

目录

- [目录](#)
- [版本](#)
- [1 引言](#)
 - [1.1 用户手册的目的和范围](#)
 - [1.2 使用本手册的前提条件](#)
 - [环境准备](#)
 - [ros_xway镜像](#)
 - [1.3 镜像通讯方式](#)
- [2 ROS2消息包构造](#)
 - [2.1 准备ros2消息包](#)
 - [2.2 构造ros2消息包](#)
 - [2.4 消息内容验证](#)
- [3 ros2命令行工具的使用](#)
 - [3.1 预准备](#)
 - [3.2 Topic监控](#)
 - [3.1.2 执行方式](#)
 - [3.1.2.1 ros2 topic list](#)
 - [3.1.2.2 ros2 topic echo](#)
 - [3.1.2.2 ros2 topic monitor](#)
 - [3.3 录制与回放](#)
 - [3.3.1 录制](#)
 - [3.3.1.1 ros2 bag record](#)
 - [3.3.1.2 ros2 bag2 record](#)
 - [3.2.2 回放](#)
 - [3.2.2.1 预准备](#)
 - [3.2.2.2 ros2 bag play](#)
 - [3.2.2.3 ros2 bag2 play](#)
 - [3.2.2.3 修改回放时使用的QoS](#)
- [4 问题及诊断方法](#)
 - [4.1 ros2 topic 无法监控](#)
 - [4.2 ros2 topic echo 显示失败](#)
 - [4.3 ros2 bag record 录制失败](#)
 - [4.3 ros2 bag play 回放失败](#)

版本

版本号	变更内容	责任人
ros_xway v1	初始版本	杨晨光
ros_xway v2	修改ros2镜像及其启动方式	杨晨光

版本号	变更内容	责任人
ros_xway v3.3	1.新增ros2 topic monitor命令使用方法的描述	潘常远
	2.root用户默认设置DOMAIN_ID为1，并使用profiles.xml	
	3.优化ros2 topic hz执行效率	
	4.优化ros2 topic echo执行效率	
	5.修改ros2 empy模板文件，支持c实现struct转yaml	
ros_xway v3.7	1.新增ros2 bag2命令	

1 引言

1.1 用户手册的目的和范围

本手册的目的是为了帮助用户快速搭建ros2环境，包括docker镜像的安装，ros2 package的编译，来实现对XwayOS在使用DDS绑定时运行过程中消息的查看，录制与回放。基于该环境用户还可以扩展进行可视化的开发，比如基于webviz或者Foxglove，这部分本文将不做介绍。

1.2 使用本手册的前提条件

环境准备

安装docker应用程序，然后从ZHITO harbor服务器拉取最新的ros_xway docker镜像。镜像拉取命令如下（'v'指示镜像版本号）：

- arm64版本

```
docker pull harbor.zhito.com/xwayos/ros_xway:v*_arm64
```

- x86版本

```
docker pull harbor.zhito.com/xwayos/ros_xway:v*_x86
```

若拉取失败，请查看本机能否登录harbor服务器。

```
docker login https://harbor.zhito.com
```

若登录失败，需查看DNS地址是否配置正确，可建议联系IT解决。

ros_xway镜像

- 操作系统
 - ubuntu 20.04

- 支持的硬件架构
 - ARMv8-A-aarch64
 - X86_64
- ros_xway docker镜像内账号信息 用户名：xway 登陆密码：xway

1.3 镜像通讯方式

镜像中与XwayOS ACore ara::com默认通过本机共享内存方式通信。若需要实现跨主机通讯，需修改镜像中/opt/xway/profiles.xml文件。

