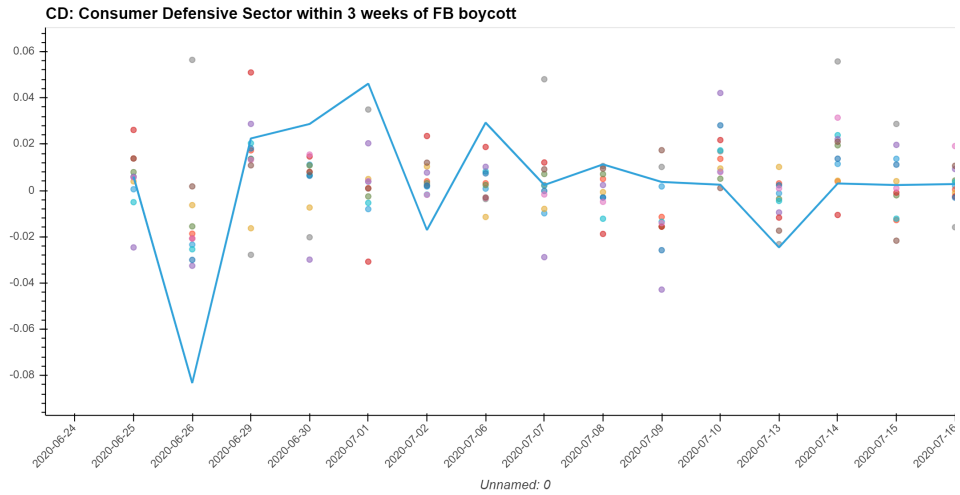
In [66]:
```
snapshot3_pctchg = snapshot3.pct_change()
```

In [67]:
```
snap3graph = snapshot3_pctchg.hvplot(kind='scatter',title='CD: Consumer Defensive Sector within 3 weeks of FB boycott',hover_cols='a
```

```
WARNING:param.main: sort option not found for scatter plot; similar options include: ['sort_date']
```

In [68]:
```
snap3graph * plot2
```

Out[68]:

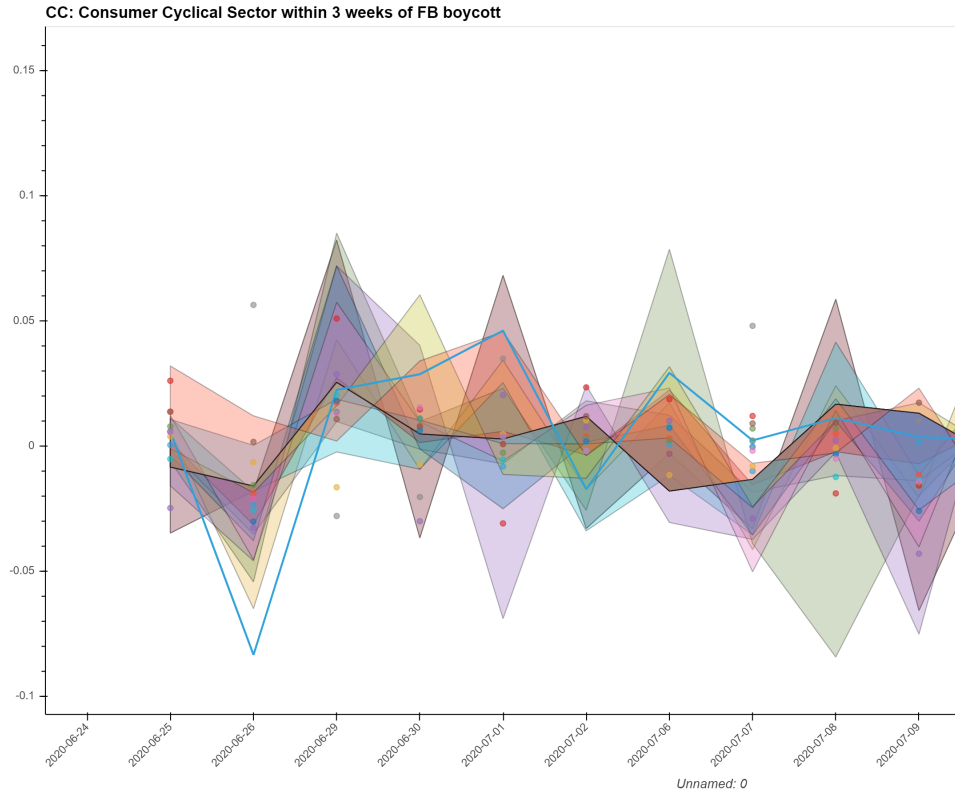


The plot CC and plot CD look very similar, the scales (y-axis) are different in amplitude

to better illustrate we can superimpose boh plots to witness scale, and to track their movements together with FB

In [69]:
```
snap2graph * plot2 *snap3graph
```

Out[69]:



- SEEN ABOVE Consumer Defensve (Scatter Plot) has the highest delta within 5-6% from center whereas, Consumer Cyclical (Area Plot) can experience wild swings, even within such a short 3 week window, deltas in 15-16% range.

