FOLLOW YOUR CELERY TASKS WITH DIRECTOR

OVHcloud

Summary

- What is Celery
- Tasks & Workflows (demo)

- As a developer, I need…
- How to use Director (demo)

1.

What is Celery

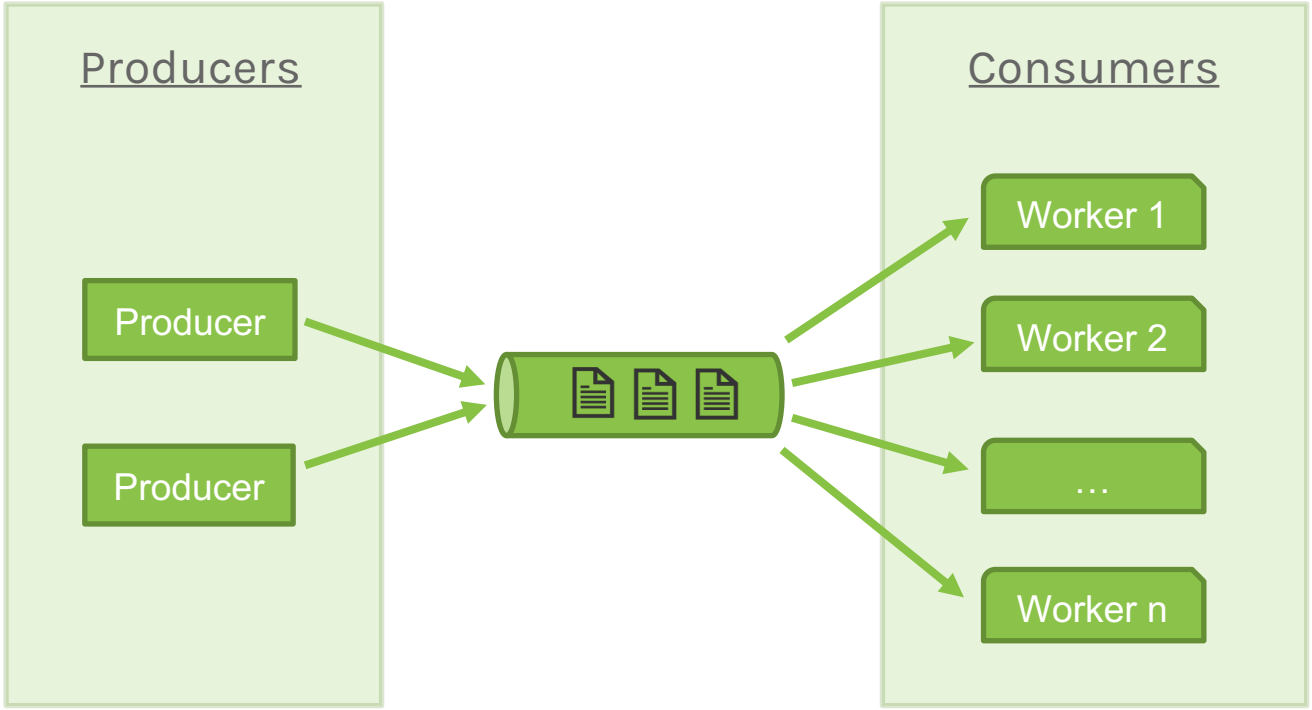
“

Celery is an asynchronous **task queue** based on distributed message passing. It is focused on real-time operation, but supports scheduling as well.

http://www.celeryproject.org/

Task Queue

- Distribute work across threads or machines



So what is Celery ??

Tl;dr

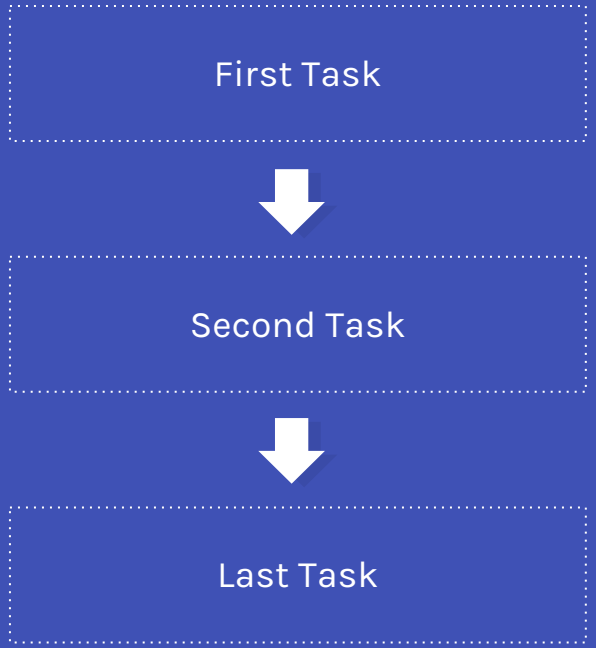
Execute a Python function “*somewhere else*”.

