

# Управление заданиями в GRID.

*Н. Клопов  
ПИЯФ, Гатчина.*

Перед началом работы с LCG Вам необходимо создать так называемый проху-сертификат:

```
$ grid-proxy-init
Your identity: /C=RU/O=DataGrid/OU=pnpi.nw.ru/CN=Ivan Petrov
Enter GRID pass phrase for this identity: *****
Creating proxy ..... Done
Your proxy is valid until: Thu Sep  2 23:36:15 2004
```

По умолчанию время жизни такого сертификата равно 12 часам. Для получения информации о проху-сертификате можно воспользоваться следующей командой:

```
$ grid-proxy-info
subject   : /C=RU/O=DataGrid/OU=pnpi.nw.ru/CN=Ivan Petrov/CN=proxy
issuer    : /C=RU/O=DataGrid/OU=pnpi.nw.ru/CN=Ivan Petrov
identity  : /C=RU/O=DataGrid/OU=pnpi.nw.ru/CN=Ivan Petrov
type      : full legacy globus proxy
strength  : 512 bits
path      : /tmp/x509up_u500
timeleft  : 11:59:51
```



- JDL определяет характеристики задания, которые будут использоваться для выбора оптимального ресурса для его запуска.
  - Характеристики самого задания
  - Требования к компьютерным ресурсам (и к software)
  - Требования к входным и выходным данным
- JDL состоит из предложений типа: ***Attribute = value;*** где value может быть Boolean, Integer, String.

## JobType

*Normal, Interactive, MPICH, Checkpointable*

## Executable (mandatory)

Имя выполняемой программы

## Arguments (optional)

Параметры программы

## StdOutput, StdError (optional)

Стандартный input/output/error

## InputSandbox (optional)

Список файлов на UI, которые необходимы для выполнения задания.  
Эти файлы будут переданы на удаленный кластер.

## OutputSandbox (optional)

Список файлов, сгенерированных заданием, которые должны быть  
затребованы на UI

```
JobType = "Normal";  
Executable = "/bin/bash";  
StdOutput = "stdout";  
StdError = "stderr";  
InputSandbox={"yourscript.sh","inpfiler.txt"};  
OutputSandbox={"stderr","stdout","outf.txt"};  
Arguments = "yourscript.sh";
```



Enabling Grids for E-science

# Job Requirements

- Rank

- Как ранжировать ресурсы, которые удовлетворяют специфицированным требованиям
- Выбирается CE с наивысшим рангом, например:
  - Rank= - *other.GlueCEStateEstimatedResponseTime* (наименьшее время доступа)
  - Rank= *other.GlueCEStateFreeCPUs* (наибольшее количество CPUs)
  - Rank= (*other.GlueCEStateWaitingJobs == 0 ? other.GlueCEStateFreeCPUs : -other.GlueCEStateWaitingJobs*)



- Есть программа на PERL `tstp.pl`, которая использует входной файл с колонкой чисел. Каждое число возводится в квадрат и записывается в выходной файл.
- Имя входного файла передается как параметр программы. Имя выходного файла то же как и входного, но с расширением 'out'.

- **JDL файл:**

**VirtualOrganisation = "nw\_ru";**

**Executable = "pinp.pl";**

**Arguments = "pinp.inp";**

**StdOutput = "std.out";**

**StdError = "std.err";**

**InputSandbox={"pinp.pl","pinp.inp"};**

**OutputSandbox = {"std.out", "std.err","pinp.out"};**

**RetryCount = 3;**

- Есть файл с исходным текстом программы на языке C `ctst.c`, которая выводит строку “Hello world” на стандартный вывод.
- Есть `Makefile` для сборки этой программы.
- Необходимо обеспечить сборку и запуск программы на удаленном ресурсе.

- **JDL файл:**

```
JobType="Normal";
```

```
VirtualOrganisation = "nw_ru";
```

```
Executable = "startC.sh";
```

```
StdOutput = "ctst.out";
```

```
StdError = "ctst.err";
```

```
OutputSandbox = {"ctst.out","ctst.err"};
```

```
InputSandbox = {"startC.sh","ctst.c","Makefile"};
```

```
RetryCount=3;
```

- Стартовый скрипт startC.sh:

```
#!/bin/bash
```

```
make //сборка программы
```

```
chmod +x ctst //разрешаем запускать ее
```

```
./ctst //запускаем
```

```
exit 0
```

- **edg-job-submit ctst.jdl**
- **Selected Virtual Organisation name (from JDL): nw\_ru**
- **Connecting to host cluster.pnpi.nw.ru, port 7772**
- **Logging to host cluster.pnpi.nw.ru, port 9002**
- **\*\*\*\*\*  
\*\*\*\*\***
- **JOB SUBMIT OUTCOME**
- **The job has been successfully submitted to the Network Server.**
- **Use edg-job-status command to check job current status. Your job identifier (edg\_jobld) is:**
- **- <https://cluster.pnpi.nw.ru:9000/KII7Xj39Mfan3xXRgvYegA>**

- **edg-job-status**  
<https://cluster.pnpi.nw.ru:9000/I5jbSjf1QpHGPSak1VXM9w>
- \*\*\*\*\*
- **BOOKKEEPING INFORMATION:**
- **Status info for the Job :**  
<https://cluster.pnpi.nw.ru:9000/I5jbSjf1QpHGPSak1VXM9w>
- **Current Status:**     Running
- **Status Reason:**     Job successfully submitted to Globus
- **Destination:**       cluster.pnpi.nw.ru:2119/jobmanager-pbs-nw\_ru
- **reached on:**        Fri Jun 2 14:22:13 2006
- \*\*\*\*\*

- **edg-job-get-output --dir .  
<https://cluster.pnpi.nw.ru:9000/KII7Xj39Mfan3xXRgvYegA>**
- **Retrieving files from host: cluster.pnpi.nw.ru ( for  
<https://cluster.pnpi.nw.ru:9000/KII7Xj39Mfan3xXRgvYegA> )**
- **\*\*\*\*\***
- **JOB GET OUTPUT OUTCOME**
- **Output sandbox files for the job:**
- **- <https://cluster.pnpi.nw.ru:9000/KII7Xj39Mfan3xXRgvYegA>**
- **have been successfully retrieved and stored in the directory:**
- **[/home/knv/tut/knv\\_KII7Xj39Mfan3xXRgvYegA](/home/knv/tut/knv_KII7Xj39Mfan3xXRgvYegA)**























