

EPROSIMA

The
Middleware
Experts



Efficient data recording and replaying in ROS 2

Speaker:

- Raúl Sánchez-Mateos, Project Manager @ eProsimia
raul@eprosima.com

October 20th, 2023

Agenda



01

Motivation



- The importance of record and replay data in robotics applications development
- Fast DDS Dynamic Types

02

ROS 2 Recorder



- Overview & Usage
- Main configuration options
- Settings for resource optimization

03

ROS 2 Recorder Controller



- Overview & Usage
- How to configure the remote controller
- Creating your own controller

04

ROS 2 Replayer



- Overview & Usage
- Main configuration options
- Settings for resource optimization

05

Visualize Data



- ROS 2 Spy
- Foxglove Studio

Motivation

Why ROS 2 Record & Replay?



Analyse any potential issues

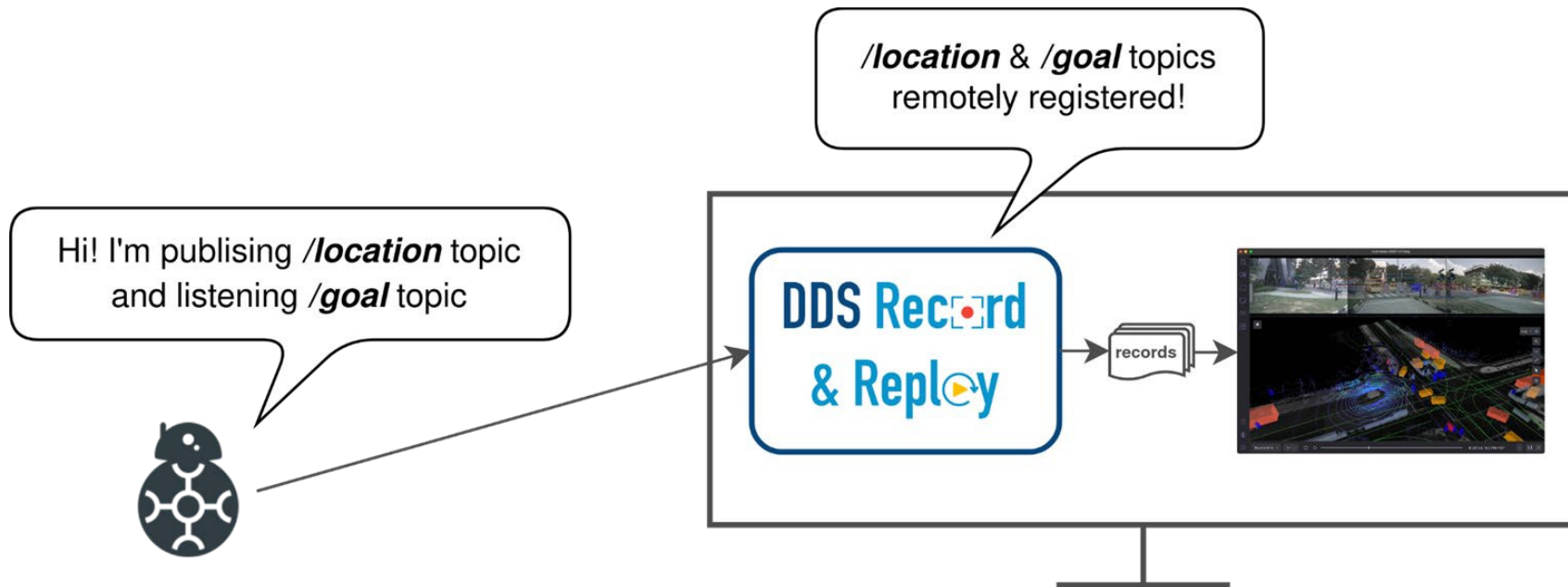
Collect data from the execution of the robots in order to detect possible errors in further data analysis.

SW & HW improvements based on data

Following data analysis and detection of possible faults, the developed systems are updated and improved.

