

# **RTI Ada Language Support**

**Platform Notes**

**Version 6.1.0**



© 2021 Real-Time Innovations, Inc.  
All rights reserved.  
Printed in U.S.A. First printing.  
April 2021.

## **Trademarks**

RTI, Real-Time Innovations, Connex, NDDS, 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.

This is an independent publication and is neither affiliated with, nor authorized, sponsored, or approved by, Microsoft Corporation.

The security features of this product include software developed by the OpenSSL Project for use in the OpenSSL Toolkit (<http://www.openssl.org/>). This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).

## **Technical Support**

Real-Time Innovations, Inc.

232 E. Java Drive

Sunnyvale, CA 94089

Phone: (408) 990-7444

Email: [support@rti.com](mailto:support@rti.com)

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

# Contents

---

<b>1 Platform Notes</b> .....	<b>1</b>
<b>2 Installation</b> .....	<b>2</b>
<b>3 Supported Platforms</b> .....	<b>3</b>
<b>4 Required Compilers</b>	
4.1 Ada Language Version .....	4
<b>5 Required System Libraries and Connex DDS Core Libraries</b> .....	<b>5</b>
<b>6 Required Compiler and Linker Flags</b> .....	<b>7</b>
<b>7 Required Environment Variables for Building Applications</b> .....	<b>8</b>
<b>8 How the Ada Language Support Libraries were Built</b> .....	<b>9</b>

# 1 Platform Notes

This document provides platform-specific instructions on how to compile, link, and run applications built with *RTI® Ada Language Support* .

*Ada Language Support* is built on top of the *RTI Connex® DDS C API*. This document supplements the *RTI Connex DDS Core Libraries Platform Notes*.

## 2 Installation

As with any target bundle, *Ada Language Support* must be installed after you've installed a host bundle of *Connex DDS*.

Use the `<NDDSHOME>/bin/rtipkginstall` utility (where NDDSHOME is where you installed the host bundle, such as `/home/user/rti_connex_dds-<version>` for non-root users, or `/opt/rti_connex_dds-<version>` for root users). The package installer is also available in *RTI Launcher*. See the instructions in "Installing Connex DDS," in *Before You Get Started*, in the [RTI Connex DDS Getting Started Guide](#) for installing a target bundle.

**Note:** Before you can compile Ada applications, you will need to install a target bundle for the supported architecture (x64Linux3gcc4.8.2). This is a separate package file, not part of *Ada Language Support*.

After installing the Ada support package, you can use *rtiddsgen* to generate Ada projects for your IDL types as follows:

```
rtiddsgen -language Ada -example x64Linux3gcc4.8.2 HelloWorld.idl
```

**Note:** An Ada source package is also available: `rti_ada_language_support-6.1.0-src.tar.gz`.

See the generated README for instructions.

## 3 Supported Platforms

*Ada Language Support* provides the libraries required to build *Connex DDS* Ada applications for the following platforms when using AdaCore GNAT Pro 18.2 (<http://www.adacore.com>):

**Table 3.1 Supported Platforms for Ada Language Support**

Operating System	CPU	Compiler	RTI Architecture Abbreviation
CentOS 7.0	x64	gcc 4.8.2	x64Linux3gcc4.8.2
Red Hat Enterprise Linux 7.0, 7.3, 7.5, 7.6			
Ubuntu 14.04 LTS			

See [4 Required Compilers on page 4](#) for important information regarding the required Adacore GNAT compiler.

Since the Ada API libraries use the *Connex DDS C* API libraries, you should also refer to the basic Linux platform information applicable to the C programming language and system requirements for Red Hat Enterprise Linux 7 described in the *RTI Connex DDS Core Libraries Platform Notes*.

This document assumes that the platform has been set up to run *Connex DDS* for the C programming language and includes only additional information required for using the Ada programming language.

## 4 Required Compilers

The required Ada compilers are GNAT Pro 18.2 and GPRbuild Pro 18.2. The code generated by *RTI Code Generator (rtiddsgen)* for custom types consists of both Ada and C code.

The Ada source package (**rti\_ada\_language\_support-6.1.0-src.tar.gz**) can be compiled with GNAT Pro version 18.2 - 21.1.

### 4.1 Ada Language Version

*Ada Language Support* has been written using the Ada 2012 version of the language. This is the default mode for GNAT Pro 18.2. Since Ada 2012 is almost completely upwards compatible with Ada 2005, programs based on Ada 2005 should compile without issues. Note: You may still build your application source in Ada 2005 compatibility mode (using `-gnat05` or `-gnat2005`).

## 5 Required System Libraries and Connex DDS Core Libraries

Once GNAT is installed on the development machine, you can use the **gprbuild** compiler to build your Ada applications with *Ada Language Support*. The **gprbuild** compiler uses project files to get information about source code directories and files, source code languages, compiler and linker flags, and other dependent libraries and Ada projects. Sample project files are included in *Ada Language Support* to help you build your Ada library or application with various library configurations.

To include *Ada Language Support* in your Ada application, the following *Connex DDS* API Ada project files (in `$NDDSHOME/lib/gnat/`) must be included in your Ada application project file: **dds.gpr** and **dds-ada.gpr**.<sup>1</sup> To do so, use the Ada "**with**" clause as follows:

```
with "dds.gpr";  
with "dds-ada.gpr";
```

These API Ada project files use the `NDDSHOME` environment variable and three additional external project variables: **OS**, **RTIDDS\_LIBRARY\_TYPE**, and **RTIDDS\_BUILD**. These variables can either be set up in your environment, or by using the **-X** option when calling **gprbuild**.

