

# CI/CD for ECU FW Development



# Hello!



Team Lead Accu &  
E 2025 Season

Computer Science  
Student TU  
Dresden

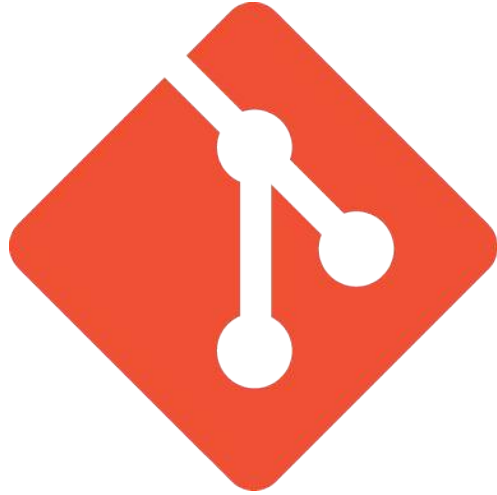
## Tim Bitteker

# Outline

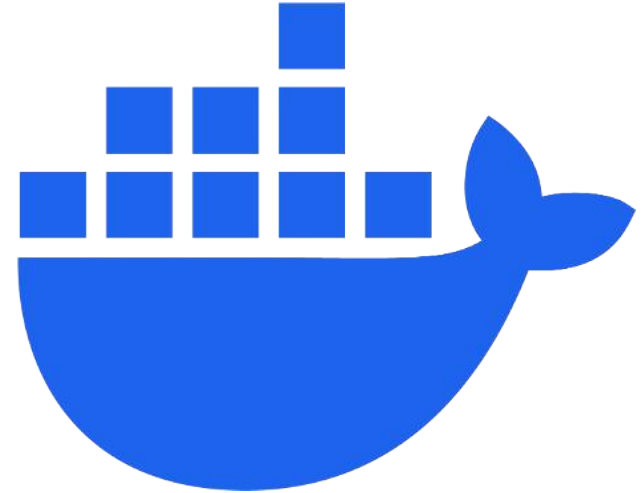


1. Motivation
2. Previous Workflow
3. Pipeline Overview
4. Pipeline in Action
5. Results
6. Outlook





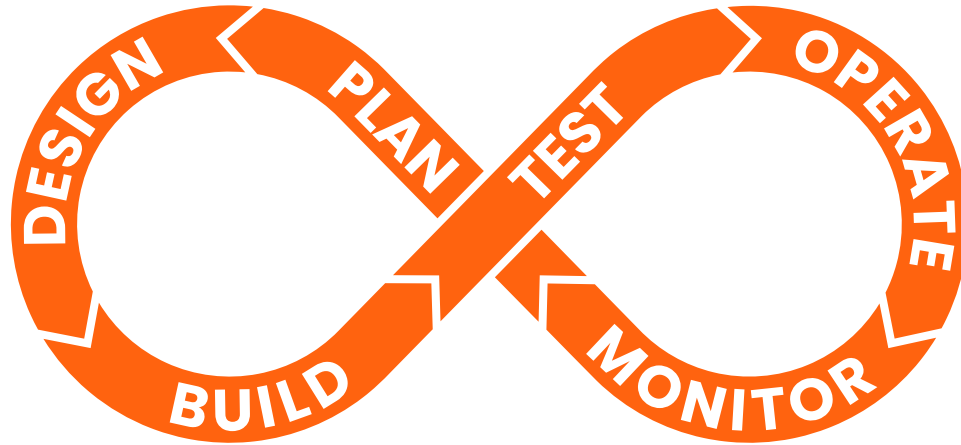
Collaboration and version control tool for software projects



Docker is a container engine. Containers allow software and its dependencies to be configured and run in an isolated and reproducible manner