Fast DDS Dynamic Types

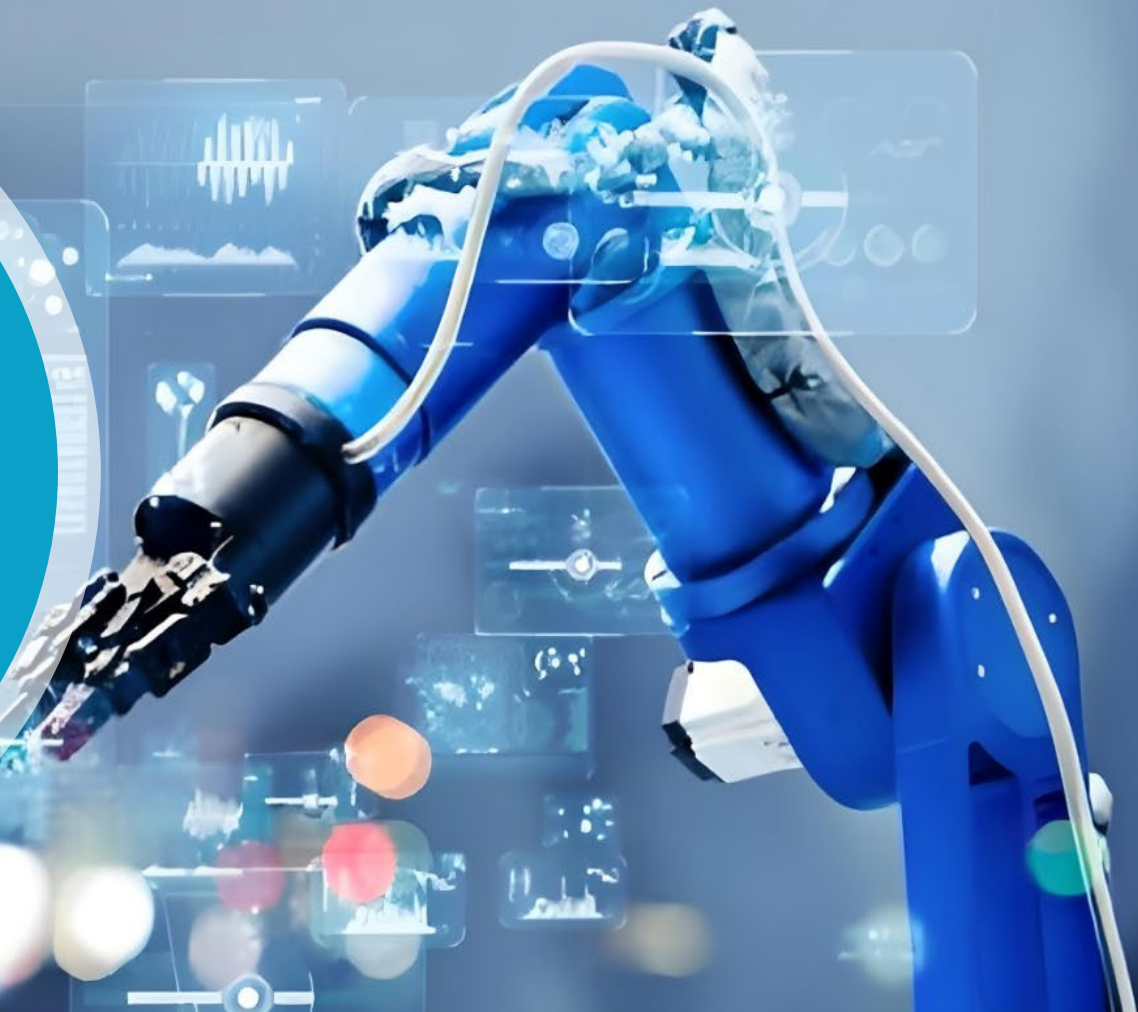
Importance of Fast DDS Dynamic Types for ROS 2 debugging tools





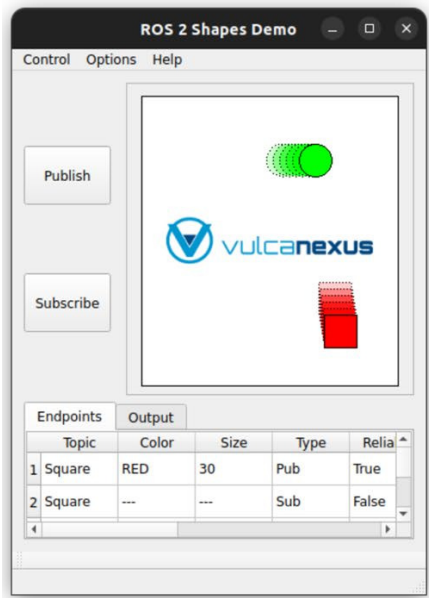
ROS 2 Recorder

CLI application to save ROS 2
published messages in a MCAP
database.



