In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
%config InlineBackend.figure_format = 'svg'
```

Newfoundland PUB Fuel Price

The zones are:

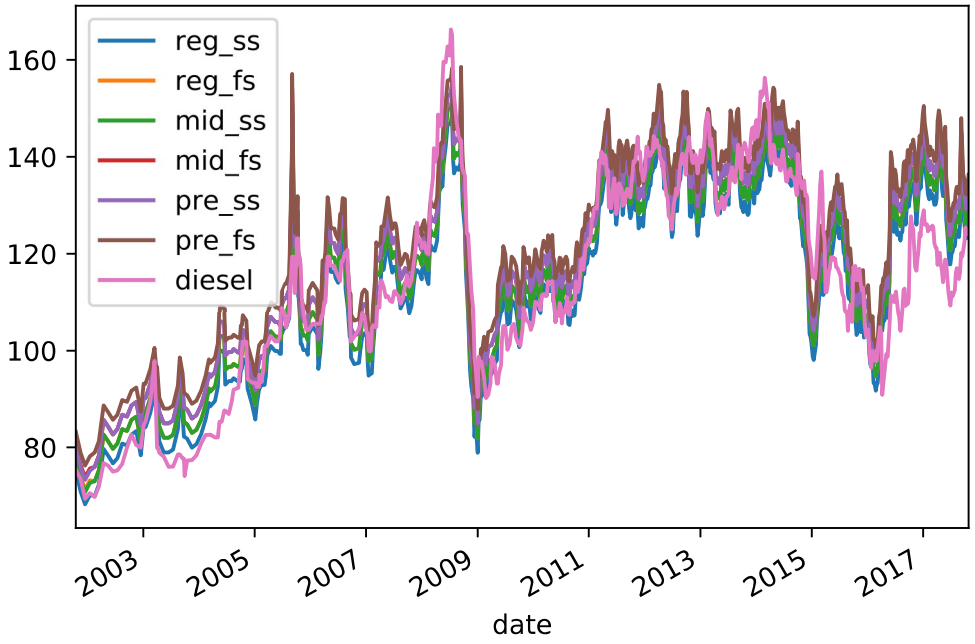
- 1, Avalon Peninsula
- 1a, Bell Island
- 2, Burin Peninsula/Bonavista Peninsula
- 3, Central Newfoundland/Notre Dame Bay East
- 3a, St. Brendan's (Island)
- 3b, Fogo Island
- 3c, Change Islands
- 4, Connaigre Peninsula
- 4a, Gaultois/McCallum/Rencontre East
- 5, Springdale - Green Bay/Triton/Baie Verte Peninsula
- 5a, Long Island
- 5b, Little Bay Islands
- 6, Deer Lake/Corner Brook/Bay of Islands/Gros Morne
- 7, Stephenville/Port au Port/Codroy Valley/Channel-Port aux Basques /Burgeo
- 7a, Ramea
- 7b, Grey River/François/Grand Bruit/La Poile
- 8, Northern Peninsula - Gros Morne National Park to Bellburns
- 9, Northern Peninsula to Englee and St. Anthony
- 10, Labrador - The Straits to Red Bay
- 11, Labrador South - Lodge Bay / Cartwright
- 11a, Coastal Labrador South - Tanker Supplied
- 11b, Coastal Labrador South - Drum Delivery
- 12, Central Labrador
- 13, Western Labrador
- 13a, Churchill Falls
- 14, Coastal Labrador North

In [2]:
```
fuel = pd.read_csv('fuel.csv', index_col='date',  parse_dates=True)
```

In [3]:
```
import re
only_digits = re.compile(r'^\d+')
def start_digits( k ) :
    m = only_digits.search( k )
    if m :
        return int(m.group())
    else:
        return 0

zone_names = list(fuel.zone.unique())
zone_names.sort(key=start_digits )
# get a view for each zone
zones = { zn : fuel[ fuel.zone == zn ].copy() for zn in zone_names }
# drop all zone columns
for z in zones.values() :
    z.drop('zone',axis=1, inplace=True)
```

In [4]:
```
zones['1'].plot(); None
```



Most of the fuel price seem to be strongly related, the only exception appears to be diesel.