In [ ]:
```
```

CLOSING PRICES - <<<*PERIOD = Q3 & Q4 2020 (mid-term)*>>>

We can now expand the focus (still reviewing CLOSE PRICE)

But include a different period, 2020 Q3 & Q4 as compared with 2019 Q3 & Q4

In [70]:
```
close2020.rename_axis('date', inplace=True)
close2020.head()
```

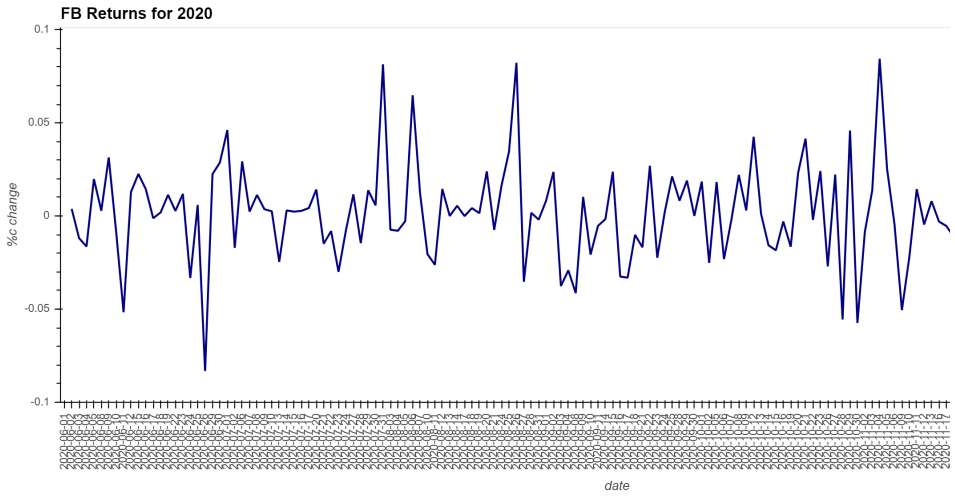
Out[70]:

date	HMC	BBY	BLNK	ANTM	KO	CTVA	CVS	LEVI	SAM	EPC	ETSY	EXC	F	FOSL	FMS	GME	CL	HP	SJM	EPC.1	VFC
2020-06-01	26.37	77.50	1.7800	290.08	46.98	26.80	65.64	13.81	569.26	30.03	78.59	38.84	5.870	3.12	42.60	4.14	72.73	20.27	115.35	30.03	58.0
2020-06-02	26.48	80.55	1.7800	294.25	46.89	28.01	66.86	14.28	568.34	29.88	81.84	38.94	5.900	3.44	43.35	4.18	72.43	21.79	115.18	29.88	59.3
2020-06-03	27.00	82.23	1.8899	289.10	47.90	28.48	66.87	15.34	555.93	30.13	80.61	39.91	6.190	3.81	43.32	4.45	71.65	22.69	114.39	30.13	64.0
2020-06-04	26.93	79.92	2.1674	282.09	47.92	30.18	66.52	15.44	529.92	29.78	80.25	38.98	6.570	4.37	43.25	4.47	70.66	24.99	109.09	29.78	64.0
2020-06-05	27.98	82.53	2.2000	287.69	49.15	30.69	68.37	16.06	522.33	31.74	78.76	40.36	7.345	5.24	43.95	4.14	72.41	26.83	107.43	31.74	68.2

In [71]:
```
# First 2020- Q3 and Q4 - baseline using FB

fb_2020_pct = close2020.loc[:,('FB')].pct_change()
```

In [72]:
```
fb_2020_plot = fb_2020_pct.hvplot(height=500, width =1200, rot=90, title = 'FB Returns for 2020', xlabel='date', ylabel='%c change',
fb_2020_plot
```

Out[72]:

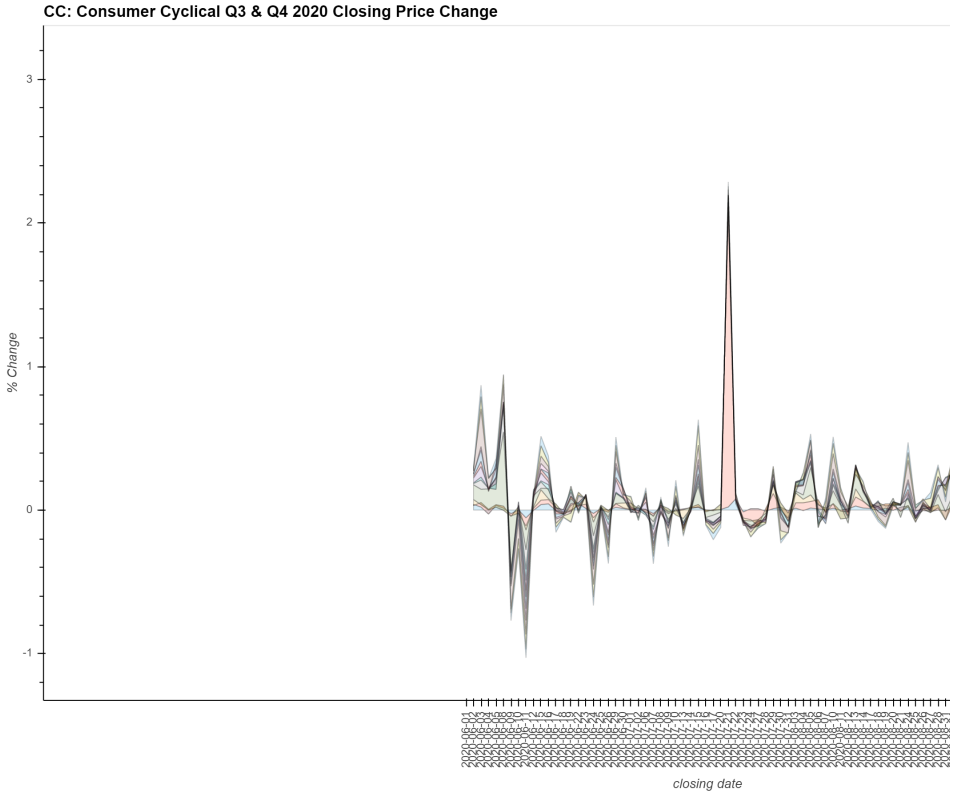


In [73]:
```
# Then, 2020- Q3 and Q4  for CONSUMER CYCLICAL SECTOR

q3q4_2020=close2020.loc[:,('CZR', 'SIG', 'SIX', 'LEVI', 'HMC', 'GME', 'FOSL', 'SBUX', 'RL', 'VFC', 'WSM','LULU','BBY','ETSY')]
```

```
C:\Users\allen\.conda\envs\pyvizenv\lib\site-packages\pandas\core\indexing.py:868: FutureWarning:
Passing list-likes to .loc or [] with any missing label will raise
KeyError in the future, you can use .reindex() as an alternative.

See the documentation here:
https://pandas.pydata.org/pandas-docs/stable/indexing.html#deprecate-loc-reindex-listlike
  return self._getitem_lowerdim(tup)
```

In [74]:
```
q3q4_2020_pctchange = q3q4_2020.pct_change()
plot2020_CC = q3q4_2020_pctchange.hvplot.area(rot =90, width = 1500, height = 800, alpha=.2,title = 'CC: Consumer Cyclical Q3 & Q4 2
plot2020_CC
```

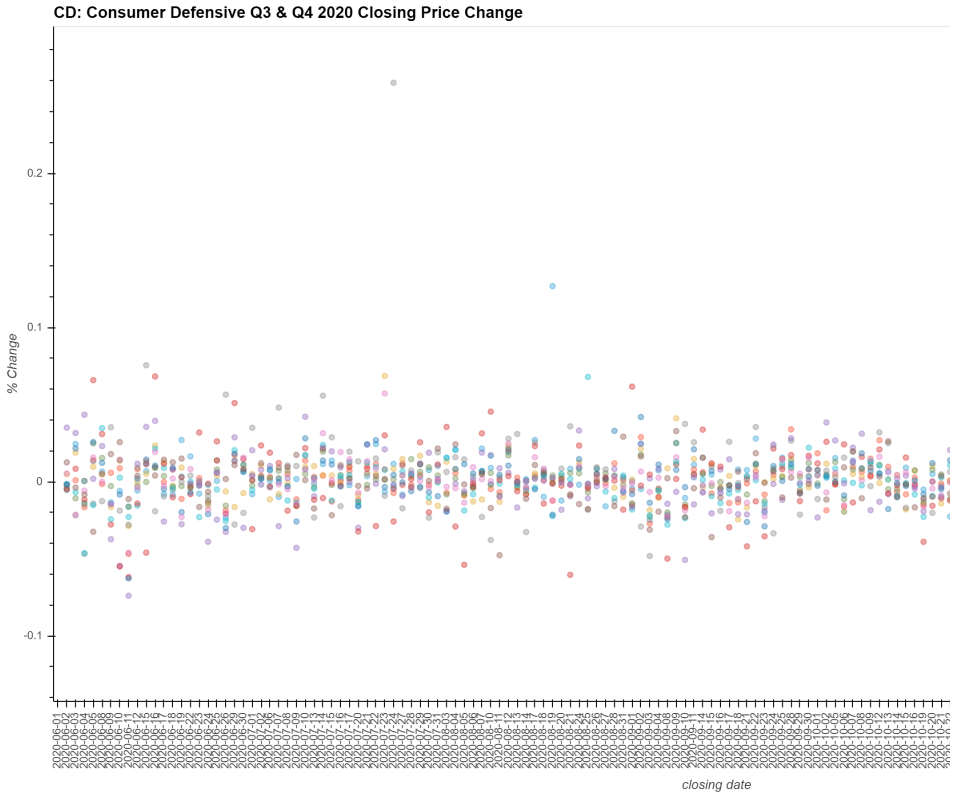
Out[74]:



In [75]:
```
# Then, 2020- Q3 and Q4  for CONSUMER DEFENSIVE SECTOR

q3q4_2020_CD=close2020.loc[:,('TGT', 'PEP', 'UL','CL','SAM','SJM','TAP','EPC','KO','HSY','CLX')]
q3q4_2020_pctchange_CD = q3q4_2020_CD.pct_change()
```

