One for all, All for one (https://www.youtube.com/watch?v=FcMCd520Ae8)

SETI@home (https://en.wikipedia.org/wiki/SETI@home) liked Volunteer Computing (https://en.wikipedia.org/wiki/Volunteer_computing) on Mixed-Platforms Cluster - using Celery and Docker

Wei Lin
20160512

Experiment procedure:

Define Dockerfile for RPi:

Celery Worker Dockerfile for ARM v7
image name: wei1234c/one_for_all_all_for_one_armv7

In [ ]:
```
# one_for_all_all_for_one_armv7
# Celery Worker Dockerfile
# for ARM v7
# 20160512

FROM wei1234c/celery_armv7

MAINTAINER Wei Lin

USER root

RUN mkdir /celery_projects

WORKDIR /celery_projects

COPY . /celery_projects/

RUN chmod +x /celery_projects/start_workers.sh

USER pi

CMD ["/bin/sh", "/celery_projects/start_workers.sh"]
```

Build Docker image for RPi:

image name: wei1234c/one_for_all_all_for_one_armv7

In [ ]:
```
HypriotOS: pi@rpi202 in ~
$ docker build -t wei1234c/one_for_all_all_for_one_armv7 /dockerfiles/ARMv7/one_for_all_all_for_one
Sending build context to Docker daemon 9.216 kB
Step 1 : FROM wei1234c/celery_armv7
 ---> 8939b7e5c928
Step 2 : MAINTAINER Wei Lin
 ---> Using cache
 ---> 186c6ea155e8
Step 3 : USER root
 ---> Using cache
 ---> b84c84193d65
Step 4 : RUN mkdir /celery_projects
 ---> Using cache
 ---> fff839303a93
Step 5 : WORKDIR /celery_projects
 ---> Using cache
 ---> 27938d9d6ae8
Step 6 : COPY . /celery_projects/
 ---> c120021c8dc0
Removing intermediate container 5b4421e5472b
Step 7 : RUN chmod +x /celery_projects/start_workers.sh
 ---> Running in f719c550c90d
 ---> efa549c140dd
Removing intermediate container f719c550c90d
Step 8 : USER pi
 ---> Running in 1c7a5587769b
 ---> 427f37517ba9
Removing intermediate container 1c7a5587769b
Step 9 : CMD /bin/sh /celery_projects/start_workers.sh
 ---> Running in ed24629cb0ba
 ---> d7bb6603b6c6
Removing intermediate container ed24629cb0ba
Successfully built d7bb6603b6c6
HypriotOS: pi@rpi202 in ~
$
```

Define Dockerfile for amd64:

Celery Worker Dockerfile for amd64
image name: wei1234c/one_for_all_all_for_one

In [ ]:
```
# one_for_all_all_for_one
# Celery Worker Dockerfile
# for amd64
# 20160512

FROM ubuntu

MAINTAINER Wei Lin

USER root

# Add user pi
RUN \
    useradd -G adm,sudo,users -s /bin/bash -m pi && \
    echo 'pi:raspberry' | chpasswd


#RUN pip3 install pandas

# Install Python. ____________________________________________________________________________________________
RUN apt-get update && \
    apt-get install -y python3 python3-pip python3-dev python3-numpy python3-scipy python3-matplotlib python3-pandas && \
    apt-get install -y python python-pip python-dev

# Install Celery  ____________________________________________________________________________________________
RUN \
    pip3 install -U celery

RUN \
    pip3 install -U redis

RUN mkdir /celery_projects

WORKDIR /celery_projects

COPY . /celery_projects/

RUN chmod +x /celery_projects/start_workers.sh

USER pi
```

```
CMD ["/bin/sh", "/celery_projects/start_workers.sh"]
```

