



## UNICORE 6 in 30 minutes

UNICORE Team

April 2010, UNICORE version 6.3.0

## Contents

<b>1</b>	<b>Preface</b>	<b>1</b>
<b>2</b>	<b>Server Components</b>	<b>1</b>
2.1	Installation . . . . .	1
2.2	Additional configuration . . . . .	2
2.3	Start server components . . . . .	2
<b>3</b>	<b>Clients</b>	<b>2</b>
3.1	Rich Client . . . . .	2
3.1.1	Installation . . . . .	2
3.1.2	Usage . . . . .	3
3.2	Commandline Client (UCC) . . . . .	6
3.2.1	Installation . . . . .	6
3.2.2	Usage . . . . .	7
	Submit a job . . . . .	7
	Batch Mode . . . . .	8
	File Imports & Exports . . . . .	8
<b>4</b>	<b>Where to go from here</b>	<b>9</b>

## 1 Preface

This document aims to give a short, step-for-step introduction to using UNICORE. For more in-depths information, see the documentation at <http://www.unicore.eu/documentation>.

If you want to use UNICORE for testing purposes, there are three simple ways:

1. Install the UNICORE Core Server Bundle plus the clients on your local machine
2. Install only the clients and use the UNICORE Testgrid
3. Use the UNICORE LiveCD

Ad 1: You need Sun Java 1.5 or higher. Follow the installation instructions in [Server Components Section 2](#).

Ad 2: You need Sun Java 1.5 or higher. Follow the installation instructions in [Clients Section 3](#).

Ad 3: You need a i386-compatible computer that boots from CD or USB stick. You can download the LiveCD from <http://www.unicore.eu/download>. To create a bootable CD, download the iso image and burn it on CD. To create a bootable USB stick, download the tar file, extract it to your USB stick (min. 500Mb) and execute the `boot/bootinst.sh` or `boot\bootinst.bat`. Skip the installation and configuration parts in this document and start with Section .

## 2 Server Components

This section describes how to install the UNICORE server components locally on your machine. If you have access to an already installed UNICORE server (e.g. if you are using the testgrid or the LiveCD), you can skip this section.

### 2.1 Installation

Note that on Windows, you need to have administrator rights in order to install the server components. For other Windows related remarks, see [http://sourceforge.net/apps/mediawiki/unicore/index.php?title=Windows\\_Issues](http://sourceforge.net/apps/mediawiki/unicore/index.php?title=Windows_Issues).

Download the Core Server Bundle from <http://www.unicore.eu/download> and execute the installer by double-clicking on the jar-file or entering in the command line:

```
java -jar unicore-servers-6.3.0.jar
```

Leave everything at its default values, but check the installation directory and ports used. You should not have any services already using these ports, however, on most machines the default ports should be free. If you are sure about the installation directory and ports, finish the installation.

## 2.2 Additional configuration

If the user submits a job with a specific application (e.g. a PovRay job or Bash script job), the server has to know where it can find the executable of this application. The paths to the application executables are configured in the IDB (Incarnation Data Base). Open the IDB of your newly installed server components:

```
UNICORE_Servers/unicorex/conf/simpleidb
```

in a text editor and check the paths of the applications. This document covers how to submit Povray and Bash Script jobs. Note that on Windows, Bash is not available, so you might want to have another scripting language installed like Python or Perl.

## 2.3 Start server components

Now it's time to start the server components:

```
cd UNICORE_Servers
./start.sh
```

or on Windows:

```
cd UNICORE_Servers
start.bat
```

# 3 Clients

UNICORE offers a set of different clients, each of which serving a different purpose. Currently the following clients are available:

- UNICORE Commandline Client (UCC)
- Rich Client: A graphical client based on the Eclipse platform for simple as well as and complex tasks; easily extensible

## 3.1 Rich Client

### 3.1.1 Installation

This section describes how to install and configure the Rich Client. If you are using the LiveCD, you can skip this part and start with Section 3.1.2.

Download the Rich Client package suitable for your system from <http://www.unicore.eu/download> and unzip it. Change to the installation directory and execute the binary UNICORE\_Rich\_Client.

At first start, you will be asked for a keystore location.<sup>1</sup> The keystore will hold all the certificates you need to access the grid. Create a new keystore by entering a non-existing jks file, and choose a password for it. Next, you will see the welcome screen, click on the `First Steps` entry.

