# Contents

## 1 Introduction

## 2 Using System Designer

2.1 Starting System Designer .............................................. 3
2.2 Create a New Project ............................................... 5
2.3 Workflow .......................................................... 6
2.4 Exploring the Tabs ................................................... 6
  2.4.1 Working with Types ........................................... 6
  2.4.2 Configuring Quality of Service (QoS) ...................... 11
  2.4.3 Handling Domains ............................................. 18
  2.4.4 Define Participants and Entities ............................ 21
  2.4.5 Using Fully Qualified Names when Referring to Objects .. 24
  2.4.6 Configuring Security Plugins ............................... 25
  2.4.7 EZ-Secure ..................................................... 27
2.5 Toolbar ............................................................... 30
2.6 Project Management ................................................. 31
  2.6.1 Create a New Project ........................................ 31
  2.6.2 Open an Existing Project ................................... 32
  2.6.3 Open a Recent Project ....................................... 33
  2.6.4 Import a Project from a Zip File ......................... 35
  2.6.5 Export a Project to a Zip File ............................ 37
  2.6.6 Remove a Project from the List of Recent Projects ..... 38
  2.6.7 Validate Project .............................................. 39
  2.6.8 Project Properties ............................................ 41
2.7 External XML Files ................................................ 46
  2.7.1 Creating a New File ........................................ 47
  2.7.2 Open an Existing File ...................................... 48
  2.7.3 Import a File ................................................ 48
2.8 XML Validation ....................................................... 50
2.9 Online Help ........................................................ 50
2.10 Info Dialog .......................................................... 51

## 3 Tutorials

3.1 Defining Your Data Type .......................................... 52
3.2 Defining an XML Application Creation Configuration File .... 54

## 4 Release Notes

61
4.1 Supported Platforms and Browsers .................................. 61
4.2 Compatibility ............................................................ 62
4.3 What's New in 6.1.0 ....................................................... 62
  4.3.1 Improved XML and IDL view with Markdown .................... 62
  4.3.2 New validation button to check XML files with Code Generator 62
  4.3.3 Ability to view content of a specific IDL file .................. 63
  4.3.4 Valuetypes enabled by advanced project properties ............ 63
  4.3.5 Support for use of environment variables as domain ID ........ 63
  4.3.6 Delete project files when removing a project .................. 63
  4.3.7 Create a domain without a domain ID ............................. 63
  4.3.8 “Add” buttons merged into a single, context-aware Add button 63
  4.3.9 Updated “Configure Security” dialog with new security properties 63
4.4 What's Fixed in 6.1.0 .................................................... 65
  4.4.1 Existing comments in XML/IDL text area not updated when modified 65
  4.4.2 Invalid content in IDL view when using aliases ................. 66
  4.4.3 Crash when trying to create a type inside a module .......... 66
  4.4.4 Uint64 member produced invalid IDLs ............................. 66
  4.4.5 Use of constants in @min, @max, and @default fields .......... 66
  4.4.6 Version control issues ............................................. 66
  4.4.7 Inheritance Chain edit icon was enabled when the QoS Profile was a QoS Snippet 67
  4.4.8 Unbounded sequences of strings converted to strings .......... 67
  4.4.9 System Designer failed to delete QoS Library, Domain Library, and Participant Library 67
  4.4.10 Union Editor did not allow addition of cases after adding default case 67
  4.4.11 System Designer didn’t allow creating a constant of an enumerator .... 67
  4.4.12 List of XML files incorrectly updated when importing invalid XML file 67
  4.4.13 List of XML files in XML view not updated when creating new XML file 68
  4.4.14 Browse button in Ez-Secure dialog didn’t work ............... 68
  4.4.15 Invalid (HTTP) 400 Bad Request when checking if a file exists 68
  4.4.16 Updated security property names ................................ 68
  4.4.17 System Designer crashed when attempting to load XML file that contained a forward_dcl ....................... 69
  4.4.18 @key fields couldn’t be set in derived structures ............ 69
  4.4.19 System Designer failed to import a zip project in Windows .... 70
4.5 Known Issues ............................................................. 71

5 Copyrights ................................................................. 72
Chapter 1

Introduction

Note: This document assumes you have a basic understanding of RTI® Connext® DDS application development and concepts, such as a DDS Domains, DomainParticipants, Topics, DataWriters, and DataReaders. For an overview of these concepts, please see the RTI Connext DDS Getting Started Guide.

RTI System Designer is based on RTI XML-Based Application Creation - a mechanism to simplify the development of Connext DDS applications. For an overview of its functionality, please see the RTI Connext DDS Core Libraries XML-Based Application Creation Getting Started Guide.

System Designer is a tool that allows you to graphically design and configure all the aspects of a Connext DDS distributed system. Think of it as a user interface (UI) to XML-Based Application Creation, a technology that allows you to specify in XML format all the aspects of a DDS system:

- The Data Types that will be used to communicate information in the system
- The Topics that will be used in the domain, associating each Topic with a Data Type
- The DomainParticipant that can potentially be used, giving each a participant-name
- The DataWriters and DataReaders within each DomainParticipant, each associated with its corresponding Topic
- The Qualities of Service (QoS) that all the entities can use

Your application code simply indicates the participant configuration name of the DomainParticipant that it wants to create. The XML-Based Application Creation infrastructure takes care of the rest: creating the DomainParticipant, registering the types and Topics, and populating all the configured Entities.