Build Docker image for amd64:

image name: wei1234c/one_for_all_all_for_one

In [ ]:
```
wei@Wei-Lenovo:~$ docker build -t wei1234c/one_for_all_all_for_one /docker/dockerfiles/amd64/one_for_all_all_for_one
Sending build context to Docker daemon 9.728 kB
Step 1 : FROM ubuntu
 ---> c5f1cf30c96b
Step 2 : MAINTAINER Wei Lin
 ---> Using cache
 ---> 626cc4694d46
Step 3 : USER root
 ---> Using cache
 ---> 9212cedf802b
Step 4 : RUN useradd -G adm,sudo,users -s /bin/bash -m pi &&    echo 'pi:raspberry' | chpasswd
 ---> Using cache
 ---> a237ec2f3a84
Step 5 : RUN apt-get update &&     apt-get install -y python3 python3-pip python3-dev python3-numpy python3-scipy python3
 ---> Using cache
 ---> f65b7e004075
Step 6 : RUN pip3 install -U celery
 ---> Using cache
 ---> 3764134da5f1
Step 7 : RUN pip3 install -U redis
 ---> Using cache
 ---> 61772207fc08
Step 8 : RUN mkdir /celery_projects
 ---> Using cache
 ---> c68f9dc73b5c
Step 9 : WORKDIR /celery_projects
 ---> Using cache
 ---> b9e490c48b98
Step 10 : COPY . /celery_projects/
 ---> Using cache
 ---> 55e921f0a082
Step 11 : RUN chmod +x /celery_projects/start_workers.sh
 ---> Using cache
 ---> 194e82a97639
Step 12 : USER pi
 ---> Using cache
 ---> 740675730169
Step 13 : CMD /bin/sh /celery_projects/start_workers.sh
 ---> Using cache
 ---> d77e8341bf85
Successfully built d77e8341bf85
```

```
wei@Wei-Lenovo:~$
```

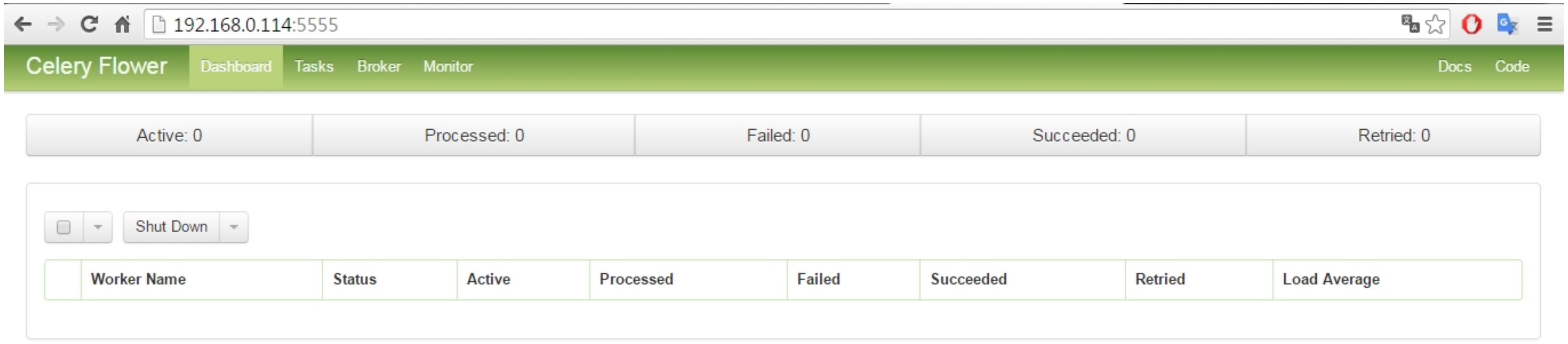
Run container for Celery Broker, using Redis

In [ ]:
```
HypriotOS: pi@rpi202 in ~
$ docker run -d -p 6379:6379 --name=redis --volume=/data:/data hypriot/rpi-redis
2ee100973b0e1317e7511de0c97b2a29ad02a688f9928c14f347922a4aa3fb5d

HypriotOS: pi@rpi202 in ~
$ docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED             STATUS              PORTS
2ee100973b0e        hypriot/rpi-redis   "/entrypoint.sh redis"   8 seconds ago       Up 7 seconds        0.0.0.0:6379->63
HypriotOS: pi@rpi202 in ~
$
```

Run Flower (http://docs.celeryproject.org/en/latest/userguide/monitoring.html#flower-real-time-celery-web-monitor) container for monitoring

In [ ]:
```
HypriotOS: pi@rpi202 in ~
$ docker run -d -p 5555:5555 --name=flower wei1234c/one_for_all_all_for_one_armv7 /bin/sh -c "cd /celery_projects && cele
3c6e9e85417b536d07562575711e5f288097ed48d6f12c0129155d01ea746e66

HypriotOS: pi@rpi202 in ~
$ docker ps
CONTAINER ID        IMAGE                                    COMMAND                  CREATED             STATUS
3c6e9e85417b        wei1234c/one_for_all_all_for_one_armv7   "/bin/sh -c 'cd /cele"   9 seconds ago       Up 7 seconds
2ee100973b0e        hypriot/rpi-redis                        "/entrypoint.sh redis"   2 hours ago         Up 2 hours
HypriotOS: pi@rpi202 in ~
$
```

No worker running, nothing shows in Flower.