## Continuous Integration and Continuous Development

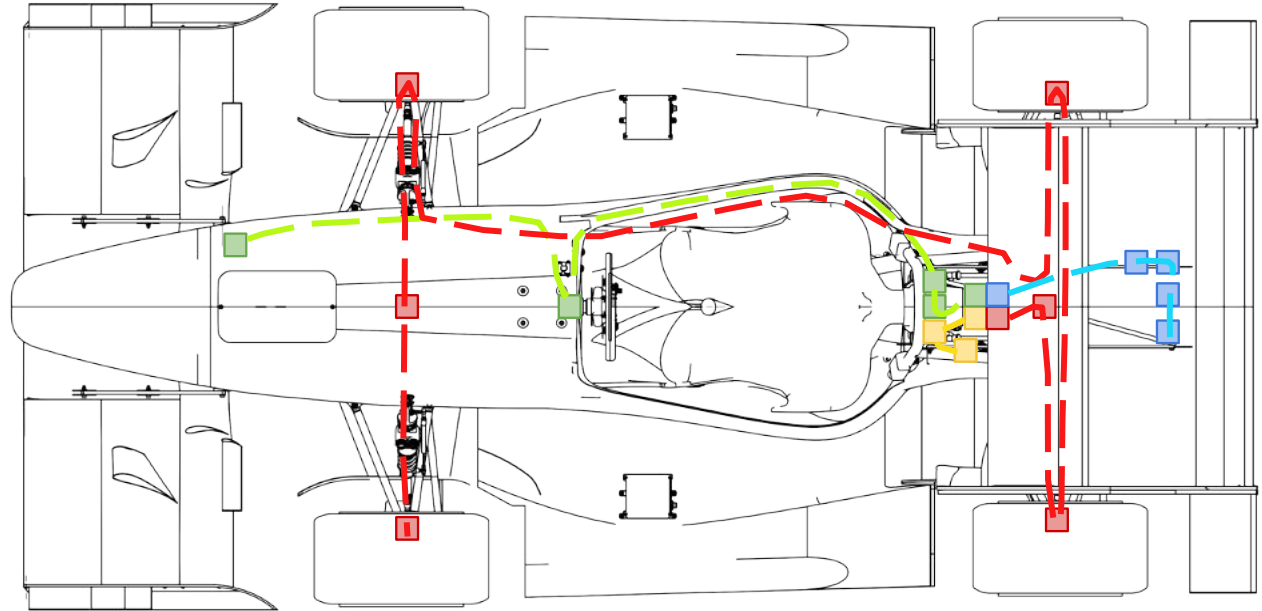


- Automated updates
- Developers always use the latest version



## Controller Area Network

- Node-based
- Predefined messages
- Widely used in automotive

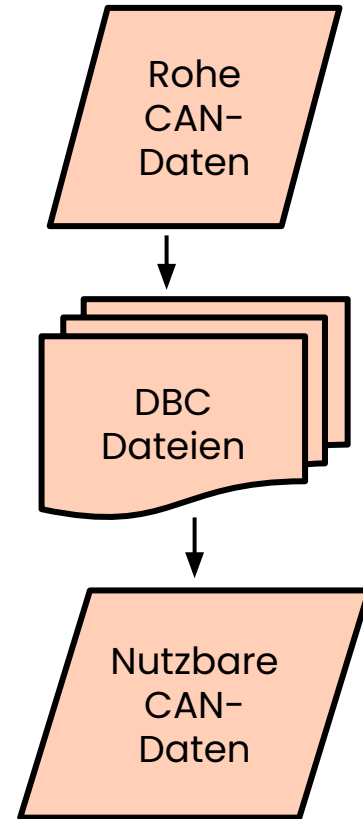


# What the DBC?



## Database CAN

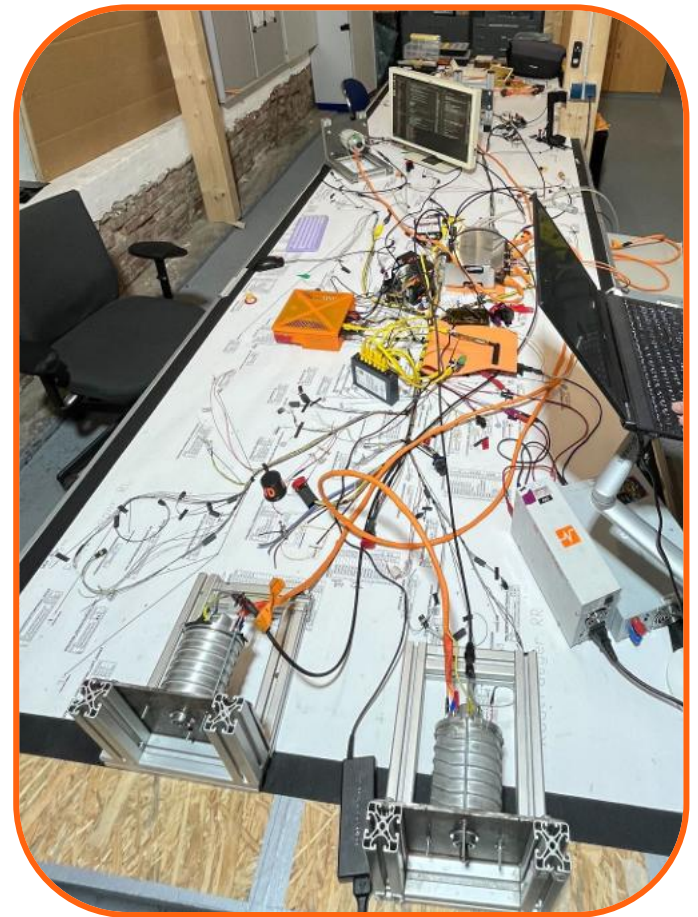
- A widely used standard for predefining CAN messages
- Defines, among other things:
  - ◆ Message name
  - ◆ Information contained in the message
  - ◆ Sender and receiver nodes
- Basis for our CI/CD pipeline



# Motivation



- The process of DBC → PCB and VCU was very manual
- Adding new CAN messages was tedious and prone to human error
- Many custom implementations were required for reading CAN data
- The CAN version on the VCU and PCBs often differed
- Automation is essential!



