

# 2のZenoh対応

# Zenohの2対応



高瀬 英希  
(東京大学)

[takasehideki@hal.ipc.i.u-tokyo.ac.jp](mailto:takasehideki@hal.ipc.i.u-tokyo.ac.jp)



Lab#8, IPC, IST, UTokyo  
Computing System Laboratory

# Affiliation



# Hobby

1,728 contributions in the last year

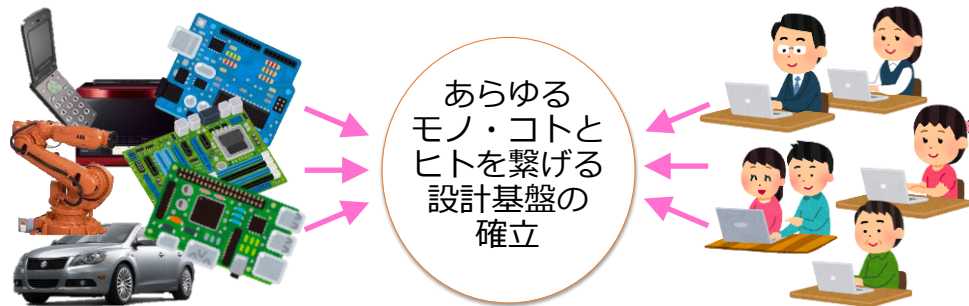
|     | May | Jun | Jul | Aug | Sep | Oct | Nov | Dec | Jan | Feb | Mar | Apr | May |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Mon |     |     |     |     |     |     |     |     |     |     |     |     |     |
| Wed |     |     |     |     |     |     |     |     |     |     |     |     |     |
| Fri |     |     |     |     |     |     |     |     |     |     |     |     |     |

Less [ ] More [ ]

@takasehideki



## 組み込み/IoTコンピューティング基盤を支えるプラットフォーム技術と設計方法論

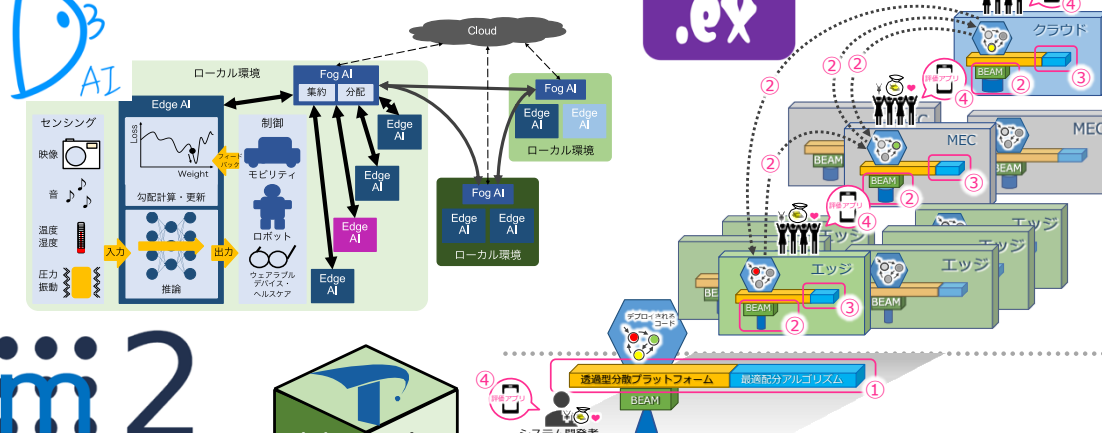
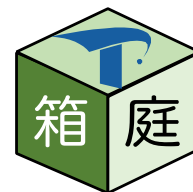


最適化

設計技術

Mission

良いモノを誰でも楽に  
つくることができる世界へ



Recent R&D

# はじめに・おことわり

---

- 本発表は「新しいパッケージやフレームワーク、ツールの提案や紹介」に該当します
  - 発表者は対象OSSの開発者やメンテナではなく、ただのいちユーザーです
  - 個人の意見や考えによる発表内容であり、あらゆる組織を代表していません
- 2024年9月時点での開発状況と発表者の調査結果に基づいています
  - 必ず最新の一次情報をご確認ください
  - どちらのOSSも(あなた次第で!!)これからもどんどん成長していきます！
- 本発表におけるデモの動作手順はGitHubで公開しています  
[https://github.com/takasehideki/rosconjp24\\_demo](https://github.com/takasehideki/rosconjp24_demo)








# Agenda

---

- 本発表の目的

- ROS 2エコシステムの成長を促進する2者のOSSプロジェクトについて、最新の開発状況を **日本語**で 発信する
- 双方のメリットや使いどころを整理してこれらを共有する
- 日本からの情報発信や各OSSプロジェクトへの貢献機会を増やす

- 発表の内容と流れ

-  **Zenoh** とはなにか？
-  **Zenoh** の  対応 : [eclipse-zenoh/zenoh-plugin-ros2dds](https://eclipse-zenoh/zenoh-plugin-ros2dds)
-  の  **Zenoh** 対応 : [ros2/rmw\\_zenoh](https://ros2/rmw_zenoh)

