

All Presenters:

- 1) Please do not delete any slides
- 2) Follow instructions on slide 2 and 3 to create your lightning talks slides



Google Summer of Code

GSoC 2025

Mentor Lightning Talks

October 24th, 2025

Speaker Order - Friday 9:30-10:30AM

Order	Mentor Name(s)
1	Sanju Tiwari
2	Cyrille Artho
3	Ankit Mahato
4	Ayush Billore
5	Stefano Garzarella
6	Ashish Yadav
7	Shivay Lamba
8	Shubham Jain
9	Bawantha Thilan

Order	Mentor Name(s)
10	Rudra Chopra
11	Diya Solanki
12	M Viswanath Sai
13	Vedant Singh
14	Ege Korkan
15	Jaideep Prasad
16	Anastasia Hernandez-Koutoucheva
17	Vissarion Fysikopoulos
18	Andreas Dräger, NRNB
19	Vedant Jain



Google Summer of Code

DBpedia Association



Sanju Tiwari (Mentor)

The DBpedia Association was founded in 2014 to support DBpedia and the DBpedia Community. The DBpedia Association is currently situated in Leipzig, Germany and affiliated with the non-profit organisation [Institute for Applied Informatics](#) (InfAI)

DBpedia is a large community driven project that aims to extract structured information from wikipedia and make it available on the web as Linked Data

It takes the semi-structured data from the wikipedia infoboxes and convert into a Knowledge-Graph by using RDF.

The screenshot shows the DBpedia GSoC dashboard. At the top, there are navigation links for "Programs" and "My Projects". On the right, it displays "DBpedia | Dr. Sanju Tiwari". Below this, four project entries are listed vertically, each with a dropdown arrow icon:

- 2025**: Program Period: Jan 27, 2025 - Nov 19, 2025; Status: Accepted
- 2024**: Program Period: Jan 22, 2024 - Nov 14, 2024; Status: Accepted
- 2023**: Program Period: Jan 23, 2023 - Nov 17, 2023; Status: Accepted
- 2022**: Program Period: Feb 7, 2022 - Dec 2, 2022; Status: Accepted

GSoC 2022: Template Discovery for Neural Question Answering over DBpedia

<https://sauravjoshi23.github.io/GSoC-Neural-QA-Model-DBpedia/>

GSoC 2023: Question-Answering over DBpedia with Pretrained Auto-regressive Models

<https://mehrzedshm.github.io/GSoC-2023-blog/>

GSoC 2024: Building the Foundational Architecture of Hindi DBpedia Chapter

<https://deba-iitbh.github.io/deba-gsoc24/>

GSoC 2025: Evals and Enhancements

<https://advenk.github.io/av-blog/>



Work during 2024-25 as a leading mentor of Hindi DBpedia Chapter

GSoC 2024 laid the groundwork by developing a Neural Hindi Wiki Triple Extraction pipeline.

The project combined mapping-based extraction (infobox mappings) with neural extractors to produce subject–predicate–object triples from Hindi text, linking them to **DBpedia ontology** and entities.

Tokenization and **POS tagging** (using Stanza), **NER** tagging (using IndicNER), rule-based mention detection, coreference resolution (using a multilingual model TransMuCoRes), relation extraction (using models like mBERT-based REBEL, IndIE, and an LLM (Mistral)), and entity linking (using mGENRE for Wikidata/DBpedia linking).

GSoC 2025 aimed to enhance the existing pipeline by first evaluating and then improving it.

Based on 2024, evaluated different prompt strategies like zero-shot, few-shot, chain-of-thought etc with small language models like gemma, llama, mistral for the task of multilingual information extraction. **gemma works best for multilingual IE** using the ReAct framework.

the SPARQL endpoint for the Hindi chapter was dockerized and benchmarked on a temporary server



Question Answering over DBpedia with Fine-tuned Autoregressive Models

Tommaso Sorò^{1,2}, Saurav Joshi^{1,3}, Sanju Tiwari^{1,4}, Mehrzad Shahinmoghadam¹ and Anand Panchbhai¹

¹DBpedia Association, Leipzig, Germany

²Liber AI Ltd, London, United Kingdom

³University of Southern California, Information Sciences Institute, Marina del Rey, CA, USA

⁴Sharda University, Greater Noida, India

Abstract

This paper presents a novel approach to question answering over DBpedia by fine-tuning autoregressive code-generation models to translate natural language questions into SPARQL queries. The authors introduce the NSpM Dataset, a large-scale dataset of 7.7 million question-SPARQL pairs covering DBpedia's ontology. We fine-tuned three open-source code generation LLMs – CodeGen-350M, StarCoder-1B, and Codellama-7B – on a curated subset of this dataset. Despite limited training time, StarCoder-1B achieved the best performance across evaluation metrics. Results from the 1st Text2SPARQL Challenge at ESWC 2025 reveal that while models partially learn SPARQL syntax, they struggle with query semantics and ranking quality.

Keywords

Semantic Web, SPARQL, semantic parsing, question answering, neural networks, large language models

1. Introduction

In recent years, large knowledge graphs (such as DBpedia [1], Wikidata [2], and Firebase [3]) have grown to contain more than billions of semantic triples forming massive repositories of structured knowledge. To democratize this knowledge, Question Answering (QA) systems have been developed to let users retrieve information from these knowledge bases using natural language questions instead of requiring them to write complex queries in formal languages like SPARQL [4, 5, 6]. However, translating natural language questions onto the rich ontology of a large graph and generating an executable SPARQL statement remains highly challenging, because a system must resolve lexical ambiguity, link surface terms to the correct entities and relations, preserve graph-structural constraints, and generate syntactically valid queries that support multi-hop, and compositional reasoning [7, 8, 9, 10, 11].

Early computational research on Text-to-SPARQL has predominantly relied on rule and template-based systems that align parsed questions with fixed SPARQL skeletons; representative examples are the Template-Based Question Answering over Linked Data (TBSL) framework [12] and the keyword-to-template generator described in [13], which require extensive manual engineering and fail to generalize to paraphrases, unseen ontologies, and multi-hop queries. Semi-automatic approaches—notably the skeleton-grammar with neural-ranking model SPARQA [14] and the classifier-guided template-selection method proposed in [15]—learn and rank patterns from data, reducing authoring effort yet still inherit rigidity from their finite template data stores and remain brittle on out-of-distribution question forms.

In this paper, we contribute to the Text-to-SPARQL research through two primary advancements. First, we introduce the NSpM dataset, a massive, automatically-generated collection comprising 7.7 million



Enhancing Multilingual Information Extraction - DBpedia Hindi Chapter

Abstract

This paper documents the journey of the DBpedia Hindi Chapter through two consecutive Google Summer of Code (GSoC) cycles. It details the framework built in 2024 which involved a foundational proof-of-concept for extracting structured triples from Hindi Wikipedia and the subsequent advancement and evaluation work done in 2025. We discuss our design decisions, the integration of specialized NLP tools, experiments with prompting strategies for Small Language Models, and the development of a hybrid end-to-end relation extraction and linking framework. The result is an open-source framework that serves as a blueprint for knowledge graph creation in other underrepresented languages, representing a tangible step toward digital equity in the semantic web.

Keywords

Knowledge Graphs, Semantic Web, Information Extraction, Language Models

1. Introduction and Motivation

Knowledge graphs have become the backbone of modern information systems, however their linguistic diversity is still lagging behind. Indic languages like Hindi which is spoken by over 600 million people are untapped in their potential for extracting structured knowledge from text.

2. GSoC 2024: Building the Foundational Architecture

3. GSoC 2025: Evals and Enhancements

The goals from 2025 were multi-fold



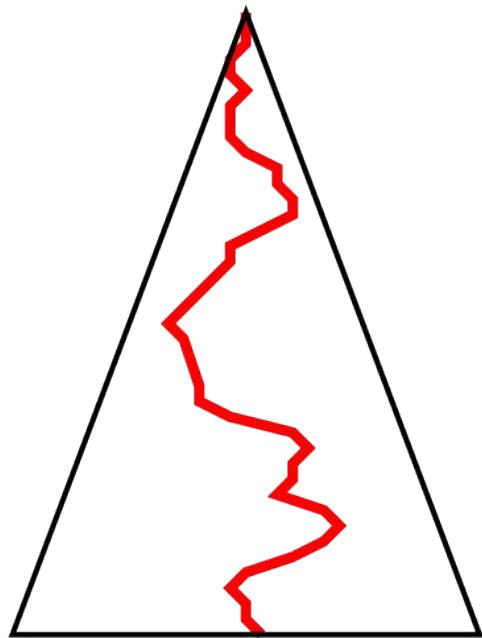
Google Summer of Code

Cyrille Artho

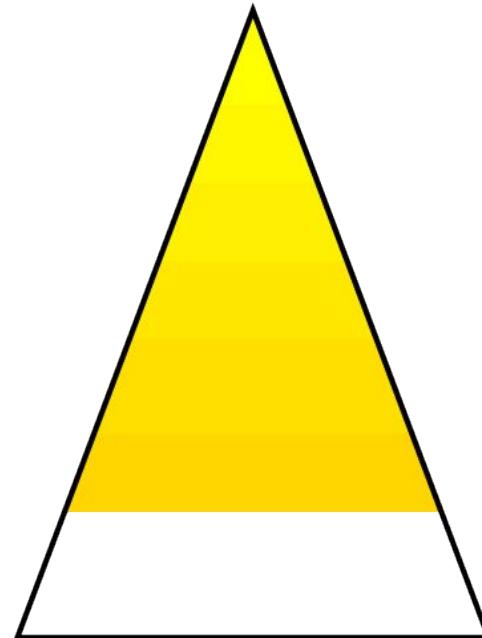
The JPF Team

Beyond Testing

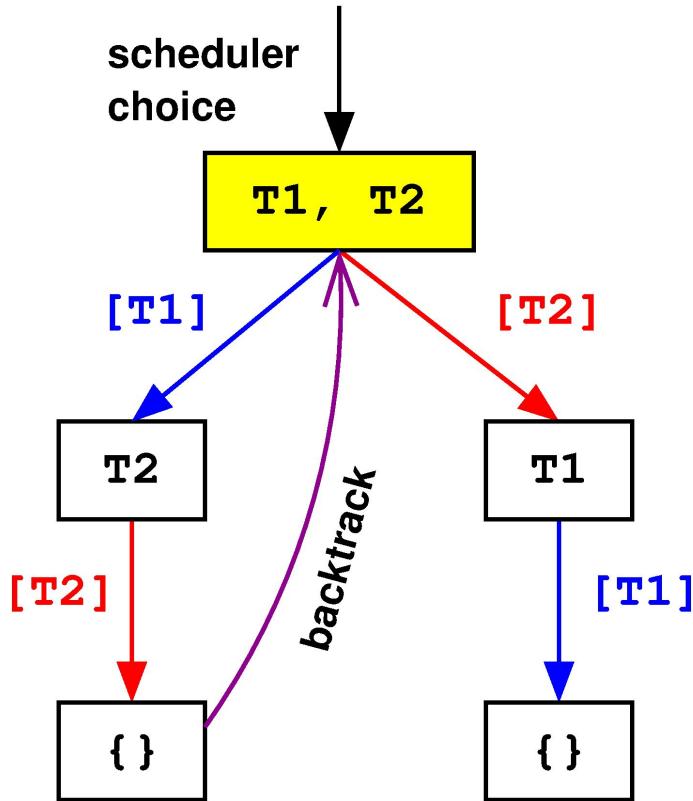
Testing



Model Checking



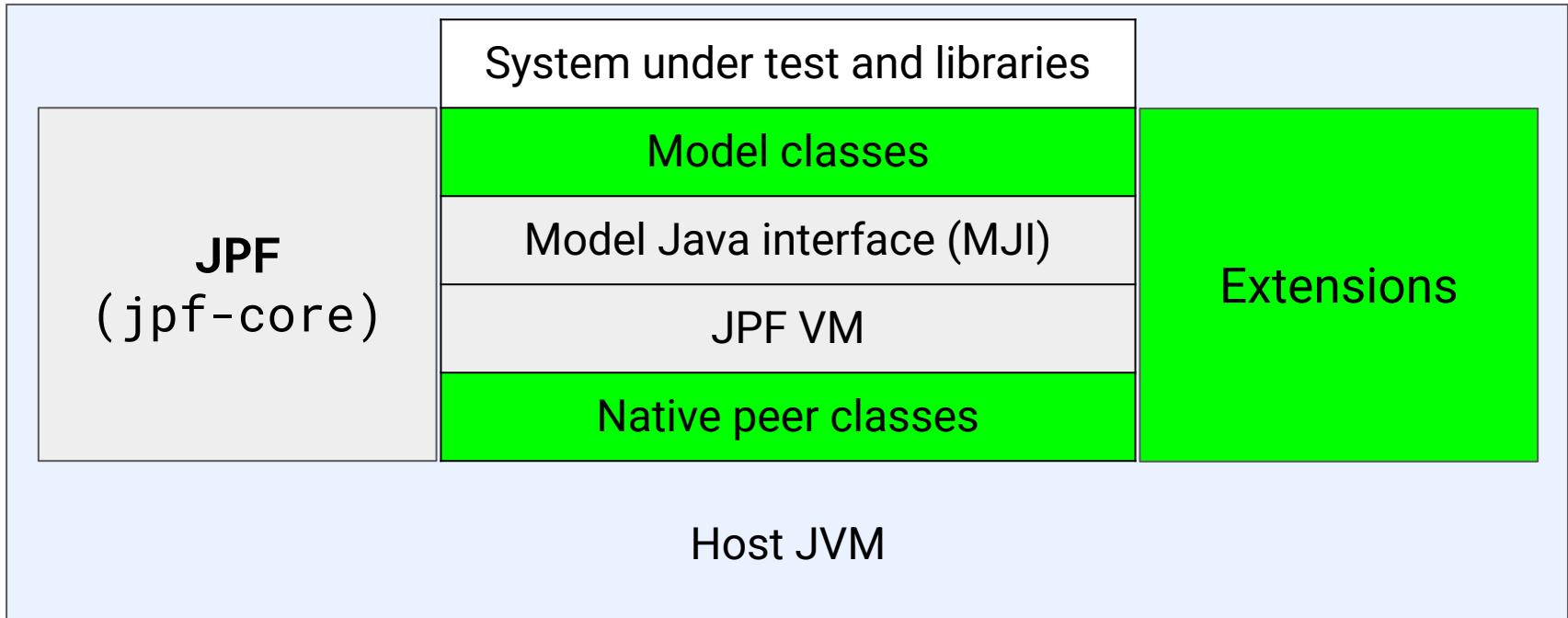
Software Model Checking



- Backtrack execution to explore all choices.
- Keep copy of visited states.

Limitations:
Memory, native code.

Mahmoud's work in GSoC 2025



Update JPF and extensions to support OpenJDK 11 and 17



Google Summer of Code

Ankit Mahato

API Dash

The screenshot shows the Postman application interface with a modal dialog open in the foreground. The modal is titled "Ollama" and contains configuration settings for an AI endpoint.

Endpoint: `http://localhost:11434/v1/chat/completions`

Models:

- qwen2.5:latest
- llama3.2:latest

Save

The background of the Postman interface shows a list of requests on the left sidebar, including "Ollama", "OpenAI", "Anthropic", "Gemini", and "Azure OpenAI". The main workspace shows a single request entry for "Ollama".

Save + New :

HTTP untitled

GET api.apidash.dev/country/data Send ➤

Requests

Variables

History

Logs

Filter by name or url

GET api.apidash.dev/country/ ...

GET sse.dev/test?interval=3

GET 172.217.19.164:80

POST 213.192.2.75:40144/transcri ...

GET sse.dev

Params Auth Headers Body Scripts

View Code

200: OK 1.11 s

Response Body Headers

Preview Raw

Search... Expand All Collapse All

data : {3}

area : 3287260.0

population : 1407563842

population_female_perc : 48.39

+ Add Param

?

⚙️

Global

↻

```
[  
  {  
    "id": 1,  
    "name": "Rick Sanchez",  
    "status": "Alive",  
    "species": "Human",  
    "type": "",  
    "gender": "Male",  
    "origin": {  
      "name": "Earth (C-137)",  
      "url": "https://rickandmortyapi.com/api/location/1"  
    },  
    "location": {  
      "name": "Citadel of Ricks",  
      "url": "https://rickandmortyapi.com/api/location/3"  
    },  
    "image": "https://rickandmortyapi.com/api/character/avatar/1.jpeg",  
    "episode": [  
      "https://rickandmortyapi.com/api/episode/1",  
      "https://rickandmortyapi.com/api/episode/2",  
      "https://rickandmortyapi.com/api/episode/3",  
    ],  
    "url": "https://rickandmortyapi.com/api/character/1",  
    "created": "2017-11-04T18:48:46.250Z"  
  },  
  {  
    "id": 2,  
    "name": "Morty Smith",  
    "status": "Alive",  
    "species": "Human",  
    "type": "",  
    "gender": "Male",  
    "origin": {  
      "name": "Unknown",  
      "url": ""  
    },  
    "location": {  
      "name": "Citadel of Ricks",  
      "url": "https://rickandmortyapi.com/api/location/3"  
    },  
    "image": "https://rickandmortyapi.com/api/character/avatar/2.jpeg",  
    "url": "https://rickandmortyapi.com/api/character/2",  
    "created": "2017-11-04T18:50:21.651Z"  
  },  
  ...  
]
```



Generated Component



Name: Rick Sanchez
Status: Alive
Species: Human
Gender: Male



Name: Morty Smith
Status: Alive
Species: Human
Gender: Male



Name: Summer Smith
Status: Alive
Species: Human
Gender: Female



Name: Beth Smith
Status: Alive
Species: Human
Gender: Female



Name: Jerry Smith
Status: Alive

Any Modifications?

Export Code

Make Modifications



Google
Summer of Code

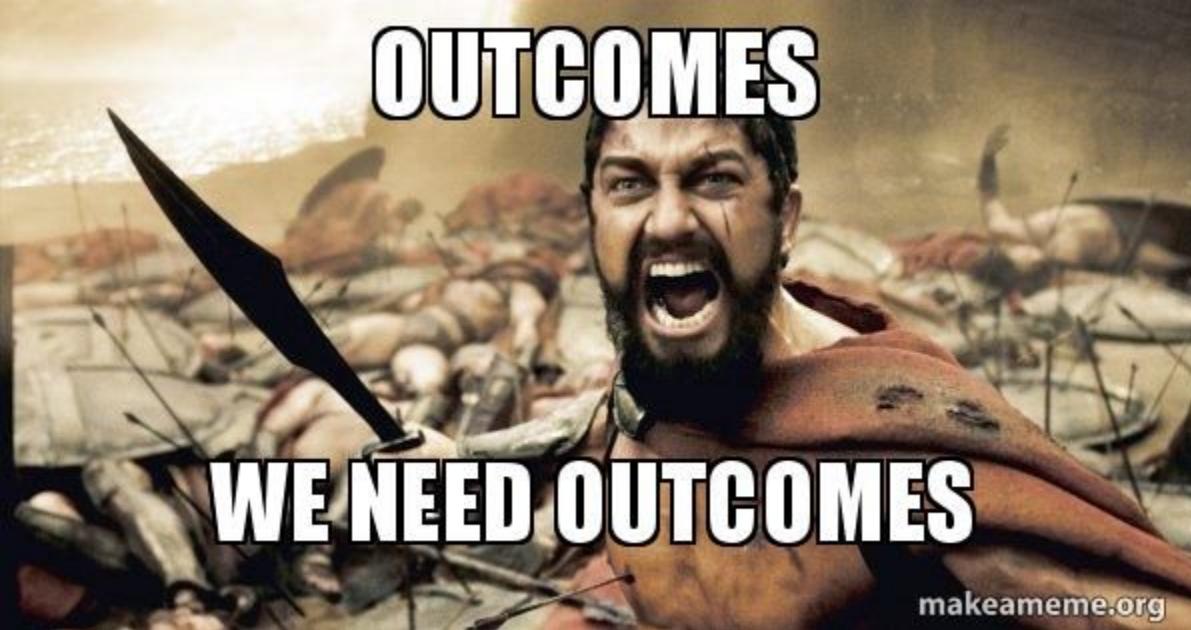
Ayush Billore

AI made proposals efficient — and identical.



First Impression != Last Impression





OUTCOMES

WE NEED OUTCOMES

makeameme.org



Google Summer of Code

Stefano Garzarella

QEMU (& rust-vmm)



Google Summer of Code

Ashish Yadav

AnkiDroid



Google Summer of Code

Review Reminders | Contributor: Eric

- **What: The Goal**

- To design and build a completely new, reliable, and feature-rich review reminder system to replace AnkiDroid's old, non-functional one.

- **Why: The Problem**

- The existing notification system had been completely broken for over 5 years, leading to it being one of the most common and persistent user complaints on the Google Play Store.

- **How: Key Achievements**

- A new, comprehensive reminder system was built from scratch, featuring reliable, customizable (global/deck-specific) notifications, advanced filters, and a clean UI.



Google Summer of Code

AnkiDroid Tablet & Chromebook UI | Contributor: Hari Srinivasan

- **Why: The Problem**

- The existing AnkiDroid interface was designed for phones, resulting in poor usability, wasted space, and a lack of desktop-standard features (like mouse controls) on large screens.

- **How: Key Achievements**

- The project introduced resizable dual-pane layouts, enhanced mouse support (like right-click), and a live note previewer to create a desktop-like experience.



Google Summer of Code

Shivay Lamba

Kubeflow

Project 11: Support Postgres for Kubeflow Pipelines backend

Goal - Implement support for PostgreSQL as an alternative to MySQL/MariaDB in Kubeflow Pipelines
([kubeflow/pipelines#9813](#))

- Kubeflow Pipelines must store information about pipelines, experiments, runs, and artifacts in a database. Currently, the only database it supports is MySQL/MariaDB.
- We plan to support PostgreSQL as an alternative to MySQL/MariaDB so users will be able to reuse existing databases, and PostgreSQL will be a good use case for supporting multiple databases.

Mentee - Yunkai

Technical Details

Architecture Design

1e

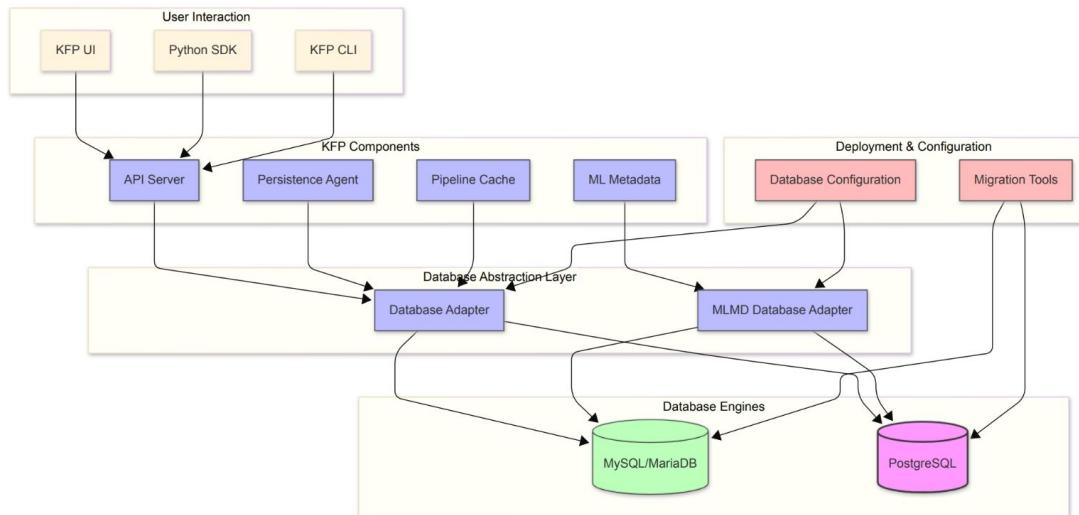


Figure 1: High-Level Architecture

The following key components interact with the database and require modification:

1. **API Server** - Provides REST APIs for pipeline management
2. **Persistence Agent** - Synchronizes execution state from Kubernetes to the database
3. **Cache Server** - Manages caching for repeated pipeline steps
4. **ML Metadata (MLMD)** - Handles metadata storage and retrieval

[WIP]Feature/postgres integration #12379

Code

Open

kaikaila wants to merge 2 commits into [kubeflow:master](#) from [kaikaila:feature/postgres-integration](#)

Conversation 8

Commits 2

Checks 159

Files changed 91

+4,045 -2,337



kaikaila commented 5 days ago · edited

Contributor

...