文件内容如下：

```
<?xml version="1.0" encoding="UTF-8" ?>
<dds xmlns="http://www.eprosima.com/XMLSchemas/fastRTPS_Profiles">
  <profiles>
    <transport_descriptors>
      <transport_descriptor>
        <transport_id>CustomUdpTransport</transport_id>
        <type>UDPV4</type>
        <interfaceWhiteList>
          <address>172.16.12.211</address>
        </interfaceWhiteList>
      </transport_descriptor>
      <transport_descriptor>
        <transport_id>SHM_BASE</transport_id>
        <type>SHM</type>
        <segment_size>102400</segment_size>
      </transport_descriptor>
    </transport_descriptors>

    <participant profile_name="CustomTcpTransportParticipant"
is_default_profile="true">
      <rtps>
        <userTransports>
          <!-- <transport_id>CustomUdpTransport</transport_id> --
          <transport_id>SHM_BASE</transport_id>
        </userTransports>
        <useBuiltinTransports>false</useBuiltinTransports>
      </rtps>
    </participant>
  </profiles>
</dds>
```

将<interfaceWhiteList>中的<address>字段修改为本机的ip地址，取消<userTransports>中的字段<transport_id>CustomUdpTransport</transport_id>注释。

2 ROS2消息包构造

2.1 准备ros2消息包

通过XwayOS Studio完成前端建模配置后，在生成用户代码的同时，会生成**ros2workspace**，其中包含：

- 所有配置的XwayOS ACore ara::com对应的事件类通信ros2消息包（message package）
- ROS_USER_DATA.sh用于设置ros2回放通道的DDS QOS USER_DATA内容，以便实现与XwayOS ACore ara::com的互通
- topic_list.csv：列举所有的事件消息信息，包括类型、主题名、ServiceID、InstanceID等

2.2 构造ros2消息包

步骤如下：

1. 运行ros_xway镜像，将XwayOS用户项目生成的ros2workspace挂载到容器目录/home/ros2_workspace。

```
docker run -it \  
  -u xway \  
  -v path/ros2workspace:/home/ros2_workspace \  
  -v /etc/localtime:/etc/localtime:ro \  
  -v /dev/shm:/dev/shm \  
  -e DISPLAY=unix$DISPLAY \  
  -e GDK_SCALE \  
  -e GDK_DPI_SCALE \  
  --ipc=host \  
  --userns=host \  
  --name ros2docker \  
  --net=host \  
  harbor.zhito.com/xwayos/ros_xway:v3_arm64 # 以orin平台为例
```

2. 执行编译 编译特定的子package：

```
colcon build --packages-select {package} ...
```

全量编译

```
cd /home/ros2_workspace  
colcon build
```

2.4 消息内容验证

消息包构造完成后，/home/ros2_workspace目录下会生成 **install/**与 **build/**文件夹。

- 执行消息包设置环境变量的脚本

```
source /home/ros2_workspace/install/local_setup.bash
```

- 查看消息接口是否可用

```
ros2 interface list
```

若添加成功，消息服接口在列表中可见。

3 ros2命令行工具的使用

将介绍几个常用ros2命令行的使用方法

3.1 预准备

- 在指定容器中执行ros_xway镜像（落阿取），并将创建的工程目录映射到docker镜像中。

```
docker run -it \  
-u xway \  
-v path/ros2workspace:/home/ros2_workspace \  
-v /etc/localtime:/etc/localtime:ro \  
-v /dev/shm:/dev/shm \  
-e DISPLAY=unix$DISPLAY \  
-e GDK_SCALE \  
-e GDK_DPI_SCALE \  
--ipc=host \  
--usersns=host \  
--name ros2docker \  
--net=host \  
harbor.zhito.com/xwayos/ros_xway:v3_arm64 # arm平台为例
```

- 将生成的msg信息加入到当前环境中，不执行该操作会导致无法获取消息的确切信息。

```
source /home/ros2_workspace/install/local_setup.bash
```

- 设置通讯的ROS_DOMAIN_ID，配置的ID值要与CM通讯中设置的domainId一致。(该值在docker环境中被默认配置为1)

```
export ROS_DOMAIN_ID=xx
```

3.2 Topic监控

Topic监控可以查看当前运行的topic名称，topic所携带的msg名称以及每包中所包含的信息。

3.1.2 执行方式

以上准备工作完成后，在 **ros_xway** 启动的docker容器中便可以监控XwayOS DDS总线上相关主题的信息。以下是常用的监控命令：

3.1.2.1 ros2 topic list

ros2 topic list可查看可订阅的主题的列表。使用方法如下：

```
usage: ros2 topic list [-h] [--spin-time SPIN_TIME] [--no-daemon] [-t] [-c]
[--include-hidden-topics]
```