Why ?

- do not block the user,
- not enough resources,
- insufficient access,
- …

2.

Tasks & Workflows



Celery API

- Task is send to the broker using **.delay()**
- Worker are launched using the **celery worker** command

- Tasks can be combined to create **workflows**
- Celery Primitives : chain, group, chord, maps…

Demo

(demo) Task

```
$ cat tasks.py
from celery import Celery

app = Celery("tasks", broker="redis://127.0.0.1:6379/0")

@app.task
def add(x, y):
    return x + y


$ python -c "from tasks import add; print(add(1, 2))"
3


$ python -c "from tasks import add; print(add.delay(1, 2))"
fedc3561-579e-4d8d-8cbd-94ed5e1af50f
```

(demo) Task

```
$ celery -A tasks worker --loglevel=info
```

```
[2020-01-22 18:27:56,238: INFO/MainProcess] Received task:
tasks.add[fedc3561-579e-4d8d-8cbd-94ed5e1af50f]

[2020-01-22 18:27:56,243: INFO/ForkPoolWorker-2] Task tasks.add[fedc3561-
579e-4d8d-8cbd-94ed5e1af50f] succeeded in 0.0013022100320085883s: 3
```

(demo) Workflow

```
from celery import chain

@app.task
def add(x, y):
    return x + y

sign = chain(
    add.s(1, 2),
    add.s(7),
    add.s(10)
)
```

```
>>> from tasks import sign
>>> sign.delay()
<AsyncResult: af0796e9-ca4d-4c70-ad75-b36a0d9869ac>
```

(demo) Workflow

```
[INFO/MainProcess] Received task: tasks.add[f7ce45e2-79e7-4733-820c-
ba4c2adf705d]
[INFO/ForkPoolWorker-2] Task tasks.add[f7ce45e2-79e7-4733-820c-
ba4c2adf705d] succeeded in 0.04351018695160746s: 3


[INFO/MainProcess] Received task: tasks.add[1bea4889-f7ed-46fa-9ace-
f3f59f7b9713]
[INFO/ForkPoolWorker-2] Task tasks.add[1bea4889-f7ed-46fa-9ace-
f3f59f7b9713] succeeded in 0.0018125399947166443s: 10


[INFO/MainProcess] Received task: tasks.add[af0796e9-ca4d-4c70-ad75-
b36a0d9869ac]
[INFO/ForkPoolWorker-2] Task tasks.add[af0796e9-ca4d-4c70-ad75-
b36a0d9869ac] succeeded in 0.00010735599789768457s: 20
```

(demo) Workflow

```
from celery import group

@app.task
def add(x, y):
    return x + y

sign = group(
    add.s(1, 2),
    add.s(3, 4)
)
```

```
>>> from tasks import sign
>>> sign.delay()
<GroupResult: 36e1f739-ef0a-4fcb-b3ad-962058a21b43
[4dba3012-e696-4746-ac20-a92a2200dd4f, d1f20413-519d-4e48-
8ae8-0271521c9740]>
```

(demo) Workflow

```
[INFO/MainProcess] Received task: tasks.add[4dba3012-e696-4746-ac20-
a92a2200dd4f]
[INFO/MainProcess] Received task: tasks.add[d1f20413-519d-4e48-8ae8-
0271521c9740]

[INFO/ForkPoolWorker-2] Task tasks.add[4dba3012-e696-4746-ac20-
a92a2200dd4f] succeeded in 0.0012087000068277125s: 3
[INFO/ForkPoolWorker-3] Task tasks.add[d1f20413-519d-4e48-8ae8-
0271521c9740] succeeded in 0.0014324290095828474s: 7
```

3.

As a developer, I need...

Our needs as Celery developers

- **track tasks evolution in the WebUI**
- execute them using API and CLI
- create workflows in YAML
- be able to periodically execute them
- relaunch a failed workflow at a specific task

4.