Summary

This PR adds full PostgreSQL (pgx driver) support to Kubeflow Pipelines backend, enabling users to choose between MySQL and PostgreSQL as the metadata database. The implementation introduces a clean dialect abstraction layer and includes a major query optimization that benefits both database backends.

Key achievements

- ✓ Complete PostgreSQL integration for API Server and Cache Server, addressing [#7512](#), [#9813](#)
- ✓ All CI tests passing (MySQL + PostgreSQL).
- ✓ Significant performance improvement for ListRuns queries. This PR is expected to address the root causes behind [#10778](#), [#10230](#), [#9780](#), [#9701](#)
- ✓ Zero breaking changes - backward compatible with existing MySQL deployments

What Changed

1. Storage Layer Refactoring - Dialect Abstraction ([backend/src/apiserver/common/sql/dialect])

• Problem

SQL syntax was tightly coupled to MySQL.

• Solution

Introduced a DBDialect interface that encapsulates database-specific behavior

Identifier quoting (MySQL backticks vs PostgreSQL double quotes)

Placeholder styles (?) vs \$1, \$2, ...)

Aggregation functions (GROUP_CONCAT vs string_agg)

Concatenation syntax (CONCAT() vs ||)

• Files

Core dialect implementation → [backend/src/apiserver/common/sql/dialect/dialect.go](#)

Dialect-aware utility functions → [backend/src/apiserver/storage/sql_dialect_util.go](#)

Reusable filter builders with proper quoting → [backend/src/apiserver/storage/list_filters.go](#)

All storage layer code now uses

```
q := s.dbDialect.QuoteIdentifier  
ob := s.dbDialect.OuervBuilder()
```

Reviewers

gmfrasca

hbelmiro

HumairAK

mprahli



Still in progress? Learn about draft PRs



Assignees

No one assigned

Labels

do-not-merge/work-in-progress

needs-ok-to-test size/XXL

Projects

None yet

Milestone

No milestone

Development

Successfully merging this pull request may close these issues.

None yet

Notifications

Customize

Subscribe

You're not receiving notifications from this thread.

2 participants



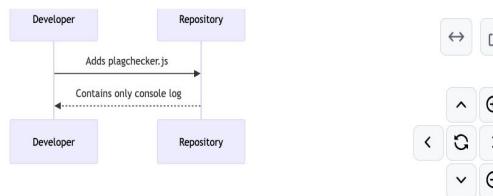
Google Summer of Code

Shubham Jain

Keploy

Code Review Summary

The submitted changes introduce a new file `plagchecker.js` that contains a single line of code to log a message indicating that the plagiarism checker has been initialized. While this is a straightforward addition, it lacks context, functionality, and adherence to best practices. The overall quality of this change is low, as it does not provide any meaningful implementation or error handling.



Critical Issues

- Lack of Functionality:** The current implementation does not provide any actual plagiarism checking logic, making it ineffective.
- No Error Handling:** There is no mechanism to handle potential errors or exceptions that may arise in a real plagiarism checking scenario.
- Logging Practices:** Using `console.log` for initialization messages is not suitable for production code. It should be replaced with a proper logging framework.

Code Quality Analysis

Security Concerns

- Information Leakage:** Logging sensitive information (if any) can lead to security vulnerabilities. Although the current log message is benign, future implementations should ensure that sensitive data is not logged.

Performance Issues

- Inefficient Logging:** Using `console.log` can lead to performance issues in production environments. It is advisable to use a logging library that can manage log levels and outputs efficiently.

Best Practices

- Code Structure:** The file should contain a module or class structure that adheres to the Single Responsibility Principle (SRP). Currently, it does not encapsulate any functionality.

▼ Click to expand issue table

Category	Issue Description	Location (File:Line)	Severity	Recommendation
Functionality	No actual plagiarism checking logic implemented	src/components/plagchecker.js	High	Implement plagiarism checking functionality.
Error Handling	No error handling mechanisms present	src/components/plagchecker.js	High	Add error handling for potential runtime issues.
Logging Practices	Use of <code>console.log</code> for logging initialization	src/components/plagchecker.js	Medium	Replace with a structured logging framework.
Code Structure	Lacks module or class structure	src/components/plagchecker.js	Medium	Refactor to follow SRP and include proper structure.
Documentation	No comments or documentation present	src/components/plagchecker.js	Low	Add comments explaining the purpose and usage.

Code Examples

Current Implementation

```
// src/components/plagchecker.js
console.log("Plagiarism checker initialized");
```

Issues:

- The code does not perform any actual functionality related to plagiarism checking.
- It lacks error handling and proper logging practices.

Suggested Improvements

```
// src/components/plagchecker.js
import { Logger } from 'some-logging-library'; // Hypothetical logging library

class PlagiarismChecker {
    constructor() {
        this.logger = new Logger();
        this.logger.info("Plagiarism checker initialized");
    }

    checkForPlagiarism(text) {
        try {
            // Implement plagiarism checking logic here
            this.logger.info("Checking for plagiarism...");
            // Placeholder for actual logic
        } catch (error) {
            this.logger.error("Error checking for plagiarism: ", error);
            throw new Error("Plagiarism check failed");
        }
    }

    export default PlagiarismChecker;
}
```

Improvements:

- Introduced a `PlagiarismChecker` class to encapsulate functionality.
- Added a method `checkForPlagiarism` for future implementation of plagiarism checking logic.
- Replaced `console.log` with a structured logging approach, allowing for better log management.
- Included error handling to manage potential issues during the plagiarism checking process.



Code Review Agent

1. Generate code reviews for PRs with open source models like Gemma 3

2. Can be run fully air-gapped

3. OOTB supports github.

Contributor: Gaurav



Performance and Security Testing

Simple and automatic:

1. Performance tests for APIs

- Request count and failure rates
- Response time percentiles (P90, P95, P99)
- Throughput (requests per second)
- Data transfer (bytes sent/received)
- Virtual user count over time

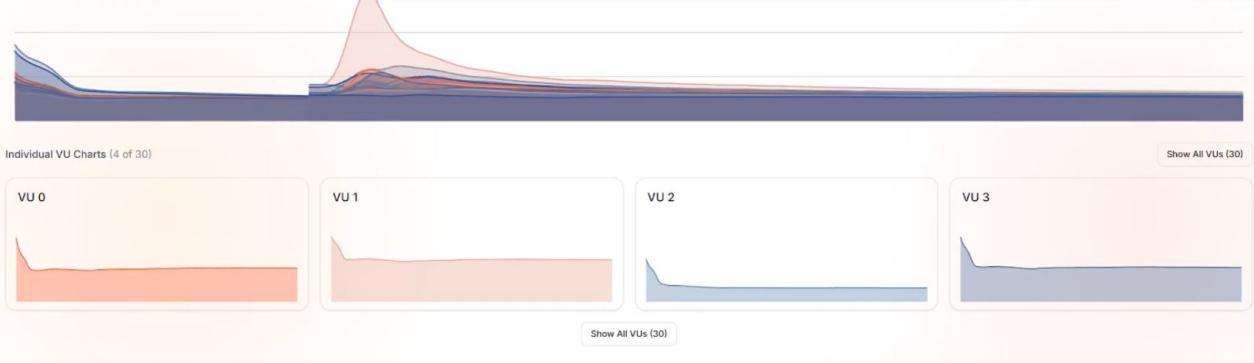
KLT Dashboard

Dashboard

Security Report

Security Summary

Total Checks: 60
Passed: 37
Failed: 23
Warnings: 0



2. Common security vulnerabilities analysis of APIs

- Vulnerability count by severity
- Check pass/fail status
- Custom rule evaluation results
- Allowlist filtering statistics

Dashboard

Security Report

Passed: 37
Failed: 23
Warnings: 0

Security Report

Test Suite: Todo_CRUD_Operations

60

Total Checks

37

Passed

23

Failed

0

Warnings

Report Generated

8/22/2025, 10:35:59 PM

Severity Summary

HIGH: 36 MEDIUM: 24

Security Check Results by Step

POST - Create_todo /todos

Passed: 6 Failed: 4 Warnings: 0

HTTPS Enforcement

Check if Strict-Transport-Security header is present

Check ID: https-enforcement

Status Code: 201

Target: Response

Details

Missing Strict-Transport-Security header in response

failed HIGH

Contributor: Ahmed Mamdouh



Dashboard for Modular Metrics Visualization

Centralized dashboard to view quality metrics across repos like test code coverage

Contributor: Swapnendu Banerjee

API Contract Testing

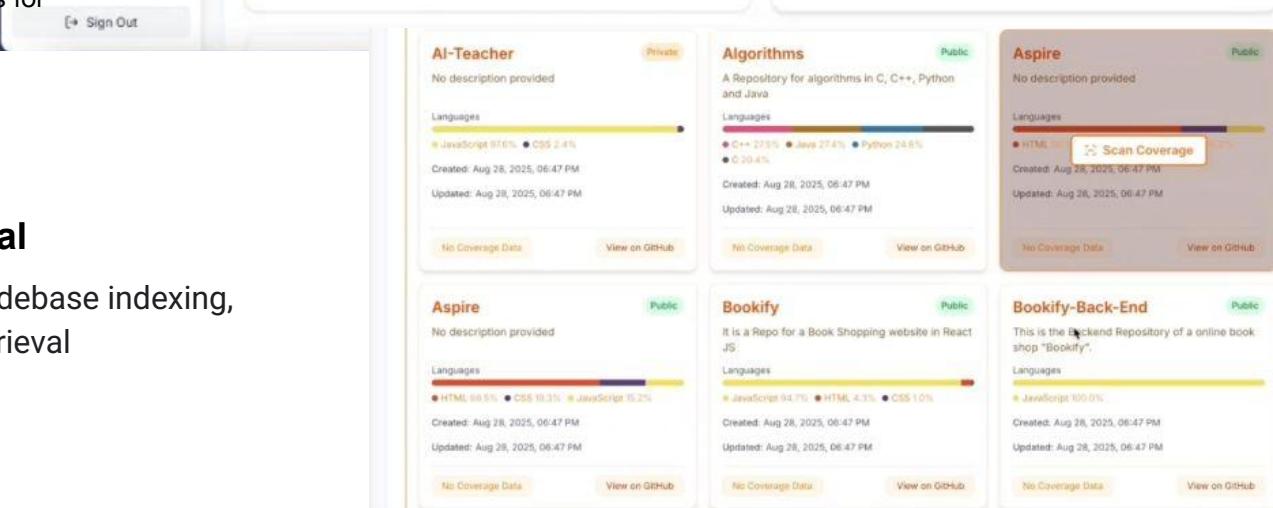
1. automatically generate client side contract tests for playwright tests
2. Implement server side contract testing

Contributor: Harsh Vashishtha

Code Indexer for Efficient Retrieval

GraphRAG system integrated including codebase indexing, dynamic index updates, hybrid context retrieval

Contributor: Sparshh





Google Summer of Code

Bawantha Thilan

OpenMRS - Performance Testing Enhancement Project

In 10 seconds... What is OpenMRS?



Open source Medical Record System



For low-resource settings



Maintained by a global community

Ranges

Stats

Users start rate

Global

Details

Concurrent users

Response time distribution

Response time percentiles

Requests / sec

Responses / sec

OpenMRSClinic



Gatling Version
Version: 3.14.3
Released: 2025-05-21

Run Information
Date: 2025-10-11 01:49:26 GMT
Duration: 3h 9m 55s
Description:
Executed on a GitHub Actions runner with 15Gi RAM, 4CPUs, and 72G disk space.

Assertions

Stats Fixed height Full size

Expand all groups

Collapse all groups

Requests ▲	🕒 Executions					⌚ Response Time (ms)							
	Total ▾	OK ▾	KO ▾	% KO ▾	Cnt/s ▾	Min ▾	50th pct ▾	75th pct ▾	95th pct ▾	99th pct ▾	Max ▾	Mean ▾	Std Dev ▾
All Requests	1007957	1007951	6	0	88.45	1	191	768	3473	56768	97596	1086	3651

How It Helped Developers

- **Data driven debugging.** no guessing.
- **Saves time**
- **Builds confidence**



Ganesh Devisetti



Google Summer of Code

Rudra Chopra

Eclipse Foundation SW360



Our Journey to Google Summer of Code 2025



Tried
2 years



Selected
2025



80+
Proposals



2 Projects



Improving
SW360-
FOSSology Integration
by Ritankar Saha



Improving Tests
for SW360 RESTAPI
by Rohit Borra

A proud start to our GSoC journey –
mentoring fresh talent and
strengthening SW360!



Enhancing REST API Test Coverage

- Added integration tests for 21 REST API areas across controllers
- Ensured multiple test scenarios per endpoint — success, invalid, empty, and auth/error



SW360 + FOSSology: Empowering Open Source Compliance



Together: FOSSology finds the facts. SW360 manages them.

SW360

Navigating an Enterprise-Scale
Open-Source Ecosystem

Challenges Faced



Complex Codebase & Architecture

Understanding SW360's modular and layered design.



Open-Source Workflow Mastery

Adapting to PR reviews, branching, and CI pipelines.



Integration & Testing Issues

Debugging multi-module test failures and API mismatches.

Key Takeaways



Technical Growth

Enhanced REST testing, CI/CD automation, and debugging precision



Code Quality Mindset

Delivered cleaner tests and maintainable Java refactoring



System Understanding

Deepened domain expertise and confidence in large-scale systems



Google Summer of Code

Diya Solanki

Processing Foundation



Processing Foundation

Our 13th year participating in GSoC!

>110 Projects since 2011

Organization Mission:

Our mission is to promote software learning within the arts, artistic learning within technology-related fields, and to celebrate the diverse communities that make these fields vibrant, liberatory, and innovative. Our goal is to support people of all backgrounds in learning how to program and make creative work with code, especially those who might not otherwise have access to tools and resources.

Project Contributor: Kamakshi Bali (mentored by Diya Solanki & Tristan Espinoza)

Enhance the developer and coding experience by adding IntelliSense features similar to those found in professional, multi-language IDEs, while continuing to support Processing's mission of making coding accessible to everyone.

<https://p5js.org/contribute/access/>



p5.js



Intellisense & User-Friendly Features in the p5.js Editor

A screenshot of the p5.js Editor interface. The code editor shows a sketch file named "sketch.js". The cursor is at line 3, column 11, where the word "draw" is being typed. A tooltip box appears below the cursor with the text "⚠️ use with caution in this context". Autocomplete suggestions for "draw" are shown in a dropdown menu, including "drawingContext" and "draw". The background of the code editor has a dark theme.

```
1 function setup() {  
2   createCanvas(400, 400);  
3   draw()  
4 } drawingContext var →  
5   draw fun →  
6   ⚠️ use with caution in this context  
7 }  
8  
9 function draw() {  
10   background(220);  
11 }
```

1. Context-Aware Autocomplete

A screenshot of the p5.js Editor interface. The code editor shows a sketch file named "sketch.js". The cursor is at line 12, column 11, where the word "Variable1" is being typed. A tooltip box appears below the cursor with the text "variable1". Renaming suggestions for "Variable1" are shown in a dropdown menu, including "Variable1", "Variable3", and "var". The background of the code editor has a dark theme.

```
1 let Variable1;  
2  
3 function draw() {  
4   background(220);  
5   const Variable2 = 1;  
6 }  
7  
8 function setup() {  
9   createCanvas(400, 400);  
10  var Variable3;  
11 } Variable1 var →  
12   Variable3 var →  
13   var keyword →
```

3. Context-Aware Renaming

A screenshot of the p5.js Editor interface. The code editor shows a sketch file named "sketch.js". The cursor is at line 10, column 11, where the word "bounce" is being typed. A tooltip box appears below the cursor with the text "bounce". Suggestion suggestions for "bounce" are shown in a dropdown menu, including "bounce". The background of the code editor has a dark theme.

```
1 class Ball {  
2   constructor() {  
3     this.speed = 5;  
4   }  
5   bounce() {}  
6 }  
7  
8 let b = new Ball();  
9 b.bounce()  
10 bounce fun →
```

2. User-Defined Code Suggestions

A screenshot of the p5.js Editor interface. The code editor shows a sketch file named "sketch.js". The cursor is at line 5, column 11, where the word "var1" is being typed. A tooltip box appears below the cursor with the text "var1". A callout box points from the word "var1" to a larger callout box containing the text "var2". The background of the code editor has a dark theme.

```
1 function setup() {  
2   createCanvas(400, 400);  
3   let var1 = 2;  
4   var1 → var2  
5 }  
6  
7 function draw() {  
8   background(220);  
9 }
```

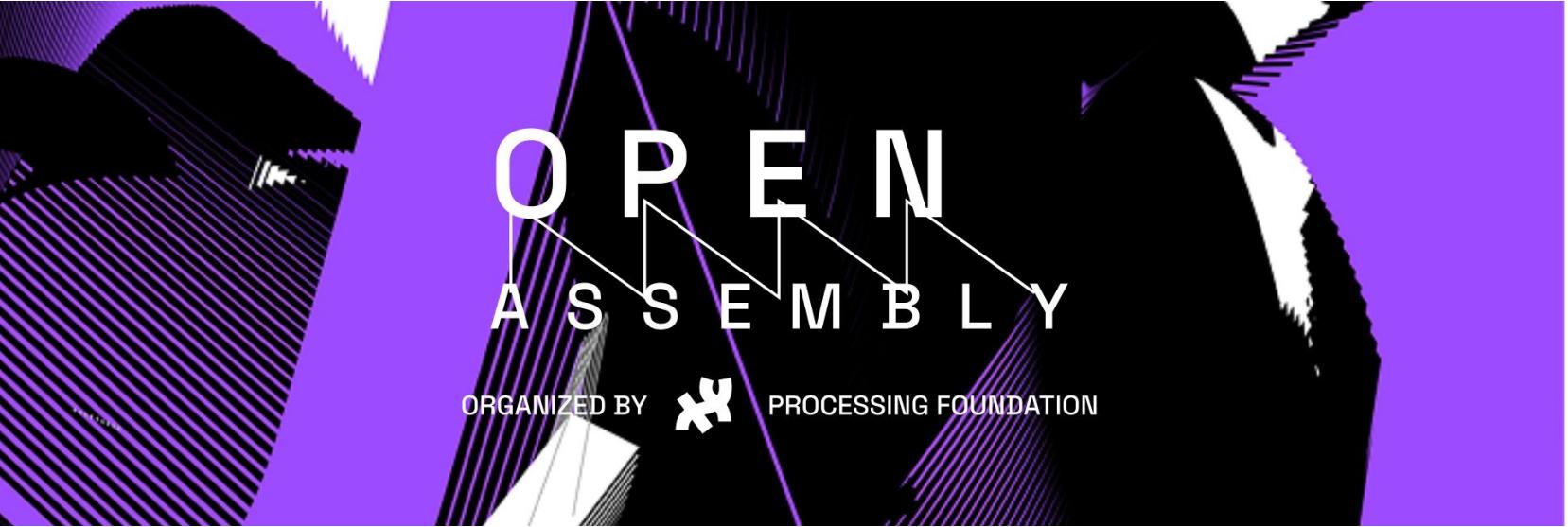
4. Jump to Definition

Accessibility Improvements:

- ARIA roles announce in Context-Aware renaming.
- Jump to Definition: Announced when a jump occurs.

- All actions can be triggered via keyboard shortcuts.





OPEN ASSEMBLY

ORGANIZED BY



PROCESSING FOUNDATION

All 3 PF GSoC 2025 projects will also be presented at the Open Assembly 2025.

Date: Wednesday, October 29, 2025 | **Time:** 1:00 - 3:30pm EST | **Format:** Free, Virtual

<https://openassembly.processingfoundation.org/>



Google Summer of Code

M Viswanath Sai

Prometheus Operator



Kubernetes

Declarative workload management

Define desired state, let K8s handle the rest



Prometheus

Collect metrics with granular control

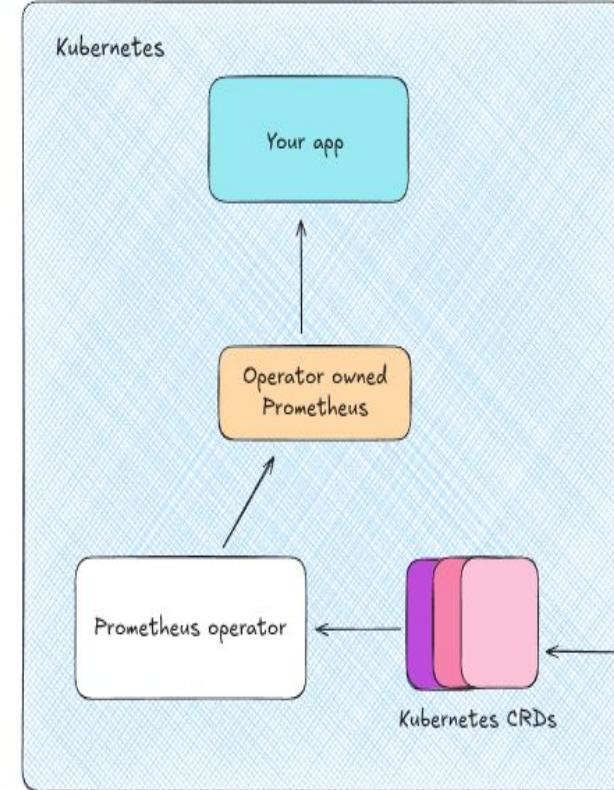
Not confined to Kubernetes deployments



Prometheus Operator

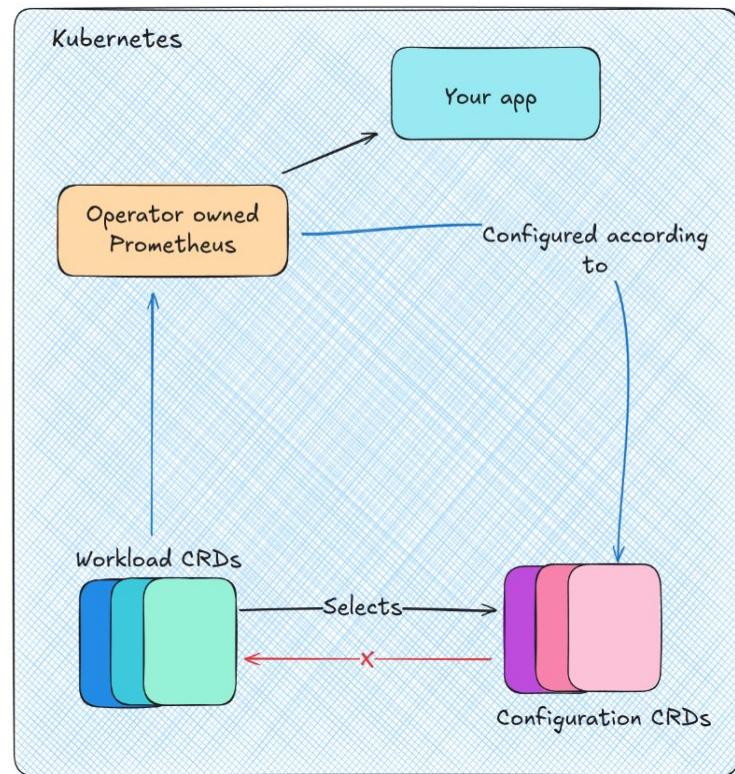
Deploy and manage Prometheus **the Kubernetes way**
Declarative configs for powerful monitoring automation within Kubernetes

