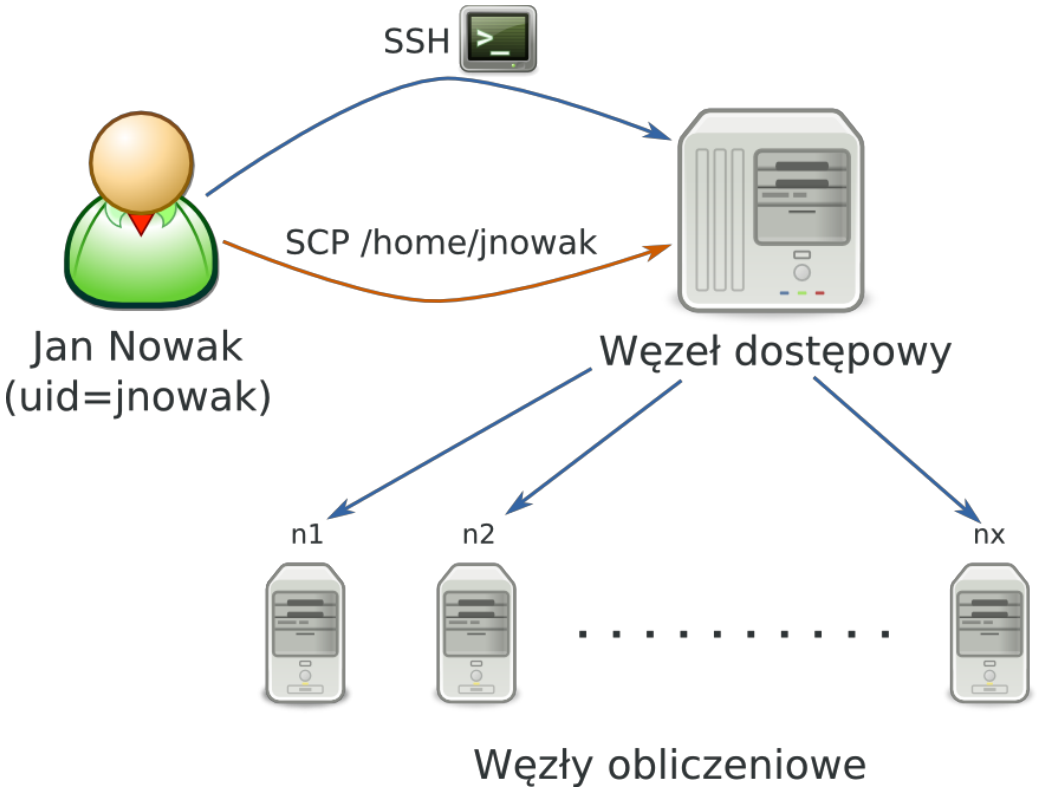
Skrócony Poradnik Użytkownika

Opracowano na podstawie zawartości bazy wiedzy Grida GRAFEN, dostępnej pod adresem http://info.grafen.ippt.gov.pl oraz materiałów dostarczonych przez firmę WASKO, wykonawcę Grida. Pod podanym adresem znaleźć można pełne wersje niniejszych skróconych informacji.

Wstęp

Z punktu widzenia użytkownika schemat systemu klastrowego można przedstawić następująco:



Komputery w klastrze zwane **węzłami** są dostępne dla użytkownika za pośrednictwem węzła dostępowego. **Węzeł dostępowy** jest „pomostem" pomiędzy użytkownikiem, a pozostałymi węzłami (i innymi zasobami) klastra. Logowanie oraz przesyłanie plików następuje wyłącznie poprzez węzeł dostępowy. Na następnych podstronach tej sekcji dowiesz się, jak za jego pośrednictwem wykorzystywać klaster. Będą to:

1. Zasady korzystania
2. Logowanie
3. Menadżer zadań
4. Monitorowanie Zadania
5. Pamięć masowa

Zasady korzystania z systemu klastrowego Kevlar

Wszyscy pracownicy naukowi IPPT PAN są zaproszeni do korzystania z systemu klastrowego. Dla węzłów obliczeniowych obowiązuje osobny system kont użytkowników, w sprawie założenia konta należy zwrócić się do p. Arkadiusza Grubby w pokoju 147, tel. wew. 351. Korzystając z zasobów obliczeniowych należy pamiętać o pewnych dodatkowych zasadach:

- nigdy nie logujemy się na węzły obliczeniowe i nie uruchamiamy na nich programów, jeśli nie zostały one przydzielone przez Menadżer Zadań,
- na węźle dostępowym nie wykonujemy żadnych programów konsumujących duże ilości zasobów; zadania wymagające kontaktu z użytkownikiem (np. kompilacje) uruchamiamy w trybie interaktywnym manedżera (`qsub -I`),
- pliki umieszczamy w odpowiednich dla nich miejscach - kody źródłowe w katalogu domowym, dane tymczasowe w `$TMPDIR`.