Writing this XML can be quite challenging. Using a good editor with autocomplete capabilities along with the provided XML Schema can be of great help. But it is still quite complicated, especially as the project gets bigger and involves multiple files. The System Designer UI tool can help ease this process and manage the complexity of a large system.

This Getting Started Guide is not a manual for System Designer and it does not cover all the functionality. The tool has a lot of inline help. Look for the symbol for more help while using the tool.
Tip: You can access a live version of System Designer without installing it by visiting: https://community.rti.com/downloads/experimental/system-designer

This URL allows you to test a demo version of System Designer and learn more before you download and install it.
Chapter 2

Using System Designer

2.1 Starting System Designer

You can run System Designer from Launcher or from the distributed command-line executable.

In Launcher, go to the Tools tab and click the System Designer button:

You can also start System Designer with the rtisystemdesigner[.bat] script in the bin directory of
your Connext DDS installation. The System Designer script will start the tool, try to open your default browser, and point to the right address.

If the browser won’t start, open a new webpage and go to:

http://localhost:5000/index.html

**Tip:** By default, System Designer uses port 5000. If that port is not available, you can change it by setting RTI_SD_PORT to be a valid port for your system.

**Note:** System Designer automatically shuts down the server if there hasn’t been a connection in the last five minutes. This is important when running System Designer from Launcher, since clicking the System Designer button again won’t create a new server instance until the five-minute period has passed. (Clicking the System Designer button again while the previous instance is running will just open your default browser.) To open a new System Designer instance from Launcher, wait at least five minutes or kill the running instance.

When you start System Designer, the application will allow you to:

- **Open…** Browse the filesystem for an existing System Designer project (.rtisdproj)
- **Import Zip…** Browse the filesystem to import a System Designer project that was previously exported and sent to you (.zip)
- **Create New…** Create a new project
- **Open Selected…** Open one of the listed Recent Projects (the list is empty right after the first execution)
2.2 Create a New Project

After you select Create New..., you will see this dialog:

![New Project Dialog](image)

You can choose a name and select the Project Path by clicking...

Press OK, and System Designer will create a new project and present you with the main UI where you can start designing your system.

When you create a new project, System Designer creates a project (.rtisdproj) file and if you checked the option when creating the project, an empty XML file. You can use System Designer to add components to the file.

You can work on your project wherever you want. System Designer finds default profiles and files in your Connext DDS installation for you.
2.3 Workflow

The System Designer main UI offers four tabs, which represent the tasks in XML-Based Application Creation:

- Types Definition
- QoS Specification
- Domain Definition
- Participant Definition

We suggest you start by defining a Type and proceed with the definition of the QoS. Then move to the Domain tab, register your types, and declare your Topic. Finally, use the Participant tab to define your participant, DataReaders, and DataWriters.

Of course, you can always (and probably will) go back and forth among these tabs to tweak your system.

System Designer automatically saves your projects and changes. (You can disable Autosave from the Projects menu: select Properties.)

2.4 Exploring the Tabs

2.4.1 Working with Types

To introduce new types in your project, right-click on the Types node in the Tree View and add one of the top-level types:
For more information on data types, see Introduction to the Type System, in the RTI Connext DDS Core Libraries User's Manual.

A pop-up will guide you on how to create a new top-level type. In this example, we are adding a new Structure:
The pop-up shown above will guide you in creating a new type. You can choose the Name and, optionally, the Base Type. You can also select one of the annotations presented on the screen.

By default, the new object will be added to the last XML file used, but you can use the drop-down menu to select any XML file or use the New… button to add new XML files to your System Designer project.

Once you have added the top level, go to the Structured view and populate the new type using the button:

A pop-up will help you select the types and the right values:
The **Types** tab allows you to explore the types in three ways:

- Structured view
- XML view
- IDL view

**Tip:** The RTI Connext DDS Core Libraries Extensible Types Guide has a mapping of type names in IDL 4.2. (For example, you may want to know that a `long` is called `int32` in IDL 4.2).

The Structured view is the simplest way to explore and modify the types in your system. The **button** allows you to go as deep as needed when exploring types.

- In the **Item** column, you will see the name and type of a specific element of your type.
- In the **Annotations** column, you can see things like the ID of a member; you can see if the member is a key, or if it is an optional or extensible member.
- The **Location** column will indicate in which file the member is defined, in case you have a project with multiple XML files.
- The **Edit** column contains buttons to alter the selected item. Container objects such as structs, unions, and modules have a **button** to add a child to them.
You can use \[\text{\small \LaTeX} \] to rearrange all the objects.

Currently you cannot move elements in or out of a container (e.g., moving a struct out of a module is not currently supported). To do that, you can always switch to the XML view and modify the XML manually.

The XML view shows the types in XML format. You can highlight a specific member by selecting it in the tree on the left panel:

You can also click on \[\text{\small \LaTeX} \] to edit your XML file manually. All the changes will be reflected in the Structured view.