192.168.0.114:5555

Active: 0 | Processed: 0 | Failed: 0 | Succeeded: 0 | Retried: 0

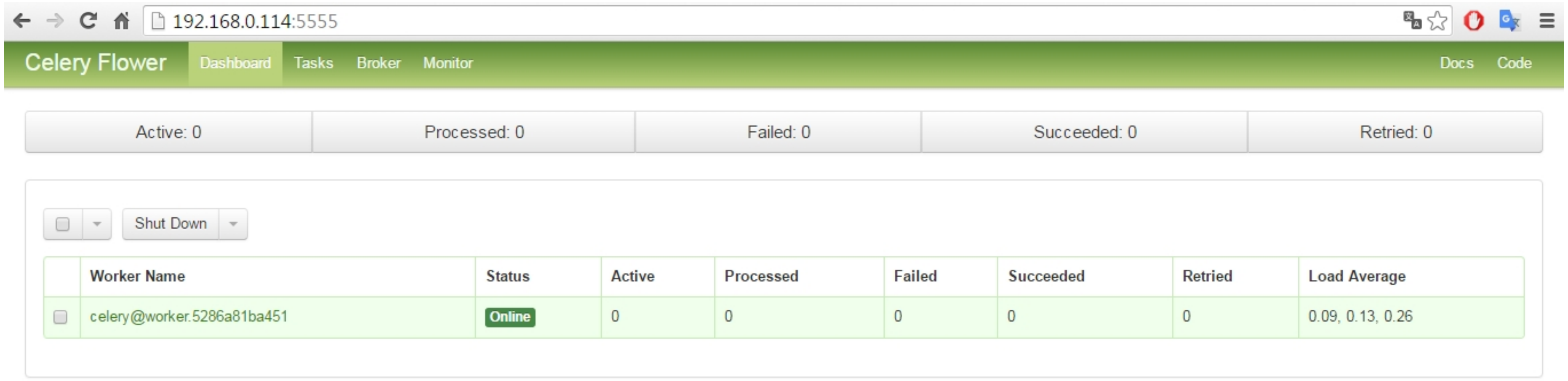
	Worker Name	Status	Active	Processed	Failed	Succeeded	Retried	Load Average

Run Celery worker container on Raspberry Pi

Volunteer can join the cluster for distributed parallel computing, simply by running this Docker image.

In [ ]:
```
HypriotOS: pi@rpi202 in ~
$ docker run -d --name=musketeer1 wei1234c/one_for_all_all_for_one_armv7
5286a81ba451b8473ef2b5e3bb965a7b0fc31511e1ed4e368659aece2478e053

HypriotOS: pi@rpi202 in ~
$ docker ps
CONTAINER ID        IMAGE                                    COMMAND                  CREATED             STATUS
5286a81ba451        wei1234c/one_for_all_all_for_one_armv7   "/bin/sh /celery_proj"   4 seconds ago       Up 3 seconds
3c6e9e85417b        wei1234c/one_for_all_all_for_one_armv7   "/bin/sh -c 'cd /cele"   5 minutes ago       Up 5 minutes
2ee100973b0e        hypriot/rpi-redis                        "/entrypoint.sh redis"   2 hours ago         Up 2 hours
HypriotOS: pi@rpi202 in ~
$
```

After Celery worker container started, it showed up in Flower. However, no task message was received yet.



