



Rational Rose Real-Time Migration to DevOps Model RealTime

-

Getting Started Guide

Author: Anders Ek, HCL

Written May 30, 2012.

Last updated July 13, 2018 for DevOps Model RealTime 10.2.

Table of Contents

1 INTRODUCTION.....	3
2 THE SOURCE MODEL.....	3
3 IMPORTING THE MODEL.....	4
4 SUMMARY.....	7
5 APPENDIX: LINKS AND REFERENCES.....	8

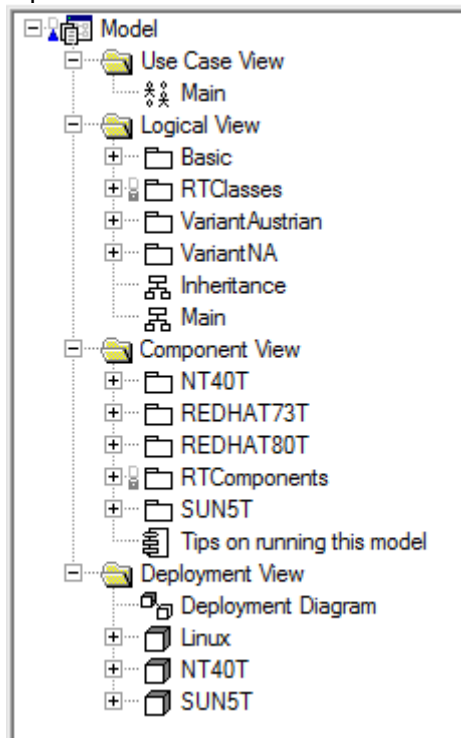
1 Introduction

The purpose of this document is to provide a quick introduction of how to move a model from Rational Rose Real-Time (RoseRT) to DevOps Model RealTime. The intention is to describe the workflow using a very simple example. For readers interested in the complete picture and how to plan a migration effort, the document "Rational Rose Real-Time to Model RealTime – Introduction" is recommended, since that document provides a view of the migration effort from a project perspective.

The example we use in the exercise is the Traffic Light example available in the RoseRT installation.

2 The Source Model

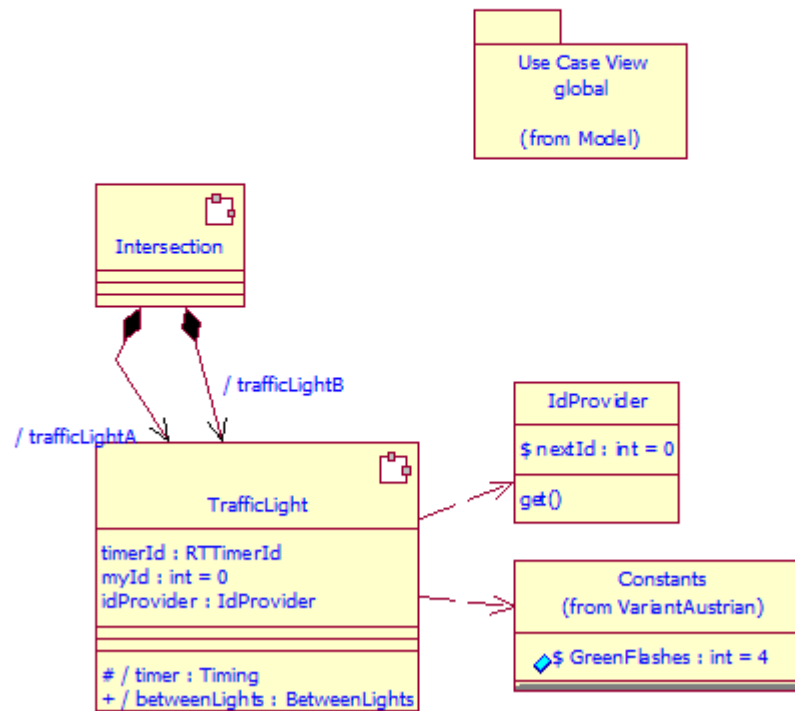
The Traffic Light example is available in the *Examples->Models->C++->TrafficLights* folder in the RoseRT installation. By opening this model, we see the following structure in the explorer.



