RTI Persistence Service

Release Notes

Version 7.1.0

Trademarks

Copy and Use Restrictions

Notices

Early Access Software

(4) any such suggestions or ideas you provide regarding the Software (collectively , "Feedback"), may be used and exploited in any and every way by RTI (including without limitation, by granting sub-licenses), on a non-exclusive, perpetual, irrevocable, transferable, and worldwide basis, without any compensation, without any obligation to report on such use, and without any other restriction or obligation to you; and

(5) TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT WILL RTI BE LIABLE TO YOU FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, EXEMPLARY OR PUNITIVE OR CONSEQUENTIAL DAMAGES OF ANY KIND, OR FOR LOST PROFITS, LOST DATA, LOST REPUTATION, OR COST OF COVER, REGARDLESS OF THE FORM OF ACTION WHETHER IN CONTRACT, TORT (INCLUDING WITHOUT LIMITATION, NEGLIGENCE), STRICT PRODUCT LIABILITY OR OTHERWISE, WHETHER ARISING OUT OF OR RELATING TO THE USE OR INABILITY TO USE THE SOFTWARE, EVEN IF RTI HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES."

Deprecations and Removals

Any deprecations or removals noted in this document serve as notice under the Real-Time Innovations, Inc. Maintenance Policy #4220 and/or any other agreements by and between RTI and customer regarding maintenance and support of RTI's software.

Deprecated means that the item is still supported in the release, but will be removed in a future release. *Removed* means that the item is discontinued or no longer supported. By specifying that an item is deprecated in a release, RTI hereby provides customer notice that RTI reserves the right after one year from the date of such release and, with or without further notice, to immediately terminate maintenance (including without limitation, providing updates and upgrades) for the item, and no longer support the item, in a future release.

Technical Support

Real-Time Innovations, Inc.
232 E. Java Drive
Sunnyvale, CA 94089
Phone: (408) 990-7444
Email: support@rti.com
Website: https://support.rti.com/

Contents

1 Supported Platforms

RTI® Persistence Service is included with *RTI Connext®*. If you choose to use it, it must be installed on top of *Connext* with the same version number.

See the column for *Persistence Service* in the table of Supported Platforms for Compiler-Dependent Products, in the RTI Connext Core Libraries Release Notes.

RTI tests *Persistence Service* with a file-system only, using PERSISTENT mode.

2 Compatibility

For backward-compatibility information between this and previous releases, see the *Migration Guide* on the RTI Community Portal (https://community.rti.com/documentation).

3 What's New in 7.1.0

3.1 Persistence Service support as a library for all supported architectures

This release adds support for *Persistence Service* Library API (static and dynamic), as an alternative to the standalone executable available in previous releases. Now you can run a *Persistence Service* instance within your application by linking with the new library and using the C API offered by the library on all supported architectures. Previously this support was available only for Integrity®.

For additional information on the *Persistence Service* Library API, see the *API Reference HTML documentation in* ***<NDDSHOME>/doc/api/persistence_service/index.html***. For an example on how to use *Persistence Service* as a library see https://github.com/rticommunity/rticonnextdds-examples/tree/release/7.1.0/examples/persistence_service/library_api.

3.2 Removed ability to share a database connection in Persistence Service and durable writer history

This release removes the ability to share a database connection in *RTI Persistence Service* (which is done by setting the tag <share_database_connection> to true for a <persistence_group>). It also removes the ability to share a database connection when using durable writer history and setting the property **dds.data_writer.history.odbc_plugin.builtin.shared** to 1.

Note that sharing a database connection was only allowed for external databases, and support for external databases was removed in 7.0.0 (see *RTI Connext Core Libraries What's New* in 7.1.0).

3.3 Third-party software upgrade

The following third-party software used by *Persistence Service* has been upgraded:

Third-Party Software	Previous Version	Current Version
SQLite®	3.39.0	3.39.4

For information on third-party software used by *Connext* products, see the "3rdPartySoftware" documents in your installation: **<NDDSHOME>/doc/manuals/connext_dds_professional/release_notes_3rdparty**.