Finally, the IDL tab shows you the IDL definition of the type:

You can download the IDL type using the \[\text{\small \LaTeX} \] button. (After you click this button, a dialog box opens; click the “Download” link at the bottom.)
2.4.2 Configuring Quality of Service (QoS)

The QoS tab allows you to add Qualities of Services and organize them in QoS Libraries and QoS Profiles.

To add new QoS to your model, click in the QoS tab or right-click **QoS Libraries** in the tree:

To add a QoS Library, enter the Library Name and the destination file:

You can add QoS Profiles to that library by using in the main panel while a QoS Library is selected or by right-clicking on the QoS Library:
A pop-up window will guide you to select the profile name, the destination file, the profile you want to inherit from, and other attributes as shown in the screenshot below:
Some profiles cannot be defined as Snippets because they inherit from a base profile.

For more information on Entities and Profiles, see these sections of the RTI Connext DDS Core Libraries User’s Manual:

- QoS Profile Inheritance and Composition
- DDS Entities
- Fundamentals of DDS Domains and DomainParticipants
- DomainParticipantFactory

Once you have your profiles, you can select one of them and add QoS for a specific Entity:

![Diagram of QoS Profile NewQosProfile1](image)

When an Entity is selected, you can set all the QoS related to that Entity in the Structured view:
See QoS Policies in the RTI Connext DDS Core Libraries User’s Manual for more information on the QoS available for each entity.

You can browse all of the QoS policies and select the button to set and get more information on a specific setting within a policy:
If you don’t create a QoS Profile or policy for an Entity, the default QoS is used for that Entity. A short explanation is shown, and a link to [More…] information for that specific policy is provided.

For each QoS policy, suggestions and checks are provided:

Once the QoS is set, it will be highlighted in green to indicate that it is being set locally:

If your profile is inheriting from another profile, System Designer will highlight the values that are inherited from the base profile in blue. To inherit the values from another profile click on next to the Base QoS(s) field, and then select the QoS profile where you want to inherit from:

2.4. Exploring the Tabs
You can see where a value is coming from by selecting next to the QoS value:

A summary will be presented to you displaying all the inheritance:

2.4. Exploring the Tabs
The XML view allows you to check the XML results and manually edit each XML file:

**Note:** If you later edit a profile name that’s already being used in an entity (such as in a `<datawriter_qos>`), the entity does not get updated with your change. You will get a warning when this occurs:

**Warning**

The following QoS Profiles are referenced but cannot be resolved:

- NewQosLibrary1::NewQosProfile1 referenced in profile "null"(publisher_qos)

You can select and edit the Entity QoS in the Structured view to use the new name, or edit the XML file directly to use the new name. **System Designer** allows you to change an already used name to give you flexibility in trying out your model. For instance, you may want to use your newly named profile in different entities. Entities that use the old (now unrecognized) name will revert to the default QoS.
Note: About using default QoS: If you create an Entity QoS but don’t modify its defaults, you won’t see it the next time you open the project. Entities with no changes to the default QoS use the default settings and won’t be displayed.

2.4.3 Handling Domains

In the Domain tab, you can add all your domains and Topics. You can organize them into Domain Libraries.

To start a new Domain Library, use the button while Domain Libraries is selected or right-click Domain Libraries in the tree:

To add a Domain Library, enter the Library Name and the destination file:
Then you can add a domain to the new library by using the button while a Domain Library is selected or by right-clicking the Domain Library:

A pop-up window will guide you to enter the Domain Name, the Domain ID, and the destination file:

Once the domain is created, you can register types with the domain and add a new Topic in the Structured view.

To register a new type, select in the Registered types table; a pop-up will allow you to choose a Type name, Type Reference, and Type Kind from a list:
After you click **OK**, the new registered type will appear in the left tree and in the Registered types table:

To register a new Topic, click **Add** in the Topics table. A pop-up will open:

Enter your topic name and select one of the available Registered type names from the drop-down menu. After you click **OK**, the new topic will appear in the left tree and in the main panel:

### 2.4. Exploring the Tabs
Of course, you can also see the XML definition in the XML tab:

2.4.4 Define Participants and Entities

The Participant tab allows you to add Participants, DataReaders, and DataWriters. You have to organize them into Participant Libraries.

To add a new Participant Library to your model, click the button while Participant Libraries is selected or right-click Participant Libraries in the tree and select Add Library:

To add a Participant Library, enter the Library Name and the destination file:
You can then add a Participant to the newly created Library by using the button or by right-clicking on the Participant Library in the tree and selecting **Add Participant**:

A dialog will guide you to pick a name and select one of the existing domains as the Domain Reference. You can also view and edit the QoS and select a file where the new participant will be stored:

Now you can configure Publications and Subscriptions in the structured view:

---

2.4. Exploring the Tabs
To add a new Publisher, select the + Add button in the Publishers table. You can then select Name, Multiplicity, and QoS Policies in the presented dialog:

Once you click OK, the new Publisher will be available in the Structured view and in the tree on the left:

Now you can add DataWriters to the selected Publisher. Use the + Add button and in the pop-up window, select the topic reference, the name of the DataWriter, Multiplicity, and QoS:

2.4. Exploring the Tabs
Follow the same logic to add Subscribers and DataReaders in the Subscriptions tab.

As always, you can see and modify the XML file directly in the XML view:

2.4.5 Using Fully Qualified Names when Referring to Objects

When referring to another type, you can either use the target data type without a namespace (in this case, the search will occur in the same module you are working on), or you must specify the full namespace of the target object. In particular, partial namespace qualification is not supported because it is naturally ambiguous.