A Simplified Birds-eye View

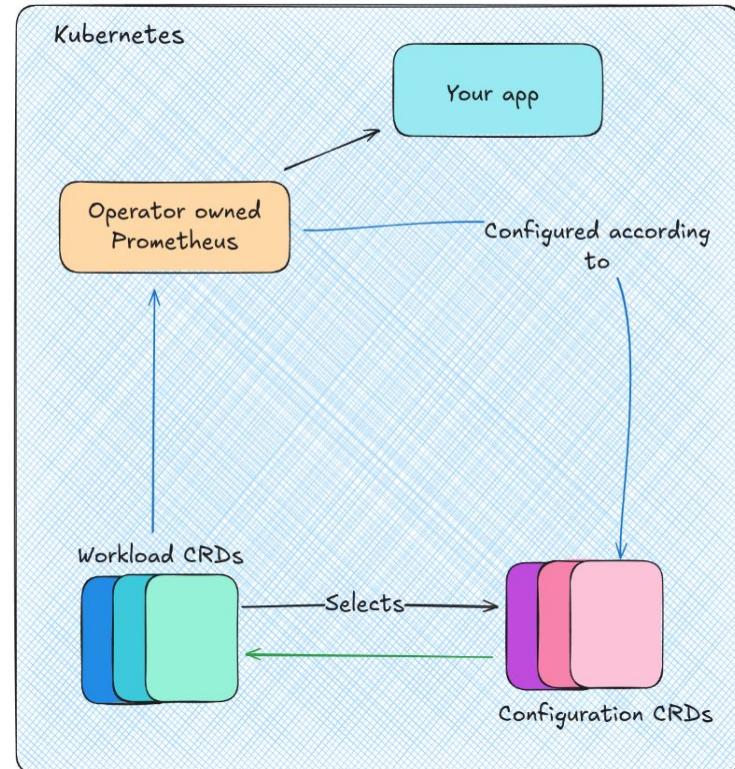


Status Subresource For Configuration Resources

Before



After

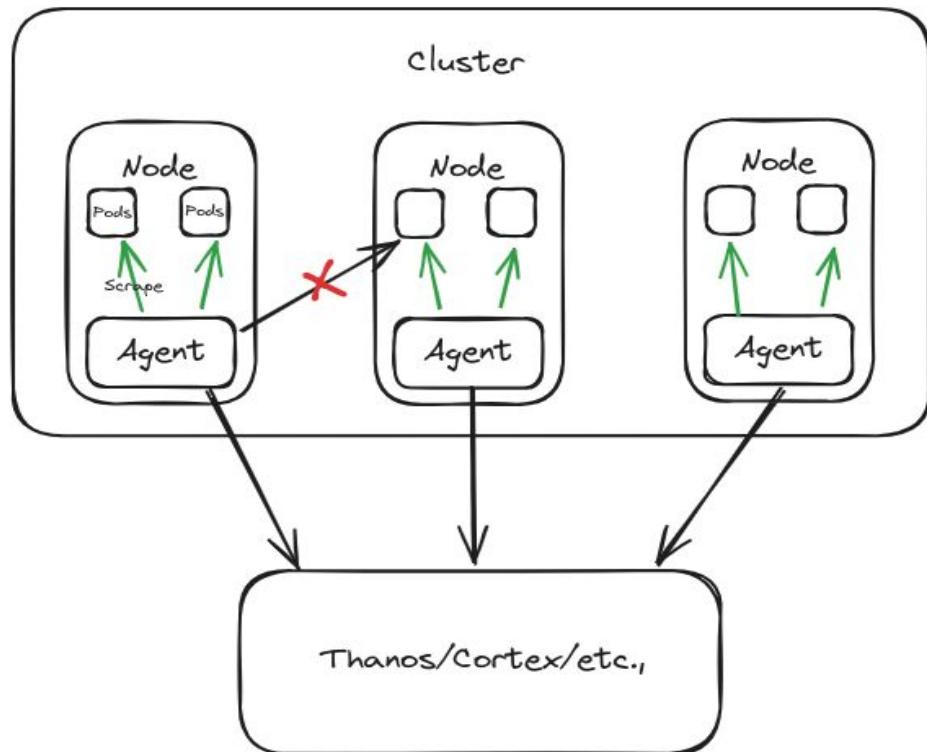


Prometheus Agent Daemonset Mode

Allows Automatic scaling

Suitable for very large clusters

Failure is node-wide





Google Summer of Code

Vedant Singh

Liquid Galaxy Project

What is Liquid Galaxy?



Liquid Galaxy is a remarkable panoramic system that is tremendously compelling. It started off as a Google 20% project created by Google engineer Jason Holt to run Google Earth across a cluster of PC's and it has grown from there!

The Liquid Galaxy Project works on extending the Liquid Galaxy system with open source software both improving its administration and enabling open source applications, so that content of various types can be displayed in the immersive panoramic environment.

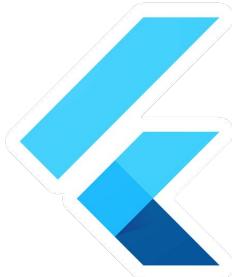


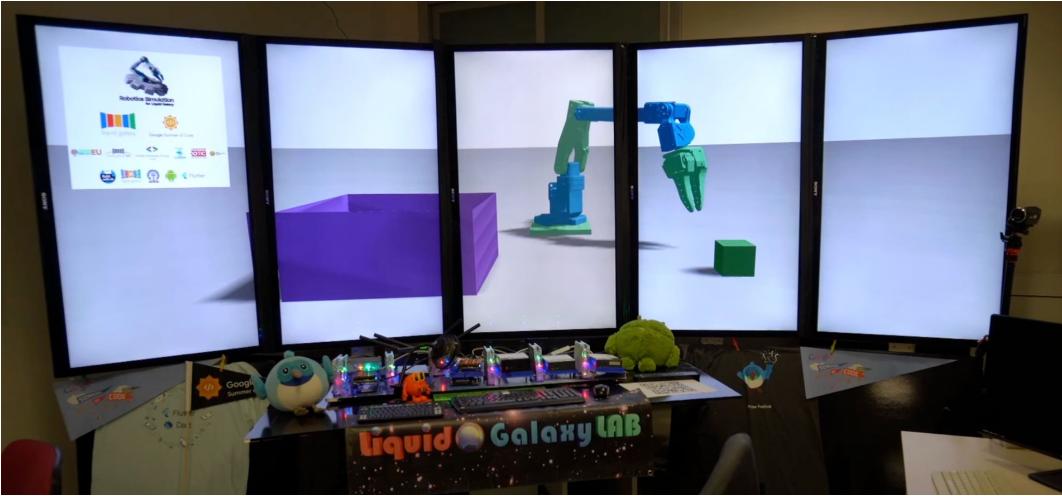
Robotics Simulation For Liquid Galaxy



GSOC 2025 Project by Debanjan Naskar

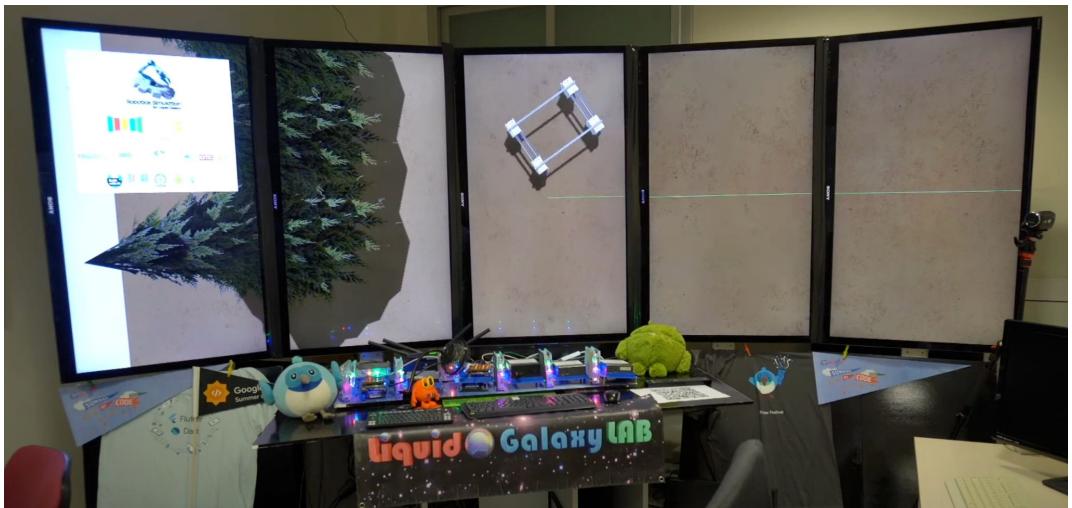
Imagine you're holding your phone and using it as a remote control for robots. But instead of watching on a tiny phone screen, you're standing in front of multiple large displays that show a panoramic, immersive view of what the robot sees and does. In simple terms, this project lets someone control virtual robots from their phone and watch them on the Liquid Galaxy Rig.





SO Arm 100 - An open source robotic arm which can also perform demo action in the simulation.

Amigabot Farm-ng - A movable platform to serve as a powerful agricultural robot.





Google Summer of Code

Ege Korkan

Eclipse Foundation - Thingweb Project

UI Components for IoT Developers

<https://github.com/eclipse-thingweb/ui-wot>

Tried to cover multiple aspects at once

- Interaction semantics
- Protocols
- Data structures
- Works with any UI framework

Leverages Thing Description standard: w3.org/wot

UI-WoT Components

Basic Slider Usage

Interactive Slider

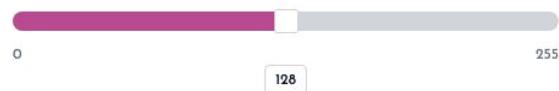
Smart Dimmer



```
<ui-slider  
    min="0"  
    max="100"  
    value="50"  
    variant="neon"  
    color="primary"  
    label="Smart Dimmer"  
    show-status="true"  
    show-last-updated="true">  
</ui-slider>
```

Mixed Examples

Color



```
<ui-slider variants="wide"  
    thumb-shape="square"  
    color="secondary">
```

Temperature



```
<ui-slider variant="neon"  
    thumb-shape="arrow"  
    min="-50" max="50">
```

Volume



```
<ui-slider variant="stepped"  
    step="10">
```

Disabled



```
<ui-slider state="disabled">
```

UI-WoT Generator

THINGWEB ui-wot < DASHBOARD DASHBOARD - 2 TDs loaded, 14 components Edit Add TD New Save ⚙

http-advanced-coffee-machine

All Available Resources

- water (integer)
- 53 +
- milk (integer)
- 99 +
- chocolate (integer)
- 100 +
- coffeeBeans (integer)
- 53 +

Save

Possible Drinks

- espresso, americano, cappuccino, latte, hotChocolate, hotWater

Maintenance Needed

Make Drink

Events (Out Of Resource) 0 events

Subscribe Clear

No events received

TestThing

Boolean

String unset

Events (Bool Property Change) 0 events

Subscribe Clear

No events received

Integer 42

Number 314

Object id (integer)
- 100 +
name (string)
abc

Save

Current Date 30/09/2025

Events (Int Property Change) 0 events

Subscribe Clear

No events received

Events (Num Property Change) 0 events

Subscribe Clear

No events received

+ - ↻

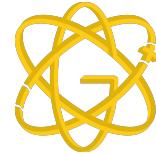


Google Summer of Code

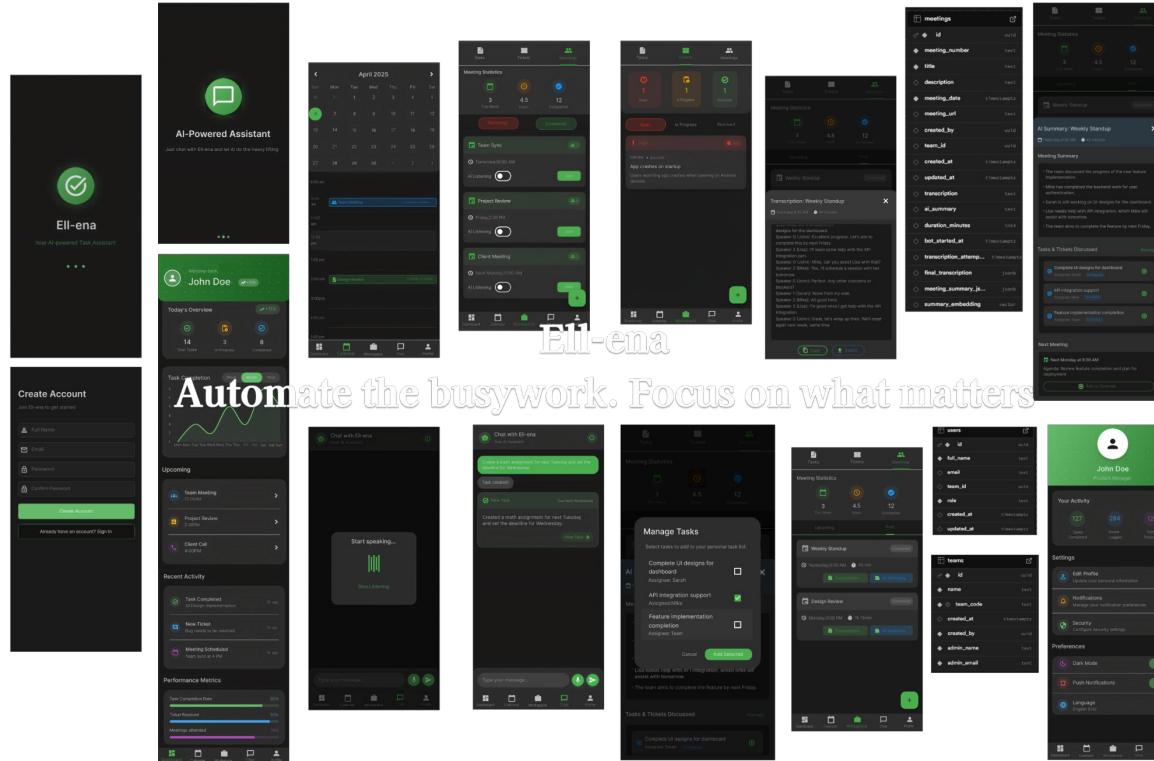
Jaideep Prasad

AOSSIE

What we do



Ell-ena : your AI Product Manager



Ell-ena : your AI Product Manager

- Automated Task Management: Your daily scrum assistant
- Context Capture: Video meetings / project notes
- Workflow automation: Anticipates needs and priorities



Google Summer of Code

Anastasia Hernandez-Koutoucheva

Wellcome Sanger Institute



Google Summer of Code

Vissarion Fysikopoulos

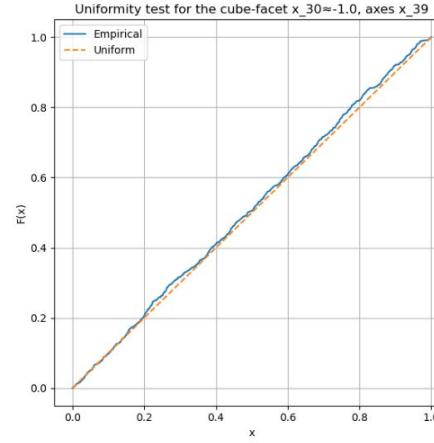
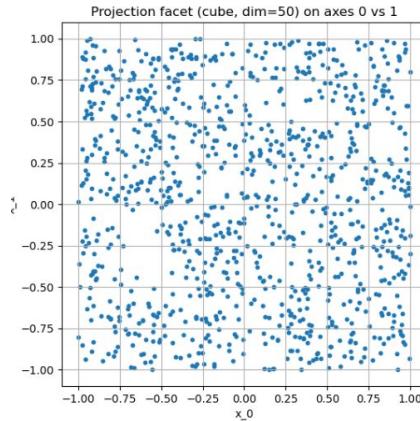
GeomScale

Shake and Bake - Sampling from the boundary of convex polytopes

- **Contributor:** Iva Janković (currently PhD student @ Edinburgh)
- **Goal:** Implement Markov Chain Monte Carlo-based Shake and Bake algorithm for uniform sampling from the boundary of a convex polytope
- **Motivation:** Fundamental problem in computational geometry & statistics + applications in metabolic network analysis
- **Results:** Implementations of new and classic algorithms in C++ with R and python interfaces

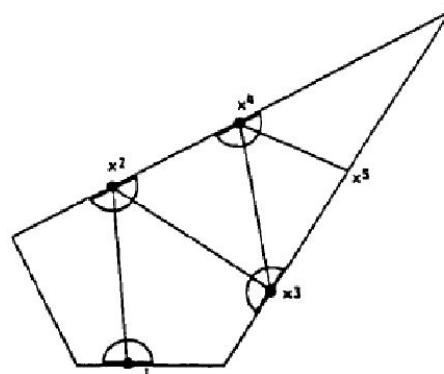
Challenges I

- Implement the basic version of 80's algorithm
- Much faster than existing implementations (`shakeandbake` R package)
- **Challenges**: evaluate the quality of samples
- **Solution**: implement a statistical test for each facet of the polytope

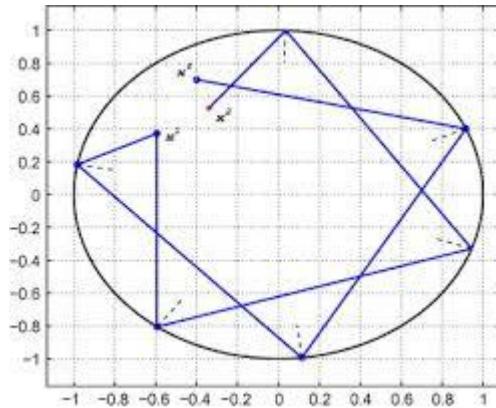


Challenges II

- Implement a new hybrid algorithm: **billiard shake and bake**
- Preliminary results show that it is much faster than the original one



+





Google Summer of Code

Andreas Dräger

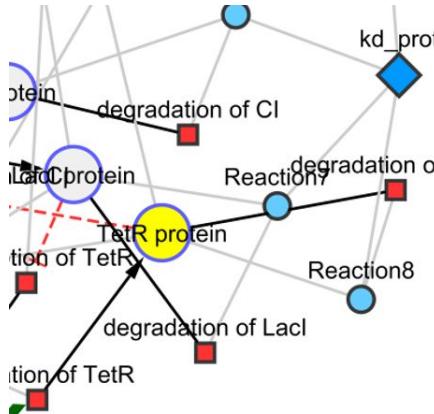
NRNB – the National Resource for Network Biology



Computational Models & Simulation of Biological Systems

Google Summer of Code projects in 2025

- **Better graphical editing of biological networks with Java™**
 - An update to the network editor CySBML: Smarter, Simpler, Automated
- **Better semantics in Biomodels with SBOannotator (Python):**
 - Coverage: Across 108 models, 3,317 reactions were upgraded from generic SBO:0000176 to more specific terms.
 - Efficiency: Mean processing time 432.99 s/model.
 - Specificity: Multi-database integration notably improves annotation specificity.
- **Better simulation with Java™**
 - A general linear optimization
 - A new ordinary differential equation solver



Universal Protein Resource using Persistent URL system
UniProt through NCBI

P04483 (TETR2_ECOLX)
Tetracycline repressor protein class B from transposon Tn10
Organism: Escherichia coli
Gene: tetR
Synonyms
Function TetR is the repressor of the tetracycline resistance element; its N-terminal region forms a helix-turn-helix structure and binds DNA. Binding of tetracycline to TetR reduces the repressor affinity for the tetracycline resistance gene (tetA) promoter operator sites



New Graphical User Interface of CySBML within Cytoscape

SBO-annotator

Types of reactions

- R1 Pseudo-reaction
- R2 Transport reaction
- R3 Transport reaction
- R4 Biochemical reaction
- R5 ...
- RX Pseudo-reaction

Pseudo-reactions

- Biomass production SBO:0000629
- Exchange SBO:0000627
- Sink SBO:0000632
- Demand SBO:0000628

Transport reactions

- No compartments = 1
- Passive SBO:0000658
- Active SBO:0000657
- Co-transport SBO:0000654
- Symporter SBO:0000659
- Antporter SBO:0000660

Annotated model

```
<cabal>
<model>
  <reaction> sboTerm="SBO:0000XXX"
  ...
  <reaction> sboTerm="SBO:0000XXX"
  ...
</species>
<species> sboTerm="SBO:0000XXX"
  ...
</model>
<sbo>
```

Further SBO Annotations

- Metabolites SBO:0000241 Cellular Compartments SBO:0000240
- Genes SBO:0000243 Kinetic Rate Law SBO:0000200
- Modelling Framework SBO:0000244 State Events SBO:0000213
- Subsystems SBO:0000245 SBO:0000174 SBO:0000223
- Subsystem Groups SBO:0000246 SBO:0000224
- Model Parameters SBO:0000213 SBO:0000236 SBO:0000246 SBO:0000246 SBO:0000246 SBO:0000246
- Isomerase SBO:0000377 Carbonyl SBO:0000319 Transferase SBO:0000402 Transaminase SBO:0000217 Glycosyltransferase SBO:0000403 Kinase & Phosphotransferase SBO:0000218 Ligase SBO:0000660
- Lyase SBO:0000178 Oxidoreductase SBO:0000220 Methyltransferase SBO:0000214 Acetyltransferase SBO:0000215 Hydrolase SBO:0000233 Deaminase & Deminase SBO:0000376 SBO:0000401

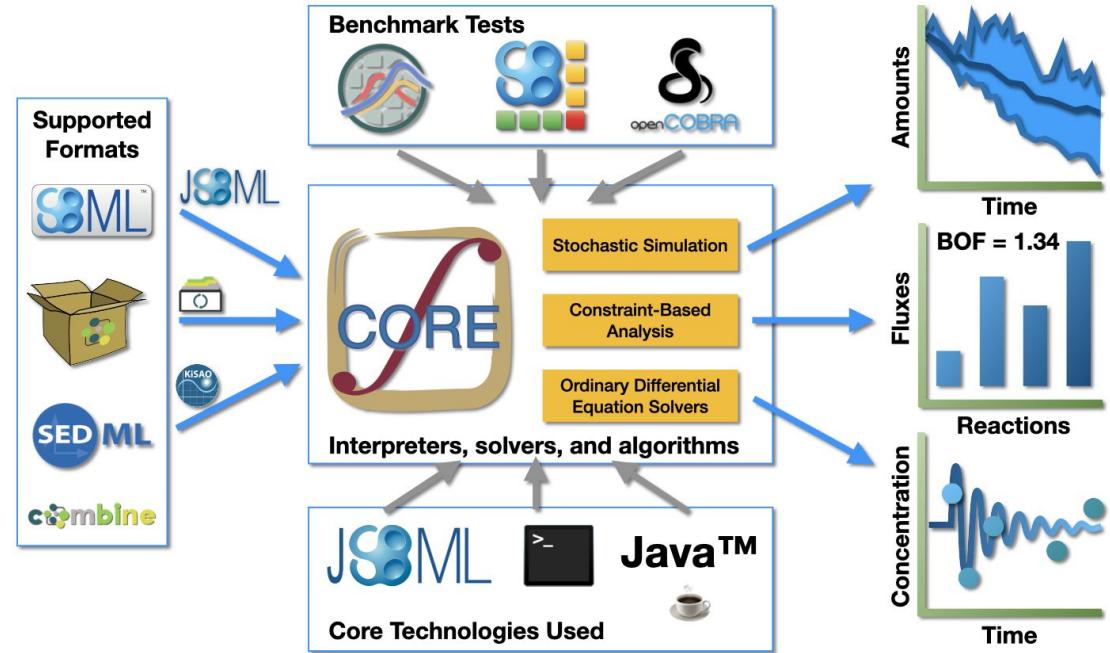
Dynamic and Steady-State Models of Metabolism

- New algorithm LSODA
 - Numerical solver for ordinary differential equations
- Separate project OptSolvX
 - abstraction layer between linear solvers and optimization problem
 - runs FBA on x86 & ARM
 - choose best suited backend (e.g., CPLEX, Gurobi, Apache)

SBSCL



OptSolvX



The Simulation Core Library (SBSCL) reads Biomodels in SBML format and performs various simulations

Our International Team at NRNB

Our Students



Hikmet Emre Kaya



Michael Gaas



Arthur Neumann



Ayush Baranwal



Jiahui Hu Neu



The Mentors



Taichi Araki 🇯🇵 coding
with Arthur in Stuttgart



Prof. Akira Funahashi



Dr. Matthias König



Dr. Nantia Leonidou



Prof. Andreas Dräger



STOP

STOP!

