

Należy utworzyć w swoim katalogu domowym plik 'workflow.json' o następującej zawartości:

```
{
  "name": "hello_task",
  "processes": [ {
    "name": "process",
    "type": "dataflow",
    "function": "amqpCommand",
    "config": {
      "executor": {
        "executable": "/bin/cat",
        "args": "input.txt > output.txt",
        "options": {
          "prefix": "gsiftp://zeus.cyfronet.pl/people/plglogin"
        }
      }
    },
    "ins": [ 0 ],
    "outs": [ 1 ]
  } ],
  "signals": [ {
    "name": "input.txt",
    "data": "1"
  }, {
    "name": "output.txt"
  } ],
  "ins": [ 0 ],
  "outs": [ 1 ]
}
```

lub pobrać gotowy plik:

[workflow.json](#)

Przygotowanie pliku wejściowego:

```
[plglogin@zeus ~]$ echo "input" > input.txt
```

Setup środowiska:

```
[plglogin@zeus ~]$ module load plgrid/tools/hflowclient

[plglogin@zeus ~]$ grid-proxy-init
(...)
[plglogin@zeus ~]$ hflowc setup
Appliance set hflowc created successfully!
wfMain created successfully!
waiting for wfMain to start.....wfMain is active!
Found port mapping 1032->443, hfmain endpoint: http://149.156.11.4:1032
starting wfworker_0
Workers created
```

Notujemy "hfmain endpoint", po zakończeniu wykonania polecenia należy odczekać jeszcze 1-2 minuty aby wszystkie usługi się podniosły. W międzyczasie możemy obserwować tworzenie się maszyn w interfejsie webowym cloudu, dostępnym pod adresem: <https://cloud.plgrid.pl> w zakładce "Workflows". Na tym etapie, w cloudzie powstało środowisko uruchomieniowe.

Uruchomienie workflowu:

```
[plglogin@zeus ~]$ hflowc runwf http://149.156.11.4:1032 workflow.json
workflow started: http://149.156.11.4:1032/apps/1
```

W tym momencie workflow zostanie przesłany do maszyny zawierającej silnik HyperFlow i uruchomiony. Wynikiem działania workflowu będą informacje znajdujące się w pliku wyjściowym.

Sprawdzenie wyników:

W następujący sposób sprawdzamy zawartość pliku wynikowego, który został stworzony na "workerze" w trakcie wykonania workflowu a następnie skopiowany jako "output" do katalogu zdefiniowanego w workflowie.

```
[plglogin@zeus ~]$ cat output.txt  
hello world
```



Usunięcie środowiska uruchomieniowego workflow:

```
[plgpawlik@zeus ~]$ hflowc teardown  
Removing appliance set: hfworkflow  
Done.
```

Zaawansowane użycie

Zaawansowane użycie sprowadza się do zmodyfikowania obrazów, tak aby zawierały narzędzia potrzebne do wykonania zadań z workflowu (konkretne aplikacje np. montage), zapisanie ich i modyfikacja konfiguracji klienta, tak aby nowo tworzone środowiska były budowane z uwzględnieniem własnych obrazów.

Nazapki

Podczas korzystania z klienta otrzymuję komunikat podobny do:

```
[Error: error getting appliance sets: 400, Bad Request]
```

Należy się upewnić, że korzystamy z proxy od SimpleCA stworzonego za pomocą komendy `grid-proxy-init` lub `voms-proxy-init -rfc` lub proxy pobranego z PLGridowego OpenId (format: RFC 3820 compliant impersonation proxy).

Gdzie szukać dalszych informacji?

Szczegółowa dokumentacja dla elementów systemu jest dostępna w następujących lokalizacjach:

- <https://github.com/dice-cyfronet/hyperflow> (opis silnika wykonawczego HyperFlow i instrukcja jak definiować workflow)
- <https://github.com/dice-cyfronet/hyperflow-client>
- <https://github.com/dice-cyfronet/hyperflow-amqp-executor>
- <http://www.rabbitmq.com/>