How to use Director

Demo

(demo) No more boilerplate

```python
from celery import chain, group
from tasks import *

ovh_etl = chain(
    extract.s(),
    transform.s(),
    group(
        load_in_db.s(),
        load_in_fs.s()
    )
)
```

```yaml
ovh.ETL:
  tasks:
    - EXTRACT
    - TRANSFORM
    - LOADS:
        type: group
        tasks:
          - LOAD_IN_DB
          - LOAD_IN_FS
```

(demo) Execute a workflow

```
$ python
>>> from tasks import ovh_etl
>>> ovh_etl.delay()
```

```
$ director run ovh.ETL '{}'
```

```
$ curl
    --header "Content-Type: application/json" \
    --request POST \
    --data '{"project":"ovh", "name": "ETL", "payload": {}}' \
    http://localhost:8000/api/workflows
```

Celery Director

Workflows

Search

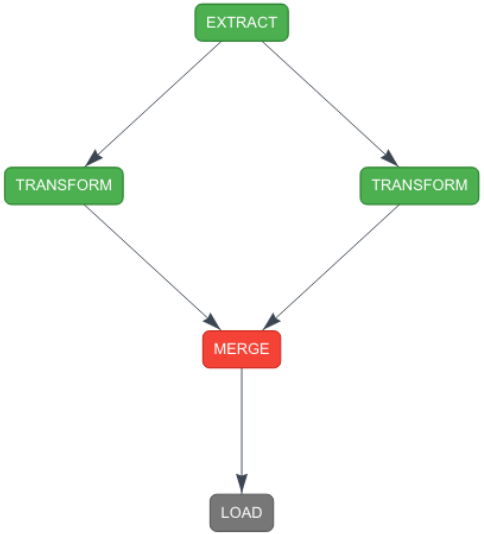
Name	Date ↓	Status
ovh.GROUPS_ETL	Jan 29, 18:58:58	success
ovh.GROUPS_ETL	Jan 29, 18:58:48	success
ovh.ETL_2	Jan 29, 18:58:11	error
ovh.GROUPS_ETL	Jan 29, 18:56:37	success
ovh.GROUPS_ETL	Jan 29, 18:54:59	success
ovh.GROUPS_ETL	Jan 29, 18:54:48	success
ovh.GROUP_ETL	Jan 29, 18:53:05	success
ovh.SIMPLE_ETL_2	Jan 29, 18:46:32	success
ovh.SIMPLE_ETL_2	Jan 29, 18:43:48	success
ovh.SIMPLE_ETL_2	Jan 29, 18:43:23	success

OVH.ETL_2

Jan 29, 18:58:11 error

Tasks : Total **5** / Success **3** / Error **1**

PAYLOAD RELAUNCH



EXTRACT
Jan 29, 18:58:12
f7f6ec66-7859-4e25-9081-c214d8cb007b

TRANSFORM
Jan 29, 18:58:12
7c661796-841c-4777-9d04-ca1327428020

TRANSFORM
Jan 29, 18:58:12
5d2089bc-27a7-496c-8552-f88810a6377d

MERGE
Jan 29, 18:58:12
741ecb71-d79f-4381-a95e-e23a97086ab7

LOAD
Jan 29, 18:58:11
0b37d3fe-3247-420d-911d-9bed73293ec1

Flower Dashboard Tasks Broker Monitor Logout Docs Code

MERGE dc96d681-be77-434a-b231-ed6f667a6336

Basic task options

Name	MERGE
UUID	dc96d681-be77-434a-b231-ed6f667a6336
State	FAILURE
args	([None, None],)
kwargs	{'workflow_id': '3a1a26bd-4664-4e40-b65e-4a646b6f35bd', 'payload': {}}
Result	None

Advanced task options

Received	2020-01-31 13:30:31.648077 UTC
Started	2020-01-31 13:30:31.649547 UTC
Failed	2020-01-31 13:30:31.672246 UTC
Retries	0
Worker	celery@localhost
Exception	ZeroDivisionError('division by zero',)
Timestamp	2020-01-31 13:30:31.672246 UTC

Traceback	Traceback (most recent call last):
	  File "/Users/ovh/myvenvs/director/lib/python3.6/site-packages/celery-4.4.0-py3.6.egg/celery/app/trace.py", line 385, in trace_task
	  R = retval = fun(*args, **kwargs)
	  File "/Users/ovh/myvenvs/director/lib/python3.6/site-packages/celery-4.4.0-py3.6.egg/celery/app/trace.py", line 650, in __protected_call__
	  return self.run(*args, **kwargs)
	  File "/Users/ovh/repos/workflows/tasks/etl.py", line 44, in load
	  print(1/0)
	ZeroDivisionError: division by zero

Clock	23

OpenSource

- https://github.com/ovh/celery-director

- pip install celery-director

THANKS!

Any questions?

You can find me at @ncrocfer & nicolas.crocfer@corp.ovh.com