**ELBFLOTRACE**  
FORMULA STUDENT TEAM TU DRESDEN

# Previous Workflow



## DBC Change

Commit to DBC Git Repo

## Write New Code

Read new message from DBC

Implement on affected nodes

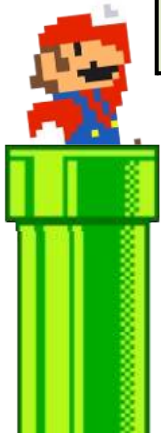
## Test and Publish

Unit Test

Integration Test

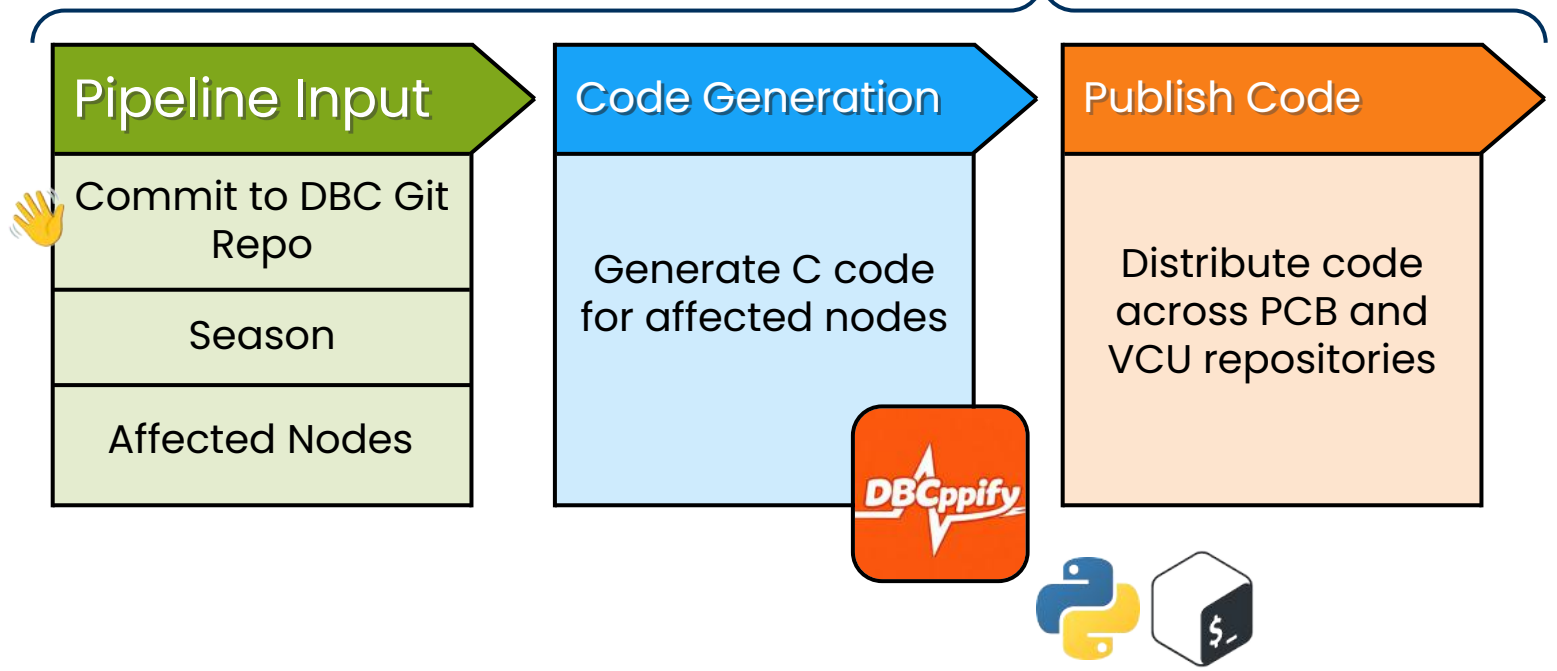
Commit Code

