Munich, Germany
elm@mail.ederlm.de

Eder Leão Moosmann

linkedin.com/in/ederlf
github.com/ederlf

Work Experience

Senior Software Engineer **DE-CIX** **Jul 2025–Present**
Network Orchestration Team Remote, Germany

- Development of a network orchestrator agent for controlling switches in multiple network sites that handle large amounts of Internet traffic. The stack was built with **Python, Kafka, Postgresql, Netbox, monitored via Grafana and Prometheus and deployed to Kubernetes.**
- Design and implementation of an API to execute commands on a network switch via SSH with **Python, using Fast API and asyncssh.**
- Wrote **Python's Django** extensions for validating the input of network inventory items in **Netbox.**

Software Engineer **Platform.sh (now Upsun)** **Jan 2025–Jun 2025**
Core Engineering team Remote, Germany

- Implemented isolated container integration tests, improving pipeline test speed and reducing the use of heavy end-to-end-tests. I worked with **Go, Podman, FuseFS.**
- Wrote new container agent services for **Mercure, Gotenberg and Postgresql with replication**.

Engineering Manager **Platform.sh (Now Upsun)** **Jul 2022–Dec 2024**
Core Engineering team Remote, Germany

- Managing a team of software and devops engineers, **setting clear goals, priorities and areas for improvement**. The team is responsible for the whole **development lifecycle** of services and applications, including design, implementation, tests and maintenance. The team also works on delivering complex orchestration features such as high availability and replication. We provide services that run for **1000s of cloud grid and enterprise customers.**
- Coordinated and implemented automated service container deployments to **Amazon S3**, transitioning from sporadic releases to weekly updates of hundreds of service images. Regular updates speed up customers' access to new features and fixes.
- Created proof-of-concept images to run **docker** inside **LXC** containers. I wrote **Python** agents that built and orchestrated the execution of the docker containers. It was not trivial because the build and execution of the docker containers are done in read-only images.
- Automated issue request creation on **Linear** for new versions of images using public **apis** and by **scrapping** public information from websites, using **Python, GraphQL API** and running regular scheduled pipelines in the **gitlab CI**.
- Eliminated information and communication friction between my team and other company sectors, by creating an easy to use **Slack application**, implemented in **Go** and using **MySQL, Meilisearch** and the **Gitlab API**.
- Delivered **Clickhouse** as a **high-availability** platform service for the observability team, which enabled log monitoring and improved analysis capabilities for Platform.sh's observability stack. Implementing high-availability in our platform is tricky and requires writing down a complex **Python** agent to **set up and coordinate** container nodes with the clustering **orchestration** tool.
- Established a culture of **incident data analysis**. I classified **root cause** and **mitigation** methods according to previous incidents. I generated reports, using **Python and Metabase**, with statistics for the most common causes and mitigations. The reports were then used to set incident related **KPIs**.

Cloud Software Engineer **Platform.sh (Now Upsun)** **2020–2022**
Core engineering team Remote, Germany

- Improved pipelines and automation for **Continuous Integration (CI) on Gitlab**. I have used **YAML**, **Gitlab CI**, **Python**, **bash** and **Go**. Achieved **speed up** of CI jobs for image releases by **75%.**
- Built and maintained multiple **Debian packages** for services and application container images. I created packages for software not available upstream and packages with the need for quick patches. Thanks to the diversity of packages we maintained, I was exposed to many **build systems**, of languages such as **C**, **Go**, **Rust**, **Java** and many more.

- Implemented integration tests in **continuous integration** for internal image's packages, **reducing by 80%** the presence of faulty packages in our container images.
- Reduced the time to identify and fix broken image builds, by introducing **CI pipeline health monitoring** with **Python**, **Influxdb and Grafana**.
- Implemented an emulator for testing the components of the company's network edge layer solution using **Python, Mininet, Go and GoBGP.** The emulator allowed lighter end-to-end tests of the network software without the need to bootstrap the whole orchestrator.
- Worked on the port from **Python 2** to **Python 3** of the platform git server. The **git server** contains tens of thousands of lines of code, with multiple internal and external dependencies that also required porting.

Researcher **Queen Mary University** **2015–2019**

QMUL Networks group

London, UK

- Developed a fast network simulator for **OpenFlow** and routing protocols (**BGP**, **OSPF**) using **C**, **Cython**, **Python, Quagga, Bird, ExaBGP and Linux** network namespaces.
- Developed network applications to scale **SDN** for large **Internet Exchange Points** using **Python**, **Shell script**, **Flask, InfluxDB, Grafana** and **MongoDB**.
- Worked on **NS3** and **Mininet** simulations of data center solutions to reduce congestion. I have used DCTCP**, BGP, Python and C++.**
- Performed Internet measurements, data analysis and visualization of **DNS**, **Netflow** and **BGP** records using **Python** (**numpy, matplotlib, scipy, networkx**) **Java** and **C**.

Software Developer **Lenovo** **2014-2015**

Campinas, Brazil

- Created, built and documented an open source version of a proprietary **Operating System** for storage devices (Unfortunately, it was never released.)

Research Developer **CPqD** **2012-2014**

Campinas, Brazil

- Developed open source routing and switching software for **Software Defined Networks**. At the time, my open source contribution was used by thousands of users in the industry and academia. I dealt, since the early days of the projects, with **Github** issues and pull requests. I worked with many languages such as **Python, C, C++ and Java**.

Education

- **PhD Computer Science,** Queen Mary University of London. **2015-2019**
- **M.Sc. Electrical Engineering,** University of Campinas, Brazil. **2013–2014**
- **B.Sc. Computer Science,** University of Sao Paulo, Brazil. **2010–2013**

Languages, Technologies and Skills

- Languages: Python, Go, Rust, C, C++, Java
- Technologies: Kafka, Netbox, Git, Linux, Debian, Linux Networking, SDN, OpenvSwitch, MySQL, Redis, MongoDB, InfluxDB, AWS, Podman, LXC, Docker, Gitlab CI
- Skills: Very strong at debugging, self-driven, goal oriented, problem solving, organization, time management, team work