Output a list of available topics

optional arguments:

-h, --help show this help message and exit

--spin-time SPIN_TIME

Spin time in seconds to wait for discovery (only

applies when not using an

already running daemon)

--no-daemon Do not spawn nor use an already running daemon

-t, --show-types Additionally show the topic type

-c, --count-topics Only display the number of topics discovered

--include-hidden-topics

Consider hidden topics as well

3.1.2.2 ros2 topic echo

ros2 topic echo命令可查看某个主题消息的内容。使用方法如下：

```
usage: ros2 topic echo [-h]
```

```
 [--qos-profile
```

```
{unknown,system_default,sensor_data,services_default,parameters,parameter_e
vents,action_status_default}]
```

```
 [--qos-depth N] [--qos-history {system_default,keep_last,keep_all,unknown}]
```

```
 [--qos-reliability {system_default,reliable,best_effort,unknown}]
```

```
 [--qos-durability {system_default,transient_local,volatile,unknown}] [--
csv]
```

```
 [--full-length] [--truncate-length TRUNCATE_LENGTH] [--no-arr] [--no-str]
```

```
topic_name [message_type]
```

3.1.2.2 ros2 topic monitor

执行 `ros2 topic monitor` 命令可查看总线上可订阅的所有的主题的列表，或通过用户指定显示特定的主题列表。

```
usage: ros2 topic monitor [-h] (-t TOPICS | -a) [--no_daemon]
                        [--qos-profile
{unknown,system_default,sensor_data,services_default,parameters,parameter_e
vents,action_status_default}]
                        [--qos-depth N] [--qos-history
{system_default,keep_last,keep_all,unknown}]
                        [--qos-reliability
{system_default,reliable,best_effort,unknown}]
                        [--qos-durability
{system_default,transient_local,volatile,unknown}]
```

- 显示指定的主题列表

```
ros2 topic monitor -t topic_1 topic_2 ...
```

- 显示所有可订阅的主题

```
ros2 topic monitor -a
```

- 根据白名单显示可订阅的主题

```
export ROS2_TOPIC_WHITELIST = "topic_1 topic_2 ..."
ros2 topic monitor -a
```

键盘操作介绍：

KEY_PPAGE (Page Up) : 向上翻页

KEY_NPAGE (Page Down) : 向下翻页

KEY_DOWN 或 's', 'S': 高亮行向下移动

KEY_UP 或 'w', 'W': 高亮行向上移动

KEY_RIGHT 或 'd', 'S': 展开当前行的下一级 (如果可以)

KEY_LEFT 或 'a', 'A': 回到上一级显示

3.3 录制与回放

ros2提供命令行工具，可以实现订阅可用的ros2消息并落盘为ros2 bag文件。同时也可以解析ros2 bag文件，通过ros2消息中间件发布出来。目前ros_xway镜像中包含ros2 bag和ros2 bag2两种工具，分别对应ros2 foxy

和humble两个版本的功能。

3.3.1 录制

3.3.1.1 ros2 bag record

执行 `ros2 bag record` 来实现对于topic进行录制。命令包含的常用参数以及对应的含义为：

```
使用方法 : ros2 bag record [-h] [-a] [-o OUTPUT] [-s STORAGE] [-f
SERIALIZATION_FORMAT] [--no-discovery] [-p POLLING_INTERVAL]
[-b MAX_BAG_SIZE] [--max-cache-size MAX_CACHE_SIZE] [--compression-mode
{none,file}]
[--compression-format {zstd}] [--include-hidden-topics]
[--qos-profile-overrides-path QOS_PROFILE_OVERRIDES_PATH]
[topics [topics ...]]
```

将ROS数据记录到一个bag文件中

位置参数：

topics 要记录的主题

可选参数：

-h, --help 显示帮助信息并退出

-a, --all 记录所有主题，如果没有明确列出主题，则此选项是必需的。

-o OUTPUT, --output OUTPUT
要创建的bag文件的目标位置，默认为当前目录中带有时间戳的文件夹

-s STORAGE, --storage STORAGE
要使用的存储标识符，默认为'sqlite3'

