

CDEventsCon

CI/CD for Autonomous Drive and Other Complex Systems

"Nobody knows how to build a car..."

CDEventsCon

CI/CD for Autonomous Drive and Other Complex Systems
(especially using events)

Erik Sternerson, doWhile

In this talk

- **CD/CD for autonomous drive**
- **Transitioning to events**
- **Future**

Building Complex Systems

Integrate

Continuously

Continuous ... Integration ... ?

CDF: Continuous Integration

"the practice of merging all developer working copies to a shared mainline several times a day"

"Our" Continuous Integration

"frequently and iteratively integrate components together in increasingly complex compositions"

Before events

Before events

- **Releases => File + e-mail**
- **Pipelines => Jenkins server per team**
- **Verification results => "Known by the team"**

Enter events

First: Infrastructure

Infrastructure

- **A vocabulary (created/updated)**
- **A transport (RabbitMQ)**
- **A triggering system (we DIYed it!)**

Releases

The release event

- Release identifier (UUID)
- Release version
- File location

Pipelines

The activity event

- **Activity name**
- **Chain ID**
- **Result**

Verification results

The confidence label event

- Release reference
- Confidence Label
- Result
- Links to verification

Hang on ... "confidence label"?

Confidence labels

- **Encourages teams to plan and communicate quality**
- **"All verification done" => "The quality I need"**

Events: What do we gain?

Automation without centralization

Observability

Collaboration without causation

What will be better with CDEvents

- Glue code => Interoperability**
- Detours => Proper path**

Key takeaways

**CI/CD is used outside of code and
cloud**

Complex / Concrete / Shared

Working on shared solutions

Thank you!

Questions?

Erik Sternerson

doWhile / Volvo Cars / Volvo Autonomous Solutions