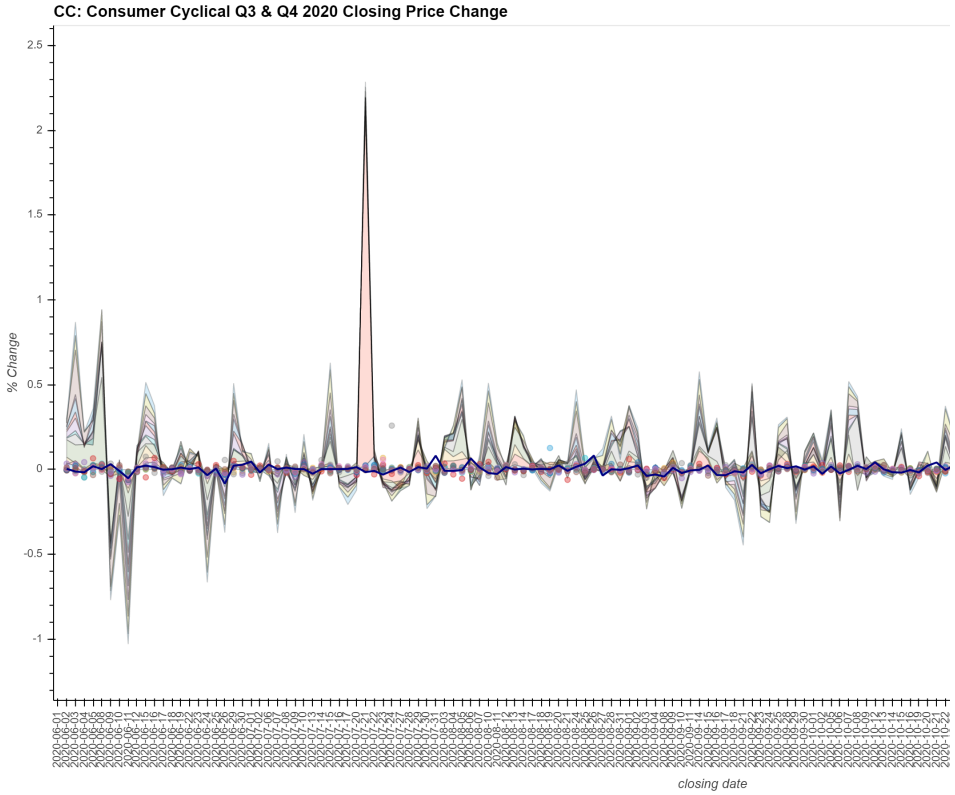
In [76]:
```
plot2020_CD = q3q4_2020_pctchange_CD.hvplot(kind='scatter',rot =90, width = 1500, height = 800, alpha=.4,title = 'CD: Consumer Defer
plot2020_CD
```

Out[76]:



In [77]:
```
# Now we can plot all 3 together to see Full 2020 Q3 and Q4 numbers

Plot2020 = plot2020_CC  * plot2020_CD  * fb_2020_plot
Plot2020
```

Out[77]:



The general trend continues

Consumer Defensive has smaller swings in daily change; most prominent in the 0-2.5% delta range Consumer Cyclical experienced more severe price swings between +/- 5 % in the same period.

CLOSING PRICES - <<<*PERIOD = Q3 and Q4 2019 (mid-term)* >>>

In [78]:
```
# Now to compare with 2019 numbers from the same period
# Direct Path, and run import fn

path = Path('closingprice2019.csv')
close2019= pd.read_csv(path, delimiter=',')
close2019.rename(columns={'Unnamed: 0':'date'},inplace=True)
close2019.set_index('date', inplace=True)
close2019.head()
```

Out[78]:

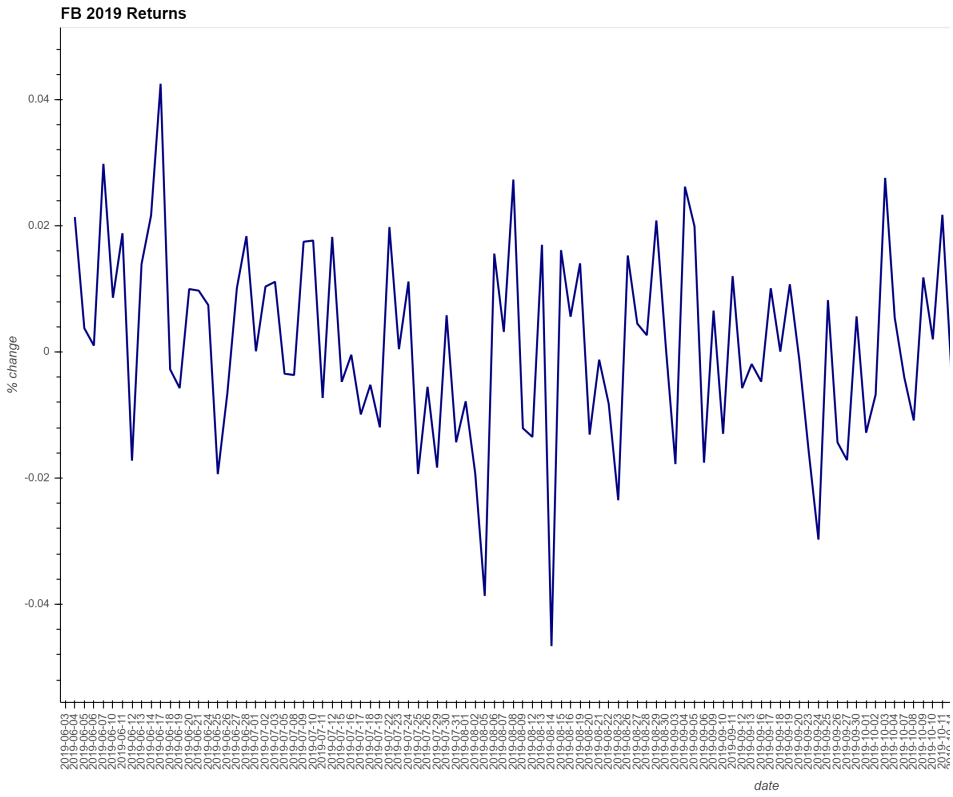
date	HMC	BBY	BLNK	ANTM	KO	CTVA	CVS	LEVI	SAM	EPC	ETSY	EXC	F	FOSL	FMS	GME	CL	HP	SJM	EPC.1
2019-06-03	24.82	NaN	2.07	NaN	49.9500	NaN	53.39	20.635	NaN	28.81	61.07	NaN	9.610	10.25	36.84	7.4600	70.66	49.78	124.30	28.81
2019-06-04	25.48	63.94	2.32	279.81	50.0000	27.48	54.63	21.480	305.77	29.90	61.20	48.71	9.920	11.51	37.69	7.8265	70.84	50.33	124.60	29.90
2019-06-05	25.63	63.24	2.29	276.96	50.8000	27.97	53.18	21.350	309.90	29.54	61.49	49.72	9.770	11.18	37.45	5.0601	71.91	50.99	125.25	29.54
2019-06-06	25.45	62.31	2.32	277.50	51.4100	26.89	54.14	20.820	319.97	30.16	61.97	50.24	9.750	10.70	37.51	5.1100	73.11	51.20	122.43	30.16
2019-06-07	25.58	63.72	2.37	278.12	51.4753	26.31	53.92	20.975	320.36	30.73	63.89	49.96	9.755	11.10	37.61	5.0100	73.99	51.87	124.51	30.73

