

RTI Connex DDS

Core Libraries

**Custom Support for
Red Hat Enterprise Linux 5.2 Platforms
on x86 CPUs with gcc 4.2.1**

Version 5.3.0



© 2017 Real-Time Innovations, Inc.

All rights reserved.

Printed in U.S.A. First printing.

May 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: support@rti.com

Website: <https://support.rti.com/>

Custom Support for Red Hat Enterprise Linux 5.2 Platforms

1 Supported Platforms

This document supplements the [RTI Connex DDS Core Libraries Release Notes](#) and [RTI Connex DDS Core Libraries Platform Notes](#). It provides information specifically for the platforms in [Table 1 Custom Supported Red Hat Enterprise Linux 5.2 Platforms](#).

Table 1 Custom Supported Red Hat Enterprise Linux 5.2 Platforms

Operating System	CPU	Compiler	RTI Architecture Abbreviation
Red Hat Enterprise Linux 5.2	x86	gcc 4.2.1	i86Linux2.6gcc4.2.1
		Sun Java Platform Standard Edition JDK 1.8	

2 Transports

These transports are supported:

- Shared memory: Supported and enabled by default
- UDPv4: Supported and enabled by default
- UDPv6: Supported. Not enabled by default, and the peers list must be modified to support IPv6. Mapping of the TransportPriority QoS is supported.

These transports are not supported:

- TCP/IPv4
- RTI Secure WAN Transport

3 Features

These features are supported:

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

These features are not supported:

- RTI Distributed Logger

4 Compiling and Running

[Table 2 Building Instructions](#) lists the compiler flags and libraries you will need to link into your application.

[Table 3 Running Instructions](#) shows the environment variables required to be set at run time.

[Table 4 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.

Table 2 Building Instructions

API	Library Format	Required RTI Libraries ^{abc}	Required System Libraries	Required Compiler Flags
C++ (Traditional and Modern APIs)	Static Release	libnddscppz.a or libnddscpp2z.a libnddscz.a libnddscorz.a librticonnextmsgcz.a	-ldl -lnsl -lm -lpthread -lrt -L/usr/lib/nptl	-DRTI_UNIX -m32
	Static Debug	libnddscppzd.a or libnddscpp2zd.a libnddsczd.a libnddscorz.d.a librticonnextmsgcppzd.a		
	Dynamic Release	libnddscpp.so or libnddscpp2.so libnddsc.so libnddscorz.so librticonnextmsgcpp.so		
	Dynamic Debug	libnddscppd.so or libnddscpp2d.so libnddscd.so libnddscord.so librticonnextmsgcppd.so		

^aThe C/C++/Java libraries are in <NDDSHOME>/lib/<architecture>; the jar files are in <NDDSHOME>/lib/java (where <NDDSHOME> is where Connexrt DDS is installed, such as /home/your user name/rti_connexrt_dds-5.x.y).

^bThe *rticonnextmsg* library only applies if you have the **Connexrt DDS** Professional, Evaluation, or Basic package type. It is not provided with the Connexrt DDS Core package type.

^cChoose libnddscpp*.* for the Traditional C++ API or libnddscpp2*.* for the Modern C++ API.

Table 2 Building Instructions

API	Library Format	Required RTI Libraries ^{abc}	Required System Libraries	Required Compiler Flags
C	Static Release	libnddscz.a libnddscorz.a librticonnextmsgcz.a	-ldl -lnsl -lm -lpthread -lrt -L/usr/lib/nptl	-DRTI_UNIX -m32
	Static Debug	libnddsczd.a libnddscorz.d.a librticonnextmsgcz.d.a		
	Dynamic Release	libnddsc.so libnddscorz.so librticonnextmsgc.so		
	Dynamic Debug	libnddscd.so libnddscorz.d.so librticonnextmsgcd.so		
Java	Release	nddsjava.jar rticonnextmsg.jar	N/A	None required
	Debug	nddsjavad.jar rticonnextmsgd.jar		

^aThe C/C++/Java libraries are in <NDDSHOME>/lib/<architecture>; the jar files are in <NDDSHOME>/lib/java (where <NDDSHOME> is where Connex DDS is installed, such as /home/your user name/rti_connex_dds-5.x.y).

^bThe *rticonnextmsg* library only applies if you have the **Connex DDS** Professional, Evaluation, or Basic package type. It is not provided with the Connex DDS Core package type.

^cChoose libnddscpp*. * for the Traditional C++ API or libnddscpp2*. * for the Modern C++ API.

Table 3 Running Instructions

RTI Architecture	Library Format	Environment Variables ^a
i86Linux2.6gcc4.2.1	Static	None required
	Dynamic	LD_LIBRARY_PATH= \${NDDSHOME}/lib/<architecture>:\${LD_LIBRARY_PATH}
i86Linux2.6gcc4.2.1 for Java	N/A	LD_LIBRARY_PATH= \${NDDSHOME}/lib/<architecture>: \${LD_LIBRARY_PATH}

Table 4 Library-Creation Details

RTI Architecture	Library Format	Compiler Flags Used by RTI
i86Linux2.6gcc4.2.1	Release	-fPIC -DLINUX -O -Wall -Wno-unknown-pragmas -DPtrIntType=long -DCSREAL_IS_FLOAT -DCPU=I80586 -DTARGET=\"i86Linux2.6gcc4.2.1\" -fmessage-length=0 -DNDEBUG -c -Wp,-MD
	Debug	-fPIC -DLINUX -g -Wall -Wno-unknown-pragmas -DPtrIntType=long -DCSREAL_IS_FLOAT -DCPU=I80586 -DTARGET=\"i86Linux2.6gcc4.2.1\" -fmessage-length=0 -c -Wp,-MD
i86Linux2.6gcc4.2.1 for Java	Release	-target 1.5 -source 1.5
	Debug	-target 1.5 -source 1.5 -g

^a \${NDDSHOME} is where is installed. \${LD_LIBRARY_PATH} represents the value of the LD_LIBRARY_PATH variable prior to changing it to support . When using nddsjava.jar, the Java virtual machine (JVM) will attempt to load release versions of the native libraries. When using nddsjavad.jar, the JVM will attempt to load debug versions of the native libraries.