As can be seen, the model contains the common structure with a Use Case View, Logical View, Component View, and Deployment view.

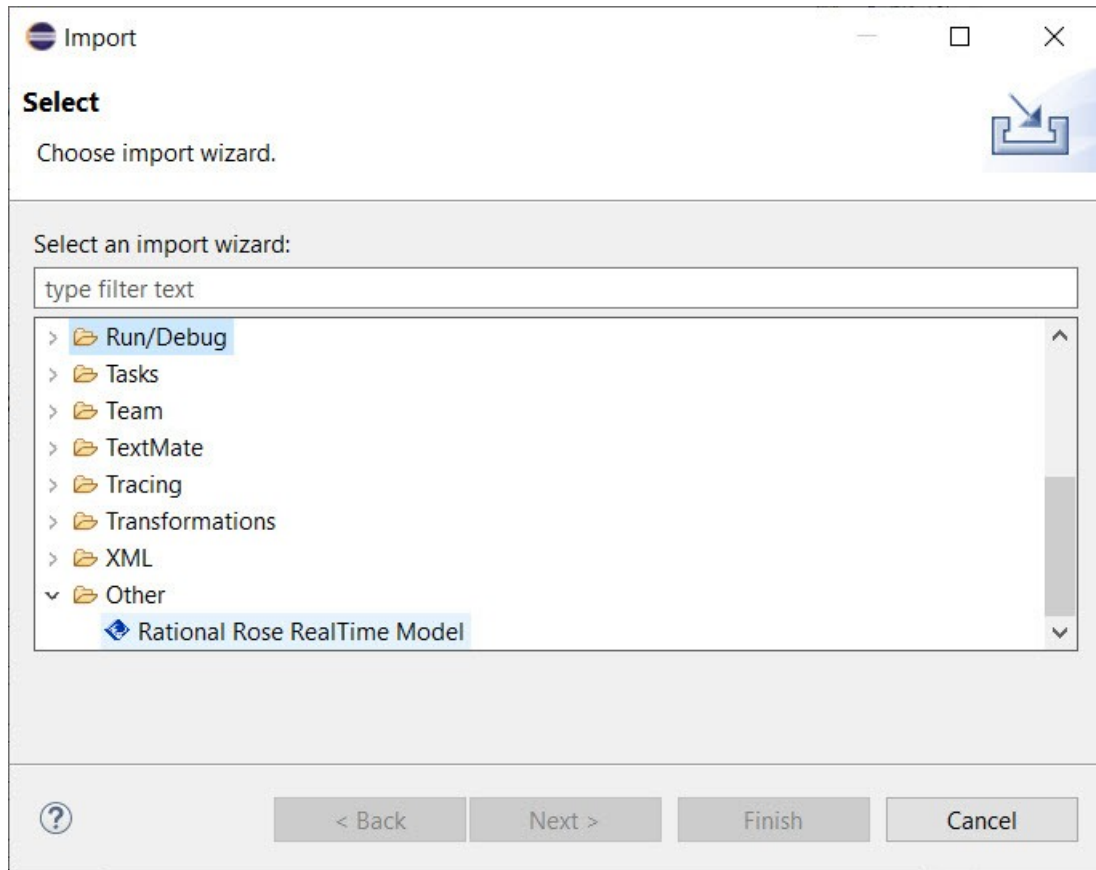
The Main diagram in the Logical view shows the key parts of the model.

Traffic Lights - Basic Implementation



3 Importing the Model

Import the RosERT model into Model RealTime using the Import wizard (*File->Import*). The RosERT importer is available in the *Other* folder in the wizard:



The next page of the wizard prompts for the RoseRT model to import and the project where the imported model should be placed. In this case, we chose a new project:

Import a Rational Rose RealTime Model

Import a Rational Rose RealTime Model

Import a Rational Rose RealTime Model

Source

Source model:

Registry file:

Destination for imported model

Existing project

Project name:

New project

Project name:

Location:

Use default location

The third wizard page displays information about controlled units and path maps.

Import a Rational Rose RealTime Model

Resolve Controlled Units and Path map symbols

There are no unresolved controlled units.