In [ ]:

In [79]:
```
# First we examine FB's perofrmance Q3 and Q4 2019
fb_2019_pct = close2019.loc[:,('FB')].pct_change()
```

In [80]:
```
fb_2019_pct_plot = fb_2019_pct.hvplot(rot =90, width = 1500, height = 800, title='FB 2019 Returns', xlabel='date', ylabel='% change'
fb_2019_pct_plot
```

Out[80]:



In [81]:
```
# First we examine Q3 Q4 2019 - within the CONSUMER CYCLICAL SECTOR

close2019.loc[:,('CZR', 'SIG', 'SIX', 'LEVI', 'HMC', 'GME', 'FOSL', 'SBUX', 'RL', 'VFC', 'WSM','LULU','BBY','ETSY')].head()
```

```
C:\Users\allen\.conda\envs\pyvizenv\lib\site-packages\pandas\core\indexing.py:868: FutureWarning:
Passing list-likes to .loc or [] with any missing label will raise
KeyError in the future, you can use .reindex() as an alternative.

See the documentation here:
https://pandas.pydata.org/pandas-docs/stable/indexing.html#deprecate-loc-reindex-listlike
  return self._getitem_lowerdim(tup)
```

Out[81]:

date	CZR	SIG	SIX	LEVI	HMC	GME	FOSL	SBUX	RL	VFC	WSM	LULU	BBY	ETSY
2019-06-03	8.61	19.09	NaN	20.635	24.82	7.4600	10.25	76.21	107.16	NaN	NaN	163.980	NaN	61.07
2019-06-04	9.00	20.55	51.18	21.480	25.48	7.8265	11.51	78.52	111.81	85.11	NaN	168.900	63.94	61.20
2019-06-05	9.02	19.44	51.25	21.350	25.63	5.0601	11.18	79.96	111.01	85.77	NaN	173.570	63.24	61.49
2019-06-06	9.13	19.16	51.68	20.820	25.45	5.1100	10.70	81.41	109.47	84.45	NaN	170.440	62.31	61.97
2019-06-07	9.45	19.72	51.84	20.975	25.58	5.0100	11.10	82.47	110.89	84.70	NaN	172.421	63.72	63.89

In [82]:
```
q3q4_2019_CC= close2019.loc[:,('CZR', 'SIG', 'SIX', 'LEVI', 'HMC', 'GME', 'FOSL', 'SBUX', 'RL', 'VFC', 'WSM','LULU','BBY','ETSY')].l
```

In [83]:
```
q3q4_2019_CC= close2019.loc[:,('CZR', 'SIG', 'SIX', 'LEVI', 'HMC', 'GME', 'FOSL', 'SBUX', 'RL', 'VFC', 'WSM','LULU','BBY','ETSY')].a
```

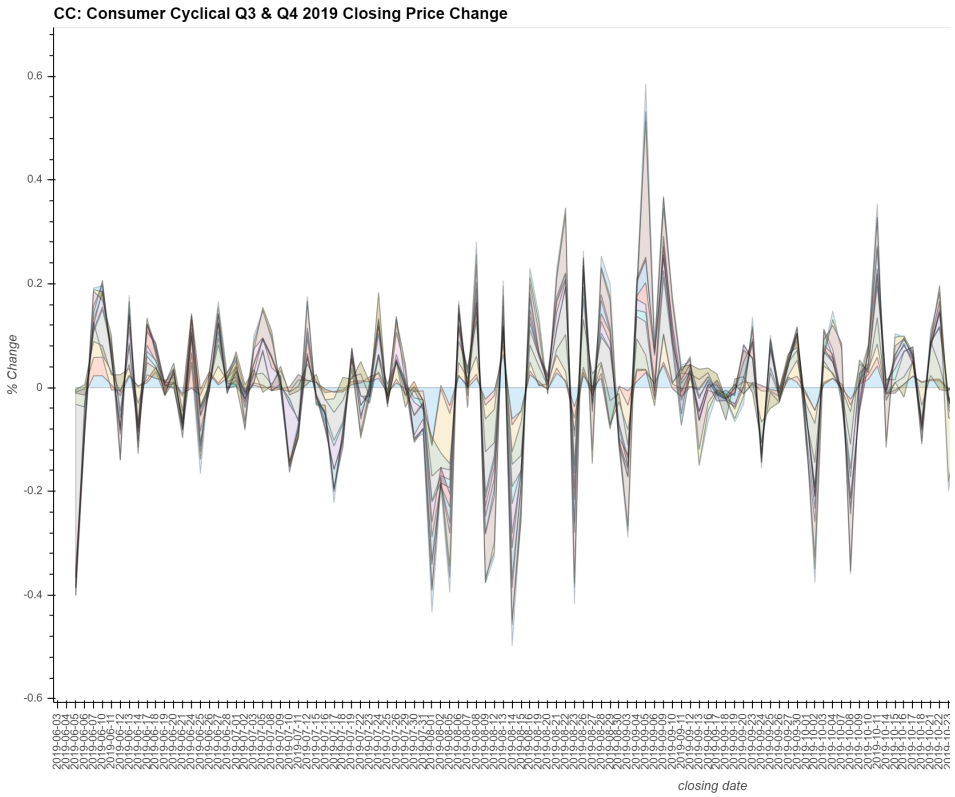
In [84]:
```
q3q4_2019_CC_pct = q3q4_2019_CC.pct_change()
q3q4_2019_CC_pct.head()
```

