FUJITSU

Fujitsu Software
Technical Computing Suite V4.0L20

Development Studio
IDE User's Guide

J2UL-2570-01ENZ0(06)
September 2025

Preface

Purpose of This Manual

This guide describes the features and the usage of the IDE(Integrated Development Environment) function for the system that has the Fujitsu CPU A64FX installed.

Intended Readers

This document is intended for integrated development environment users who build programs and execute jobs. Readers of the document are assumed to have knowledge about program development work, the Job Operation Software, and Eclipse.

Organization of This Manual

This manual consists of the following sections.

Chapter 1 Overview of the IDE

This chapter provides an overview of the integrated development environment.

Chapter 2 Installation Procedures for the Integrated Development Environment

This chapter describes procedures for installing the integrated development environment.

Chapter 3 Basic Usage of Eclipse

This chapter describes basic usage of the integrated development environment.

Chapter 4 Using Fujitsu Extended Functions

This chapter describes procedures for using the Fujitsu extended functions included in the integrated development environment.

Glossary

This appendix describes the terms used in this manual.

Related Manuals

This book relates to the following manuals. If necessary, refer also to these manuals.

- "Fortran Language Reference"
- "Fortran User's Guide"
- "Fortran User's Guide Additional Volume COARRAY"
- "Fortran Compiler Messages"
- "C User's Guide"
- "C++ User's Guide"
- "C/C++ Compiler Optimization Messages"
- "Fortran/C/C++ Runtime Messages"
- "MPI User's Guide"

Also, refer to the manuals provided with the following related software:

- "Job Operation Software"
- "FEFS/LLIO"

Export Controls

Exportation/release of this document may require necessary procedures in accordance with the regulations of your resident country and/or US export control laws.

Trademarks

- Linux(R) is the registered trademark of Linus Torvalds in the U.S. and other countries.
- Microsoft, Windows, and Windows Server are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.
- Mac is registered trademarks of Apple Inc.
- Eclipse is a registered trademark of Eclipse Foundation, Inc. or its affiliates in the United States and/or other countries.
- Java and OpenJDK are trademarks or registered trademarks of Oracle and/or its affiliates.
- All other trademarks are the property of their respective owners.
- The trademark notice symbol (TM,(R)) is not necessarily added in the system name and the product name, etc. published in this material.

Date of Publication and Version

Version	Manual code
September 2025, Version 1.6	J2UL-2570-01ENZ0(06)
September 2023, Version 1.5	J2UL-2570-01ENZ0(05)
March 2023, Version 1.4	J2UL-2570-01ENZ0(04)
November 2020, Version 1.3	J2UL-2570-01ENZ0(03)
September 2020, Version 1.2	J2UL-2570-01ENZ0(02)
March 2020, Version 1.1	J2UL-2570-01ENZ0(01)
February 2020, 1st Version	J2UL-2570-01ENZ0(00)

Copyright

Update History

Changes	Location	Version
Changed the supported basic software from Windows 10 to Windows 11.	2.1	Version 1.6
Changed the operation environments.	2.1	Version 1.5
Removed Windows 8.1 from Basic Software.	2.1	Version 1.4
Changed procedure for applying the installation package.	2.3.4	Version 1.3
Changed figures.	2.3.4 2.3.5 3.1 4.1.5.1 4.2 4.3	
Changed "Table 3.1 Setting Details of Synchronized Project".	3.1	
Changed "Table 4.15 Settings in [Libraries] (C)".	4.1.5.2	
- Changed the version of Eclipse. - Added note.	1	Version 1.2
Changed "Table 2.1 Environments Where Operation Has Been Tested".	2.1	
- Changed "Table 2.3 Components in the idefiles.zip". - Changed "Deployment Steps".	2.3.2	

Changes	Location	Version
Added procedure for applying the installation package.	2.3.4	
Added procedure for uninstalling the applied installation package.	2.3.5	
Changed "Table 3.1 Setting Details of Synchronized Project".	3.1	
Added note.	3.2.2	
Added note.	3.4	
Changed the note in "Table 4.3 Settings in [Preprocessor] (Fortran)".	4.1.5.1	
- Changed "Table 4.27 Settings in [Resources] - [Basic Settings] tab". - Added note to "Table 4.28 Settings in [Resources] - [Advanced] tab". - Changed "Table 4.29 Settings in [Application] tab". - Changed "Table 4.34 Setting in [Download Rule] window". - Added note.	4.2	
Added note.	4.3	
Added procedure for displaying the usage of computational resources.	4.3.1	
Added procedure for operating a job in a job view.	4.3.2 4.3.2.1 4.3.2.2 4.3.2.3 4.3.2.4 4.3.2.5	
Changed the procedure.	4.4	
Deleted Red Hat Enterprise Linux 7.6 from "Table 2.1 Environments Where Operation Has Been Tested".	2.1	Version 1.1
Deleted step 5.	2.3.1	
- Added note. - Changed the deployment steps.	2.3.2	
Added note.	3.4	
Changed the explanation of item name "Command". - "Table 4.1 Settings in [FUJITSU Fortran Compiler]" - "Table 4.6 Settings in [FUJITSU Compiler Fortran Linker]" - "Table 4.9 Settings in [FUJITSU C Compiler]" - "Table 4.14 Settings in [FUJITSU Compiler C Linker]" - "Table 4.17 Settings in [FUJITSU C++ Compiler]" - "Table 4.22 Settings in [FUJITSU Compiler C++ Linker]"	4.1.5.1 4.1.5.2 4.1.5.3	
- Added the explanation to "Figure 4.30 [Synchronize] tab". - Added the explanation about creating CPU Performance Analysis Report to the item name "Selected file(s):" in "Table 4.34 Setting in [Download Rule] window".	4.2	
Added note.	4.3 4.3.1	
Changed the procedure.	4.4	

Contents

Chapter 1 Overview of the IDE

This chapter describes the integrated development environment.

This document collectively refers to the "Eclipse IDE for Scientific Computing" package provided by Eclipse Foundation and the extended function (hereinafter referred to as the "Fujitsu extended function") as the "IDE (Integrated Development Environment)".

Note

The explanation in this document uses an environment where "Eclipse IDE 2020-06 R Packages" are installed. The procedures or windows may vary depending on the version of Eclipse. Replace the terms used there as necessary.

Eclipse IDE for Scientific Computing

Eclipse IDE for Scientific Computing is one of the Eclipse packages provided by the Eclipse Foundation. For details, see the official site of the Eclipse Foundation (https://www.eclipse.org). "Eclipse IDE 2020-06 R Packages" are hereinafter referred to as "Eclipse".

Fujitsu extended functions

The following table lists the extended functions to Eclipse provided for Technical Computing Suite. Hereinafter, the C/C++ compiler and FORTRAN compiler of the Technical Computing Suite are collectively referred to as compiler. The Technical Computing Suite's job operation software and profiler are referred to as job operation software and profiler, respectively.

Function Name	Outline
Build using Fujitsu compiler	Can use the compilers.
Job submission	Can submit jobs that use the Job Operation Software. (*1)
Job status check and operation	Displays the status of jobs on compute nodes and the node status. The function can also control jobs. (2)
CPU Performance Analysis Report display	It enables "CPU Performance Analysis Report," which is provided by the Profiler, to make measurements and create reports. For details on CPU performance analysis reports, see the *Profiler Use Guide*.

Note

(*1) The function for job submission supports only the submission of "normal job". For information about types of job like "normal job", see "End-user's Guide", which is a Job Operation Software manual.

(*2) You can operate only your jobs submitted from the integrated development environment.

See

For details on the terms used in this document, see "Glossary".

Chapter 2 Installation Procedures for the Integrated Development Environment

This chapter describes installation procedures for the integrated development environment.

2.1 Operation Environments

The integrated development environment is used via SSH connection from a client machine where Eclipse is installed to a login node. For details on the operating environment of Eclipse, see the official website of Eclipse Foundation. Operation check has been performed in the following environment.

Table 2.1 Environments Where Operation Has Been Tested

Basic Software	Java Runtime Environment (JRE) Java Development Kit (JDK)	Eclipse IDE for Scientific Computing
Microsoft Windows 11 (64bit)	OpenJDK 8 (1.8.0.332-1)	Eclipse IDE 2020-06 R Packages
macOS Catalina		
Red Hat Enterprise Linux 8.1		

In "4.4 CPU Performance Analysis Report Display", Microsoft Excel is used to display a CPU performance analysis report. For details on the operation environment for CPU performance analysis reports, see the *Profiler Use Guide*.

Note

For OpenJDK, please obtain (https://github.com/ojdkbuild) and use it at your own risk in accordance with the applicable terms of use (license conditions).

2.2 Eclipse Installation

This section describes how to install Eclipse.

Note

Even if you have already installed Eclipse, additionally install Eclipse for the integrated development environment.

2.2.1 JRE or JDK Installation

JRE (Java Runtime Environment) or JDK (Java Development Kit) is required for using Eclipse. Install the JRE or JDK as described in "Table 2.1 Environments Where Operation Has Been Tested".

2.2.2 Installation of Eclipse IDE for Scientific Computing

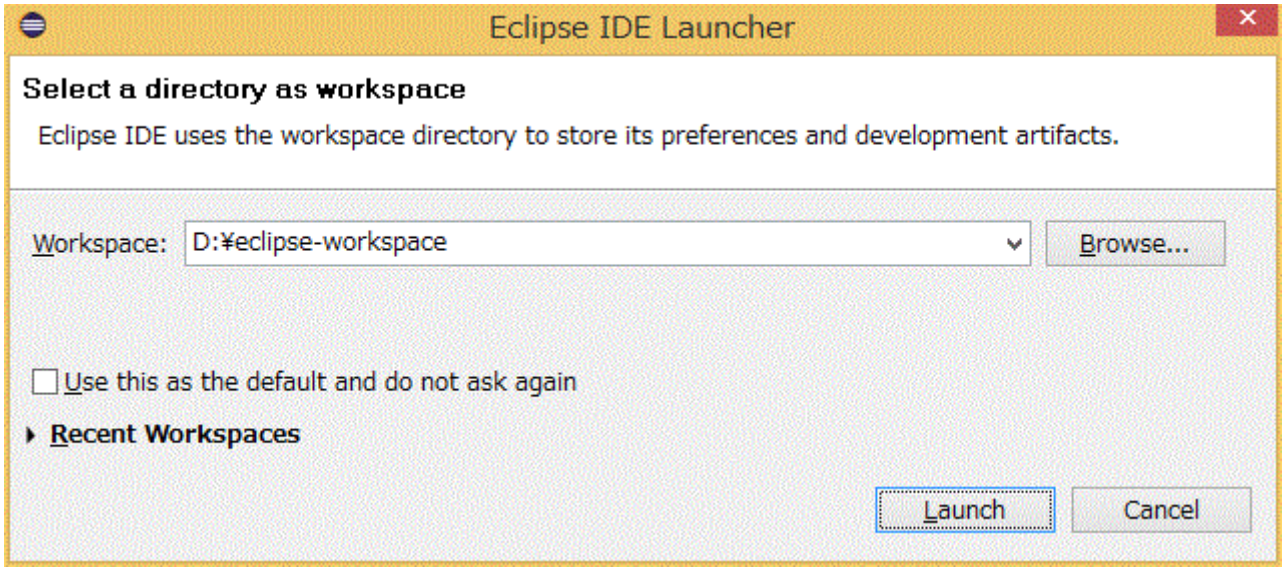
Install Eclipse IDE for Scientific Computing from the official site of the Eclipse Foundation (https://www.eclipse.org/). For the installation method, see the official site of the Eclipse Foundation. When using Eclipse Installer, select [Eclipse IDE for Scientific Computing] on the installation package selection screen.

Information

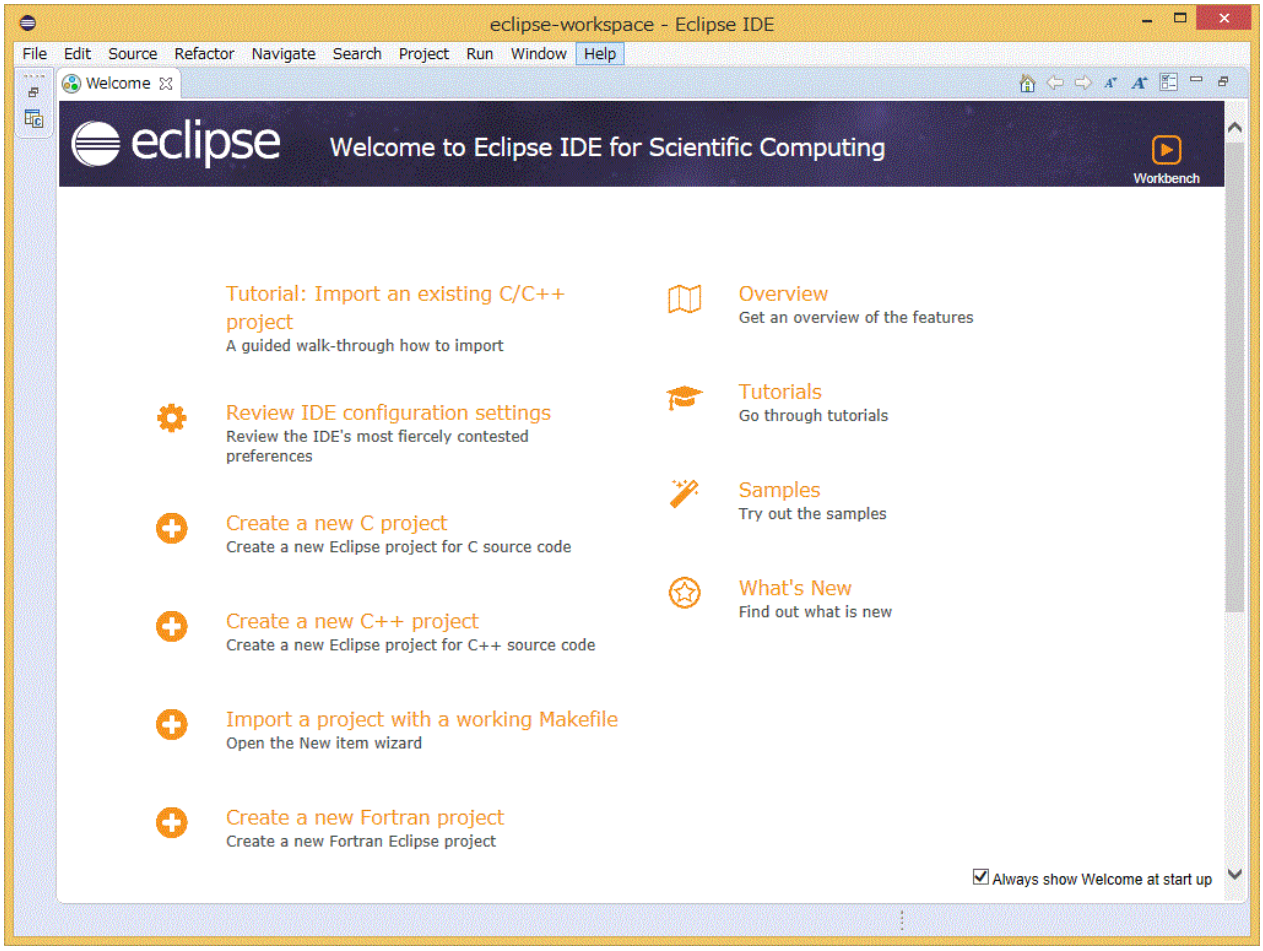
Depending on the environment, installation cannot be done correctly by Eclipse Installer. In such cases, download the Eclipse IDE for Scientific Computing package from the official site of the Eclipse Foundation, and deploy it at an arbitrary location.

2.2.3 Starting Eclipse

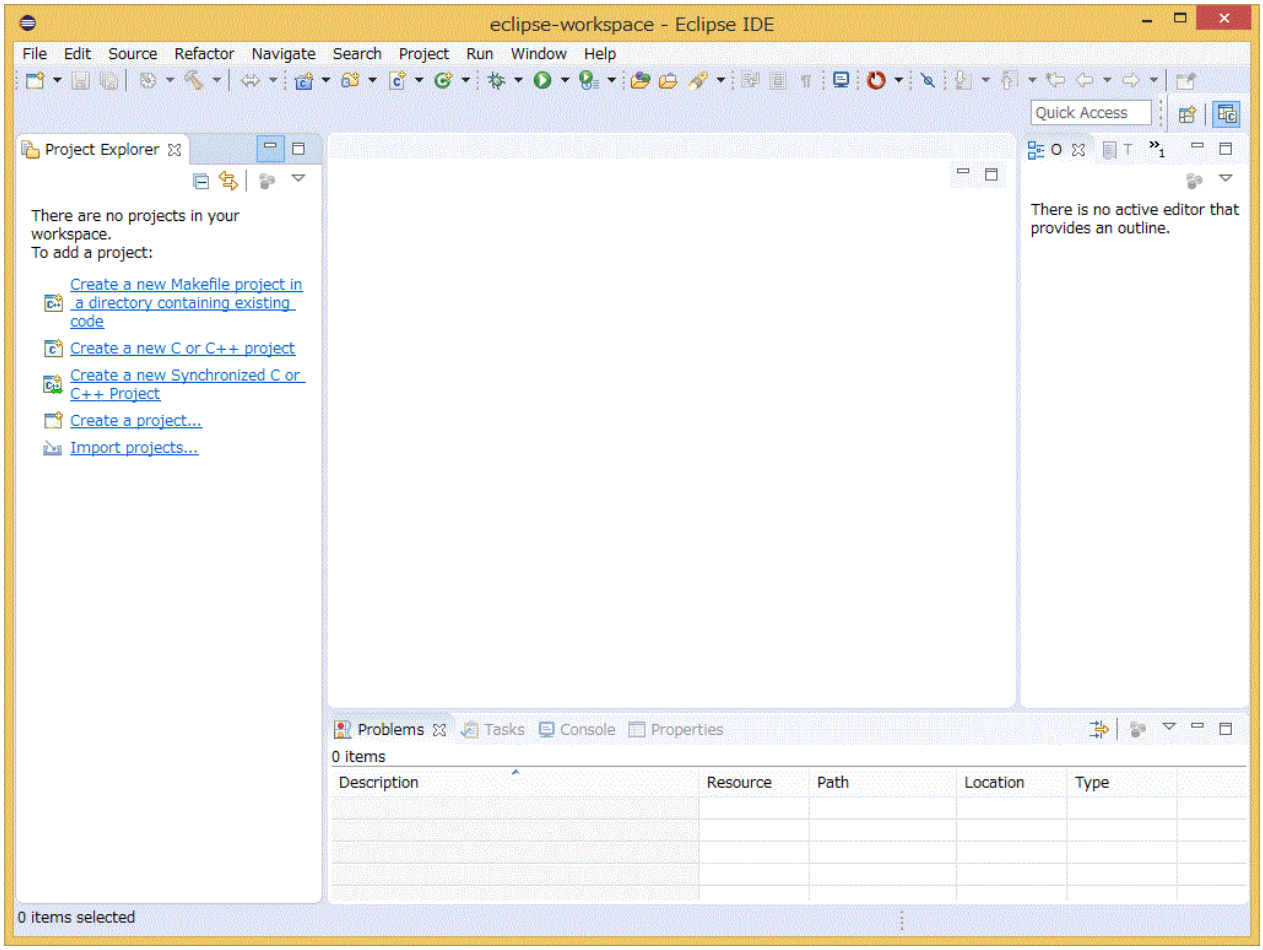
1. Start Eclipse. A window appears for specifying a workspace. In [Workspace:], specify the directory where you want to create a workspace. If the specified directory does not exist, it is automatically created. After [Workspace:] is set, click [Launch] button.