-f SERIALIZATION_FORMAT, --serialization-format SERIALIZATION_FORMAT
保存消息的rmw序列化格式，默认为当前使用的rmw

--no-discovery 在录制过程中禁用主题自动发现：仅记录启动时存在的主题

-p POLLING_INTERVAL, --polling-interval POLLING_INTERVAL
查询可用主题以进行记录之间等待的时间（以毫秒为单位）。如果启用了--no-discovery，则不起作用。

-b MAX_BAG_SIZE, --max-bag-size MAX_BAG_SIZE
在bag文件被拆分之前的最大大小（字节）。默认值为零，将写入单个bag文件并禁用拆分。

--max-cache-size MAX_CACHE_SIZE
在写入磁盘之前在缓存中保存的最大消息数量。默认值为零，直接将每条消息写入磁盘。

--compression-mode {none,file}
确定是否压缩bag文件。默认值为"none"。

--compression-format {zstd}
指定压缩格式/算法。默认值为"none"。

--include-hidden-topics
也记录隐藏主题。

--qos-profile-overrides-path QOS_PROFILE_OVERRIDES_PATH
定义特定主题的QoS配置文件覆盖的yaml文件路径。

以下是一些常用选项的解释：

- -a, --all：记录所有话题。如果未显式列出话题，则此选项为必需。

- `-o OUTPUT, --output OUTPUT` : 指定要创建的bag文件的目的地，默认为当前目录中带有时间戳的文件夹。
- `-s STORAGE, --storage STORAGE` : 要使用的存储标识符，默认为'sqlite3'。
- `--compression-mode {none,file}` : 确定是否压缩bag文件。默认值为"none"。
- `--compression-format {zstd}` : 指定压缩格式/算法。默认值为"none"。

在录制结束后，可以执行 `ros2 bag info <bag name>` 来查看录制包的信息，例如：

```
ros2 bag info record-2023-12-21-1.bag/record-2023-12-21-1.bag_0.db3
[INFO] [1662659550.957778680] [rosvbag2_storage]: Opened database 'record-2023-12-21-1.bag/record-2023-12-21-1.bag_0.db3' for READ_ONLY.

Files:                record-2023-12-21-1.bag/record-2023-12-21-1.bag_0.db3
Bag size:              16.0 KiB
Storage id:            sqlite3
Duration:              95.26s
Start:                 Sep  9 2022 01:19:33.168 (1662657573.168)
End:                   Sep  9 2022 01:21:08.195 (1662657668.195)
Messages:              52
Topic information: Topic: /XwayOS/HelloWorldTopic | Type:
example/msg/HelloWorld | Count: 52 | Serialization Format: cdr
```

3.3.1.2 ros2 bag2 record

```
usage: ros2 bag2 record [-h] [-a] [-e REGEX] [-x EXCLUDE] [--include-unpublished-topics] [--include-hidden-topics] [-o OUTPUT]
[-s {sqlite3}] [-f {}] [--no-discovery] [-p POLLING_INTERVAL] [-b MAX_BAG_SIZE] [-d MAX_BAG_DURATION]
[--max-cache-size MAX_CACHE_SIZE] [--compression-mode {none,file,message}]
[--compression-format {}]
[--compression-queue-size COMPRESSION_QUEUE_SIZE] [--compression-threads COMPRESSION_THREADS]
[--snapshot-mode] [--ignore-leaf-topics] [--qos-profile-overrides-path QOS_PROFILE_OVERRIDES_PATH]
[--storage-preset-profile STORAGE_PRESET_PROFILE] [--storage-config-file STORAGE_CONFIG_FILE]
[--start-paused] [--use-sim-time]
[topics [topics ...]] Record ROS data to a bagpositional arguments:
topics                要记录的话题列表:
-h, --help: 显示帮助信息并退出。
-a, --all: 记录所有话题。如果没有显式提供话题列表或正则表达式过滤器，则此选项是必需的。
-e REGEX, --regex REGEX: 只记录包含指定正则表达式的话题。覆盖--all，且作用于话题列表之上。
-x EXCLUDE, --exclude EXCLUDE: 排除包含指定正则表达式的话题。在--all，--regex或话题列表的基础上生效。
--include-unpublished-topics: 发现并记录没有发布者话题。对这些话题的订阅将使用默认的QoS，除非在QoS覆盖文件中另有规定。
--include-hidden-topics: 也发现并记录隐藏话题。这些是ROS 2实现内部使用的话题。
-o OUTPUT, --output OUTPUT: 要创建的包文件的目的地，默认为当前目录中带时间戳的文件
```

