Modernizing Fink Infrastructure with Kubernetes & Open Source Contributions

Fabrice Jammes
LSST Builder / Research engineer
Laboratoire de Physique de Clermont

Julien Peloton
FINK PM / Research engineer
IJCLab, Orsay

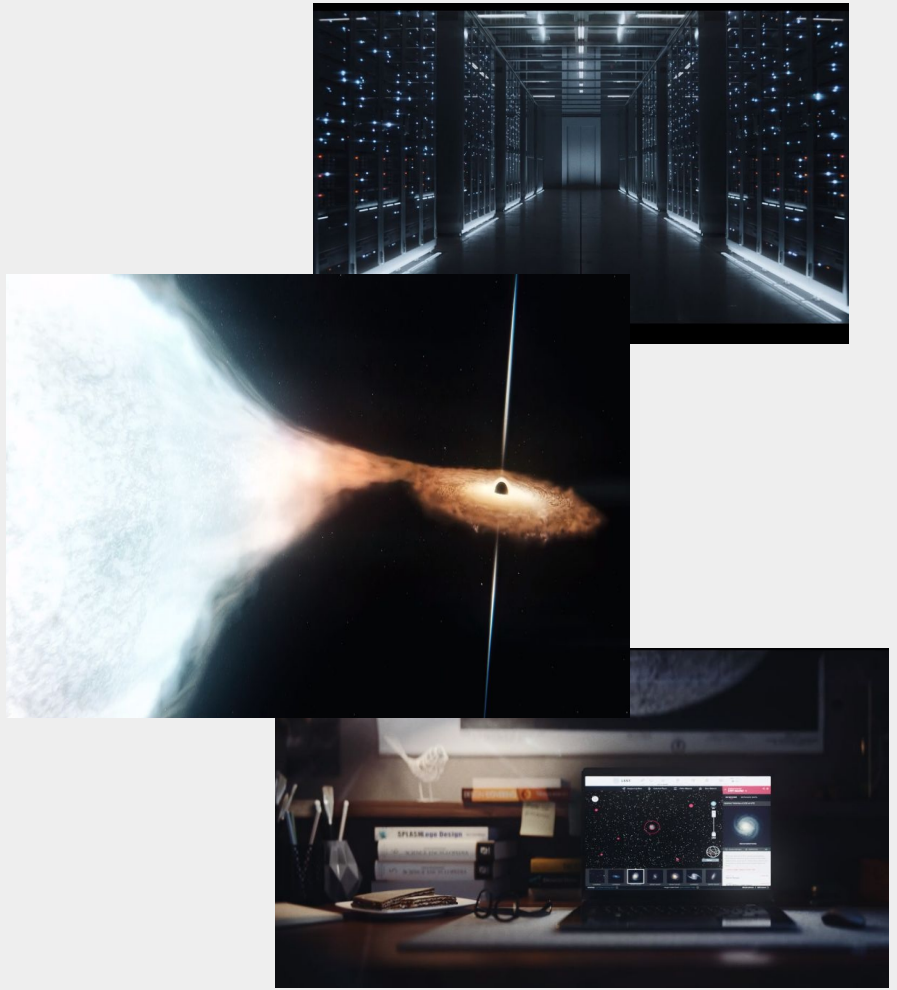
Emille Ishida
FINK PM / Research engineer
Laboratoire de Physique de Clermont