2. The [Welcome] view is not used. Close the view.



3. This is the initial screen.



2.3 Installation of Fujitsu Extended Functions

This section describes how to install Fujitsu extended functions. Perform this work when creating a new workspace.

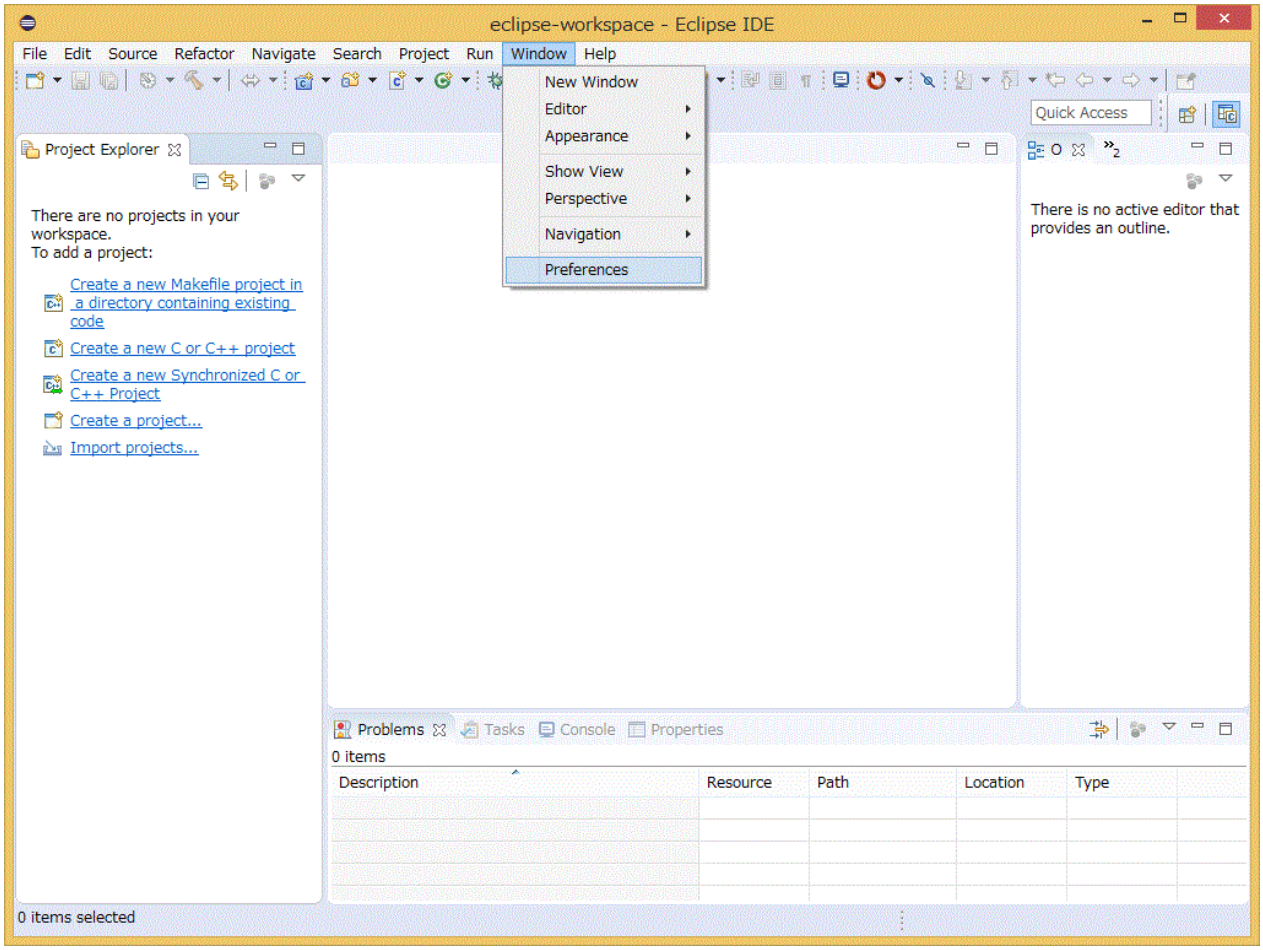
Note

Perform this work on each created workspace.

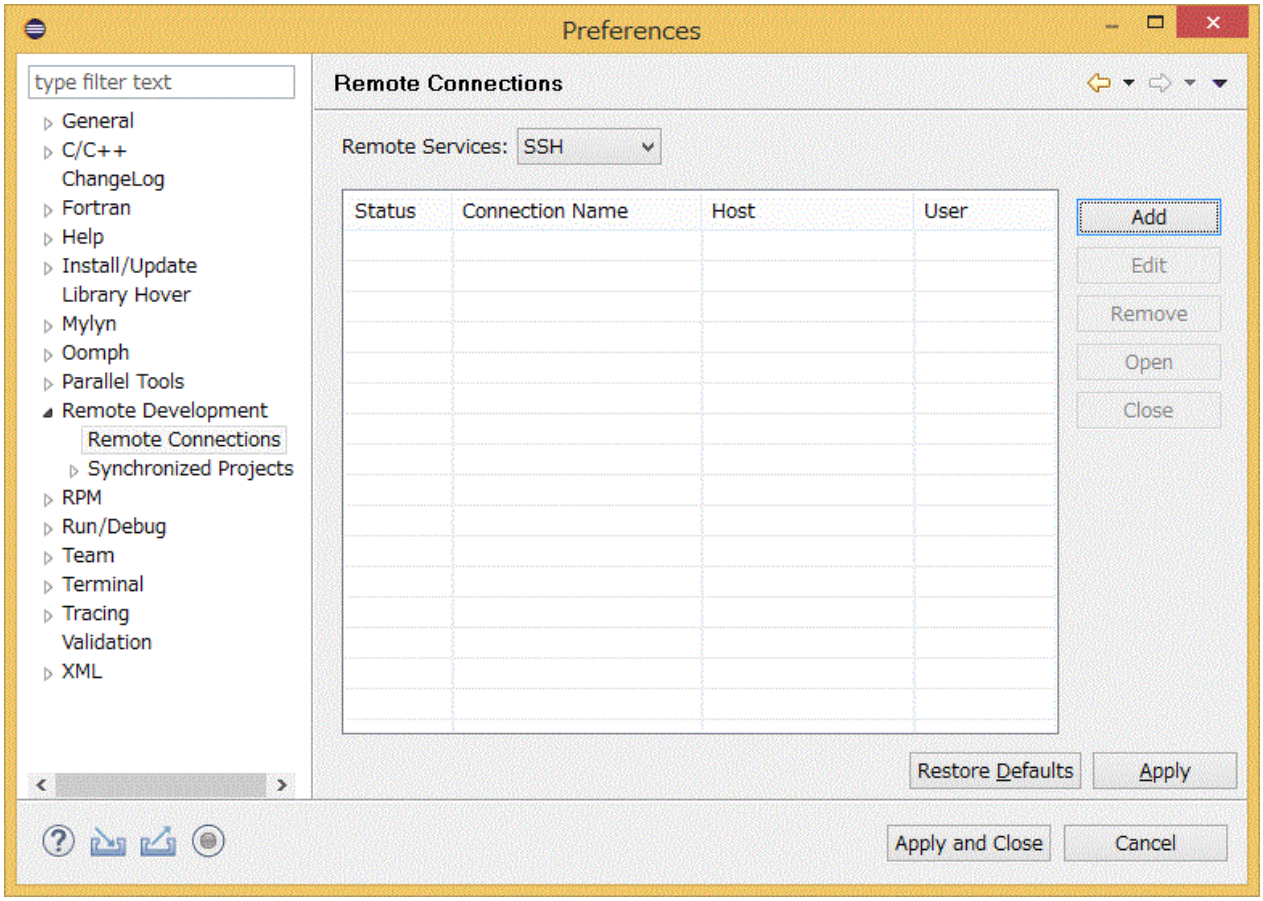
2.3.1 Connecting to the Login Node (Remote System)

Make settings for connecting the login node

1. Click [Window] - [Preferences] on the menu bar.



2. Select [Remote Development] - [Remote Connections] from the left pane in the [Preferences] window, and click [Add] button.



3. Set the necessary information in the [New Connection] window, and click [Finish] button. The following table shows setting details.

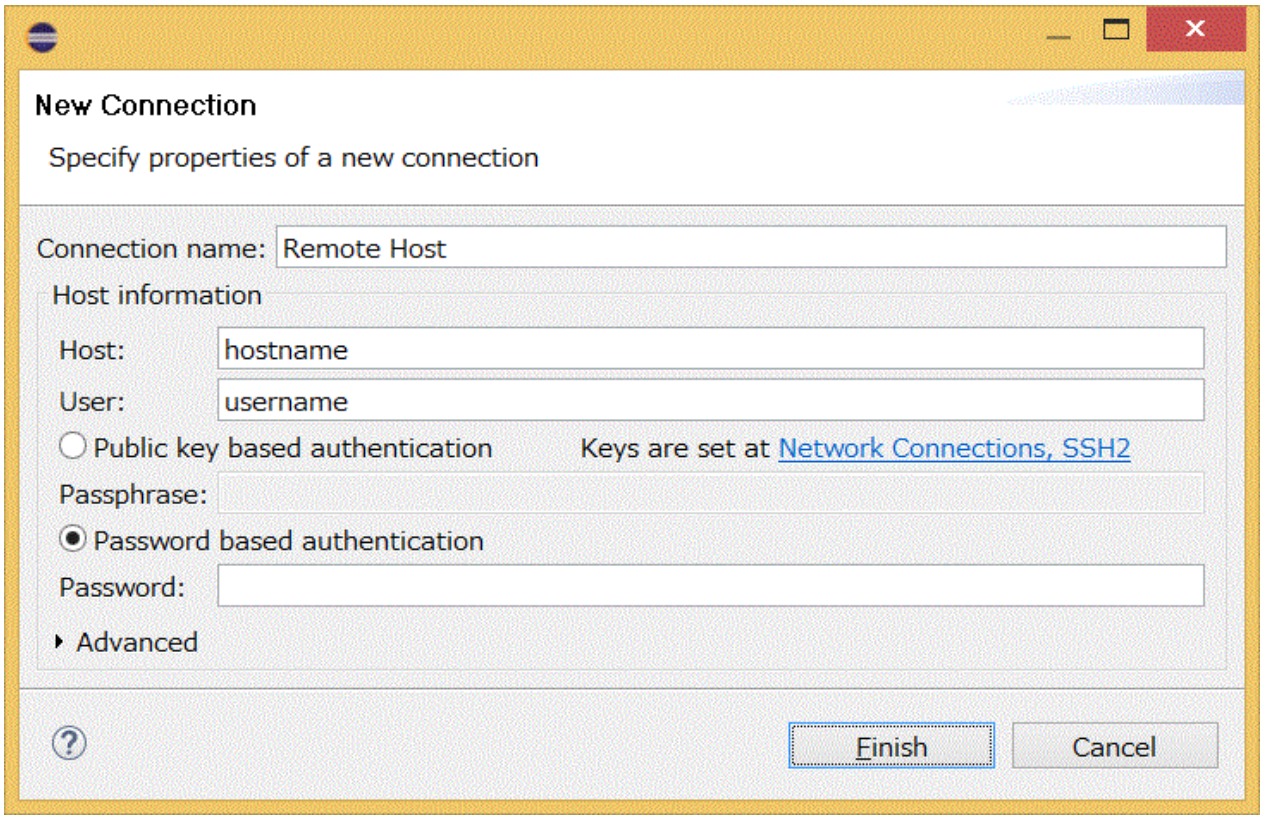


Table 2.2 [New Connection] Window Setting Details

Item Name		Setting Details
Connection name		Specify an identification name for [Remote Connections]. Specify an arbitrary name.
Host information	Host	Specify the host name or IP address of the login node.
	User	Specify the login user name of the login node.
	Public key based authentication	Select an authentication method for the login node connection. Public key based authentication: Public key authentication method Password based authentication: Password authentication method
	Password based authentication	
Advanced		Extended settings. Make settings as required.

4. In [Remote Connections], settings made in the [New Connection] window are added. Click the [Open] button while the added settings are selected, and confirm that Status becomes "open". Click the [Apply and Close] button to close the window.

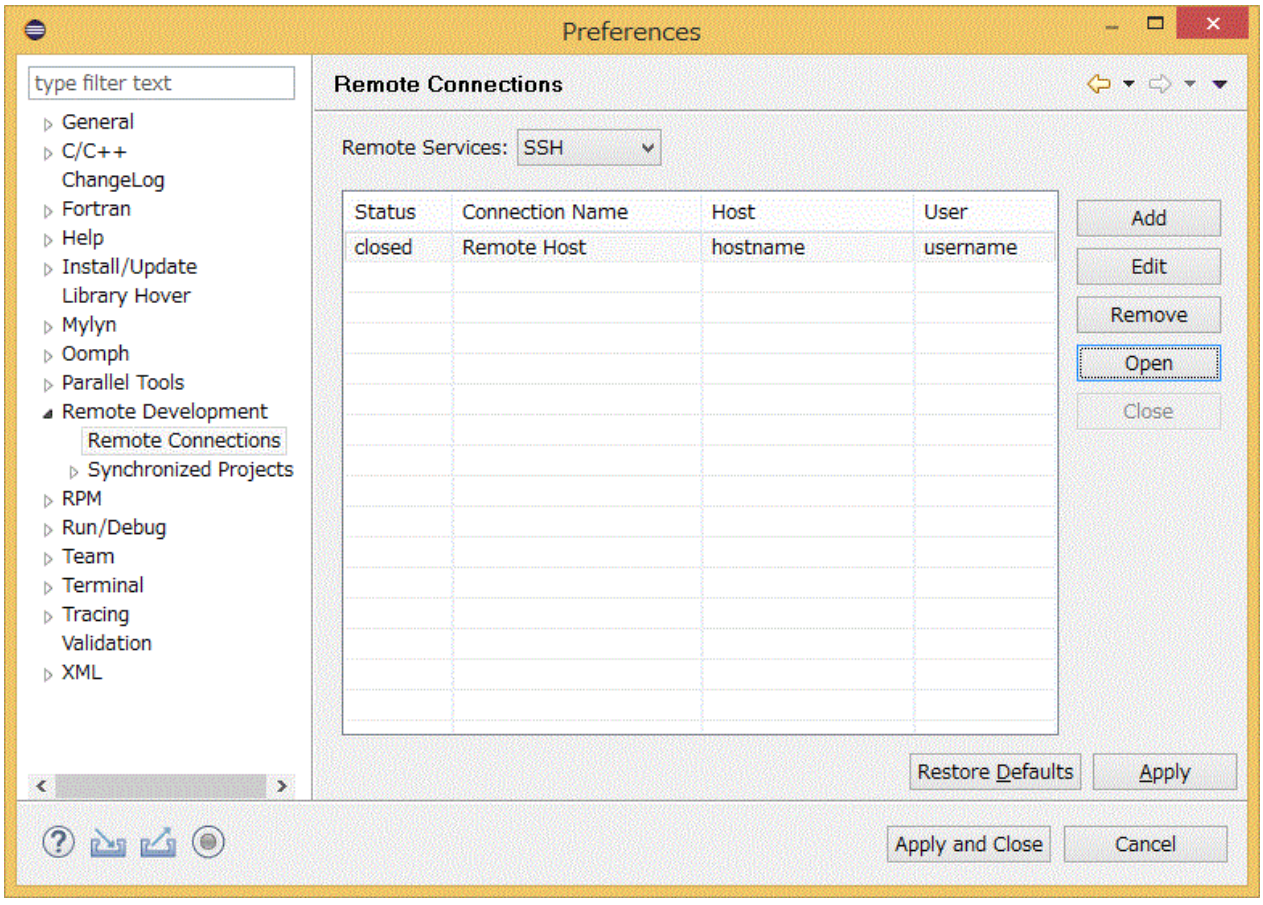


Figure 2.1 Status "open"

Status	Connection Name
open	Remote Host

2.3.2 Deploying Files for Fujitsu Extended Functions

Place files required to use the Fujitsu extended function. This work needs to be performed on both the login node and the client machine. The files required to use the Fujitsu extended function are stored in the following location on the login node.

Note

The "layout_default_TC_SUITE.xml" file used for this task may be distributed by your system administrator. Check with your system administrator for distribution.

File storage location for Fujitsu extended functions

```
/installation_path/misc/ide/idefiles.zip
```

For details on "*installation_path*", contact the system administrator.

The configuration in the idefiles.zip is as follows:

Table 2.3 Components in the idefiles.zip

Directory Name/File Name	Description	storage location
eclipsesettings	The files to be deployed in /*User's_home_directory*/.eclipsesettings on the login node are stored. (*)	Login node
fj_extention	The stored XML files are used in "2.3.3 Importing Configuration Files for Fujitsu Extended Functions".	Client machine
patches	The stored installation package is applied to Eclipse.	Client machine

(*) *User's_home_directory* is the home directory of user specified in "2.3.1 Connecting to the Login Node (Remote System)".