If you want to use the **Testgrid**, click on `UNICORE Testgrid` in the left column, and fill out the form. The needed certificates and the Registry URL will be automatically set up for you. Leave the welcome screen by clicking the arrow icon in the top right corner and continue with Section 3.1.2.

If you want to access your **local installation**, click on `Add Registry` in the right column. In the dialog, if you used the default values during the server installation, use the following URL:

```
https://localhost:8080/DEMO-SITE/services/Registry?res= ↵  
default_registry
```

Next, click on `Import Certificates` and choose the file `demosuer.jks` to be imported, the password is `the!user`. Leave the welcome screen by clicking the arrow icon in the top right corner and continue with Section 3.1.2.

### 3.1.2 Usage

The video [http://www.unicore.eu/documentation/tutorials/unicore6/files/urc\\_singlejob.htm](http://www.unicore.eu/documentation/tutorials/unicore6/files/urc_singlejob.htm) shows how to submit a single script job with the Rich Client. This section describes the necessary steps as seen in the video.

Go to the `Grid Browser` tab in the upper left panel, right-click on a site and select `create Job` from the pop-up menu. In the wizard that pops up, choose the script application and press `Finish`, see Figure 1.

---

<sup>1</sup>UNICORE uses X.509 certificates for user authentication. The certificates are maintained in the client's keystore, which should be password-protected.

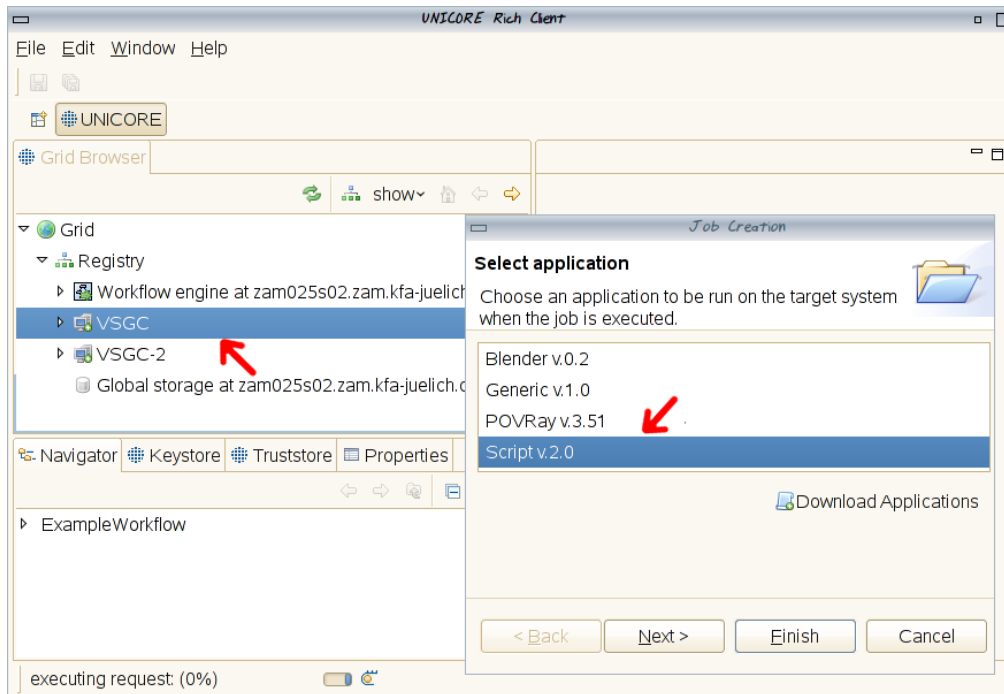


Figure 1: Creating a single job

On the left hand side, an editor for your script job will open. Enter a script and use the green button in the toolbar to submit the job, see the Figure 2. You don't need to explicitly save your job project or the script file.