4 What's Fixed in 7.1.0

4.1 Persistence Service stored/forwarded samples multiple times when there were two or more equivalent versions of a type for a Topic

Persistence Service stored and forwarded incoming samples multiple times when there were two or more equivalent versions of a type for a given *Topic* in the system.

Two types are equivalent when they only differ on typedef. For example, MyType and MyType2 are equivalent in this IDL snippet:

```
struct MyType {
    long m1;
};

typedef long MyLong;

struct MyType2 {
    MyLong m1;
};
```

This problem has been fixed.

[RTI Issue ID PERSISTENCE-269]

4.2 Persistence Service XSD schema was broken

In release 7.0.0, the *Persistence Service* XSD schema was broken due to an additional closing tag. This was a regression that only affected the 7.0.0 release. This issue has been fixed.

[RTI Issue ID PERSISTENCE-276]

4.3 Unexpected fatal error when number of instances reached the limit

In 7.0.0, an unexpected fatal error could be logged when the following occurred:

- *Persistence Service* was running in PERSISTENT mode.
- The number of instances reached the **max_instances** limit set in one of the *Persistence Service DataWriters'* RESOURCE_LIMITS QoS.
- *Connext* could not find an instance to delete (such as an unregistered one), to replace with the new instance. So the new instance could not be added.

This log message is expected, but it is not a fatal error, so its verbosity has been updated to WARNING, as follows:

```
WriterHistoryOdbcPlugin_createResources:FIND FAILURE | Instance for replacement
WriterHistoryOdbcPlugin_addInstance:OUT OF RESOURCES | Exceeded the number of instances.
Current registered instances (128), maximum number of instances (128)(writer_qos.resource_
limits.max_instances)
```

[RTI Issue ID CORE-13496]

4.4 Fixes related to vulnerabilities

4.4.1 Potential arbitrary SQL query execution when enabling database locking

There was the potential for arbitrary SQL query execution in *Persistence Service* running with database locking enabled (which is not the default setting). This issue has been fixed.

4.4.1.1 User Impact without Security

A SQL Injection vulnerability in *Persistence Service* could have resulted in the following:

- Arbitrary SQL query execution.
- Exploitable from the same host Persistence Service is running.
- Potential impact on integrity and confidentiality of Persistence Service.
- CVSS Base Score: 7.1 HIGH
- CVSS v3.1 Vector: AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:N

4.4.1.2 User Impact with Security

Same impact as described for "User Impact without Security" above.

[RTI Issue ID PERSISTENCE-272]

5 Previous Release

5.1 What's New in 7.0.0

5.1.1 Support for external databases is discontinued

External databases are no longer supported by Persistence Service.

The Release Notes for 6.1.1 included a deprecation notice, in keeping with the Real-Time Innovations, Inc. Maintenance Policy #4220.

5.1.2 Default journal_mode and synchronization changed to WAL and NORMAL, respectively

In this release, the default values for the following configuration parameters have changed:

- <journal_mode> has changed from DELETE to WAL
- <synchronization> has changed from OFF to NORMAL

This change provides the best out-of-the-box performance without sacrificing database integrity in the event of a crash or power failure.

5.1.3 Third-party software upgrade

The following third-party software used by *Persistence Service* has been upgraded:

Third-Party Software	Previous Version	Current Version
SQLite®	3.37.2	3.39.0

For information on third-party software used by *Connext* products, see the "3rdPartySoftware" documents in your installation: **<NDDSHOME>/doc/manuals/connext_dds_professional/release_notes_3rdparty**.

5.2 What's Fixed in 7.0.0

5.2.1 Schema files not compliant with DDS-XML specification

The following change has been made to the schema file **rti_persistence_service.xsd**, and its included files, to make them compliant with the DDS-XML specification (https://www.omg.org/spec/DDS-XML/1.0/PDF):

- Renamed <participant_qos> to <domain_participant_qos>

The old tag is still accepted by the Connext XML parser and the XSD schema to maintain backward compatibility.