Out[84]:

date	CZR	SIG	SIX	LEVI	HMC	GME	FOSL	SBUX	RL	VFC	WSM	LULU	BBY	ETSY
2019-06-03	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2019-06-04	0.045296	0.076480	NaN	0.040950	0.026591	0.049129	0.122927	0.030311	0.043393	NaN	NaN	0.030004	NaN	0.002129
2019-06-05	0.002222	-0.054015	0.001368	-0.006052	0.005887	-0.353466	-0.028671	0.018339	-0.007155	0.007755	NaN	0.027649	-0.010948	0.004739
2019-06-06	0.012195	-0.014403	0.008390	-0.024824	-0.007023	0.009861	-0.042934	0.018134	-0.013873	-0.015390	NaN	-0.018033	-0.014706	0.007806
2019-06-07	0.035049	0.029228	0.003096	0.007445	0.005108	-0.019569	0.037383	0.013021	0.012972	0.002960	NaN	0.011623	0.022629	0.030983

In [85]:
```
q3q4_2019_CC_plot= q3q4_2019_CC_pct.hvplot.area(rot =90, width = 1500, height = 800, alpha=.2,title = 'CC: Consumer Cyclical Q3 & Q4
q3q4_2019_CC_plot
```

Out[85]:

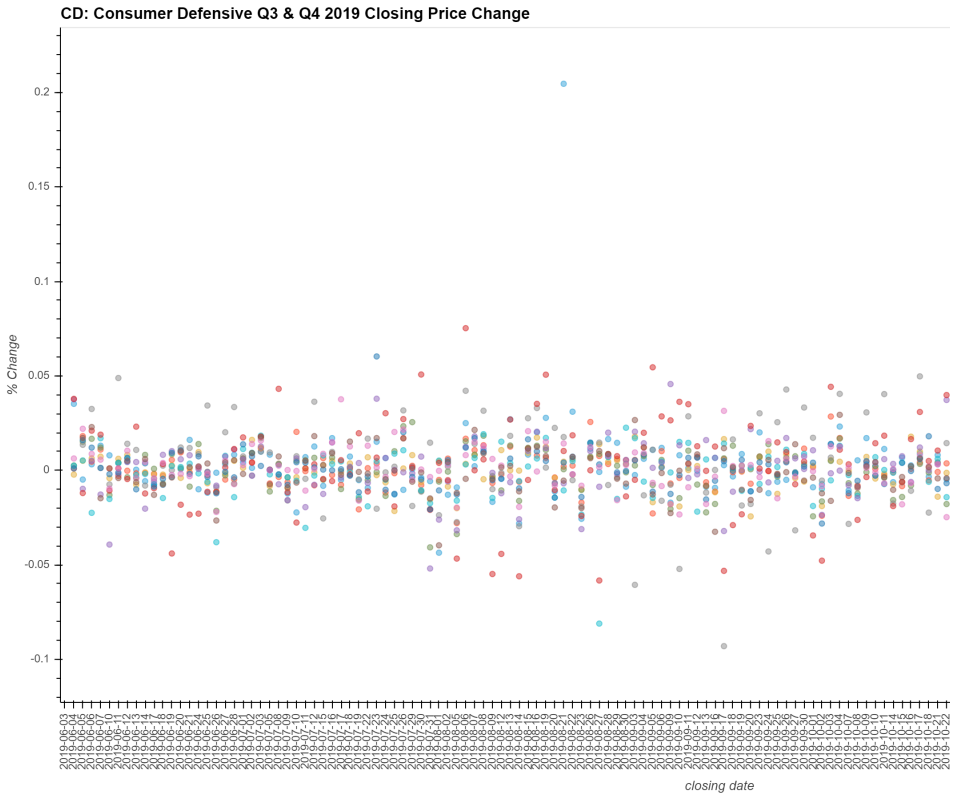


In [86]:
```
# Then we can see the same for CONSUMER DEFENSIVE SECTOR

q3q4_2019_CD = close2019.loc[:,('TGT', 'PEP', 'UL','CL','SAM','SJM','TAP','EPC','KO','HSY','CLX')]
```

In [87]:
```
q3q4_2019_CD_pct = q3q4_2019_CD.pct_change()
```

In [88]:
```
q3q4_2019_CD_plot = q3q4_2019_CD_pct.hvplot(kind='scatter',rot =90, width = 1500, height = 800, alpha=.5,title = 'CD: Consumer Defer
q3q4_2019_CD_plot
```

