

RTI DDS Toolkit

Release Notes

Version 2.1.1



Your systems. Working as one.



© 2017 Real-Time Innovations, Inc.
All rights reserved.
Printed in U.S.A. First printing.
November 2017.

Trademarks

Real-Time Innovations, RTI, NDDS, RTI Data Distribution Service, DataBus, Connex, Micro DDS, the RTI logo, 1RTI and the phrase, "Your Systems. Working as one," are registered trademarks, trademarks or service marks of Real-Time Innovations, Inc. All other trademarks belong to their respective owners.

Copy and Use Restrictions

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form (including electronic, mechanical, photocopy, and facsimile) without the prior written permission of Real-Time Innovations, Inc. The software described in this document is furnished under and subject to the RTI software license agreement. The software may be used or copied only under the terms of the license agreement.

Technical Support

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

Contents

1	Supported Platforms.....	1-1
2	Compatibility	1-1
2.1	Incompatible with Older Versions of Connex DDS using UDPv6 and Shared Memory	1-1
2.2	Incompatible with Older Versions of Connex DDS.....	1-2
2.3	Potential Incompatibility when Regenerating Complex Data Types that were Originally Created in 2.0.0.104.....	1-2
2.4	Additional Steps when Upgrading from a Release Older than 1.2.0.90	1-2
2.5	Additional Steps when Upgrading from a Release Older than 1.3.0.91	1-2
2.6	Required: VI Package Manager Version 2014.....	1-2
2.7	Toolkit Uses String Length of 1024.....	1-3
2.8	Improved Performance when Managing Large Data.....	1-3
3	What's Fixed in 2.1.1	1-3
3.1	Wrong Path to Getting Started Guide in Generated Examples.....	1-3
3.2	Improper Default Name for Simple Strings Caused Code Generation Error.....	1-3
3.3	Security Library not Loaded from 'data' Folder when Creating Executable	1-3
3.4	Wrong Value for Property pool_buffer_max_size in RTI_LABVIEW_CONFIG.documentationONLY.xml File	1-4
3.5	Crash after Any Error while Loading Library.....	1-4
3.6	Crash when Using Arrays in LabVIEW 2017	1-4
3.7	Strange Characters Appended to Enum Element Names.....	1-4
3.8	Memory Leak when Writing Sequences of Doubles	1-4
3.9	Inability to Unregister a Type in Rare Cases.....	1-4
3.10	Possible DDS Errors when Completely Deleting DomainParticipant	1-4
3.11	Small Memory Leak when Creating Entities	1-4
3.12	ContentFilteredTopic could not be Reused after DataReader Creation Failed.....	1-4
4	Previous Releases	1-5
4.1	What's New in 2.1.0.....	1-5
4.1.1	Ability to Provide String Size from LabVIEW	1-5
4.1.2	Ability to Dispose or Unregister Instances.....	1-5
4.1.3	Ability to Read Samples whose INSTANCE_STATE is Different than ALIVE	1-5
4.1.4	Example Generated Types now Password Protected.....	1-5
4.2	What's Fixed in 2.1.0.....	1-5
4.2.1	Inconsistent Error Message when Deletion of Topic not Done Correctly	1-5
4.2.2	LabVIEW Crashed if Read or Write subVI Called with NULL Value	1-6
4.2.3	Potential Crash Getting Distributed Logger Info	1-6
4.2.4	Topic 'refcount' not Correctly Updated when Greater than 256	1-6
4.2.5	Potential Memory Leak when Setting Reader Condition	1-6