For example, suppose you have this type (using IDL syntax):

```idl
module MyModule {
    module AnotherModule {
        struct BaseType {
            short x;
            short y;
        }
    }
}
```

Then suppose you want to create another structure inside `AnotherModule` that extends `BaseType`. You have two choices:

- Use just the name `BaseType` to specify the base class
- Use the fully qualified name `MyModule::AnotherModule::BaseType`
System Designer does not accept partial qualification, such as `AnotherModule::BaseType`, because it is ambiguous.

### 2.4.6 Configuring Security Plugins

The QoS tab also allows you to configure the Security Plugins of a DomainParticipant's QoS.

Click the `Configure Security...` button in a DomainParticipant's QoS.

A new dialog appears that will guide you through configuring the Security Plugins on the selected DomainParticipant's QoSs.

First, you will be asked to add a new configuration. Click the `Add...` button and add the name your security
configuration is going to use.

Once the configuration has been added, start configuring the security properties for your newly created configuration.

Apart from allowing you to change specific security properties associated with each configuration, System Designer can also be used to configure security properties that are not bound to a specific configuration. These properties can be configured by clicking on $\textit{Globals...}$ . The following dialog will open, allowing you to configure global properties such as Authentication and Access Control.

Tip: The dialogs shown above can be used to check whether the security configuration is valid. To validate the configuration, click $\text{Validate}$.
For more information on Security Plugins, see the RTI Security Plugins User’s Manual.

2.4.7 EZ-Secure

EZ-Secure is an experimental feature that can be used to automatically configure the Security Plugins settings. By default, it is disabled but can be enabled by selecting the Show EZ-Secure option in the Advanced Properties.
Once enabled, EZ-Secure will be shown in the Projects menu:
A dialog will guide you through configuring security and generating the certificates:
2.5 Toolbar

*System Designer’s* toolbar:

- Allows you to manage your projects
- Allows you to handle existing XML files in your system
- Shows the name of the current project and last used XML file
- Has a box to add a comment to the selected XML object
- Shows general information and warnings via the Info button
- Has a Help menu
2.6 Project Management

The Projects menu in the toolbar helps you with all operations related to project management. With this menu you can perform the actions described in the following sections.

2.6.1 Create a New Project

To create a brand new project, select “Create New…” from the Projects Menu:

A pop-up will be presented where you can choose the name of the project and select where the project will be saved on your filesystem. You can select if you want to create an XML file now or load/create one later.

You can also choose to autosave the project, see information like the time and date of the last time the project was saved, and view the System Designer version and the Connect DDS version you are creating a file for.
2.6.2 Open an Existing Project

To open an existing project, select the **Open...** option in the Project menu:
A file browser will be presented. You can use it to find your .rtisdproj file.

### 2.6.3 Open a Recent Project

To open a recent project, select **Open Recent** from the Projects Menu:
You will see a list of recently opened projects. Select one and click **Open Selected**:
2.6.4 Import a Project from a Zip File

To import a project from a Zip file, select **Import Zip…** from the Projects menu:

Then use the file browser to find the Zip file:
Once you find it, select and open the file. A new dialog will ask you to select a destination directory in which to unzip the file and save the imported project:

Press OK to import and open the project in System Designer.
2.6.5 Export a Project to a Zip File

To export a project to a Zip file, select Export Zip… from the Projects menu:

A Zip file will be downloaded in your default download directory. The Zip file will contain the project file and all the XML files used in that project.

Note: Some browsers (such as Safari) automatically unzip the folder. Disable this behavior or zip the directory again for sharing.
2.6.6 Remove a Project from the List of Recent Projects

To remove a project from the list of recent projects, select **Remove…** from the Projects menu:

![Projects menu with Remove... highlighted]

Then select the project you want to remove from the list and select **Delete**:
2.6.7 Validate Project

Two validation tests are currently available:

- **Validate XSD** This test uses the tool `xmllint`. The tool must already be installed; it is available for the major operating systems here: <http://xmlsoft.org/>.

- **Check for unbounded strings/sequences** This test looks in the current project to see if there are any data types that are being used (in Topics used by DataReaders or DataWriters) that use unbounded sequences or strings. If so, this test prints a warning reminding you of the limitations (and potential problems) of using unbounded sequences or strings. In future releases, *System Designer* will also analyze each Entity’s QoS to ensure the required resources are correctly declared.

To validate a project, select **Validate**… from the Projects Menu.
This will open a Validate dialog. You can run some or all of the available tests:
2.6.8 Project Properties

To see the properties of the current project, select Properties from the Projects menu:
This will show a window with four tabs.

- The **General** tab shows the name of the project, where it is saved, if autosave is enabled or not, the last time the project was saved, and the **System Designer** and **Connext DDS** version numbers:
• The **Notes** tab allows you to add a free-form comment for the current project:
• The Advanced tab allows you to fine-tune some of System Designer's behavior.
Click for more details on each setting.

