Markus Schneider-Pargmann

mail@schneider-pargmann.com
C++ Software Engineer — Linux Expert — Berlin/Germany

Skills

C++: I have a good understanding of C++11 to which I am currently limited due to compatibility. Smart pointers, lambdas and templates are the best things available in C++11, I am curious to use C++20 ranges. I think C++ should only be used with static analyzers.

C: During Linux Kernel development I gathered a good understanding of C. I love the Kernel-specific datastructures like lists and the easy and explicit way of object orientation.

Python: My primary scripting language for whenever I want to automate something that is bothering me, which happens quite often. I especially appreciate the functional aspects of Python like list comprehensions.

Linux: I am a passionate Linux user at the job and in private for my server and workstation. I love that it is open source, flexible, and everything can be easily automated.

Other: Bash, Git, gdb, CMake, Make, Jenkins, Prometheus, Grafana, Linux Kernel, PTXdist, Oscilloscope/Logic Analyzer, Digital Electronics, Ansible, German, English

Experience

Quobyte GmbH, Berlin — **C++ Software Engineer** — Sep. 2016 – Present

Quobyte is a software defined storage in form of a distributed file system providing fault tolerance, fast access and different interfaces (Linux using **FUSE**, Windows, Mac OS, S3, NFS, Kubernetes, OpenStack).

- I increased throughput of the **C++** client (used by all interfaces) for networks like 100Gb InfiniBand.
  - Fixed a lot of inefficient code after analyzing with perf, Brendan Gregg's Flame Graphs, etc.
  - Detected heavy lock contention and fixed it by implementing concurrent data structures.
  - Redesigned the non-scaling single-threaded network layer with a multi-threaded implementation using **Boost.Asio** capable of higher throughput.
  - The previous prefetcher was not able to saturate the disks. I implemented an advanced and configurable prefetcher which scales much better. It is able to saturate disks and network links.
- Created automated performance regression tests using **Ansible** and **Python** detecting several regressions.
- Improved the quality of the client code significantly bringing down the number of incoming crashreports.
  - Added important compile warning flags and fixed about 4000 warnings with my colleagues.
  - Introduced the use of static analyzers (clang-tidy, cppcheck, clang-format) into **Jenkins** review builds posting new warnings directly to Gerrit.
  - I automated everything to process minidumps of crashes in Python, doing analysis, updating Mantis and determining whether it is new or fixed, removing a lot of hands-on work for engineers.
- Introduced a weekly C++ meeting to learn about C++ from each other (as suggested by Jonathan Boccara).

Pengutronix e.K., Hildesheim — **Embedded Linux Kernel Developer** — Dec. 2012 – Aug. 2016

Pengutronix provides Embedded Linux Services for industrial applications bringing Linux to custom hardware.

- Development in ARM subsystems (ASoC, regmap, regulator, Industrial IO, Multi-Function-Device, pincontrol, BATMAN) of the **Linux Kernel**, submitting **248 Patches**. I also implemented/fixed device drivers based on Datasheets.
- I brought up custom embedded systems for customers based on Freescale/NXP i.MX and TI AM335x SoCs, created **PTXdist BSP**s, including **Barebox** bootloader and Linux Kernel. This included Hardware Debugging using **Oscilloscope** and Logic Analyzer.
- I was the main developer on a Mesh-network project streaming data, making heavy use of systemd.
- Maintainer of the **Network Block Device (NBD)** kernel driver.

PC², Paderborn — **Student Developer** — Jan. 2012 – Nov. 2012

Project Enhance

Development of a heterogeneous scheduler for GPUs and CPUs in C++.

PC², Paderborn — **Student Developer** — Jul. 2010 – Nov. 2012

Project Lonestar

Development (Java) and evaluation (Python) of a longterm energy efficient storagesystem.

Paderborn University — **Linux System Administrator** — Jan. 2009 – Jun. 2010

Department of Computer Science

Linux Administration of two fileservers, desktop PCs and other infrastructure as a student.

Education

Paderborn University **Bachelor of Computer Science** 2008 – 2012

Extracurricular

Freifunk Hildesheim **Founder** 2015 – 2017

Initiated the Freifunk Hildesheim initiative bringing free WiFi to people. I set up and maintained the backend and node infrastructure and provided support to customers that wanted to set up their own nodes.

Cbenchsuite 2013 – 2016

Developed a configurable modular benchmark suite in C to make highly accurate performance measurements. Results are stored in sqlite3 and are visualized with a generic visualizer using Python and matplotlib.

BeagleBone Black Cape 2014

Developed and created a BeagleBone Black extension PCB using KiCad EDA that provides common signals (I2C, SPI, ADC, UART, GPIOs) on grouped pin headers as well as providing level transformation to RS232.

$\mathcal{O}(1)$ **Linux CPU-Scheduler**, **Bachelor Thesis** 2012

Paderborn University

I developed alternative scheduling concepts to improve high throughput behavior while keeping fairness between entities. It was implemented in the Linux Kernel by replacing parts of the existing scheduler. The focus was on per-CPU scheduling.

Interests

I love outdoor activities like hiking (especially in the mountains), cycling, kayaking and sailing. I am a huge fan of Norway and love taking photos in my free time. I am always interested in new programming language features, hardware and software and I like to tinker on my current setup to improve my work environment, doesn't matter whether it is my photo pipeline or my C++ editor.

Presentations

FrOSCon 2015, Sankt Augustin **NBD – Network Block Device** Aug. 2015

Introduction to the NBD and what the current state is.

EasterHegg 2015, **Linux Kernel Scheduler 101** Apr. 2015

Braunschweig

Introduction into the structure of the Linux Kernel CPU Scheduler.

Publications

[1] M. Grawinkel, M. Pargmann, H. Domer, and A. Brinkmann. Lonestar: An energy-aware disk based long-term archival storage system. In *2011 IEEE 17th International Conference on Parallel and Distributed Systems*, pages 380–387, 2011.

Berlin, February 7, 2021