Active: 0	Processed: 0	Failed: 0	Succeeded: 0	Retried: 0

	Worker Name	Status	Active	Processed	Failed	Succeeded	Retried	Load Average
	celery@worker.5286a81ba451	Online	0	0	0	0	0	0.09, 0.13, 0.26

In package "stock" there is a file "tasks.py", containing function "get_table" with which we define task message.

```
In [ ]: from stock.celery import app

        import pandas as pd
        from datetime import datetime


        def get_url(stock_id, year = datetime.today().year, month = datetime.today().month):
            return 'http://www.twse.com.tw/ch/trading/exchange/STOCK_DAY/genpage/Report{year}{month:02}/{year}{month:02}_F3_1_8_{


        @app.task
        def get_table(stock_id, year = datetime.today().year, month = datetime.today().month):

            url = get_url(stock_id, year, month)
            targetTableIndex = 0

            table = pd.read_html(url,
                                 attrs = {'border': '0' ,
                                          'width': '598',
                                          'align': 'center',
                                          'cellpadding': '0',
                                          'cellspacing': '1',
                                          'class': 'board_trad'},
                                 header = 1
                                )[targetTableIndex]

            table['stock_id'] = stock_id
            table = table.reindex(columns = ['stock_id', 'date', 'quantity', 'amount', 'open', 'highest', 'lowest', 'close', 'off

            return table.tail(1).values
```

```
In [14]: # Load stock.tasks, which contains the definition of function "get_table".
         from stock.tasks import *
         import numpy as np
```

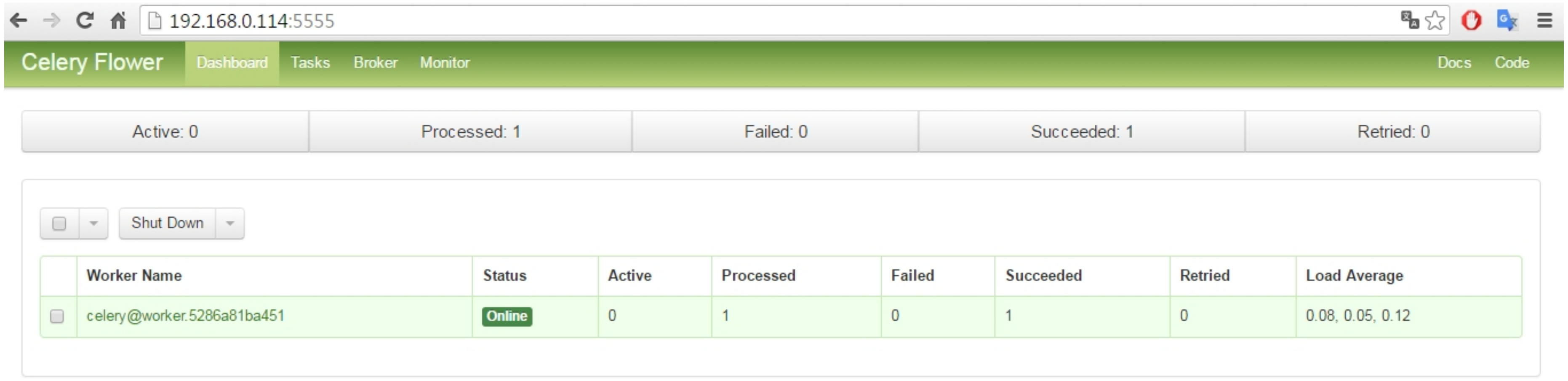
```
In [7]: # Excute get_table funtion directly from local host (OS: Windows 7)
        # No task message was sent to Celery broker
        get_table(2356)

Out[7]: array([[2356, '105/05/12', 5614182, 114086911, 20.2, 20.55, 20.1, 20.3,
                -0.1, 3159]], dtype=object)
```

Asynchronous function call

In IPython Notebook on local host, we sent a task message to Celery cluster, demanding computing service.
There is no Docker mechanism on local host (OS: Windows 7).
With "get_table.apply_async()" a task message will be sent to Celery Broker.
Celery Broker will put the task message into a queue.
Worker on Raspberry Pi will pick up the message from queue and excute it, and return the result.

```
In [8]: r = get_table.apply_async(args = [2356])
        r.get()

Out[8]: array([[2356, '105/05/12', 5614182, 114086911, 20.2, 20.55, 20.1, 20.3,
                -0.1, 3159]], dtype=object)
```

Soon after we sent a task message to Celery broker, it shows in Flower that there is a task processed successfully.