4.2.6	Input Pin 'data' in Generated Write VI was not in Correct Position	1-6
4.2.7	Inconsistent State when Setting Writer/Reader QoS at Run Time	1-6
4.2.8	Potential Error Reading Samples when History Kind KEEP_ALL	1-6
4.2.9	Potential Crash when Setting Reader/Writer QoS at Run Time	1-6
4.2.10	Potential Memory Leaks with Some Specific Configurations	1-7
4.3	What's New in 2.0.0.....	1-7
4.3.1	Product Name Change	1-7
4.3.2	Support for ContentFilteredTopics	1-7
4.3.3	Ability to Get All Values from a Secure Profile	1-7
4.3.4	New Error Messages upon Failed Secure Entity Creation.....	1-7
4.3.5	Changes to All RTI DDS Toolkit VIs	1-7
4.3.6	Wizard for Creating Custom Complex Data Types SubVIs	1-7
4.3.7	Increased participant_property_string_max_length	1-8
4.3.8	New Examples.....	1-8
4.3.9	Additional Installation of OpenSSL No Longer Needed	1-8
4.3.10	Windows and RT Target Bundles Merged	1-8
4.3.11	Internal VIs now Password-Protected	1-8
4.3.12	Updates to CDFs for Installing RT Bundle.....	1-8
4.3.13	RTI DDS Toolkit now Wire Aligned with DDS Security Specification.....	1-8
4.3.14	RTI DDS Toolkit is under NI License Agreement.....	1-8
4.3.15	Deleted 'Force New DomainParticipant' Flag.....	1-8
4.3.16	Previously Deprecated DDS Create Reader/Writer VIs have been Deleted	1-9
4.4	What's Fixed in 2.0.0.....	1-9
4.4.1	TypeName was Truncated to 128 Characters	1-9
4.4.2	Arrays of Enums were Unsupported	1-9
4.4.3	Possible Crash when Opening Administration Panel if External QoS Profile Loaded Incorrectly	1-9
4.4.4	Update to Description for Error 5048.....	1-9
4.4.5	Error Deleting Topics	1-9
4.4.6	Crash when Connecting Reader Output ref num to Writer Input ref num (and Vice Versa)	1-9
4.4.7	Error 5058 when Reading in Parallel with Reentrant Read SubVI	1-9
4.4.8	Error Loading RTI DDS Secure Plugins using Standalone Application	1-10
5	Known Issues.....	1-10
5.1	Reader/Writer Create SubVIs Fail if QoS Settings not Provided	1-10
5.2	Monitoring Library cannot be Used as DomainParticipant's Base Profile when Creating Custom Secure Profile.....	1-10
5.3	Possible Crash when using Query Conditions.....	1-10
6	Additional Documentation.....	1-10

Release Notes

1 Supported Platforms

RTI® *DDS Toolkit* is supported on these platforms:

- ❑ Windows® Systems:
 - Windows 7 SP1 (32-bit and 64-bit)
 - Windows 8.1 (32-bit and 64-bit)
 - Windows 10 (32-bit and 64-bit)
 - Windows Server 2008 R2 SP1 (64-bit)
 - Windows Server 2012 R2 (64-bit)

On 64-bit platforms, it runs in 32-bit mode.

- ❑ Real-Time Targets:
 - NI™ Linux® 3 on ARMv7 CPU (tested on cRIO-9068 target)
 - NI Linux 3 on 64-bit Intel CPU (tested on cRIO-9031 target)

You will also need:

- ❑ National Instruments® LabVIEW® 2015 SP1 or later (32-bit)
- ❑ JKI VI Package Manager 2014 or later

2 Compatibility

2.1 Incompatible with Older Versions of Connex DDS using UDPv6 and Shared Memory

RTI Connex® *DDS* 5.1.0 and earlier releases used a UDPv6 locator kind that was not compliant with the value in the RTPS specification. The value used in *Connex DDS* 5.1.0 was 5 while the RTPS specification specifies a value of 2. Because of this issue, *Connex DDS* could not interoperate with other DDS vendors over UDPv6.

This problem is resolved starting with 5.2.0. Note, however, that out-the-box backward compatibility with *Connex DDS* 5.1.0 and lower, when using both the UDPv6 and SHMEM transports, is broken.

See the *RTI Connex DDS Core Libraries Release Notes'* section on Transport Compatibility for information on how to resolve this compatibility issue.