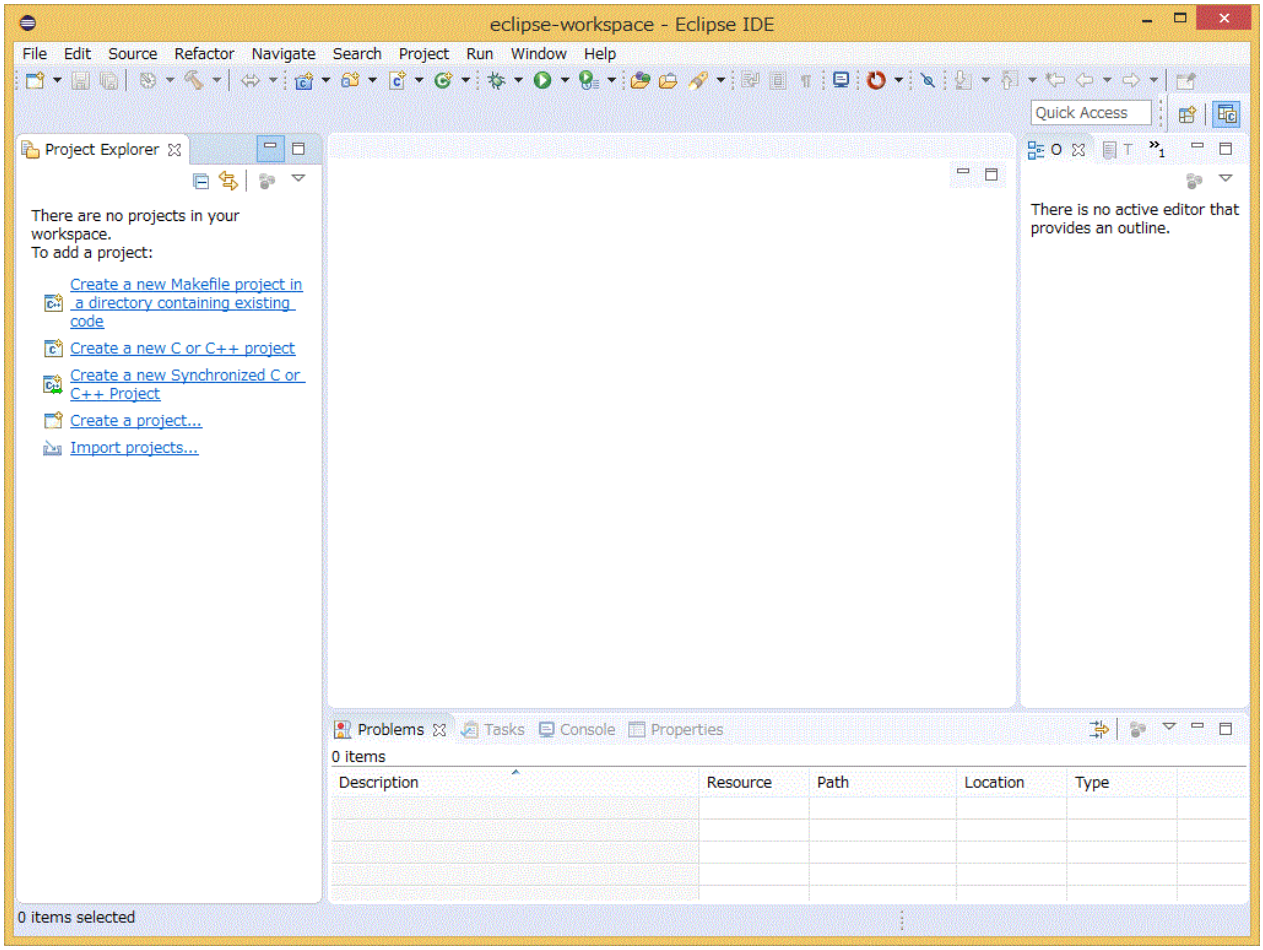
Deployment Steps

1. Expand the "idefiles.zip" file to any location on the login node.
2. Rename the "eclipsesettings" directory to the ".eclipsessettings" directory and copy it directly under the "/home/*username*" directory on the login node.
3. If your system administrator has distributed a "layout_default_TC_SUITE.xml" file, overwrite that file with the same name in the "/home/*username*/.eclipsessettings/samples" directory. If not, this is not necessary.
4. Transfer the "idefiles.zip" file to the client machine and extract it to any location.

2.3.3 Importing Configuration Files for Fujitsu Extended Functions

Import the necessary configuration files for using Fujitsu extended functions into Eclipse.

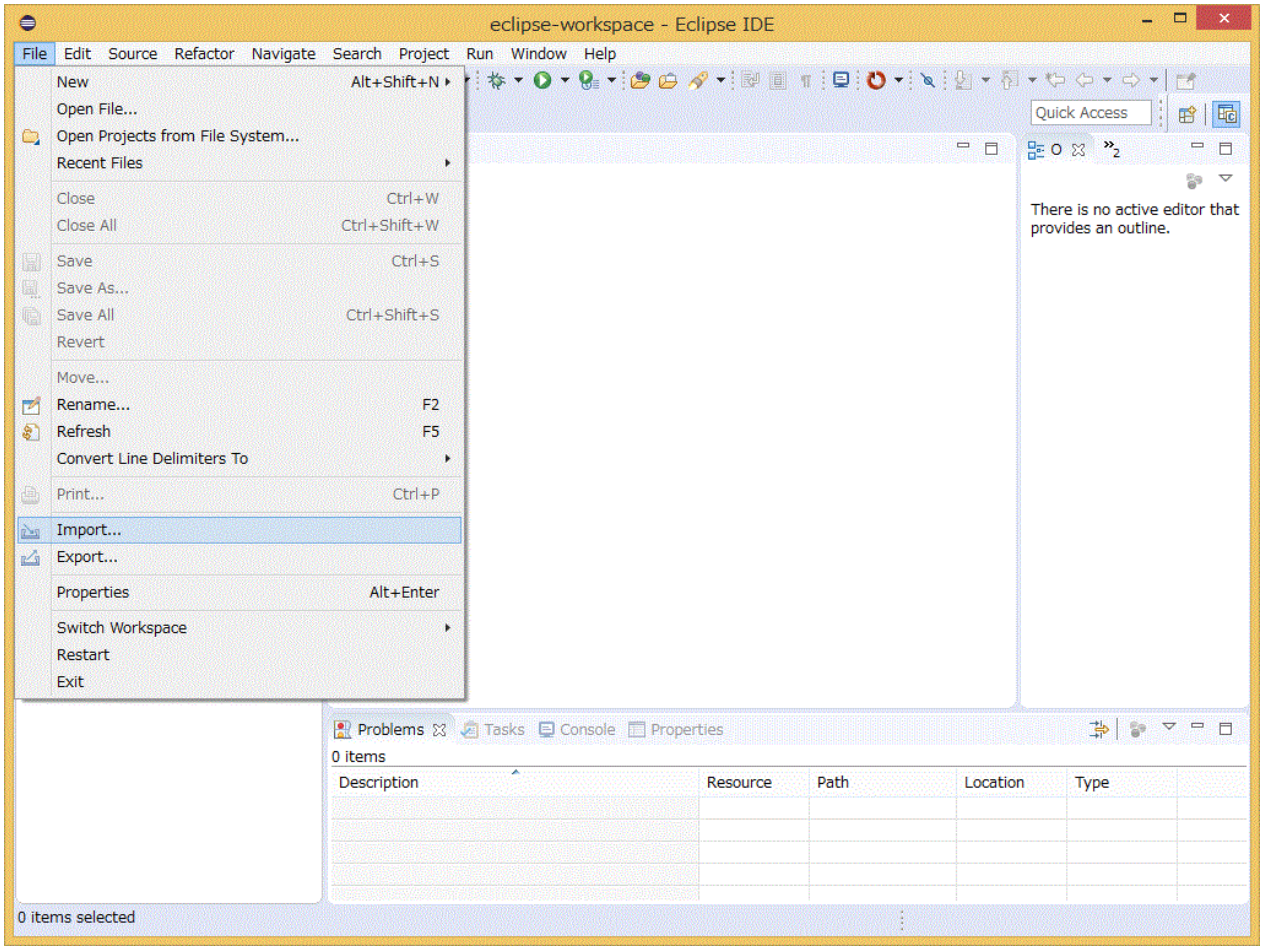
1. Confirm that the [Project Explorer] view is displayed.



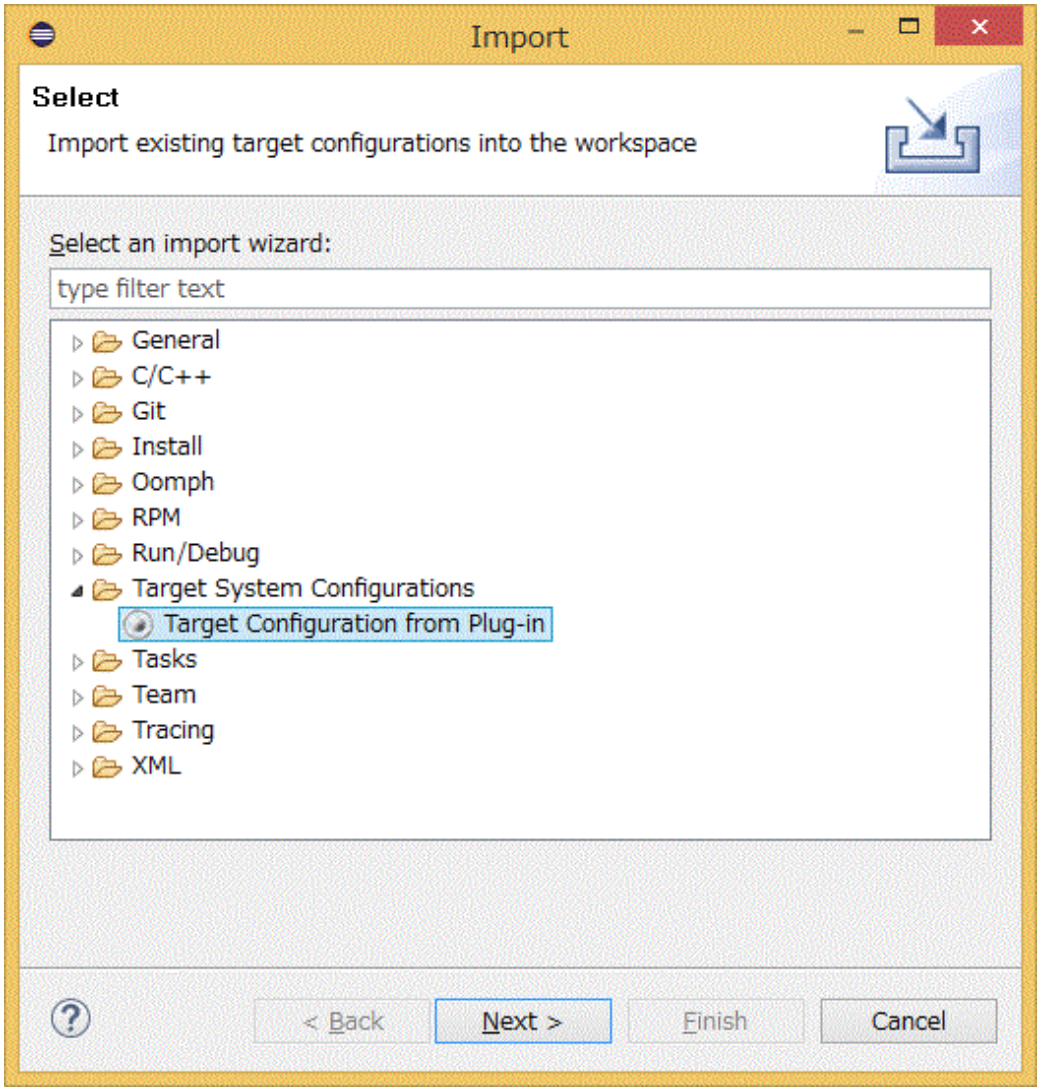
Information

If the [Project Explorer] view is not displayed, click [Window] - [Show View] - [Project Explorer] on the menu bar.

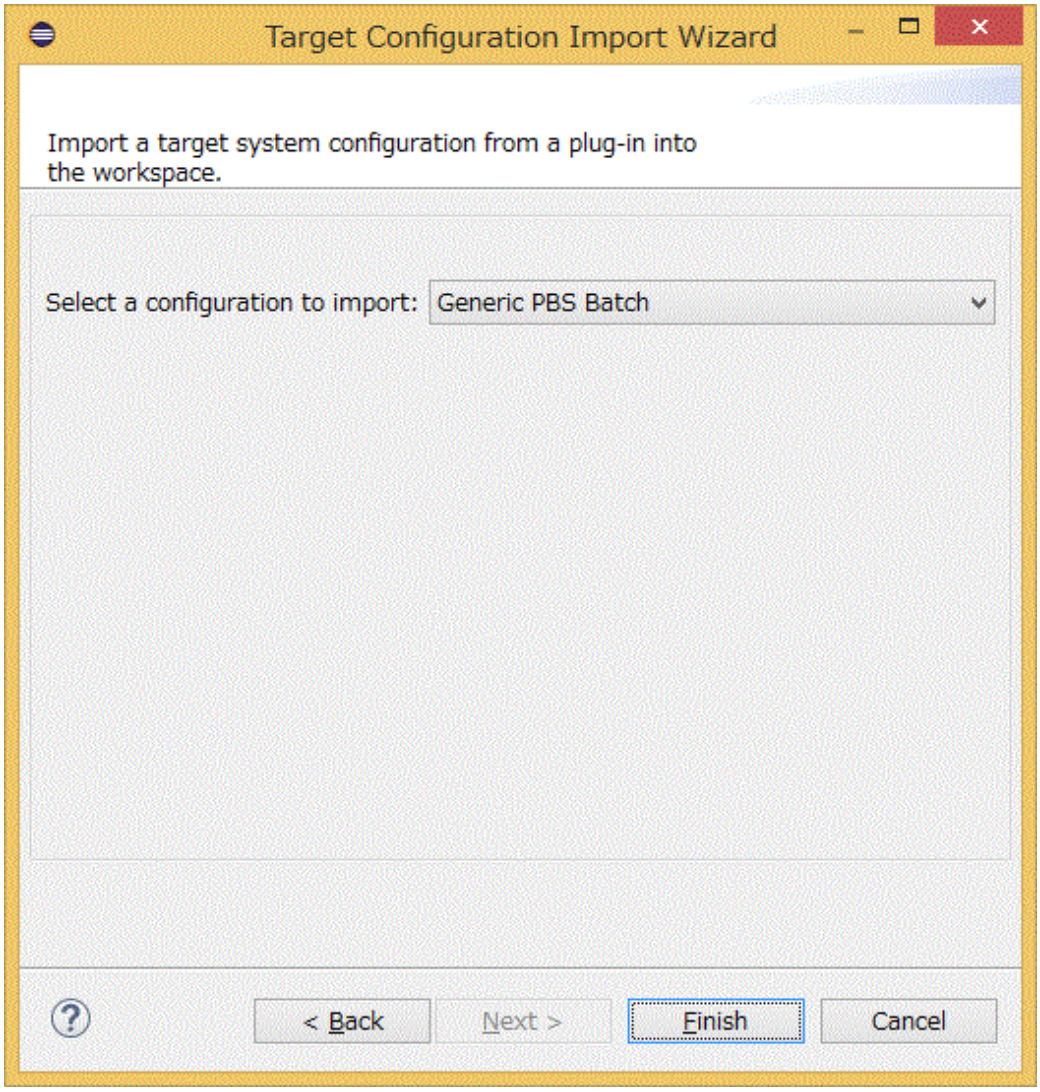
2. Click [File] - [Import...] on the menu bar.



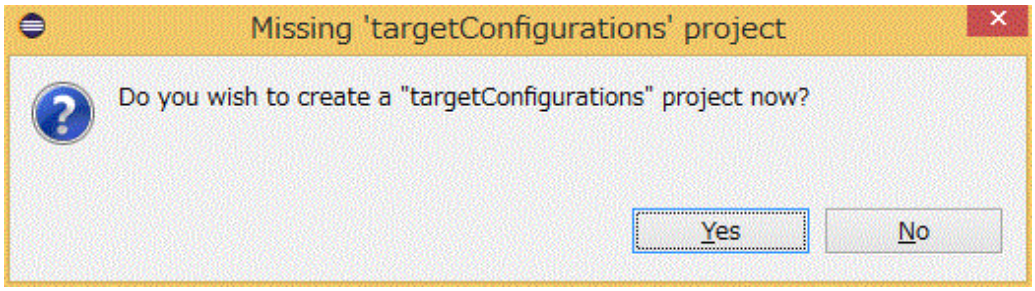
3. From [Select an import wizard] in the [Import] window, select [Target System Configurations] - [Target Configuration from Plug-in], and click [Next >] button.



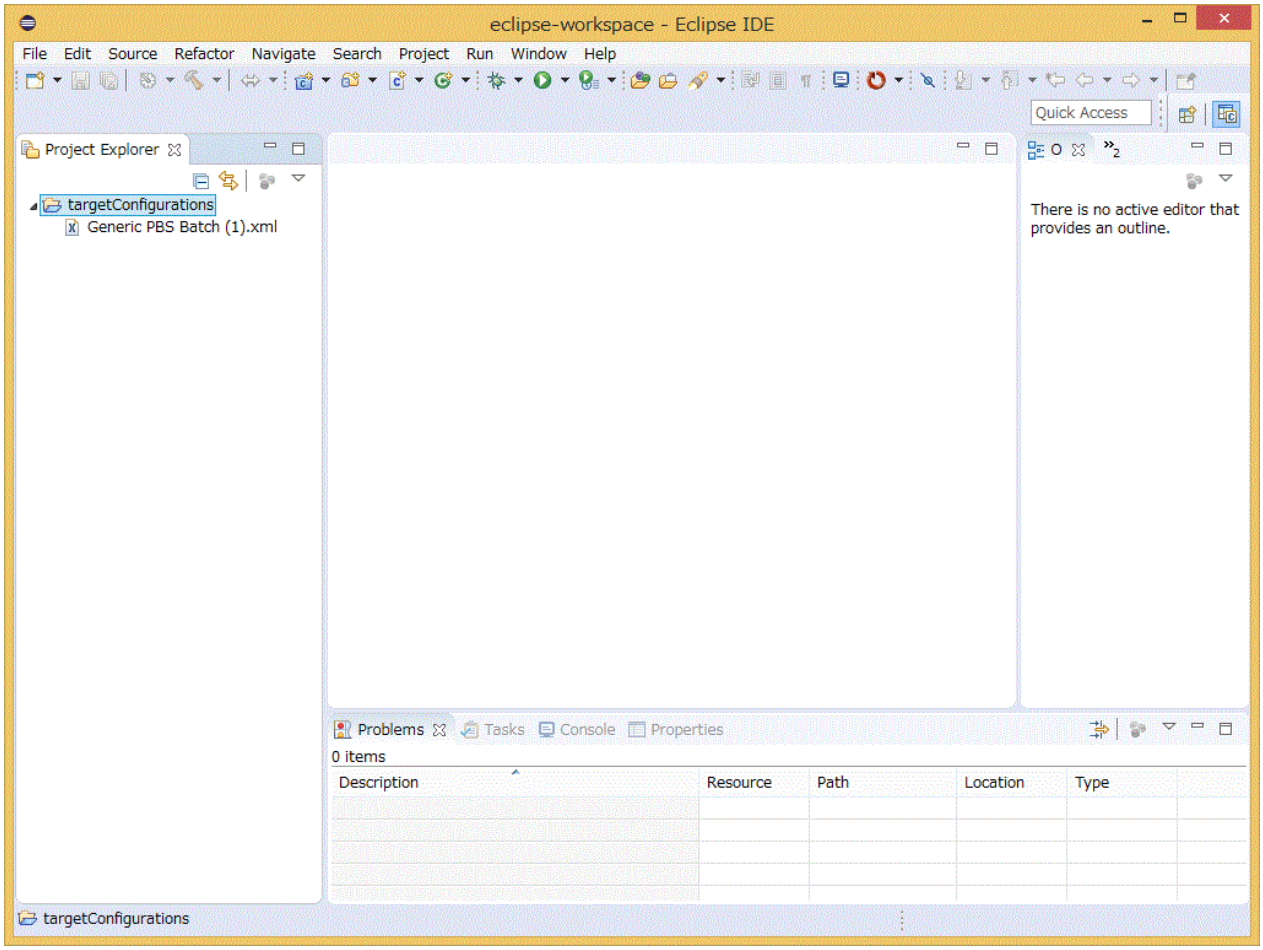
4. Select [Generic PBS Batch] from the pull-down menu, and click [Finish] button. The configuration created here is tentative one and is not used.