ROS 2 Recorder

ROS 2 Recorder overview



DDS Record

```
/* output.mcap */  
  
SCHEMA  
-----  
Name = /shapes  
Encoding = CDR  
  
Data Type =  
  struct ShapeType {  
    @key string color;  
    long x, y;  
    long shapesize;  
  };  
  
RECORDS  
-----  
timestamp 0 -> 02 ff 25 ...  
timestamp 1 -> 56 12 de ...
```

ROS 2 Recorder

How to use ROS 2 Recorder



- Installation
 - Windows & Linux distribution (compiled with Colcon)
 - Docker image
 - Vulcanexus Humble & Vulcanexus Iron
- Configuration via *yaml* file
- Command-Line Interface (CLI) application

```
$ ddsrecorder --config-file recorder.yml ...
```

ROS 2 Recorder

Main configuration options



1

DDS

- ROS 2 Domain ID

```
# recorder.yaml
```

```
dds:
```

```
  domain: 0
```



ROS 2 Recorder

Main configuration options



1 DDS

- ROS 2 Domain ID
- Topic allowlist & blocklist

```
# recorder.yaml

dds:
  allowlist:
    - name: "*"
      type: "topic_type"
  blocklist:
    - name: "/secret"
```



ROS 2 Recorder

Main configuration options



1 DDS

- ROS 2 Domain ID
- Topic allowlist & blacklist
- Topic QoS settings

```
# recorder.yaml

dds:
  builtin-topics:
    - name: "/chatter"
      qos:
        reliability: true
        durability: true
        keyed: false
        partitions: true
        ownership: false
        downsampling: 4
        max-reception-rate: 10
```



ROS 2 Recorder

Main configuration options



1 DDS

- ROS 2 Domain ID
- Topic allowlist & blacklist
- Topic QoS settings
- Transport configuration

```
# recorder.yaml

dds:
  ignore-participant-flags:
    filter_different_host
  transport: shm
  whitelist-interfaces:
    - "127.0.0.1"
```



ROS 2 Recorder

Main configuration options



1 DDS

- ROS 2 Domain ID
- Topic allowlist & blacklist
- Topic QoS settings
- Transport configuration

2 Recorder

```
# recorder.yaml
```

```
recorder:
```



ROS 2 Recorder

Main configuration options



1 DDS

- ROS 2 Domain ID
- Topic allowlist & blocklist
- Topic QoS settings
- Transport configuration

2 Recorder

- Output file
- Downsampling /
Max reception rate
- Compression
- Save only if type discovered
- Record discovered types

```
# recorder.yaml

recorder:

  output:
    filename: "output"
    path: "."
    timestamp-format: "%Y_%H%M%S_%Z"
    local-timestamp: false

  downsampling: 3
  max-reception-rate: 20
  only-with-type: false
  compression:
    algorithm: lz4
    level: slowest
  record-types: true
```



ROS 2 Recorder

Settings for resource optimization



- Internal buffer size
- Number of threads
- Max. pending samples
- Clean-up period

```
# recorder.yaml

recorder:

  buffer-size: 50

specs:

  threads: 8
  max-pending-samples: 10
  cleanup-period: 90
```





ROS 2 Recorder Remote Controller

DDS Recorder internal module
to remote control the DDS
Recorder execution.



ROS 2 Recorder Remote Controller

Overview of the ROS 2 Recorder remote control system



DDS Recorder controller

File Help

eProsima DDS Recorder status: **RUNNING**

	Start	Timestamp	Source	Message
Pause	1	2023-03-08 14:21:17,448	Controller	DDS Recorder found!
Stop	2	2023-03-08 14:21:17,528	DDS ...	Update from CLOSED to STOPPED
Event	3	2023-03-08 14:22:20,639	Controller	start [] command sent
Event_Start	4	2023-03-08 14:22:20,643	DDS ...	Update from STOPPED to RUNNING
Event_Stop				
Close				

```
struct DdsRecorderStatus  
{  
    string previous;  
    string current;  
    string info;  
};
```

DdsRecorderStatus

DdsRecorderCommand

```
struct DdsRecorderCommand  
{  
    string command;  
    string args;  
};
```