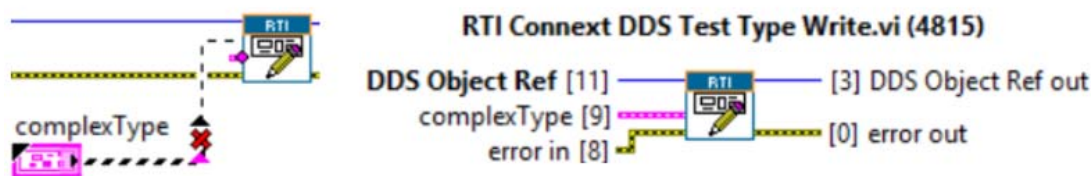
2.2 Incompatible with Older Versions of Connex DDS

In *Connex DDS* 5.1.0, the default `message_size_max` for the UDPv4, UDPv6, TCP, Secure WAN, and shared-memory transports changed to provide better out-of-the-box performance. *RTI DDS Toolkit* 1.1.0 and higher also uses the new default value for `message_size_max`. Consequently, *RTI DDS Toolkit* 1.1.0 and higher is not out-of-the-box compatible with applications running older versions of *Connex DDS* or *RTI Data Distribution Service*.

See the *RTI Connex DDS Core Libraries Release Notes'* section on Transport Compatibility for instructions on how to resolve this compatibility issue.

2.3 Potential Incompatibility when Regenerating Complex Data Types that were Originally Created in 2.0.0.104

The 'complexType' input of the previous *Write* subVI has been changed from the 2nd left pin to the 3rd one. This doesn't affect the functionality if the *Write* subVI is not regenerated (because it is saved and used as it was originally generated). However if you regenerate the type, the VIs that were using it will no longer be able to use it until the wire is reconnected to the correct pin.



2.4 Additional Steps when Upgrading from a Release Older than 1.2.0.90

If you are upgrading from a release older than 1.2.0.90, there are important steps you must take. Briefly, changes are required because:

- The *Create Reader/Writer* subVIs have been deprecated. We strongly recommend that you upgrade the VIs to use the *Simple Create Reader/Writer* or *Advanced Create Reader/Writer*.
- The *RTI DDS Toolkit* library name changed from `lvdds.1.0.dll` to `rtilvdds.dll`.

See Sections 1.5 and 1.5.1 in the *Getting Started Guide* for details.

2.5 Additional Steps when Upgrading from a Release Older than 1.3.0.91

If you are upgrading from a release older than 1.3.0.91, there are important steps you must take. Briefly, changes are required because:

- The DDS Write/Read Object Reference has been removed for all our VIs.
- The configuration of several Call Library Function Nodes has been modified and they no longer receive a reference as a parameter.

See Sections 1.5 and 1.5.2 in the *Getting Started Guide* for details.

2.6 Required: VI Package Manager Version 2014

The *RTI DDS Toolkit* requires VI Package Manager (VIPM) to be installed. Due to a limitation in VIPM, version 2014 of VIPM is required. For details visit JKI website: <http://support.jki.net/entries/66745297-VIPM-2013-cannot-install-packages-built-in-VIPM-2014>.

2.7 Toolkit Uses String Length of 1024

In *RTI DDS Toolkit*, the default string length is 1024 characters. This may create incompatibilities with other DDS data types in your system that use string lengths != 1024. See Section 6.2 in the *Getting Started Guide* to learn how to change the string length. (RTI Issue ID LABPLG-565)

2.8 Improved Performance when Managing Large Data

We have improved the performance when managing large data by setting these properties. (Note: they cannot be changed from the QoS XML file):

❑ Dynamic data:

- `serialization.trim_to_size = DDS_BOOLEAN_TRUE`
- `serialization.max_size_serialized = DDS_LENGTH_UNLIMITED`
- `serialization.min_size_serialized = TypeCode's minimum serialized size`

❑ Data Writer:

- `dds.data_writer.history.memory_manager.fast_pool.pool_buffer_max_size = 1024`

❑ Data Reader:

- `dds.data_reader.history.memory_manager.fast_pool.pool_buffer_max_size = 1024`

3 What's Fixed in 2.1.1

3.1 Wrong Path to Getting Started Guide in Generated Examples

The generated examples pointed to an incorrect path to the *Getting Started Guide*. This problem has been resolved.