Context

Fink and real time alert processing

Goals

Improve testability, observability, and deployment agility



 Key achievements this year

Highlights of the Year

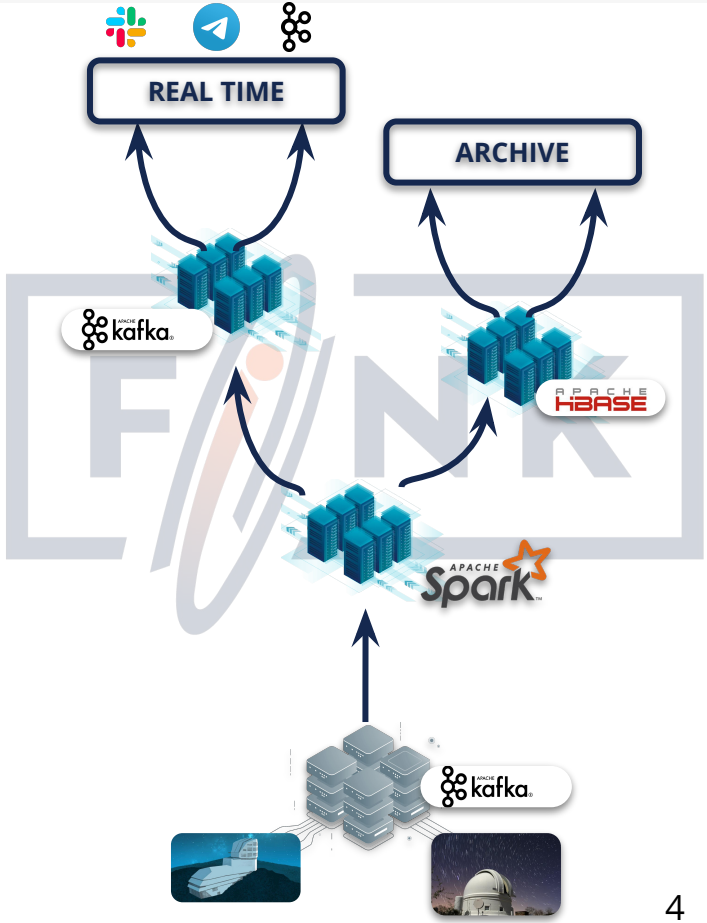
DONE: Continuous Integration of Fink on Kubernetes with integration tests

Open-source contributions to the Spark ecosystem

Spark observability: Prometheus (Linux Foundation) and sparkMeasure (CERN)

Fink deployment in French certified cloud

Early evaluation of AI coding assistants



CI & Global Integration Testing on Kubernetes

What was missing?

- Fink lacked **automated, Kubernetes-native integration testing**

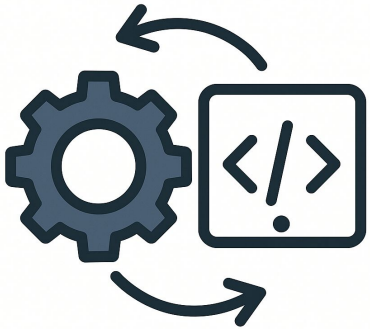
- Previous tests were not environment-consistent:
  - Significant divergence between the test and the production environment