The rest of the slides are for later!

Speaker Order - Friday 4:00-5:00PM

Order	Mentor Name(s)
1	Antonio Jimenez
2	Marc Gonzalez
3	Victor Cuadrado
4	Nikhil Agarwal
5	Zeeshan Rafique
6	Turan Furkan TOPAK
7	Kannupriya Kalra
8	Atharva Arya
9	Mohit P. Tahiliani
10	Peter Dudfield

Order	Mentor Name(s)
11	Dmitriy Pastushenkov
12	David Pascual
13	Lorena Goldoni
14	Felipe Borges
15	Till Kamppeter
16	Christoph Badura
17	Renan Matsuda
18	Sreeja Kamishetty
19	Yuriko Kikuchi

Waitlist for Lightning Talks

1. Sarrah Bastawala - Slides added
2. Oliver Kopp - JabRef e.V. (slides already in)
- 3.
- 4.
- 5.
- 6.

Lightning Talk Slides

This shared deck will be where you save your completed slides.

To keep to Google's brand guidelines, a few reminders when creating slides:

- Please use a white background
- No other company logos (example: twitter logo), memes or pictures of people may be used unless you have explicit permission to use them
- Use the [template](#) for your title slide and then add up to 3 additional slides
- Your presentation should be about the GSoC projects this year for your org, not about what your org does. You can spend maybe 15 seconds briefly discussing what your org does for context but the focus should be on the GSoC 2025 projects.

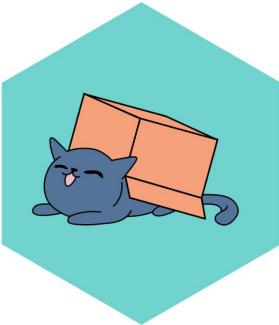
Slides are due no later than October 22nd



Google Summer of Code

Antonio Jimenez

Typelevel



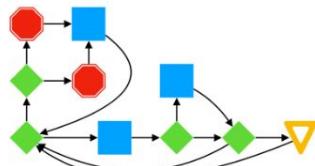
Cats Effect: multi-threaded I/O runtime

work stealing

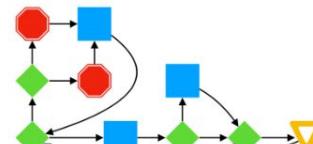
work stealing

work stealing

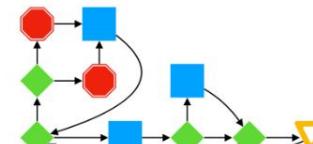
Thread 1



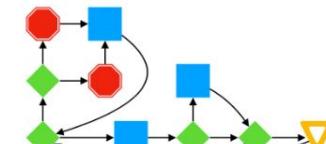
Thread 2



Thread 3



Thread 4



Benchmark polling system powered Ember and friends #3692

armanbilge opened this issue on Jun 14 · 11 comments

Open

armanbilge commented on Aug 28 • edited

For folks following along here: we've also published a snapshot of [@antoniojimeneznieta](#)'s work implementing a JVM polling system based on io_uring. The initial prototype piggy-backs on Netty's internal io_uring APIs.

Please give it a try! The linked PR demonstrates how to create a JVM Ember server using the fs2-io_uring snapshot.

- [Implement UringSystem and FS2 Sockets using netty io_uring API armanbilge/fs2-io_uring#78 \(comment\)](#)

wjoel commented on Aug 29

I ran the TFB benchmarks with my branch using your latest and greatest snapshots last weekend, and it wasn't worth writing about, slightly worse results than before. Well, turns out I didn't notice that `withSocketGroup(UringSocketGroup[IO])` in your example, and it makes quite the difference...

Baseline: <https://www.techempower.com/benchmarks/#section=test&shareid=9b70928b-24e8-4a39-a5dc-7832d8b02cd6&test=plaintext>
fs2_iouring and also withSocketGroup(UringSocketGroup[IO])
<https://www.techempower.com/benchmarks/#section=test&shareid=35-b366-12dbeb063575&test=plaintext>

On "meaningless" benchmarks you've made Ember app real-world improvements as well. Whereas before, my C those benchmarks, they're now blazing along at 100%, host 30% loaded when running

real money that it translates into
340% faster,
or vCPU core

Fantastic. Amazing. Well done.

7 8

Added Processes to IONative #3575

Open rahulrangers wants to merge 31 commits into [typelevel:main](#) from [rahulrangers:native-process](#)

Conversation 29 · Commits 31 · Checks 16 · Files changed 5

rahulrangers commented on Jun 4

Implemented ProcessBuilder-based process spawning for Scala Native using posix_spawn, with support for stdin, stdout, and stderr piping. Integrated pidfd_open and fileDescriptorPoller to asynchronously and non-blocking wait for process termination. Falls back to waitpid when pidfd_open is unavailable.

Added DatagramSocket for Native with Polling #3582

Closed rahulrangers wants to merge 56 commits into [typelevel:topic/net2](#) from [rahulrangers:datagram-socket](#)

Conversation 2 · Commits 56 · Checks 16 · Files changed 7

rahulrangers commented on Jul 13 · edited

Add Datagram Socket Support with Polling

This PR adds support for datagram sockets with polling for native. The key changes include:

- `FdPollingDatagramSocket.scala`: Defines an interface for datagram sockets.
- `FdPollingIpDatagramSocketsProvider.scala`: Creates and binds IP-based datagram sockets.
- `IpMulticast.scala`: Provides types for low-level multicast support (`ip_mreq`, `ip_mreq_source`, etc.).
- `SocketHelpers.scala`:
 - Adds `join`, `drop`, `block` and `unblock` helpers for `multicast`.
 - Adds new `socketOptions` for `datagramSockets`.
 - Adds `setsockopt` and `getsockopt`.

Add Cats-Effect selector-based Datagramsocket implementation for JVM #3589

Closed rahulrangers wants to merge 9 commits into [typelevel:topic/net2](#) from [rahulrangers:jvm-datagram](#)

Conversation 0 · Commits 9 · Checks 16 · Files changed 3

rahulrangers commented on Aug 19

This PR introduces a non-blocking, Selector-driven implementation of UDP sockets for the JVM.

Key Changes

New implementation

- `SelectingIpDatagramSocketsProvider`
Binds UDP sockets backed by a `(DatagramChannel)` integrated with `cats.effect.Selector`.
- `SelectingDatagramSocket`
Provides a `DatagramSocket[F]` implementation with mutex-guarded non-blocking I/O for read and write operations, supporting:
 - Asynchronous reads and writes via `Selector`.
 - Standard socket options (`getOption`, `setOption`)
 - Multicast group membership (`join`, `drop`, `block`, `unblock`)

Add awaitEvent to expose the full featureset of kqueue #4423

Merged armanbilge merged 11 commits into [typelevel:series/3.x](#) from [rahulrangers:kqueue](#)

Conversation 20 · Commits 11 · Checks 38 · Files changed 1

rahulrangers commented on Jun 12

Added a `kqueue` trait with an `awaitEvent` method to enable direct kevent handling in the `kqueue`-based polling system. This provides access to `kqueue`-specific functionality beyond standard file descriptor polling.

armanbilge reviewed on Jun 13

armanbilge left a comment

Awesome work!! Because this PR adds a new feature (instead of a bug fix) it should be targeted to `series/3.x`.



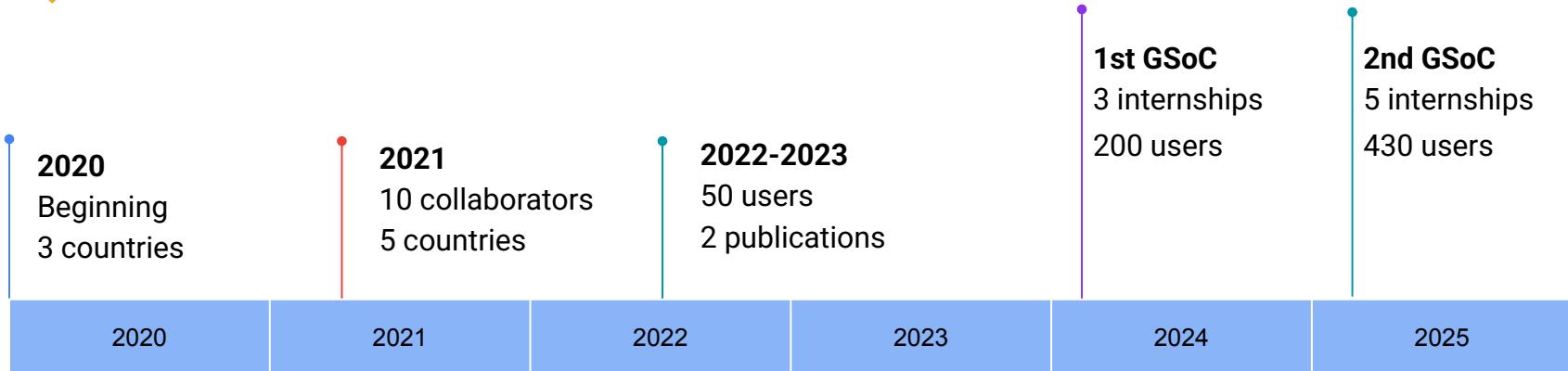
Google Summer of Code

Marc Gonzalez

Uramaki LAB



Uramaki LAB



960 Commits!

1759 Commits!

⭐ Stars: 174

🔗 Forks: 306

👥 Contributors: 111

🔀 PR: 419

📈 Issues: 602

📣 Commits: 6568

RUXAILAB

The UX laboratory powered by AI

Open-source platform for cognitive UX testing

[Get Started](#)[GitHub](#)

Remote UX Evaluation

Conduct usability testing entirely online with the aim of artificial intelligence algorithms such eye tracking, sentimental analysis, transcription, etc.



Multimodal Usability Methods

Expand beyond testing with focus groups, cognitive walkthroughs, card sorting, and future mixed methods.



Accessibility by Design

Build inclusive interfaces from the ground up with adaptive UIs and support for WCAG compliance.



Open Science & Community

RUXAILAB is 100% open-source, reproducible, and developed by an international research-driven community.

Category

Methods

Study Type

Details

1

2

3

4

Choose Evaluation Category

Select the type of evaluation you want to conduct for your study



Test

Conduct controlled testing with real users to measure usability and performance.

Multiple Methods



Inquiry

Gather insights through surveys, interviews, and other research methods.

Coming Soon



Inspection

Expert evaluation using established usability principles and guidelines.

Multiple Methods



Accessibility

Assess your product for accessibility compliance and best practices.

Coming Soon



Google Summer of Code

Victor Cuadrado

Kubewarden (CNCF)



KUBEWARDEN



CLOUD NATIVE
COMPUTING FOUNDATION

- CNCF Sandbox project
- K8s Policy Engine, production ready
- Policy As Code: **Wasm** shipped as OCI artifacts
 - OCI registry
 - policy-server
 - K8s controller
 - kwctl CLI utility
 - Language SDKs

We want to have this javascript SDK as a high quality TypeScript library that is a first citizen in the Kubewarden ecosystem.
 @esosaoh will be mainly working on this task.

For that, we need to do the following:

- Have a Continuous integration GitHub job that runs on each PR and gates PRs to this repo [Add Continuous Integration to the project #121](#)
- Implement the host capabilities. The host capabilities such provide caching as needed, and logging. These are, as listed in our [reference docs](#):
 - Sigstore capabilities, [v2/verify](#) (we can skip `v1/verify`). This includes the following inputs:
 - SigstorePublicKeyVerify
 - SigstoreKeylessVerify
 - SigstoreKeylessPrefixVerify
 - SigstoreGitHubActionsVerify
 - SigstoreCertificateVerify
 - Implement Sigstore host capabilities #201
 - OCI capabilities. This includes:
 - v1/manifest_digest [Feature request: Implement v1/manifest_digest OCI host capability #91](#)
 - v1/oci_manifest [Implement v1/oci_manifest OCI host capability #123](#)
 - v1/oci_manifest_config [Implement v1/oci_manifest_config OCI host capability #124](#)
 - Network capabilities (`v1/dns_lookup_host`). Was done as initial POC.
 - Cryptographic capabilities:
 - v1/is_certificate_trusted [Implement Crypto Host Capability #189](#)
 - Kubernetes capabilities. This includes:
 - list_resources_all
 - list_resources_by_namespace
 - get_resource [Feature request: Implement kubernetes/get_resource capability #135](#)
 - can_1 (added in the near future, see [here](#)) [Host capability to check service account permissions #116](#)
 - Refactor code as needed to:
 - Have a Typescript library and javy plugin that can be imported and used when creating a policy written in Typescript/Javascript. Export the host capabilities.
 - Move demo policy to its own repo. [Remove demo policy from the SDK directory #122](#)
 - Implement caching for the needed capabilities [Add caching to get_resource host capability #120](#)
 - (optional) Document the library functions with JSDoc and publish them somewhere (e.g. with gh-pages). [Generate and publish JSDoc in gh-pages #165](#)
 - Document library usage with a tutorial [docs: Add tutorial for policy-sdk-is #190](#)
 - Have a Continuous Delivery GitHub job that runs on each git tag pushed to the repo, and makes a GH release [Add Continuous Delivery, do first alpha release #148](#)
 - Blog post on [kubewarden.io/blog](#)
 - Announce post (in Slack, Mastodon, Bluesky).
 - Create a policy template to consume lib from npm: [Create policy-template-is policy repo to test the consumption of the library from npmjs #153](#)
 - If needed, create a policy demo
 - (optional) Add reusable GH reusable workflows to publish.js policies into [github.com/kubewarden/github-actions](#)

For each host capability, we should provide e2e tests, unit tests, cache if needed (see other SDKs), logging, JSDocs.

Close [Allow policies to be written using JavaScript community#371](#)

Create sub-issue

>— vccuad assigned esosaoh on Jul 2

- Typescript/Javascript SDK by Esosa Ohangbon

<https://github.com/kubewarden/policy-sdk-is>

- July 17,
- K8s
- Wasm with Javy
- Wasm Host/Guest APIs in JSON
-





Google Summer of Code

Nikhil Agarwal

Oppia Foundation

Oppia Foundation



+ = ?

1 $\frac{3}{4}$

$\frac{5}{6}$

2 $\frac{1}{2}$

Acceptance Tests

- Cover all Critical User Journeys using acceptance testing.
- Reduce manual testing.
- Remove flakiness from the current test suites.
- Complete migration from E2E tests to Acceptance tests.



Google Summer of Code

Zeeshan Rafique

Mircroelectronic Research Lab (MERL)

About MERL

MERL is a research lab at Usman Institute of Technology University (UITU) in Karachi, Pakistan.

- The lab focuses on microelectronics using the open-source RISC-V architecture, bridging hardware and software to give undergrad students hands-on chip design and “tape-out” experience.
- MERL has participated in [Google Summer of Code](#) for the past four consecutive years.
- GitHub: <https://github.com/merledu>



MERL GSoC Projects 2025

This year, MERL got five slots.

1. Integrating TCAM IP into Chipyard via MMIO and RoCC Interfaces
2. Automated UVM Tuning Orchestrated Via AI Reasoning
3. AEGIS - AI-Enhanced Generation of Intelligent Scenarios
4. Running Secure and Vectorized Applications on SoC-Now
5. Generic Test Bench for RISC-V CPUs Using PyUVM

GSoC Org: <https://summerofcode.withgoogle.com/programs/2025/organizations/micro-electronics-research-lab-uitu>

My Project: Generic TB for RISC-V Cores

The project is about creating a generic test bench for RISC-V CPUs.

- RISC-V is an open source Instruction Set Architecture (ISA).
- Written in PyUVM, means no commercial compiler needed.
- Easy to integrate with any RISC-V CPU using RVFI interface.
- Spike is used as a golden model, which is open source and written in cpp.
- Compares the result of each executed instruction with golden model.
- Report mismatch immediately (exception / write back data / load store).
- Generate execution log for debugging.

GitHub Repo: <https://github.com/merledu/coco-rvtb>

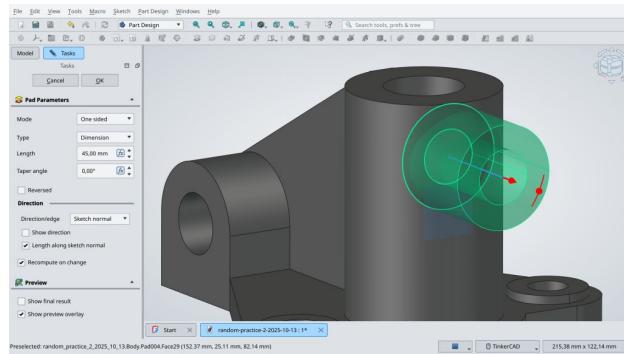


Google Summer of Code

Turan Furkan TOPAK

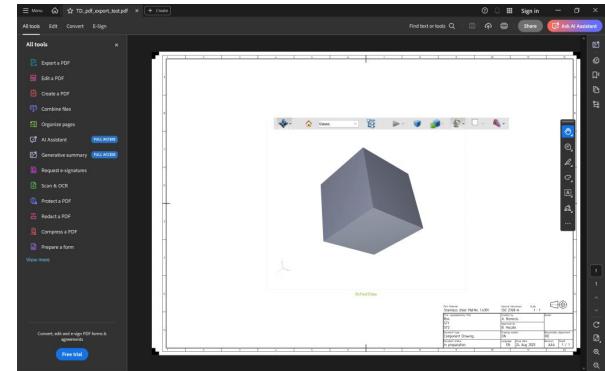
FreeCAD

The Story of 3 Projects



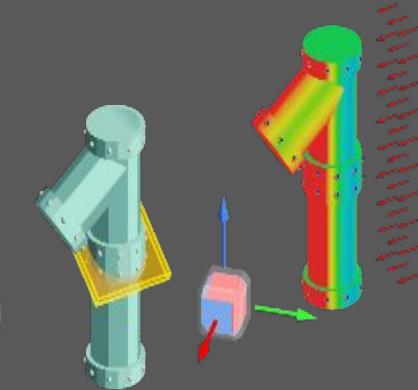
Interactive controls in
the 3D View

Multiple files parallel
editing



3D PDF Export in FreeCAD

FreeCAD





Google Summer of Code

Kannupriya Kalra

Scala Center





Google Summer Of Code Mentor Summit 2025



LLM4S x GSoC 2025: ENGINEERING GENAI AGENTS IN FUNCTIONAL SCALA

Thursday, October 23 - Saturday, October 25th

@Munich Marriott Hotel, Hosted by Google,
Germany 🇩🇪 (live in person!)



Kannupriya Kalra

The Key Outcome: Traceability Unlocked

- **Tracing Support (PR #77, #119):** Added execution flow tracking, spans, and timing. Integrated **Langfuse** for developer-friendly traces.
- **Type-Safe Tracing (PR #165):** Rebuilt with **sealed traits & composable backends**. Ensured reliability and backward compatibility.
- **Demo & Multi-Backend (PR #184):** Built a calculator agent for sequential tool-use. Showcased tracing with Console, Langfuse, and NoOp.



Shubham Vishwakarma

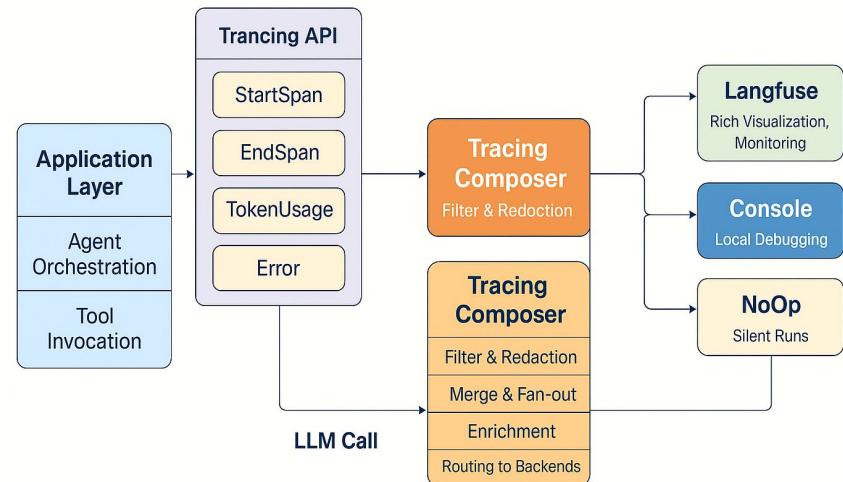
Total PRs Raised	PRs Closed / Merged	PRs Still Open
20	18	2

How It Works: The Architectural Clarity

LLM4S Tracing Architecture - High-Level Detailed View:

- **Diagram Flow:** Application Layer (Agent Orchestration/Tool Invocation) → Tracing API (StartSpan, EndSpan, TokenUsage, Error) → Tracing Composer → Backends (Langfuse, Console, NoOp)
- **Tracing Composer functions:** Filter & Redaction, Merge & Fan-out, Enrichment, Routing to Backends.

LLM4S Tracing Architecture – High-Level Detailed View



The Foundation for Future AI Systems

Beyond Tracing: The Next Chapter for LLM4S:

- **Multi-Agent Systems In Progress:** DAG-based planner, Pub-sub signals, Scala-native, type-safe orchestration.
- **A Real Use Case: Meeting Minutes Generator**
Planned → Full traceability → every step tracked with spans/events.



LLM4S Project Link



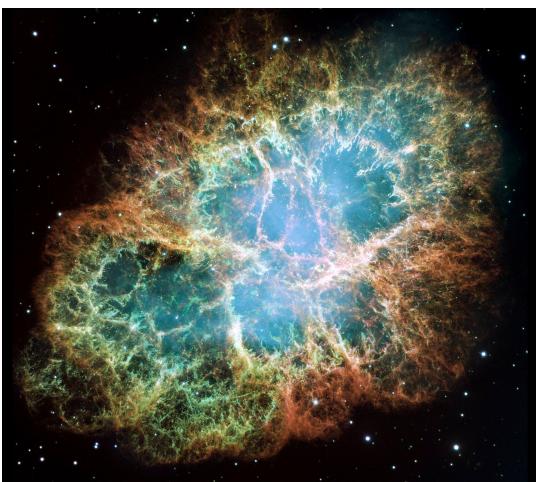
Google Summer of Code

Atharva Arya

TARDIS



1994D in galaxy NGC 4526



SN 1054 (1054 AD), Milky Way!