In [5]:
```
# determine the mean difference between all gas products for zone 1
import itertools as it
z = zones['1']
cols = list(z.columns)
cols.remove('diesel')
for p1,p2 in it.combinations(cols,2) : # compare all pairs of products
    diff = (z[p1] - z[p2]).describe()
    print(p1, p2, diff['mean'], diff['std'] )
```

```
reg_ss reg_fs -2.9998427673 0.00396525792859
reg_ss mid_ss -2.99669811321 0.0397445789447
reg_ss mid_fs -5.99544025157 0.0411790163206
reg_ss pre_ss -5.99229559748 0.0798842511819
reg_ss pre_fs -8.99213836478 0.0799674325826
reg_fs mid_ss 0.00314465408805 0.0395588008828
reg_fs mid_fs -2.99559748428 0.04100456967
reg_fs pre_ss -2.99245283019 0.0798006727904
reg_fs pre_fs -5.99229559748 0.0798842511819
mid_ss mid_fs -2.99874213836 0.0111534543706
mid_ss pre_ss -2.99559748428 0.04100456967
mid_ss pre_fs -5.99544025157 0.0411790163206
mid_fs pre_ss 0.00314465408805 0.0395588008828
mid_fs pre_fs -2.99669811321 0.0397445789447
pre_ss pre_fs -2.9998427673 0.00396525792859
```

Since the standard deviation seems so low, an initial guess is that there is mostly a constant price difference between all the gas grades and service levels.

	reg_ss	reg_fs	mid_ss	mid_fs	pre_ss	pre_fs
reg_ss	0	-3	-3	-6	-6	-9
reg_fs	3	0	0	-3	-3	-6
mid_ss	3	0	0	-3	-3	-6
mid_fs	6	3	3	0	0	-3
pre_ss	6	3	3	0	0	-3
pre_fs	9	6	6	3	3	0

In summary, pre is 3 cents more than med, med is 3 cents more than reg, and full service adds 3 cents.

An examination of where the differences occur is:

In [6]:
```
z = zones['1']
z[((z.reg_ss+3.0) - z.reg_fs).abs() > 0.001]
```

Out[6]:

	reg_ss	reg_fs	mid_ss	mid_fs	pre_ss	pre_fs	diesel
date							
2006-08-19	112.7	115.6	115.6	118.6	118.6	121.6	120.2

Thus there is only one entry where `reg_ss` differs from `reg_fs` by a different amount than 3 cents.

In [7]:
```
# check that reg_fs cost 3 cents more than reg_ss
cond = ((z.reg_ss+3.0) - z.reg_fs).abs() > 0.001
z.reg_ss[cond] - z.reg_fs[cond]
```

Out[7]:
```
date
2006-08-19   -2.9
dtype: float64
```

The differents dates for reg_ss and mid_ss.

In [8]:
```
z = zones['1']
z[((z.reg_ss+3.0) - z.mid_ss).abs() > 0.001]
```

Out[8]:

	reg_ss	reg_fs	mid_ss	mid_fs	pre_ss	pre_fs	diesel
date							
2001-10-15	75.3	78.3	77.8	80.8	80.3	83.3	75.5
2001-11-15	70.9	73.9	73.4	76.4	75.9	78.9	73.5
2001-12-15	68.3	71.3	70.8	73.8	73.3	76.3	69.4
2002-01-15	70.2	73.2	72.7	75.7	75.2	78.2	70.6
2006-08-19	112.7	115.6	115.6	118.6	118.6	121.6	120.2

A python routine can be written to see how may entries differ by a set amount.

In [9]:
```
def differs_by( ser1, ser2, amount, espilon = 0.001 ) :
    return ((ser1+amount) - ser2).abs() > espilon
```

In [10]:
```
z = zones['1']
cond = differs_by(z.reg_ss,z.mid_ss,3.0)
z.reg_ss[cond] - z.mid_ss[cond]
```

Out[10]:
```
date
2001-10-15   -2.5
2001-11-15   -2.5
2001-12-15   -2.5
2002-01-15   -2.5
2006-08-19   -2.9
dtype: float64
```

Check where `reg_ss` differs from `mid_fs` by 6 cents.