PCB features were unused because CAN implementation was missing on the VCU





# Pipeline Overview pt. 1

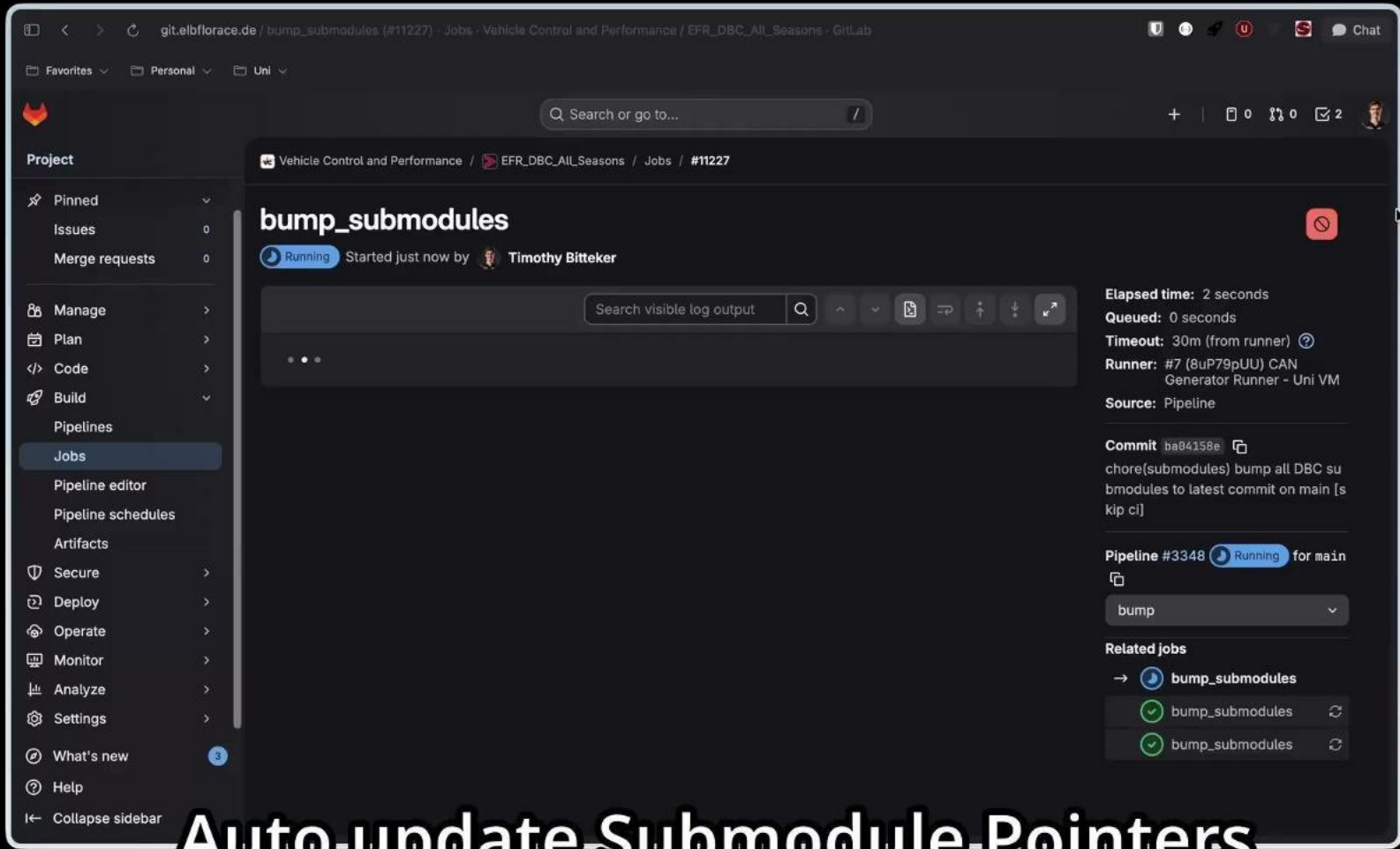




can-dbc-efr18 — -zsh — 200x58

```
(base) timbitteker@Tims-MacBook-Air can-dbc-efr18 % g
```

**Commit DBC change**



# Auto-update-Submodule-Pointers

git.elbforce.de / Pipelines - Vehicle Control and Performance / DBCppify - GitLab

Search or go to...