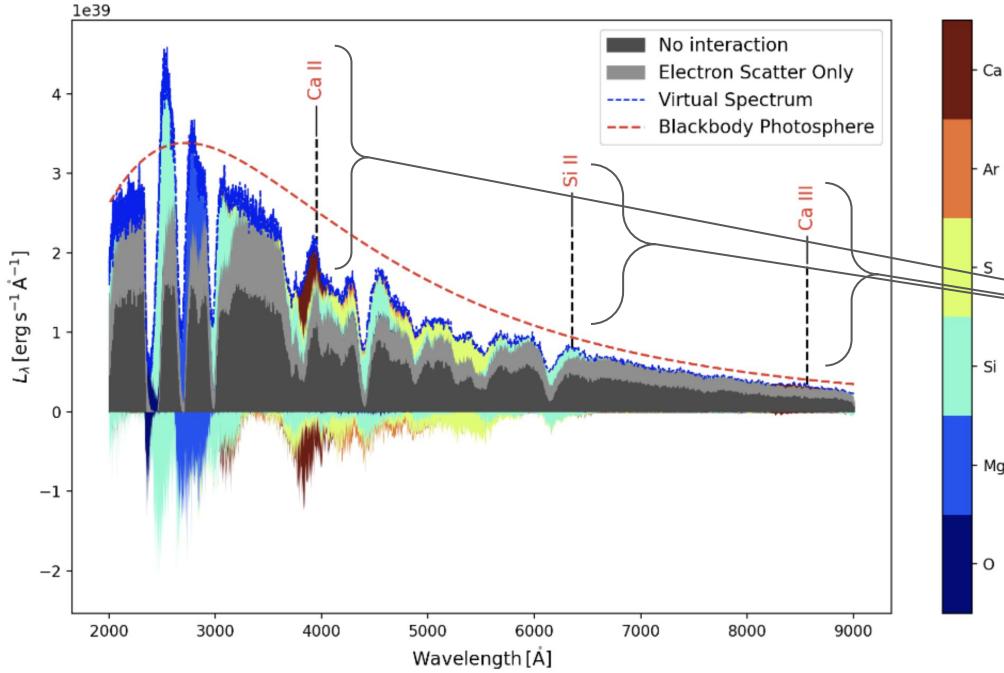


TARDIS is an open-science collaboration that develops *python* packages to simulate and analyse exploding stars

More technically, it's a Monte Carlo radiative transfer code that calculates theoretical spectra for supernovae and other transients

36+ GSoC projects completed since 2013 (12 years!)

This year, 2 projects completed!



Haille also restructured the formal integral and its source function calculation to improve maintainability.

The restructure also laid the foundation for different code acceleration methods like Jax, for while Haille did initial investigation.



Haille's work

Allows researchers to **mark** which **atomic species** are responsible for **specific spectral features** in supernova spectra

```
line_waves = [3951, 6355, 8567]
line_labels = ['Ca II', 'Si II', 'Ca III']
```

30 tardis/opacities/tests/test_opacity_solver/test_new_macro_atom_solver__downbranch-False__.h5
31 tardis/opacities/tests/test_opacity_solver/test_new_macro_atom_solver__downbranch-True__.h5
32 tardis/opacities/tests/test_opacity_solver/test_new_macro_atom_solver__macroatom-False__.h5
33 tardis/opacities/tests/test_opacity_solver/test_new_macro_atom_solver__macroatom-True__.h5
34 tardis/opacities/tests/test_tau_sobolev/test_calculate_beta_sobolevs.npy
35 tardis/opacities/tests/test_tau_sobolev/test_calculate_sobolev_line_opacity.h5

Files 2cbdea-6483ea 77c993-2cbdea

M M
M M
M M
M M
• •
M M



The Regression Data package ensures each TARDIS commit produces **exactly the same** scientific output.

Riddhi's code runs the test suite over a **series of commits** to find out when **which regression data files changed and where**.

Riddhi also investigated the TARDIS testing framework carefully and pointed out places where tests could be improved.



Google Summer of Code

Mohit P. Tahiliани

The ns-3 Network Simulator Project

List of Successful Projects (%)

- Upgrade the AQM Evaluation Suite for ns-3
 - Mentee: David Lin, Canada
 - Mentors: Mohit P. Tahiliani, Tom Henderson and Aniket Singh
- 6LoWPAN Neighbor Discovery Protocol
 - Mentee: Boh Jie Qi, Singapore
 - Mentors: Tommaso Pecorella and Adnan Rashid
- IPv6 Global Routing
 - Mentee: Shashwat Patni, India
 - Mentors: Tommaso Pecorella and Manoj Kumar Rana
- NTN Helper and a LEO example for 5G NR
 - Mentee: Thiago Miyazaki, Brazil
 - Mentors: Gabriel Ferreira, Biljana Bojovic, Amir Ashtari Gargari, and Katerina Koutlia

Upgrade the AQM Evaluation Suite for ns-3

- **Student:** David Lin, University of Toronto, Canada
- **Mentors:** Mohit P. Tahiliani, Tom Henderson and Aniket Sing.
- **Project Goal:** Upgrade AQM Evaluation Suite for ns-3 to match requirements of RFC 7928
- **Milestones**
 - **Waf to CMake Migration:**
 - Migrated the AQM Evaluation Suite's build system from Waf to CMake
 - **TCP and ECN Enhancements:**
 - Upgraded the configuration defaults, including replacing TCP BIC with TCP CUBIC
 - **Support newer AQM algorithms:**
 - Added support for new algorithms, such as: COBALT, FQ-PIE
 - **Packaging:**
 - Module packaged and added to the ns-3 app store!

Launched on ns-3 app store! available with ns-allinone-3.45

AQM Evaluation Suite for ns-3

An Automated Framework to Evaluate ns-3 Queue Disciplines

GET STARTED



RFC 7928 Compliant

The automated test cases provided in this project are in line with those mentioned in RFC 7928. RFC 7928 is an Informational RFC that provides a set of guidelines to characterise AQM algorithms.



Easy to Use

This project automates the cycle of simulation setup, topology creation, traffic generation, program execution, results collection and graphical representation (as recommended in RFC 7928).



Easy to Modify

The procedure to add a new AQM algorithm (such as the one designed by you) to this suite and compare its performance with others is very simple, provided the new algorithm is implemented in ns-3.

Available Scenarios

RFC Section	Scenario Name	Description
5.1.1	TCPFriendlySameInitCwnd	TCP flows with identical initial congestion windows
5.1.2	TCPFriendlyDifferentInitCwnd	TCP flows with varying initial congestion windows
5.2	AggressiveTransportSender	Single aggressive TCP flow (CUBIC variant)
5.3.1	UnresponsiveTransport	Single UDP flow without congestion control
5.3.2	UnresponsiveWithFriendly	UDP flow competing with TCP traffic
8.2.1	MildCongestion	Light network congestion scenario
8.2.2	MediumCongestion	Moderate network congestion scenario
8.2.3	HeavyCongestion	Heavy network congestion scenario

Configuration Options

Parameter	Description	Default
--number	Run scenario by RFC section number	-
--name	Run scenario by name or "All"	-
--QueueDiscMode	Queue discipline mode (PACKETS/BYTES)	PACKETS
--isBql	Enable Byte Queue Limits	false
--ecn	Enable Explicit Congestion Notification	false
--BaseOutputDir	Output directory path	aqm-eval-output



Google Summer of Code

Peter Dudfield

Open Climate Fix



Open Climate Fix

01

- » Founded in 2019, in the UK
- » Non Profit product lab developing open source AI solutions to decarbonise the electricity grid
- » 40 years of AI and energy experience across Google Deep Mind, Siemens Energy, Origin, Kaluza, Nasa and more



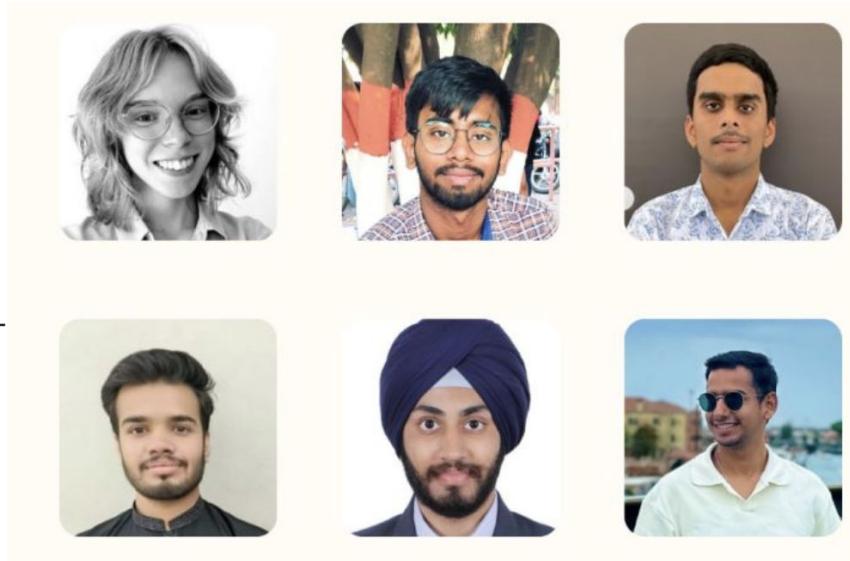
Live software: app.quartz.solar



Open Climate Fix

02

- » Open Data PVNet
- » Efficient Cloud-Based Data Streaming for ML: Implementing Ice Chunk + Zarr 3 for PVNet Training
- » Adjuster This! TabPFN for Solar Forecast error



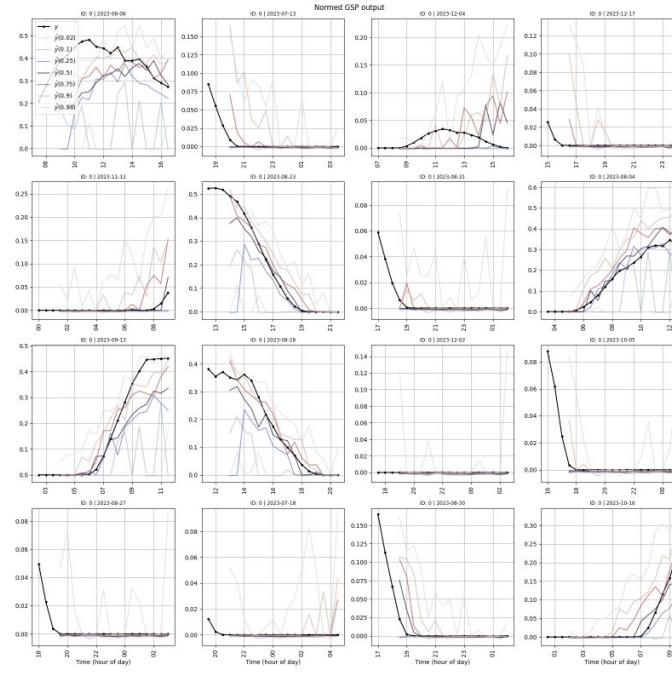
- » Cloudcasting UI
- » Improving Probabilistic Solar Forecasts
- » Quartz Solar – New data source in ML model

Open Climate Fix - Open Data PVNet

[Github code](#)

03

- » Train ML models to prediction Solar generation on Open Numerical Weather Data
- » Data pipeline to make training samples ([ocf-data-sampler](#))
- » Used ML models ([PVNet](#))
- » Repeatable for open community to build on

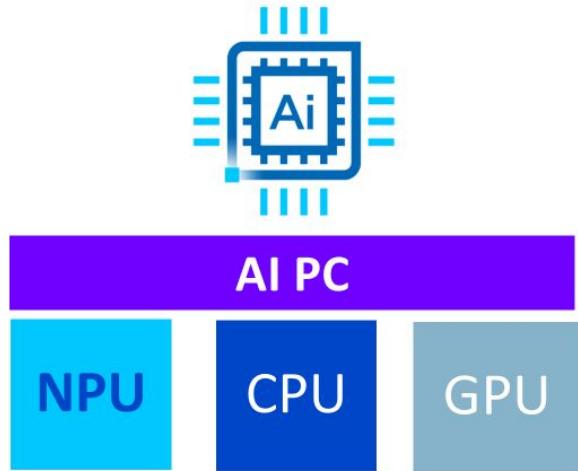




Google Summer of Code

Dmitriy Pastushenkov

OpenVINO Toolkit



**Limited computing
resources, but...**

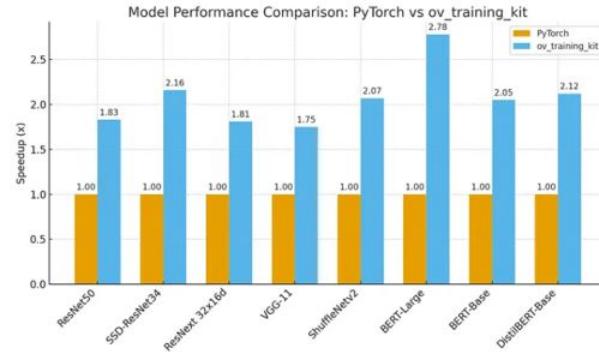
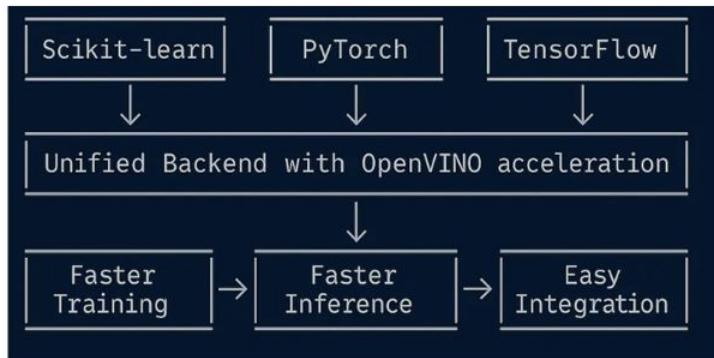
...there is

OpenVINO™

From AI Training

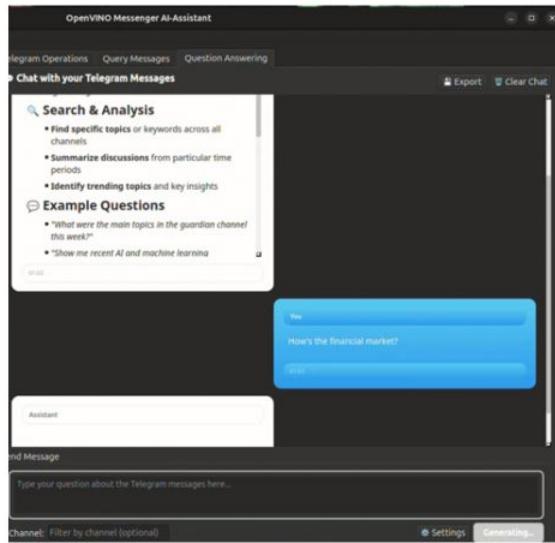
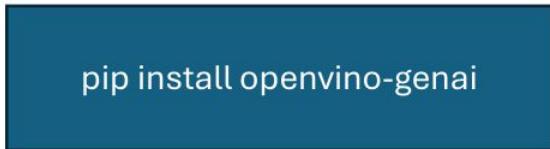
AI PC Model Training Kit

```
pip install ov-training-kit
```

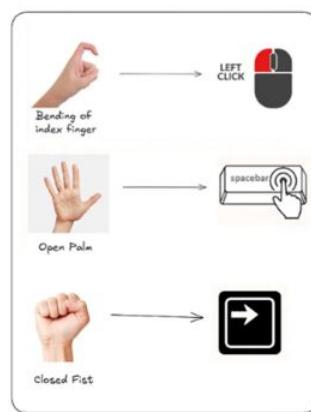
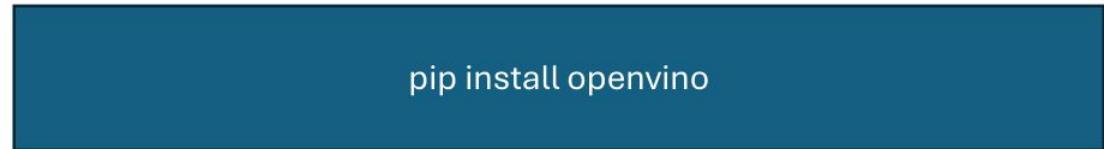


....to AI (and GenAI) inference

AI-Assistant for AI PC



Gesture Control System



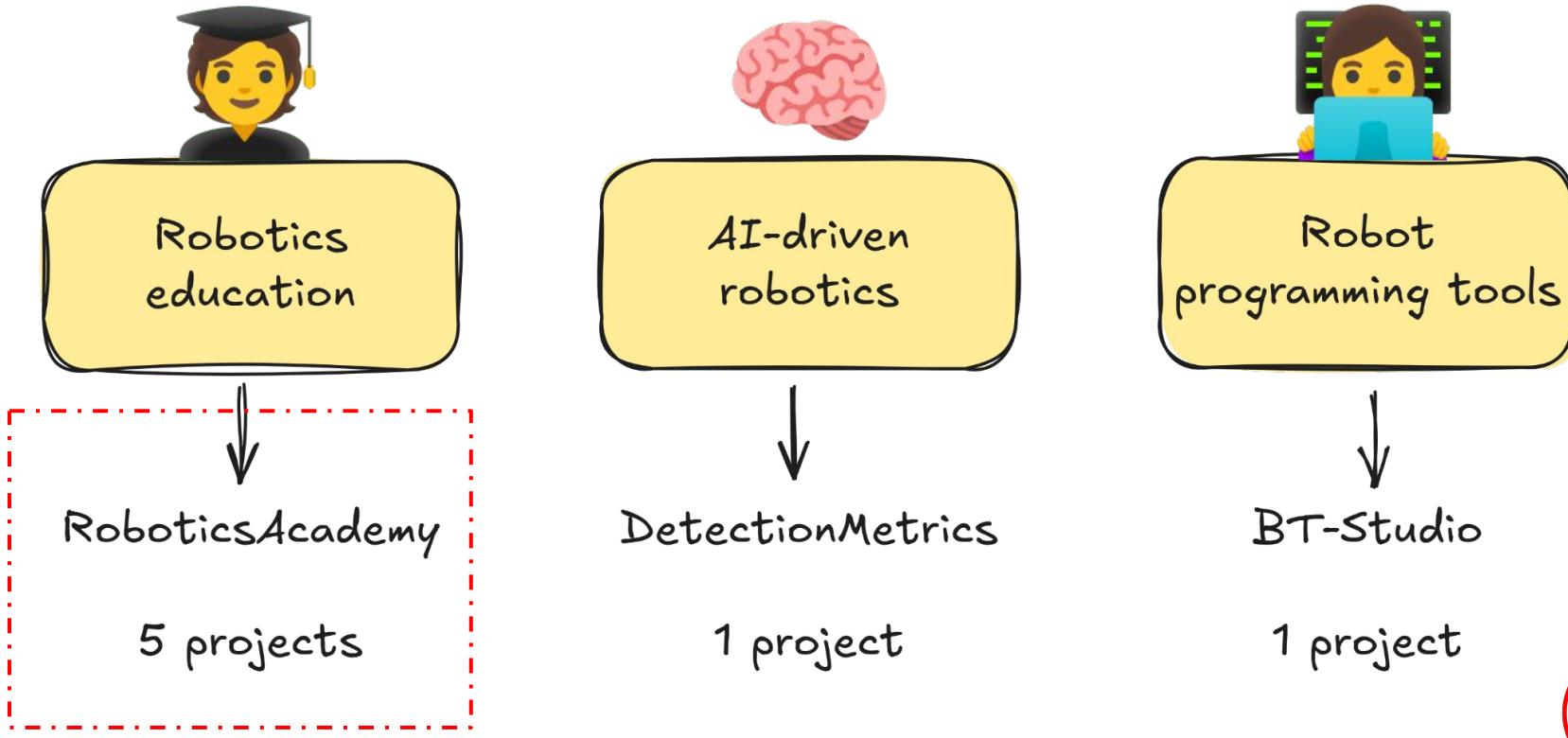


Google Summer of Code

David Pascual

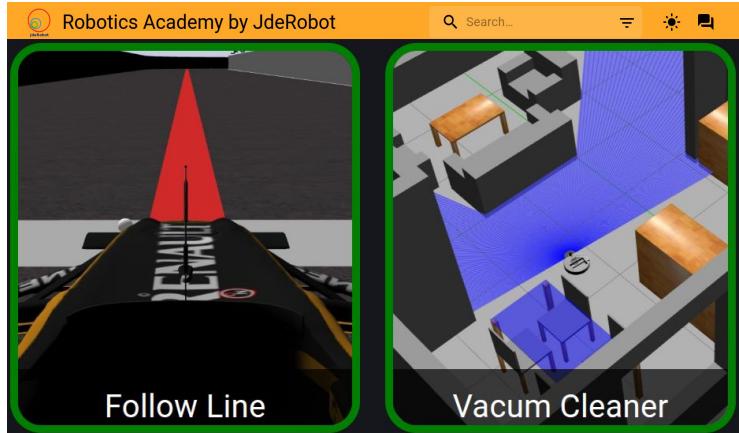
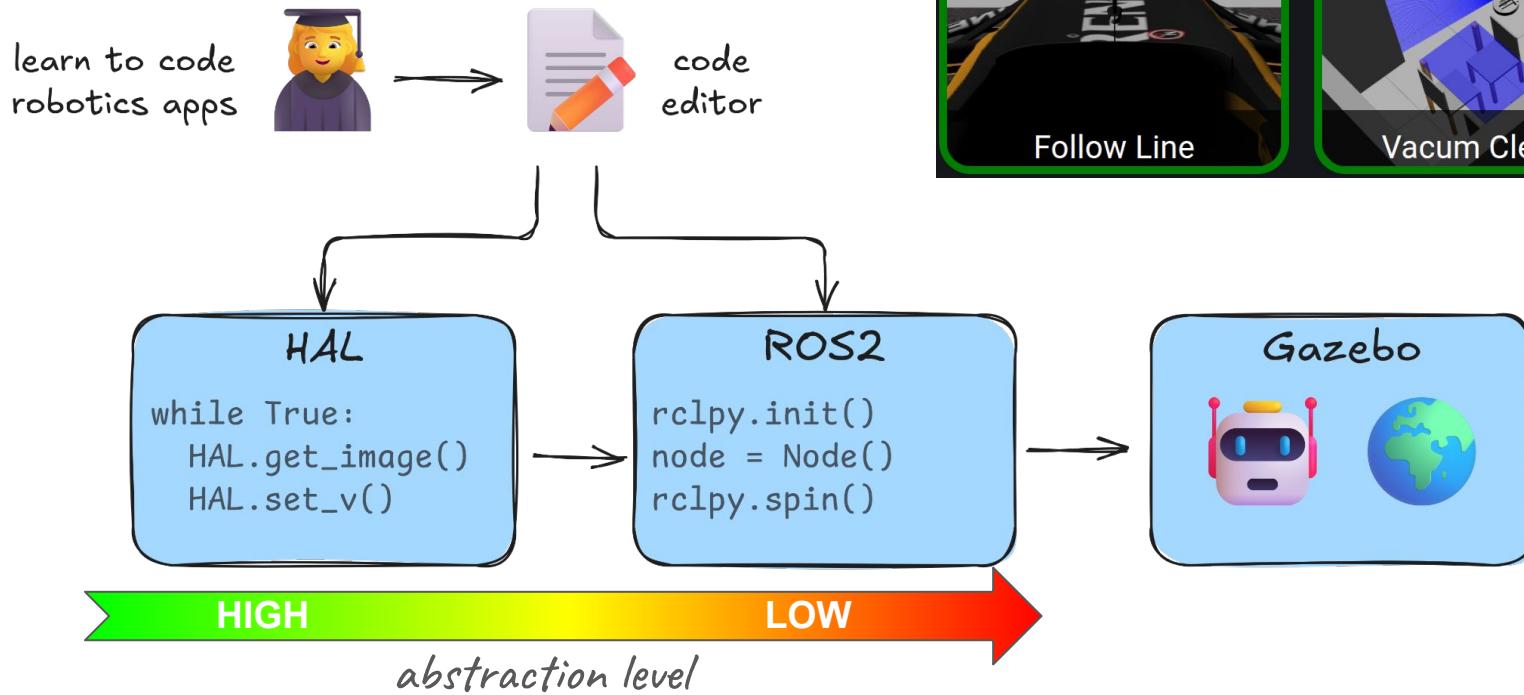
JdeRobot