In [11]:
```
z = zones['1']
z[ differs_by(z.reg_ss, z.mid_fs, 6.0)]
```

Out[11]:

	reg_ss	reg_fs	mid_ss	mid_fs	pre_ss	pre_fs	diesel
date							
2001-10-15	75.3	78.3	77.8	80.8	80.3	83.3	75.5
2001-11-15	70.9	73.9	73.4	76.4	75.9	78.9	73.5
2001-12-15	68.3	71.3	70.8	73.8	73.3	76.3	69.4
2002-01-15	70.2	73.2	72.7	75.7	75.2	78.2	70.6
2006-07-01	114.0	117.0	117.0	119.9	119.9	122.9	115.4
2006-07-03	114.2	117.2	117.2	120.1	120.1	123.1	115.4
2006-07-15	119.3	122.3	122.3	125.2	125.2	128.2	114.4
2006-07-22	119.3	122.3	122.3	125.2	125.2	128.2	118.5
2006-08-19	112.7	115.6	115.6	118.6	118.6	121.6	120.2
2006-09-01	108.7	111.7	111.7	114.6	114.6	117.6	120.8
2006-09-02	108.7	111.7	111.7	114.6	114.6	117.6	117.4
2006-09-15	102.8	105.8	105.8	108.7	108.7	111.7	110.0
2006-09-22	102.8	105.8	105.8	108.7	108.7	111.7	101.9

The amount of the difference is:

In [12]:
```
z = zones['1']
cond = differs_by(z.reg_ss, z.mid_fs, 6.0)
z.reg_ss[cond] - z.mid_fs[cond]
```

Out[12]:
```
date
2001-10-15   -5.5
2001-11-15   -5.5
2001-12-15   -5.5
2002-01-15   -5.5
2006-07-01   -5.9
2006-07-03   -5.9
2006-07-15   -5.9
2006-07-22   -5.9
2006-08-19   -5.9
2006-09-01   -5.9
2006-09-02   -5.9
2006-09-15   -5.9
2006-09-22   -5.9
dtype: float64
```

Zone 1 seems to follow the price difference observation, what about the other zones.

A table of all the prices differences can be created and printed with:

In [13]:
```
