Eclipse Cyclone DDS makes ROS 2 Easier, Smaller, Faster

ROS-Industrial Stuttgart
December 11th 2019
bit.ly/eclipsecyclone for this presentation and links

Joe Speed
@JoeSpeeds
linkedin.com/in/joespeed
jspeed12
How to use Eclipse Cyclone DDS

Installing CycloneDDS:
apt install ros-eloquent-rmw-cyclonedds-cpp
(or dashing)

Running ROS with CycloneDDS: Set env variable and run as usual
RMW_IMPLEMENTATION=rmw_cyclonedds_cpp

Confirming RMW: In Eloquent, to confirm which RMW you're using
ros2 doctor --report

bit.ly/eclipsecyclone for this presentation and links
Joe Speed
Field CTO
ROS Technical Steering Committee,
OMG Board member
Made MQTT open, helped OEMs & Tier-1s adopt it
Launched IBM IoT and Watson IoT AutoLab
Likes cars, loves AV mobility for elderly & disabled

ADLINK Technology
1,800 people making rugged Edge AI & IoT HW and SW
Industrial, Robotics, Aerospace, Military, Transportation
co-invented DDS, co-chair the DDS Technical Committee

bit.ly/eclipsecyclone for this presentation and links
The Eclipse IoT community is the open source center of gravity for the Internet of Things.
The Eclipse Foundation - By the Numbers

- 370+ Projects
- 275+ Members
- 1550+ Committers
- 195M+ Lines of Code
- 30 Staff Members
- 10+ Working Groups
Strategic Focus Areas

**IoT & Edge**

enable industry leaders to collaborate on an end-to-end IoT architecture that is secure, flexible, and fully based on open source and open standards.

**Automotive**

provide leading automotive OEMs, their suppliers, and partners with a sustainable, transparent, and vendor-neutral platform to collaborate on open technologies and standards.

**Tools**

Eclipse IDE is the critical development environment for more than 4 million active users. Our community is innovating on the next generation of cloud native developer tools.
Used by Industry Leaders

“No company can realize the IoT on its own... Within the Eclipse Community, through the contribution of many IoT developers, tools and standards are created on an open platform that many companies can benefit from for their IoT applications.”

Stefan Ferber
CEO, Bosch Software Innovations
The briefest of recaps: ROS1 & ROS2

ROS1 created in 2007 by Willow Garage as a platform for robotics research

• Moved into stewardship by the Open Source Robotics Foundation
• Vibrant ecosystem of tools, algorithms and drivers for constructing robots
• Much used for research, education, prototypes

ROS2 is a rewrite of ROS1 driven by Open Robotics, first released 2017

• Addresses the weaknesses in the architecture of ROS1
• Maturing nicely, ecosystem is migrating to it
• Already the choice for mobile robotics, gaining ground in industrial robotics
ROS1 & ROS2 Architecture

[Diagram of ROS1 and ROS2 Architecture]

bit.ly/eclipsecyclone for this presentation and links
DDS is IoT, DDS is ...

An eventually-consistent shared data space atop a publish-subscribe messaging system

DDS is an architecture, not just a wire protocol

bit.ly/eclipsecyclone for this presentation and links
What is Eclipse Cyclone DDS?

a) Eclipse IoT open source project
b) very performant DDS implementation
c) Easy, Small and Fast
d) Loves github issues & PRs!
e) Led by IoT mad scientist & 2nd gen DDS inventor Erik Boasson eboasson

bit.ly/eclipsecyclone for this presentation and links
Footprint of Eclipse Cyclone DDS

Pure C code
Run-time library has minimal dependencies

Compact

• code size can go down to ~0.5MB
• latency testing needs ~0.5MB of heap memory (macOS)
• minimizing footprint does reduce throughput (~0.5x) & increase latency a bit (~5µs)
• even so, still the fastest

Platforms

• Supported: Linux, Windows, macOS, FreeRTOS, x86, Arm, Jetson
• Planned: QNX, possibly VXWorks and others
• Known to work: Solaris 2.6, OpenIndiana (Erik & Morgan are sentimental)

bit.ly/eclipsecyclone for this presentation and links
“The (ROS 2) RAM usage goes from 150MB to 15MB and we basically had no late and no lost messages. This is a huge improvement in terms of performance simply changing the DDS implementation”
Problem Statement