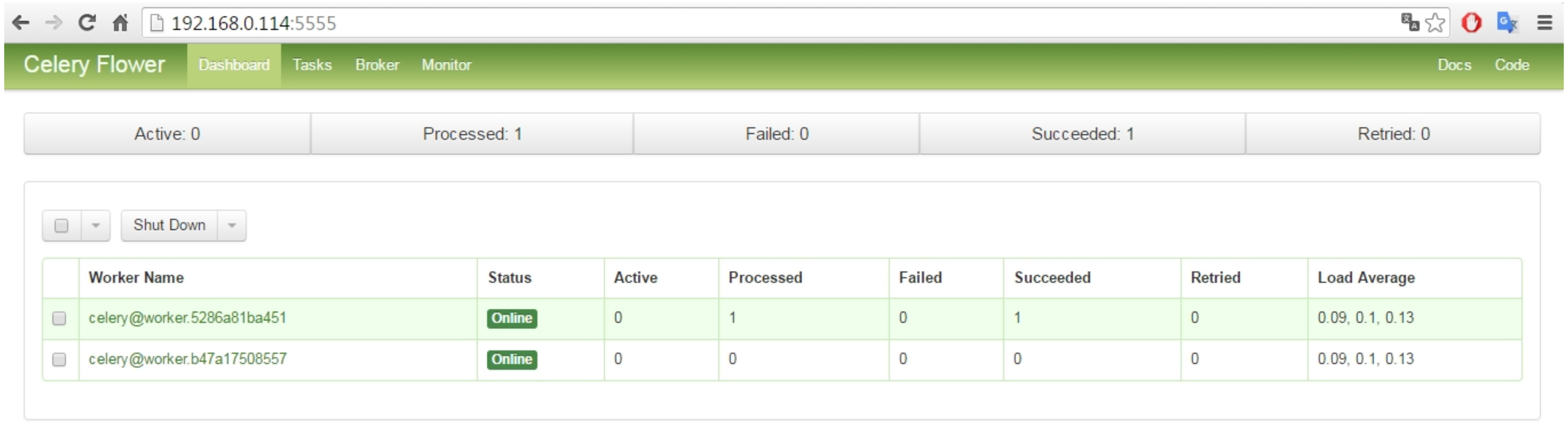
Active: 0	Processed: 1	Failed: 0	Succeeded: 1	Retried: 0

	Worker Name	Status	Active	Processed	Failed	Succeeded	Retried	Load Average
☐	celery@worker.5286a81ba451	Online	0	1	0	1	0	0.08, 0.05, 0.12

In Raspberry Pi, we run another Celery worker container.

In [ ]:
```
HypriotOS: pi@rpi202 in ~
$ docker run -d --name=musketeer2 wei1234c/one_for_all_all_for_one_armv7
b47a17508557cd48bbb21d48d7ad6b652e492058cf209d1c920a28db361e3568

HypriotOS: pi@rpi202 in ~
$ docker ps
CONTAINER ID        IMAGE                                    COMMAND                  CREATED             STATUS
b47a17508557        wei1234c/one_for_all_all_for_one_armv7   "/bin/sh /celery_proj"   4 seconds ago       Up 3 seconds
5286a81ba451        wei1234c/one_for_all_all_for_one_armv7   "/bin/sh /celery_proj"   21 minutes ago      Up 21 minutes
3c6e9e85417b        wei1234c/one_for_all_all_for_one_armv7   "/bin/sh -c 'cd /cele"   26 minutes ago      Up 26 minutes
2ee100973b0e        hypriot/rpi-redis                        "/entrypoint.sh redis"   3 hours ago         Up 3 hours
HypriotOS: pi@rpi202 in ~
$
```

We can see in Flower that we have two workers running now.