def print_mean_table( zones, zone ):
    z = zones[zone]
    cols = list(z.columns)
    cols.remove('diesel')
    n = len(cols)
    tab = np.zeros((n,n), dtype=np.float )
    for i in range( 0, n ):
        for j in range(i,n):
            tab[i,j] = (z[cols[i]] - z[cols[j]]).mean()
            tab[j,i] = -tab[i,j]

    for i in range(tab.shape[0]) :
        for j in range(tab.shape[1]) :
            print("%5.1f" % tab[i,j], end='')
        print()
    return tab

for z in zone_names:
    print('Zone:', z)
    print_mean_table( zones, z )
```

```
Zone: 1
 -0.0 -3.0 -3.0 -6.0 -6.0 -9.0
  3.0 -0.0  0.0 -3.0 -3.0 -6.0
  3.0 -0.0 -0.0 -3.0 -3.0 -6.0
  6.0  3.0  3.0 -0.0  0.0 -3.0
  6.0  3.0  3.0 -0.0 -0.0 -3.0
  9.0  6.0  6.0  3.0  3.0 -0.0
Zone: 1a
 -0.0 -3.0 -3.0 -6.0 -6.0 -9.0
  3.0 -0.0  0.0 -3.0 -3.0 -6.0
  3.0 -0.0 -0.0 -3.0 -3.0 -6.0
  6.0  3.0  3.0 -0.0  0.0 -3.0
  6.0  3.0  3.0 -0.0 -0.0 -3.0
  9.0  6.0  6.0  3.0  3.0 -0.0
Zone: 2
 -0.0 -3.0 -3.0 -6.0 -6.0 -9.0
  3.0 -0.0  0.0 -3.0 -3.0 -6.0
  3.0 -0.0 -0.0 -3.0 -3.0 -6.0
  6.0  3.0  3.0 -0.0  0.0 -3.0
  6.0  3.0  3.0 -0.0 -0.0 -3.0
  9.0  6.0  6.0  3.0  3.0 -0.0
Zone: 3
 -0.0 -3.0 -3.0 -6.0 -6.0 -9.0
  3.0 -0.0  0.0 -3.0 -3.0 -6.0
  3.0 -0.0 -0.0 -3.0 -3.0 -6.0
  6.0  3.0  3.0 -0.0  0.0 -3.0
  6.0  3.0  3.0 -0.0 -0.0 -3.0
  9.0  6.0  6.0  3.0  3.0 -0.0
Zone: 3a
 -0.0 -3.0 -3.0 -6.0 -6.0 -9.0
  3.0 -0.0  0.0 -3.0 -3.0 -6.0