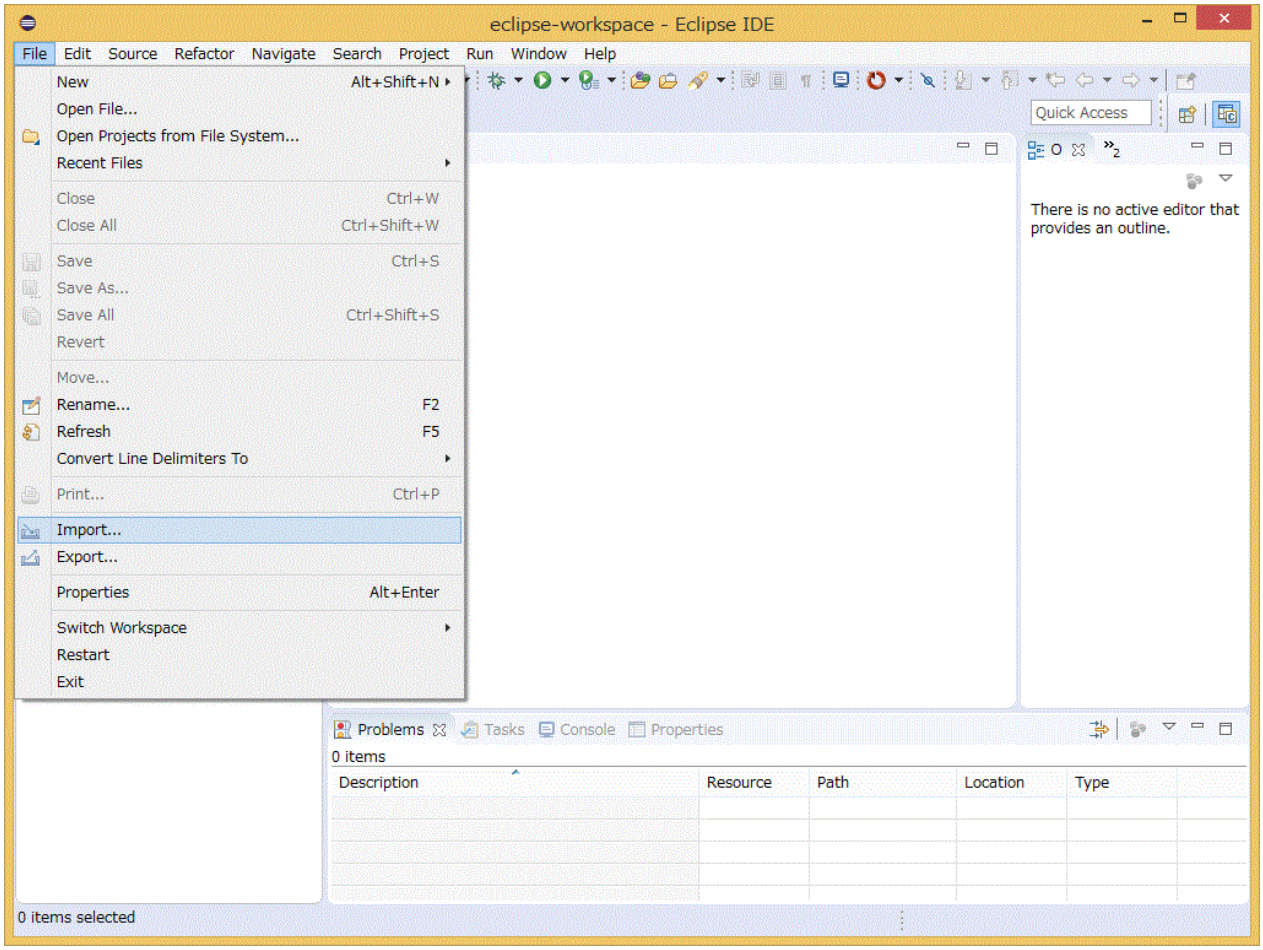
5. When asked whether to create the targetConfigurations project, click [Yes] button.



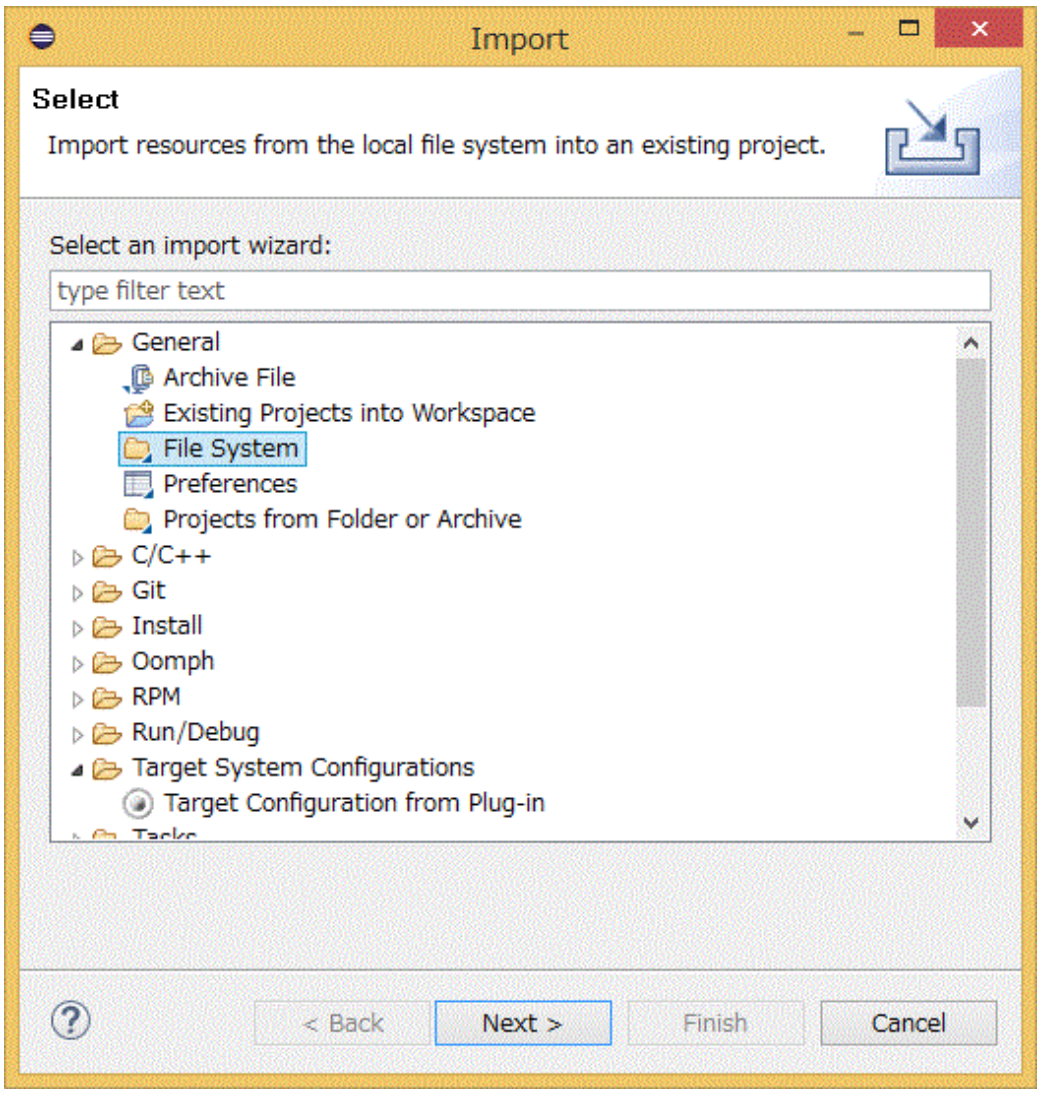
6. Confirm that [targetConfigurations] has been added to the [Project Explorer] view. Select [targetConfigurations].



7. Click [File] - [Import...] on the menu bar. The [Import] window appears.

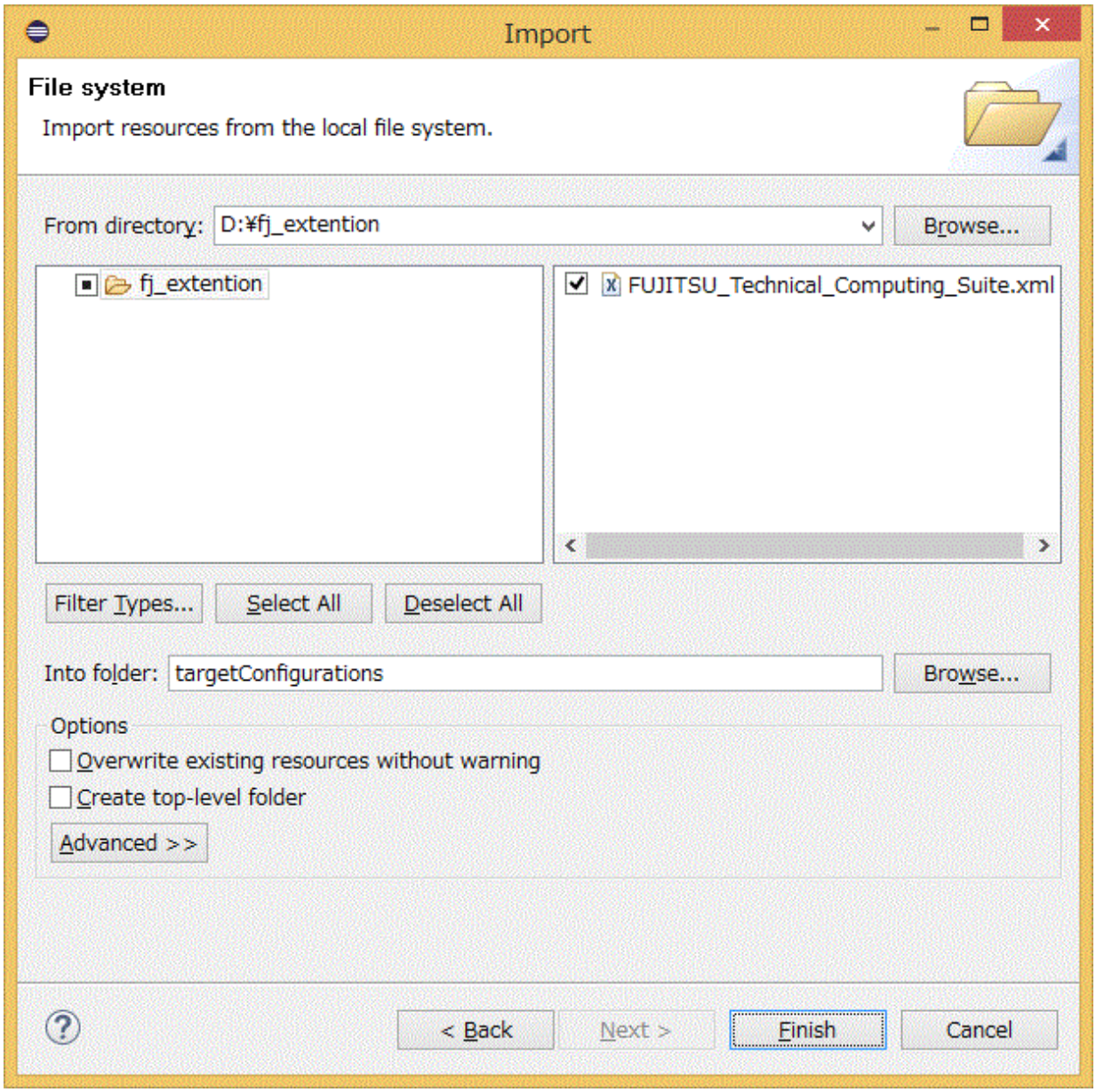


8. Select [General] - [File System] in the [Import] window, and click [Next >] button.

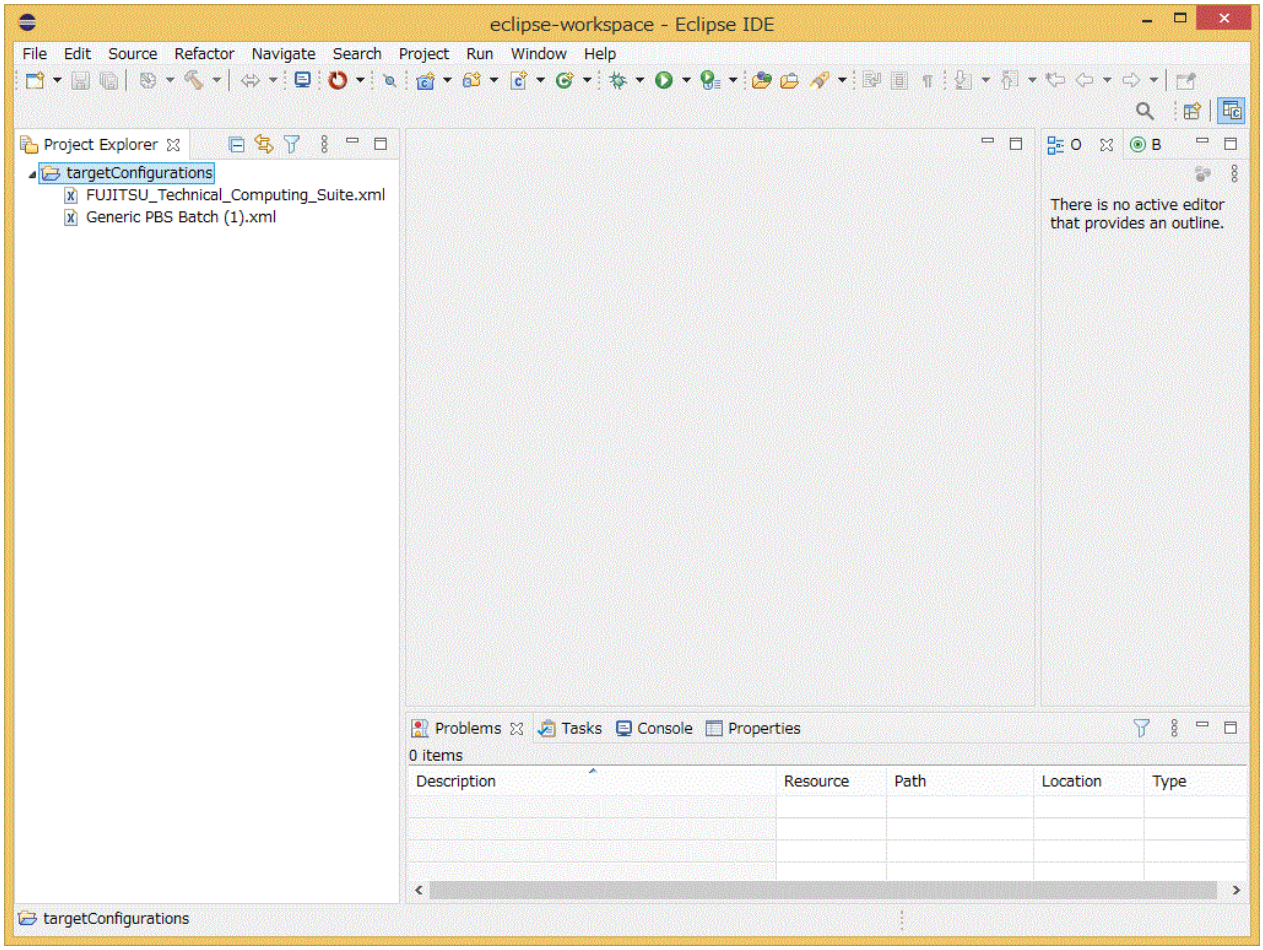


9. Click [Browse...] button at [From directory:], and specify the fj_extention directory deployed in "2.3.2 Deploying Files for Fujitsu Extended Functions". The box below the [Browse] button displays the files in the directory. Check the check box of the FUJITSU_Technical_Computing_Suite.xml file. Also, confirm that "targetConfigurations" is specified in [Info folder:]. If it is not

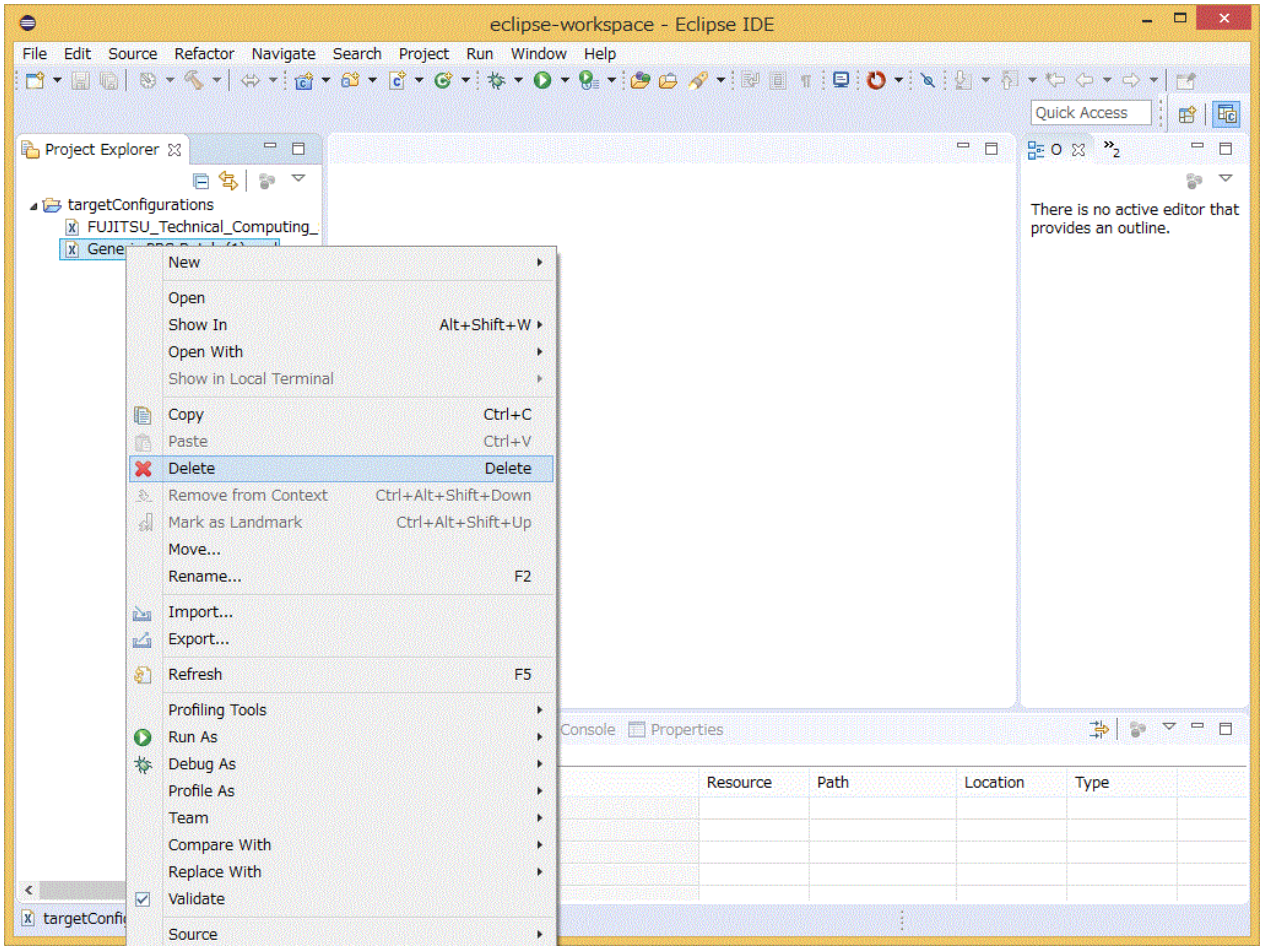
specified, click [Browse...] button on the right, and select "targetConfigurations" from the [Import into Folder] window. After completing all the settings, click [Finish] button.



