



Dziedzinowo zorientowane
usługi i zasoby infrastruktury
PL-Grid dla wspomagania
Polskiej Nauki w Europejskiej
Przestrzeni Badawczej



Dostęp do infrastruktury PL-Grid z
wykorzystaniem usług i narzędzi QosCosGrid
oraz gridu dziedzinowego Chamia Kwantowa

Bartosz Bosak
Piotr Kopta
Tomasz Piontek



Agenda



- Wstęp
- Specyfikacja zasobów
- Zadania interaktywne
- Zadania parametryczne
- Śledzenie postępu obliczeń
- Informacja o dostępnych zasobach
- Pytania, dyskusja





WSTĘP



Podręcznik użytkownika



<http://docs.plgrid.pl/qcg>



Wymagania



- Konto w infrastrukturze PL-Grid
- Certyfikat
- Globalny dostęp do usług QosCosGrid



Dostęp



- ssh plgID@
- Maszyny dostępne:
 - qcg.man.poznan.pl
 - moss.man.poznan.pl
 - ui.plgrid.wcss.wroc.pl
 - ui.cyfronet.pl
 - ui.grid.icm.edu.pl

Pierwsze uruchomienie:

creds keyfs



Najprostszy przykład



hostname





- Zlecenie zadania

```
$ qcg-sub plik
```

- Lista zadań

```
$ qcg-list [parametry]
```

- Informacje o zadaniu

```
$ qcg-info ID
```

- Zatrzymanie zadania

```
$ qcg-cancel ID
```

Podgląd wyjścia

```
$ qcg-peek ID
```





- Dyrektywy systemu QCG

```
#QCG {nazwa} [=parametry]
```

- Komentarze

```
#
```

- Polecenia powłoki *bash*

```
# przykładowy opis zadania
```

```
#QCG host=galera
```

```
#QCG stage-out-dir=.->result
```

```
hostname
```

```
pwd
```

```
date
```





SPECYFIKACJA ZASOBÓW



- Procesory, rdzenie

```
#QCG nodes={1węzłów}:{1rdzeni/węzeł}  
[:1wątków]
```

```
#QCG procs={1rdzeni} nie zalecane
```

- Pamięć

```
#QCG memory={mb}
```

```
#QCG memory-per-slot={mb}
```

- Klaster

```
#QCG host={reef,galera,inula,moss,nova,  
hydra,zeus}
```



- Parametry węzła

```
#QCG properties={parametry}
```

Wymagane oprogramowanie

```
#QCG module={nazwa modułu}
```

- Kolejka

```
#QCG queue={nazwa kolejki}
```

- Czas obliczeń

```
#QCG walltime=PT[nH][nM][nS]
```





PLIKI





```
#QCG output={plik-lokalny}
#QCG error={plik-lokalny}
#QCG input={plik-lokalny}
#QCG stage-in-file={plik-lokalny}[->plik-zdalny]
#QCG stage-in-dir={katalog-lokalny}[->katalog-zdalny]
#QCG stage-out-file={plik-zdalny}[->plik-lokalny]
#QCG stage-out-dir={katalog-zdalny}[->katalog-lokalny]
#QCG stage-out-dir=.->{katalog-lokalny}
```



Przykładowe zadanie



```
#QCG queue=plgrid-long
#QCG note=etanal Gaussian
#QCG output=${JOB_ID}.output
#QCG error=${JOB_ID}.error
#QCG stage-in-file=etanal.gjf -> etanal.gjf
#QCG stage-out-file=results.tar -> ${JOB_ID}.tar
#QCG nodes=1:12
#QCG host=galera.task.gda.pl
#QCG walltime=P7D
#QCG notify=mailto:jon@example.com
#QCG memory=15360
#QCG application=g09
#QCG argument=etanal.gjf
#QCG postprocess=tar cvf results.tar *
```





ZADANIA INTERAKTYWNE



Cel



- Uzyskanie zdalnej konsoli **na węźle obliczeniowym**
- Uzyskanie zdalnej konsoli na węźle na których uruchomiona jest nasza aplikacja
- Dostęp do wszystkich klastrów z jednego miejsca
- Kompilacja i testowanie oprogramowania
- Podgląd katalogu roboczego zadania



Polecenia



```
$ qcg-sub -I {opis}
```

```
$ qcg-connect {ID}
```



Przykład



```
#QCG host=inula  
/bin/bash
```





ZADANIA PARAMETRYCZNE



Cel



- Zdefiniowanie grupy zadań różniących się parametrami
- Parametrami mogą być:
 - zbiór wartości tekstowych
 - zbiór wartości numerycznych
 - zbiór plików
- W jednym zadaniu można zdefiniować wiele parametrów (iloczyn kartezjański)



Przykład 1



```
#QCG output=out.${var1}.${var2}
#QCG parameter-sweep=var1->list:one,two,three
#QCG parameter-sweep=var2->for:1..3

echo "${var1} x ${var2}"
```



Przykład 2



```
#QCG parameter-sweep=file->pattern:input/*.gjf
#QCG application=g09
#QCG argument=in.gjf
#QCG stage-in-file=${file}->in.gjf
#QCG output=out.${JOB_ID}
```





ŚLEDZENIE POSTĘPU OBLICZEŃ



notify



- Wiadomości email lub xmpp (jabber,gtalk) o starcie i zakończeniu zadania

#QCG notify=<mailto:pkopta@man.poznan.pl>

#QCG notify=xmpp:piotr.kopta@plgrid.pl



watch-output



- Cykliczne wysyłanie wiadomości email lub xmpp (jabber,gtalk) z:
 - ostatnim fragmentem pliku wyjściowego, lub
 - ostatnim fragmentem pliku wyjściowego pasującego do wzorca, lub
 - wyjściem z własnego skryptu uruchomionego w katalogu roboczym.



watch-output przykłady



- Ostatni fragment pliku wyjściowego

```
#QCG watch-output=mailto:pkopta@man.poznan.pl
```

- Ostatni fragment pliku wyjściowego zawierający wzorzec

```
#QCG watch-output=consumer->mailto:pkopta@man.poznan.pl,  
pattern->SCF Done:.,delay->30
```

- Wyjście z własnego skryptu

```
#QCG watch-output=consumer->mailto:pkopta@man.poznan.pl,  
script->parse.sh,delay->40  
#QCG stage-in-file=parse.sh
```



monitor



- Serwis wyświetlający aktualny stan obliczeń aplikacji
- Predefiniowane style zawierające tabele, wykresy, obrazy dla aplikacji **gaussian** oraz **piernik**
- Możliwość wyświetlania dowolnego tekstu z każdej aplikacji
- Prezentacja informacji o zasobach na których uruchomiona jest aplikacja

<https://qcg-monitoring.man.poznan.pl>



monitor przykład



#QCG application=g09

#QCG monitor





DOSTĘPNE ZASOBY OBLICZENIOWE



qcg-offer



- Wyświetla informacje o aktualnym stanie zasobów obliczeniowych dostępnych poprzez system QCG
 - liczba oraz stan węzłów
 - dostępna pamięć
 - liczba rdzeni
 - właściwości węzłów
- Dostępne na maszynie:
qcg.man.poznan.pl





- Podręcznik użytkownika

<http://docs.plgrid.pl/qcg>

▫ Email:

qcg@plgrid.pl