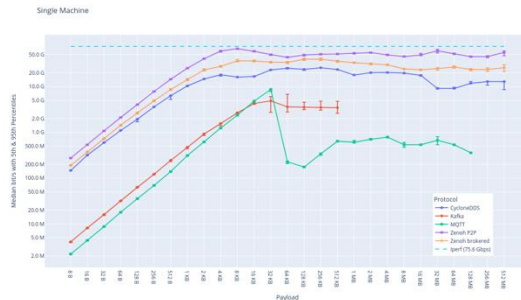


# Zenoh

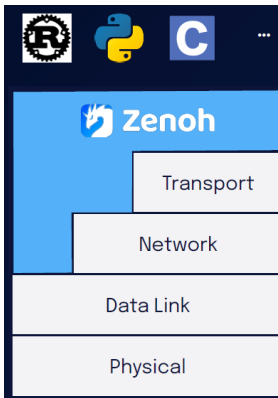
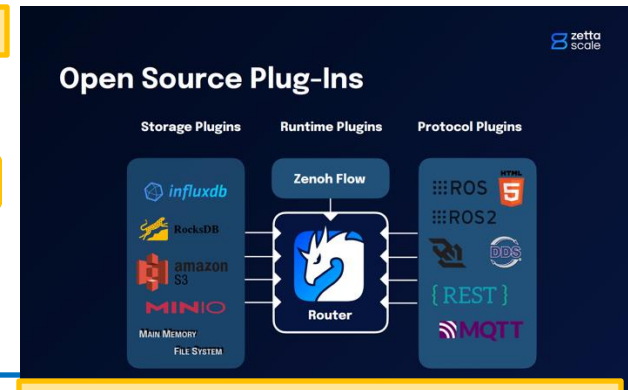
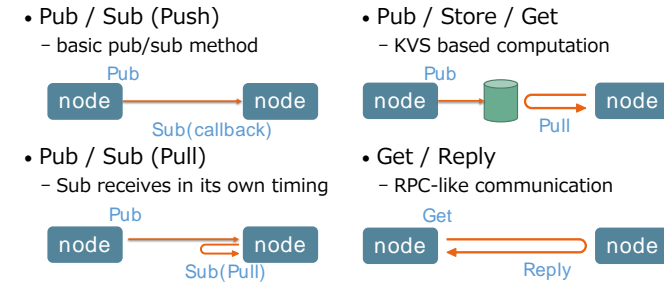
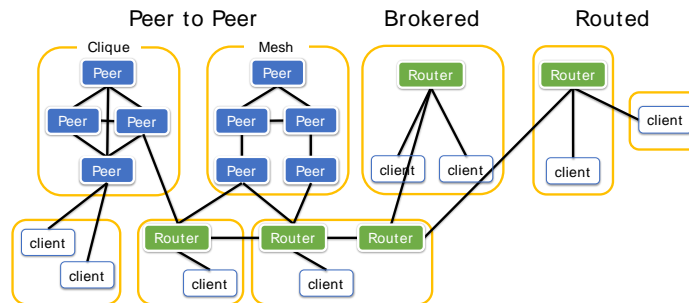
- Zero Overhead Pub/Sub, Store/Query and Compute.
  - **Z**ero **n**etwork **o**ver**h**ead protocol
  - ネットワーク内ではDDSライク, ネットワーク間ではMQTTライクな通信方式
  - 開発主体: [ZettaScale Technology](https://www.zettascale.com/) / GitHub: [eclipse-zenoh/zenoh](https://github.com/eclipse-zenoh/zenoh)

## 5つのポイント

- ① とにかくかるい!
- ② なんでもつかえる!
- ③ いろいろつながる!
- ④ よしなにしゃべれる!
- ⑤ どこでもいごく!



<https://zenoh.io/blog/2023-03-21-zenoh-vs-mqtt-kafka-dds/>



<https://connpass.com/event/304464/presentation/>



# Zenoh プログラミングモデル

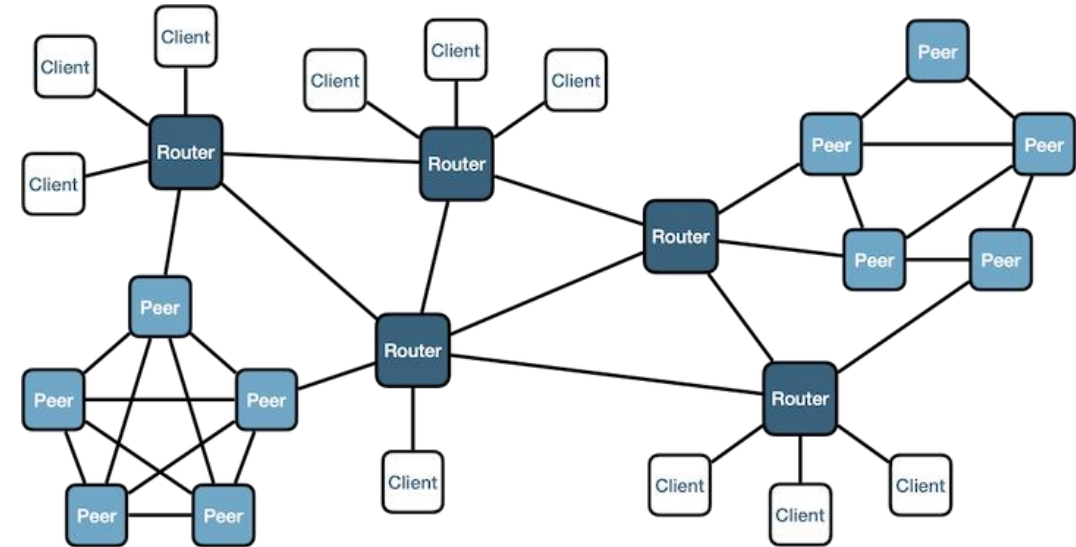
- Resource
  - (key, value)ペアの名前付けされたデータ
  - 例 : home/kitchen/sensor/temp, 21.5  
home/kitchen/sensor/humi, 0.67
- Key expression
  - keyの集合表現 \* \$\* \*\* で指定可能
  - 例 : home/kitchen/sensor/\*  
home/\*\*/temp
- Selector
  - resource 集合を特定する表現
  - 例 : home/\*/sensor/air?\_where=co2>12&\_project=humidity
- zenoh::open() / session.close()
  - zenohセッションを開始/終了する
  - 引数等でネットワーク構成を指定する
- session.put()
  - key, value のペアを1回出版
- session.declare\_publisher()
  - 出版者として宣言する
- session.declare\_subscriber()
  - 購読者として宣言する
  - recv\_async().await するか購読時に実行されるコールバック関数を指定する





# Zenoh ネットワークモデル

- Peer
  - Zenohノードとしての基本形式
  - ローカルネットワーク内では通信相手を自律的に探索できる (like DDS)
- Client
  - ルータを介して通信する構成
- Router (zenohd)
  - 通信を仲介するモジュール
  - [JSON](#)で様々な構成を設定できる
- Scouting: 通信相手の探索方法
  - multicast : ネットワーク内を自律的に探索 (via 224.0.0.224:7446)
  - gossip : routerのIPとポート番号(7447)を指定





# Zenoh 詳しくは??

- ZettaScale CEO/CTO直伝！Zenoh完全理解セミナー！ @2023/12/19

- [スライド資料](#)

- [YouTubeアーカイブ](#)

- [speakerdeck.com/takasehideki](https://speakerdeck.com/takasehideki)

- [なんしかいろいろつながるZenohの紹介](#) @FA\_Study#17 2024/02/03

- [ZenohのROS 2向け対応の現状と方向性](#) @rosjp#54 2024/02/09

- [すべてが #Zenoh になる](#) @robosemi#44 2024/03/15

- [すべてが #Zenoh になる ~柔軟にして軽量~](#) @SWEST24 2024/08/30

- [ZenoHex完全に理解した \(しらんけど:D](#) @daimon.ex 2024/07/19

- ✓ [YouTubeアーカイブ](#)







# Zenoh の ::: 2 対応

- Zenoh で ROS 2 が つながる

- ROS 2/DDS のメッセージを Zenoh で 橋渡し

- ✓ 主たる開発 / サポート 対象は CycloneDDS

- ✓ 現在の最新版は 1.0.0-beta.3

- [zenoh-plugin-dds](#) から派生してより ROS 2 フレンドリーに!