Possible values for these variables are:

- **OS**: Can be **Linux**, **Windows\_NT**, or **unix**. (Default: **unix**)
- **RTIDDS\_LIBRARY\_TYPE**: Either **static** (statically links the *Connex DDS* Core Libraries into the applications's executable or library) or **relocatable** (dynamically links the *Connex DDS* Core Libraries with the application or library). (Default: **static**)

---

<sup>1</sup>Two other project files are provided (`dds-libnndsada.gpr` and `dds-libnndsc.gpr`); they are no longer needed but are provided for backward compatibility.



- **RTIDDS\_BUILD**: Either **release** or **debug**. (The **debug** option will include debugging information). (Default: **release**)

**Note:** All possible values for these variables can be found in the API Ada project file, **\$NDDSHOME/lib/gnat/dds.gpr**.

The **gprbuild** compiler uses an environment variable called **ADA\_PROJECT\_PATH**, to specify Ada project file paths. **\$NDDSHOME/lib/gnat/** must be added to the **ADA\_PROJECT\_PATH** environment variable. These project files use another Ada project file, **dds-core.gpr**, found in the same path, as well as the specification (header) files of the API for Ada found in **\$NDDSHOME/include/ndds/dds\_ada/**. You do not need to add these files/paths to an environment variable since their locations and names are already specified in the provided API Ada project files.

The libraries required by GNAT to compile an Ada application (used on a development machine) and run an Ada application (used on both development and deployment machines) can be found in the GNAT documentation. Follow those instructions to set up a development and a deployment machine.

The *Ada Language Support* libraries are located in **\$NDDSHOME/lib/<architecture>**. Their ALI files are in **\$NDDSHOME/lib/<architecture>.ali**. (Where *<architecture>* is a supported architecture from [3 Supported Platforms on page 3](#), such as x64Linux3gcc4.8.2).

**Table 5.1 Libraries**

Build Type	Library Filenames
Static Release	librtiddsadaz.a libnddscz.a libnddscorez.a
Static Debug	librtiddsadazd.a libnddsczd.a libnddscorezd.a
Dynamic Release	librtiddsada.so libnddsc.so libnddscore.so
Dynamic Debug	librtiddsadad.so libnddscd.so libnddscored.so

If you are using the dynamic version of the library, you must add its paths to the **LD\_LIBRARY\_PATH** environment variable on the deployment machine.

## 6 Required Compiler and Linker Flags

The flags required by the Ada compiler and linker, and its C compiler, are defined in the provided *Connex DDS API* for Ada project file and the provided Ada specification file `$NDDSHOME/include/ndds/dds_ada/dds-linker_options__Linux.ads`.

The compiler and linker flags are already configured in the provided Ada project files (64-bit is assumed in the compiler flag). The **gprbuild** compiler will use the configuration corresponding to the desired build type, indicated by Ada project variables provided either as command-line parameters or environment variables to **gprbuild**. These external variables (**OS**, **RTIDDS\_LIBRARY\_TYPE**, and **RTIDDS\_BUILD**) are described in [5 Required System Libraries and Connex DDS Core Libraries on page 5](#). These variables can be specified using the **-X** switch when calling **gprbuild** to build an Ada application that uses the provided *Connex DDS API* project files for Ada.

For example, if a Linux Ada application is intended to use the dynamic debug version of the *Connex DDS API* for Ada library on a 64-bit single processor architecture, the Ada application project file must include the *Connex DDS API* for Ada library project files as described in [5 Required System Libraries and Connex DDS Core Libraries on page 5](#) and call **gprbuild** in the following way:

```
gprbuild -p -P application.gpr -XOS=Linux -XRTIDDS_BUILD=debug  
-XRTIDDS_LIBRARY_TYPE=relocatable
```

This will instruct **gprbuild** to use the dynamic debug version of the library for *Ada Language Support*, including the necessary compiler and linker flags, on a Linux platform running on a single 64-bit processor architecture.

## 7 Required Environment Variables for Building Applications

The *Connex DDS* API for Ada project files need the **NDDSHOME** environment variable to locate library, specification, and header files when building an Ada application. This variable is not needed to run such applications.

# 8 How the Ada Language Support Libraries were Built

[Table 8.1 Library-Creation Details](#) provides details on how the libraries for *Ada Language Support* 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 8.1 Library-Creation Details**

RTI Architecture	Library Format	Language	Compiler Flags Used by RTI
x64Linux3gcc4.8.2	Static Release	Ada	-gnatf -gnatQ -gnatq -gnat12 -gnatyM2048 -g -fno-strict-aliasing -fPIC -gnatwa -gnato -fcallgraph-info=su -fstack-usage -fstack-check -gnaty3aAbcefhiklnprstx -O2 -ffunction-sections -fdata-sections
		C	-DRTI_UNIX -DRTI_64BIT
	Static Debug	Ada	-gnatf -gnatQ -gnatq -gnat12 -gnatyM2048 -g -fno-strict-aliasing -fPIC -gnatwa -gnato -fcallgraph-info=su -fstack-usage -fstack-check -gnaty3aAbcefhiklnprstx -O0 -ffunction-sections -fdata-sections
		C	-DRTI_UNIX -DRTI_64BIT -g
	Dynamic Release	Ada	-gnatf -gnatQ -gnatq -gnat12 -gnatyM2048 -g -fno-strict-aliasing -fPIC -gnatwa -gnato -fcallgraph-info=su -fstack-usage -fstack-check -gnaty3aAbcefhiklnprstx -O2
		C	-DRTI_UNIX -DRTI_64BIT
	Dynamic Debug	Ada	-gnatf -gnatQ -gnatq -gnat12 -gnatyM2048 -g -fno-strict-aliasing -fPIC -gnatwa -gnato -fcallgraph-info=su -fstack-usage -fstack-check -gnaty3aAbcefhiklnprstx -O0
		C	-DRTI_UNIX -DRTI_64BIT -g