夹。

```
-s {sqlite3}, --storage {sqlite3}: 要使用的存储标识符, 默认为'sqlite3'。
-f {}, --serialization-format {}: 保存消息的rmw序列化格式, 默认为当前使用的rmw。
--no-discovery: 在记录过程中禁用话题自动发现: 只记录启动时存在的话题。
-p POLLING_INTERVAL, --polling-interval POLLING_INTERVAL: 查询可用话题进行记录的等待时间(毫秒)。如果启用了--no-discovery, 则此选项无效。
-b MAX_BAG_SIZE, --max-bag-size MAX_BAG_SIZE: 包文件在分割之前的最大大小(字节)。默认为零, 以单个包文件写入, 不启用分割。
-d MAX_BAG_DURATION, --max-bag-duration MAX_BAG_DURATION: 包文件在分割之前的最大持续时间(秒)。默认为零, 以单个包文件写入, 不启用分割。如果同时启用了按大小和持续时间分割, 包将在先达到的阈值处分割。
--max-cache-size MAX_CACHE_SIZE: 每个缓冲区中保留的消息的最大尺寸(字节)。默认为100兆字节。缓存通过双缓冲处理, 这意味着在悲观情况下可能需要多达两倍参数值的内存。一个经验法则是缓存总记录数据量约为一秒的数量级。如果指定的值为0, 则每条消息直接写入磁盘。
--compression-mode {none,file,message}: 确定是按文件还是按消息进行压缩。默认为'none'。
--compression-format {}: 指定压缩格式/算法。默认为无。
--compression-queue-size COMPRESSION_QUEUE_SIZE: 在被丢弃之前可以排队压缩的文件或消息数量。默认为1。
--compression-threads COMPRESSION_THREADS: 允许并行压缩的文件或消息数量。默认为0, 将被解释为CPU核心数。
--snapshot-mode: 启用快照模式。直到调用"/rosvbag2_recorder/snapshot"服务之前, 消息不会被写入包文件。
--ignore-leaf-topics: 忽略没有发布者的话题。
--qos-profile-overrides-path QOS_PROFILE_OVERRIDES_PATH: 定义特定话题的QoS配置覆盖的yaml文件路径。
--storage-preset-profile STORAGE_PRESET_PROFILE: 为存储选择一个配置预设。这个标志的设置仍然可以通过传递--storage-config-file的相应设置进行覆盖。
--storage-config-file STORAGE_CONFIG_FILE: 定义存储特定配置的yaml文件路径。对于默认的存储插件设置通过语法: write: pragmas: [] 进行规定。
```

跟ros2 bag的区别在于新增了以下功能:

- -e REGEX, --regex REGEX: 只记录包含指定正则表达式的话题。覆盖--all, 且作用于话题列表之上。
- -x EXCLUDE, --exclude EXCLUDE: 排除包含指定正则表达式的话题。在--all, --regex或话题列表的基础上生效。
- --include-unpublished-topics: 发现并记录没有发布者的话题。对这些话题的订阅将使用默认的QoS, 除非在QoS覆盖文件中另有规定。
- -d MAX_BAG_DURATION, --max-bag-duration MAX_BAG_DURATION: 包文件在分割之前的最大持续时间(秒)。默认为零, 以单个包文件写入, 不启用分割。如果同时启用了按大小和持续时间分割, 包将在先达到的阈值处分割。
- --compression-queue-size COMPRESSION_QUEUE_SIZE: 在被丢弃之前可以排队压缩的文件或消息数量。默认为1。
- --compression-threads COMPRESSION_THREADS: 允许并行压缩的文件或消息数量。默认为0, 将被解释为CPU核心数。
- --snapshot-mode: 启用快照模式。直到调用"/rosvbag2_recorder/snapshot"服务之前, 消息不会被写入包文件。
- --ignore-leaf-topics: 忽略没有发布者的话题。
- --qos-profile-overrides-path QOS_PROFILE_OVERRIDES_PATH: 定义特定话题的QoS配置覆盖的yaml文件路径。

