# VerCors

# Verification of Concurrent and **Distributed Software**

# Problem

**Concurrency** in systems can cause subtle bugs that are difficult to detect. As a result, concurrent systems are notoriously difficult to build. To help build correct software, we develop VerCors, a tool for the verification of concurrent and distributed software.



### How does it work?

- Specification describes the intended behaviour of the system
- The user provides the program code and specifications to VerCors
- VerCors determines whether the program is correct w.r.t. the specification using logical inference
- VerCors supports multiple languages including Java, C, CUDA and OpenCL!



#### Achievements

- Verified Parallel Nested DFS, an important verification algorithm
- Case study with **Technolution** to detect bugs in their tunnel control software
- VeyMont: Given a verified program, we can generate a correct parallelised version
- Alpinist: Automatic transformation of specifications for GPU optimisations

## What's next?

- Translate specifications between tools
- Generate specifications
- Apply VerCors to embedded & industrial systems
- Improve usability and scalability of the approach

#### To your project?





#### **UNIVERSITY OF TWENTE.**



#### **Current collaborators**

Marieke Huisman (Project lead), Lukas Armborst, Petra van den Bos, Pieter Bos, Paula Herber, Sophie Lathouwers, Raúl Monti, Robert Rubbens, Mohsen Safari, Ömer Şakar, Philip Tasche.

#### **Funding projects**