  3.0 -0.0 -0.0 -3.0 -3.0 -6.0
  6.0  3.0  3.0 -0.0  0.0 -3.0
  6.0  3.0  3.0 -0.0 -0.0 -3.0
  9.0  6.0  6.0  3.0  3.0 -0.0
Zone: 3b
 -0.0 -3.0 -3.0 -6.0 -6.0 -9.0
  3.0 -0.0  0.0 -3.0 -3.0 -6.0
  3.0 -0.0 -0.0 -3.0 -3.0 -6.0
  6.0  3.0  3.0 -0.0  0.0 -3.0
  6.0  3.0  3.0 -0.0 -0.0 -3.0
  9.0  6.0  6.0  3.0  3.0 -0.0
Zone: 3c
 -0.0 -3.0 -3.0 -6.0 -6.0 -9.0
  3.0 -0.0  0.0 -3.0 -3.0 -6.0
  3.0 -0.0 -0.0 -3.0 -3.0 -6.0
  6.0  3.0  3.0 -0.0  0.0 -3.0
  6.0  3.0  3.0 -0.0 -0.0 -3.0
  9.0  6.0  6.0  3.0  3.0 -0.0
Zone: 4
 -0.0 -3.0 -3.0 -6.0 -6.0 -9.0
  3.0 -0.0  0.0 -3.0 -3.0 -6.0
  3.0 -0.0 -0.0 -3.0 -3.0 -6.0
  6.0  3.0  3.0 -0.0  0.0 -3.0
  6.0  3.0  3.0 -0.0 -0.0 -3.0
  9.0  6.0  6.0  3.0  3.0 -0.0
```

A data frame with only the reg_ss price for all the zones will allow analysis of prices between zones.

In [14]:
```
# collect all the regular self server gas prices
reg_ss_zones = { zn : zones[zn].reg_ss for zn in zone_names }
# specify the order of the column names
reg_ss_zones = pd.DataFrame( reg_ss_zones, columns = zone_names)

reg_ss_zones.head(10)
```

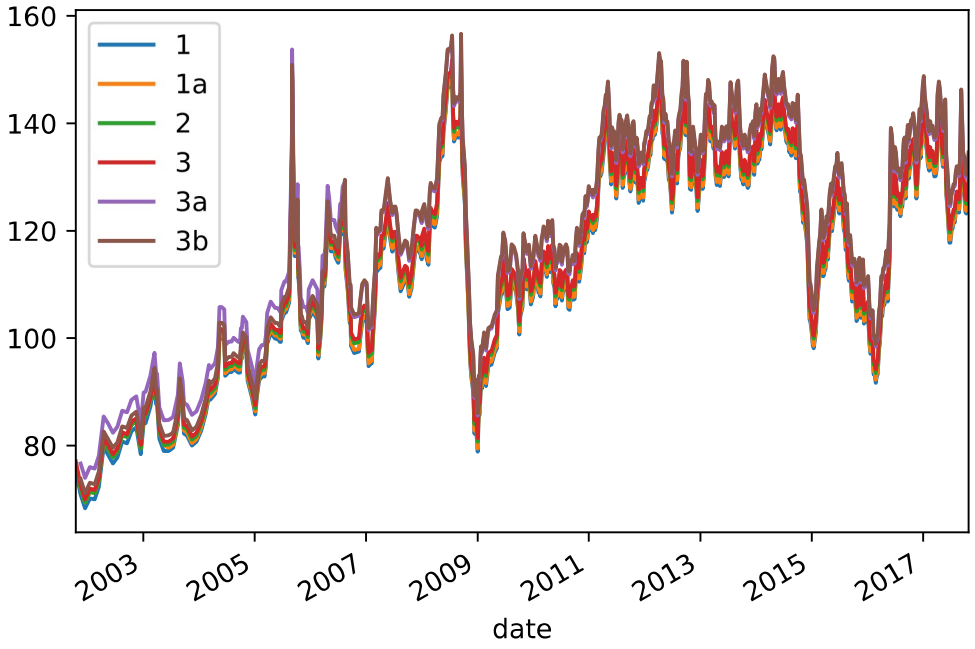
Out[14]:

	1	1a	2	3	3a	3b	3c	4	4a	5	...	9	10	10a	1
date															
2001-10-15	75.3	NaN	76.5	77.1	NaN	NaN	NaN	78.8	80.5	78.2	...	78.8	81.7	88.0	N
2001-11-15	70.9	72.1	72.1	72.7	76.7	73.8	76.1	74.4	81.3	73.8	...	74.4	77.8	85.3	N
2001-12-15	68.3	69.4	69.4	70.0	74.0	71.1	73.4	71.7	NaN	71.1	...	71.7	75.2	85.3	N