- The **Open Files** tab shows which files are in use:
2.7 External XML Files

The **Files** menu in the toolbar helps you with all operations relative to file management in the currently open project:

With this menu you can:

- Create a new XML file and add it to the project
- Open an existing XML file and add it to the project
- Import an existing XML file

**Note:** Opening a file means that *System Designer* will be working (and modifying) that file directly. Importing a file helps you make a copy of that file. The original file will not be modified by *System Designer.*
2.7.1 Creating a New File

To create a new file and add it to the current project, select New… from the Files menu:

This will open a file browser to select a name and destination for the new file:
2.7.2 Open an Existing File

To open an existing file and add it to the current project, select **Open**… from the File menu:

This will open a simple file browser to search for an XML file to add to the current project.

2.7.3 Import a File

To import a file and add it to the current project, select **Import**… from the Files menu:

This will open two file browsers. The first one looks for an XML file to import:
Once a file is selected, a second window lets you copy the content of the file into an existing XML file or create a new XML file with it:
2.8 XML Validation

*System Designer* has an option to validate any XML against *Code Generator*. For example, you can see if the data type that you are defining is valid or not.

To validate the XML, select the XML view, select the XML file from the list of XML files used by the current project, then click the **CodeGen** button.

*CodeGen* will generate code with the file that is selected in the XML File drop-down menu. The result of this operation will be shown once *CodeGen* finishes parsing the input file.

2.9 Online Help

When using the tool, you will often see a help icon: ![Help Icon]. Click the icon to see online tips and understand what to do next.
2.10 Info Dialog

This dialog can be accessed using the Info button on the far right of the toolbar:

The Orange badge indicates the number of unread messages; it is reset when the button is clicked.

Once clicked, you will see any Info Messages:

```
Output:
Warning: NDDSHOME is undefined; XSD validation may not work.
Warning: xsd schema not found, Try setting NDDSHOME. XSD Validation will not work!
Warning: builtin xml directory not found, Try setting NDDSHOME.
```

Close
Chapter 3

Tutorials

This chapter describes several examples, all of which require a Project. You can create your own Project by following Section 2.2.

3.1 Defining Your Data Type

This example shows how to define your own data type using System Designer. In this example, we are going to create the ShapeTypeExtended data type used by the Shapes Demo:

```csharp
enum ShapeFillKind {
    SOLID_FILL,
    TRANSPARENT_FILL,
    HORIZONTAL_HATCH_FILL,
    VERTICAL_HATCH_FILL
};

struct ShapeType {
    string<128> color; //@key
    long x;
    long y;
    long shapesize;
};//@Extensibility EXTENSIBLE_EXTENSIBILITY

struct ShapeTypeExtended : ShapeType {
    ShapeFillKind fillKind;
    float angle;
};//@Extensibility EXTENSIBLE_EXTENSIBILITY
```

1. Create a new project (see Section 2.2) and name it ShapesDemo.
2. Go to the Types View. By default it should be the first view shown by System Designer.

3. Add an enumeration by right-clicking on Types and selecting Add Enumeration… Name your enumeration ‘ShapeFillKind’ and click OK.

4. Right-click on the recently created enum and select Add Member… to start adding each one of the following four enumerator value names: SOLID_FILL, TRANSPARENT_FILL, HORIZONTAL_HATCH_FILL, and VERTICAL_HATCH_FILL. You can leave the Value blank and @default_literal unchecked.

5. Create a struct and name it ‘ShapeType’ by right-clicking on Types and selecting Add Struct… Se-
lect APPENDABLE as its @extensibility value and click OK. Fill the struct, adding each member by right-clicking on the struct and selecting Add Member…:

- Add a bounded (128) string named ‘color’. This element will be the key of our type, so select the @key checkbox. (You can accept the other defaults in this window.)

- Add three longs named ‘x’, ‘y’ and ‘shapesize’. Although ‘long’ is not in the list of suggestions for the Type field, you can type it in. Or use the newer int32 instead of long. See Working with Types.

6. Create a second struct and name it ‘ShapeTypeExtended’. Choose ShapeType as its base type. Add its elements by right-clicking on the struct and selecting Add Member…:

- Add a ShapeFillKind enumerator named ‘fillKind’.
- Add a float named ‘angle’.

When the type definition is complete, you can download the IDL by clicking on the Download IDL button in the IDL view.

Now you have an IDL file to use to create your own Shapes Demo application.

### 3.2 Defining an XML Application Creation Configuration File

This example shows how to define an XML configuration file using System Designer. In this example, we are going to create the same configuration file used by the hello_world_xml_dynamic example.

For more information about XML-Based Application Creation, see the RTI Connext DDS Core Libraries XML-Based Application Creation Getting Started Guide.

1. Create a new XML file inside the open Project by clicking the Files icon at the top and selecting New…

You can use the same project as you used in the previous tutorial, or a new project. Name the file USER_QOS_PROFILES.xml. Click OK.

**Note:** If you see a default XML file already associated with the project, that’s because you kept the default selection, “Automatically create an empty XML file” when you created the project. You can still create USER_QOS_PROFILES.xml, it will not replace the default one.

