RTI Ada Language Support

Platform Notes

Version 5.2.3

rti
Your systems. Working as one.

Trademarks

Copy and Use Restrictions

Technical Support

Platform Notes

This document provides platform-specific instructions on how to compile, link, and run applications built with *RTI® Ada Language Support* 5.2.3.

Ada Language Support for Connext DDS is built on top of the *RTI Connext™ DDS* C API. This document supplements the *RTI Connext DDS Core Libraries Platform Notes*, version 5.2.3.

1 Installation

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

Use the **<NDDSHOME>/bin/rtipkginstall** utility (where NDDSHOME is where you installed the host bundle, such as **/home/user/rti_connext_dds-<version>** for non-root users, or **/opt/rti_connext_dds-<version>** for root users). The package installer is also available in *RTI Launcher*. See the instructions in the *RTI Connext DDS Core Libraries 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 (x64Linux2.6gcc4.4.5). This is a separate package file, not part of *Ada Language Support for Connext DDS*.

2 Supported Platforms

Ada Language Support provides the libraries required to build *Connext DDS* Ada applications for the following platforms when using AdaCore GNAT Pro 7.3.1 (http://www.adacore.com):

Table 2.1 **Supported Platforms for Ada Language Support**

Operating System	CPU	Compiler	RTI Architecture Abbreviation
CentOS 6.0, 6.2-6.4 (2.6 kernel)	x64	gcc 4.4.5	x64Linux2.6gcc4.4.5
Red Hat® Enterprise Linux® 6.0-6.5 (2.6 kernel)			

See Section 3 for important information regarding the required Adacore GNAT compiler.

Since the Ada API libraries use the *Connext 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 6 described in the *RTI Connext DDS Core Libraries Platform Notes*.

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

3 Required Compilers

The required Ada compilers are GNAT Pro 7.3.1 and GPRbuild Pro 2.2.1. The code generated by *RTI Code Generator (rtiddsgen)* for custom types consists of both Ada and C code. Therefore, you will also need the 64-bit C compiler, **gcc** 4.4.5.

3.1 Ada Language Version

Ada Language Support for Connext DDS has been written using the Ada 2012 version of the language. This is the default mode for GNAT Pro 7.3.1. Since Ada 2012 is almost completely upwards compatible with Ada 2005, programs based on Ada 2005 should compile without issues. However, if you were setting the switch **-gnat05** or **-gnat2005**, you may need to replace it with **-gnat12** or **-gnat2012**, respectively, in order to compile your programs.

3.2 GNAT Pro Compiler Version

For *Ada Language Support* 5.1.0, the required compiler was GNAT Pro 7.2.2. After the release of this 5.1.0, a bug was found in GNAT Pro 7.2.2 (Adacore's reference number for this issue is NA24-002) that caused compilation issues when using DDS Sequences. This bug was fixed in GNAT Pro 7.3.1.

Customers that do not experience issues with sequences and wish to continue using GNAT Pro 7.2.2 can compile RTI Ada Language Support from the source code provided at http://support.rti.com.

4 Required System Libraries and Connext 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 for Connext DDS*. 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 for Connext DDS* to help you build your Ada library or application with various library configurations.

To include *Ada Language Support for Connext DDS* in your Ada application, the following *Connext DDS* API Ada project files (in **$NDDSHOME/lib/gnat/**) must be included in your Ada application project file: **dds.gpr** and **dds-ada.gpr**.[1] 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:

---

1. Two other project files are provided (dds-libnddsada.gpr and dds-libnddsc.gpr); they are no longer needed but are provided for backward compatibility.

❏ **OS**: Can be **Linux**, **Windows_NT**, or **unix**. (Default: **unix**)

❏ **RTIDDS_LIBRARY_TYPE**: Either **static** (statically links the *Connext DDS* Core Libraries into the applications's executable or library) or **relocatable** (dynamically links the *Connext DDS* Core Libraries with the application or library). (Default: **static**)

❏ **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 for Connext DDS* libraries are located in **$NDDSHOME/lib/<architecture>**. Their ALI files are in **$NDDSHOME/lib/<architecture>/ali.** (Where *<architecture>* is a supported architecture from Section 2, such as x64Linux2.6gcc4.4.5).

Table 4.1 **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.

5   Required Compiler and Linker Flags

The flags required by the Ada compiler and linker, and its C compiler, are defined in the provided *Connext DDS* API for Ada project file and the provide 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 Section 4. These variables can be specified using the **-X** switch when calling **gprbuild** to build an Ada application that uses the provided *Connext DDS* API project files for Ada. For example, if a Linux Ada application is intended to use the dynamic debug version of the *Connext DDS* API for Ada library on a 64-bit single processor architecture, the Ada application project file must include the *Connext DDS* API for Ada library project files as described in Section 4 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 for Connext DDS*, including the necessary compiler and linker flags, on a Linux platform running on a single 64-bit processor architecture.

6   Required Environment Variables for Building Applications

The *Connext 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.

7   How the Ada Language Support Libraries were Built

The following table provides details on how the libraries for *Ada Language Support for Connext DDS* 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 7.1 **Library-Creation Details**

RTI Architecture	Library Format	Language	Compiler Flags Used by RTI
x64Linux2.6gcc4.4.5	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