~1000 robots in the same network
~35 topics from each robot
~5 different subnetworks across the globe

We want to be able to individually access any robot's messages from a remote debugging machine
Addressing Scalability

Proposed solution 1:
Use default DDS configuration and a different namespace for each robot

Problems:
ALL robots discover each other!
Slow/unreliable discovery, network saturation, multicast requires forwarding between subnets
Addressing Scalability

Proposed solution 2:
Solution 1 + ROS_DOMAIN_ID

Problems:
Limited numbers of domain IDs. What about assignment? Still considerable network usage and multicast limitations
Addressing Scalability

Proposed solution 3:
Disable multicast discovery. When a remote machine wants to connect to a robot, add the robot’s IP to the initial peers list (unicast)

Problems:
The user needs direct access to the DDS configuration. This can be solved by new APIs.
Solution 3 with CycloneDDS

```
<Domain>
  <Id>any</Id>
</Domain>
<DDS12E>
  <General>
    <NetworkInterfaceAddress>auto</NetworkInterfaceAddress>
    <AllowMulticast>false</AllowMulticast>
    <EnableMulticastLoopback>true</EnableMulticastLoopback>
  </General>
  <Discovery>
    <ParticipantIndex>0</ParticipantIndex>
    <Peers>
      <Peer Address="${ROBOT_IP}:7410"/>
    </Peers>
  </Discovery>
</DDS12E>
```

ROBOT_IP=10.22.22.90 ros2 topic list
Solution 4 with Eclipse Cyclone DDS

Erik eboasson added domainTag to DDSI 2.3 spec, implemented iRobot’s use case - find any Roomba by serial number among ~1,000 robots on the network

DDSI domainTag things talk if domain + tag match

//CycloneDDS/Domain/Discovery/Tag

Text

String extension for domain id that remote participants must match to be discovered. The default value is: ""

bit.ly/eclipsecyclone for this presentation and links
Try iRobot’s test with Eclipse Cyclone DDS

github.com/irobot-ros/ros2-performance

on your RPi2B or other machine

on AWS RoboMaker aws.amazon.com/robomaker

on The Construct’s ROS Dev Studio http://www.rosject.io/l/ebd6221/

bit.ly/eclipsecyclone for this presentation and links
bit.ly/eclipsecyclone + iceoryx = extremely fast ROS!
A True Zero-Copy RMW Implementation for ROS2

The Use Case

automated driving is a data processing chain with a sensor input of up to 10 GB/s
A True Zero-Copy RMW Implementation for ROS2

The Problem

A typical middleware...

- copies when passing messages from the publisher to the middleware
- copies when passing messages from the middleware to the subscriber
- does internally even more copies and/or serialization/deserialization
- does at least n+1 copies for an inter-process-communication with n subscribers

No time to copy and serialize n GB/s while driving!
A True Zero-Copy RMW Implementation For ROS2

The Solution: True Zero-Copy Inter-Process-Communication

True zero-copy means...

- it is an end-to-end zero-copy approach from publishers to subscribers, based on shared memory
- the publisher directly writes to a chunk of memory provided by the middleware
- the middleware passes message references to subscribers and manages their liveliness

zero-copy communication is a must-have for automated driving!
ECLIPSE CLOE
Eclipse CLOE provides simulation middleware and simulation-engine bindings for connecting simulation engines to the "software under test".

PROJECT PROPOSAL

ECLIPSE CYCLONE DDS
Eclipse Eclipse Cyclone DDS is a very performant and robust open-source DDS implementation. Cyclone DDS is developed completely in the open as an Eclipse IoT project. Eclipse Iceoryx is complementary and is being built into Eclipse Cyclone DDS.

VIEW PROJECT

ECLIPSE ICEORYX
Eclipse Iceoryx is a middleware with a zero-copy shared memory approach which is optimized for the huge data inter-process-communication.