Przejdź na następną stronę poradnika, Logowanie.

Logowanie

Aby połączyć się z węzłem dostępowym (i korzystać z klastra obliczeniowego), użytkownik powinien posiadać na swojej stacji roboczej odpowiednie oprogramowanie:

1. **klient SSH** do połączenia z linią komend węzła dostępowego. Jeśli chcemy korzystać z aplikacji z graficznym interfejsem użytkownika, klient powinien mieć włączone przekierowanie protokołu X11 (*X11 forwarding*)
    1. w systemach uniksowych: OpenSSH (polecenie `ssh`),
    2. w systemach Windows: PuTTY lub openssh z pakietu Cygwin.
2. **klient SCP** do kopiowania plików poprzez protokół SSH
    1. w systemach uniksowych: OpenSSH (polecenie `scp`),
    2. w systemach Windows: WinSCP.
3. **serwer X** do wyświetlania graficznych interfejsów użytkownika
    1. w systemach uniksowych: X.Org, instalowany domyślnie w środowiskach graficznych,
    2. w systemach Windows: Xming.

W systemie klastrowym IPPT węzeł dostępowy znajduje się pod adresem **headnode.kevlar.ippt.gov.pl** (IP 10.100.255.73).

Logowanie na wszystkie kolejne węzły systemu następuje **bez hasła**.

> *UWAGA* Nie należy logować się ani ręcznie uruchamiać żadnych procesów na węzłach obliczeniowych, które nie zostały przydzielone przez menadżer zadań. Funkcję uruchamiania procesów udostępnia menadżer zadań opisany na następnej stronie. Ręczne uruchamianie zadań powoduje zakłócenia w działaniu procesów innych użytkowników.

Menadżer Zadań

Menadżer zadań to specjalne oprogramowanie, zainstalowane na klastrze. Zarządza on uruchamianiem zadań, realizując następujące usługi:

- Informuje użytkowników o dostępnych zasobach i stopniu ich wykorzystania,
- Przyjmuje zadania do wykonania i uruchamia w najwłaściwszym momencie,
- Uwzględnia priorytety zadań,
- Umożliwia monitorowanie przebiegu wykonywania zadania,
- Szereguje zadania w kolejkach.

Menadżer zadań stara się realizować te zadania w sposób optymalny, tj. tak by zasoby sprzętowe były wykorzystane w jak największym stopniu, a zadania kończyły się możliwie szybko. Niestety, aby to osiągnąć musi czasem odsunąć w czasie wykonanie zadania, dlatego nie zawsze uruchamia się ono w chwili wydania polecenia, jak na zwykłym komputerze osobistym. Dodatkowo, menadżer zadań może także uwzględniać pewne dodatkowe kryteria (np. by osiągnąć z góry założoną proporcję między zasobami przydzielonymi zadaniom z poszczególnych jednostek organizacyjnych, nagradzać zadania spełniające pewne szczególne wymagania szybkim wykonaniem itp.). Funkcję menadżera zadań w klastrze IPPT pełni TORQUE.

Wyświetlanie dostępnych zasobów

Aby sprawdzić rozmiar dostępnej pamięci i rdzeni obliczeniowych użytkownik może wykorzystać polecenie **`pbsnodes`**. Jest to polecenie menadżera zadań. Najbardziej popularne wywołania tego polecenia to:

- **`pbsnodes -a`** pokazuje dostępne zasoby wszystkich węzłów,
- **`pbsnodes <nazwa węzła>`** pokazuje dostępne zasoby wybranego węzła.