- ✓ plugin: router にロードして使う

- ✓ bridge: スタンドアロンで動作する

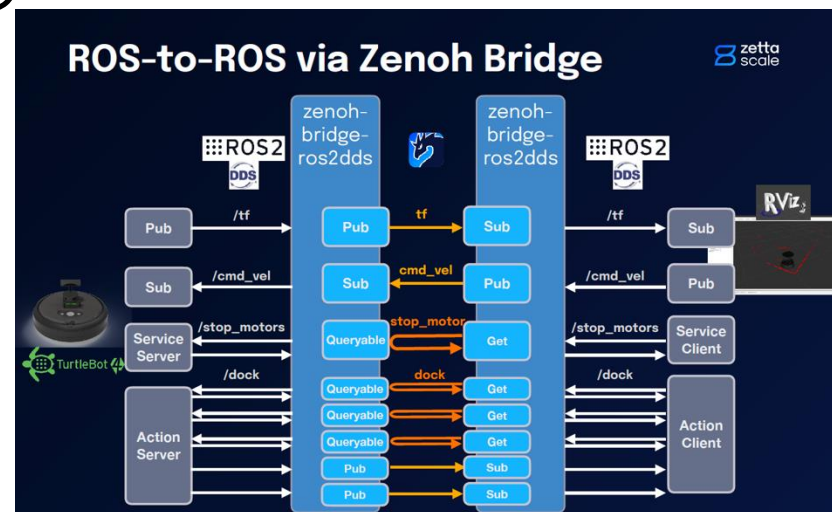
- core maintainer

- Julien Enoch ([@JEnoch](#) at ZettaScale)

eclipse-zenoh/zenoh-plugin-ros2dds

A Zenoh plug-in for ROS2 with a DDS RMW.

15 Contributors   1 Used by   112 Stars   26 Forks



ROSCon 22 Kyoto

Zenoh

ROSCon 2022 - October, 20th 2022 - Kyoto

How to Make ROS 2 Work at any Scale and Integrate with Anything

JULIEN ENOCH

video archive sponsored by: AMD

zetta scale

open robotics



# Zenoh の ::: 2 対応

eclipse-zenoh/zenoh-  
plugin-ros2dds

A Zenoh plug-in for ROS2 with a DDS RMW.



15 Contributors 1 Used by 112 Stars 26 Forks

## • 主な使用方法

– ネットワークの適切な設定 (強く推奨)

✓ ROS\_LOCALHOST\_ONLY=1 やシステムごとの ROS\_DOMAIN\_ID の設定 など

✓ トラフィックの重複やループバックを抑止するため

– 双方での zenoh-bridge-ros2dds の起動

✓ DOMAIN\_ID 内のノードとトピック等を検出

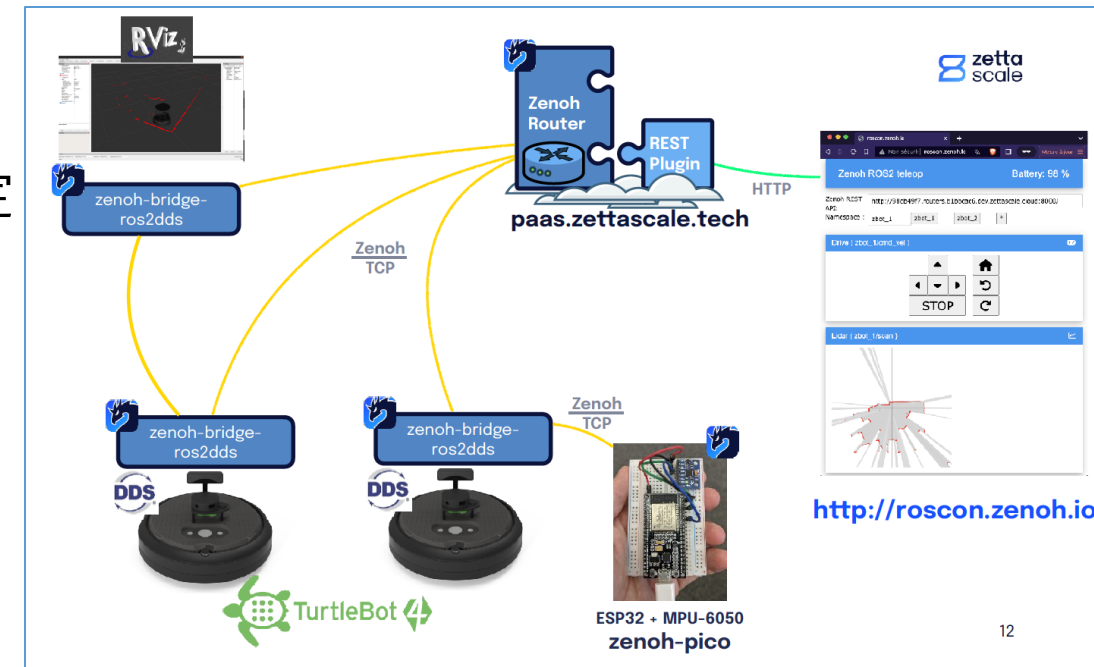
✓ 片方からはエンドポイントのIPアドレスを指定

## • ユースケース

– 異なる ROS\_DOMAIN\_ID での相互接続

+ 特定トピックの allow or deny

– **NAT越え** : 別ネットワーク同士の接続





# Zenoh の ::: 2 対応

論よりRUN!!

zenoh-bridge-ros2dds on local

zenoh-bridge-ros2dds on cloud

ローカル側

クラウド側

allowed topics

not allowed (denied) topics

turtlesim

The screenshot displays a multi-terminal environment. On the left, a terminal window shows the local side logs for 'zenoh-bridge-ros2dds', including messages like 'Discovered ROS Node /teleop\_turtle' and 'Discovered ROS Node /\_ros2cli\_2230'. Below this, a terminal window shows the command 'ros2 run turtlesim turtle\_teleop\_key' and its output, listing allowed topics such as 'linear\_velocity' and 'angular\_velocity'. On the right, another terminal window shows the cloud side logs, including 'Remote bridge fb5e859fd99a1e18d9e62fa85b07d1a8 announces Publisher turtle1/cmd\_vel' and 'Remote bridge fb5e859fd99a1e18d9e62fa85b07d1a8 announces Subscriber turtle1/pose'. Below this, a terminal window shows the command 'ros2 run turtlesim turtlesim node' and its output, including 'Starting turtlesim with node name /turtlesim' and 'Spawning turtle [turtle1] at x=[5.544445], y=[5.544445], theta=[0.000000]'. At the bottom right, a 'Turtlesim' window displays a 2D simulation of a turtle on a blue field with a white path.