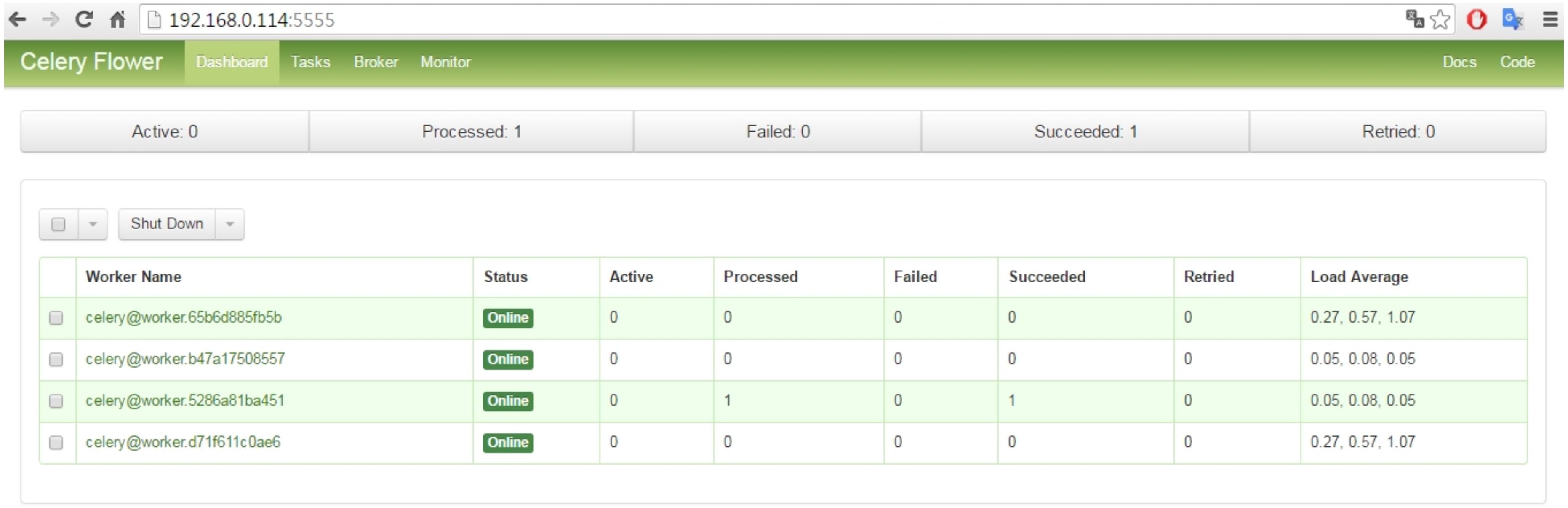
Active: 0 | Processed: 1 | Failed: 0 | Succeeded: 1 | Retried: 0

	Worker Name	Status	Active	Processed	Failed	Succeeded	Retried	Load Average
☐	celery@worker.5286a81ba451	Online	0	1	0	1	0	0.09, 0.1, 0.13
☐	celery@worker.b47a17508557	Online	0	0	0	0	0	0.09, 0.1, 0.13

On an AMD64 machine, we run two Celery worker containers.

In [ ]:
```
wei@Wei-Lenovo:~$ docker run -d --name=musketeer3 wei1234c/one_for_all_all_for_one
65b6d885fb5bf06f10517c79325d19639446d939a0b2395aada323674e2eb121

wei@Wei-Lenovo:~$ docker run -d --name=musketeer4 wei1234c/one_for_all_all_for_one
d71f611c0ae6123f5139f02b1a5a5936f162259271061e7c256cc06d9b9d2511

wei@Wei-Lenovo:~$ docker ps
CONTAINER ID        IMAGE                              COMMAND                  CREATED              STATUS
d71f611c0ae6        wei1234c/one_for_all_all_for_one   "/bin/sh /celery_proj"   6 seconds ago        Up 3 seconds
65b6d885fb5b        wei1234c/one_for_all_all_for_one   "/bin/sh /celery_proj"   About a minute ago   Up About a minute
wei@Wei-Lenovo:~$
```

Now in Flower, we can see four workers - two workers on Raspberry Pi, and another two workers on the AMD64 machine.



