

# Project Seminar: Parallel and Distributed Systems

---

*Assignment 5 (Submission deadline: Feb 3rd 2016, 13:30 CET)*

## General Rules

The assignment solutions have to be submitted at:

<https://www.dcl.hpi.uni-potsdam.de/submit/>

Our automated submission system is intended to give you feedback about the validity of your file upload. A submission is considered as accepted if the following rules are fulfilled:

- You did not miss the deadline.
- Your file upload can be decompressed with a zip / tar decompression tool.
- Your submitted solution contains only the source code files and a Makefile. Please leave out any Git / Mercurial repository clones or SVN / CVS meta-information.
- Your solution can be compiled using the “make” command, without entering a separate sub-directory after decompression.
- Your program runs without expecting any kind of keyboard input or GUI interaction.
- Our assignment-specific validation script accepts your program output / generated files.

If something is wrong with your submission, you will be informed via email (console output, error code). Re-uploads of corrected solutions are possible until the deadline.

**All tasks must be submitted accordingly in order to pass the assignment.**

## Assignment 5

The fifth assignment covers monitoring and profiling the runtime behaviour of parallel applications.

### Task 5.1 Linux Perf

**PRO TIP: compiler your application with `-fno-omit-frame-pointer` . Otherwise perf is not able to get proper stacktraces.**

Make yourself familiar with the concepts and usage of Perf. As a starting point you can have a look at <http://www.brendangregg.com/linuxperf.html> . Use one of your previous submitted tasks as your application to profile. You can use your own machines, but please also profile your application on our test-system:

```
pvprog-perf (172.16.64.180)
```

Optional you can profile your application on one of our ARM-based systems, but keep in mind that you can only count the performance counters for the whole runtime of the application (no perf record/report). Nevertheless this could be interesting to see, if your application also behave in a similar fashion on ARM.

odroid02 (172.16.64.152)

You should receive username/password via Mail.

**UPDATE: We managed to get perf with all features running on the Parallella-Boards (2-Core ARM)**

parallella03 (172.16.164.143)

parallella04 (172.16.164.144)

After you profiled your application, please answer the following questions:

1. Which counters did you use?
2. What did you observe, where are the hotspots? (3 - 5 sentences)
3. What did you change? (Code Snippets before and after the profiling)

Also please produce an image showing the runtime characteristics of your profiled application using `perf timechart` .

Please submit a single zipped file containing the resulting image file and a textfile with the answers to the questions.