=> Hard to reproduce bugs and validate infrastructure changes safely

What was done

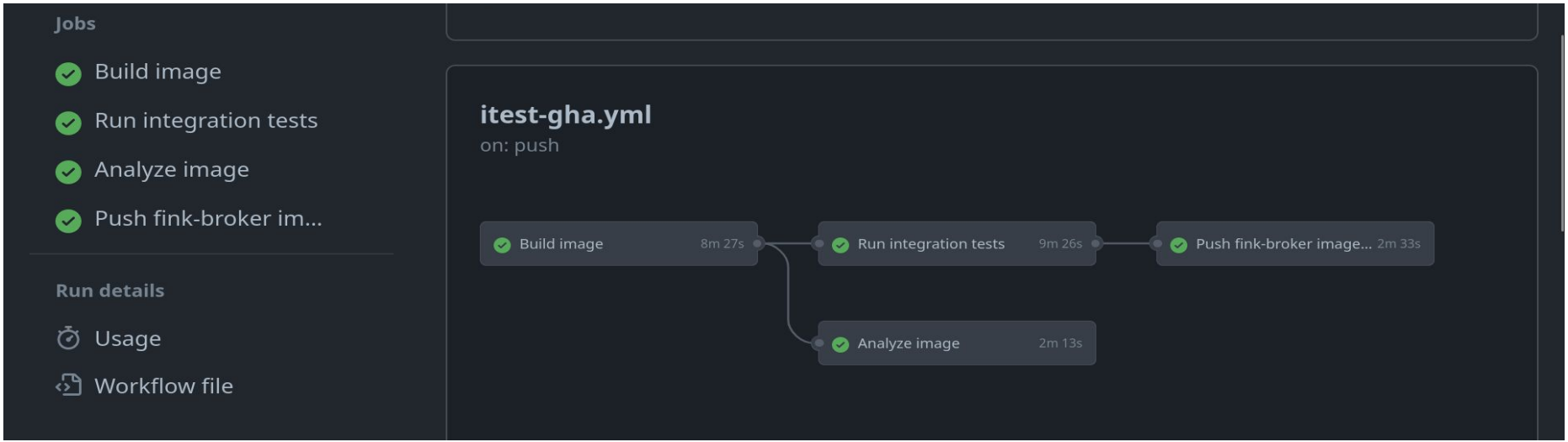
- **CI/CD pipelines** to deploy full Fink stack on ephemeral K8s clusters
- CI is **triggered on every developer commit**: rapid feedback
- Run **end-to-end integration tests** with simulated data flows
- Fink environment can be spun up **locally on a laptop or on production platform** in minutes
- Greatly improves **developer onboarding**

Modern devops tools



Opens source and free devops tools

- GitHub Actions
- Kind (Kubernetes in Docker)
- ArgoCD - Helm (also used by LSST construction team)
- Security: vulnerabilities and code scanning
- Smoke tests on alert ingestion, Kafka flow, filtering & output



Contributions to the Open Source Ecosystem

Direct engagement

- Discussions & feedback on Rust-based Kubernetes SDK
  - https://github.com/kube-rs/kube/discussions/1678
- Bugs and improvements on HDFS Operator
  - https://github.com/stackabletech/hdfs-operator/issues/625
  - https://github.com/stackabletech/hdfs-operator/issues/626
- Bugs and improvements on Spark Operator
  - https://github.com/kubeflow/spark-operator/issues/2380
  - https://github.com/kubeflow/spark-operator/issues/2004

Official Docs contribution

New section in the Spark Operator documentation about Prometheus integration

Monitoring Spark Applications with Prometheus and JMX Exporter | Kubeflow

Related grafana dashboard

Apache Spark JMX Metrics Dashboard for Kubernetes | Grafana Labs

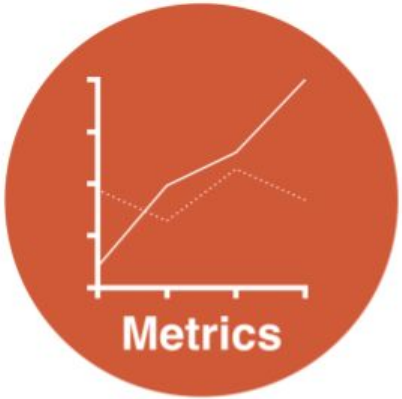
Observability: Prometheus, Spark, and More

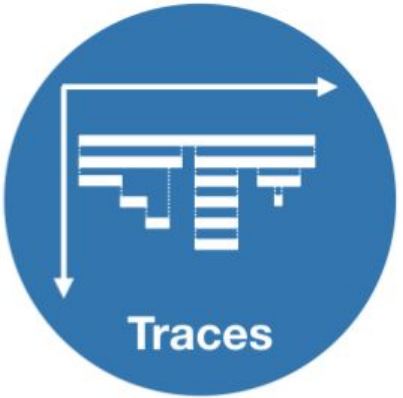
Why observability matters?

Real-time systems like Fink need fine-grained metrics

Especially for **Spark** science jobs processing ZTF/LSST streams

Need to understand what's happening under the hood!