Active: 0 | Processed: 1 | Failed: 0 | Succeeded: 1 | Retried: 0

Worker Name	Status	Active	Processed	Failed	Succeeded	Retried	Load Average
celery@worker.65b6d885fb5b	Online	0	0	0	0	0	0.27, 0.57, 1.07
celery@worker.b47a17508557	Online	0	0	0	0	0	0.05, 0.08, 0.05
celery@worker.5286a81ba451	Online	0	1	0	1	0	0.05, 0.08, 0.05
celery@worker.d71f611c0ae6	Online	0	0	0	0	0	0.27, 0.57, 1.07

In [29]:
```
# The stock id list, about wich we will collect data.
stocks = [1101, 1102, 1103, 1104, 1108, 1109, 1110, 1201, 1203, 1210, 1213, 1215, 1216, 1217, 1218, 1219, 1220,
          1225, 1227, 1229, 1231, 1232, 1233, 1234, 1235, 1236, 1256, 1702, 1737, 1301]
```

In [30]:
```
from pandas import DataFrame

# collect data into a Pandas.DataFrame
def reduce(results):
    data = []
    for result in results: data.append(result[0])

    table = DataFrame(list(data),
                      columns = ['stock_id', 'date', 'quantity', 'amount', 'open', 'highest', 'lowest', 'close', 'offset'

    return table
```

```
In [31]: def get_stock_prices(stocks):

             # send task messages to Celery broker
             asyncResults = [get_table.apply_async(args = [stock]) for stock in stocks]

             # get results from AsyncResults into a list
             results = [asyncResult.get() for asyncResult in asyncResults if asyncResult.get() is not None]

             return reduce(results)

         %time prices = get_stock_prices(stocks)
         prices[:5]
```

```
Wall time: 19.2 s
```

Out[31]:

	stock_id	date	quantity	amount	open	highest	lowest	close	offset	trades
0	1101	105/05/12	3242925	96161650	29.50	29.80	29.45	29.70	0.05	2064
1	1102	105/05/12	3094327	80204709	25.60	26.30	25.55	26.00	0.40	2046
2	1103	105/05/12	56511	492804	8.78	8.78	8.70	8.74	-0.04	32
3	1104	105/05/12	138794	2766347	20.15	20.15	19.85	20.00	-0.15	85
4	1108	105/05/12	85995	865446	10.05	10.10	10.00	10.05	0.00	43

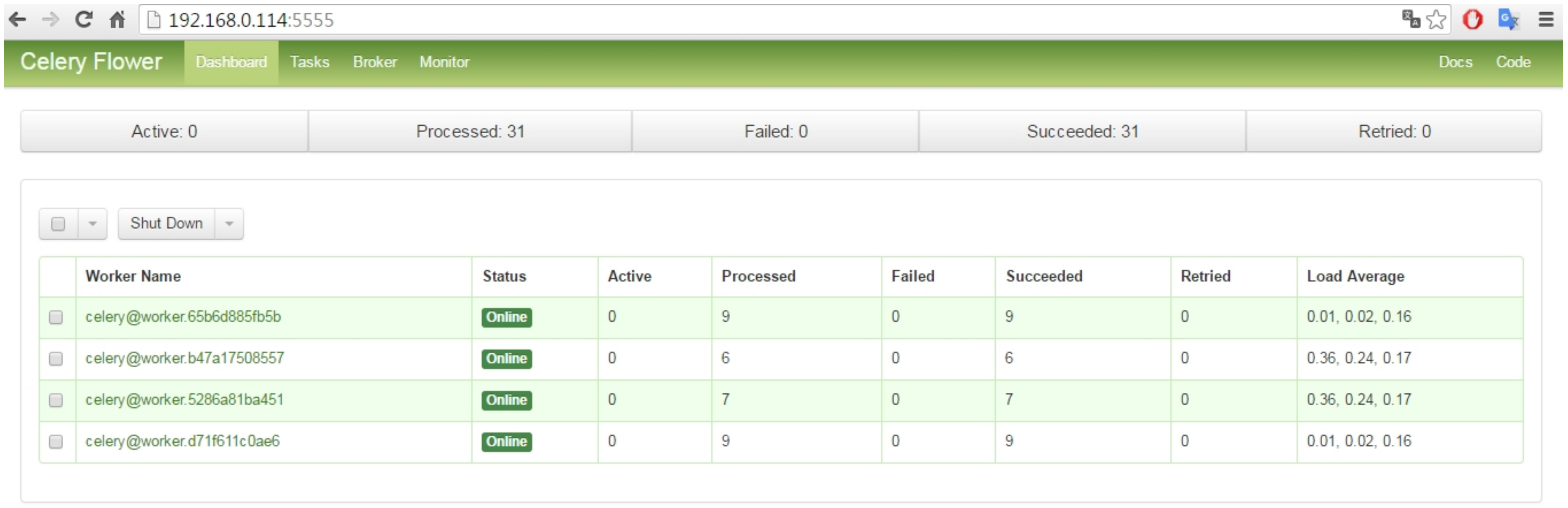
In [32]:
```
# list all results
prices
```