Source model:

Unresolved Controlled Units:

Controlled Unit	Original Location	Resolved Location

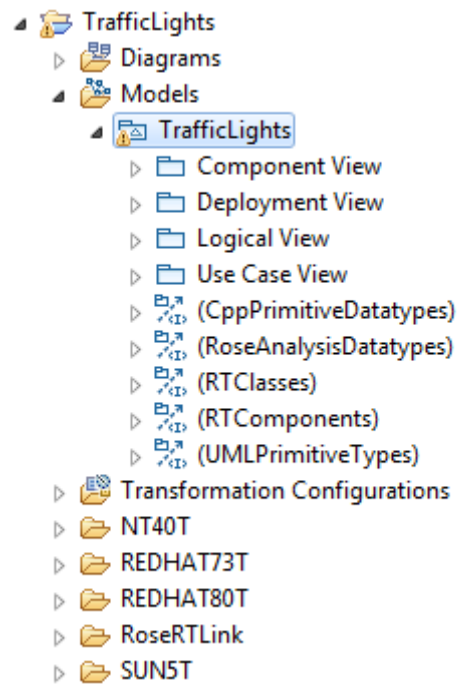
Specify the value for the unresolved Path map symbols:

Path Map Symbol	Value
ROSERT_HOME	D:/Eclipse/eclipse/eclipse/...

Ignore unresolved controlled units

For more complex models, there are more wizard pages that enable you to control various features of the import; however, for our simple example, we do not change anything and simply click **Finish** to import the model (or in case of Linux press Next in the remaining wizard pages until the Finish button is enabled).

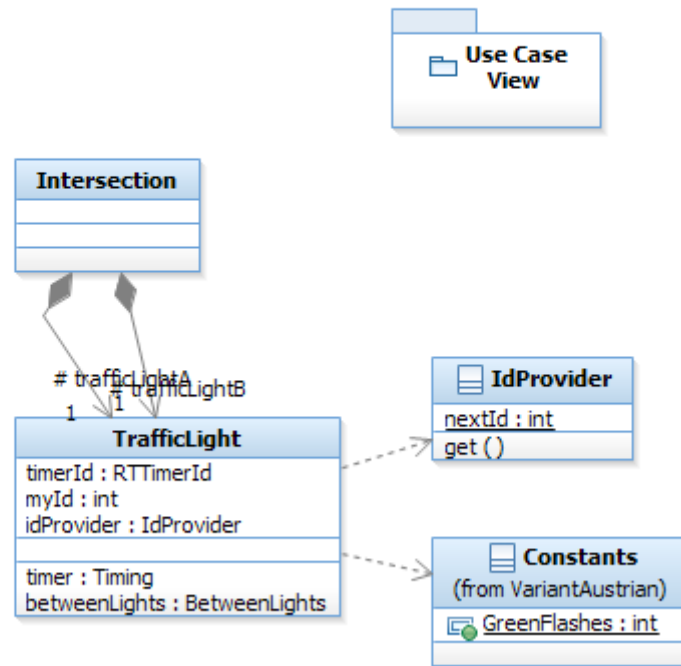
The result of the import is a new project in Model RealTime that is visible in the Project Explorer.



As can be seen, this corresponds closely to what was available in RoseRT with packages for the different model views; the Component, Deployment, Logical, and Use Case views.

The imported diagrams are, in most cases, very similar even though they might differ slightly in layout. As an example, we can look at the *main* diagram in the *Logical View*.

Traffic Lights - Basic Implementation



As we can see, the layout and contents of the different symbols are almost the same as in the RoseRT diagram we looked at previously.

4 Summary

In this short tutorial, you saw how to import a model from RoseRT to Model Real-Time. You import the model using the Import wizard in Model RealTime, based on a RoseRT model file. The import process preserves the model structure and semantics. The contents of the diagrams are preserved even though there might be some minor changes in the layout.

5 Appendix: Links and References

The following articles related to RoseRT migration are also available in the Model RealTime Help (at *Model RealTime User's Guide > Wiki > Importing Models from Rose RT*):

- Rational Rose Real-Time Migration to Model RealTime – Introduction
- Pre-migration Best Practices from Rose RealTime to Model RealTime
- Migration Best Practices from Rose RealTime to Model RealTime