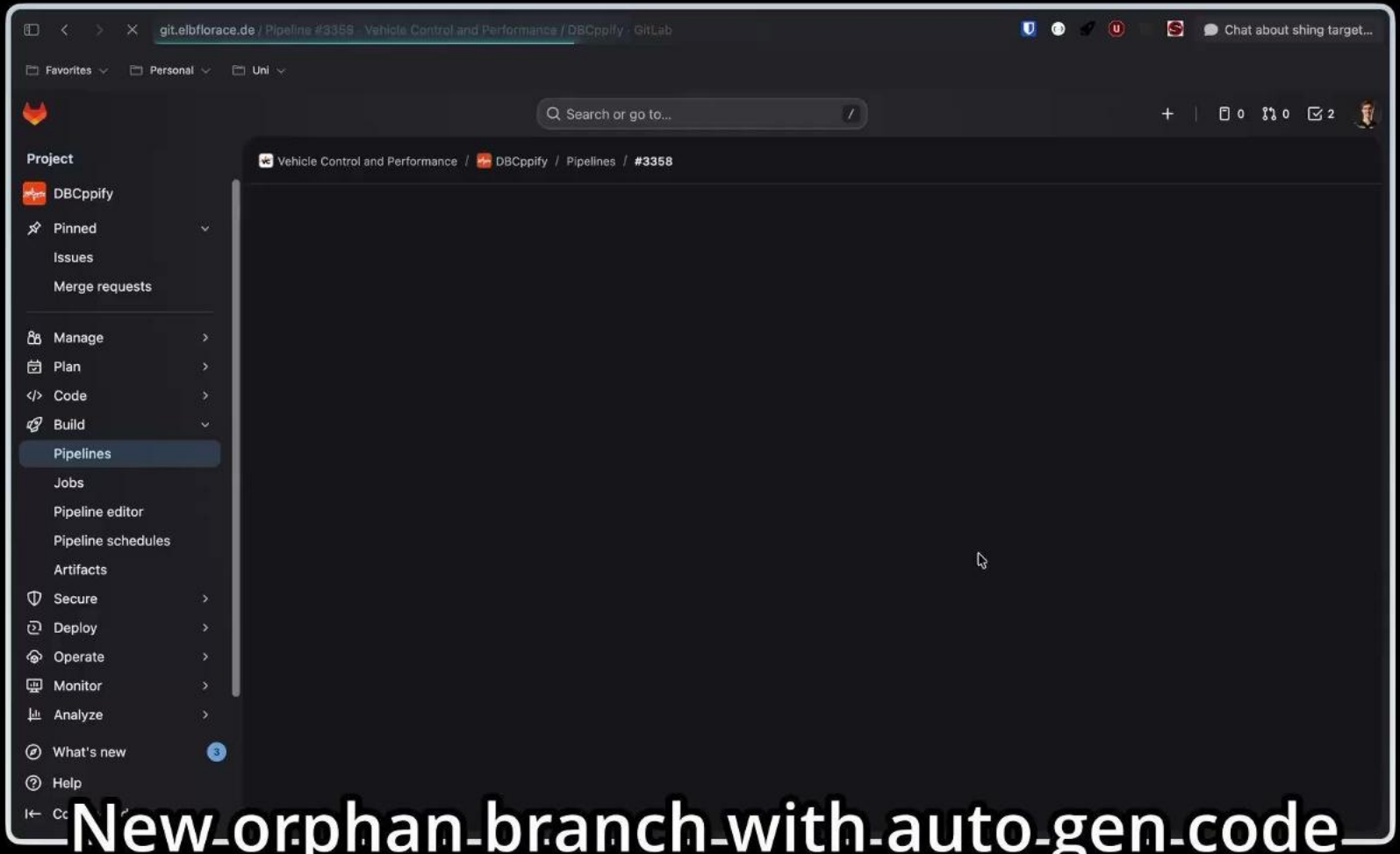
Vehicle Control and Performance / DBCppify / Pipelines

All 86 Finished Branches Tags View analytics Clear runner caches New pipeline

Filter pipelines Show Pipeline ID

Status	Pipeline	Created by	Stages	Actions
Skipped	chore(submodules) bump efr_dbc_a... #3354 main a6ed661b latest branch			Download
Passed 00:01:15 2 minutes ago	chore(submodules) bump efr_dbc_a... #3353 main 079a1769 trigger token branch		2 green checkmarks	Download
Skipped	chore(submodules) bump efr_dbc_a... #3339 main 079a1769 branch			Download
Failed 00:01:34 16 hours ago	Add IMU to non generatable nodes l... #3338 main e1c9f3ba trigger token branch		1 green checkmark, 1 red X	Refresh Download
Skipped	chore(submodules) bump efr_dbc_a... #3330 main a399ce92 branch			Download

