



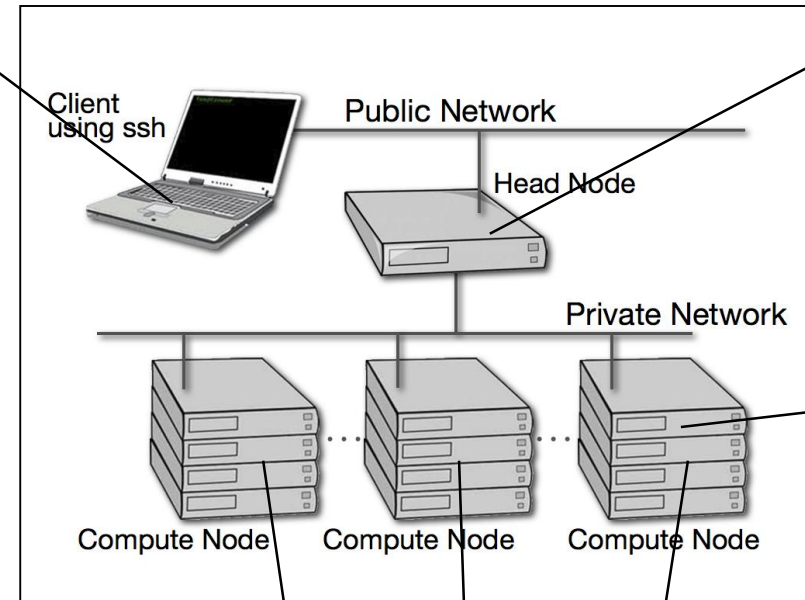
arm

Arm cross- platform tools Hands-on

VI-HPS platform
October 16, 2018

Different ways to run Arm Forge...

Here
(remote launch +
reverse connect)



There
(interactive mode +
reverse connect)

There
(offline OR
interactive mode)

Ultimately, that's where the tools will run.
But what about the GUI?

Download Forge “Client” for Reverse - Connect

18.1.1 :

http://content.allinea.com/downloads/arm-forge-18.1.1-Redhat-7.0-x86_64.tar

http://content.allinea.com/downloads/arm-forge-18.1.1-Suse-12-x86_64.tar

http://content.allinea.com/downloads/arm-forge-18.1.1-Ubuntu-16.04-x86_64.tar

18.2.2 :

http://content.allinea.com/downloads/arm-forge-18.2.2-Redhat-7.0-x86_64.tar

http://content.allinea.com/downloads/arm-forge-18.2.2-Suse-12-x86_64.tar

http://content.allinea.com/downloads/arm-forge-18.2.2-Ubuntu-16.04-x86_64.tar

19.0 :

http://content.allinea.com/downloads/arm-forge-19.0-preview2-Redhat-7.0-x86_64.tar

http://content.allinea.com/downloads/arm-forge-19.0-preview2-Suse-12-x86_64.tar

http://content.allinea.com/downloads/arm-forge-19.0-preview2-Ubuntu-16.04-x86_64.tar

Download Mac OS/X Client

18.1.1 :

http://content.allinea.com/downloads/arm-forge-client-18.1.1-MacOSX-10.7.5-x86_64.dmg

18.2.2 :

http://content.allinea.com/downloads/arm-forge-client-18.2.2-MacOSX-10.7.5-x86_64.dmg

19.0 :

http://content.allinea.com/downloads/arm-forge-client-19.0-preview2-MacOSX-10.7.5-x86_64.dmg

Install Forge “Client” for Reverse - Connect

```
wget http://content.allinea.com/downloads/arm-forge-18.1.1-Redhat-7.0-x86_64.tar
```

```
tar -xvf arm-forge-18.1.1-Redhat-7.0-x86_64.tar
```

```
cd arm-forge-18.1.1-Redhat-7.0-x86_64
```

```
./text-install
```

[Accept the license and specify the path of the Install]

```
export PATH=PATH:$<path_of_installation>/bin
```

Configure remote client

- Open your Remote Client
- Create a new connection: Remote Launch → Configure → Add
 - Hostname: user@romeologin1.univ-reims.fr
 - Remote installation directory: `/apps/arm_forge/18.2.3/`

Connect!