DDS Recorder

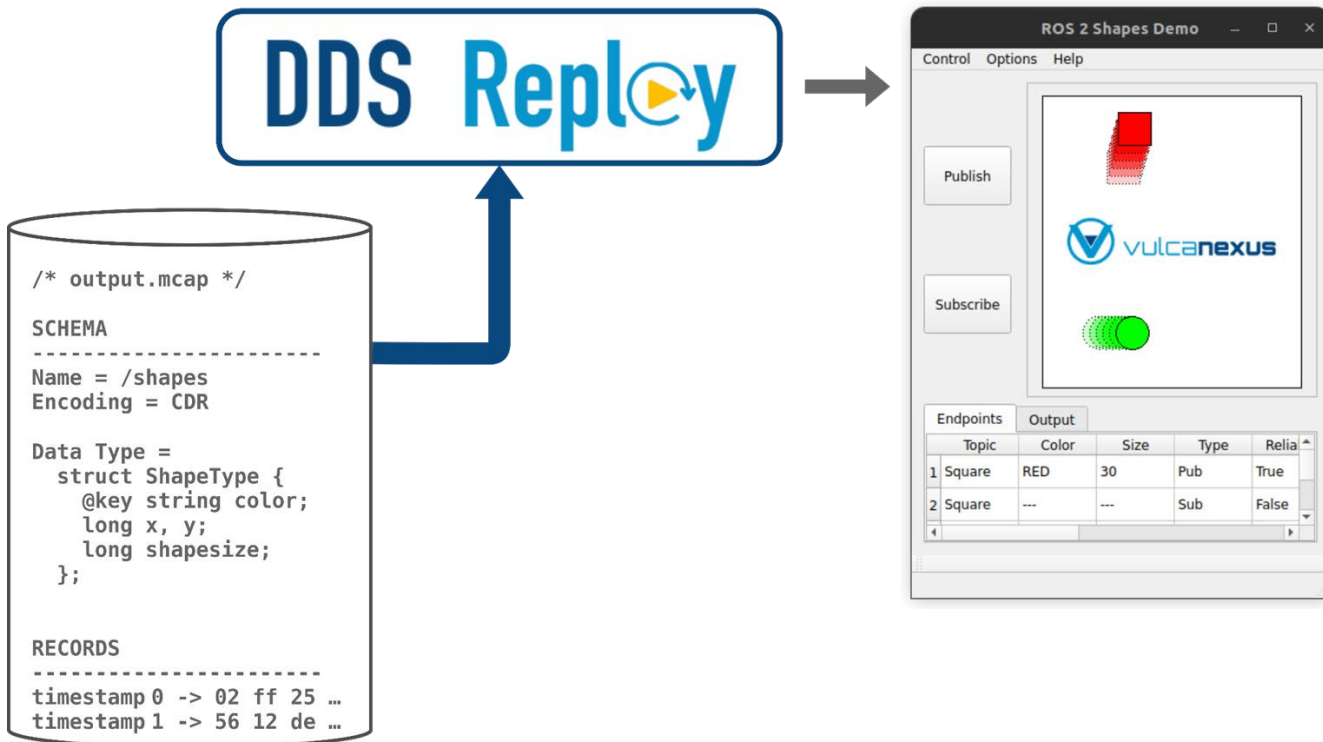


ROS 2 Replayer

—
CLI application to replay
recorder ROS 2 messages saved
in a MCAP database.

ROS 2 Replayer

ROS 2 Replayer overview



ROS 2 Replayer

How to use ROS 2 Replayer



- Installation
 - Windows & Linux distribution (compiled with Colcon)
 - Docker image
 - Vulcanexus Tools (Humble & Iron)
- Configuration via *yaml* file
- Command-Line Interface (CLI) application

```
$ ddsreplayer --input-file output.mcap --config-file replayer.yaml ...
```

ROS 2 Replayer

Main configuration options



- 1 DDS
 - ROS 2 Domain ID
 - Topic allowlist & blocklist
 - Topic QoS settings
 - Transport configuration
- 2 Replayer
 - Input file
 - Begin / end time
 - Start replay time
 - Replay sample rate
 - Replay types
- 3 Specs
 - Number of threads
 - Wait for all messages “ACKed”

```
# recorder.yml

dds: ...

replayer:

  input-file: output.mcap
  begin-time: ...
  end-time: ...
  start-replay-time: ...
  rate: 2
  replay-types: true

specs:

  threads: 8
  wait-all-acked-timeout: 10
```





