Summer School Parallel Programming
From Multi-core to Many-core and beyond

Introduction to the Computing Infrastructure

Roy Bakker
R.Bakker@uva.nl

Roeland Douma
R.J.Douma@uva.nl

Raphael 'kena' Poss
R.C.Poss@uva.nl

July 9, 2012

UNIVERSITY OF AMSTERDAM

1 Introduction

Welcome to the summer school parallel programming *From Multi-core to Many-core and beyond*. During this summer school you can use the Windows desktops present in this room. Accounts will be provided. We only use the machine to access the internet. The actual computation and development will be done on the DAS-4 cluster[1].

DAS-4 (The Distributed ASCI Supercomputer 4) is a six-cluster wide-area distributed system designed by the Advanced School for Computing and Imaging (ASCI). DAS-4 is funded by NWO/NCF (the Netherlands Organization for Scientific Research), and the participating universities and organizations. Accounts for the DAS-4 are not the same as you use for the workstations at the UvA, and will also be provided.

During the summer school, we will use the DAS-4 cluster at the VU. This cluster has one head node and 74 worker nodes. The standard compute node type of DAS-4 has a dual-quad-core 2.4 GHz CPU configuration and 24GB memory. In addition, several DAS-4 nodes include additional hardware for specific research purposes. For example the VU cluster has 16 nodes with an NVidia GTX480 GPU, and two nodes with an Nvidia C2050 Tesla GPU.

Each node has two quad-core CPUs, so it is actually an eight-core machine. Remember this for all of the practical sessions.

During all the labs: *If you need help or if anything is not clear, do not hesitate and ask the lab assistants early.* We are there to help you.

All documents you need (slides, assignments, etc.) that we make public can be found at:
`http://www.science.uva.nl/~bakkerr/summerschool/`

2 Getting started with the environment

At the end of this activity, you should be able to:

- log in remotely to the DAS-4 cluster,
- start remote X applications,
- write and compile a simple C program in the Unix environment,
- run the program on a cluster node.

However we asume that you are confident in using a command-line environment and secure shell (**ssh**), this introduction is quite detailed in how to deal with this. To edit files you can use a command-line editor like **vi** or **nano**. If you are not confident with using a command-line editor, you can also use a (remote) graphical editor such as **gedit**.

2.1 Start the X server

To be able to run any graphical user interface on the DAS-4 cluster, you need an **X** server running on your windows desktop. A shortcut (**Xming-Xserver-multiwindow**) to start the **Xming** X server is located in:

[1] http://www.cs.vu.nl/das4/

```
\\gene\bakkerr\summerschool\
```

You might want to create a shortcut to this location on your desktop. Start **Xming**, dismiss the warning (if any), and notice the **Xming** logo in the windows system tray (the X).

2.2 Start a shell on the DAS-4

To get access to the cluster (or any remote system) we need a secure shell (**ssh**) client. There are several clients available, including **PuTTY**. **PuTTY** is available in the same directory as **Xming**. Double-clicking on the shortcut **DAS4-VU-login** will automatically start an **ssh** session to the head-node of the DAS-4 cluster at the VU (`fs0.das4.cs.vu.nl`) with X forwarding enabled. It should now ask you for your username and password. After successful login, you should see a prompt like this:

```
[bakkerr@fs0 ~]$
```

Test the X settings by typing `xclock`. If everything went well, you will see a graphical clock displaying the time of the DAS-4 head node.

3 Running a program on the DAS-4 cluster

Programs are started on the DAS-4 compute nodes using the Sun Grid Engine (SGE) batch queuing system. The SGE system reserves the requested number of nodes for the duration of a program run. The default run time for SGE jobs on DAS-4 is 15 minutes, which is also the maximum for jobs on DAS-4 during working hours.

3.1 Hello World

In this section you will edit, compile and run a sample C (Hello World) program.

1. Start a shell on the DAS-4
2. Create a directory (`mkdir helloworld`)
3. Enter that directory (`cd helloworld`)
4. Start an editor and create / open a file (`gedit helloworld.c`)
5. Write your program:

```
#include <stdio.h>

int main(){
  printf("Hello World!\n");

  return 0;
}
```

6. Compile your program (`gcc -o hello helloworld.c`)

At this point you should have a working binary, but it is forbidden to run a program on the head node. We may only run the program on the worker nodes. Running a program on the worker nodes can be done using a **SGE job script** file, or using **prun**. For a simple program like this we use **prun**.

1. Load the module prun (`module load prun`)
2. Now we should think of the number of nodes that we would like to use. For a simple program like this, we normally execute it only once.
3. Issue the **prun** command. (`prun -v -np 1 hello`)

   `-v` means verbose, we would like to have lots of information (in case the run fails).

   `-np` specifies the number of nodes, first run with only one. Note that it reserves complete *nodes* for you, not that number of *cores*, since each node is an eight-core CPU.

   `hello` The name of the executable, followed by the arguments it takes (none in this case).
4. You should now have output similar like this:

```
[bakkerr@fs0 ~]$ mkdir helloworld
[bakkerr@fs0 ~]$ cd helloworld
[bakkerr@fs0 helloworld]$ gedit helloworld.c
[bakkerr@fs0 helloworld]$ gcc -o hello helloworld.c
[bakkerr@fs0 helloworld]$ module load prun
[bakkerr@fs0 helloworld]$ prun -v -np 1 hello
Reservation number 318849: Reserved 1 hosts for 900 seconds
Run on 1 hosts for 960 seconds from Thu Jun 30 10:25:36 2011
: node042/0
Hello World!
[bakkerr@fs0 helloworld]$
```

5. See what happens when we use more nodes (`prun -v -np 4 hello`) [2].

```
[bakkerr@fs0 helloworld]$ prun -v -np 4 hello
Reservation number 318850: Reserved 4 hosts for 900 seconds
Run on 4 hosts for 960 seconds from Thu Jun 30 10:26:59 2011
: node031/0 node007/0 node002/0 node027/0
Hello World!
Hello World!
Hello World!
Hello World!
[bakkerr@fs0 helloworld]$
```

4 Working from home

Unfortunately it is not possible to work on the DAS-4 cluster from home as firewall restrictions only allow login from a range of trusted IP addresses, unless

[2] Do not allocate too many nodes, you have to share the cluster with all of your fellow students and possibly other users.

you are a student at UvA (i.e. If you have a *UvAnetID*). In this case you can access `fs0.das4.cs.vu.nl` when you are connected through `UvAvpn`.

The assignment framework we provide on day 1 can be used at home. In this case, use `make runlocal` and `make mpilocal`. For MPI you have to install least the (Ubuntu) packages `libopenmpi-dev`, and `openmpi-bin`.