2002-01-15	70.2	71.4	71.4	72.0	76.0	73.1	75.4	73.7	NaN	73.1	...	73.7	77.1	85.3	N
2002-02-15	70.0	71.1	71.1	71.7	75.7	72.8	75.1	73.4	NaN	72.8	...	73.4	76.9	85.3	N
2002-03-15	72.3	73.5	73.5	74.0	78.1	75.2	77.5	75.8	NaN	75.2	...	75.8	79.2	85.3	N
2002-04-15	79.7	80.8	80.8	81.4	85.4	82.6	84.9	83.1	NaN	82.6	...	83.1	86.6	85.3	N
2002-05-15	78.3	79.5	79.5	80.1	84.1	81.2	83.5	81.8	NaN	81.2	...	81.8	85.2	85.3	N
2002-06-15	76.7	77.8	77.8	78.4	82.4	79.6	81.9	80.1	NaN	79.6	...	80.1	83.6	85.3	N
2002-07-15	77.8	78.9	78.9	79.5	83.5	80.6	82.9	81.2	NaN	80.6	...	81.2	84.7	92.1	N

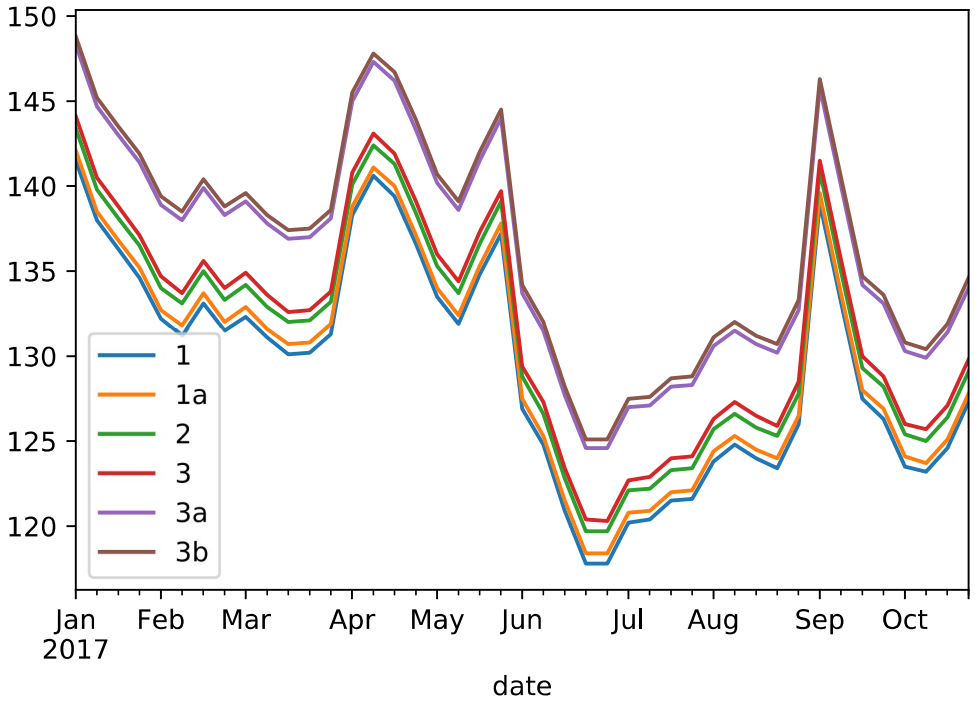
10 rows × 27 columns

In [15]:
```
# save the reg_ss prices
reg_ss_zones.to_csv('reg_ss_zones.csv')
```

In [16]:
```
reg_ss_zones[zone_names[:6]].plot(); None
```



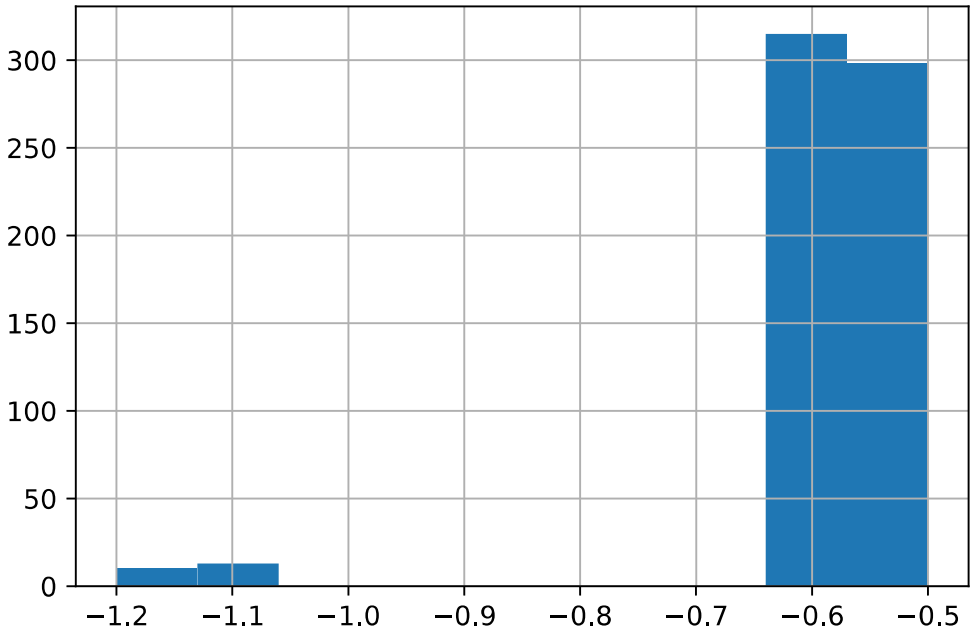
In [17]:
```
reg_ss_zones['2017':][zone_names[:6]].plot(); None
```



The price difference between zone 1 (Avalon Peninsula) and zone 1a (Bell Island) is shown by:

In [18]:
```
t = (reg_ss_zones['1'] - reg_ss_zones['1a'])
orig_len = len(t)
t = t.dropna() # drop any NaN rows
print('number of dropped samples', orig_len - len(t))
_ = t.hist()
```

```
number of dropped samples 1
```

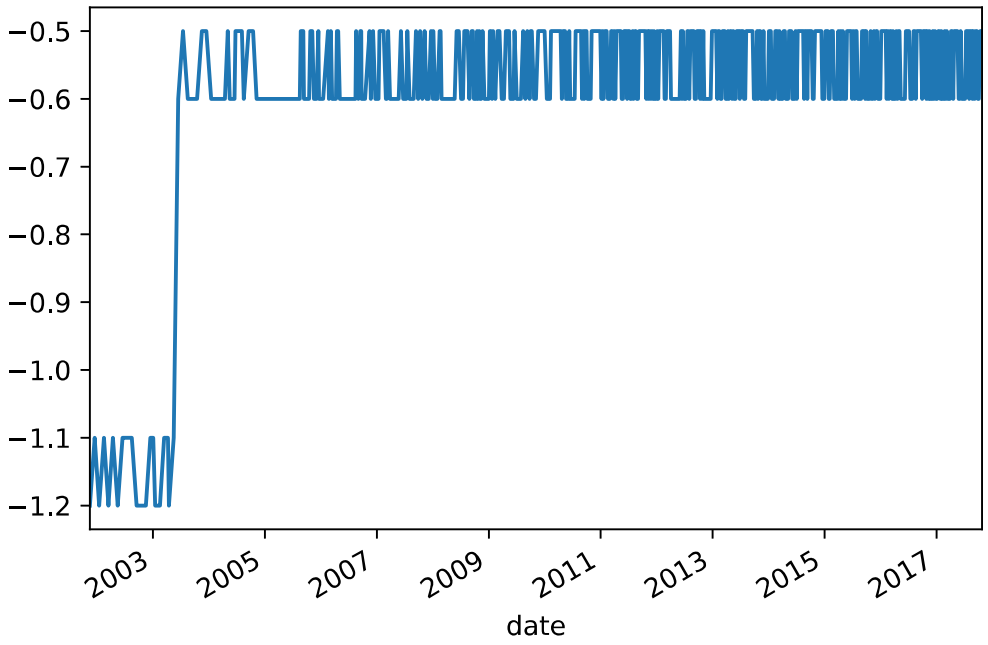


The price difference between zone 1 and 1a is mostly -0.5 cents, with several differences of around -1.15 cents.