VIEW PROJECT

bit.ly/eclipsecyclone for this presentation and links
high velocity sensor fusion

... because race car
Suspension Analysis / Tire Modeling

- CAN Bus over LTE
- Intel RealSense
- Embedded ML
- ADLINK ROScube Jetson Xavier
Demo - Vehicle Dynamic (AI Driver Model)

trained driver model, can currently uses a combination of neural networks to predict with 85% accuracy:

- Steering Angle
- Brake Position
- Throttle Position
- Gear and Speed
AMRs with Eclipse Cyclone DDS

Mobile vision
“Camera on wheels”
Education
Inspection
100kg payloads

bit.ly/eclipsecyclone for this presentation and links
Professional Sports with Eclipse Cyclone DDS

**THE ACE™**

- **Ball Orienter™**
  - Patent-pending system that positions ball orientation for pitch-specific release

- **One-Click Pitch**
  - Practicing against any pitch is simple, upload your data and start

- **Alpha-Tilt™ for Gyro Spin**
  - Ball spin has 3 degrees of freedom, introducing Alpha-Tilt™, enabling accurate trajectory replication
“Eclipse Cyclone DDS has given the fast & precise control of our ROS 2 baseball pitching robot to accurately recreate any throw from any pitcher”

- Rowan Ferrabee
  CTO Trajekt Sports

bit.ly/eclipsecyclone for this presentation and links
9 microcontrollers run Eclipse Cyclone DDS

Eclipse Cyclone DDS is built into components of military vehicles with 32-bit NXP microcontrollers

Intra-vehicle communication!

Inter-vehicle communication?

Can’t say who, so here is a picture of an old jeep


bit.ly/eclipsecyclone for this presentation and links
WiFi happy Eclipse Cyclone DDS

“multicast autotune” – not really, but sort of 😁

Sped Nav2 WiFi bringup while others learned how

“just works” in hospital with terrible multipath issues, no config needed

image source: edmtips.com/how-to-autotune-properly

bit.ly/eclipsecyclone for this presentation and links
Worked reliably in Last Vegas Convention Center despite 666 WiFi APs “a devilish number” 😈

Fast Serialization soon in Eclipse Cyclone DDS

End-to-end preliminary results vs previous Eclipse Cyclone DDS version used by iRobot:

- PointCloud512k - **2.1x** faster, **49%** less CPU
- Struct32k - **5.6x** faster, **80%** less CPU
- Array60k – little faster, bit less CPU

Thanks to #1 Eclipse Cyclone DDS fan and top contributor Dan Rose [rotu](https://github.com/rotu) at Rover Robotics measured with [RoverRobotics-forks/performance_test/releases/](https://github.com/RoverRobotics-forks/performance_test/releases/) forked from [ApexAI/performance_test](https://github.com/ApexAI/performance_test)

[bit.ly/eclipsecyclone](https://bit.ly/eclipsecyclone) for this presentation and links
Cyclone DDS work in progress ...

Functionality
• Transient/persistent data support
• Improved coverage of DDS QoS
• DDS Security
• DDS XTypes

Performance, predictability
• Static memory allocation
• Zero-copy with Eclipse iceoryx
• Time-Sensitive Networking support

Integration
• Language bindings: C++, Python, Java, Haskell, Rust, Ada, ...

bit.ly/eclipsecyclone for this presentation and links
Middleware Vendor benchmarks?

Just try it!

bit.ly/eclipsecyclone for this presentation and links

Source dharmapunxnycom/2013/08/dont-believe-hype.html
ros2/rmw_cyclonedds try it

apt install ros-eloquent-rmw-cyclonedds-cpp
(or dashing)
RMW_IMPLEMENTATION=rmw_cyclonedds_cpp

bit.ly/eclipsecyclone for this presentation and links
iRobot ROS 2 performance ROScon `19 slides
iRobot ROS 2 performance test tool for mobile robots
ApexAI/performance_test RMW and DDS evaluation tool
Eclipse Cyclone DDS makes ROS 2 Easier, Faster, Smaller

ROS-Industrial Stuttgart
December 11th 2019

bit.ly/eclipsecyclone for this presentation and links

Joe Speed
@JoeSpeeds
linkedin.com/in/joespeed
jspeed12