2. Go to the Types View. By default it should be the first view shown by System Designer.
3. Add two constants (right-click Types > Add Constant…) as follows:
   - Name ‘MAX_NAME_LEN’, Type ‘long’, value 64, File USER_QOS_PROFILES.xml
   - Name ‘MAX_MSG_LEN’, Type ‘long’, value 128, File USER_QOS_PROFILES.xml

![Add Constant](image)

4. Create a struct (right-click Types > Add Struct…) and name it ‘HelloWorld.’ Fill the struct by adding each of the following members (right-click on the struct and select Add Member…):
   - Add a bounded string named ‘sender’ with a maximum length of MAX_MSG_LEN. (For this and the following members, you can accept the defaults in the window.)
   - Add a second bounded string named ‘message’ with a maximum length of MAX_NAME_LEN.
• Add a long named ‘count’.

5. Click the XML view to view how your data type definition looks in XML:

6. Define the QoS that the application is going to use by clicking on the QoS view. Since we are creating the same XML file as the hello_world_xml_dynamic example has, to match the content of that file you only need to add a new QoS Library named ‘qosLibrary’ and a QoS Profile named ‘DefaultProfile’ inside the created libray. (You do not have to select any of the checkboxes.)

In the XML view, your QoS Profile should look like this:

3.2. Defining an XML Application Creation Configuration File 56
7. Now that the QoS has been defined, define the system’s Topic and its corresponding data types, first by defining the Domain:

- In the Domain view, right-click on Domain Libraries and select **Add Library**. Name your domain ‘MyDomainLibrary’ and click OK.

  **Note:** If your project contains more than one XML file, select the USER_QOS_PROFILES.xml file.

- Add a domain to your library by right-clicking MyDomainLibrary and selecting **Add Domain**. Name it ‘HelloWorldDomain,’ entering 0 as its Domain Id.

- Select HelloWorldDomain and click the **Add** button to add a registered type. Name the type ‘HelloWorldType,’ selecting the ‘HelloWorld’ struct that we created previously as its type reference.
• In the Topics panel, click the **Add** button to add a topic named ‘HelloWorldTopic,’ selecting the ‘HelloWorldType’ registered type as its registered type reference.

![Topic Properties](image)

In the XML view, your domain configuration should look like this:

```xml
<domain_library name="MyDomainLibrary">
  <domain name="HelloWorldDomain" domain_id="0">
    <register name="HelloWorldDomain" type_ref="HelloWorld"/>
    <topic name="HelloWorldTopic" type_ref="HelloWorldType"/>
  </domain>
</domain_library>
```

8. Define the elements that are going to be part of our system by clicking on the Participant view. In this example, we are going to create a simple application containing one DomainParticipant, one Publisher, one Subscriber, one DataReader, and one DataWriter.

• Add a new Participant Library, naming it ‘MyParticipantLibrary.’ Add a Participant inside that library named ‘PublicationParticipant,’ selecting the previously created ‘HelloWorldDomain’ as its domain reference.

**Note:** If your project contains more than one XML file, select the USER_QOS_PROFILES.xml file.
• Click the **Add** button to add a publisher named ‘MyPublisher.’

• Click the **Add** button to add a DataWriter named ‘HelloWorldWriter,’ selecting ‘HelloWorldTopic’ as its topic reference.

• Add a second Participant inside that library named ‘SubscriptionParticipant,’ selecting the previously created ‘HelloWorldDomain’ as its domain reference.

• Click Subscriptions and add a subscriber named ‘MySubscriber.’

• Add a DataReader named ‘HelloWorldReader,’ selecting ‘HelloWorldTopic’ as its topic reference. Unlike for the DataWriter, for the DataReader we are going to select the QoS profile that we created earlier, named DefaultProfile (in the Default QoS): click on View/Edit inside the DataReader dialog.
and name the QoS Profile ‘HelloWorld_reader_qos’. Select ‘qosLibrary::DefaultProfile’ as its Base QoS.

9. Your XML view should look like this:

```
<domain_participant_library name="MyParticipantLibrary">
  <domain_participant name="PublicationParticipant" domain_ref="MyDomainLibrary::HelloWorldDomain">
    <publisher name="MyPublisher">
      <data_writer name="HelloWorldWriter" topic_ref="HelloWorldTopic"/>
    </publisher>
  </domain_participant>
  <domain_participant name="SubscriptionParticipant" domain_ref="MyDomainLibrary::HelloWorldDomain">
    <subscriber name="MySubscriber">
      <data_reader qos_name="HelloWorld_reader_qos" base_name="qosLibrary::DefaultProfile"/>
    </subscriber>
  </domain_participant>
</domain_participant_library>
```

You can use this XML file to define your Connext DDS system using XML-Based Application Creation.
Chapter 4

Release Notes

4.1 Supported Platforms and Browsers

*System Designer* 6.1.0 is compatible with *Connext DDS* 6.1.0.

*System Designer* is available for the following platforms:

- **Linux® platforms**: All Intel 64-bit Linux platforms listed in the *RTI Connext DDS Core Libraries Release Notes* with the same version number. Tested on Ubuntu® 18.04 LTS only, with Chrome® version 77, Firefox® version 69 and Safari® version 12.