A plot of the time series is

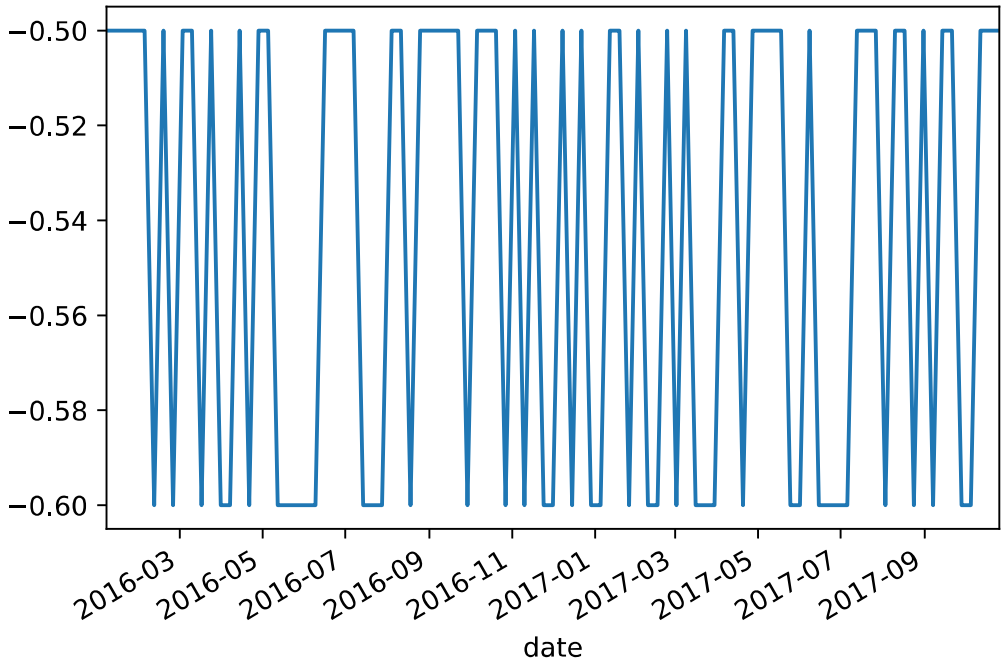
In [19]:
```
t.plot()
```

Out[19]: <matplotlib.axes._subplots.AxesSubplot at 0x7f6cb22151d0>



In [20]:
```
t['2016':].plot()
```

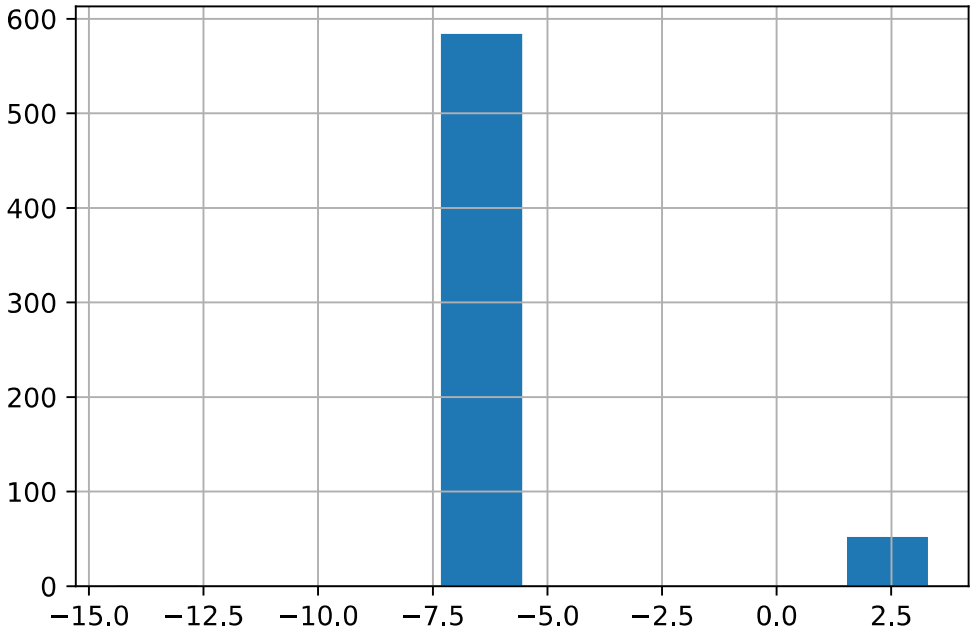
Out[20]: <matplotlib.axes._subplots.AxesSubplot at 0x7f6cb3b3c668>



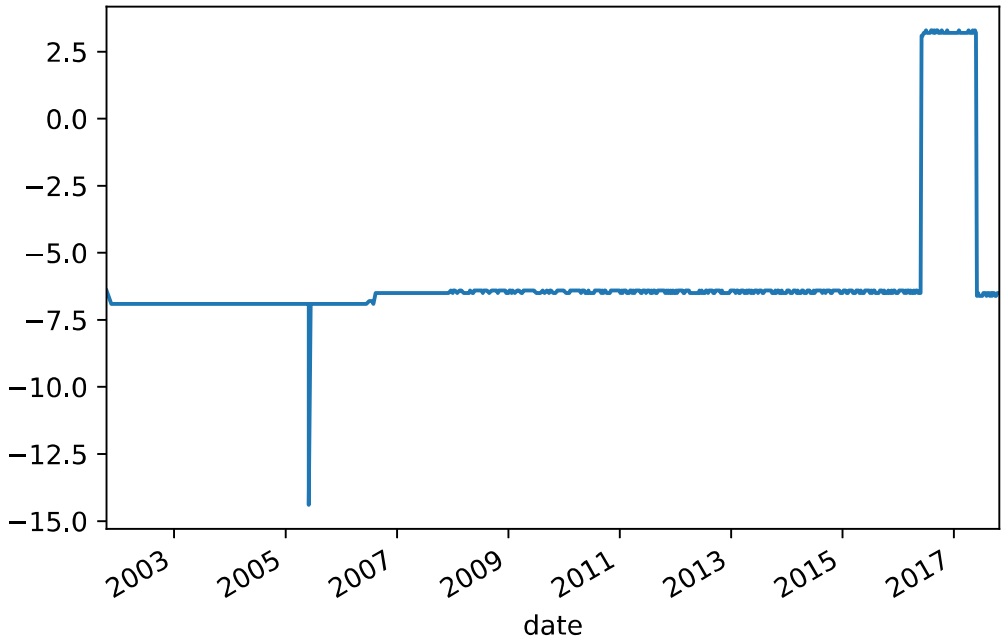
The difference for zone 1 and zone 10 (Labrador - The Straits to Red Bay) is:

In [21]:
