Tracer – a short user manual (steps to follow)

1. Go to the directory where parallel application resides.

2. Define the ideal memory simulator (it will be used to produce the **ideal trace** file). This means that you should edit *makefile* to insert:

```
SIM = ideal
SIMHOME = ideal
```

or you can just comment out the exisiting memory simulator parameters. Save *makefile* in the end.

3. Compile the application using the ideal memory simulator. This means that you should type, (in this order):

```
make totalclean
make dep
make
```

4. Now you should make a trace file. You do that by starting a simulation with one of the options:

```
-t             - send the simulation output (trace) to stdout
-tx filename   - the output (trace) is in text format, it will be saved in filename
-tb filename   - the output (trace) is in binary (condensed) format, it will be saved in filename
```

Start the simulation using the desired parameters for a given multiprocessor model. For example:

```
APPLICATION parameters -- -tx TRACE.txt -dcache_size=32
-dcache_line_size=16 -dcache_way=4
```

will start the simulation (cache parameters are given directly in the command line, thus overriding the ones in the limes.ini file) and produce the trace file called TRACE.txt. After the simulation is finished, an ideal trace is obtained ready to be used as input for trace-driven simulation with Tracer.

5. Move to the *tracer* directory. Edit *makefile* to define the multiprocessor memory model along with the name of the target simulator. For example, inserting:

```
SIM = MESI
SIMHOME = snoopy/MESI
TARGET = $(SIM)tracer
```

in *makefile* will enable us to get a simulator of the SMP system with MESI cache coherence protocol called MESItracer. Do not forget to save *makefile*.

6. Type (in this order):

```
make clean
make
```

to obtain a compiled (runnable) simulator called MESItracer.

7. Start the simulation with any of the desired options for the multiprocessor model you want to simulate, and using trace file options as explained under **4**. For example, typing:

```
MESItracer  -fx /usr/local/limes-v1.1/splash2/fft/TRACE.txt  -tx MESITRACE.txt
-dcache_size=32  -dcache_line_size=16  -dcache_way=4
```

will produce a text trace file called MESITRACE.txt in a local directory based on an ideal trace file called TRACE.txt which resides in the fft directory.