JdeRobot open-source robotics software



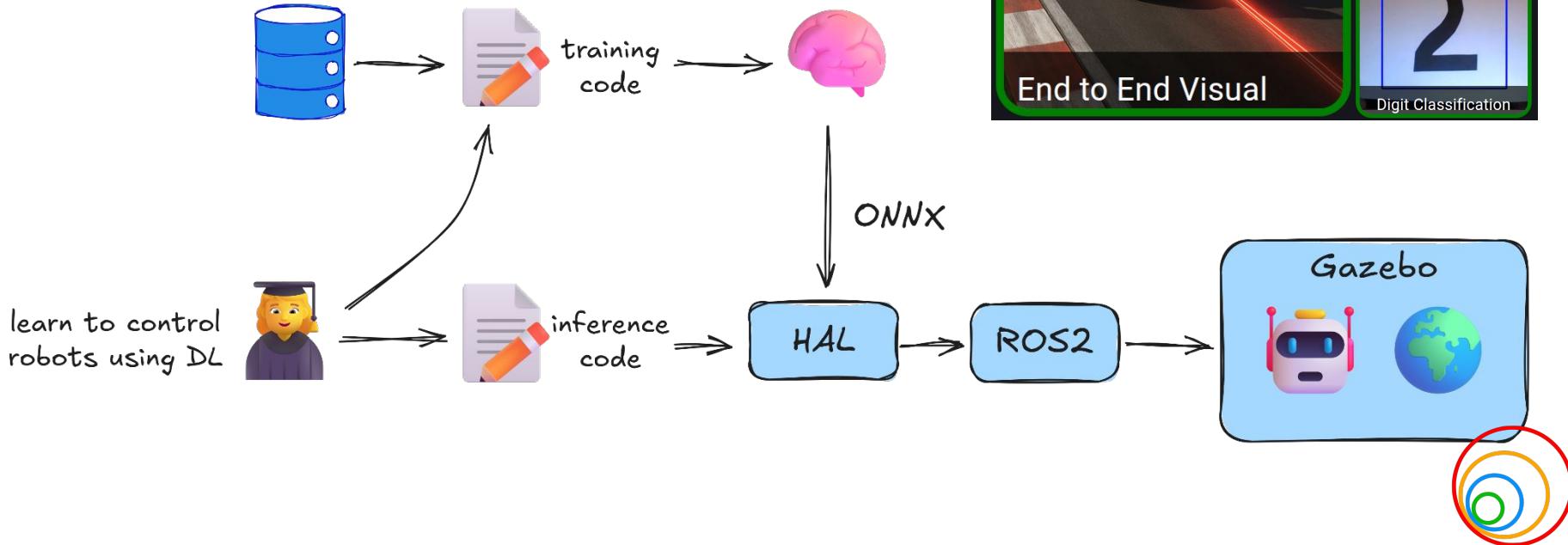
RoboticsAcademy

ROSification



RoboticsAcademy

Deep Learning Exercises





Google Summer of Code

Lorena Goldoni

The Honeynet Project

BuffaLogs - Login Anomaly Detection

A brief project overview



Three login seen:

- 3:00 Milan
- 3:30 Rome
- 12:50 Vienna → New Country
 - ... 30 days later: Austria Login → Atypical Country

If also:

- with different user-agents → **New Device**
- with anonymous IPs → **Anonymous IP Login**

GSoC '25 - Alerting Module Enhancements

By Kunal Gurtatta

The screenshot shows the Django admin interface at localhost:8000/admin/impossible_travel/alert/. The left sidebar has categories like AUTH TOKEN, AUTHENTICATION, and IMPOSSIBLE_TRAVEL (Alerts). The main area is titled "Select alert to change" and lists two alerts:

Action	ID	CREATED	UPDATED	USERNAME	NAME	DESCRIPTION	LOGIN RAW DATA
View	4308	Oct. 18, 2025, 2:47 p.m.	Oct. 18, 2025, 2:47 p.m.	Zayne Ward	User Risk Threshold	User risk score increased for User: Zayne Ward, who changed risk_score from No risk to Low	[Redacted]
View	4307	Oct. 18, 2025, 2:47 p.m.	Oct. 18, 2025, 2:47 p.m.	Zayne Ward	Anonymous IP Login	Login from an anonymous IP from IP: 203.0.113.38 by User: Zayne Ward	[Redacted]

From monitoring in Django... ... to

✉️ Email

🔔 Push notifications

💬 Messaging Platforms

- multi-channel alerts
- formatting templates
- exponential backoff
- reports

GSoC '25 - BuffaCLI

By Onunwa Goodness

Command-line access for log & alert management

```
Usage: buffacli [OPTIONS] COMMAND [ARGS]...

Options
--verbose      -v      [0|1|2|3] Verbose level [default: None]
--install-completion
--show-completion
--help

Commands
setup  Configure BuffaCLI-specific setups.
show   View buffalogs component configurations
query  Query users and alerts record
```

 Config

 Query

 Export

 Customize



Google Summer of Code

Felipe Borges

GNOME Foundation



Google Summer of Code

Till Kamppeter

The Linux Foundation



Google Summer of Code

Christoph Badura

The NetBSD Foundation

Project overview: Asynchronous I/O by Ethan Miller

- Improve performance and scalability of NetBSD AIO.
- The original model espoused one worker thread per process where jobs were serialized.
- The new model implements per-process service pools with multiple worker threads.
- Threads recycled from a freelist to reduce creation overhead.
- The principle is to embrace cheap submission, sensible coalescing, and spawn threads only when blocking is demanded.

All the work done by Ethan as part of GSoC was imported in NetBSD on 2025-10-10 by Christos Zoulas!

Service Pool Architecture WIP

- Each process owns a service pool that spawns and manages service threads.
- Jobs are grouped per vnode; one thread services a given file. Avoids lock thrashing on the same vnode.
- Enables concurrency across different files and devices.
- Synchronous fallback remains when blocking is unavoidable.
- Multi-file workloads already benefit from parallel vnode execution.

I/O Path Challenges

- Synchronous VFS path (bread/bread + biowait) serialized asynchronous IO on cache miss hence introducing a bottleneck.
- We tried a VOP_BMAP direct bypass, but safety and cross-filesystem portability were problematic.
- Using B_ASYNC with biodone required fragile ordering and bookkeeping, and releasing higher-level locks early would be precarious if not handled with the utmost precision.

Comparison and Future Work

- In FreeBSD struct bio decouples the I/O token from the cache, allowing GEOM to split, clone, and queue bio asynchronously. Where advisory record locks do not block in-flight I/O completion.
- In NetBSD such a bio-like layer is absent, and struct buf serves as both the cache object and the I/O handle.
- The next step is to introduce a bio-like abstraction and integrate it within the existing service pool architecture.
- The expected impact is true concurrency on the same vnode and reducing blocking within the VFS.

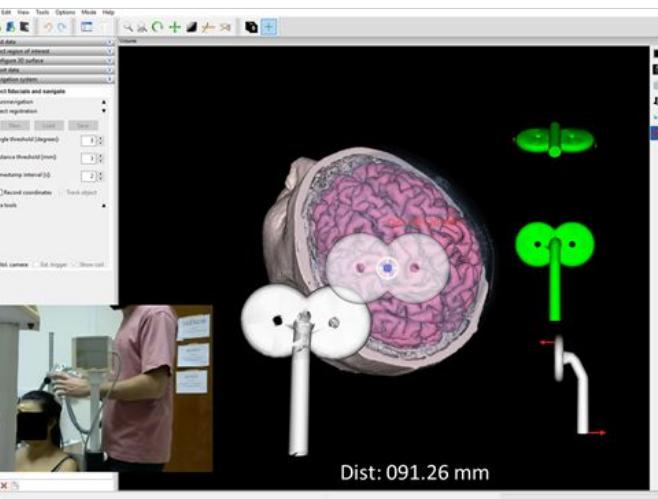
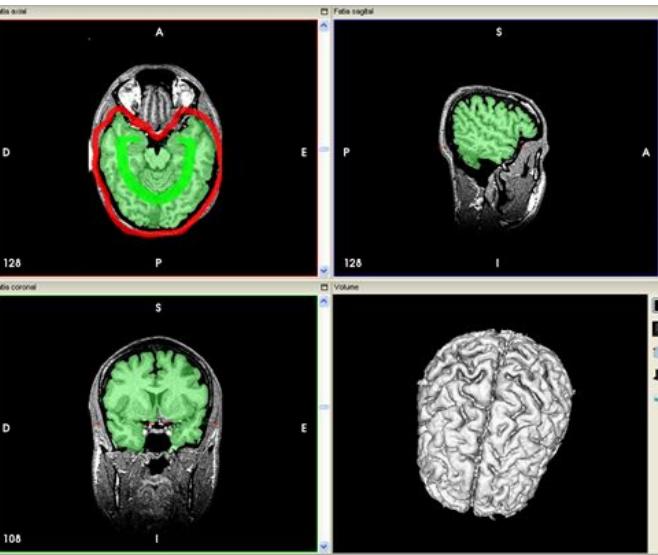
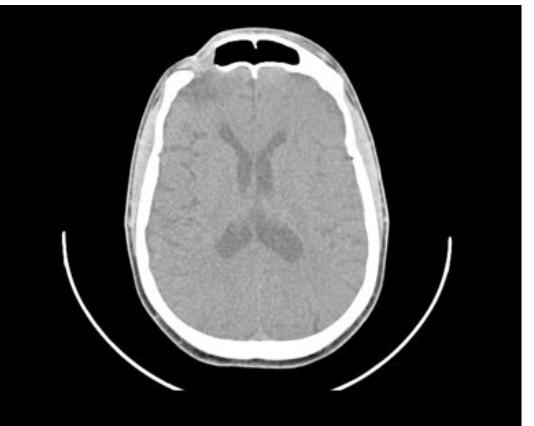
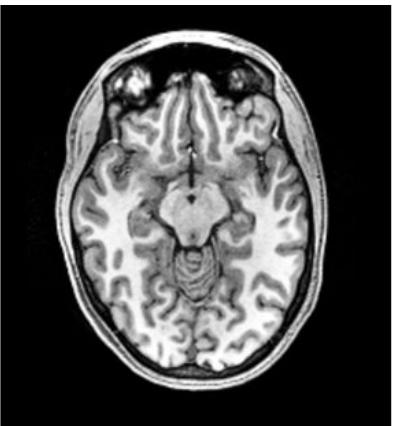


Google Summer of Code

Renan Matsuda

InVesalius





InVesalius - Brain parcellation

Subpart Segmentation X

Backend Pytorch

Device NVIDIA GeForce RTX 2050

Overlap
0% 10% 25% 50% (selected)

Mask Types to Generate
 Whole Brain

Cortical
 Subcortical
 White Matter
 Cerebellum
 Ventricles
 Brain Stem
 Choroid Plexus

Level of certainty
75%

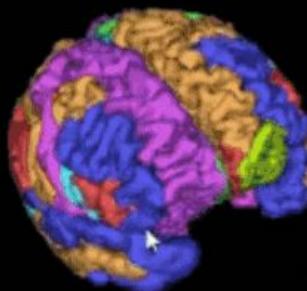
Apply WWL
 Create new mask

Data

Masks 3D surfaces Measures

► General
► cortical
► subcortical (selected)

Ready

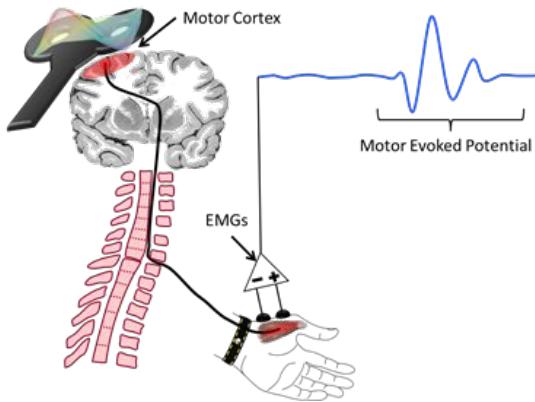


Close Stop Segment

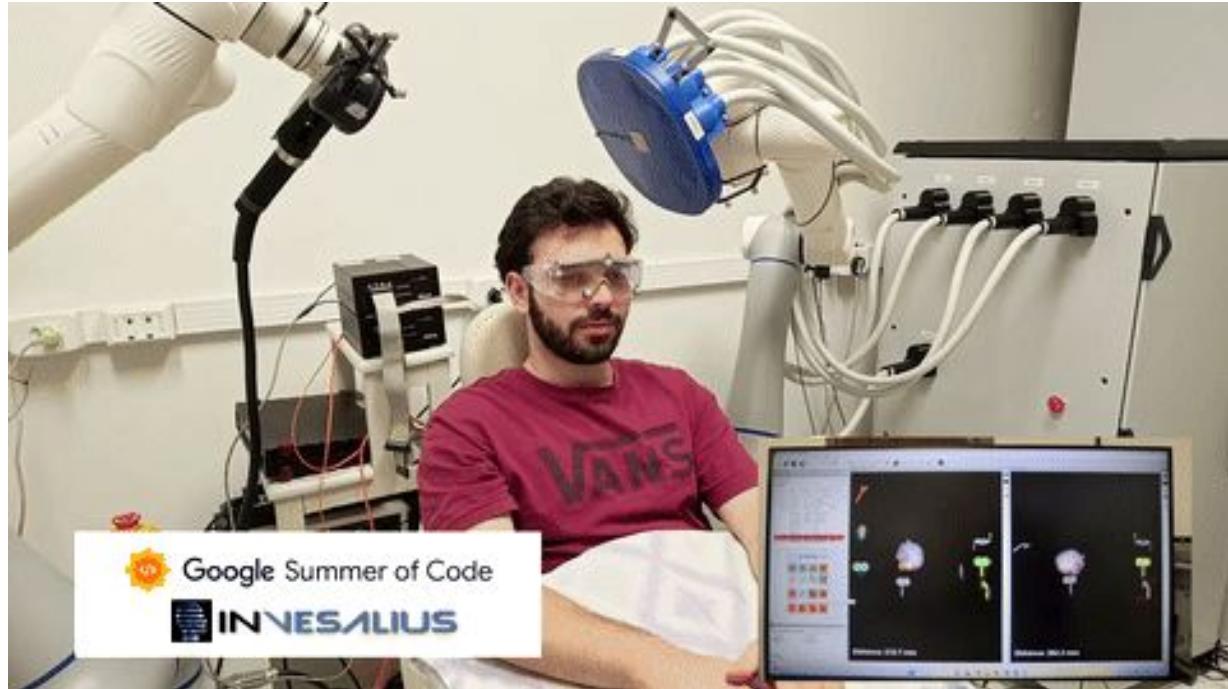


Shuvam Pal
2025

InVesalius - Dual robot control



Marcio Campos
2025





Google Summer of Code

Vissarion Fysikopoulos

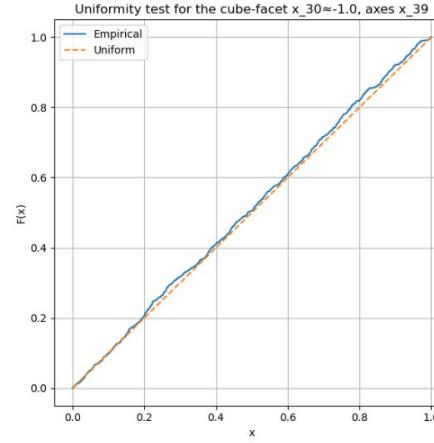
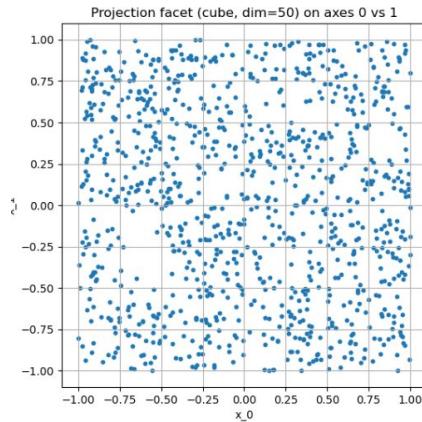
GeomScale

Shake and Bake - Sampling from the boundary of convex polytopes

- **Contributor:** Iva Janković (currently PhD student @ Univ. of Edinburgh)
- **Goal:** Implement Markov Chain **Monte Carlo**-based Shake and Bake algorithm for uniform sampling from the boundary of a convex polytope
- **Motivation:** Fundamental problem in computational **geometry & statistics** + applications in metabolic network analysis
- **Results:** Implementations of new and classic algorithms in **C++** with **R** and **python** interfaces

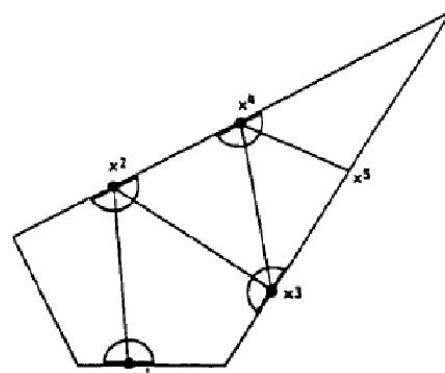
Challenges I

- Implement the basic version of 80's algorithm
- Much faster than existing implementations (`shakeandbake` R package)
- **Challenges**: evaluate the **quality** of samples
- **Solution**: implement a **statistical test** for each facet of the polytope

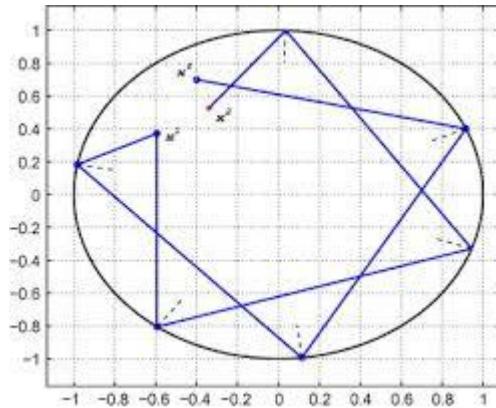


Challenges II

- Implement a new hybrid algorithm: **billiard shake and bake**
- Preliminary results show that it is much **faster** than the original one



+





Google Summer of Code

Sreeja Kamishetty

Representing our Chromium GSoC Community



Celebrating 5th year - Chromium GSoC

Mission: To foster an open source culture, bring more voices to web, mentor and hire diverse next generation talent to making the web better for everyone.

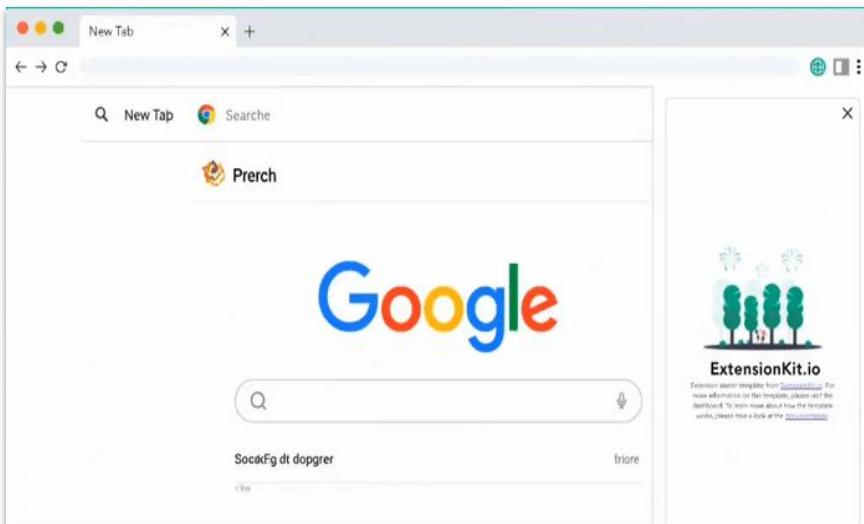


Our Impact so far

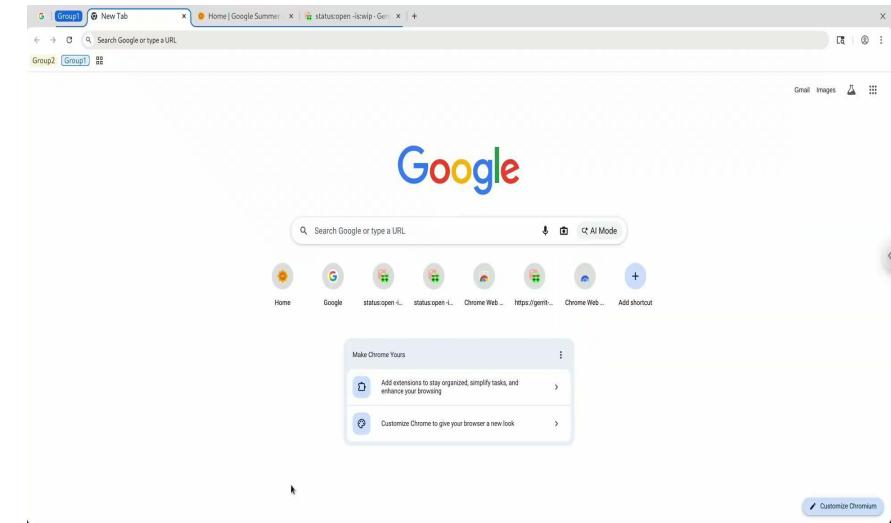
- Total Successful Projects: 64 (Chrome & Web Ecosystem: 52, Chrome OS: 12)
- 100% success rate
- 7000+ contributors
- 1000+ proposals and contributions
- Continuous contributions through the year

Chromium Projects Showcase

Side Panel APIs (Harsh Singh): Introduced new methods (`getLayout()`, `close()`) and lifecycle events (`onOpened`, `onClosed`) for greater developer control.



Debug UI for Chrome Tabs (Brijesh Giri) : Build an internal debugging page that displays the live-state of Chrome's Tab system in a tree-like representation.



We <3 GSoC

Thank you to our Chromium Mentors



David Pennington



Alison Gale



David Adrian



Andrew Williams



Jingyuan Liang



Oliver Dunk



Devlin Cronin



Solomon Kinard



Atharv Maan



Darryl James



James Price



dan sinclair



Solomon Peña Moreno



Christian Biesinger



Michal Mocny



Annie Sullivan



Kenneth Albanowski



Sarthak Kukreti



Alexis Savery



Hongchan Choi



Michael Wilson



Ray Xu



Nathan Muggli



Benjamin Gordon



Jenna Himawan

- Big Thank you to the GSoC Team - Stephanie, Mary, Lucy.
- Google OSS Team for the GSoC Mentor Summit
- GSoC and the impact it had on the open source community wouldn't be possible with all your efforts.



Google Summer of Code

Yuriko Kikuchi

rocket.chat



Why Communication Matters in GSoC?

Answer:

It's contributor-led. Communication is execution.

- Contributor-led → **mentee owns scope & cadence**
- Global & async → decisions must **survive meetings**
- Volunteer time → **clarity** prevents stalls; **artifacts** keep continuity



My Case: From Waiting to Leading

Before

- Spoke little in calls; English anxiety froze me
- Waited for prompts; no agenda, no explicit asks
- Mentors couldn't infer my intent or blockers

Turning Point

- Realized **I own the project; mentors support**
- Said: “English isn’t my strength — please support me.”
- Sent a 3-line agenda and chaired each call

After

- Speaking turns per meeting ≈5×; decisions faster
- Quicker unblocks; fewer resets after missed weeks



Mentor Playbook: Enable Speaking, Ownership, Flow

- **Ownership hand-off:** ask a **3-line agenda**
 - Status → Decision → Next step (owner/date)
- **Normalize clarification:** script the first minute
 - “Name two unclear points.” / “It’s fine to say I don’t understand.”
- Single source + async default: one Living Doc
 - Context / Decisions / Next Issue / Blockers / Links
 - Batch questions daily with **options + default.**



Waiting List Talks



Google Summer of Code

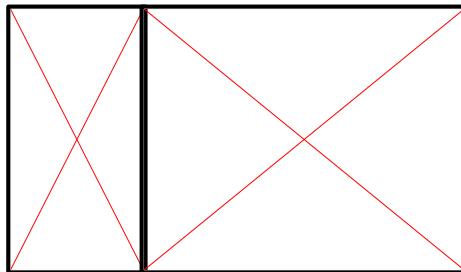
Sarrah Bastawala

INCF | Orthogonal Research and Education Lab

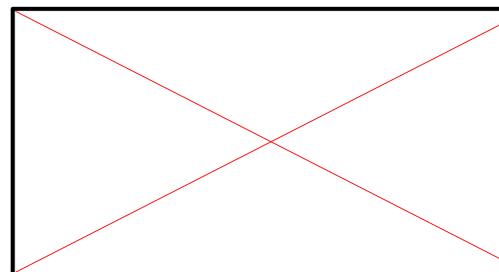
The Agent-Based World of Open Source Community Sustainability

The vision: To build a framework where agent behavior mirrors the complexity, collaboration, and learning found in real-world open-source communities.