[RTI Issue ID LABPLG-577]

3.2 Improper Default Name for Simple Strings Caused Code Generation Error

The default struct name for simple String types was `DDS_String`. However, that is a reserved word in the IDL files. Therefore *RTI DDS Toolkit* couldn't use the IDL generated by *Admin Console* to create an application with *Code Generator*. This name has been changed to `StringStruct`.

[RTI Issue ID LABPLG-581]

3.3 Security Library not Loaded from 'data' Folder when Creating Executable

When creating an executable using the LabVIEW Wizard, the external libraries are usually copied to a 'data' folder in the same location as the application executable.

In the previous release, *RTI DDS Toolkit* could only load the `nddssecurity.dll` library if it was in the same folder as the executable (it didn't look for it the 'data' folder).

This problem has been resolved. Now the library can be loaded from either (a) the same folder as the executable or (b) the 'data' folder. Note, however, that *all* the libraries must be in the same folder.

[RTI Issue ID LABPLG-592]

3.4 Wrong Value for Property `pool_buffer_max_size` in `RTI_LABVIEW_CONFIG.documentationONLY.xml` File

The file `RTI_LABVIEW_CONFIG.documentationONLY.xml` under the folder `<LabVIEW installation folder>/vi.lib/RTI DDS Toolkit/` had an incorrect value of 4096 for the property `history.memory_manager.fast_pool.pool_buffer_max_size`, in both `reader_qos` and `writer_qos`. This problem has been resolved. Now the correct value of 1024 is used.

[RTI Issue ID LABPLG-605]

3.5 Crash after Any Error while Loading Library

If an error occurred while the library was being initialized, LabVIEW would crash. This problem has been resolved. Now the error will be reported but LabVIEW will not crash because of it.

[RTI Issue ID LABPLG-616]

3.6 Crash when Using Arrays in LabVIEW 2017

If you were using arrays (that is, the `forceArrayMapping?` flag was ON) you may have seen a crash in LabVIEW 2017. This problem has been resolved.

[RTI Issue ID LABPLG-622]

3.7 Strange Characters Appended to Enum Element Names

When creating an enum, strange characters may have been appended to the text values of enum elements. This problem has been resolved.

[RTI Issue ID LABPLG-623]

3.8 Memory Leak when Writing Sequences of Doubles

Writing sequences of doubles may have caused a memory leak. This problem has been resolved.

[RTI Issue ID LABPLG-624]

3.9 Inability to Unregister a Type in Rare Cases

In some situations when using `ContentFilteredTopics`, a created Type (identified by a `Type-Name`) could not be unregistered. This problem has been resolved.

[RTI Issue ID LABPLG-631]

3.10 Possible DDS Errors when Completely Deleting `DomainParticipant`

DDS errors related to `ContentFilteredTopics` may have been logged when a `DomainParticipant` was completely deleted. This problem has been resolved.

[RTI Issue ID LABPLG-632]

3.11 Small Memory Leak when Creating Entities

There was a small memory leak when creating the data type for `Writer/Readers`. This problem has been resolved.

[RTI Issue ID LABPLG-634]

3.12 `ContentFilteredTopic` could not be Reused after `DataReader` Creation Failed

If there were errors during the creation of a `DataReader` that would have used a `ContentFiltered-Topic`, that `ContentFilteredTopic` could not be reused. This problem has been resolved.

[RTI Issue ID LABPLG-662]

4 Previous Releases

This section includes:

- [What's New in 2.1.0 \(Section 4.1\)](#)
- [What's Fixed in 2.1.0 \(Section 4.2\)](#)
- [What's New in 2.0.0 \(Section 4.3\)](#)
- [What's Fixed in 2.0.0 \(Section 4.4\)](#)

4.1 What's New in 2.1.0

4.1.1 Ability to Provide String Size from LabVIEW

Previously, there were only two different string sizes in DDS: 1024 and unbounded. Now you can provide a specific value for the string length from LabVIEW. To do so, write a number with the string size as the value of that string as the type input in the *Create Reader/Writer* subVI.

For example, if you want to create an array of 75 characters, connect an array whose content is "75" to the *Create Reader/Writer* subVI.