Prometheus in brief

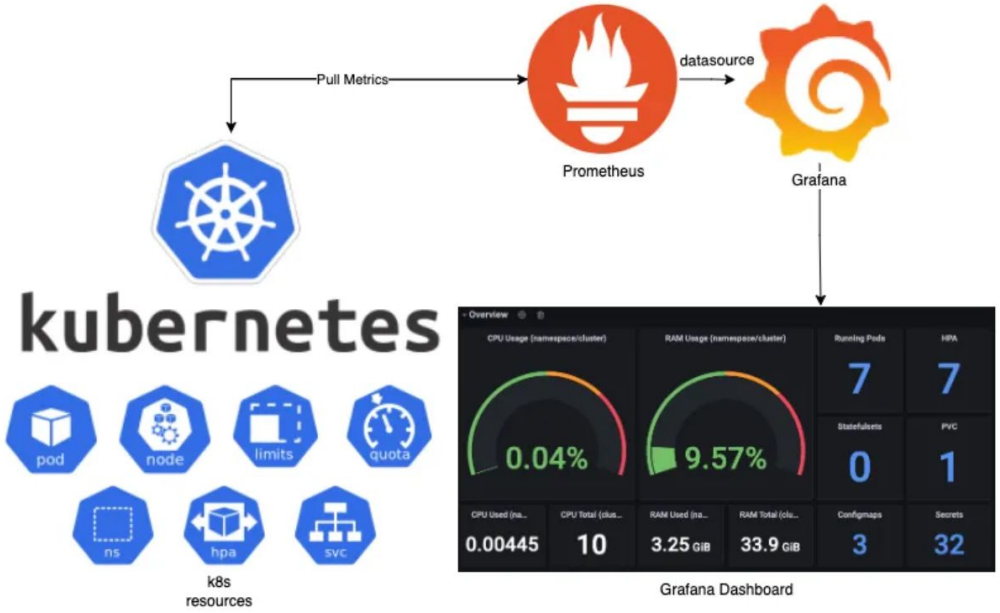
Time-series DB & query engine (PromQL)

Pull-based metrics model

Native K8s support

Graduated level by Linux

Foundation (CNCF)



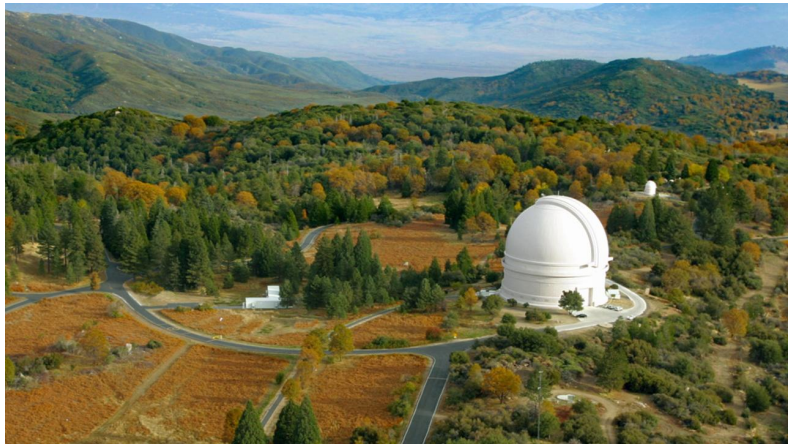
Production Deployment in French Secured Cloud

Fast & Reliable

- Full Fink deployment, with ZTF integration, in < **1 day**
- Target platform: **SecNumCloud-labeled** 🇫🇷 provider
- Platform available thanks to DECALOG Master project

