P. Koumoutsakos
ETH Zentrum, CLT E 13
CH-8092 Zürich

Tutorial on UPCXX

This tutorial contains instructions for the installation and setupf of UPC++.

Login Firstly log in to your euler account. Open a terminal and type

```
$ ssh <user_name>@euler.ethz.ch
```

Then type your password to log-in.

Download After logging in to your euler account you need to download the following version of UPCXX with the wget command:

```
$ wget https://bitbucket.org/berkeleylab/upcxx/downloads/upcxx
    -2018.9.0.tar.gz
```

Decompress the files with

```
$ tar xvzf upcxx-2018.9.0.tar.gz
```

and delete the compressed file with

```
$ rm upcxx-2018.9.0.tar.gz
```

Now get in the directory with

```
$ cd upcxx-2018.9.0
```

Installation Before installing the package, we need to make sure that we use the correct compiler. In order to make sure that UPCXX will work for your homework submissions load the following modules:

```
$ module load new gcc/6.3.0
$ module load intel/2018.1
```

Next, we are ready to install our package with

```
$ ./install $HOME/usr/upcxx
```

After running this command (and waiting some time) you should get the message

```
UPC++ successfully installed
```

If you encounter the following error

```
  File "/cluster/home/<user_name>/upcxx-2018.9.0/nobs/tool.py", line 15
    from nobs.async import shutdown
                    ^
SyntaxError: invalid syntax
GASNET must point to local tarball file, tarball url, or gasnet source
    tree.
```

It is because the installer is incompatible with the python version you are currently using. Try unloading any python modules you have loaded and use the system default, which you can check with the following two commands

```
$ which python
    /usr/bin/python
$ python --version
    Python 2.7.5
```

The package is then installed in the directory `$HOME/usr/upcxx`, passed as an argument to the `./install` command.

In order to compile an app you may use the UPCXX compiler wrapper. Run the following command to make the binaries of UPCXX visible to your path:

```
$ export PATH="$HOME/usr/upcxx/bin/:$PATH"
```

If you want to make this configuration "permanent" you can add the line above to your `$HOME/.bashrc` file (edit it with any text editor).

Helloworld.cpp program Next, we will write a small UPCXX program and try to compile and run it. Write the following small program in a file called `helloworld.cpp`:

```cpp
#include <iostream>
// Loading the UPCXX header file
#include <upcxx/upcxx.hpp>

int main(int argc, const char **argv) {

    // Initializing UPCXX
    upcxx::init();

    // Printing the rank number and the total number of ranks
    std::cout << "Hello UPC++ World! I'm rank " << upcxx::rank_me() << "
         of total " << upcxx::rank_n() << "." << std::endl;

    // Finalizing/ shutting down UPC++
    upcxx::finalize();

    return 0;
}
```

Of course you could spare the addition of `upcxx::` in the beginning of the methods by including a namespace in the beginning, i.e. `using namespace upcxx;`, and plainly calling `rank_me()` and similar when needed.

UPC++ Backends UPC++ provides various backends that allow the user to specify the configuration of parallel communication. The backends are characterized by three dimensions: conduit, thread-mode, and code-mode.

1. The variable `UPCXX_GASNET_CONDUIT` controls the GASNet conduit to use. The default depends on the architecture.
   - `smp` is the typical high-performance choice for single-node multi-core runs.
   - `udp` is a useful low-performance alternative for testing and debugging.
   - `aries` is the high-performance Cray XC network.
   - `ibv` is the high-performance InfiniBand network.
2. The variable `UPCXX_THREADMODE` can take either the value `seq` or `par`. The value `seq` limits the application to only calling "communicating" UPC++ routines from the thread that invoked upcxx::init, and only while that thread is holding the master persona. The benefit being that `seq` can be synchronization free in much of its internals. A thread-mode value of `par` allows any thread in the process to issue communication as allowed by the specification, allowing for greater injection concurrency from a multi-threaded application but at the expensive of greater internal synchronization (higher overheads per operation). The default value is always `seq`.
3. The variable `UPCXX_CODEMODE` allows the user to select between highly optimized (`O3`) and highly debuggable code (`debug`). The default value is `O3`.

For the purpose of this tutorial, pick the following options:

```
$ export UPCXX_GASNET_CONDUIT=smp
$ export UPCXX_THREADMODE=seq
$ export UPCXX_CODEMODE=O3
```

Compile and link Now that our first helloworld program is set up, we can compile it. Compile and link the program using the following commands:

```
$ upcxx -O -c helloworld.cpp -o helloworld.o
$ upcxx -O -o helloworld helloworld.o -lm
```

These commands will create the executable `helloworld`.

Launching Now, we can employ the `upcxx-run` launcher provided in the installation to run our program. Firstly, locate the launcher. If you followed the steps of this tutorial the launcher should be `/usr/upcxx/bin/upcxx-run`. We can run our program with one processor as

```
$ upcxx-run -n 1 ./helloworld
```

Please, mind the ./ in the begging of the file. If you forget it the program will return an error as it is not searching the current folder for the executable.

Requesting interactive node Now, in order to test if our program can indeed be run in multiple nodes, we can request an interactive bash script (for one hour) with:

```
$ bsub -W 01:00 -n 4 -Is bash
```

Launching After getting the requested node allocation, try to recompile-link and run

```
$ upcxx-run -n 4 ./helloworld
```

Again, do not forget to include ./ when passing the argument ./helloworld. The output of your program should be something like:

```
Hello UPC++ World.
Hello world! I'm rank 0 of 4.
Hello world! I'm rank 3 of 4.
Hello world! I'm rank 2 of 4.
Hello world! I'm rank 1 of 4.
```

Important For more information about the installation process please refer to the website `https://bitbucket.org/berkeleylab/upcxx`.