For complex types, we recommend setting a default value when you create the LabVIEW type definition (*.ctl).

4.1.2 Ability to Dispose or Unregister Instances

Now you can dispose or unregister an instance from a LabVIEW writer. A new input has been added to the *Write* subVI. This input is an enum with the following fields:

- WRITE (default): Write the sample that has been connected as input. Current behavior.
- DISPOSE: Dispose the instance defined by the provided key on the connected input 'data'.
- UNREGISTER: Unregister the instance defined by the provided key on the connected input 'data'.

4.1.3 Ability to Read Samples whose INSTANCE_STATE is Different than ALIVE

Now you can read DDS samples whose INSTANCE_STATE is not ALIVE. You can read NOT_ALIVE_NO_WRITERS and NOT_ALIVE_DISPOSED samples. This is only available when using an Exclusive Reader without forcing read (that is, when the **ForceRead?** flag in the Advanced Reader Configuration is disabled).

4.1.4 Example Generated Types now Password Protected

Previously the support subVIs that were generated using the ComplexType Generator were not password protected. This has been changed so now they are password protected.

4.2 What's Fixed in 2.1.0

4.2.1 Inconsistent Error Message when Deletion of Topic not Done Correctly

Trying to delete a Topic with the same Topic Name but a different Type Name was shown as a Warning instead of an Error. The error message that was shown after triggering this error has also been modified to better represent the possible cause.

[RTI Issue ID LABPLG-559]

4.2.2 LabVIEW Crashed if Read or Write subVI Called with NULL Value

If the *Read* or *Write* subVI was called with a NULL value (DDS Object Ref), LabVIEW would crash. This problem has been resolved.

[RTI Issue ID LABPLG-562]

4.2.3 Potential Crash Getting Distributed Logger Info

A crash may occurred when receiving the Distributed Logger Info if an error happened during initialization. This problem has been resolved.

[RTI Issue ID LABPLG-569]

4.2.4 Topic 'refcount' not Correctly Updated when Greater than 256

The topic **refcount** was not correctly updated when that number was higher than 256. This problem has been resolved.

[RTI Issue ID LABPLG-576]

4.2.5 Potential Memory Leak when Setting Reader Condition

Under a certain condition there may have been a memory leak while setting any reader condition in one string sequence. This problem has been resolved.

[RTI Issue ID LABPLG-583]

4.2.6 Input Pin 'data' in Generated Write VI was not in Correct Position

The pin where the input data is connected in the generated complexType *Write* subVI was not in the same position as the *Write* subVI for simple types. This problem has been resolved. Now, new generated Complextype *Write* subVIs will have the input in the same position as in simple types.

[RTI Issue ID LABPLG-584]

4.2.7 Inconsistent State when Setting Writer/Reader QoS at Run Time

Using the *Set Writer/Reader Qos* subVIs may have put your application in an inconsistent state (that is, the entities were not working as they were configured). This problem has been resolved.

[RTI Issue ID LABPLG-587]

4.2.8 Potential Error Reading Samples when History Kind KEEP_ALL

It was possible to have a configuration with History kind set to KEEP_ALL and exclusive readers forced to use **read()** instead of **take()**. However when using this configuration, only the first sample was shown, and after some time the reader and writer queues became full. Then an error was shown. This configuration is no longer allowed.

[RTI Issue ID LABPLG-589]

4.2.9 Potential Crash when Setting Reader/Writer QoS at Run Time

There may have been a crash when using the *Set Reader/Writer QoS* subVIs if the error wire was not connected to the subVI and the *Create Reader/Writer* subVIs got an error. This problem has been resolved.

[RTI Issue ID LABPLG-590]

4.2.10 Potential Memory Leaks with Some Specific Configurations

To avoid memory leaks, some configurations in a LabVIEW reader are no longer available. Such configurations had one or more of the following parameters:

- History.kind and history.depth: Provided QoS
- Exclusive reader: Advanced Reader Configuration
- Force Read (only exclusive readers): Advanced Reader Configuration
- Only new samples: *Read* subVI parameter
- Instance state: ANY_INSTANCE_STATE or ALIVE_INSTANCE_STATE are automatically chosen to avoid memory leaks (keeping the current functionality).