PCB-CAN-code-is-generated



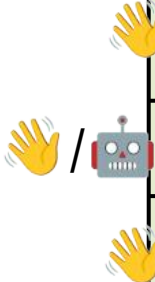
New-orphan-branch-with-auto-gen-code

# Pipeline Overview pt. 2



**Merge & Release**

- Developer merges new files
- Unit test node
- Create release



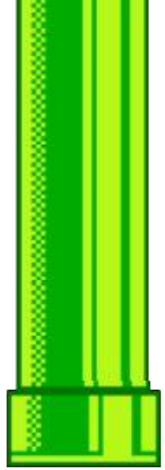
**Compile Code**

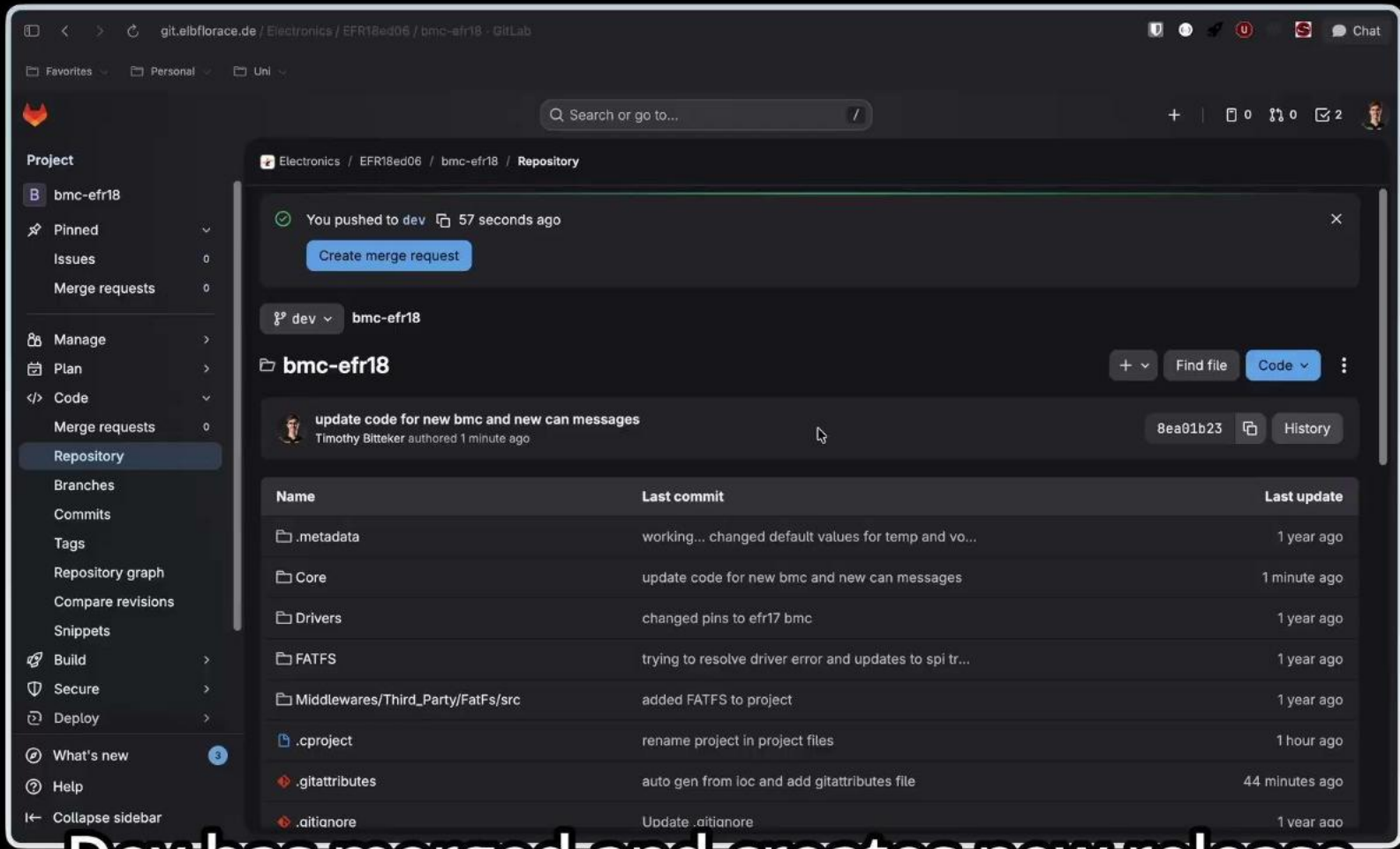
- Compile node code online
- Version compiled code



**Wireless Update**

- Send compiled binary to car via LTE or WiFi
- Flash nodes via CAN bus





Dev has merged and creates new release

git.elbflorace.de / Vehicle Control and Performance / STM32-Headless-Build · GitLab

Vehicle Control and Performance / STM32-Headless-Build

# STM32-Headless-Build

main stm32-headless-build

revert last commit  
Timothy Bitteker authored 1 hour ago

Name	Last commit	Last update
Pipelines		
Jobs	revert last commit	1 hour ago
Pipeline editor	add docu	1 day ago
Pipeline schedules	add rule to only trigger from upstr...	1 day ago
Artifacts		
Dockerfile	make entrypoint.sh executable in ...	5 days ago
README.md	add formating and general info	1 day ago
entrypoint.sh	Merge both test branches to singl...	5 days ago

Project information

- 62 Commits
- 2 Branches
- 0 Tags
- 7.6 MiB Project Storage
- README
- CI/CD configuration
- + Add LICENSE
- + Add CHANGELOG
- + Add CONTRIBUTING
- + Add Kubernetes cluster
- + Configure Integrations