# 2 の Zenoh対応

The screenshot shows a Discourse forum post from 'clalancette' dated Sep 28. The post title is 'ROS 2 Alternative middleware report'. The content discusses the development of an alternative middleware (RMW) for ROS 2, mentioning that the core team is developing an alternative RMW alongside existing DDS RMWs. The goal is to create a Tier-1 RMW for ROS 2, with a short-term goal of having a source-installable RMW for the community. The post includes a link to a PDF report: '2023-09 ROS 2 RMW alternate.pdf (1.2 MB)'. The post has 41 likes and 2 replies. The forum interface includes a sidebar with categories like 'General', 'Jobs', and 'Next Generation ROS', and a list of tags such as 'galactic', 'humble', and 'noetic'.

<https://discourse.ros.org/t/ros-2-alternative-middleware-report/33771>



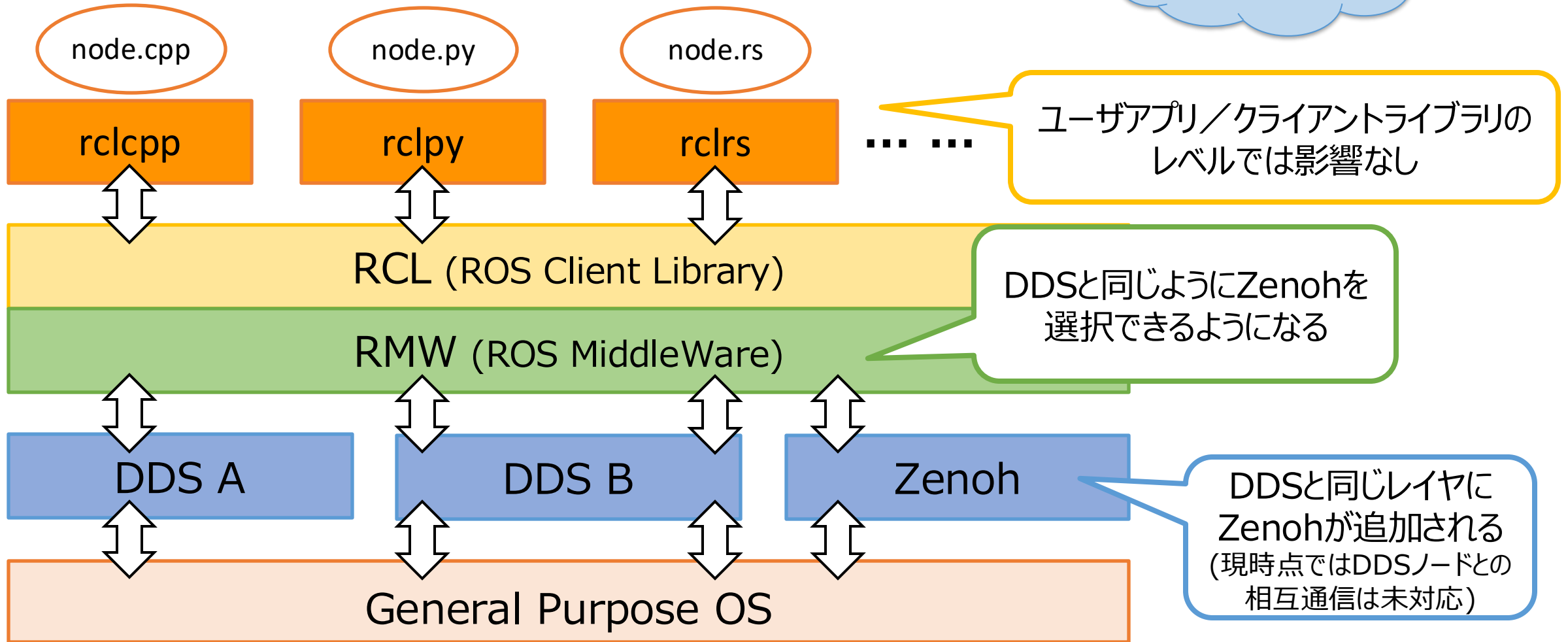
## Conclusion

Requirements from ROS 2 users were gathered, and middleware options that are available today were investigated. The research has concluded that Zenoh best meets the requirements, and will be chosen as an alternative middleware. Zenoh was also the most-recommended alternative by users. It can be viewed as a modern version of the TCPROS implementation, and meets most of the ROS 2 requirements. There are still a number of design decisions to be made regarding this implementation; those details will be discussed on <https://discourse.ros.org> as development begins.

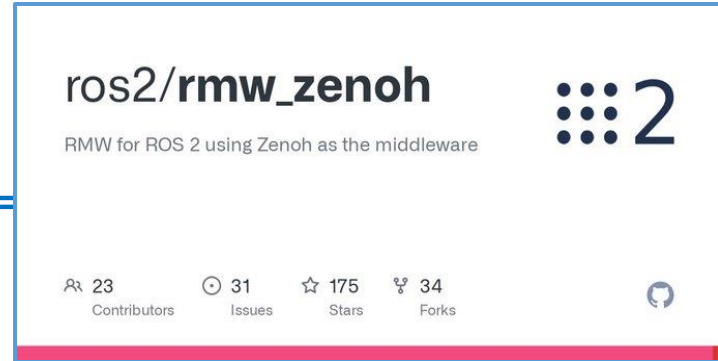


# 2の Zenoh対応

つまり、  
こうなる！



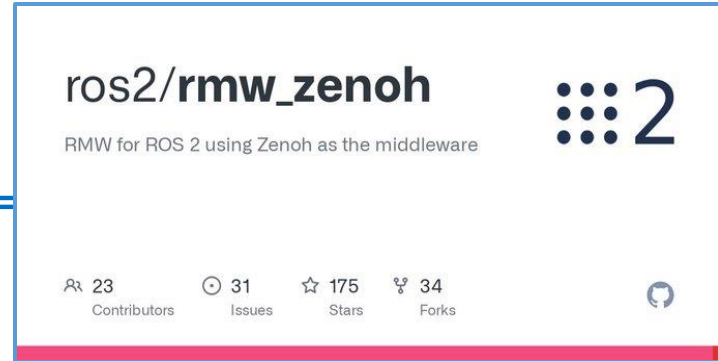
# ROS 2 の Zenoh対応



- **ROS 2**が**Zenoh**でつながる
  - DDS以外では初の公式のRMWサポート対象
  - Jazzy では preview 対応に
    - ✓ Rolling/Iron もサポート対象として開発中（Humble 以前は対象外）
    - ✓現状では git clone && colcon build が必要
    - ✓とはいえ、今からすぐに試せる！
- core maintainers
  - Yadunund Vijay ([@Yadunund](#) Iron ROS Boss)
  - Chris Lalancette ([@clalancette](#) ROS 2 Technical Lead)