Przykładowy wynik tego polecenia:

```
n4
     state = free
     np = 12
     properties = kevlar,intel,mpi
     ntype = cluster
     status = rectime=1327145281,varattr=,jobs=,state=free,netload=804429755,gres=,loadave=0.00,ncpus=12,
        physmem=24685132kb,availmem=32322812kb,totmem=32685460kb,idletime=563910,nusers=0,nsessions=? 15201,
        sessions=? 15201,uname=Linux n4 2.6.18-238.19.1.el5 #1 SMP Fri Jul 15 00:48:58 EDT 2011 x86_64,opsys=linux
```

```
    gpus = 0
```

Oznacza on, że węzeł n4:

- jest włączony i gotowy do pracy (`state=free`),
- posiada 12 rdzeni obliczeniowych (`ncpus=12`),
- nie jest obciążony (`loadave=0.00`),
- posiada 24 GB pamięci RAM (`physmem=24685132kb`),
- nie posiada akceleratora GPU (`gpus=0`).

> Użytkownik może czasem napotkać węzły ze statusem **`offline`**. Oznacza to, że węzeł jest czasowo wyłączony z systemu obliczeń, np. z powodu interwencji serwisowej.

Uruchamianie zadania

Do uruchamiania zadań na węzłach służy polecenie **qsub**. Szczegółową dokumentację można znaleźć w instrukcji użytkownika, wyświetlanej poleceniem `man qsub` lub na stronie dokumentacji TORQUE. Podstawowy parametr (podawany na końcu) to skrypt do wykonania, najważniejsze opcje to:

- `-N <nazwa zadanie>` umożliwia podanie nazwy zadania, która będzie wykorzystywana do monitorowania go,
- `-l nodes=<liczba węzłów>:ppn=<liczba rdzeni>` rezerwuje dla zadania określoną liczbę węzłów obliczeniowych i określoną liczbę rdzeni na każdym z nich,
- `-l walltime=<godziny>:<minuty>:<sekundy>` rezerwuje określony czas obliczeń, po upłynięciu którego zadanie zostanie usunięte,
- `-I` wykonuje zadanie interaktywne - w chwili przydzielenia węzłów użytkownik zostaje przekierowany na konsolę pierwszego z nich i może wykonywać tam zadania ręcznie. W tym trybie powinno się wykonywać np. kompilacje czy pre- i postprocessing danych,
- `-X` uruchamie przekierowanie systemu graficznego X,
- `-l pmem=<ilość pamięci>B` rezerwuje określoną ilość pamięci na rdzeń.

Przykładowe wykonania:

`qsub -N nowezadanie skrypt.sh`	Dodaje zadanie polegające na wykonaniu skryptu `skrypt.sh` do kolejki obliczeń. Nazwa zadania to `nowezadanie`.
`qsub -l nodes=10:ppn=12 -l walltime=0:10:00 zadanie.sh`	Jak wyżej, tylko rezerwujemy do obliczeń 10 węzłów po 12 rdzeni, a zadanie potrwa maksymalnie 10 minut.
`qsub -I -X nodes=4`	Wykonuje zadanie interaktywne - użytkownik dostaje do dyspozycji konsolę na pierwszym węźle przydzielonym do zadania. Dodatkowo uruchomione jest przekierowanie X.
`qsub -l nodes=12:ppn=12 -l pmem=1GB`	Rezerwuje 12 węzłów, na każdym 12 rdzeni, na każdy rdzeń 1 GB pamięci fizycznej.

Zadania MPI

Gdy korzystamy z jednej z dostępnych bibliotek MPI (sposób wyboru przedstawiono na stronie Oprogramowanie), przy wywołaniu polecenia musimy przekazać dodatkowe informacje - ile egzemplarzy procesu chcemy uruchomić, oraz na jakich komputerach. W ogólności polecenie to wygląda następująco:

```
mpirun -np <liczba procesow> -f <plik z nazwami hostów> <nazwa programu>
```

Dla ułatwienia Torque udostępnia zmienne systemowe, ustawione na odpowiednie wartości na podstawie parametrów zadania, Dzięki temu prawie wszystkie wywołania programów MPI mogą wyglądać tak samo:

```
mpirun -np $PBS_NP -f $PBS_NODEFILE <nazwa programu>
```

Monitorowanie zadania

Stan wykonania

Aby sprawdzić, na jakim etapie znajduje się nasze zadanie, należy użyć polecenia **qstat**. Jest to polecenie menadżera zadań. Najbardziej popularne wywołania tego polecenia to:

- **`qstat`** pokazuje stan wszystkich zadań danego użytkownika w formie skróconej,
- **`qstat -f <identyfikator zadania>`** pokazuje szczegółowe informacje o wybranym zadaniu.