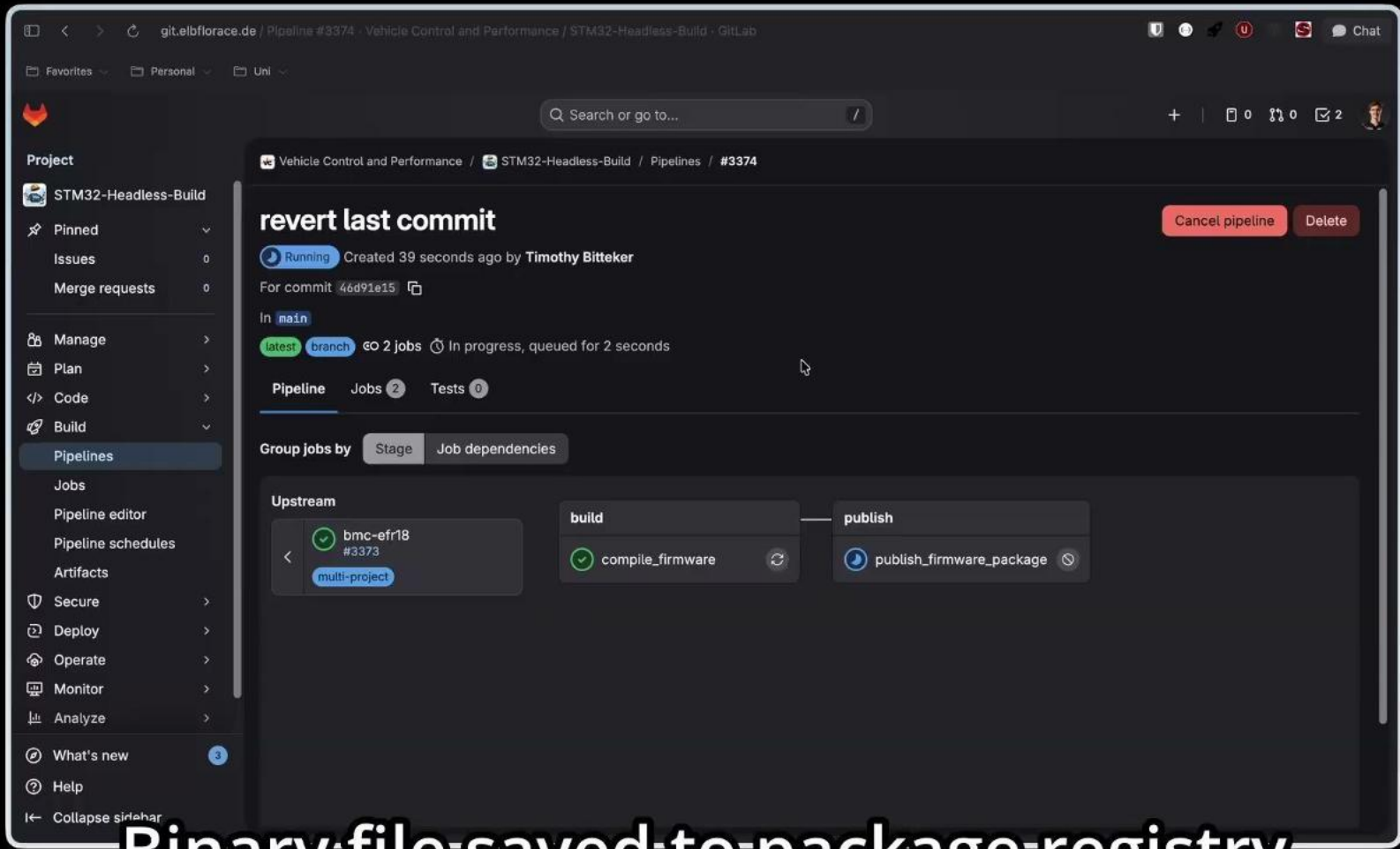
Created on  
March 04, 2026

What's new 3

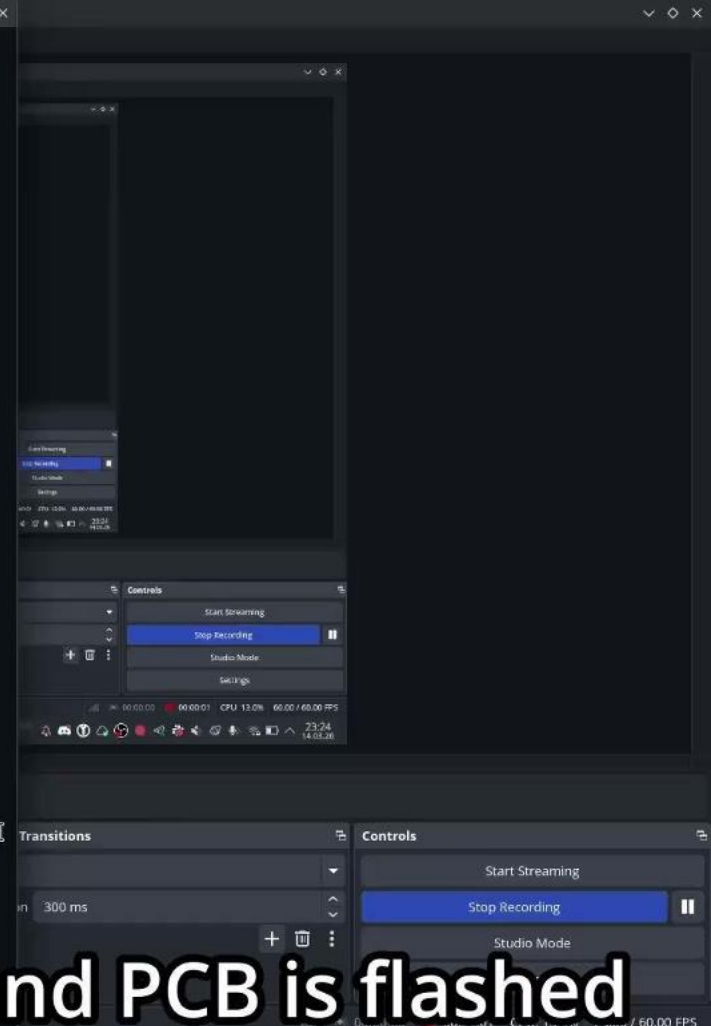
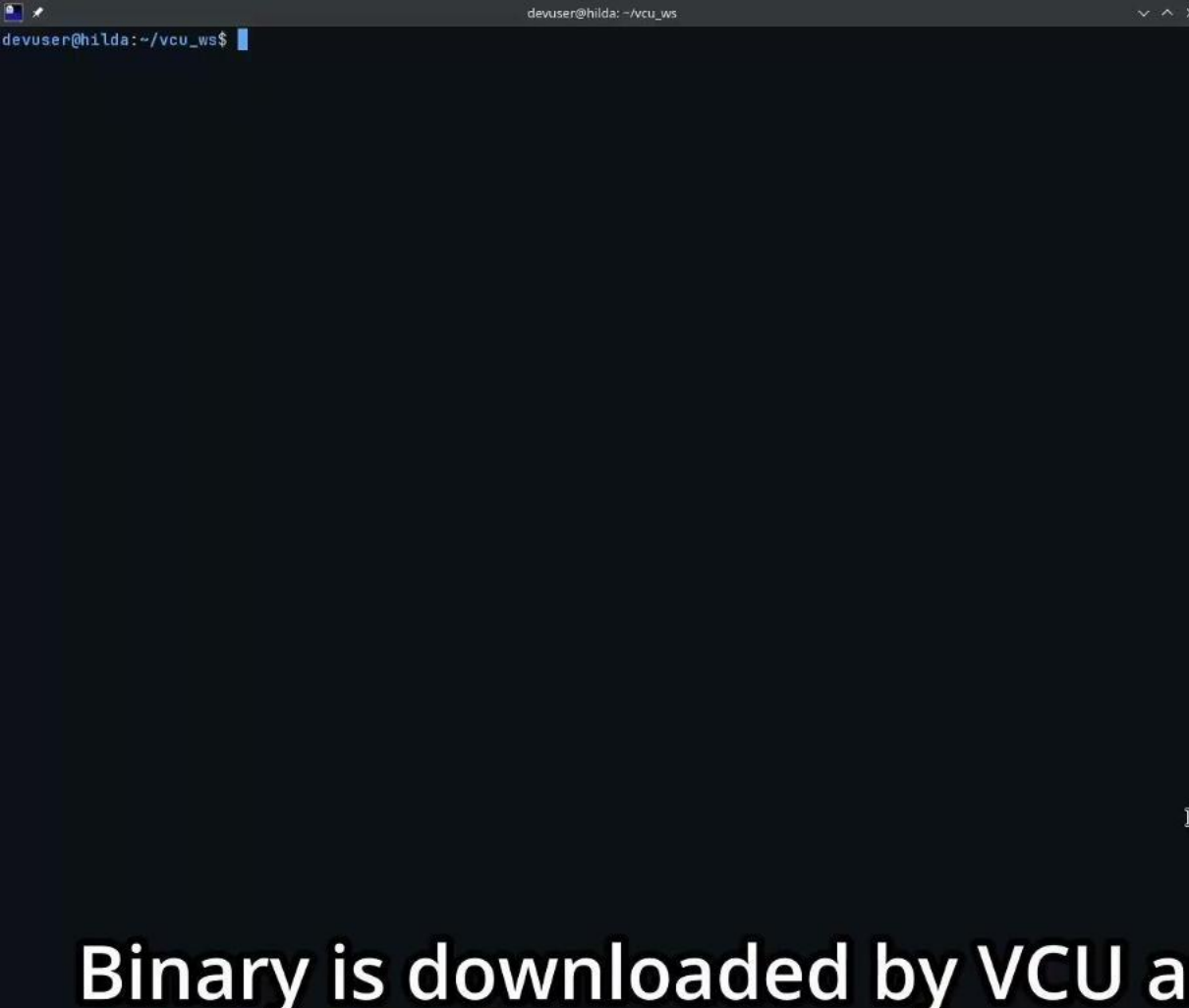
Help

Collapse sidebar

Compile-release-code



Binary-file-saved-to-package-registry



**Binary is downloaded by VCU and PCB is flashed**

# Effects



- No manual CAN implementation on PCB and VCU
- Fewer errors, more time for new features
- Developers always have access to the latest CAN version
- VCU and boards are always on the same CAN version





- System is still very new
- System commissioning
  - ◆ Faster development
  - ◆ Significantly fewer CAN interface problems
- During the testing phase
  - ◆ Use test time more effectively
  - ◆ Resolve problems faster
- Feature
  - ◆ Compatibility checks (PCB and VCU)
  - ◆ Automatic rollback when release tag is removed



# Questions?

# Image Credits



Mario Attributes: © Nintendo

Python Logo: © Python Software Foundation

Docker Logo: © Docker Inc.

CubeIDE Logo: © STMicroelectronics

Gitlab Logo: © GitLab Inc.

OpenBLT Logo: © Feaser

Bash Logo: © Free Software Foundation

Cloud Upload Icon: by Cassandra Cappello from Noun Project (CC BY 3.0)