Temporary Specific Configuration (TBC)

Define PATH to MAP

- **module load openmpi/2.0.4.1.1_icc_mt**
- **module load intel/2018.3**
- **export PATH=\$PATH:/apps/arm_forge/18.2.3/bin/**

Compiler MPI Wrapper

- **make-profiler-libraries**

Complete slurm script

- **export PATH=\$PATH:/apps/arm_forge/18.2.3/bin/**
- **export ALLINEA_MPI_WRAPPER=/scratch_p/xxxx/yyyy/libmap-sampler-pmpi.so**

Arm MAP cheat sheet

Prepare the code

- `$ <TODO mpicc> -O3 -g myapp.c -o myapp.exe`

Edit the job script to run Arm MAP in “profile” mode

- `$ map --profile <TODO mpirun> ./myapp.exe arg1 arg2`

Open the results

- On the login node:
 - `$ map myapp_Xp_Yn_YYYY-MM-DD_HH-MM.map`
 - (or load the corresponding file using the remote client connected to the remote system or locally)

Arm MAP cheat sheet (reverse connect)

Prepare the code

- `$ <TODO mpicc> -O3 -g myapp.c -o myapp.exe`

Edit the job script to run Arm MAP in “profile” mode + Launch MAP on the remote machine

- Launch MAP GUI on remote machine
- `$map -connect <TODO mpirun> ./myapp.exe arg1 arg2`

Arm Performance Reports Cheat Sheet

Edit the job script to prefix the mpirun command

- **perf-report** <TODO mpirun> -n 8 ./myapp.exe

Submit the job

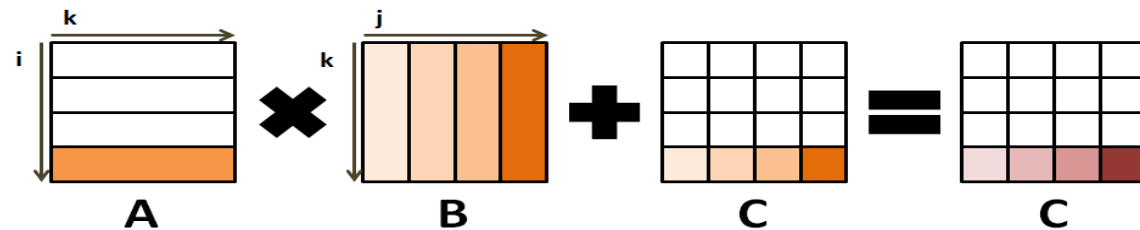
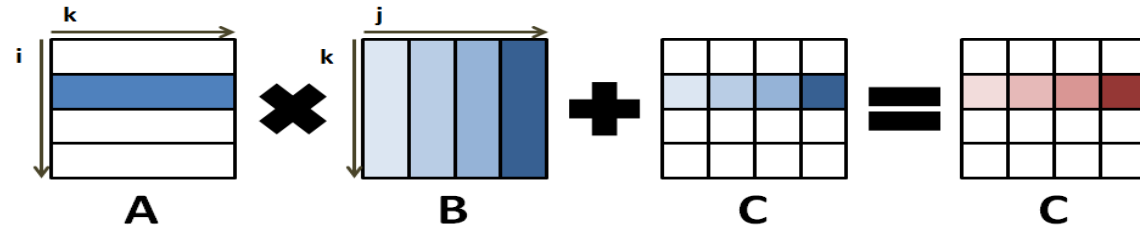
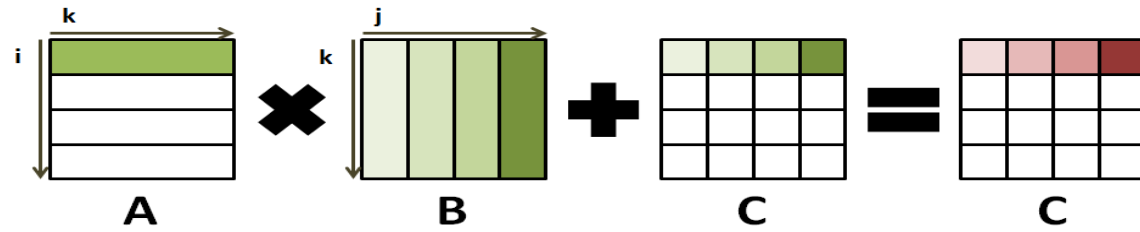
- \$ <TODO qsub> job.sub

Analyse the results

- \$ firefox myapp_8p_1n_YYYY-MM-DD_HH:MM.html

Matrix-multiply example

$$C = A \times B + C$$



Matrix-multiply example

Step1 - RUN

1. Load relevant modules (MPI and compilers)
2. Compile the program
3. Submit with Slurm

Step2 – PROFILE

- Check that `-g` flag is used in the Makefile
- Generate map profiles

Thank You!

Danke!

Merci!

谢谢!

ありがとう!

Gracias!

Kiitos!

arm