# 2の Zenoh対応



## • 実装の仕組み

- [zenoh-c](#) (C API for Zenoh) の binding として実装
  - ✓ 階層 : RMW(C++) |> Zenoh-API(C) |> Zenoh(Rust)
  - ✓ Zenoh Router はC++で(再)実装
- 基本は Zenoh Router で discovery |> P2P で pub/sub

## • 主な使用方法

- ```
$ colcon build
```

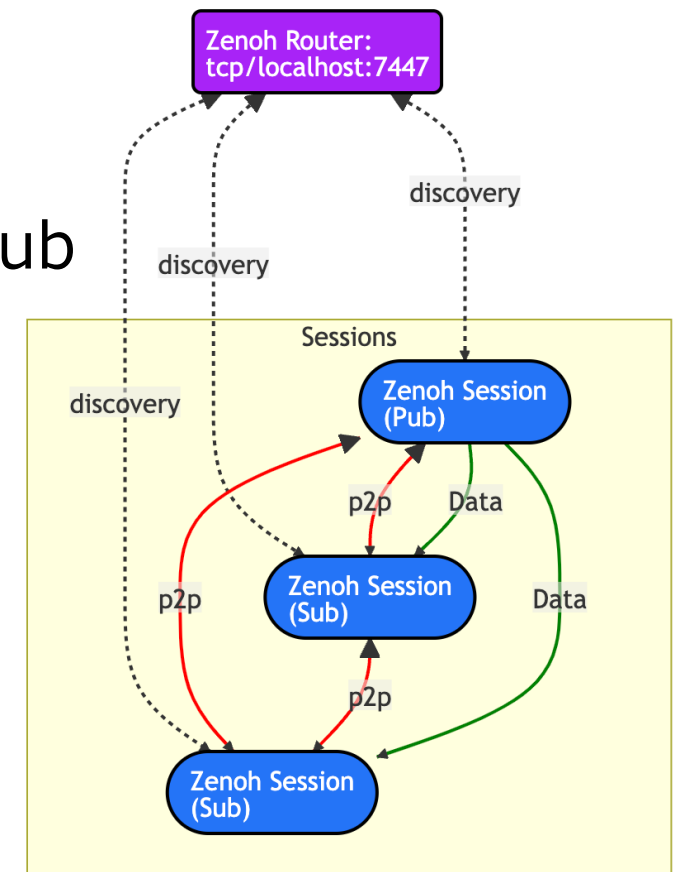
```
$ source ~/ws_rmw_zenoh/install/setup.bash
```

```
$ ros2 run rmw_zenoh_cpp rmw_zenohd
```

  - ✓ デフォルトでは router の起動が必要 (将来的には不要に?)
- ```
$ source ~/ws_rmw_zenoh/install/setup.bash
```

```
$ RMW_IMPLEMENTATION=rmw_zenoh_cpp ros2 run ...
```

発表時の記載に誤りがありました  
ご容赦ください



# 2の Zenoh対応

Default Configuration for Zenoh Sessions:

| Config          | Zenoh Session      | Zenoh Router |
|-----------------|--------------------|--------------|
| Mode            | Peer               | Router       |
| Connect         | tcp/localhost:7447 | -            |
| UDP Multicast   | Disabled           | Disabled     |
| Gossip Scouting | Enabled            | Enabled      |

- ネットワーク構成のカスタマイズ
  - デフォルトでは Gossip Scouting による探索
  - 環境変数 `ZENOH_{ROUTER,SESSION}_CONFIG_URI` で任意の構成（JSONファイル）を指定できる
    - ✓ デフォルト構成：[rmw\\_zenoh\\_cpp/config/\\*.json5](https://github.com/eclipse-zenoh/rmw_zenoh_cpp/blob/master/config/*.json5)

## • multicast scoutingへの変更

```
$ export ZENOH_ROUTER_CHECK_ATTEMPTS=-1
```

```
scouting: {  
  multicast: {  
    enabled: true,  
    ...  
  },  
},
```