Przykładowy wynik pierwszego polecenia widzimy poniżej:

```
[tester@headnode ~]$ qstat
Job id                    Name             User            Time Use S Queue
------------------------- ---------------- --------------- -------- - -----
293.master1                listing          tester          00:00:00 C batch
294.master1                zadanie-wait1    tester          00:02:13 R batch
295.master1                zadanie-wait2    tester                 0 R batch
296.master1                zadanie4         tester                 0 R batch
297.master1                zadanie5         tester                 0 Q batch
```

Znaczenie kolumn tabeli jest następujące:

- `Job id` to unikalny identyfikator zadania, tworzony na podstawie numeru porządkowego i nazwy hosta, z którego wywołano zadanie,
- `Name` to nazwa nadane zadaniu przez użytkownika,
- `User` to nazwa użytkownika, który zlecił zadanie,
- `Time Use` to ilość wykorzystanego czasu procesora,
- `S` to stan zadania (Q-czeka w kolejce, R-jest uruchomione, C-zakończone),
- `Queue` to nazwa kolejki, do której trafiło zadanie.

Widzimy zatem, że powyższy wydruk informuje nas o następujących zadaniach - `listing`, które nie zajęło czasu procesora i zostało zakończone, `zadanie-wait1`, które zajęło 2 minuty i trwa jego wykonywanie, podobnie, jak zadań `zadanie-wait2` i `zadanie4`, natomiast `zadanie5` oczekuje na zwolnienie zasobów.

Usuwanie zadania

W każdej chwili możemy usunąć dowolne swoje zadanie - niezależnie od jego bieżącego stanu. Służy do tego polecenie **`qdel <identyfikator zadania>`**. W przypadku zadania oczekującego, zostanie ono usunięte z kolejki, zaś program trwający zostanie zatrzymany. W obu tych przypadkach zadanie przyjmie stan **`C`** - zakończone.

Dane wyjściowe

Zadanie może przekazywać rezultaty swego działania w różny sposób. Jeśli robi to poprzez pliki, to sytuacja jest prosta - katalogi domowe użytkowników na wszystkich węzłach są współdzielone, więc po zakończeniu działania programu można zajrzeć do katalogu roboczego i obejrzeć wyniki tak, jakby się to odbyło na standardowym komputerze PC. Specjalnego omówienia wymaga natomiast kwestia danych przekazywanych na standardowe wyjście i wyjście błędów. Dane te w normalnej sytuacji pokazywane są na ekranie konsoli, jednak w przypadku menedżera zadań program może trafić do wykonania, gdy użytkownika nie ma przy konsoli, np. w nocy.

Z tego powodu do każdego zadania tworzone są dodatkowe pliki, zawierające dane wyjściowe, w katalogu roboczym zadania o nazwach według następującego schematu:

- `<nazwa zadania>.o<numer zadania>` - standardowe wyjście programu,
- `<nazwa zadania>.e<numer zadania>` - standardowe wyjście błędów programu.

Pamięć masowa

Wszystkie węzły obliczeniowe podłączone są do współdzielonego systemu plików Lustre. Węzły nie posiadają lokalnych dysków, startują z zasobów Lustre, wobec czego posiadają jednolitą konfigurację (jeśli administrator nie zdecyduje inaczej, wszystkie węzły obliczeniowe startują z tego samego obrazu systemu operacyjnego). Z lustre zamontowane są również katalogi:

1. **`/home`** - katalogi domowe użytkowników. Tam powinny znaleźć się pliki źródłowe i wykonywalne, dane wejściowe i wyjściowe.

Na wszystkich węzłach klastra katalogi domowe są zsynchronizowane i widoczne w jednolity sposób.

2. `/scratch` - tymczasowa szybka przestrzeń obliczeniowa na dyskach SSD. W tej części brak jest podziału na użytkowników; aby uniknąć konfliktów należy umieszczać dane w indywidualnym katalogu tworzonym na potrzeby każdego zadania, jego nazwa znajduje się w zmiennej systemowej `$TMPDIR`. Po zakończeniu zadania dane te nie są już dostępne.
3. `/tmp` - katalog ramdysku (wirtualny system plików umieszczony w pamięci RAM węzła). **UWAGA!** Nie wykorzystywać bez wyraźnego polecenia administratora.