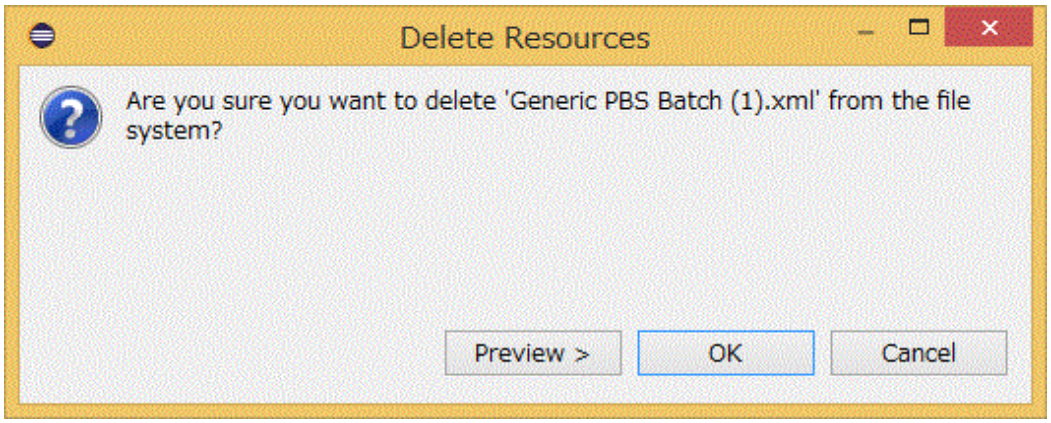
10. Confirm that "FUJITSU_Technical_Computing_Suite.xml" has been added to [targetConfigurations] in the [Project Explorer] view.



11. [Generic PBS Batch (1).xml] was created at the beginning but will not be used later.
    Delete it. Right-click [Generic PBS Batch (1).xml] in the [Project Explorer] view, and click [Delete].



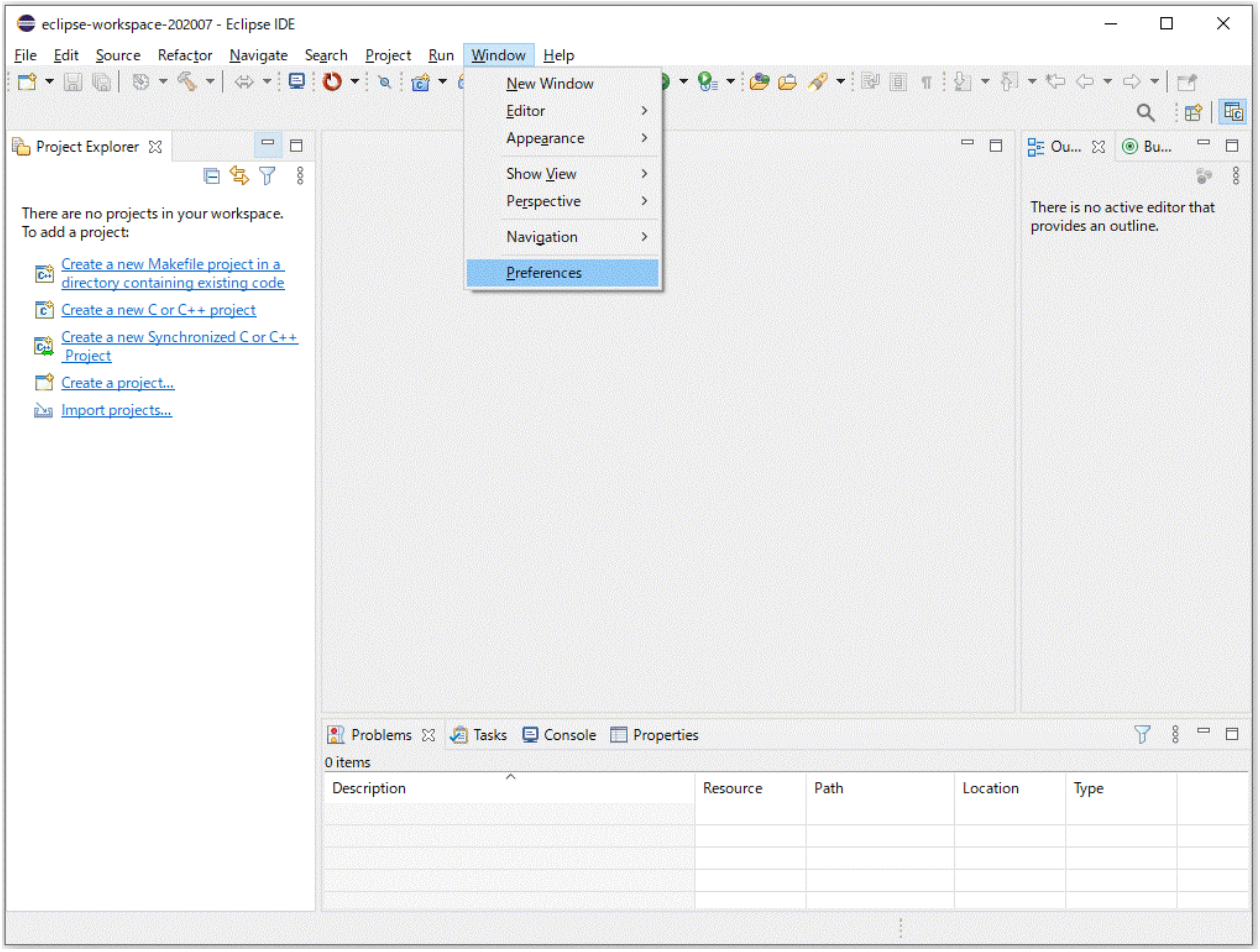
12. A confirmation window appears. Confirm that the target is "Generic PBS Batch (1).xml" and click the [OK] button.



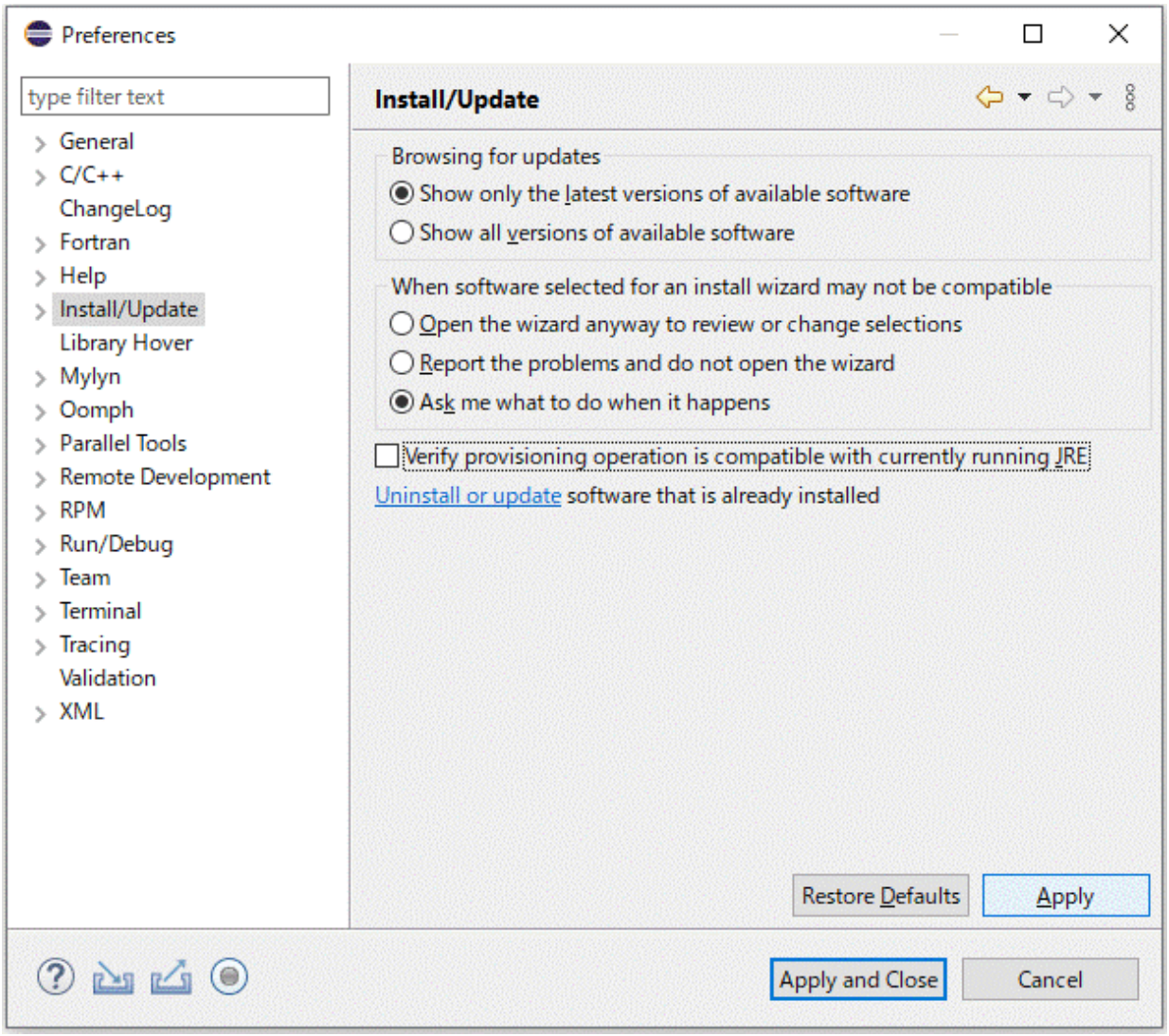
2.3.4 Applying the Installation Package

Apply the installation package to Eclipse.

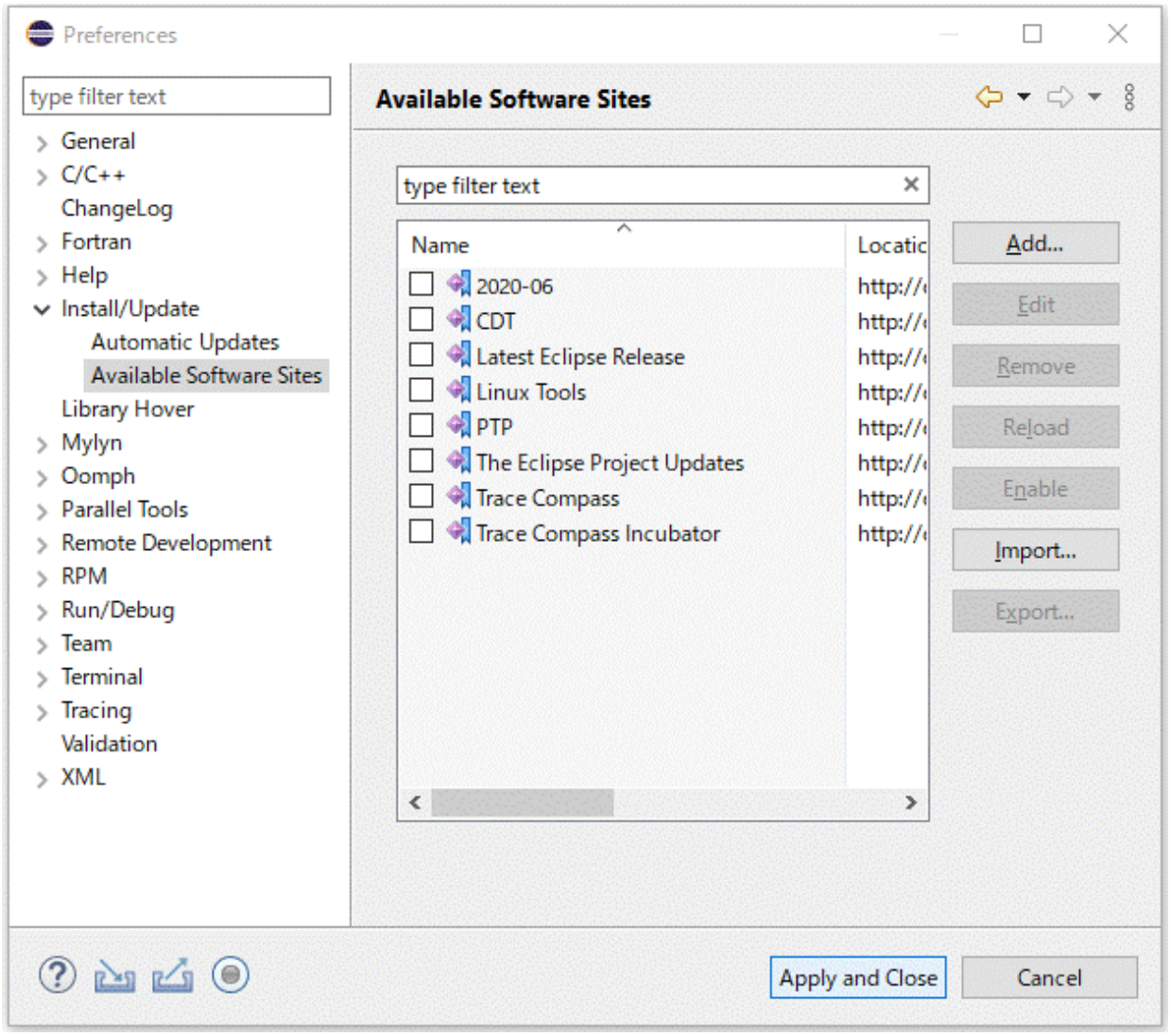
1. Click [Window] - [Preferences] on the menu bar.



2. Select [Install/Update] from the left pane in the [Preferences] window, uncheck [Verify provisioning operation is compatible with currently running JRE], and click the [Apply] button.



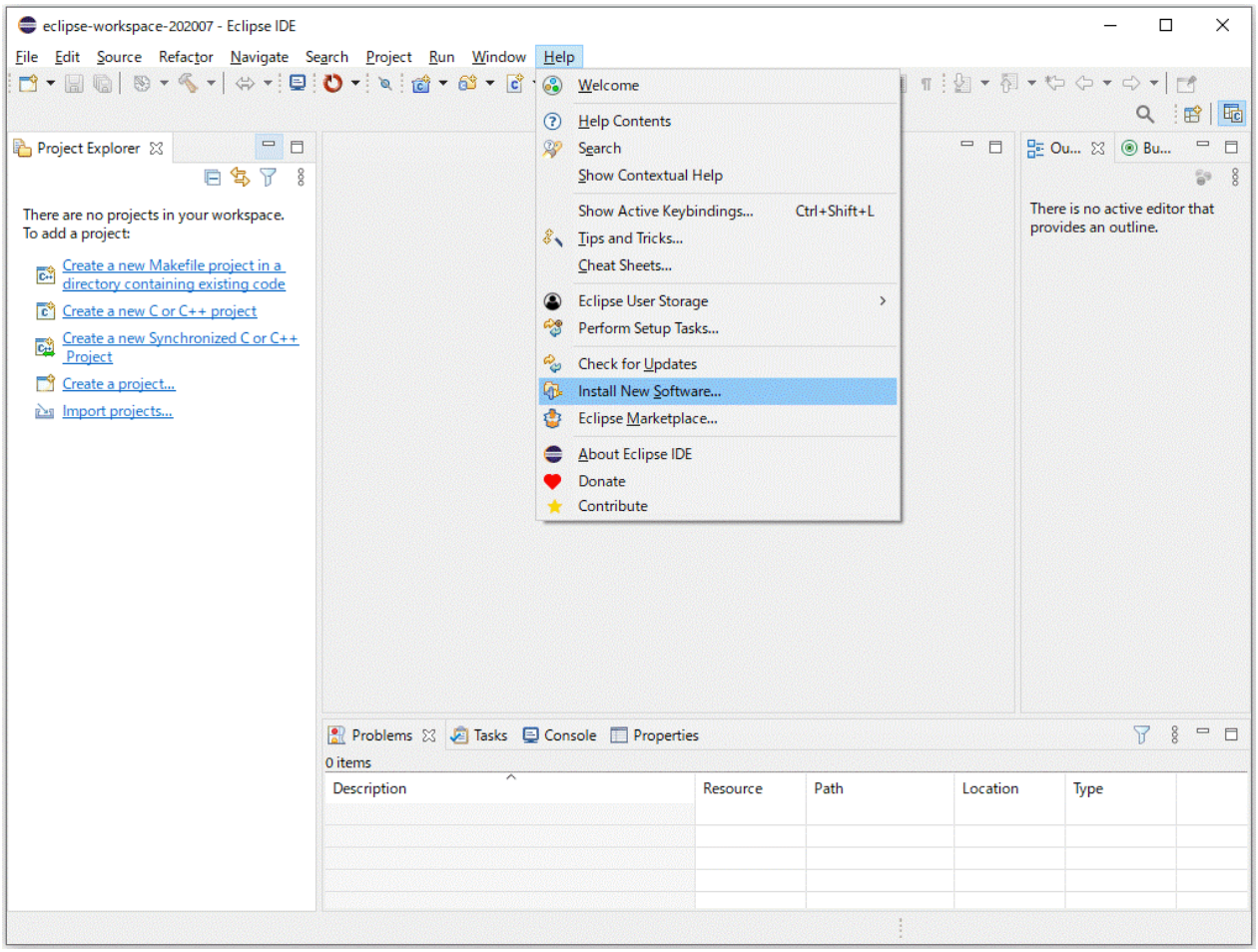
3. Expand [Install/Update] from the left pane in the [Preferences] window and select [Available Software Sites]. Uncheck all checkboxes under [Name] and click the [Apply and Close] button.