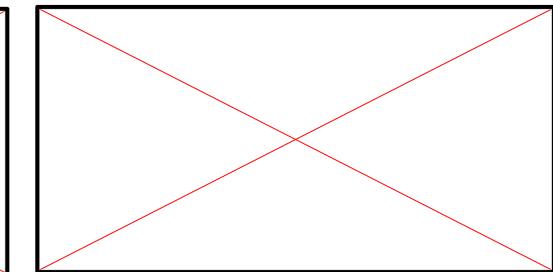
2022



Active Inference



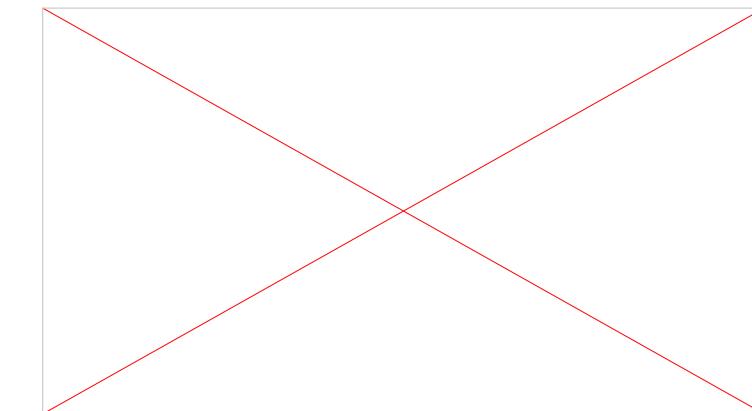
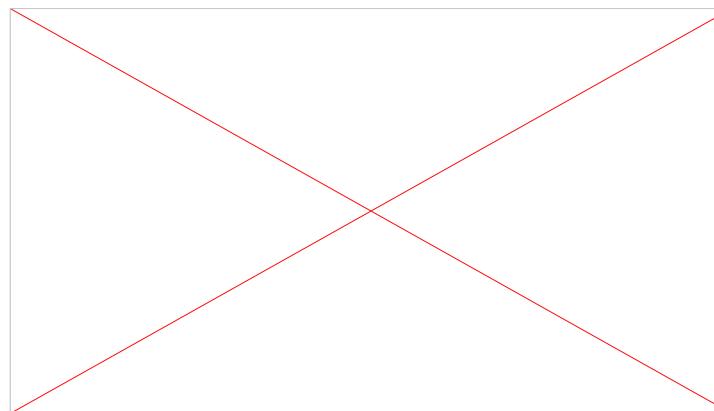
Reinforcement Learning



Big 5 Personality

Alicea, Bradly & Ather, Hussain & Chougule, Himanshu & McCorkle, Brian & Parent, Jesse. (2023). Open-source Community Sustainability using Agent-based Models.

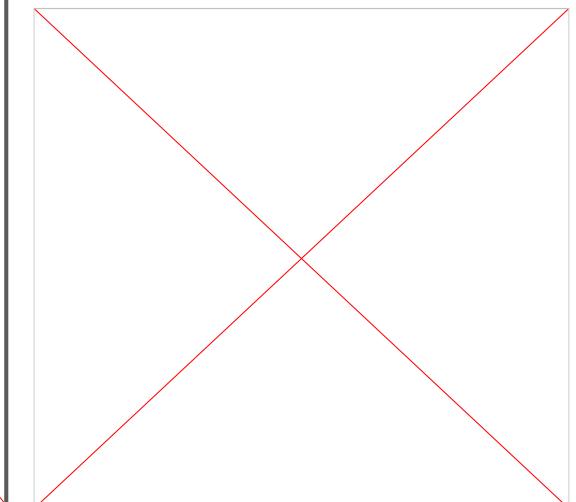
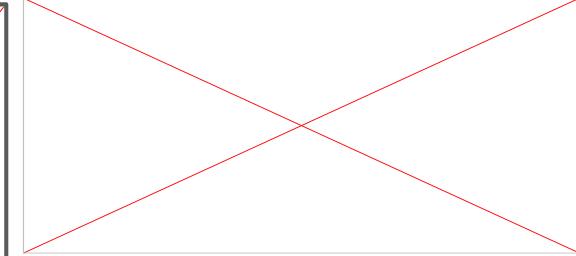
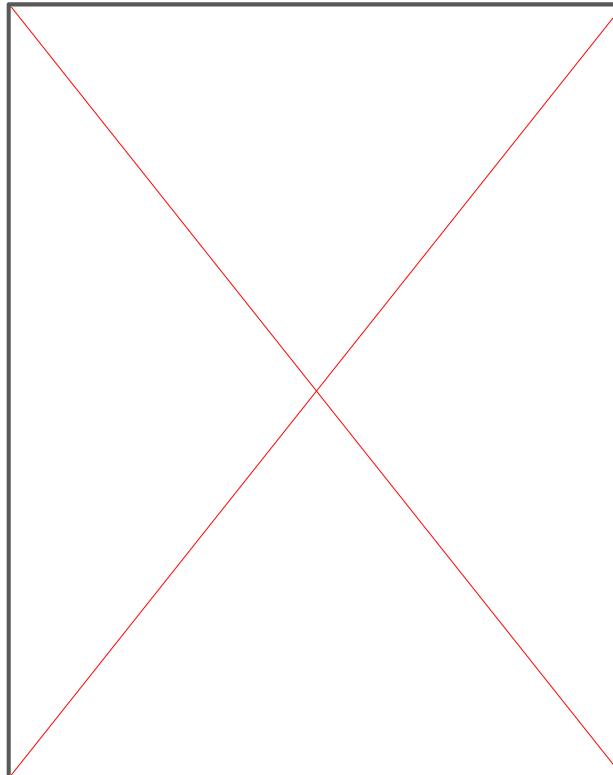
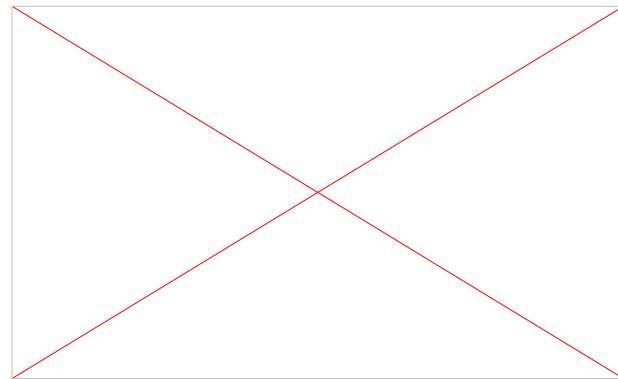
2024



Continuing the Project in 2025 - My Mentoring Experience

SustainHub - Open Source Community Sustainability using RL and QLearning

Mentee: Vidhi Rohira | Mentors : Bradly Alicea, Sarrah Bastawala, Jesse Parent





Google Summer of Code

Oliver Kopp

JabRef e.V.

Document the Why

In the context of <use case/
user story/component ctx>,

facing <non-functional concern c>,

we decided for <option o1>

and neglected <options o2 to on>



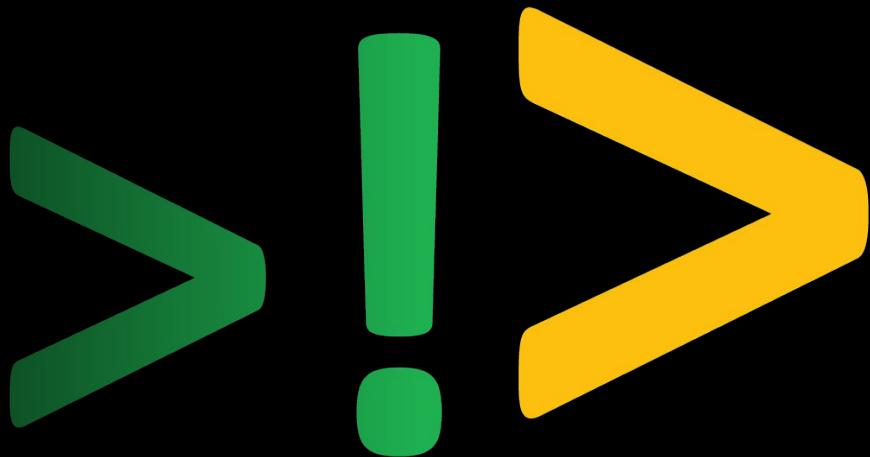
to achieve <quality q>,
accepting downside <consequence c>.

- <https://socadk.github.io/design-practice-repository/artifact-templates/DPR-ArchitecturalDecisionRecordYForm.html>
- <https://github.com/adr/madr>



The Palisadoes Foundation

Equitable access to post-STEM
education opportunities

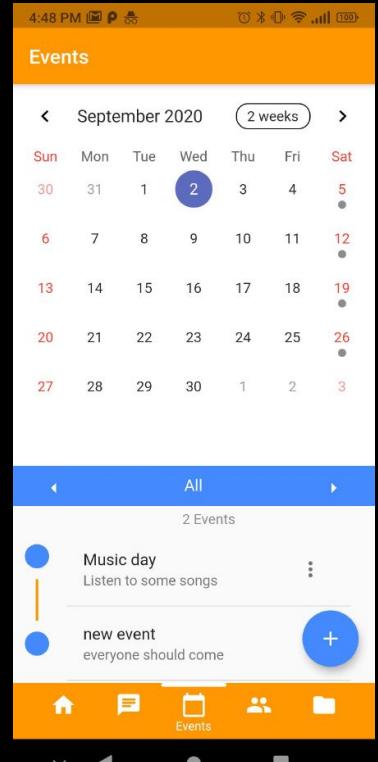


palisado

Talawa:

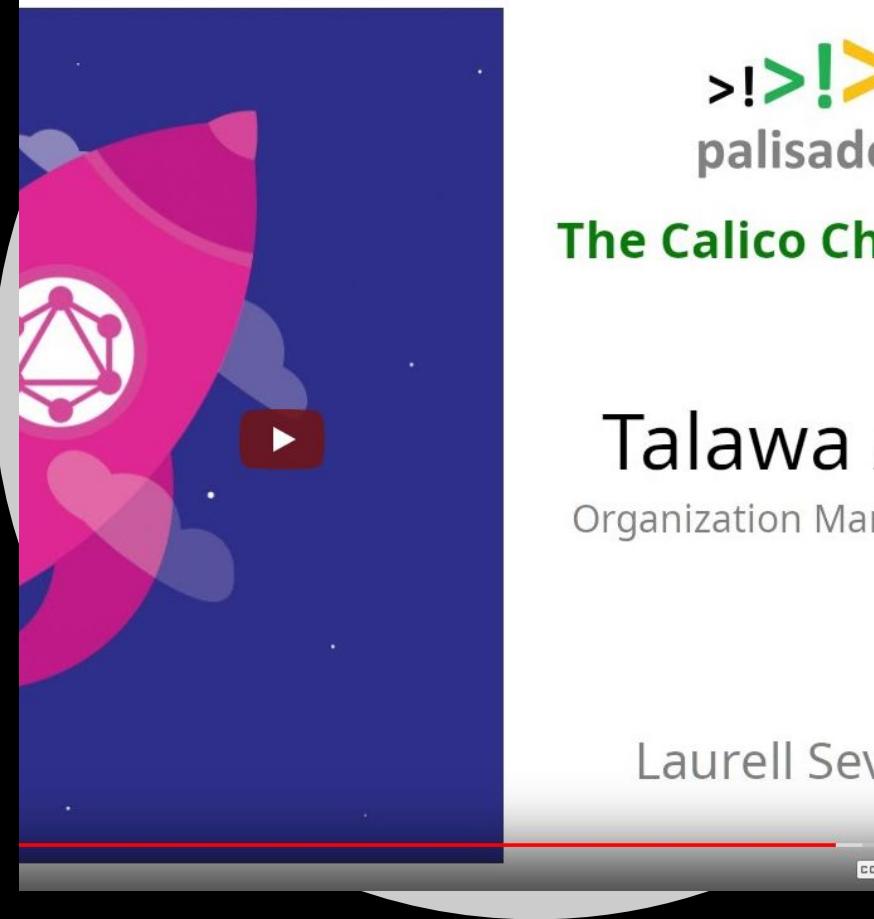
Community Management

- Religious institution focus
 - Announcements
 - News
 - Group chat
 - Events calendar
 - Events project management
- Highly customizable
- Mobile App



See us in Action

- [Palisadoes.org](https://palisadoes.org)
- YouTube (<https://www.youtube.com/@PalisadoesOrganization>)
- GitHub (<https://github.com/PalisadoesFoundation>)
- Talawa (<https://docs.talawa.io/>)





contribute and
connect with
Us





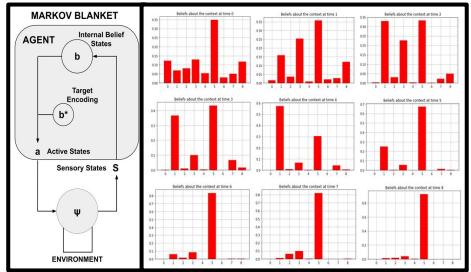
Sarrah Bastawala

INCF | Orthogonal Research and Education Lab

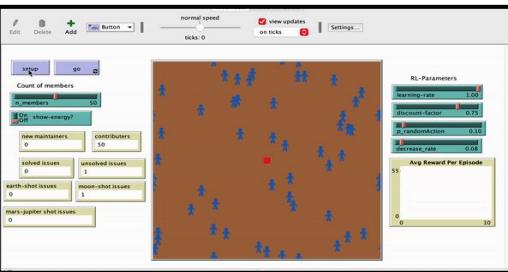
The Agent-Based World of Open Source Community Sustainability

The vision: To build a framework where agent behavior mirrors the complexity, collaboration, and learning found in real-world open-source communities.

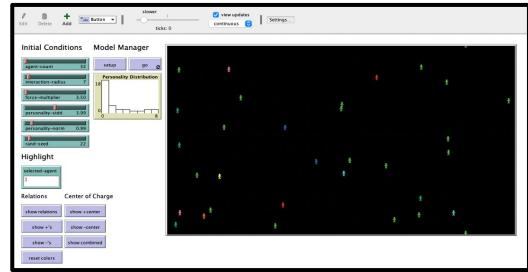
2022



Active Inference



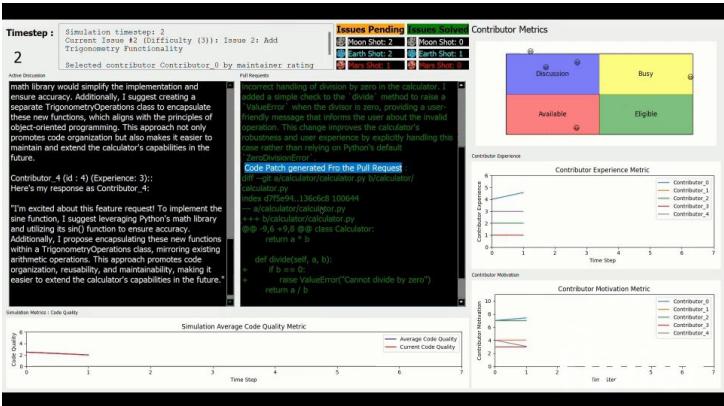
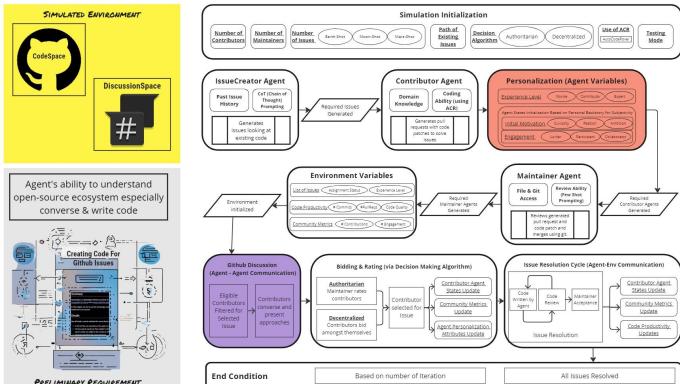
Reinforcement Learning



Big 5 Personality

Alicea, Bradly & Ather, Hussain & Chougule, Himanshu & McCorkle, Brian & Parent, Jesse. (2023). Open-source Community Sustainability using Agent-based Models.

2024

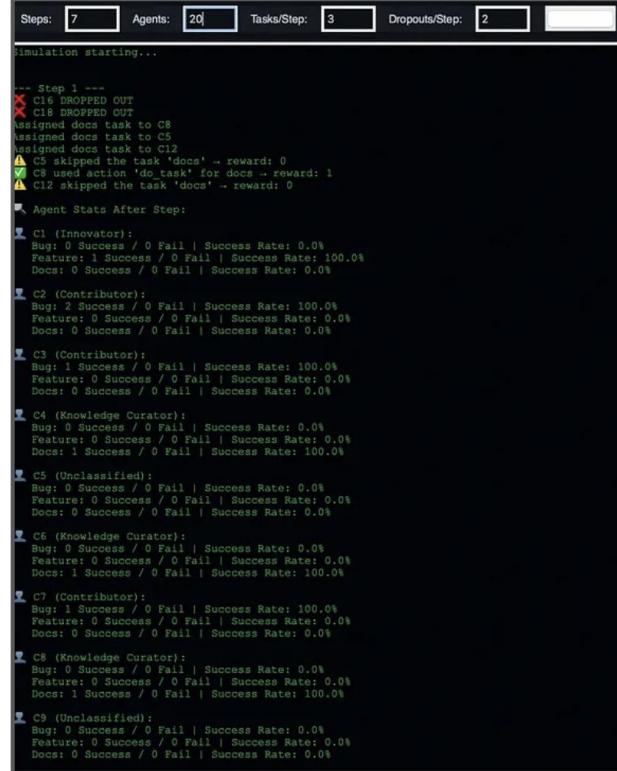
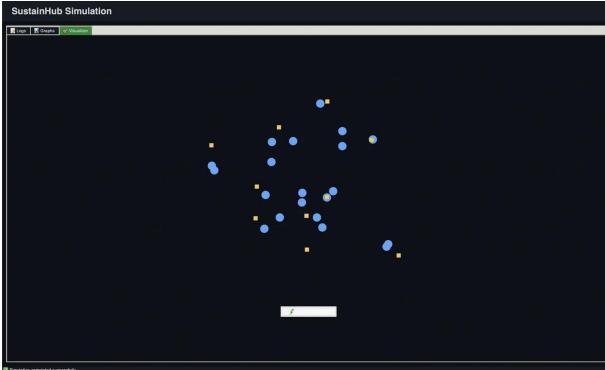


LLM Powered Agent-Based Model for Open Source Community Sustainability - Sarrah Bastawala

Continuing the Project in 2025 - My Mentoring Experience

SustainHub - Open Source Community Sustainability using RL and QLearning

Mentee: Vidhi Rohira | Mentors : Bradly Alicea, Sarrah Bastawala, Jesse Parent



ABSTRACT

Open-source communities thrive on collaborative contributions, but maintaining long-term sustainability requires understanding how contributors engage, resolve issues, and manage knowledge. SustainHub addresses this challenge by introducing an adaptive agent-based model that simulates contributor behavior, task management, and knowledge sharing to optimize workload distribution and enhance community stability. The model introduces diverse agent roles: Innovators, Maintainers, Knowledge Curators, and Contributors, each contributing to project sustainability. SustainHub uses Multi-Armed Bandit (MAB) algorithms for task allocation and Q-Learning/SARSA to refine decision-making, ensuring adaptive task assignments. It incorporates Harmony Index and Resilience Quotient to assess community health, task diversity, and contributor engagement. This project aligns with the mission of the Orthogonal Research and Education Lab (OREL), an open-science lab focused on computational science and the societal impact of AI. By contributing to SustainHub, the project supports OREL's commitment to advancing open science and sustaining thriving open-source ecosystems.

A preprint summarizing the project so far.

Scenarios tested:

1. Scaling up to 30 agents → Could SustainHub handle larger ecosystems?
2. Task overload → Injecting 3x the usual tasks in one cycle.
3. Mass dropout → Simulating 30–40% of contributors suddenly leaving.

Findings:

- SustainHub remained stable with 30 agents and hundreds of tasks.
- Harmony Index dipped temporarily under overload but recovered as SARSA policies adjusted.
- Mass dropouts highlighted a limitation: when too many agents left, even RL could not save sustainability. However, RQ values still showed significantly better resilience than random allocation.

This week taught me a key lesson: scalability is not just about handling numbers, but about maintaining meaning in the metrics as the system grows.

