

# Benjamin Doerry

Email: [ben@doerry.uk](mailto:ben@doerry.uk)

LinkedIn: [in/bendoerry](https://www.linkedin.com/in/bendoerry)

GitHub: [bendoerry](https://github.com/bendoerry)

Software Engineer with 4 years of industry experience in Software Development and Test Engineering. Interested in development roles with innovative potential, high impact and learning opportunities.

## Work Experience

### Software Engineer at Suade Labs: (03/2023 – Present)

Worked in the Platform team, working on maintaining and implementing all Rust Services, as well as optimisations of existing Python services.

- Created Rust service for running background jobs. Integrating with existing job management logic, as well as providing automatic dependency injection of database connections and deserialisation of job parameters.
- Lead fortnightly talks for the engineering team and company. Topics range from testing and best practices to competitors and outage post-mortems.
- Took on the responsibility to mentor and lead RegTech engineers, teaching them best practices and providing them with peer programming and review opportunities. This has resulted in engineers moving over into the Platform team.
- Refactored background job handling logic to reliably identify non-responsive jobs and improve job parameter handling and deserialisation. This also implemented checking job declarations at startup ensuring any misconfigurations are discovered immediately.
- Migrated existing data ingestion service from a 2-year-old nightly release to stable Rust. This required replacing an unmaintained framework with an alternative that still met requirements.
- Proactively identified and fixed suboptimalities in docker builds, cutting image size by up to 50%.
- Extended linting and typechecking of the codebase. This involved fixing all existing violations as well as providing education on how to avoid future issues. At times requiring upstream contributions to ensure linters could be used effectively.
- Refactored configs/templates access logic to increase reliability and massively reduce the need for special casing logic. This clearly defined failure modes and identified existing violations, allowing them to be fixed before being encountered by customers.
- Improved integration of CI pipeline results with GitLab merge requests, automatically presenting linting and test results, as well as highlighting code coverage in diffs.

### Software Engineer at Pexip: (10/2022 – 01/2023)

Worked in the R&D Team, working on migrating an internal support service to Kubernetes and rewriting a core media layout engine in Rust.

### Software Engineer at Pexip: (06/2021 – 10/2021)

Interned in the R&D team, working on modernising and expanding a customer facing support service.

- Migrated file storage from local disks to Google Cloud Storage, Allowed for easier backups and serving download links with built-in expiry. The file upload mechanism pointed directly to GCS, thus offloading the balancing and upload logic to Google.
- Expanded access controls for customer uploaded files, reducing permission scopes.
- Rewrote GitHub and Zendesk integrations to use newer auth methods and standardised data structures.
- Migrated service from isolated GitHub repository to a SRE shared GitLab monorepo. Creating CI pipelines for running tests and linters, as well as service deployment.
- Split up deployment secrets to give services access to only required secrets.
- Created a container-based development environment with preinstalled dev utilities. Thus reducing time taken for new engineers to get a working dev environment.

### Software Engineer at StarLeaf: (06/2020 – 10/2020)

Interned in the Cloud R&D Team, working on internal monitoring and tooling.

- Created dashboards of resource usage and user feedback scores for each PoP in the StarLeaf Cloud. Thus giving Support and Operations insight into call quality worldwide.
- Standardised statistics charts generation and retrieval for internal statistics pages, using asynchronous loading of charts to increase page load speed significantly.
- Implemented caching for customer/log analysis tooling, reducing friction when accessing commonly requested logs.

### Test Engineer at StarLeaf: (09/2018 – 09/2019)

Interned in the Cloud R&D Team, where I was testing lead of integrations of the StarLeaf cloud with 3rd party software/plugins (such as Outlook 365 & Slack).

## Skills & Experience

### Development Tools:

- Git & git-svn
- GitHub & GitLab
- VS Code & PyCharm
- GKE
- Jira & Confluence

### Tools & Libraries:

- Docker
- Kubernetes
- Kind
- Kustomize
- Sentry
- Grafana
- Flask (-RestX)
- FastAPI
- Celery
- Pydantic
- Redis
- Postgres

### Programming Languages:

- Python
- Bash
- Rust
- Javascript (some)

### Languages:

- English (native)
- German (native)

## Side Projects

Details for all projects can be found at [ben.doerry.uk/projects](https://ben.doerry.uk/projects).

### Shorelark:

A simulation of evolution written in Rust, targeting the Web via WebAssembly. It involved writing Neural Network and Genetic Algorithm Libraries from scratch.

### Zero To Production:

An exploration into API development in Rust, focussing on Domain Driven Development, Observability, and Scaling.

## Education

### Durham University: (2019 – 2022)

BSc in Computer Science and Mathematics within the Natural Sciences programme.

Modules including: Security Engineering, Parallel Scientific Computing, Networks & Systems, Complex Analysis, Algebra, Geometry, and Analysis in Many Variables.

Technologies Officer of The Durham Union Society for 2 years during the pandemic.

## Hackathons & CTFs

- Snyk 2022 CTF (Team placed in top 20 of 400 teams)
- 2020 Durham BAE Systems CTF
- Durhack 2020 Hackathon