Eclipse iceoryx™ Speeds up your SDV

EclipseCon Community Day - Automotive@Eclipse
24 October 2022

Simon Hoinkis
Eclipse iceoryx Speeds up your SDV

Agenda

1. Why does inter-process communication (IPC) matter?
2. From AUTOSAR Classic to AUTOSAR Adaptive
3. Why is iceoryx the perfect match for your SDV?
4. How to integrate iceoryx?
   a. Generic template
   b. AUTOSAR Adaptive ara::com
   c. Rust binding
5. Summary and outlook
Why does IPC matter?
Why does IPC matter?

Eclipse iceoryx Speeds up your SDV
From AUTOSAR Classic to AUTOSAR Adaptive
Eclipse iceoryx Speeds up your SDV

Problem: AUTOSAR Classic copies data for every subscriber

Diagram:
- Sense App
- Middleware
- Copy
- Plan App A
- Plan App B
- Latency & Runtime vs. Message size
Eclipse iceoryx Speeds up your SDV

Solution: zero-copy API

1) Allocate memory
2) Write data
3) Send sample

Latency & Runtime
- Message size

Sense App
Shared Memory Pool
- Plan App A
- Plan App B
Eclipse iceoryx Speeds up your SDV

The history of Eclipse iceoryx

**2009 - 2012**
- Robert Bosch GmbH creates zero-copy middleware *Driver Assistance Data DeliverY* (DADDY) for advanced driver assistance systems (ADAS)
- Running today on millions of microcontroller-based ECUs

**2016**
- Bosch contributes DADDY’s zero-copy API to AUTOSAR Adaptive together with BMW (ara::com)
- Bosch starts development of a POSIX middleware based on DADDY for ara::com

**2019**
- Bosch open-sources the POSIX middleware as Eclipse iceoryx
  - First embedded ECU code on Eclipse
  - Most popular Eclipse Automotive project (951⭐️)
Why is iceoryx the perfect match for your SDV?
Eclipse iceoryx Speeds up your SDV

Why is iceoryx the perfect match for your SDV?

- Everyone needs fast IPC
  - Why reinvent the wheel?
  - Why waste money?
  - Use iceoryx 😊

- iceoryx is developed since 2019 as open-source with an automotive-centric community e.g.
  - ETAS GmbH
  - Robert Bosch GmbH
  - Apex.AI Inc.
  - Avin Systems
  - Continental AG
  - SAIC Motor Ltd.
  - NXP N.V.
  - Lear Corp.
  - Waabi

- Some of those companies build commercial products on top of Eclipse iceoryx

- iceoryx is integrated in Eclipse Cyclone DDS and used in ROS 2 as middleware by a huge community

- Open source is the right choice
  - Transparency builds trust with customers & partners
  - Better quality and less bugs
  - Faster development with the community

- ISO 26262 certification ongoing
  - All iceoryx maintainers have a safety background
  - The work in the open is conducted according to established guidelines and practises e.g.
    - Review checklists
    - Commit traceability
    - Usage of sanitizers
    - Static code analysis
How to integrate iceoryx?
Eclipse iceoryx Speeds up your SDV

Integration with the generic CMake template

- Ready to use and easy to get started
- No need to fight with CMake
- Using FetchContent from CMake 3.11

```cmake
# fetch iceoryx if not found
if(NOT iceoryx_hoofs_FOUND OR NOT iceoryx_posh_FOUND)
    # .. snip ..
    include(FetchContent)
    FetchContent_Declare(
        iceoryx
        GIT_REPOSITORY https://github.com/eclipse-iceoryx/iceoryx.git
        GIT_TAG v2.0.2
    )

https://github.com/eclipse-iceoryx/iceoryx-project-template
```
Eclipse iceoryx Speeds up your SDV

Integration in AUTOSAR Adaptive ara::com

Skeleton App
- MinimalSkeleton
  - EventPublisher
  - FieldPublisher
  - MethodServer

Proxy App
- MinimalProxy
  - EventSubscriber
  - FieldSubscriber
  - MethodClient
  - ServiceDiscovery

The whole API is zero-copy!

https://github.com/eclipse-iceoryx/iceoryx-automotive-soa
Eclipse iceoryx Speeds up your SDV

Integration into Rust projects

[dependencies]
iceoryx-rs = "0.1"
...

use iceoryx_rs::marker::ShmSend;

#[repr(C)]
#[derive(Default)]
pub struct Counter {
    pub counter: u32,
}

unsafe impl ShmSend for Counter {}

use iceoryx_rs::PublisherBuilder;
use iceoryx_rs::Runtime;

mod topic;
use topic::Counter;

fn main() -> Result<(), Box<dyn Error>>
{
    Runtime::init("publisher");
    let publisher = PublisherBuilder::<Counter>::new("Foo",
        "Bar", "Blubb").create()?;
    ...
}

https://crates.io/crates/iceoryx-rs
Summary and outlook

Eclipse iceoryx Speeds up your SDV

Summary

● iceoryx has a long history and its predecessor is used in millions of cars

● Development of iceoryx heavily influenced AUTOSAR Adaptive and its zero-copy API became an industry standard

● The iceoryx community is looking forward to working with the SDV community and speed up their frameworks

Outlook

● More ROS 2 support is coming
  ○ rmw_iceoryx will be updated to Blueberry
    ■ https://github.com/ros2/rmw_iceoryx

● ISO 26262 certification efforts will continue
  ○ Expect many more patches going upstream
Questions?
Eclipse iceoryx Speeds up your SDV

Backup: State of the ISO 26262 certification

- iceoryx_hoofs and iceoryx_posh will get certified
  - hoofs: Handy Objects Optimised For Safety
  - posh: POSIX SHared memory

- Started bottom-up with iceoryx_hoofs
  - Addressing AUTOSAR C++14 ruleset violations with Axivion Suite in the upstream repository
  - Some work not happening in the upstream repository e.g.
    - Requirements elicitation
    - MC/DC tests

Overall certification progress

20%