```
t = (reg_ss_zones['1'] - reg_ss_zones['10'])
orig_len = len(t)
t = t.dropna() # drop any NaN rows
print('number of dropped samples', orig_len - len(t))
_ = t.hist()
```

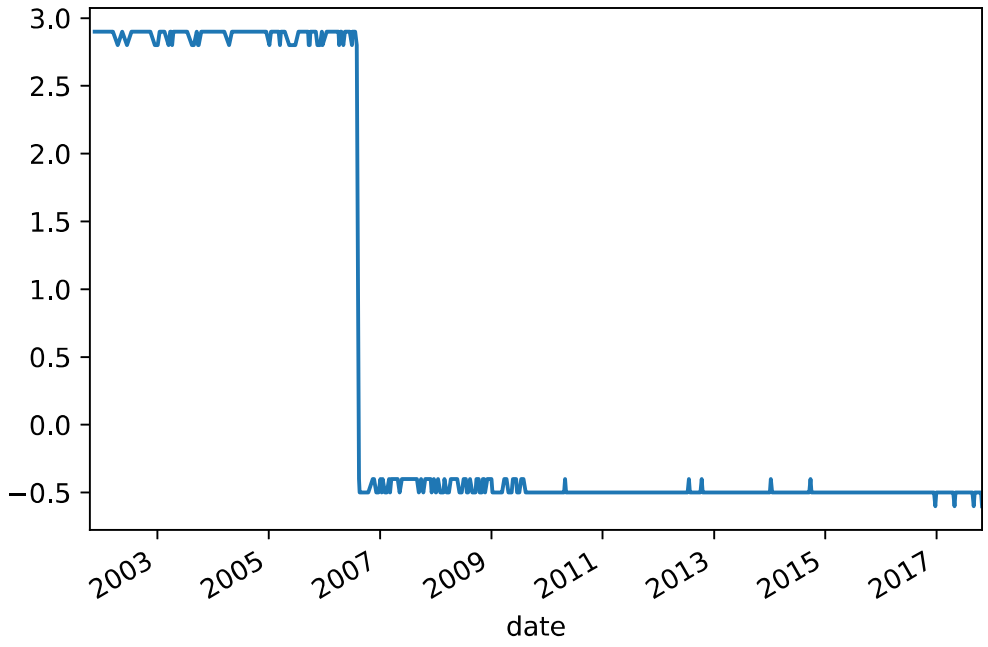
```
number of dropped samples 0
```



In [22]:
```
t.plot(); None
```



In [23]:
```
t = (reg_ss_zones['3a'] - reg_ss_zones['3b'])
t.plot(); None
```



In [24]:
```
t = (reg_ss_zones['3b'] - reg_ss_zones['3c'])
t.plot(); None
```