- `--storage-preset-profile STORAGE_PRESET_PROFILE`: 为存储选择一个配置预设。这个标志的设置仍然可以通过传递`--storage-config-file`的相应设置进行覆盖。
- `--storage-config-file STORAGE_CONFIG_FILE`: 定义存储特定配置的yaml文件路径。对于默认的存储插件设置通过语法：`write: pragmas: []`进行规定。

3.2.2 回放

3.2.2.1 预准备

上文介绍过使用XwayOS工具链生成代码时会同时生成 `ROS_USER_DATA.sh` 脚本文件用于设置DDS QOS USER_DATA,在启动录制之前需要执行该环境变量。

```
source ROS_USER_DATA.sh
```

注：若用户希望省略该操作，可固定`ros2_workspace`挂载的路径，将其添加到`.bashrc`，在进入docker容器时自动执行

另外，用户也可根据需要手动实现ROS_USER_DATA，DDS QOS USER_DATA的规则为：

`"ara.com://services/ServiceID_InstanceID-VersionID&ServiceID_InstanceID-VersionID"`。ServiceId与 InstanceId可以在生成目录下 `integration_path/build/install/etc/process/process_name/etc/dds.json` 中或者 `ros2_worksapce` 下的 `topic_list.csv` 文件中查看。

设置示例如下：

```
export ROS_USER_DATA="ara.com://services/2438_71-1.0&2400_70-1.0"
```

3.2.2.2 ros2 bag play

```
usage: ros2 bag play [-h] [-s STORAGE] [--read-ahead-queue-size
READ_AHEAD_QUEUE_SIZE] [-r RATE] [--topics TOPICS [TOPICS ...]]
[--qos-profile-overrides-path QOS_PROFILE_OVERRIDES_PATH] [-l] [--remap
REMAP [REMAP ...]]
bag_filePlay back ROS data from a bagpositional arguments:
bag_file          要重新播放的包文件
```

可选参数：

`-h, --help` 显示此帮助消息并退出

`-s STORAGE, --storage STORAGE`

要使用的存储标识符，默认为 `'sqlite3'`

`--read-ahead-queue-size READ_AHEAD_QUEUE_SIZE`

`rosbag` 尝试在内存中保持的消息队列大小，以帮助确定性回放。较大的大小将导致更大的内存需求，但可能会防止消息回放的延迟。

`-r RATE, --rate RATE` 播放消息的速率。有效范围为 `> 0.0`。

`--topics TOPICS [TOPICS ...]`

要重新播放的话题，用空格分隔。如果未指定，则将重新播放所有话题。

--qos-profile-overrides-path QOS_PROFILE_OVERRIDES_PATH

指定一个 YAML 文件的路径，定义特定话题的 QoS 配置覆盖。

-l, --loop 在播放包文件时启用循环播放：当到达末尾时重新开始并无限循环播放。

--remap REMAP [REMAP ...], -m REMAP [REMAP ...]

要重映射的话题列表：格式为"old_topic1:=new_topic1 old_topic2:=new_topic2 等"

示例：

```
ros2 bag play record-2023-12-21-1.bag/record-2023-12-21-1.bag_0.db3
[INFO] [1662660026.601710621] [rosbag2_storage]: Opened database 'record-2023-12-21-1.bag/record-2023-12-21-1.bag_0.db3' for READ_ONLY.
```