- **macOS® platforms**: All macOS platforms listed in the *RTI Connext DDS Core Libraries Release Notes* with the same version number. Tested on macOS 10.14 only, with Chrome version 77, Firefox version 69 and Safari version 12.

- **Windows® platforms**: All 64-bit Windows platforms listed in the *RTI Connext DDS Core Libraries Release Notes* with the same version number. Tested on Windows 10 only, with Chrome version 77, Firefox version 69 and Safari version 12.

*System Designer* is not supported on POSIX-compliant architectures that end with “FACE_GP” or any custom-supported platforms.

*System Designer* works with these browsers:

- Chrome version 77
- Firefox version 69
- Safari version 12
4.2 Compatibility

For backward compatibility issues between System Designer 6.1.0 and previous releases, see the Migration Guide on the RTI Community Portal (https://community.rti.com/documentation).

4.3 What’s New in 6.1.0

System Designer 6.1.0 is the first official release of System Designer.

If you had access to previous experimental releases, this release makes the product more robust and adds new functionality.

System Designer 6.1.0 is built on Connext DDS 6.1.0. The previous, experimental release of System Designer (version 0.4.0) was built on Connext DDS 6.0.1.

4.3.1 Improved XML and IDL view with Markdown

The text area of the XML and IDL view has been improved with Markdown. For example, tags and attributes are now highlighted in XML.

4.3.2 New validation button to check XML files with Code Generator

A new validation button enables you to select an XML file and have it validated by RTI Code Generator to confirm that the current state of the file can be used to generate code. This validation also provides the output from Code Generator if there are errors.
4.3.3 Ability to view content of a specific IDL file

Now you can view the content of a specific IDL file. To do so, select the Types tab, then the IDL tab. Choose a file from the list of IDL files in the drop-down menu. This is similar to the behavior of the XML tab.

4.3.4 Valuetypes enabled by advanced project properties

Valuetypes are deprecated and therefore can only be activated via a project setting in the project properties file. To use valuetypes in System Designer, navigate to Project -> Properties -> Advanced -> and check the Enable Valuetypes check box.

4.3.5 Support for use of environment variables as domain ID

System Designer now allows setting the domain ID as an environment variable $(ENV_VAR_DOMAIN_ID).

4.3.6 Delete project files when removing a project

Now when you remove a project from the System Designer workspace, you can also choose to delete the project files from the filesystem.

4.3.7 Create a domain without a domain ID

It is no longer a requirement to set a domain ID when creating a domain.

4.3.8 “Add” buttons merged into a single, context-aware Add button

The Add buttons in the QoS, Domain, and Participant panels have been merged into a single context-aware Add button. This new button contains the same functionality as the previous two add buttons in each panel. Previously, to create a QoS you had to click first on +Library to create a library and then +Profile to create a profile inside the library. Now, you need only click on the Add button to first create a Library, and then, with the library selected by default, click Add again to create a profile inside the library.

4.3.9 Updated “Configure Security” dialog with new security properties

The “Configure Security” dialog for a Participant QoS has been updated. The following fields have been added to configure new security properties; an example is shown below:

- DataWriter Data and Metadata Key Sharing Enabled (Cryptography tab)
- RTPS Protection Key (Cryptography tab)
- Alternative Permissions CA certificate(s) (Access control tab)
- Alternative CA file(s) (Authentication tab)
- Enforce X.509 v3 Extension (Authentication tab)
In addition, the following fields and properties have been added to the “Security Global Settings” dialog; an example is shown below:

- **Private Key Password** (Authentication tab)
- **Authentication Request Timeout (sec)** (Participant Trust tab)
- **Simple Participant Discovery DataReader's Mode** (Participant Trust tab)
4.4  What’s Fixed in 6.1.0

4.4.1 Existing comments in XML/IDL text area not updated when modified

Editing an existing comment in the XML/IDL view via the Comment box in the upper-right didn’t update the comment in the text area. The updated comment was only shown after reopening the XML/IDL view. Now the comment is updated without reopening the XML/IDL view.

[RTI Issue ID SYSD-727]
4.4.2 Invalid content in IDL view when using aliases

Using aliases (long, short, …) for types resulted in “undefined” types in the IDL view. Now aliases are properly shown in the IDL view.
[RTI Issue ID SYSD-740]

4.4.3 Crash when trying to create a type inside a module

System Designer crashed when trying to create a type inside a module under the following circumstances:

- Select a module and click on the + button located in the Structured View. Don’t select any element and instead click outside of the menu.
- Click on the + button near the Help button. Again, don’t select any element and click outside the menu.
- Click again on the + button located in the Structured View.

This problem has been resolved.
[RTI Issue ID SYSD-765]

4.4.4 Uint64 member produced invalid IDLs

Adding a Uint64 member to a structure produced an invalid IDL. This issue has been resolved.
[RTI Issue ID SYSD-766]

4.4.5 Use of constants in @min, @max, and @default fields

System Designer now allows you to use constant value names in @min, @max, and @default fields.
[RTI Issue ID SYSD-767]

4.4.6 Version control issues

System Designer saved the project path (ui_projectPath), causing issues when dealing with the version control system. This issue has been resolved.
[RTI Issue ID SYSD-769]
4.4.7 Inheritance Chain edit icon was enabled when the QoS Profile was a QoS Snippet

The Inheritance Chain edit icon was enabled when the QoS Profile was a QoS Snippet. Now the edit button is only enabled when the QoS profile is not an QoS Snippet.