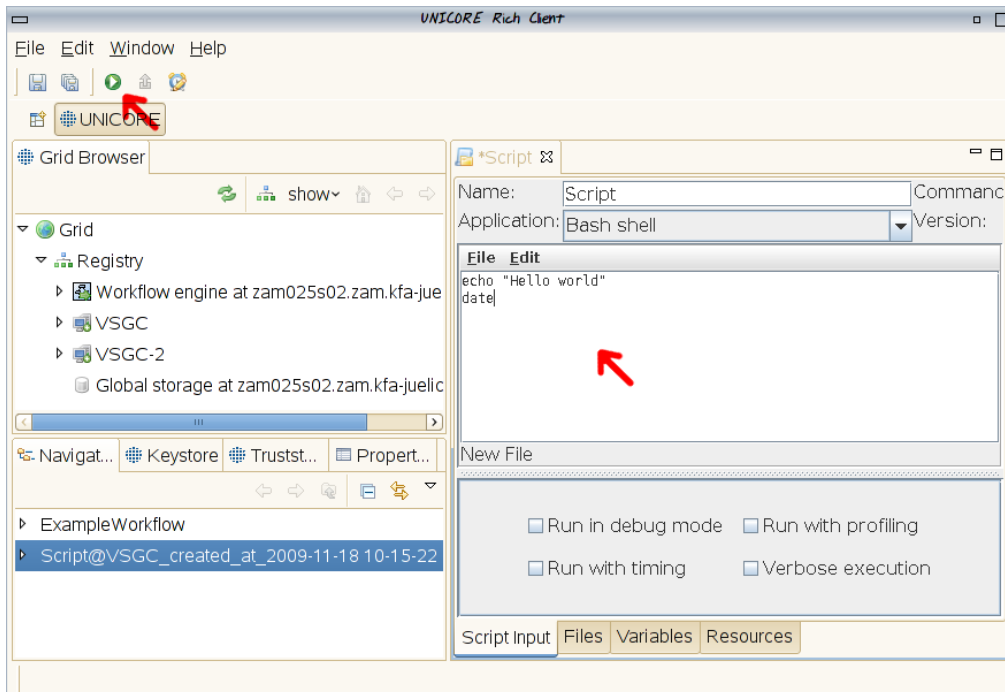


Figure 2: Submitting a single job

A new tab will be opened for monitoring the submitted job. If you want to re-edit or re-submit your job, use the first tab. The new tab also shows the status of the job's execution, as soon as it shows *finished*, you can fetch the output files using the button next to the submit button, see the Figure 3. A dialog will pop up and ask you which of the output files you want to be transferred to your local hard drive. The selected files will be displayed in new tab in the lower right panel once they are downloaded.

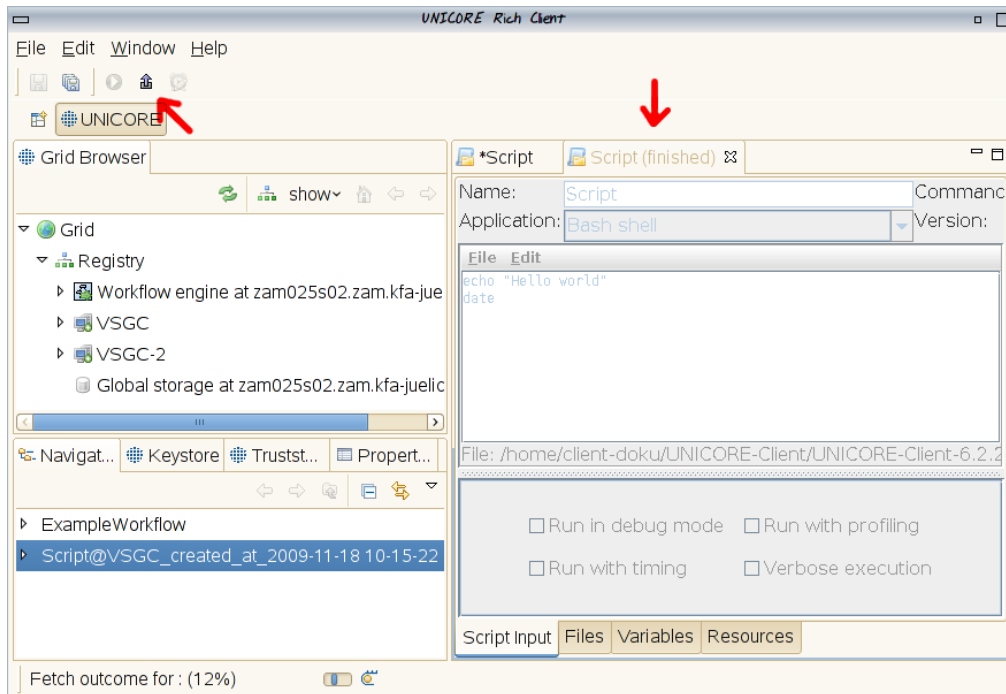


Figure 3: Fetching output

## 3.2 Commandline Client (UCC)

### 3.2.1 Installation

This section describes the installation of the UNICORE Commandline Client (UCC). If you are using the LiveCD, you can skip this part and start with Section 3.2.2.