3.2.2.3 ros2 bag2 play

Usage: ros2 bag2 play [-h] [-s {sqlite3}] [--read-ahead-queue-size READ_AHEAD_QUEUE_SIZE] [-r RATE] [--topics TOPICS [TOPICS ...]] [--qos-profile-overrides-path QOS_PROFILE_OVERRIDES_PATH] [-l] [--remap REMAP [REMAP ...]] [--storage-config-file STORAGE_CONFIG_FILE] [--clock [CLOCK]] [-d DELAY] [--disable-keyboard-controls] [-p] [--start-offset START_OFFSET] [--wait-for-all-acked TIMEOUT] [--disable-loan-message]

bag_pathPlay back ROS data from a bagpositional arguments:

bag_path 要打开的包文件

可选参数：

-h, --help 显示此帮助消息并退出

-s {sqlite3}, --storage {sqlite3}

包的存储实现。默认情况下会尝试自动检测 - 使用此参数进行覆盖。

--read-ahead-queue-size READ_AHEAD_QUEUE_SIZE

rosbag 尝试在内存中保持的消息队列大小，以帮助确定性回放。较大的大小将导致更大的内存需求，但可能会防止消息回放的延迟。

-r RATE, --rate RATE 播放消息的速率。有效范围为 > 0.0。

--topics TOPICS [TOPICS ...]

要重新播放的话题，用空格分隔。如果未指定，则将重新播放所有话题。

--qos-profile-overrides-path QOS_PROFILE_OVERRIDES_PATH

指定一个 YAML 文件的路径，定义特定话题的 QoS 配置覆盖。

-l, --loop 在播放包文件时启用循环播放：当到达末尾时重新开始并无限循环播放。

--remap REMAP [REMAP ...], -m REMAP [REMAP ...]

要重映射的话题列表：格式为"old_topic1:=new_topic1 old_topic2:=new_topic2 等"

--storage-config-file STORAGE_CONFIG_FILE

指定存储具体配置的 YAML 文件的路径。对于默认存储插件，设置通过以下语法指定：read:

pragmas: ["<setting_name>" = <setting_value>] 请注意，适用的设置仅限于 ros2 bag play 的只读模式。有关 sqlite3 设置的列表，请参考 sqlite3 文档

--clock [CLOCK] 以特定频率 (Hz) 发布到 /clock，作为 ROS 时间源。值必须为正

数。默认情况下不发布。

-d DELAY, --delay DELAY

播放之前的休眠持续时间（每次循环），单位为秒。负持续时间无效。

--disable-keyboard-controls

禁用回放的键盘控制

-p, --start-paused 在暂停状态下启动回放播放器。

--start-offset START_OFFSET

将回放播放器开始此秒数进入包文件。

--wait-for-all-acked TIMEOUT

在播放终止之前，等待所有订阅者确认所有发布的消息或直到超时（毫秒）。特别适用于在短时间内发送大尺寸的消息。负超时无效。0 表示永远等待，直到所有订阅者确认所有发布的消息。请注意，此选项仅在发布者的 QoS 配置为 RELIABLE 时有效。

--disable-loan-message

禁用作为借出消息发布。默认情况下，如果可以使用借出消息，则作为借出消息发布消息。这有助于减少数据副本的数量，因此对于发送大数据有更大的好处。

ros2 bag2 play命令还提供按键控制：

- SPACE 暂停和恢复
- CURSOR_RIGHT：回放下一帧消息
- CURSOR_UP：提高回放速率
- CURSOR_DOWN：降低回放速率

还可通过命令行 `ros2 service` 控制回放：

- `~/burst [rosbag2_interfaces/srv/Burst]`
 - 只能在播放器暂停时使用，尽可能快地按顺序发布 `num_messages`，推进播放头。
- `~/get_rate [rosbag2_interfaces/srv/GetRate]`
 - 返回当前播放速率。
- `~/is_paused [rosbag2_interfaces/srv/IsPaused]`
 - 返回播放是否暂停。
- `~/pause [rosbag2_interfaces/srv/Pause]`
 - 暂停播放。如果已经暂停，则不起作用。
- `~/play_next [rosbag2_interfaces/srv/PlayNext]`
 - 从包中播放下一条消息。仅在暂停时有效。
- `~/resume [rosbag2_interfaces/srv/Resume]`
 - 如果暂停，恢复播放。
- `~/seek [rosbag2_interfaces/srv/Seek]`
 - 将播放头更改为指定的时间戳。可以向前或向后调整时间，播放的下一条消息是在寻找的时间戳之后立即播放的消息。