## • 複数ホスト間での接続

```
connect: {  
  endpoints: ["tcp/192.168.1.1:7447"],  
},
```

マルチホスト間を接続できる！  
もちろんクラウドとも！





# 2の Zenoh対応

論よりRUN!!

```
iTerm2 Shell Edit View Session Scripts Profiles Toolbelt Window Help
root@6dc540c03dee: ~/rosconjp24
root@6dc540c03dee:~/rosconjp24# source ~/ws_rmw_zenoh/install/setup.bash
root@6dc540c03dee:~/rosconjp24# ros2 run rmw_zenoh_cpp rmw_zenohd
2024-09-24T07:49:21.953048Z INFO ThreadId(02) zenoh::net::runtime: Using ZID: 4d04140f44db01b7a6bc88e4b568826
2024-09-24T07:49:21.953535Z INFO ThreadId(02) zenoh::net::runtime::orchestrator: Zenoh can be reached at: tcp/172.18.0.2:7442
Started Zenoh router with id 4d04140f44db01b7a6bc88e4b568826.
^Croot@6dc540c03dee:~/rosconjp24#

[INFO] [1727164178.558306422] [talker]: Publishing: 'Hello World: 13'
[INFO] [1727164179.557199672] [talker]: Publishing: 'Hello World: 14'
[INFO] [1727164180.557608797] [talker]: Publishing: 'Hello World: 15'
[INFO] [1727164181.556008256] [talker]: Publishing: 'Hello World: 16'
[INFO] [1727164181.990101256] [rclcpp]: signal_handler(signum=2)

[INFO] [1727164173.556667377] [listener]: I heard: [Hello World: 8]
[INFO] [1727164174.555905295] [listener]: I heard: [Hello World: 9]
[INFO] [1727164175.556792503] [listener]: I heard: [Hello World: 10]
[INFO] [1727164176.558720671] [listener]: I heard: [Hello World: 11]
[INFO] [1727164177.556589421] [listener]: I heard: [Hello World: 12]
[INFO] [1727164178.559730963] [listener]: I heard: [Hello World: 13]
[INFO] [1727164179.558754297] [listener]: I heard: [Hello World: 14]
[INFO] [1727164180.559306006] [listener]: I heard: [Hello World: 15]
[INFO] [1727164181.557096590] [listener]: I heard: [Hello World: 16]

root@6dc540c03dee:~/rosconjp24# source ~/ws_rmw_zenoh/install/setup.bash
root@6dc540c03dee:~/rosconjp24# RMW_IMPLEMENTATION=rmw_zenoh_cpp
ZENOH_ROUTER_CHECK_ATTEMPTS=-1 ZENOH_SESSION_CONFIG_URI=./SESSION_MULTICAST.json5 ros2 run demo_nodes_cpp talker
[INFO] [1727164175.212621920] [talker]: Publishing: 'Hello World: 1'
[INFO] [1727164176.208964212] [talker]: Publishing: 'Hello World: 2'
[INFO] [1727164177.210306754] [talker]: Publishing: 'Hello World: 3'
[INFO] [1727164178.210487421] [talker]: Publishing: 'Hello World: 4'
[INFO] [1727164179.208655005] [talker]: Publishing: 'Hello World: 5'
[INFO] [1727164180.209385172] [talker]: Publishing: 'Hello World: 6'
[INFO] [1727164181.211751798] [talker]: Publishing: 'Hello World: 7'

root@6dc540c03dee:~/rosconjp24# source ~/ws_rmw_zenoh/install/setup.bash
root@6dc540c03dee:~/rosconjp24# RMW_IMPLEMENTATION=rmw_zenoh_cpp ZENOH_ROUTER_CHECK_ATTEMPTS=-1 ZENOH_SESSION_CONFIG_URI=./SESSION_MULTICAST.json5 ros2 run demo_nodes_cpp listener
[INFO] [1727164178.216507796] [listener]: I heard: [Hello World: 4]
[INFO] [1727164179.210557838] [listener]: I heard: [Hello World: 5]
[INFO] [1727164180.211212089] [listener]: I heard: [Hello World: 6]
[INFO] [1727164181.215414506] [listener]: I heard: [Hello World: 7]
```

zenoh router (zenohd)

pub node by gossip (default)

sub node by gossip (default)

pub node by multicast

sub node by multicast

# 2の Zenoh対応

論よりRUN!!

ローカル側

customized zenoh router to comm. cloud

teleop on local with rmw\_zenoh

```
root@6dc540c03dee:~/rosconjp24# source ~/ws_rmw_zenoh/install/setup.bash
root@6dc540c03dee:~/rosconjp24# ZENOH_ROUTER_CONFIG_URI=./ROUTER_MULTIHOP_T.json5 ros2 run rmw_zenoh_cpp rmw_zenohd
2024-09-24T08:15:07.032352Z INFO ThreadId(02) zenoh::net::runtime: Using ZID: 3334296a201c9453b80ebb1ea4fd0c47
2024-09-24T08:15:07.032929Z INFO ThreadId(02) zenoh::net::runtime::orchestrator: Zenoh can be reached at: tcp/172.18.0.2:7447
Started Zenoh router with id 3334296a201c9453b80ebb1ea4fd0c47.

root@6dc540c03dee:~/rosconjp24# source ~/ws_rmw_zenoh/install/setup.bash
root@6dc540c03dee:~/rosconjp24# RMW_IMPLEMENTATION=rmw_zenoh_cpp ros2 run turtlesim turtle_teleop_key
[INFO] [1727165710.791462964] [rmw_zenoh_cpp]: Successfully connected to a Zenoh router with id 3334296a201c9453b80ebb1ea4fd0c47.
Reading from keyboard
-----
Use arrow keys to move the turtle.
Use g|b|v|c|d|l|e|l|r|t keys to rotate to absolute orientations. 'f' to cancel a rotation.
'q' to quit.
```

zenoh router on cloud

クラウド側

turtlesim on cloud with rmw\_zenoh

# 2の Zenoh対応

- RMWの性能を**基礎的かつ暫定的**に比較評価してみた
  - Raspberry Pi 5 8GB / Ubuntu Server 24.04.1 LTS
    - ✓ 1デバイスのみでの試行
  - apt install ros-rolling-desktop @2024/09/13
  - rmw\_zenoh @[rolling:aa69da9](https://github.com/ros2/rmw_zenoh)
    - ✓ デフォルトのネットワーク構成での実行
- 用いたベンチマーク
  - [ApexAI/performance\\_test](https://github.com/ApexAI/performance_test) : シンプルなPub/Sub
  - [irobot-ros/ros2-performance](https://github.com/irobot-ros/ros2-performance) : より複雑なトポロジ構成

2024年9月時点の暫定的な実装に対する評価に基づいており、傾向は今後大きく変わりうる可能性が高いです。  
また、あくまで発表者による私的な試行結果であり、あらゆる団体を代表しない非公式の情報です。