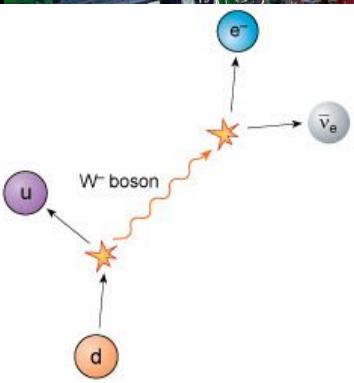
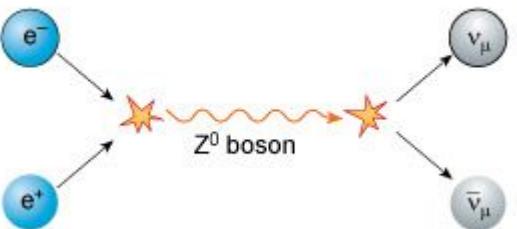


Google Summer of Code

Sanjiban Sengupta

CERN-HSF — ML4EP



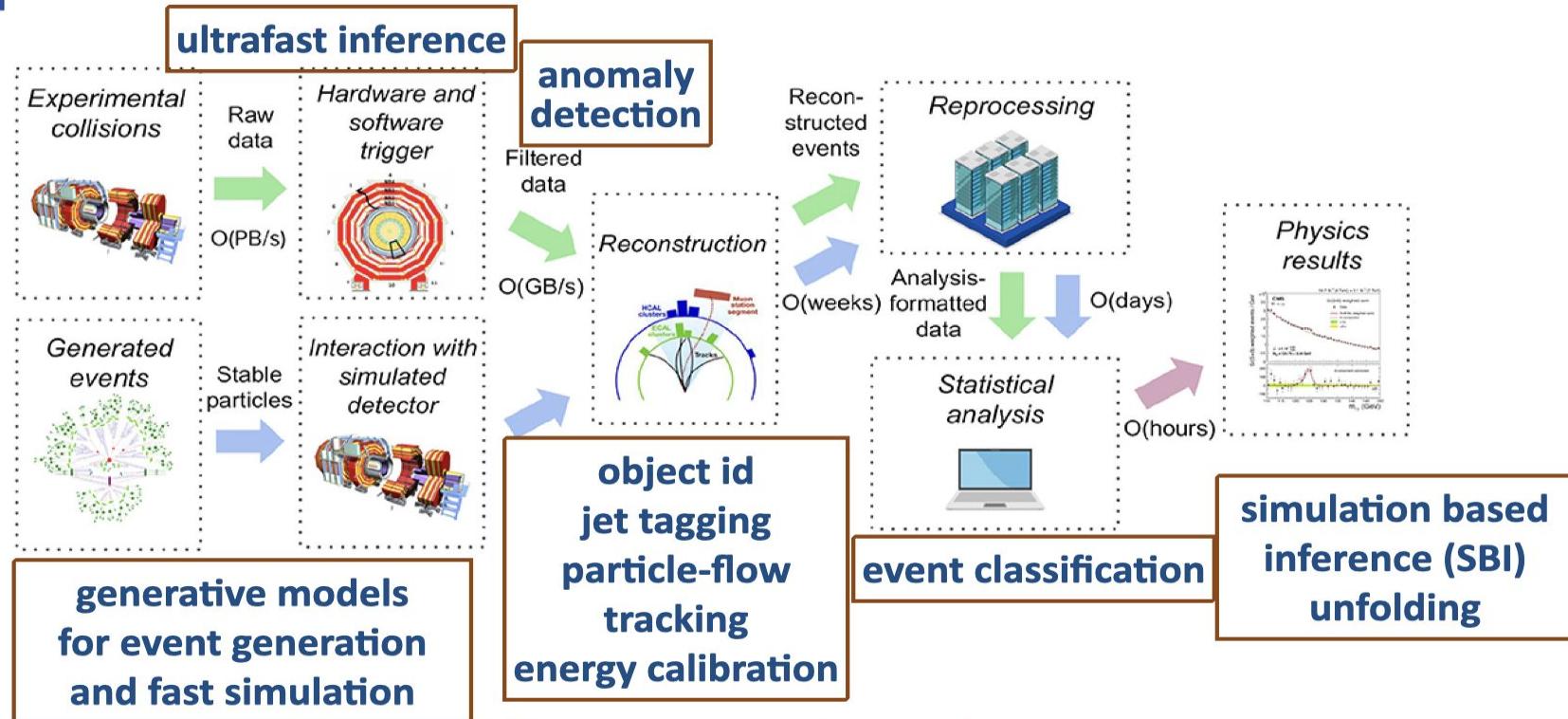


THE HIGGS BOSON





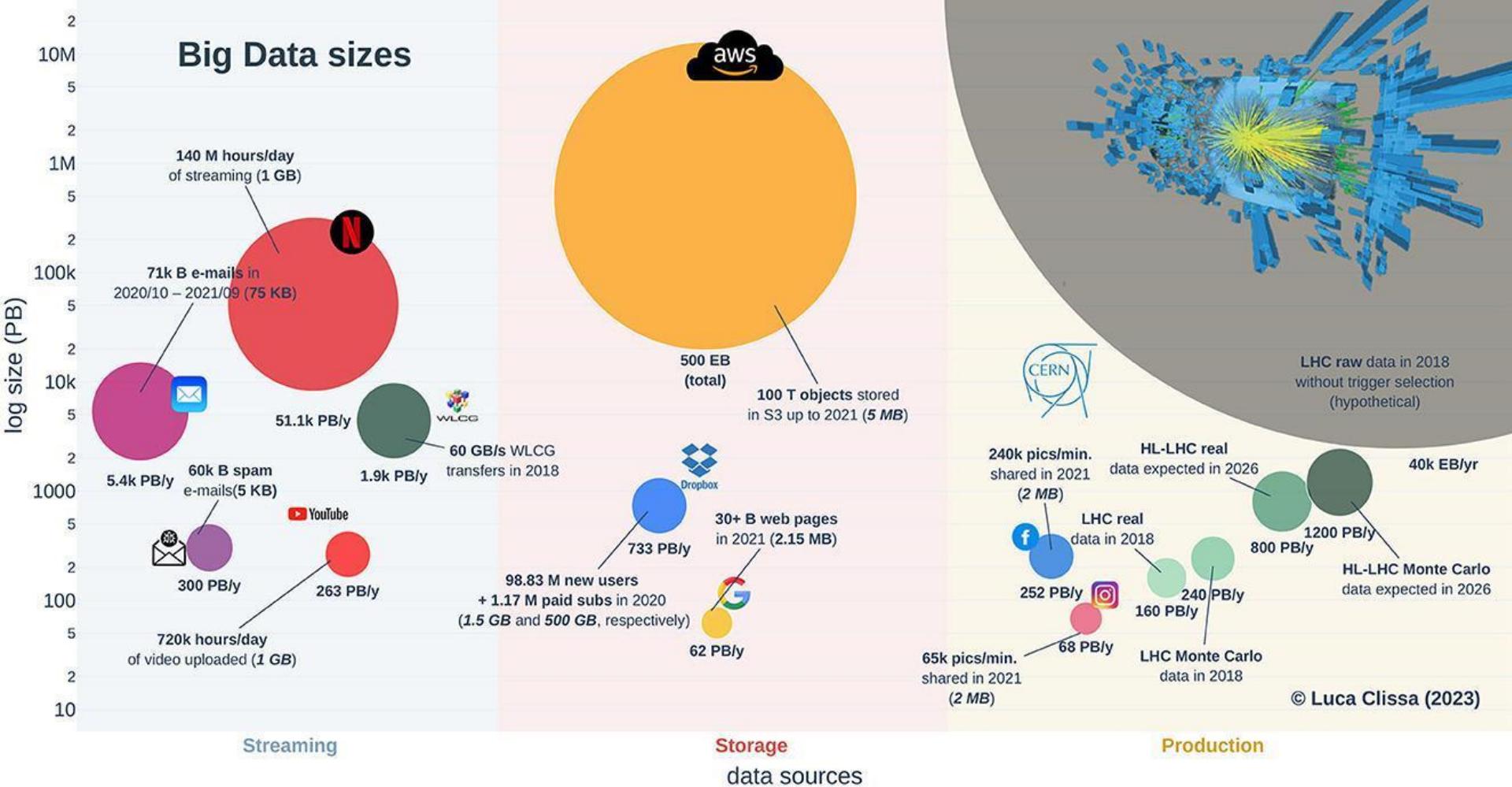
AI in Experiment Data Analysis



AI is everywhere !

pic from [fdata.2021.661501](#)

Big Data sizes

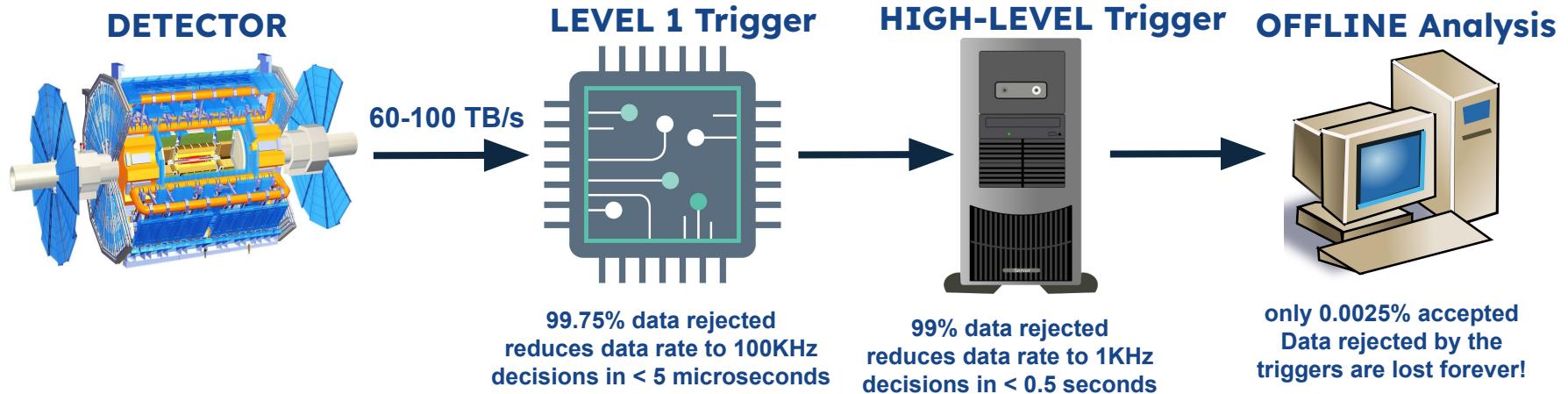


© Luca Clissa (2023)



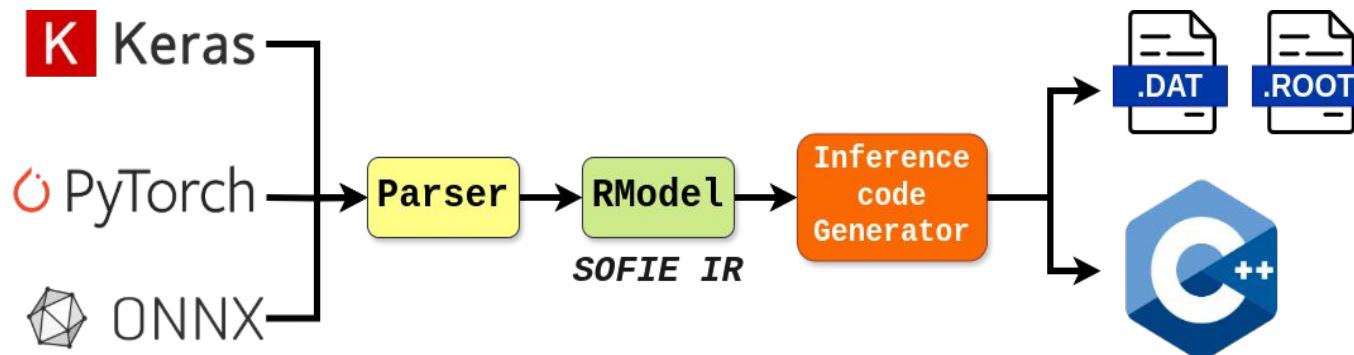
A paradigm shift

dealing with new collision data ~40 million times a second!



SOFIE

- SOFIE - System for Optimized Fast Inference code Emit
- Tool for optimized ML Inference - developed at CERN, that
 - Parses a model trained in Keras or PyTorch, or in ONNX format to its IR
 - Generates inference code in the form of C++ functions (only dependent on BLAS)
 - Supports several ONNX operators, capable of inferring Transformers, GNNs from Deepmind Graphnets.



SOFIE

- **SOFIE - Enhancing Keras Parser**

- **Contributor:** Prasanna Kasar
- **Mentors:**
 - **Lorenzo Moneta (Senior Staff, CERN)**
 - **Sanjiban Sengupta (Doctoral Student, CERN, University of Manchester)**
- **Tasks**
 - Rewrite the Keras model parser in Python, replacing earlier C++ logic to improve modular design and flexibility, and simplify future extensions
 - Extend parser functionality to support Pooling and LayerNormalization layers
 - Enable support for Keras 3, while preserving support for Keras 2.x models, ensuring full backward compatibility
 - Add support for both types of models i.e. models built using Keras' Functional as well as Sequential API.
 - Design comprehensive unit tests for the parser to guarantee robustness and correctness

SOFIE

- **SOFIE - GPU Support for Machine Learning Inference**

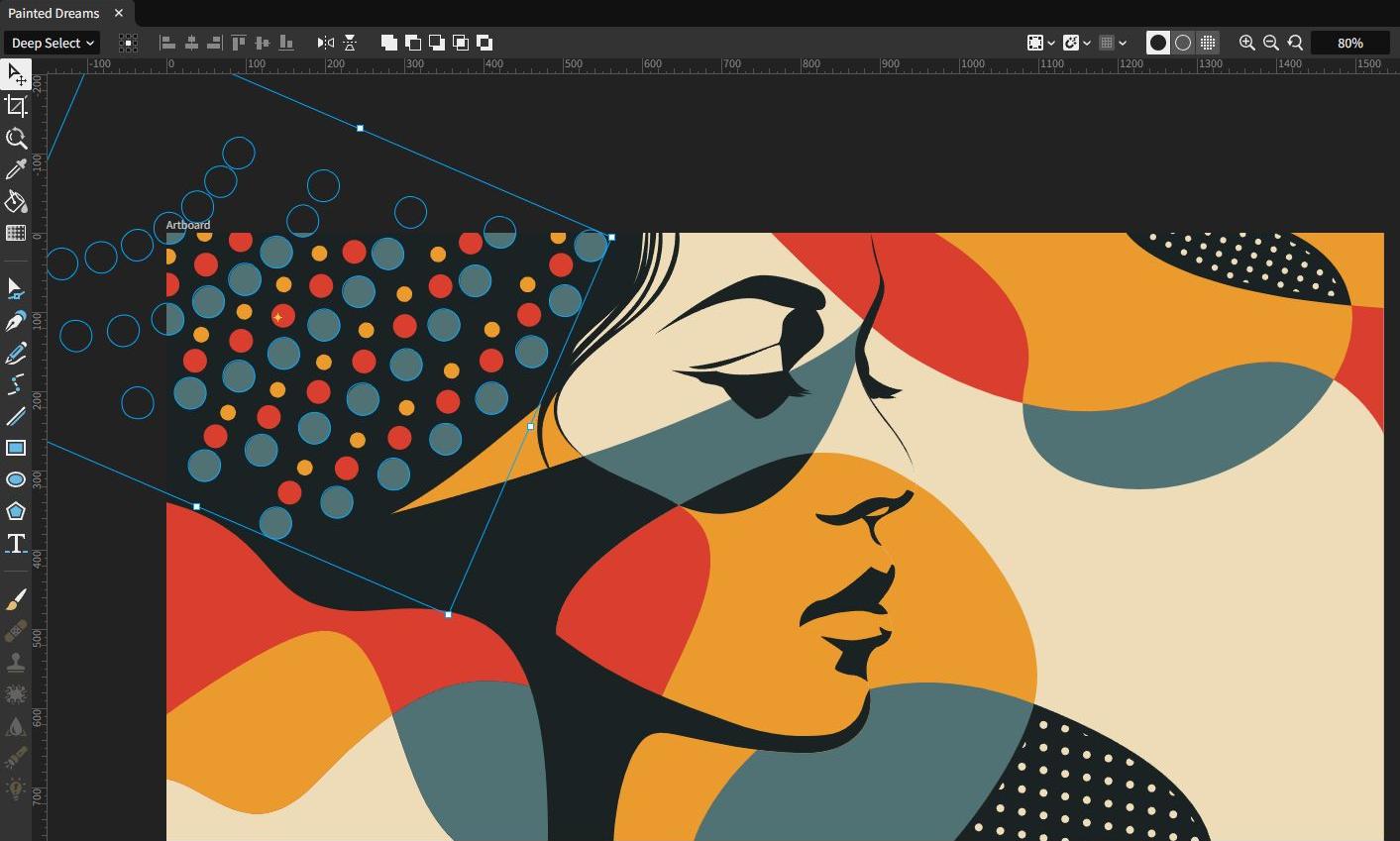
- **Contributor:** S Akash
- **Mentors:**
 - **Lorenzo Moneta (Senior Staff, CERN)**
 - **Sanjiban Sengupta (Doctoral Student, CERN, University of Manchester)**
- **Tasks**
 - Investigate CUDA, ROCm, and ALPAKA frameworks for GPU acceleration. Evaluate pros and cons of each framework in terms of performance and portability.
 - Develop GPU kernels for operations in SOFIE using ALPAKA as an abstraction layer for use of CPU, CUDA or ROCm.
 - Integrate the operators within SOFIE.



Google Summer of Code

Dennis Kober

Graphite



Properties

Repeat X 0 Y 416.7

Direction Angle Instances

Repeat X 459.4 Y 0

Direction Angle Instances

Fill Fill Solid Gradient

Circle Radius 20

Layers

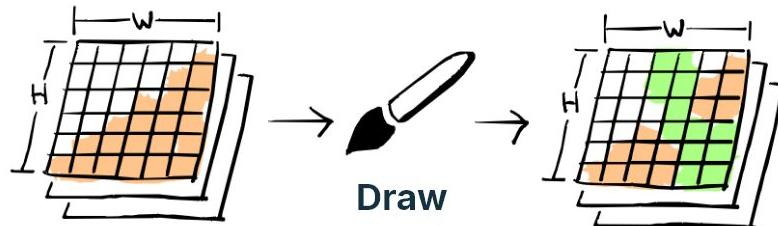
- Normal Opacity 100%
- > Top Right Red
- > Top Right Blue
- > Cheek Shadow
- > Face Black
- > Face Blue
- > Bottom Left Blue
- > Bottom Left Orange
- > Bottom Left Red
- > Bottom Left White
- > Hair Dots Orange
- > Hair Dots Red
- > Hair Dots Blue
- > Face Shadow
- > Face Red
- > Mouth Orange

Deconstructing destructive design

- » Graphics editing has traditionally been based on layers of destructively edited content

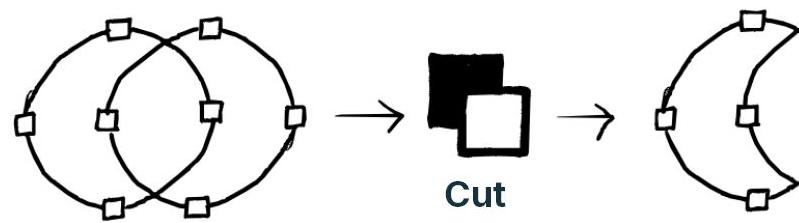
Raster

- » Pixels with dimensions, not infinitely scalable
- » Drawing permanently replaces pixel colors



Vector

- » Points and curves are infinitely scalable
- » Editing permanently mutates the geometry

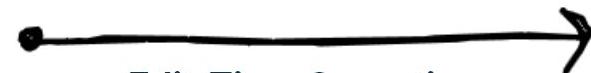
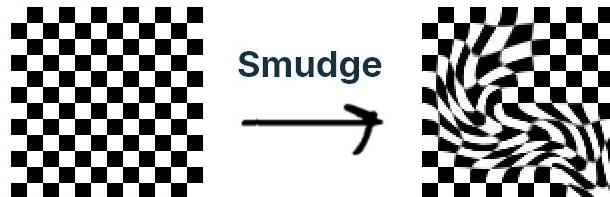


- » Raster and vector don't mix nicely since they scale differently, so apps specialize on just one

Node-based editing

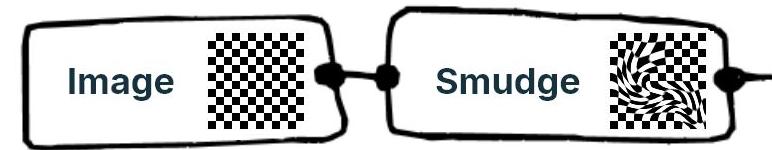
- » Replaces an editing operation with a data operation

Destructive (pixel layer)



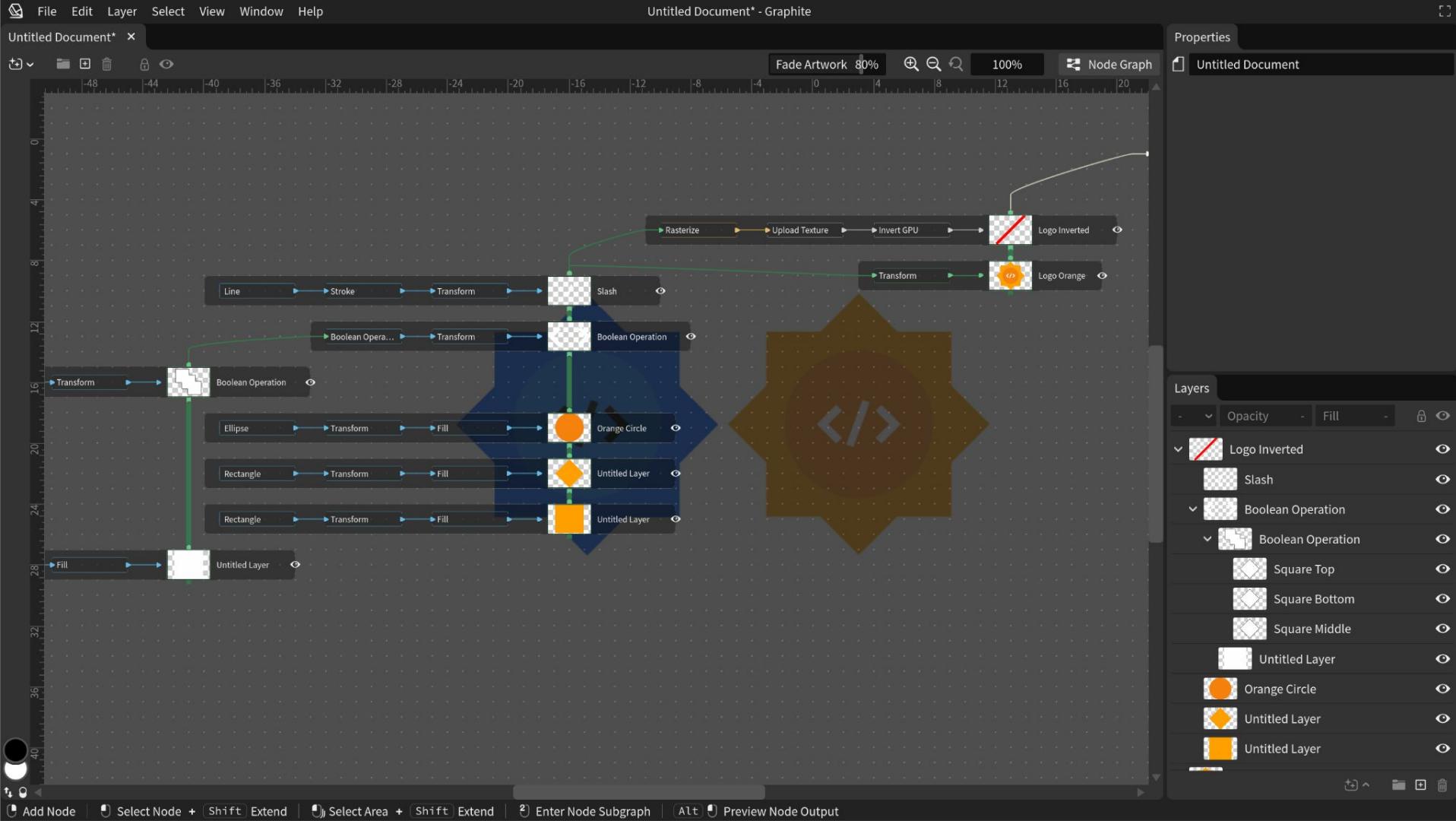
Edit-Time Operation

Nondestructive (node graph)



Render-Time Operation

- » Generates data at any resolution on the fly — scalable raster and vector, harmoniously combined



Where to find us (and get involved)

- » Website: graphite.rs
- » Get involved: graphite.rs/volunteer
- » Ideal skills: compilers, graphics, gamedev, GPU/low-level
- » Ask on Discord, we'll help you begin: discord.graphite.rs





Google Summer of Code

Ivan Ogasawara

MAKIM

YAML-first task runner for humans

Makim lets you define repeatable tasks with arguments, conditionals, hooks, scheduling, and remote execution — all in a single `.makim.yaml`. Clean, documented, CI-friendly.

[Get Started](#)[Read the Docs](#)[Try on Colab](#)

CI passing

pypi v1.27.0

python 3.9 | 3.10 | 3.11 | 3.12 | 3.13

license BSD 3 Clause



chat 17 online

</> Example `.makim.yaml`

```
groups:  
build:  
env:  
  GROUP_ENV: group_value  
tasks:  
clean:  
help: Clean build artifacts  
args:  
cache:  
  type: bool  
  action: store_true  
  help: Remove all cache files  
log:  
  path: ./logs/clean.txt  
  level: err  
  format: "%(asctime)s - %(file)s - %(level)s - %(message)s"  
run: |  
  echo "Cleaning build directory..."  
  rm -rf build/  
compile:  
help: Compile the project  
hooks:  
pre-run:  
  - task: build.clean  
run: |  
  echo "Compiling the project..."  
  
scheduler:  
daily-clean:  
task: build.clean  
schedule: "0 0 * * *"
```