[RTI Issue ID LABPLG-591]

4.3 What's New in 2.0.0

4.3.1 Product Name Change

The name *RTI DDS Toolkit for LabVIEW* has been replaced with *RTI DDS Toolkit*.

4.3.2 Support for ContentFilteredTopics

ContentFilteredTopics are now supported. To create them, they need to be specified in the *Advanced Create Reader VI*. This means the ContentFilteredTopic cannot be modified once it is running.

Currently the only supported **Filter Type** is **DDS_SQLFILTER_NAME**. If any field of 'Content-FilteredTopic Name' or 'Filter Expression' is empty, the ContentFilteredTopic will not be created.

A ContentFilteredTopic is owned by a DDS Reader. As long as the DDS Reader is still alive, the ContentFilteredTopic will also be alive.

4.3.3 Ability to Get All Values from a Secure Profile

In the previous release, only the *names* of previously created Secure Profiles could be recovered. Now DDS Toolkit can load *all* the information contained in these profiles. You can display these values with the *Get Security Profile Values VI*.

4.3.4 New Error Messages upon Failed Secure Entity Creation

New error messages have been added to report when an error occurs creating a Secure DataWriter or DataReader.

4.3.5 Changes to All RTI DDS Toolkit VIs

All the VIs included in *RTI DDS Toolkit* have been modified in order to be more LabVIEW compliant.

4.3.6 Wizard for Creating Custom Complex Data Types SubVIs

A wizard for creating custom complex data types subVIs has been added. You can find this wizard under the menu **Tools/RTI DDS Toolkit/RTI DDS ComplexType Generator**.

This wizard allows you to create the minimal set of subVIs needed to run a *Connex DDS* application for any type. A Creation writer and reader, as well as write and read subVIs, will be generated based on the provided Type Definition (*.ctl).

You can choose Simple or Advanced creation. You can also choose whether a simple example is created for this type using this generated subVIs.

4.3.7 Increased participant_property_string_max_length

Because the Secure DDS properties point to some files using the full path, the previous value of `participant_property_string_max_length` was not large enough. This value has been increased to 4096.

4.3.8 New Examples

Two new examples have been added to *RTI DDS Toolkit*. One of them shows how to use `ContentFilteredTopics`. The other one shows how to use *RTI Secure DDS Plugins* in *RTI DDS Toolkit*.

4.3.9 Additional Installation of OpenSSL No Longer Needed

Previously, to use *RTI Secure DDS Plugins* with *RTI DDS Toolkit*, you had to install OpenSSL separately. Now *RTI DDS Toolkit* will use OpenSSL that is installed with LabVIEW. This affects both supported operating systems: Windows and NI Linux (RT Targets).

4.3.10 Windows and RT Target Bundles Merged

The *RTI DDS Toolkit* bundles for Windows platforms and RT Targets are now shipped in the same package.

A correct installation of the full package may require administration privileges, otherwise it may not be correctly installed.

4.3.11 Internal VIs now Password-Protected

Some internal VIs which make direct calls to a Call Library Function are now protected by a password.

4.3.12 Updates to CDFs for Installing RT Bundle

The Component Definition Files (CDFs) that are used to install *RTI DDS Toolkit* in a Real-Time Target have been updated.

4.3.13 RTI DDS Toolkit now Wire Aligned with DDS Security Specification

RTI DDS Toolkit is now wire aligned with the DDS Security Specification.

For the Shared Secret Algorithm:

- `rsa` is no longer an option.
- `ecdsa-ecdh` and `dsa-dh` have been renamed to `ecdh` and `dh`, respectively. Their meanings remain the same.

The value `aes-128-ctr` is no longer an option for the Encryption Algorithm.

4.3.14 RTI DDS Toolkit is under NI License Agreement

Since *RTI DDS Toolkit* is now part of LabVIEW 2017, it will follow the same license that LabVIEW does. Both of them are under the NI License Agreement.