Download the UCC zip archive from <http://www.unicore.eu/download> and unzip it. Open up a Shell/Command Line and change to the location of the ucc executable:

```
cd ucc-1.3.0/bin
```

If you want to use the **Testgrid**, execute:

```
ucc connect-to-testgrid
```

Fill out the form and continue with Section 3.2.2, but use the ucc command as given in the output of the `connect-to-testgrid` command.

If you want to access your **local installation**, on Linux create a directory `~/ .ucc` and a file `~/ .ucc/preferences` with the following content :

```
keystore=/path/to/serverinstallation/certs/demouser.jks
password=the!user
registry=https://localhost:8080/DEMO-SITE/services/Registry?res= ↵
default_registry
```

On Windows, create the directory `C:\Documents and Settings\\.ucc`. It might be necessary to create the directory from the commandline using `mkdir`. Give path names with double backslashes, so that on Windows the file `.ucc/preferences` looks like:

```
keystore=C:\\path\\to\\serverinstallation\\certs\\demouser.jks
password=the!user
registry=https://localhost:8080/DEMO-SITE/services/Registry?res= ↵
default_registry
```

Note that you don't need to enter the password in your configuration file; if you don't, UCC will ask for it on the command line.

### 3.2.2 Usage

Open up a Shell/Command Line and change to the location of the `ucc` executable:

```
cd ucc-1.3.0/bin
```

To test your UCC installation, invoke some help commands:

```
ucc help
ucc <command> -h
```

UCC also has an own shell in which you can issue UCC commands directly:

```
ucc shell
> help
```

This is helpful to avoid typing in your password over and over again.

### Submit a job

First, you have to connect to the UNICORE registry entered in the configuration file:

```
ucc connect
```

You should get a message saying that you can access (at least) one target system.

To get a list of all available UNICORE sites, enter

```
ucc list-sites
```

UCC takes job descriptions in the form of JSON-based text files. You can find examples for simple jobs in the directory `ucc-1.3.0/samples`. You can run such a job file using the command

```
ucc run -v -s DEMO-SITE ../samples/date.u
```

The `-v` option (v for verbose) produces more output than usual and can be omitted, and the `-s` option specifies the site to which the job will be submitted. If you don't specify a site, UCC will submit to an arbitrary site. The job outcome (in this case standard error and standard output) is automatically fetched from the site and stored in your local working directory. If you use the `-a` option (asynchronous), UCC will not wait for the job to finish.

You can get a list of all submitted jobs with

```
ucc list-jobs
```

With the commands

```
ucc get-status <job-id>
ucc get-output <job-id>
```

you can get a job's status and outcome.

### Batch Mode

If you want to submit a large amount of jobs, you can use UCC's batch mode. Create a directory `jobs` and put several copies of the file `ucc-1.2.2/samples/date.u` therein. When you enter the command

```
ucc batch -i jobs/
```

UCC will submit all the job scripts in the given directory. Each script will be deleted after submission. Try submitting dozens of scripts and note how fast UCC processes them!

Normally, UCC exits as soon as all jobs in the given directory are submitted. You can tell UCC to wait for more jobs instead:

```
ucc-1.3.0/bin/ucc batch -f -i jobs/
```

### File Imports & Exports

Here you see a sample job that handles file import and export:

```
{
  Executable: "/bin/bash",
  Arguments: [ "myscript.sh", ],
```

```
Imports: [
    { From: "scripts/myscript.sh", To: myscript.sh },
    { From: input.txt, To: input.txt },
],

Exports: [
    { From: output.txt , To: output.txt },
],
}
```

On Windows, use double backslashed for paths, e.g. `scripts\\myscript.sh`.

## 4 Where to go from here

This document just gives a very brief overview. One main aspect not discussed here is UNICORE's workflow capabilities. On the testgrid and on the LiveCD, the workflow services are already installed, but on your local installation you would need to install them separately.

For additional information about the individual clients or the UNICORE servers, see <http://www.unicore.eu/documentation/manuals/unicore6/>.