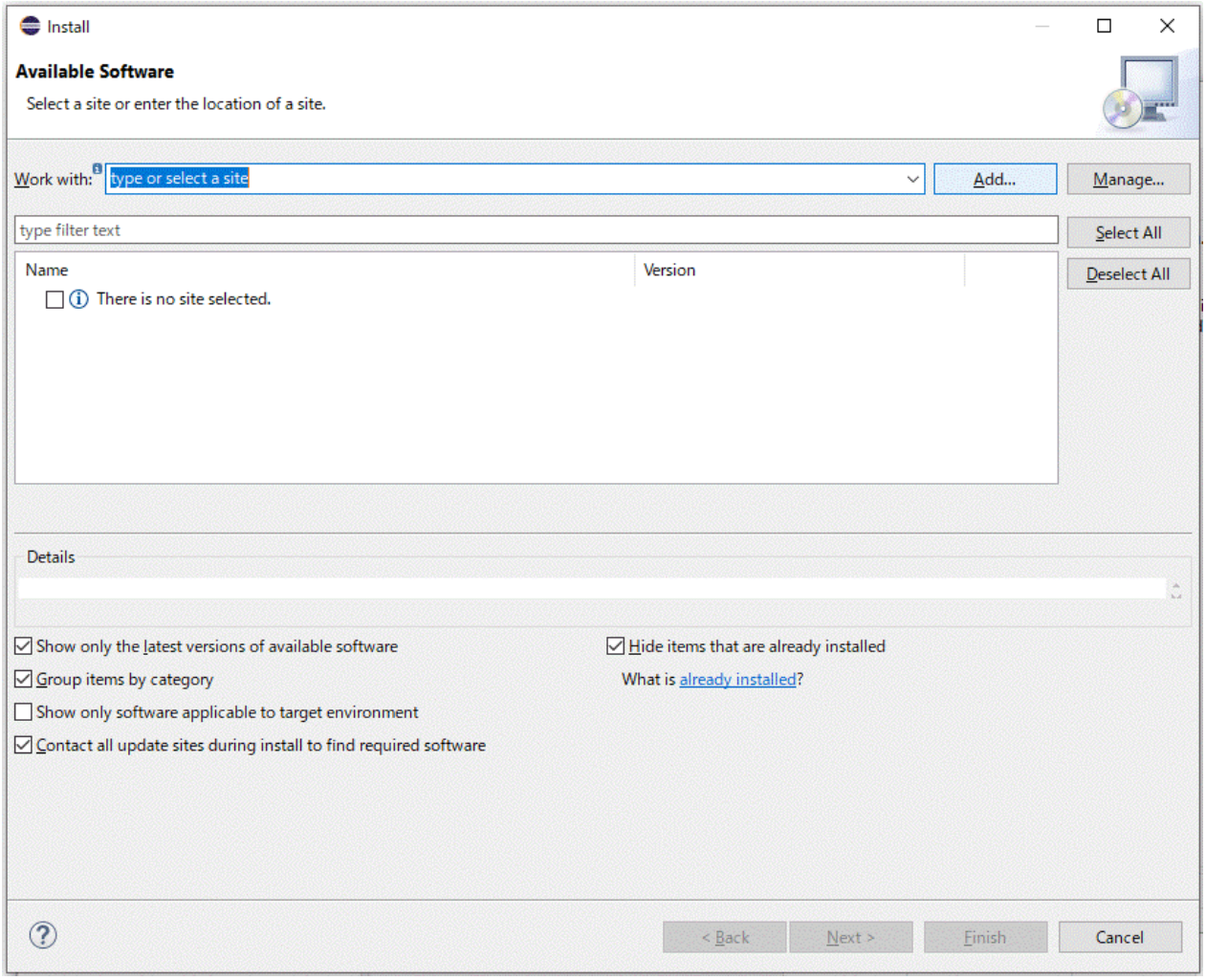
Note

Do not apply updates to the software installed in Eclipse. Also, do not install new software in Eclipse except the one described in this section.

4. Click [Help] - [Install New Software] on the menu bar.



5. Click the [Add] button in the [Install] window.



6. Set the necessary information in the [Add Repository] window, and click the [Add] button.

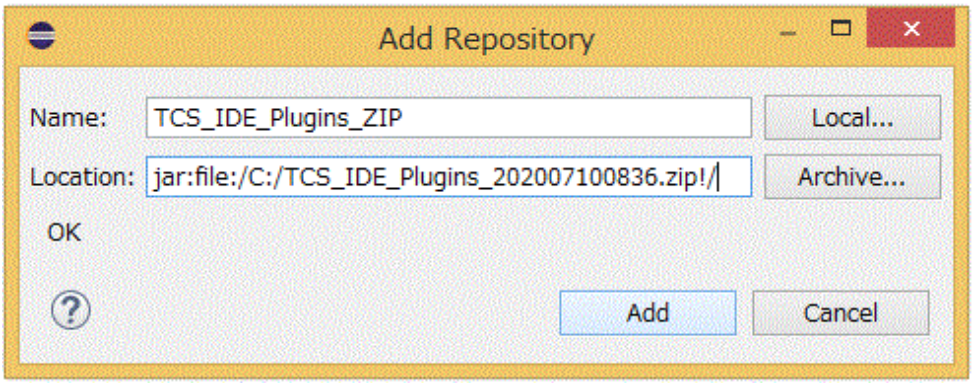
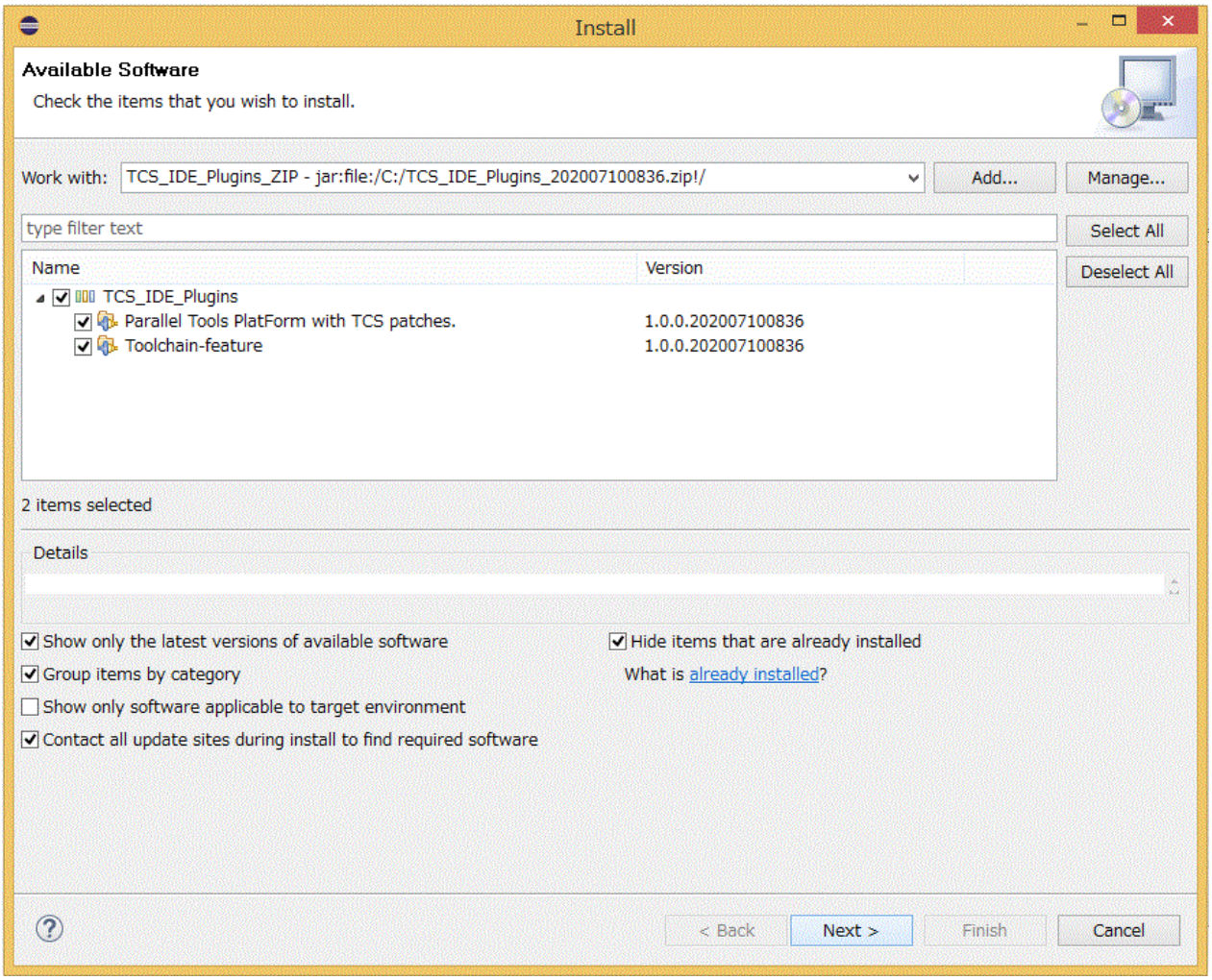


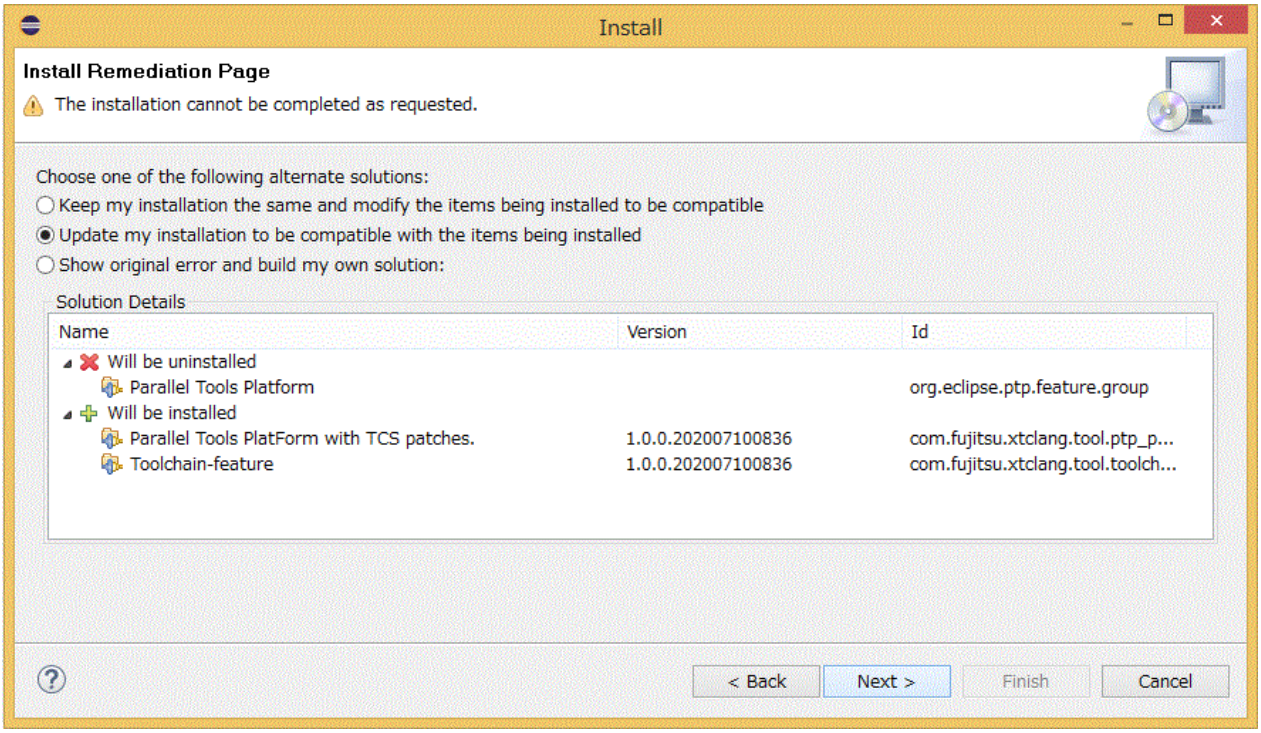
Table 2.4 [Add Repository] Window Setting Details

Item Name	Setting Details
Name:	Specify an arbitrary name.
Location:	Click the [Archive] button, and select the installation package TCS_IDE_Plugins_YYYYMMDDhhmm.zip, which is located in a local directory. The installation package is stored in the "patches" directory of files for Fujitsu extended functions.

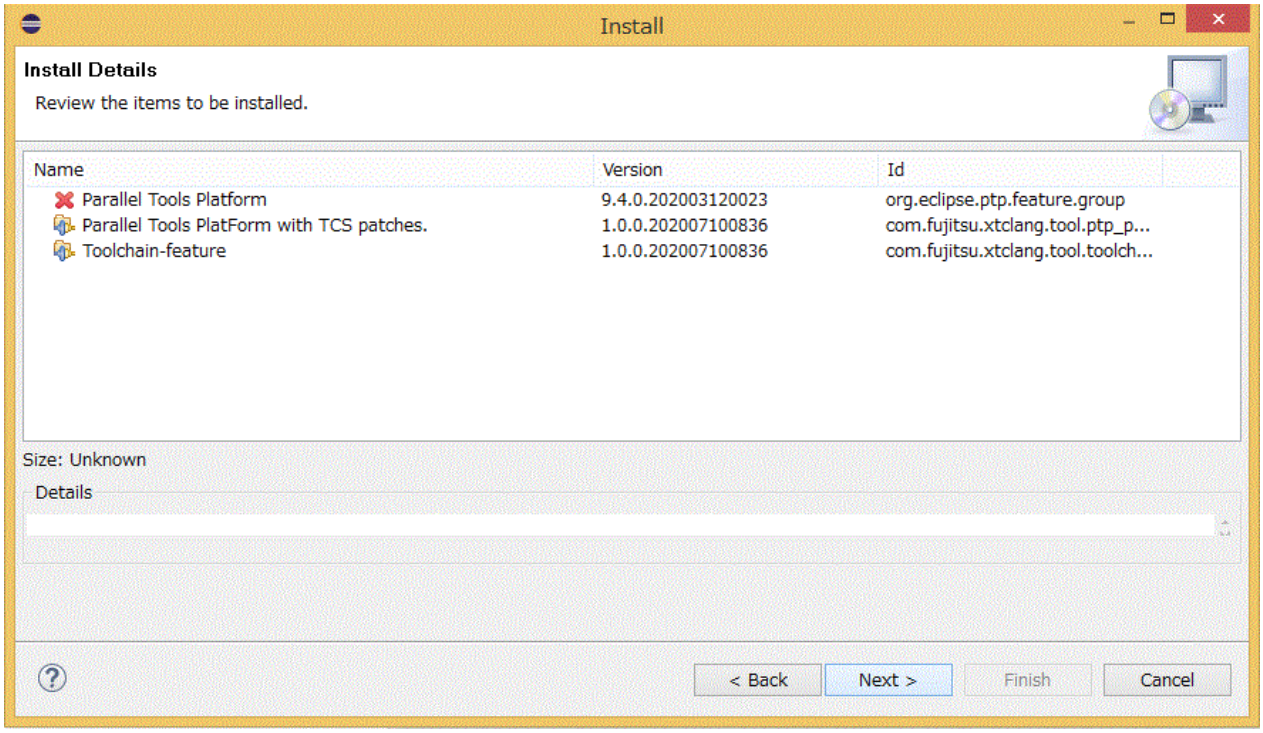
7. The window displays a list of software that can be applied to Eclipse. Click the [Select All] button to select all, and click the [Next] button.



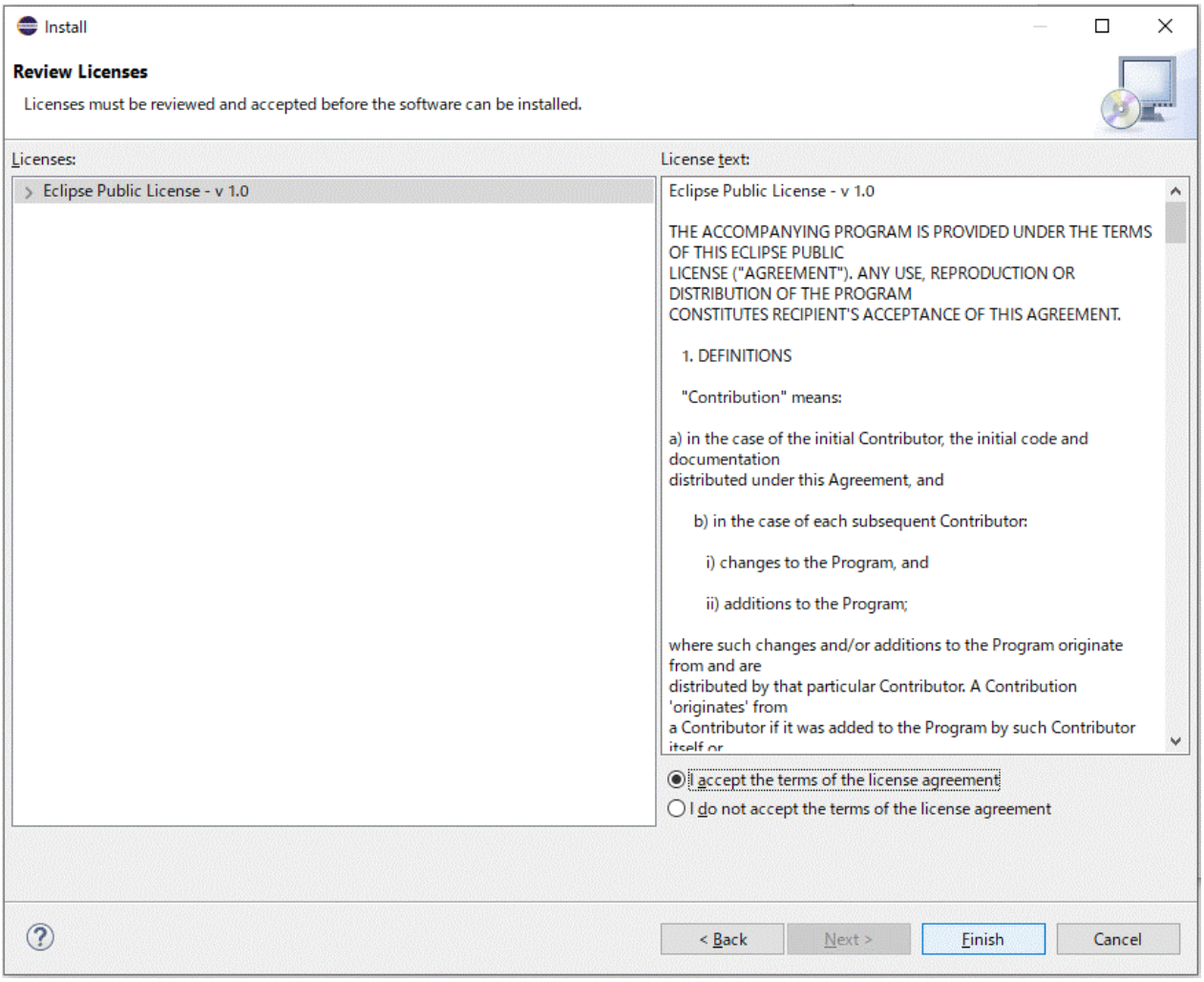
8. Select [Update my installation to be compatible with the items being installed], and click [Next].



