RTI Connext DDS

Core Libraries

Custom Support for NI Linux Real-Time Platforms

Version 5.2.3

Printed in U.S.A. First printing.
April 2016.

Trademarks

Copy and Use Restrictions

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/

Custom Support for NI Linux Real-Time Platforms

2.1 Supported Platforms

This document supplements the Release Notes and Platform Notes. It provides information specifically for the platform listed in Custom Supported NI Linux Platforms .

Custom Supported NI Linux Platforms

Operating System	CPU	Compiler	RTI Architecture Abbreviation
NI Linux Real-Time 3.2[a]	ARMv7	gcc 4.4.1	armv7AngstromLinux3.2gcc4.4.1.cortex-a9

2.2 Transports

These transports are supported:

- Shared Memory: Supported and enabled by default. To clean up shared memory resources, reboot the kernel.
- UDPv4: Supported and enabled by default.

These transports are not supported:

- UDPv6
- TCP/IPv4

2.3 Features

These features are supported:

[a]Requires NI-RIO 13.1 release or a patch from NI for NI-RIO 13.0

- Modern C++ API
- Multicast
- Monotonic clock
- Control of CPU core affinity for RTI threads
- RTI Distributed Logger

2.4 Compiling and Running

Building Instructions lists the compiler flags and libraries you will need to link into your application.

Note: Additional libraries are required when using *RTI Secure WAN Transport* and *RTI TCP Transport*. For more information, see the Linux section of the *RTI Connext DDS Core Libraries Platform Notes*.

Running Instructions shows the environment variables required to be set at run time.

Library-Creation Details provides details on how these custom libraries were built. This table is provided strictly for informational purposes; you do not need to use these parameters to compile your application. You may find this information useful if you are involved in any in-depth debugging.

Building Instructions

API	Library Format	Required RTI Libraries[ab c]	Required System Libraries	Required Compiler Flags
C++ (Traditional and Modern APIs)	Static Release	libnddscppz.a or libnddscpp2z.a libnddscz.a libnddscorez.a librticonnextmsgcz.a	-ldl -lnsl -lm -lpthread -lrt	-DRTI_UNIX -m32
	Static Debug	libnddscppzd.a or libnddscpp2zd.a libnddsczd.a libnddscorezd.a librticonnextmsgcppzd.a		
	Dynamic Release	libnddscpp.so or libnddscpp2.so libnddsc.so libnddscore.so librticonnextmsgcpp.so		
	Dynamic Debug	libnddscppd.so or libnddscppd.so libnddscd.so libnddscored.so librticonnextmsgcppd.so		

[a]The C/C++ libraries are in <NDDSHOME>/lib/<architecture> (where<NDDSHOME> is where Connext DDS is installed, such as /home/your user name/rti_connext_dds-5.x.y).

[b]The *rticonnextmsg* library only applies if you have the Connext DDS Professional, Evaluation, or Basic package type. It is not provided with the Connext DDS Core package type.

[c]Choose libnddscpp*.* for the Traditional C++ API or libnddscpp2*.* for the Modern C++ API.

Building Instructions

API	Library Format	Required RTI Libraries[ab c]	Required System Libraries	Required Compiler Flags
C	Static Release	libnddscz.a libnddscorez.a librticonnextmsgcz.a	-ldl -lnsl -lm -lpthread -lrt	-DRTI_UNIX -m32
	Static Debug	libnddsczd.a libnddscorezd.a librticonnextmsgczd.a		
	Dynamic Release	libnddsc.so libnddscore.so librticonnextmsgc.so		
	Dynamic Debug	libnddscd.so libnddscored.so librticonnextmsgcd.so		

Running Instructions

RTI Architecture	Library Format	Environment Variables
armv7AngstromLinux3.2 gcc4.4.1.cortex-a9	Static	None required
	Dynamic	LD_LIBRARY_PATH= ${NDDSHOME}/lib/<*architecture*>: ${LD_LIBRARY_PATH}[d]

[a]The C/C++ libraries are in <NDDSHOME>/lib/<architecture> (where<NDDSHOME> is where Connext DDS is installed, such as /home/your user name/rti_connext_dds-5.x.y).

[b]The *rticonnextmsg* library only applies if you have the Connext DDS Professional, Evaluation, or Basic package type. It is not provided with the Connext DDS Core package type.

[c]Choose libnddscpp*.* for the Traditional C++ API or libnddscpp2*.* for the Modern C++ API.

[d]${NDDSHOME} represents the root directory of your Connext DDS installation. ${LD_LIBRARY_PATH} represents the value of the LD_LIBRARY_PATH variable prior to changing it to support Connext DDS.

Library-Creation Details

RTI Architecture	Library Format	Compiler Flags Used by RTI
armv7AngstromLinux3.2 gcc4.4.1.cortex-a9	Release	-fpic -DLINUX -O -Wall -Wno-unknown-pragmas -DPtrIntType=long -DCSREAL_IS_FLOAT -DCPU=cortex-a9 -DTARGET=\"armv7AngstromLinux3.2gcc4.4.1.cortex-a9\" -DNDEBUG -c -Wp,-MD
	Debug	-fpic -DLINUX -g -Wall -Wno-unknown-pragmas -DPtrIntType=long -DCSREAL_IS_FLOAT -DCPU=cortex-a9 -DTARGET=\"armv7AngstromLinux3.2gcc4.4.1.cortex-a9\" -c -Wp,-MD

2.5 Workaround for Known Issue with Checksum Calculation

There is a problem in the checksum calculation on National Instruments (NI) Linux (formerly know as Angstrom Linux) version Linux NI-cRIO-9068 3.2.35-rt52-1.0.0f1.
National Instruments will fix this in a later release. A workaround is to patch the IP stack to disable checksum calculation as follows:

1. Create a file named **disableHwEthChecksum** that contains:

```
#!/bin/sh
# init script that disables the hardware checksum on both 9068 ethernet adapters
ethtool -K eth0 tx off
ethtool -K eth1 tx off
```

2. Copy the file to **/etc/init.d/**.
3. Open an SSH connection to the target using Putty or another SSH client.
4. Login as admin.
5. Enter the following commands:

```
:~# chmod +x /etc/init.d/disableHwEthChecksum
:~# update-rc.d disableHwEthChecksum start 99 S . stop 99 S .
```

6. Restart the target. The hardware checksum should be disabled on subsequent boots.