[RTI Issue ID SYSD-810]

4.4.8 Unbounded sequences of strings converted to strings

Refreshing System Designer caused unbounded sequences of strings to be transformed into strings. This problem has been resolved.

[RTI Issue ID SYSD-816]

4.4.9 System Designer failed to delete QoS Library, Domain Library, and Participant Library

Previously, when you deleted a QoS Library, Domain Library, or Participant Library, then later reopened the project, System Designer had actually failed to delete the library. Now System Designer deletes it.

[RTI Issue ID SYSD-833]

4.4.10 Union Editor did not allow addition of cases after adding default case

Previously, System Designer prevented you from adding any cases to a union after the “default” case was added. Now you can. As part of this fix, System Designer now also prevents you from adding duplicate default cases.

[RTI Issue ID SYSD-835]

4.4.11 System Designer didn’t allow creating a constant of an enumerator

System Designer now allows creating a constant of an enumerator.

[RTI Issue ID SYSD-839]

4.4.12 List of XML files incorrectly updated when importing invalid XML file

If the process of importing an XML file failed—for example, the XML file contained an element already defined in the open project—System Designer still showed that file in the list of XML files. This problem has been resolved.

[RTI Issue ID SYSD-854]
4.4.13 List of XML files in XML view not updated when creating new XML file

The list of XML files in the XML view was not updated when creating a new XML file. Now, the newly created file will be shown in the list of XML files.

[RTI Issue ID SYSD-855]

4.4.14 Browse button in Ez-Secure dialog didn’t work

Selecting a folder by the Browse button in the EZ-Secure dialog didn’t work. This prevented System Designer from generating all the required Secure files and updating the participant QoS. This issue has been resolved.

[RTI Issue ID SYSD-862]

4.4.15 Invalid (HTTP) 400 Bad Request when checking if a file exists

System Designer produced a (HTTP) 400 Bad Request if, when checking if a file exists, the file didn’t exist. This problem has been resolved. System Designer no longer produces an error in this case.

[RTI Issue ID SYSD-871]

4.4.16 Updated security property names

The following security properties have been renamed to match the latest DDS Security specification:

<table>
<thead>
<tr>
<th>Legacy property name (prefix with com.rti.serv.secure)</th>
<th>New property name (no prefix)</th>
</tr>
</thead>
<tbody>
<tr>
<td>authentication.ca_file</td>
<td>dds.sec.auth.identity_ca</td>
</tr>
<tr>
<td>authentication.private_key_file</td>
<td>dds.sec.auth.private_key</td>
</tr>
<tr>
<td>authentication.certificate_file</td>
<td>dds.sec.auth.identity_certificate</td>
</tr>
<tr>
<td>access_control.permissions_authority_file</td>
<td>dds.sec.access.permissions_ca</td>
</tr>
<tr>
<td>access_control.governance_file</td>
<td>dds.sec.access.governance</td>
</tr>
<tr>
<td>access_control.permissions_file</td>
<td>dds.sec.access.permissions</td>
</tr>
</tbody>
</table>

The legacy property names are no longer supported. System Designer will generate a warning if it detects any of the legacy property names being used by any XML file. For example, if the following XML file is loaded:

```xml
<participant_qos>
  <property>
    <value>
      <element>
        <name>com.rti.serv.secure.authentication.ca_file</name>
      </element>
    </value>
  </property>
</participant_qos>
```

(continues on next page)
System Designer will generate the following warning:

```
com.rti.serv.secure.authentication.ca_file has been deprecated, System...
Designer will ignore its value. Please update the property name with the new one:
<dds.sec.auth.identity_ca>
```

These properties have been moved from the Security Configuration dialog to the Global Security configuration. This dialog can be accessed by clicking on the “Globals” button in the Security Configuration dialog.

[RTI Issue ID SYSD-875]

### 4.4.17 System Designer crashed when attempting to load XML file that contained a forward_dcl

System Designer crashed when attempting to load an XML file that contained a forward declaration (forward_dcl). This issue has been resolved.

[RTI Issue ID SYSD-876]

### 4.4.18 @key fields couldn’t be set in derived structures

A derived structure couldn’t configure any of its members to be keyed. Now it can.
4.4.19 System Designer failed to import a zip project in Windows

System Designer failed to import a zip project in Windows with the following error:

```
SyntaxError: Unexpected token U in JSON at position XX
```

This problem has been resolved.

[RTI Issue ID SYSD-898]
4.5 Known Issues

Modules can be defined in multiple files, and each file might define the module with different annotations:

- @autoid
- @transfer_mode
- @language_binding
- @allowed_data_representation

Semantically, those annotations (when defined in a module) apply only to the contained entities.

Since internally modules are identified by a single object, when their definition comes from multiple files, their annotations must be the same.
Chapter 5

Copyrights

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

Trademarks

RTI, Real-Time Innovations, Connext, 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 Website: https://support.rti.com/

© 2021 RTI