[RTI Issue ID PERSISTENCE-213]

5.2.2 Samples published out of order from same virtual GUID were dropped

If *Persistence Service* received samples for a given virtual GUID with sequence numbers out of order, *Persistence Service* dropped samples with sequence numbers lower than the highest received sequence number. This issue has been resolved.

[RTI Issue ID PERSISTENCE-250]

5.2.3 Fatal error when persisting unkeyed Topics upon restore or IP mobility event

Persistence Service generated the following fatal error and shut down when persisting unkeyed Topics if all of the following conditions were met:

- <use_durability_service> was set to true in the <persistence_group> OR <writer_qos>/<writer_data_lifecycle>/<autopurge_disposed_instances_delay> was set to zero in the <persistence_group>
- <writer_in_memory_state> was set to false in the <persistence_group>.
- There was an IP mobility event (for instance, an interface went down) OR *Persistence Service* was started with the **-restore** command-line option set to true.

The error backtrace was as follows:

```
#4  WriterHistoryOdbcPlugin_logAndCheckODBCError ??:? [0x31590B]
#5  WriterHistoryOdbcPlugin_handleODBCError ??:? [0x315CE5]
#6  WriterHistoryOdbcPlugin_beginDisposedInstanceIteration ??:? [0x34B202]
```

This problem has been resolved.

[RTI Issue ID PERSISTENCE-255]

6 Known Issues

Note: For an updated list of critical known issues, see the Critical Issues List on the RTI Customer Portal at https://support.rti.com.

6.1 Coherent Changes not Propagated as Coherent Set

Persistence Service will propagate the samples inside a coherent change. However, it will propagate these samples individually, not as a coherent set.

6.2 TopicQueries not Supported in PERSISTENT Mode

Getting TopicQuery data from a *Persistence Service* instance configured to store data on disk is not currently supported.

> **Note:** Getting TopicQuery data from a *Persistence Service* instance running in TRANSIENT (storing data in memory) mode is supported.

[RTI Issue ID PERSISTENCE-143]

6.3 <comm_ports> not Supported when Using Real-Time WAN Transport

Persistence Service can use the *RTI Real-Time WAN Transport*. However, the port configuration using <comm_ports> or the property **dds.transport.UDPv4_WAN.builtin.comm_ports** is not currently supported by *Persistence Service*.

[RTI Issue ID PERSISTENCE-206]

6.4 Persistence Service DataReaders Ignore Serialized Key Propagated with Dispose Updates

Persistence Service DataReaders ignore the serialized key propagated with dispose updates. *Persistence Service DataWriters* cannot propagate the serialized key with dispose, and therefore ignore the **serialize_key_with_dispose** setting on the *DataWriter* QoS.

[RTI Issue ID PERSISTENCE-221]

7 Available Documentation

The following documentation is provided with the *Persistence Service* distribution. (The paths show where the files are located after *Persistence Service* has been installed in **<NDDSHOME>**):

- General information, configuration, use cases, and execution of *Persistence Service*: *RTI Connext Core Libraries User's Manual* (**<NDDSHOME>/doc/manuals/connext_dds_professional/users_manual/RTI_ConnextDDS_CoreLibraries_UsersManual.pdf**)

- Example code

  By default, the *Persistence Service* examples are copied here:

  - macOS systems:

    /Users/*your user name*/rti_workspace/*version*/examples/persistence_service/<*language*>/hello_world_persistence
  - Linux systems:

    /home/*your user name*/rti_workspace/*version*/examples/persistence_service/<*language*>/hello_world_persistence
  - Windows systems:

    <*your home directory*>\rti_workspace*version*\examples\persistence_service\<*language*>/hello_world_persistence
- Overview of persistence and durability features:
  Open **<NDDSHOME>/ReadMe.html**, choose your desired API (C, C++, or Java), then select **Modules**, **RTI Connext API Reference**, **Durability and Persistence**.