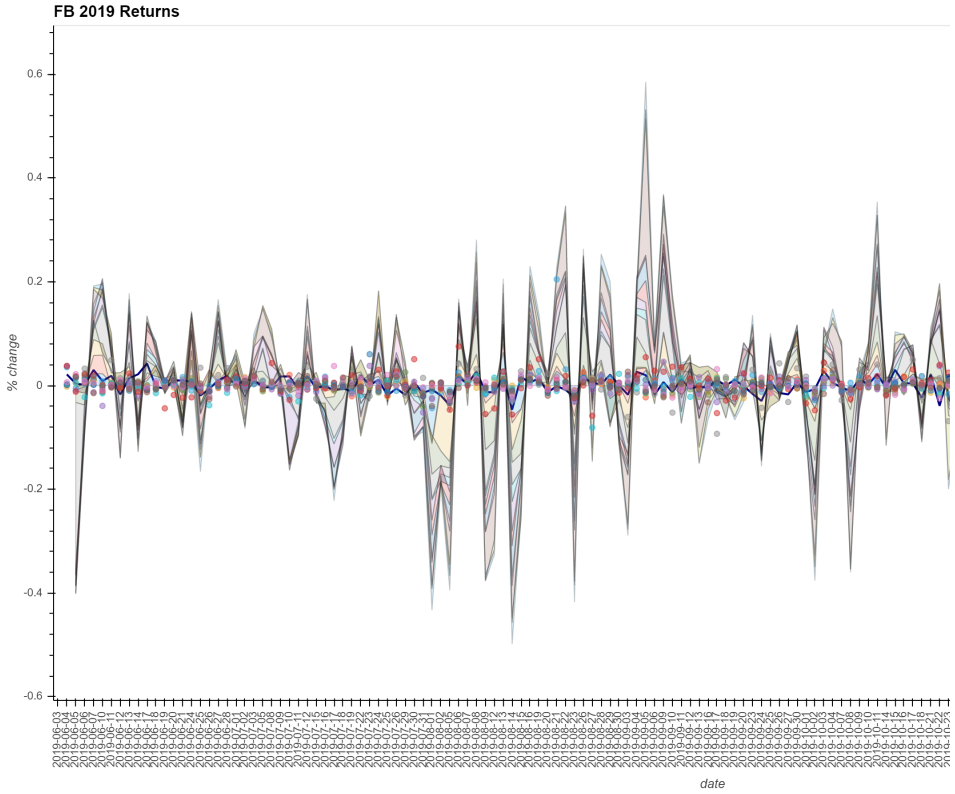
Out[88]:



In [89]:
```
all_plots_2019 = fb_2019_pct_plot*q3q4_2019_CC_plot*q3q4_2019_CD_plot
```

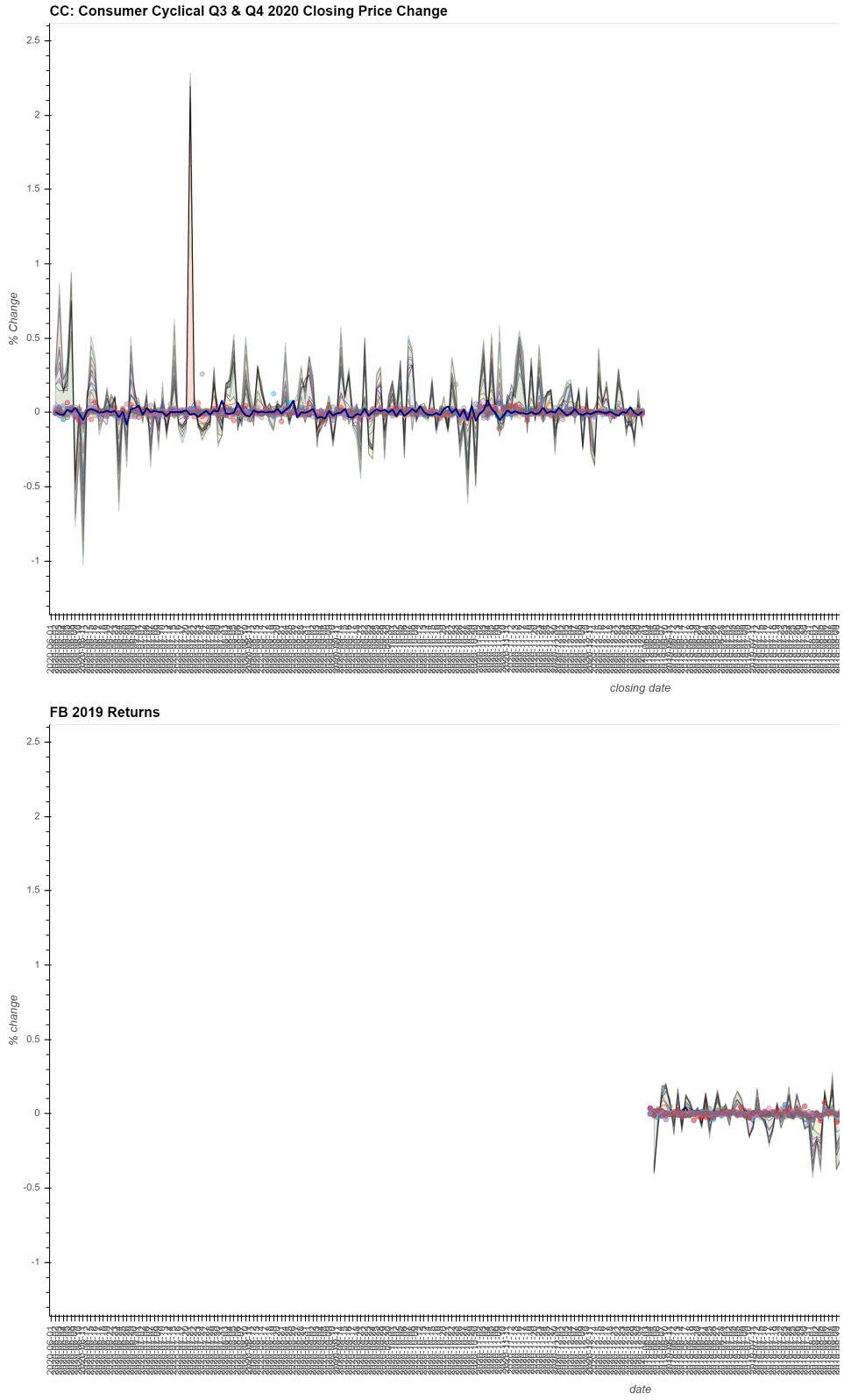
In [90]:
```
all_plots_2019
```

Out[90]:



In [91]:
```
two_year_plot = hv.Layout(Plot2020+all_plots_2019).cols(1)
two_year_plot
```

Out[91]:



PRICE TO SALES RATIO

<<<PERIOD = Q3 and Q4 2020 (mid-term RATIO) >>>

In [92]:
```
# Now comparing Price to sales between FB and both sectors -
#   CONSUMER CYCLICAL &
#   CONSUMER DEFENSIVE
```

In [93]:
```
# CONSUMER CYCLICAL

path = Path('ptosales_consumer_cyclical.csv')
p2s_consumer_cyclical= pd.read_csv(path, delimiter=',')
p2s_consumer_cyclical.rename(columns={'Unnamed: 0': 'Company'},inplace=True)
```

In [94]:
```
# CONSUMER DEFENSIVE

path = Path('ptosales_consumer_defensive.csv')
p2s_consumer_defensive = pd.read_csv(path, delimiter = ',')
p2s_consumer_defensive.rename(columns={'Unnamed: 0': 'Company'},inplace=True)
p2s_consumer_defensive
```

Out[94]:

	Company	revenue	Gross_Profit_ratio	price_to_sales	Market_Capit
0	CL	16471000000	0.608160	4.150505	6.836298e+10
1	CLX	6721000000	0.455736	3.677042	2.471340e+10
2	EPC	1949700000	0.451813	1.107973	2.160216e+09
3	HSY	8149719000	0.454159	5.304401	4.322938e+10
4	KO	33014000000	0.593112	6.907499	2.280442e+11
5	PEP	70372000000	0.548158	2.815475	1.981306e+11
6	SAM	1851813000	0.439813	8.408127	1.557028e+10
7	SJM	7801000000	0.384822	1.875714	1.463245e+10
8	TAP	10579400000	0.397111	1.057986	1.119286e+10
9	TGT	93561000000	0.292686	1.100360	1.029508e+11
10	UL	50724000000	0.434508	0.000000	0.000000e+00

In [95]:
```
# Finally, price-to-sales-ratio for FaceBook itself

path = Path('p2s_fb.csv')
p2s_fb = pd.read_csv(path, delimiter=',')
p2s_fb.rename(columns={'Unnamed: 0':'Company'},inplace=True)
```

In [96]:
```
fb_p2s = p2s_fb.hvplot(kind='bar', x='Company', y='price_to_sales')
```

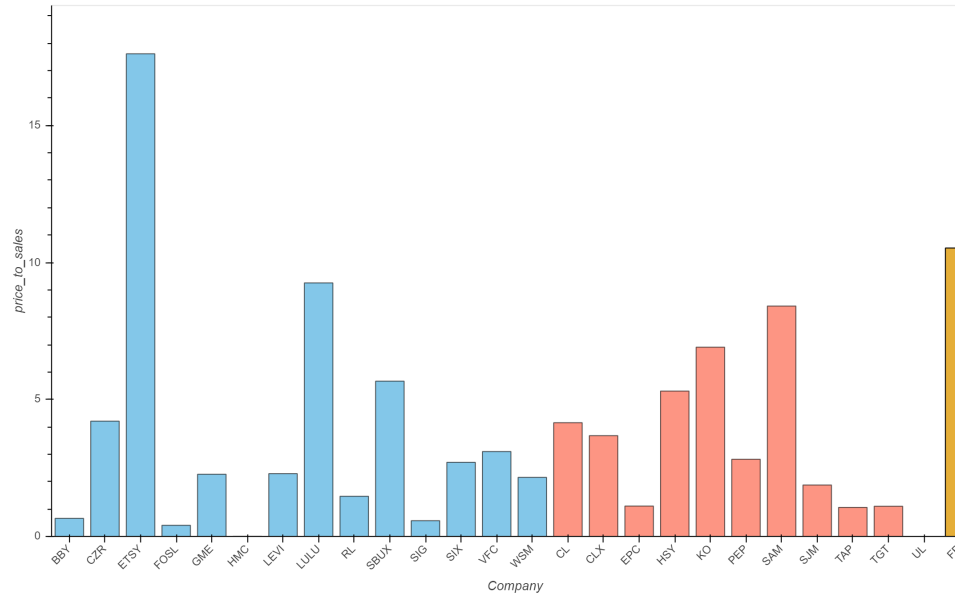
In [97]:
```
p2scyclical =p2s_consumer_cyclical.hvplot(kind='bar', x='Company',y='price_to_sales', height=600, width= 1000, alpha=.6, rot=45)
```

In [98]:
```
p2sdefensive = p2s_consumer_defensive.hvplot(kind='bar', x='Company',y='price_to_sales', height=600, width= 1000, alpha=.6, rot=45)
```

In [99]:
```
p3 = p2scyclical * p2sdefensive * fb_p2s
```

In [100]:
```
p3
```

Out[100]:



In [ ]:
```
```