# 2の Zenoh対応

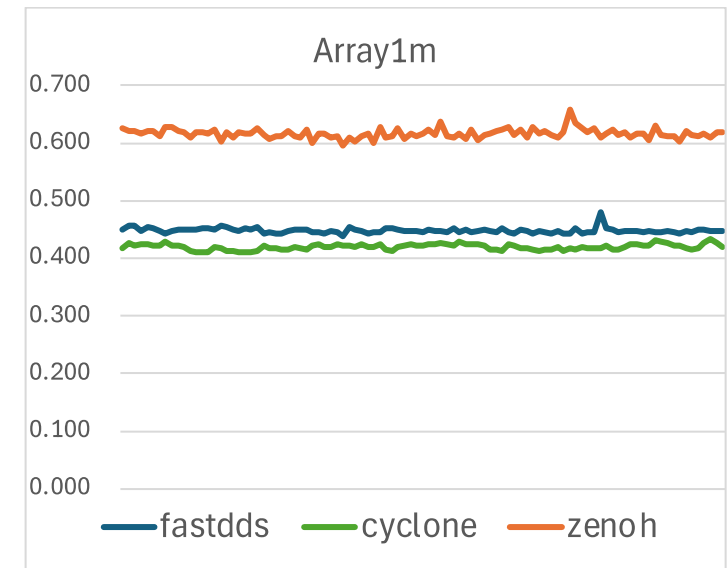
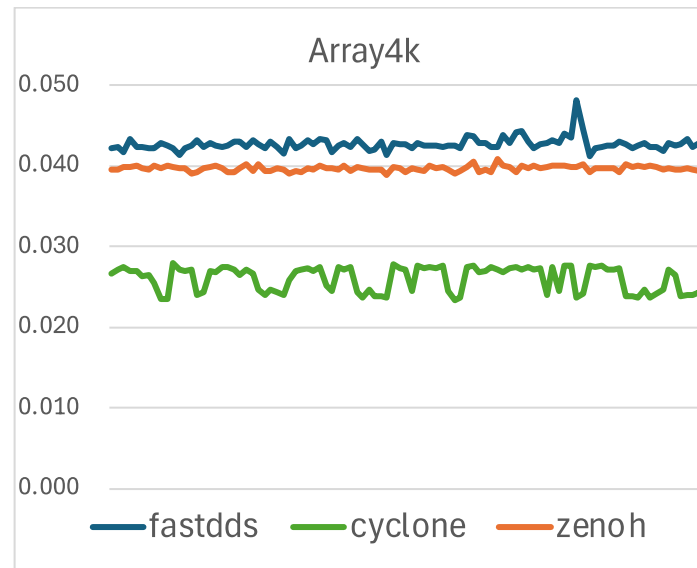
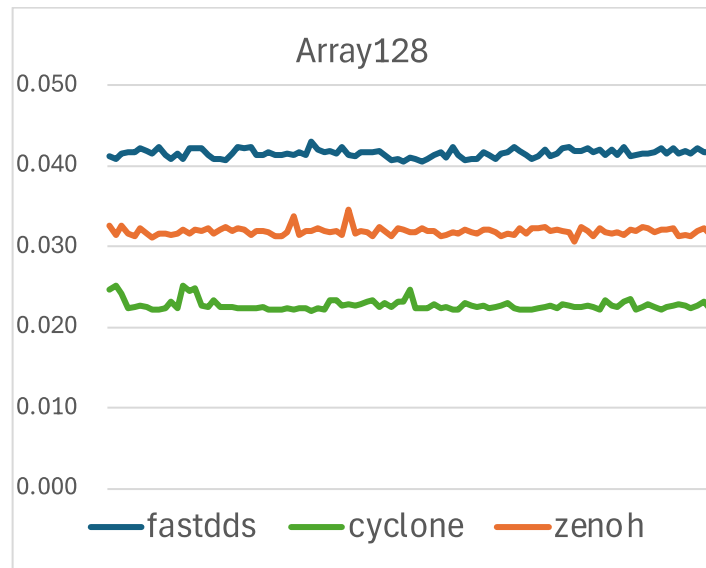
- [ApexAI/performance\\_test](#) : シンプルなPub/Sub

- “latency\_mean (ms)” の値を集計

- ✓ 出版周期 : 100 Hz

- ✓ rclcpp-single-threaded-executor (default)

- ✓ QoS : BEST\_EFFORT, VOLATILE, KEEP\_LAST 16 (default)



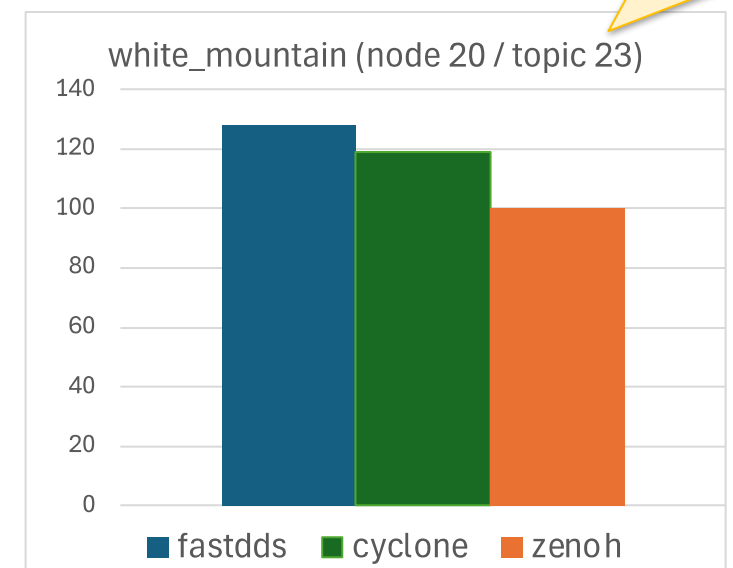
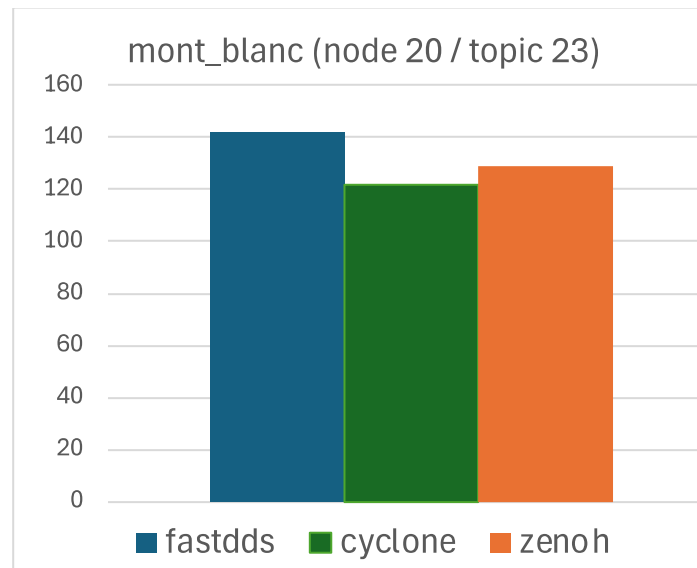
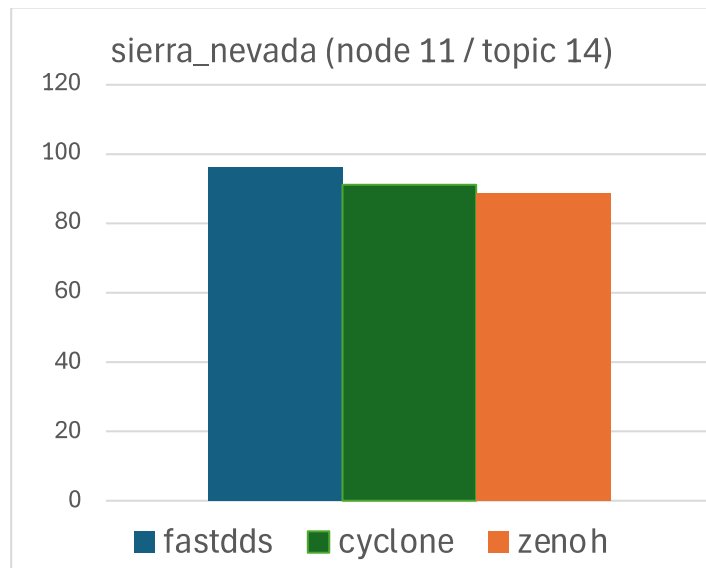
小メッセージでは **FastDDS** より優位？ただし **CycloneDDS** ほどではない？