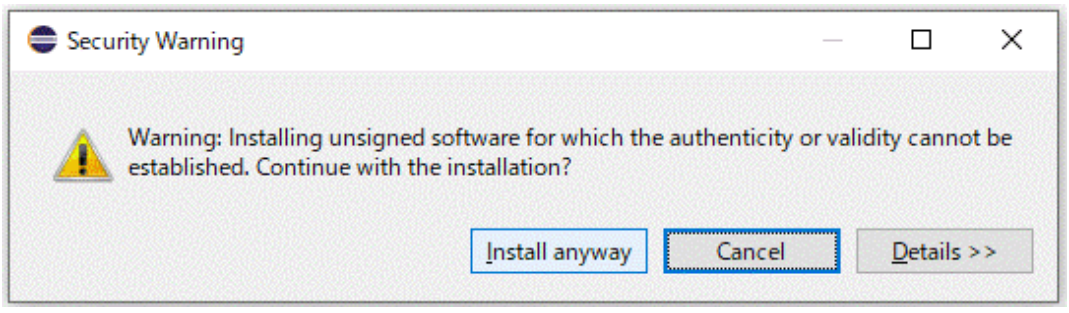
9. The window displays the software to be installed. Confirm that "Parallel Tools PlatForm with TCS patches" and "Toolchain-feature" are to be installed, and click the [Next] button.



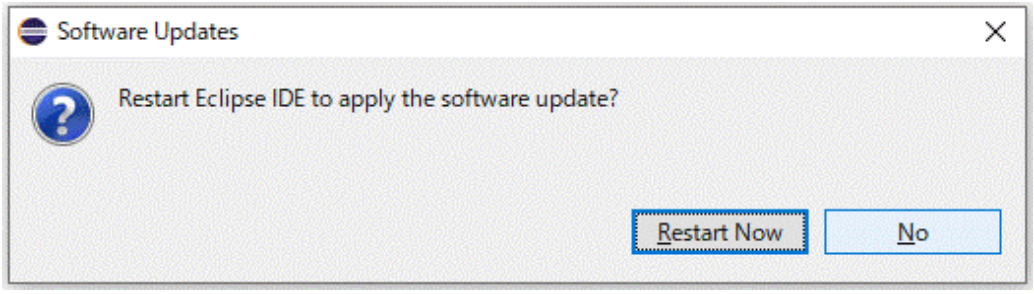
10. Select [I accept the terms of the license agreement], and click the [Finish] button.



11. The security warning screen appears. Click the [Install anyway] button.



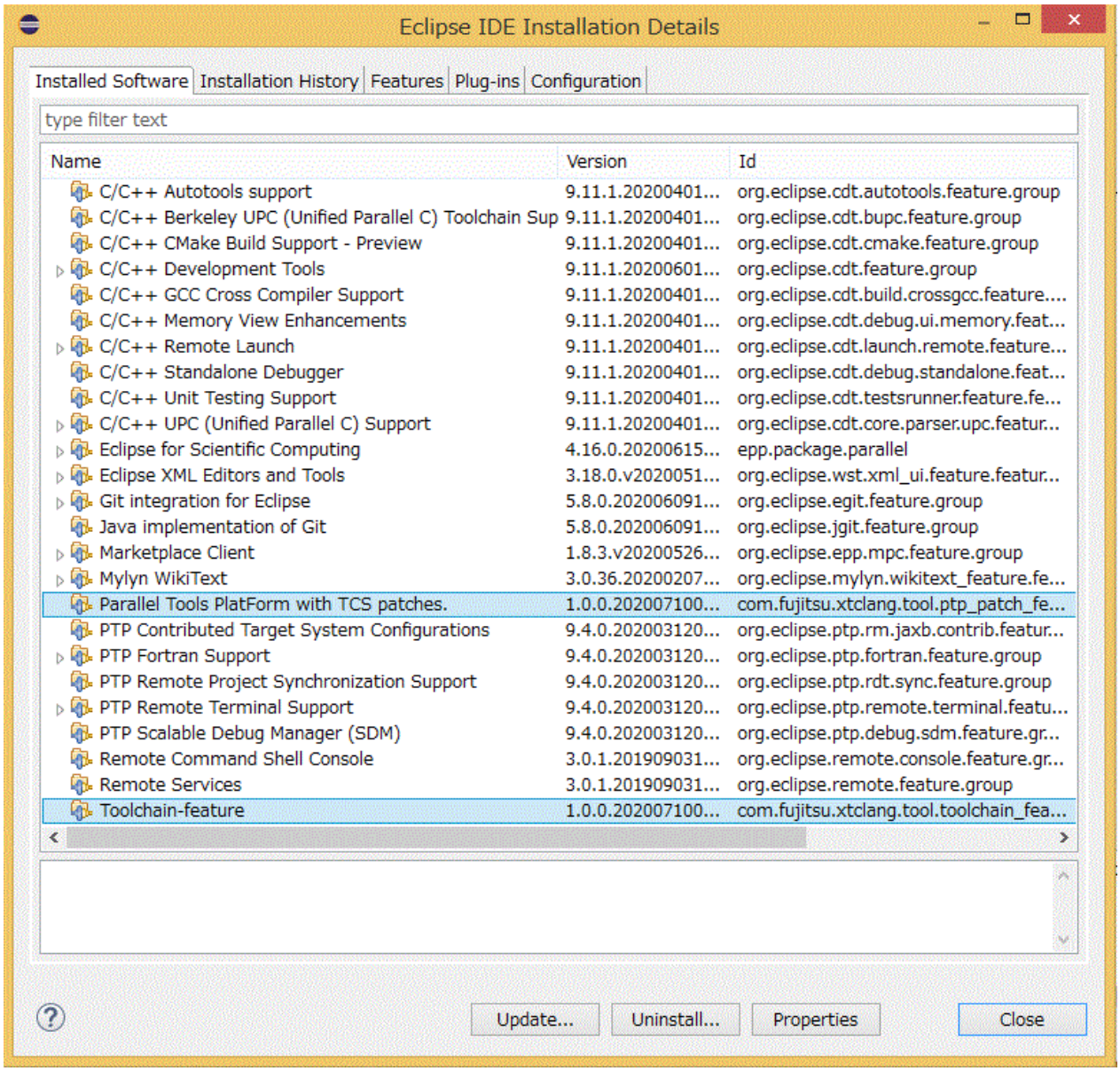
12. You are asked whether to restart the IDE. Click the [No] button. Then click the X button in the upper right corner of Eclipse to exit Eclipse.



13. Start Eclipse by using "eclipse -clean" from the terminal or command prompt. After the start, confirm that the software is correctly installed in Eclipse. Click [Help] - [About Eclipse IDE] - [Installation Details] on the menu bar. Confirm that the list on the [Installed Software] tab includes the two specified software names.

Note

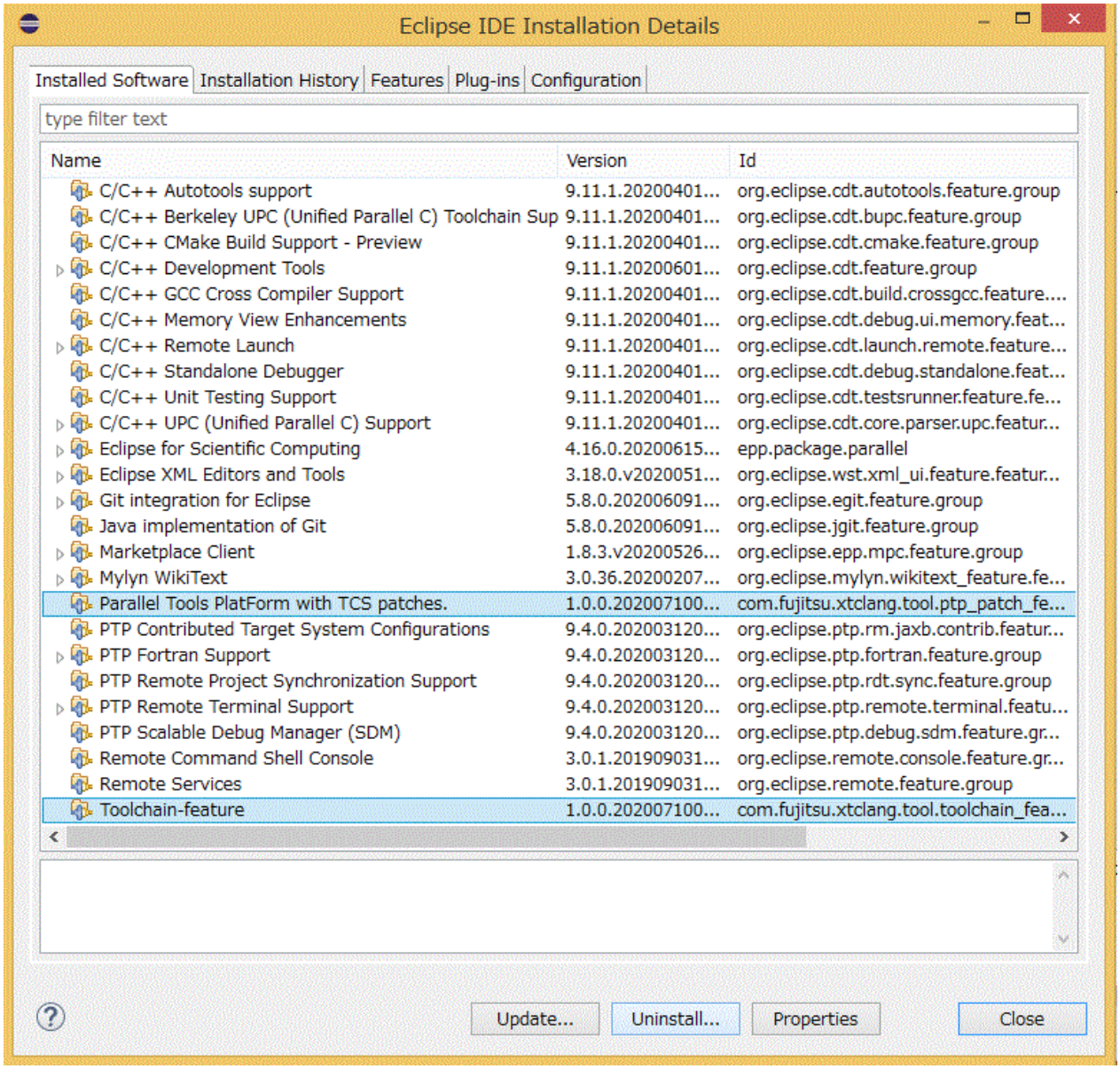
Use "eclipse -clean" on the directory where "eclipse.exe" exists.



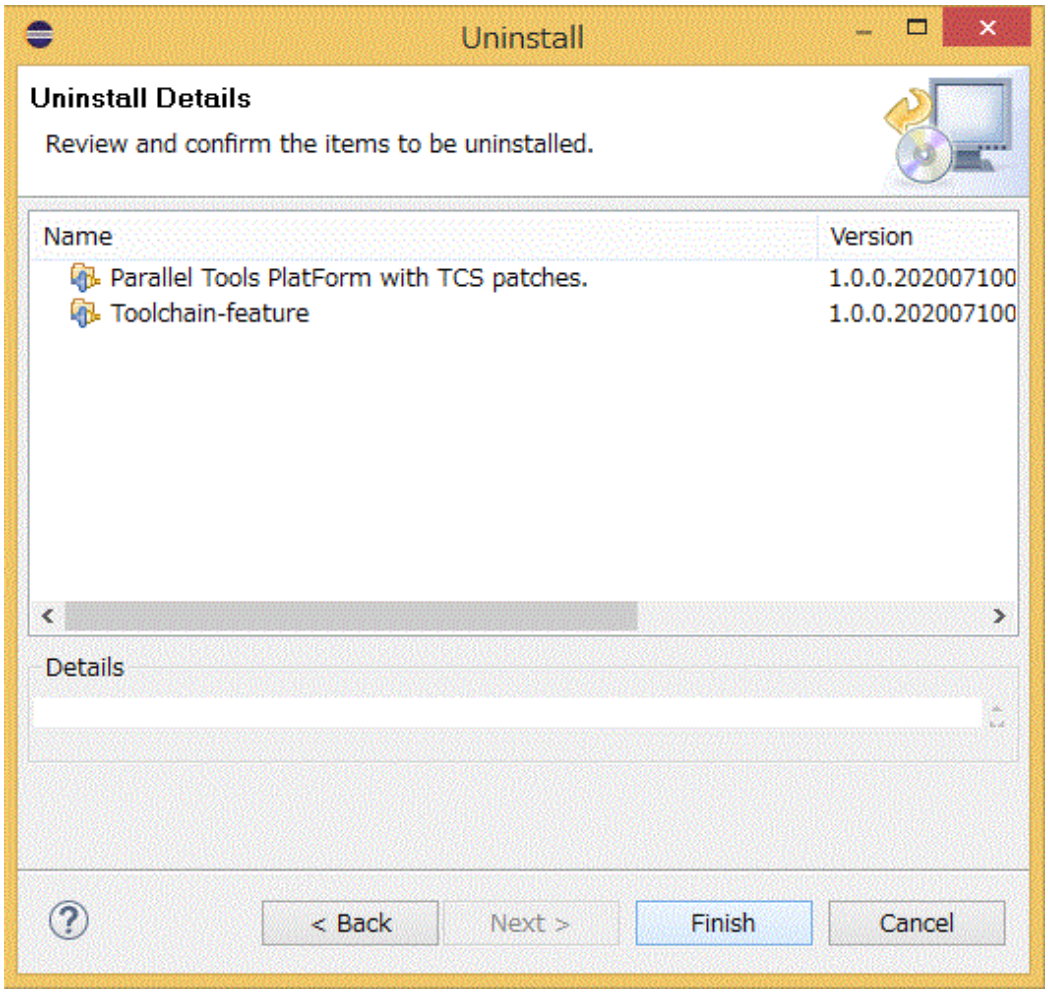
2.3.5 Uninstalling the Applied Installation Package

To uninstall the installation package applied to Eclipse, perform the following procedure.

1. Click [Help] - [About Eclipse IDE] - [Installation Details] on the menu bar. From the list on the [Installed Software] tab, select the software to delete, and click the [Uninstall] button.



2. The window displays the software to be uninstalled. After confirmation, click the [Finish] button. Then click the X button in the upper right corner of Eclipse to exit Eclipse. Finally, to complete the uninstallation, start Eclipse by using "eclipse -clean" from the terminal or command prompt.



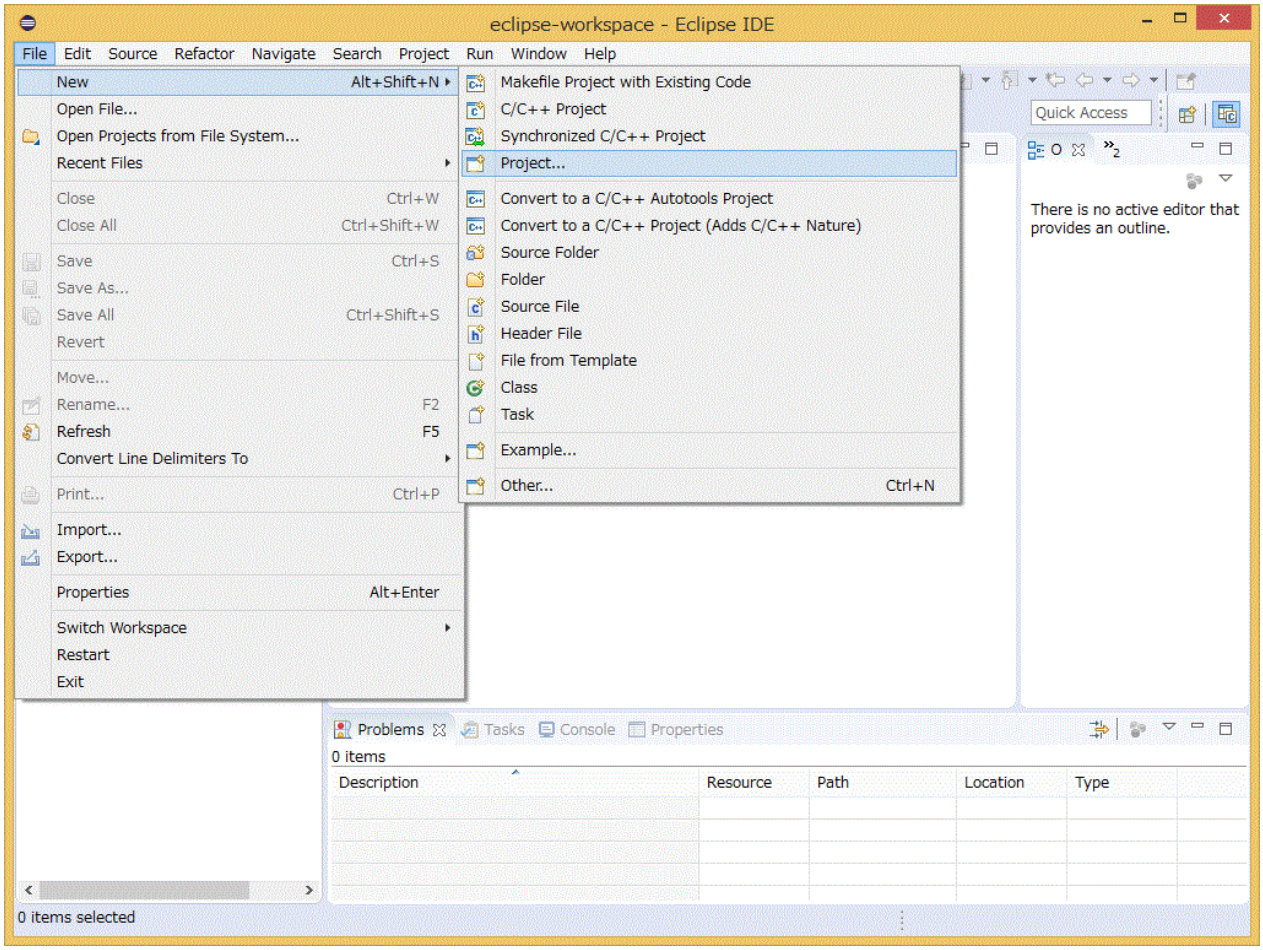
Chapter 3 Basic Usage of Eclipse

This chapter describes basic usage of Eclipse by providing procedures from creating a new project to building a program.