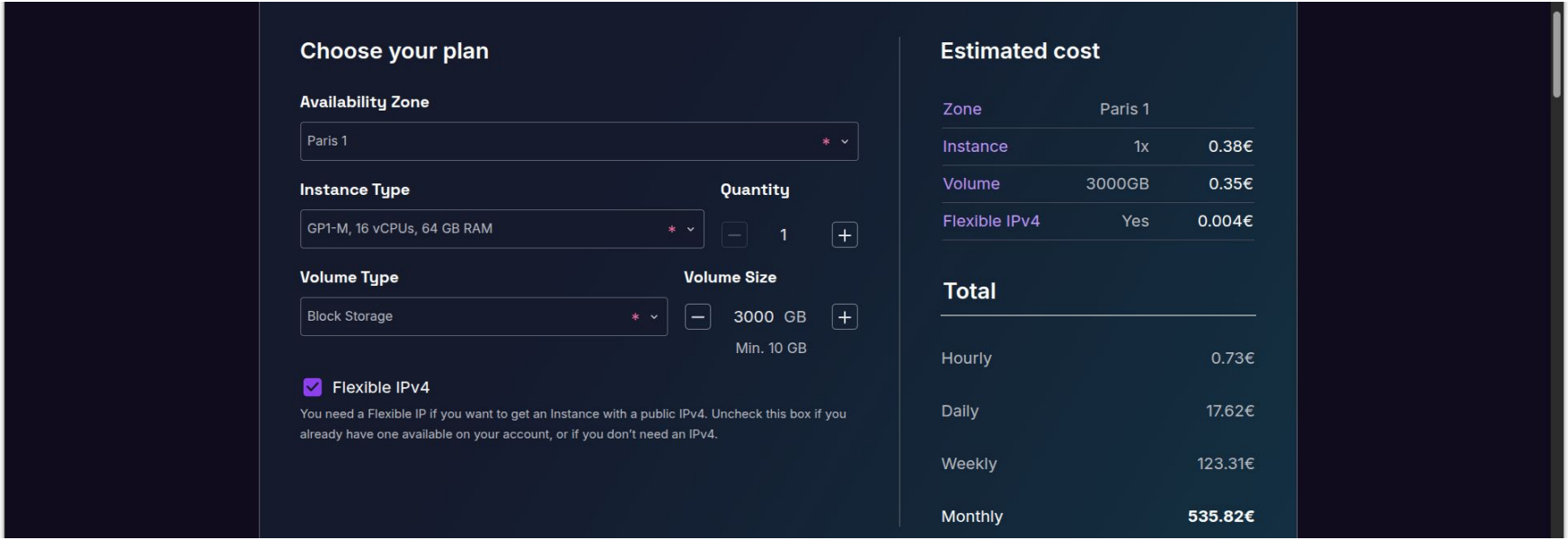
Validation

- Initial data tests with Julien using ZTF alerts for one night
- Infra behaves as expected
  - Kafka, Spark, Science modules **OK**
- Larger-scale validation planned: at least one week



Cost@Scaleway

- <300 euros/month for the compute (16 cpus 64 GB memory), all inclusive (manpower, power, cooling)
- add 250 euros/month for 3TB storage



Using AI Tools for Coding

Tools tested

- **GitHub Copilot** (free for academics)
- **ChatGPT Codex** (for code comprehension & exploration)
- An experimentation with **Cursor AI** is planned

Insights

- Great for
  - **boilerplate**, simple logic, shell commands
  - Documentation
  - Unit test skeleton
- Less effective for:
  - Complex refactoring
  - Large open-source codebase
  - Complex algorithms involving multiple projects/services

Bottom line

> They code fast, but not always right. Human review and refactor are still essential.

- Crucial to **track the evolution** of these tools
- Their use can accelerate productivity in targeted tasks

📌 Summary & Takeaways

1. CI on K8s greatly improves trust in Fink's infra
2. Prometheus integration advances our ability to debug & optimize Spark jobs
3. Open-source involvement benefits both community & Fink
4. Real-time deployment is achievable in secure European clouds
5. AI tools can help, but require human oversight — especially in research

Conclusion

Q&A

Let’s discuss !

- Contributions
- Deployment strategies
- Prometheus integrations
- AI coding