4.3.15 Deleted 'Force New DomainParticipant' Flag

The 'Force New DomainParticipant' flag is no longer used. This flag has been deleted from the "Advanced Reader Configuration" and "Advanced Writer Configuration" clusters. If these clusters were being used in any VIs from a previous release of DDS Toolkit, they may need to be modified manually.

4.3.16 Previously Deprecated DDS Create Reader/Writer VIs have been Deleted

The old *DDS Create Reader/Writer* VIs, which were deprecated in version 1.2.0.90, have been deleted from *DDS Toolkit*.

4.4 What's Fixed in 2.0.0

4.4.1 TypeName was Truncated to 128 Characters

Previously the `TypeName` may have been truncated to 128 characters, even though the maximum length was 256 characters. This problem has been resolved.

[RTI Issue ID LABPLG-432]

4.4.2 Arrays of Enums were Unsupported

Previously the documentation stated that *RTI DDS Toolkit* supported arrays of enums. However, that type (or others that included it) couldn't be created. This problem has been resolved. Arrays of enums are now supported.

[RTI Issue ID LABPLG-483]

4.4.3 Possible Crash when Opening Administration Panel if External QoS Profile Loaded Incorrectly

RTI DDS Toolkit may have crashed when opening the Administration Panel. This occurred if an external QoS profile was loaded incorrectly. This problem has been resolved.

[RTI Issue ID LABPLG-494]

4.4.4 Update to Description for Error 5048

Error 5048 is reported when two incompatible QoS profiles (e.g., having the same profile name) are loaded. Previously the description for this error was inaccurate. This problem has been resolved.

[RTI Issue ID LABPLG-499]

4.4.5 Error Deleting Topics

Sometimes a Topic was not deleted even though no `DataWriters` or `DataReaders` were using it. It was only deleted when the `DomainParticipant` was deleted. This problem has been resolved.

[RTI Issue ID LABPLG-503]

4.4.6 Crash when Connecting Reader Output ref num to Writer Input ref num (and Vice Versa)

A crash may have occurred when connecting a Reader's output ref num to a Writer's input ref num and vice versa. This problem has been resolved.

Note: The name `ref num` has been replaced in this release with `DDS Object Ref`.

[RTI Issue ID LABPLG-506]

4.4.7 Error 5058 when Reading in Parallel with Reentrant Read SubVI

Error 5058 may have been thrown when two readers were working in parallel. If the error was thrown, that reader would have been stopped. This only happened when we encapsulated the Read CLF node into a subVI and made that subVI reentrant. This problem has been resolved.

[RTI Issue ID LABPLG-526]

4.4.8 Error Loading RTI DDS Secure Plugins using Standalone Application

An error occurred when using *RTI Secure DDS Plugins* in a standalone application. This problem has been resolved.

[RTI Issue ID LABPLG-526]

5 Known Issues

5.1 Reader/Writer Create SubVIs Fail if QoS Settings not Provided

When creating a new Reader or Writer, QoS settings are optional. However, if there are no QoS settings when working on a cRIO-9068 system, you may encounter error 5052 ("XML Configuration File not found").

A workaround is to add an empty string to the `qos_profile` pin.

[RTI Issue ID LABPLG-240]

5.2 Monitoring Library cannot be Used as DomainParticipant's Base Profile when Creating Custom Secure Profile

When creating a DomainParticipant from a Custom Secure Profile, non-secure Monitoring cannot be enabled for that DomainParticipant. If this situation occurs, the toolkit will throw error 5080, which means that the DomainParticipant cannot be created.

[RTI Issue ID LABPLG-474]

5.3 Possible Crash when using Query Conditions

A crash may occur after a Query Condition initially evaluates to false.

If a condition initially evaluates to false, it works as expected—data is not received. However, if the condition later becomes true again, the condition may not work correctly. And if the condition then goes back to evaluating as false, this may cause a crash.

[RTI Issue ID LABPLG-644]

6 Additional Documentation

RTI DDS Toolkit uses *RTI Connex DDS* for communication. For details on *RTI Connex DDS* and the Quality of Service (QoS) settings, visit <http://community.rti.com/documentation>.