2024年9月時点の暫定的な実装に対する評価に基づいており、傾向は今後大きく変わりうる可能性が高いです。  
また、あくまで発表者による私的な試行結果であり、あらゆる団体を代表しない非公式の情報です。

# 2の Zenoh対応

- [irobot-ros/ros2-performance](#) : より複雑なトポロジ構成
  - “latency\_total / mean\_us” の値を集計
  - ✓ 計測時間 : 100秒
  - ✓ QoS : 設定項目なし
  - ✓ SingleThreadedExecutor

トポロジは同じ  
実行形式が異なる



大規模なシステムでは Zenoh の通信性能が発揮される可能性が出てくる？

2024年9月時点の暫定的な実装に対する評価に基づいており、傾向は今後大きく変わりうる可能性が高いです。  
また、あくまで発表者による私的な試行結果であり、あらゆる団体を代表しない非公式の情報です。

# ただの個人的な予測

from rosjp#53  
ROSCon'23 参加報告



- DDSは素晴らしい！が、やはり課題もある
  - OMG標準, 自律的なP2P通信
  - スケールしにくい, NAT越えできない
- Zenoh(の併用)が役立つ場面はある
  - 高信頼な処理ではDDS? Wi-Fi通信とか性能が必要ならZenoh?
  - IoT/広域分散向けの処理ではZenoh?
- いつやってくるか??
  - 公的には Jazzy で Tier-1 化/とはいえ L-Turtle くらいでは?
  - おそれる必要はない!

not The RMW Wars,  
but RMW Cambrian Explosion!!



# まとめにかえて

- zenoh\_bridge\_ros2dds を選ぶべき状況！
  - DOMAIN をまたがるトピックを選別したい
  - Humble 以前のシステムでさくっとNATを越えたい
- rmw\_zenoh を選ぶべき状況！
  - discovery やネットワーク構成を(Wi-Fi環境下でも)簡易化したい
  - **IoT／広域分散向けの処理**で ROS 2 ecosystem を活用したい
- でもやっぱり DDS を使いたい！（これはこれで正義）
  - Tier 1 の品質保証は重視したい（とはいえ、いずれ rmw\_zenoh も,,,）
  - きめ細かく QoS を制御したい

だからこそ！！



## Items to get rmw\_zenoh to Tier-1 status #265

Open clalancette opened this issue on Aug 22 · 1 comment

clalancette commented on Aug 22 · edited by Yadunund Collaborator

The goal for Kilted is to get `rmw_zenoh_cpp` to Tier-1 status. To get there, we need to do all of the following (this list will be updated as items come up):

| Status      | Item   | Why?  | Related issues/PRs   |
|-------------|--|---|--|
| In progress | Make <code>rmw_zenoh_cpp</code> thread-safe                                | <code>rmw_zenoh_cpp</code> must be thread-safe between e.g. <code>rmw_destroy_subscriber</code> and a callback that is coming in at the same time | <a href="#">#258</a> , <a href="#">#259</a> (more to come) |
| Done ✓      | Fix transient-local missed messages  | Transient local should work in all circumstances  | <a href="#">#263</a>                                       |
| Done ✓      | Make <code>rmw_zenoh_cpp</code> compatible with Iron, Jazzy, and Rolling   | Rolling recently removed the <code>localhost_only</code> option, so we need some follow-up so it will compile there                               | <a href="#">#255</a>                                       |
| In progress | Make <code>rmw_zenoh_cpp</code> pass all core tests                        |   | <a href="#">#267</a>                                       |
| Done ✓      | Test <code>rmw_zenoh_cpp</code> on RHEL-9                                  | RHEL-9 is a Tier-2 platform, and thus should be supported   | ✓ <a href="#">#265</a> (comment)                           |
| In progress | Switch to using the upstream <code>zenohd</code>                           | The <code>zenohd</code> shipped here is a subset of functionality; thus we should switch to the upstream one which has full functionality         | <a href="#">#262</a>                                       |
| In progress | Make <code>rmw_zenoh_cpp</code> work on Windows                            | Windows is a Tier-1 platform, thus it must be supported   | <a href="#">#232</a> and <a href="#">#268</a>              |
| In progress | Switch to Zenoh 1.0  | Zenoh 1.0 changes the on-the-wire protocol, we should keep up with these changes  |  |
| Not started | Implement Security (i.e. SROS2) for <code>rmw_zenoh_cpp</code>             | This is required for Tier-1 RMW implementations   |  |
| Not started | Fix shutdown unsoundness   | Zenoh cannot be destroyed during atexit handlers, so we have to do that ahead of time   | <a href="#">#170</a>                                       |
| Not started | Reduce the session creation time   | It currently takes 1.5 seconds to create a session, but ROS 2 assumes it is more-or-less instantaneous  |  |
| Not started | Figure out exactly how and when the <code>zenohd</code> router will be run |   |  |
| Not started | Add in a <code>QUALITY_DECLARATION.md</code> file                          |   |  |

Assignees  
No one assigned

Labels  
None yet

Projects  
None yet

Milestone  
No milestone

Development  
No branches or pull requests

2 participants

[https://github.com/ros2/rmw\\_zenoh/issues/265](https://github.com/ros2/rmw_zenoh/issues/265)



We need you, you & you!!

Better late than never!!





# zetta score

The world belongs to those who





*Thank you!  
merci!!*



*Arigato!!!*



**Zenoh**



*Lab#8, IPC, IST, UTokyo  
Computing System Laboratory*



**@takasehideki**



A part of this work is going as collaborative research with SoftBank Corp., and is supported by JST, CREST Grant Number JPMJCR21D2, Japan.