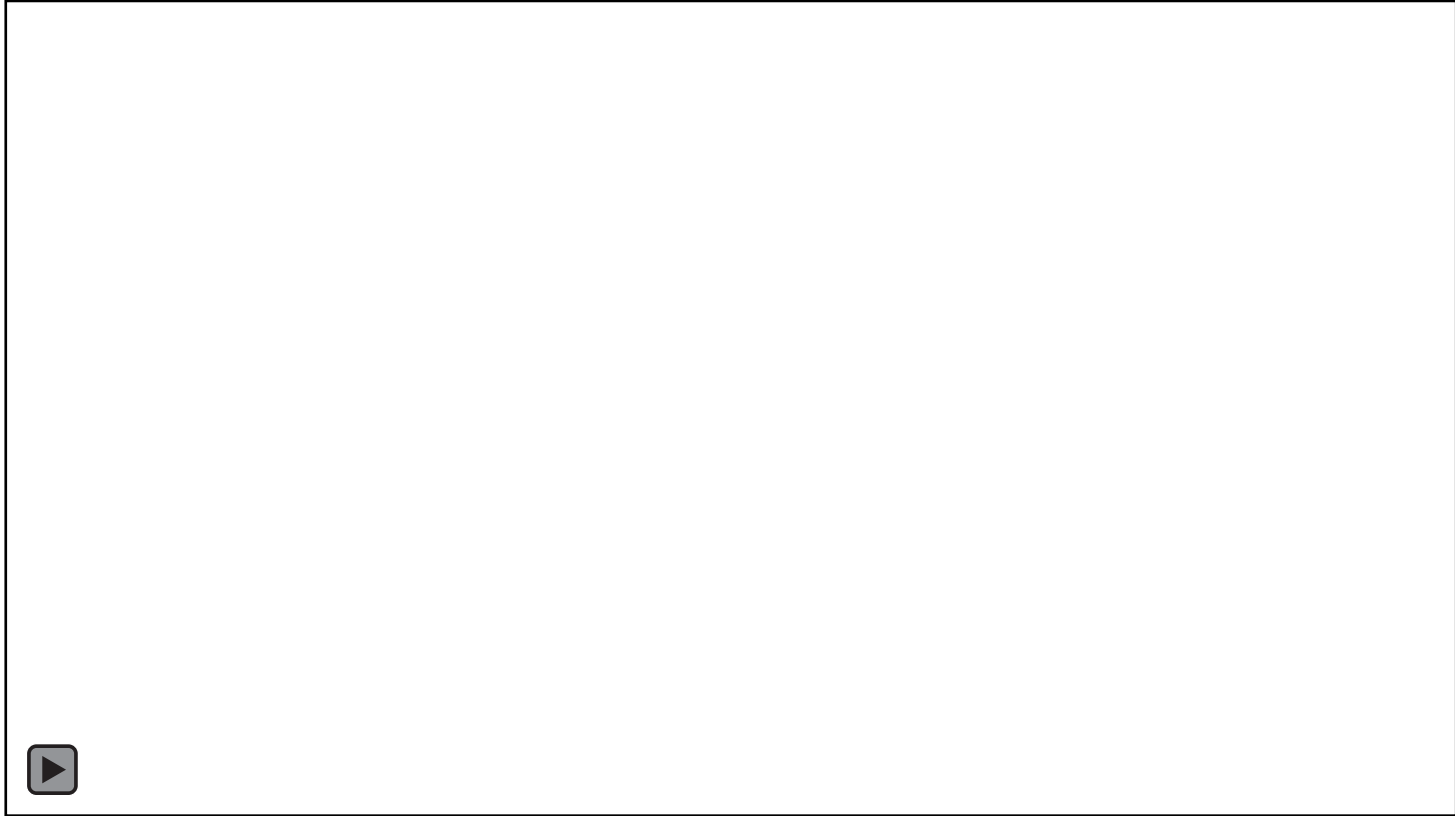
Data Visualization

Visualize recorded data with
ROS 2 Replayer and inspect
ROS 2 Replayer execution with
ROS 2 Spy.



Foxglove Studio

Visualize recorded data with Foxglove Studio



ROS 2 Record & Replay

ROS 2 Record & Replay documentation and GitHub repository



Documentation



GitHub Repository





EPROSIMA

The Middleware Experts

www.eProsima.com



[Linkedin.com/company/eProsima](https://www.linkedin.com/company/eProsima)



[Twitter.com/EProsima](https://twitter.com/EProsima)