Out[32]:

	stock_id	date	quantity	amount	open	highest	lowest	close	offset	trades
0	1101	105/05/12	3242925	96161650	29.50	29.80	29.45	29.70	0.05	2064
1	1102	105/05/12	3094327	80204709	25.60	26.30	25.55	26.00	0.40	2046
2	1103	105/05/12	56511	492804	8.78	8.78	8.70	8.74	-0.04	32
3	1104	105/05/12	138794	2766347	20.15	20.15	19.85	20.00	-0.15	85
4	1108	105/05/12	85995	865446	10.05	10.10	10.00	10.05	0.00	43
5	1109	105/05/12	4000	40050	10.05	10.05	10.00	10.00	0.00	4
6	1110	105/05/12	31000	445500	14.20	14.45	14.20	14.45	0.00	13
7	1201	105/05/12	1015742	19595770	19.65	19.70	19.00	19.10	-0.60	384
8	1203	105/05/12	12004	257184	21.55	21.55	21.40	21.40	-0.30	11
9	1210	105/05/12	4576158	107022964	23.05	23.75	23.00	23.40	0.50	1946
10	1213	105/05/12	26010	447070	17.30	17.30	17.15	17.15	-0.10	21
11	1215	105/05/12	4658698	135637699	28.65	29.40	28.65	28.95	0.25	1992
12	1216	105/05/12	5823532	337920756	57.30	58.50	57.30	58.40	1.10	2945
13	1217	105/05/12	430450	3402141	7.85	8.00	7.81	7.85	0.00	173
14	1218	105/05/12	268733	3345762	12.45	12.55	12.35	12.40	-0.15	106
15	1219	105/05/12	37208	562998	15.20	15.20	15.00	15.05	-0.15	17
16	1220	105/05/12	50000	519600	10.45	10.50	10.30	10.45	0.00	25
17	1225	105/05/12	55382	1757142	31.65	31.90	31.20	31.20	-0.45	19
18	1227	105/05/12	511892	39680787	77.50	77.80	77.20	77.60	-0.10	401
19	1229	105/05/12	354427	7157504	20.40	20.40	20.05	20.10	0.10	183
20	1231	105/05/12	619161	18610131	29.25	30.55	29.25	30.20	1.05	328

	stock_id	date	quantity	amount	open	highest	lowest	close	offset	trades
21	1232	105/05/12	118000	9158100	77.30	77.90	77.30	77.40	0.10	55
22	1233	105/05/12	62200	2293460	36.80	37.05	36.80	36.85	0.05	32
23	1234	105/05/12	207595	6888635	33.60	33.60	32.95	32.95	-0.95	107
24	1235	105/05/12	25160	595842	23.45	23.95	23.30	23.90	0.00	23
25	1236	105/05/12	18010	361551	20.15	20.15	20.00	20.00	-0.15	16
26	1256	105/05/12	13150	1573200	120.00	120.00	118.50	118.50	-2.50	13
27	1702	105/05/12	1020054	63916873	61.70	63.20	61.70	62.60	0.60	601
28	1737	105/05/12	207912	5610143	27.00	27.10	26.85	26.85	-0.10	97
29	1301	105/05/12	3896203	300815138	77.00	77.40	76.80	77.10	0.10	2088

Tasks were distributed among four workers - two wokers on Raspberry Pi, and another two wokers on the AMD64 machine.



Active: 0 | Processed: 31 | Failed: 0 | Succeeded: 31 | Retried: 0

Worker Name	Status	Active	Processed	Failed	Succeeded	Retried	Load Average
celery@worker.65b6d885fb5b	Online	0	9	0	9	0	0.01, 0.02, 0.16
celery@worker.b47a17508557	Online	0	6	0	6	0	0.36, 0.24, 0.17
celery@worker.5286a81ba451	Online	0	7	0	7	0	0.36, 0.24, 0.17
celery@worker.d71f611c0ae6	Online	0	9	0	9	0	0.01, 0.02, 0.16