- `~/set_rate [rosbag2_interfaces/srv/SetRate]`
 - 设置播放速率，例如 2.0 将以两倍速播放消息。
- `~/toggle_paused [rosbag2_interfaces/srv/TogglePaused]`
 - 如果正在播放，则暂停；如果已暂停，则恢复播放。

3.2.2.3 修改回放时使用的QoS

ros2 bag中记录了topic对应的qos策略，若需要使用自定义的qos回放，可使用以下方式：创建一个YAML文件，用来配置 topic 对应的QoS策略，YAML文件格式：

```
topic_name: str
qos_policy_name: str
...
qos_duration: object
  sec: int
  nsec: int
```

例如，我们对/XwayOS/HelloWorldTopic”设置对应的qos策略，创建一个名为"qos.yaml"的文件。

```
/XwayOS/HelloWorldTopic:
  history: keep_last
  depth: 30
```

然后通过参数 `--qos-profile-overrides-path` 来使用 .yaml 文件来重载QoS。

```
ros2 bag play record-2023-12-21-1.bag/record-2023-12-21-1.bag_0.db3 --qos-
profile-overrides-path qos.yaml
[INFO] [1662661602.197208624] [rosbag2_storage]: Opened database 'record-
2023-12-21-1.bag/record-2023-12-21-1.bag_0.db3' for READ_ONLY.
[INFO] [1662661602.197598640] [rosbag2_transport]: Overriding QoS profile
for topic /XwayOS/HelloWorldTopic
```

YAML文件中QoS字段对应的类型如下：

```
history: [keep_all, keep_last]
depth: int
reliability: [system_default, reliable, best_effort, unknown]
durability: [system_default, transient_local, volatile, unknown]
deadline:
  sec: int
  nsec: int
lifespan:
  sec: int
```



```
nsec: int
liveliness: [system_default, automatic, manual_by_node, manual_by_topic,
unknown]
liveliness_lease_duration:
  sec: int
  nsec: int
avoid_ros_namespace_conventions: [true, false]
```

4 问题及诊断方法

4.1 ros2 topic 无法监控

诊断方法：

- 确保通讯端与ros2镜像中的DOMAIN_ID保持一致。
- 确保topic的名称以rt/开始。在ros2保存的topic名称会将rt/删除。
- 确保配置的qos.xml中transport_id存在UDPV4_BASE，该配置可以配置多个。
- 确保配置的qos.xml中interfaceWhiteList存在通讯的所有IP地址，这里不建议设置该配置项，该配置可以配置多个。

4.2 ros2 topic echo 显示失败

诊断方法：

- 确保通讯端与ros2镜像中的DOMAIN_ID保持一致。
- echo指定topic名称时，只需要提供去除rt/之后的topic名称。
- 确保已经将新生成的msg信息添加到环境变量中，执行source /home/ros2workspace/install/local.setup.bas。

4.3 ros2 bag record 录制失败

诊断方法：

- 确保通讯端与ros2镜像中的DOMAIN_ID保持一致。
- 确保已经将新生成的msg信息添加到环境变量中，执行source /home/ros2workspace/install/local.setup.bas。
- 确保topic的名称使用(_)下划线或者/斜杠来分割字母或者数字，在分割的字段中必须以字母开头Xway0S/xway_777，不可以以数字开头Xway0S/777_xway。也不允许在分割字段中单独存在数字Xway0S/777/。

4.3 ros2 bag play 回放失败

诊断方法：

- 确保通讯端与ros2镜像中的DOMAIN_ID保持一致。
- 确保已经将新生成的msg信息添加到环境变量中，执行source /home/ros2workspace/install/local.setup.bas。

- 确保已经将`userData`导入到了环境变量中，`export ROS_USER_DATA="ara.com://services/2438_71-1.0&2400_70-1.0"`。环境变量格式为`ara.com://services/ServiceID_InstanceID-VersionID&ServiceID_InstanceID-VersionID`。