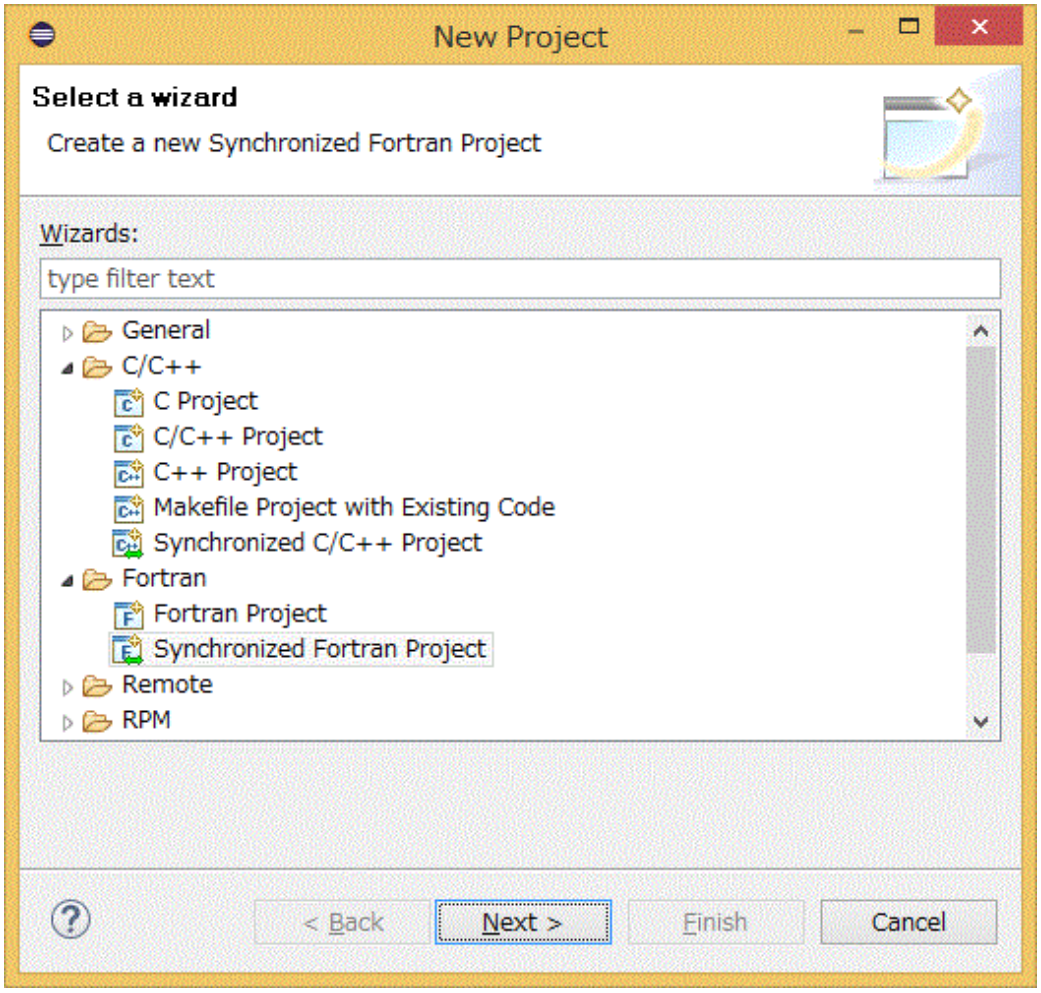
3.1 Creating a Project

This section describes the procedure for creating a new project.

1. Click [File] - [New] - [Project...] on the menu bar.



2. Expand [C/C++] or [Fortran] from the [New Project] window, and select the type of the project to create. Select either [Synchronized C/C++ Project] or [Synchronized Fortran Project] according to the language to use. After selection, click the [Next >] button.



3. Make project-related settings in the [New Synchronized Fortran Project] window. After completing all the settings, click [Finish] button. The following table shows setting details.

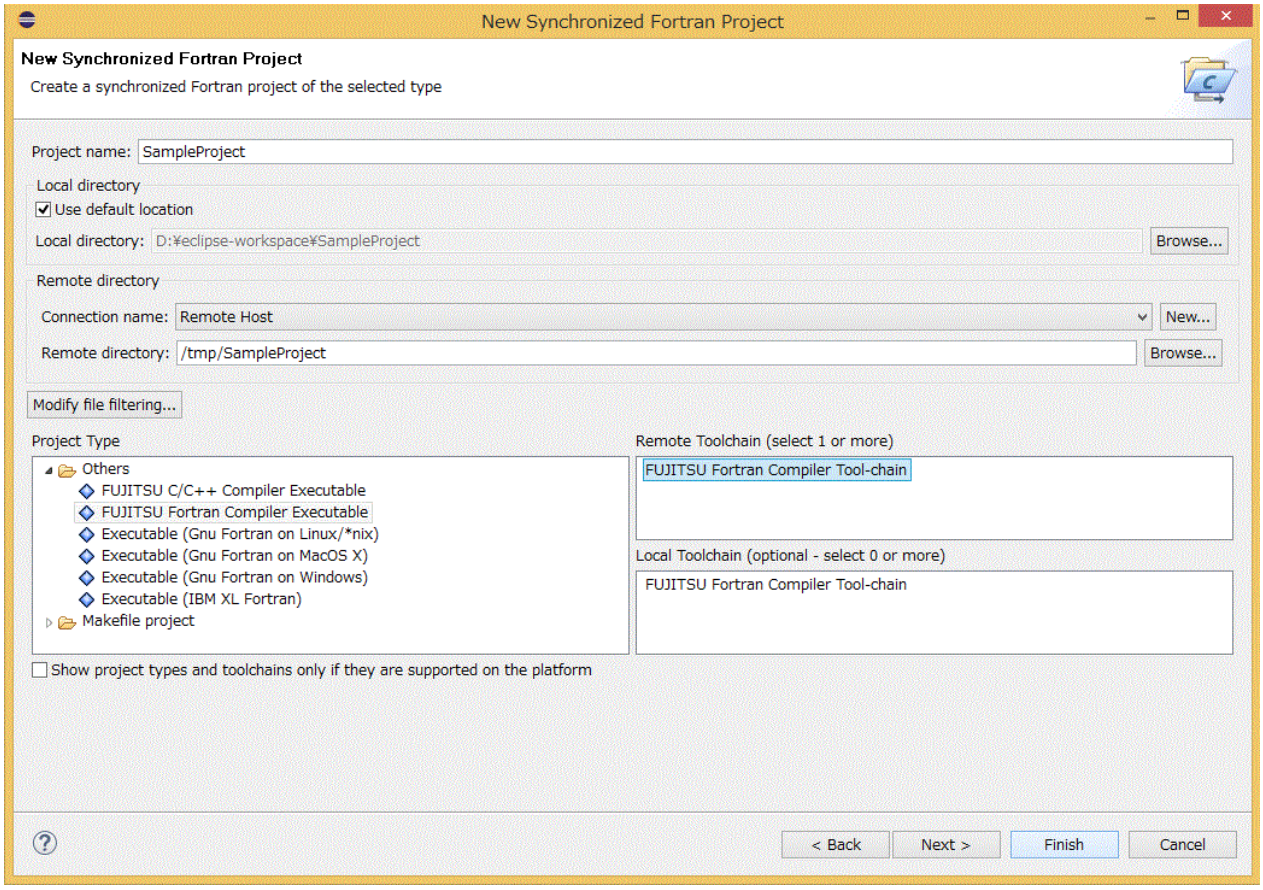
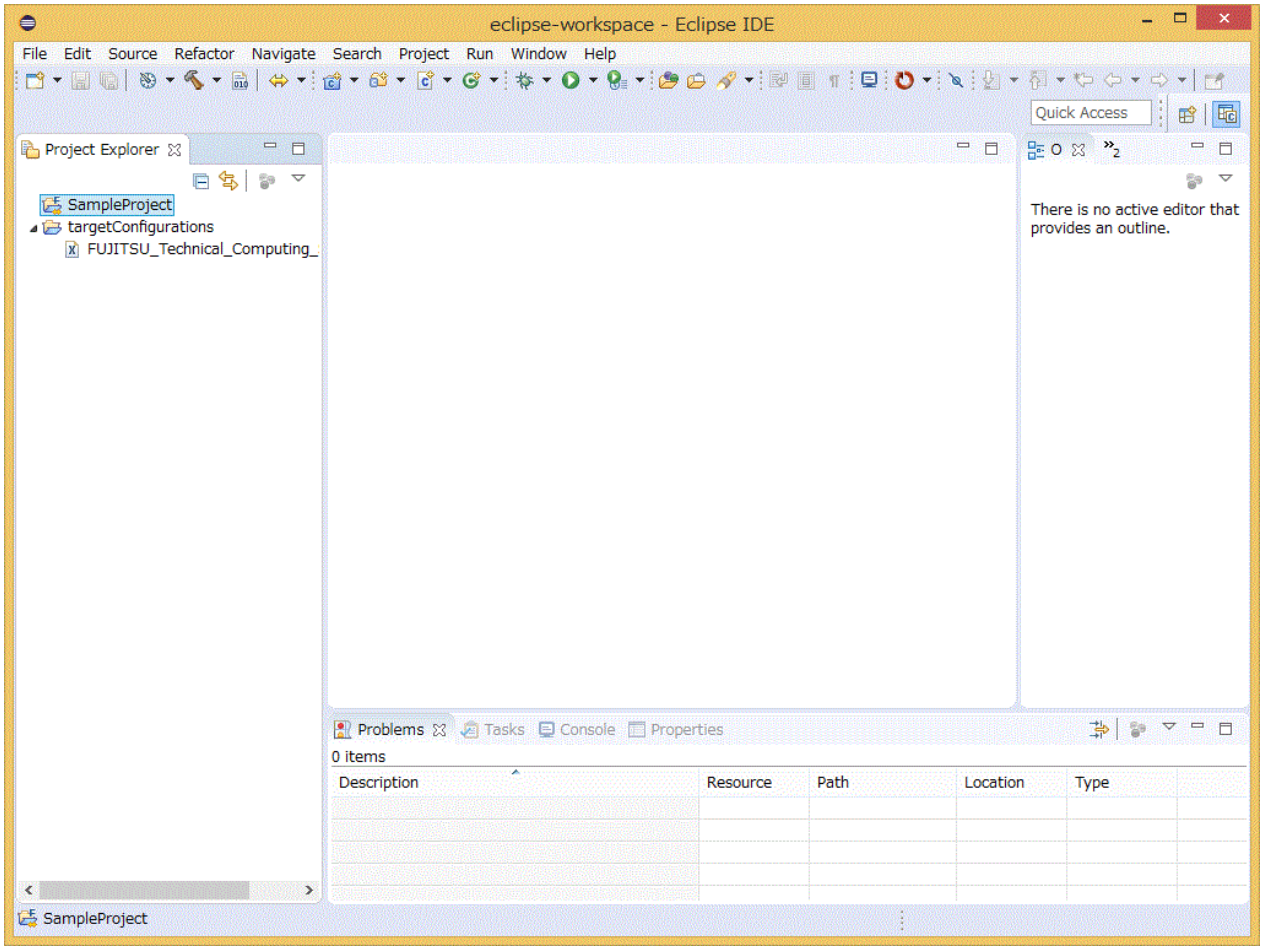


Table 3.1 Setting Details of Synchronized Project

Item Name	Setting Details
Project name:	Specify a project name. Specify an arbitrary name that is unique in the workspace. The name specified here will be the executable file name at build.
Local directory:	Specifies a directory on the client machine where the project is to be stored. If [Use default location] is checked, "workspace full path" + "directory whose name is specified in [Project name:]" is automatically specified. If you want to store the project in a different directory, specify a storage directory path in [Local directory:].
Remote directory:	Specifies a directory on the login node where the project is to be stored. In [Connection name:], set [Connection name:] created in "2.3.1 Connecting to the Login Node (Remote System)". In [Remote directory:], specify the directory path you want to save. You can specify an arbitrary directory name that does not match any directory name used in another project.
Project Type	Select a project type. To use FUJITSU compiler, expand [Others] and select [FUJITSU C/C++ Compiler Executable] or [FUJITSU Fortran Compiler Executable] according to the language to use.
Remote Toolchain	Select a toolchain from the following, according to the language you want to use: FUJITSU C Compiler Tool-chain FUJITSU C++ Compiler Tool-chain FUJITSU Fortran Compiler Tool-chain
Local Toolchain	Not used.

4. The created project is added to the [Project Explorer] view.



3.2 Adding a Source File

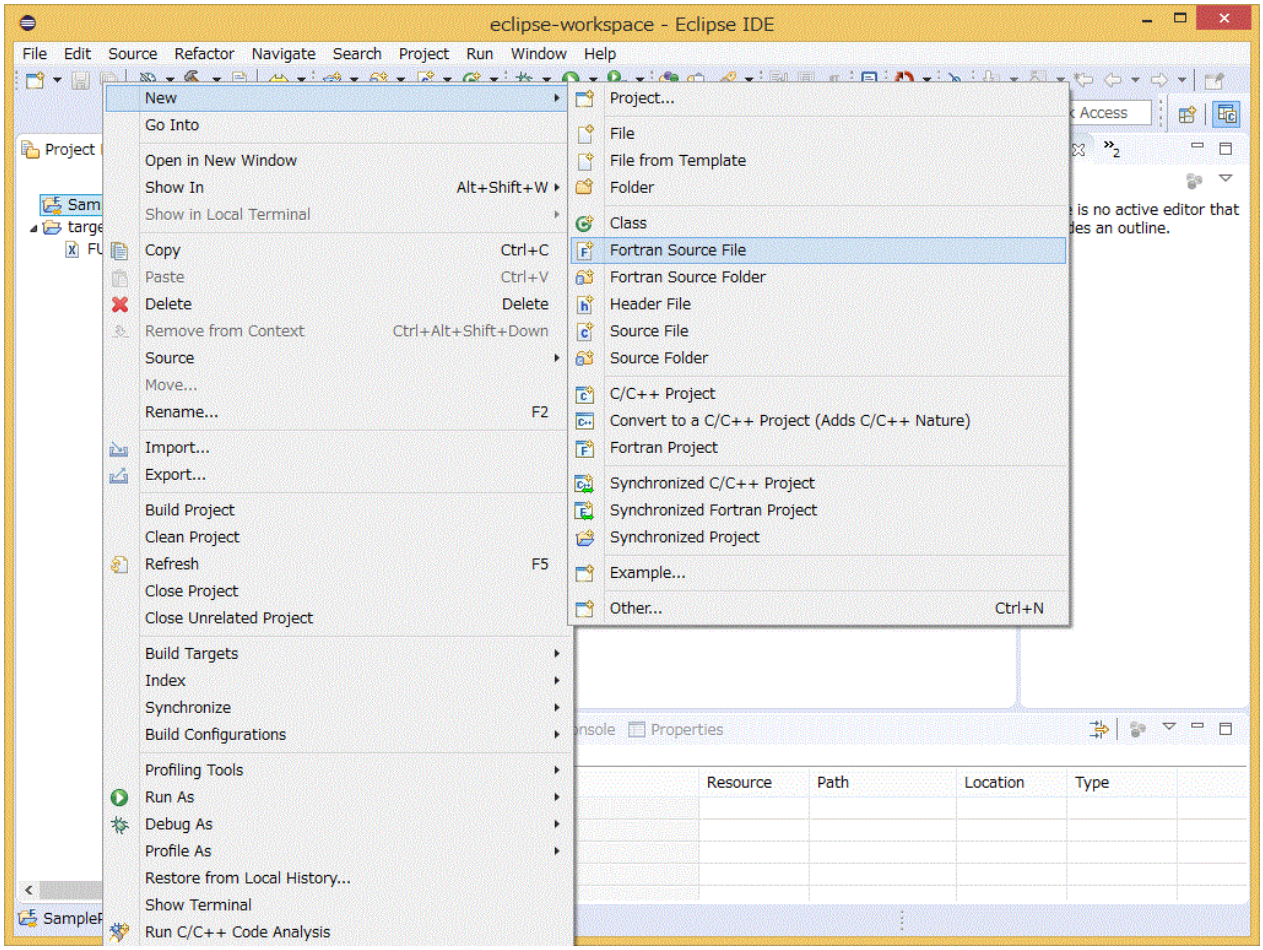
Add a source file to the project.

- When creating a new source file, see "3.2.1 Creating a New Source File".
- When importing existing source files, see "3.2.2 Importing Source Files".

3.2.1 Creating a New Source File

This section describes the procedure for creating a new source file for a project.

1. Select and right-click the project created in "3.1 Creating a Project". Select the [New] submenu from the displayed menu, and then select the type of file you want to add. This time, select [Fortran Source File].



Note

The file types that can be selected from the [New] submenu vary depending on the project type.

2. When you have selected [Fortran Source File], the settings on the screen are as shown below. Specify the necessary information in the [New Fortran Source File] window, and click the [Finish] button. The following table shows setting details.

Note

If you have selected another file type, interpret this content accordingly.

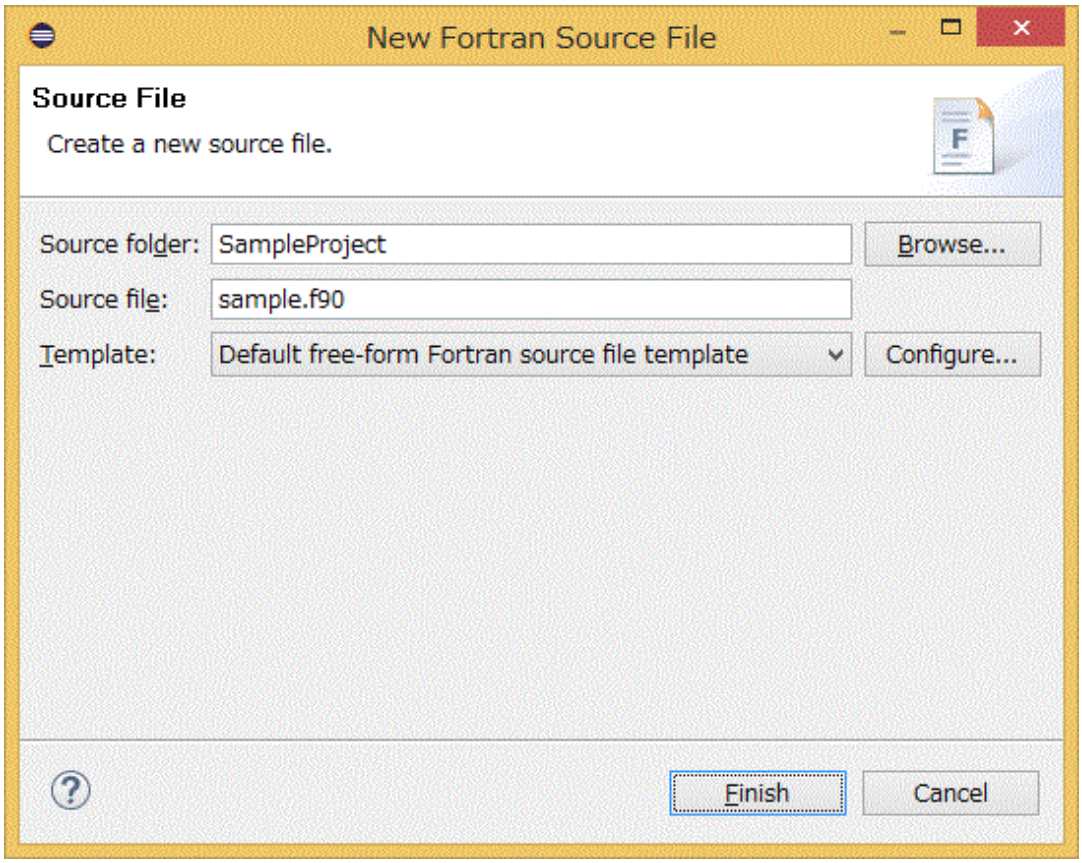
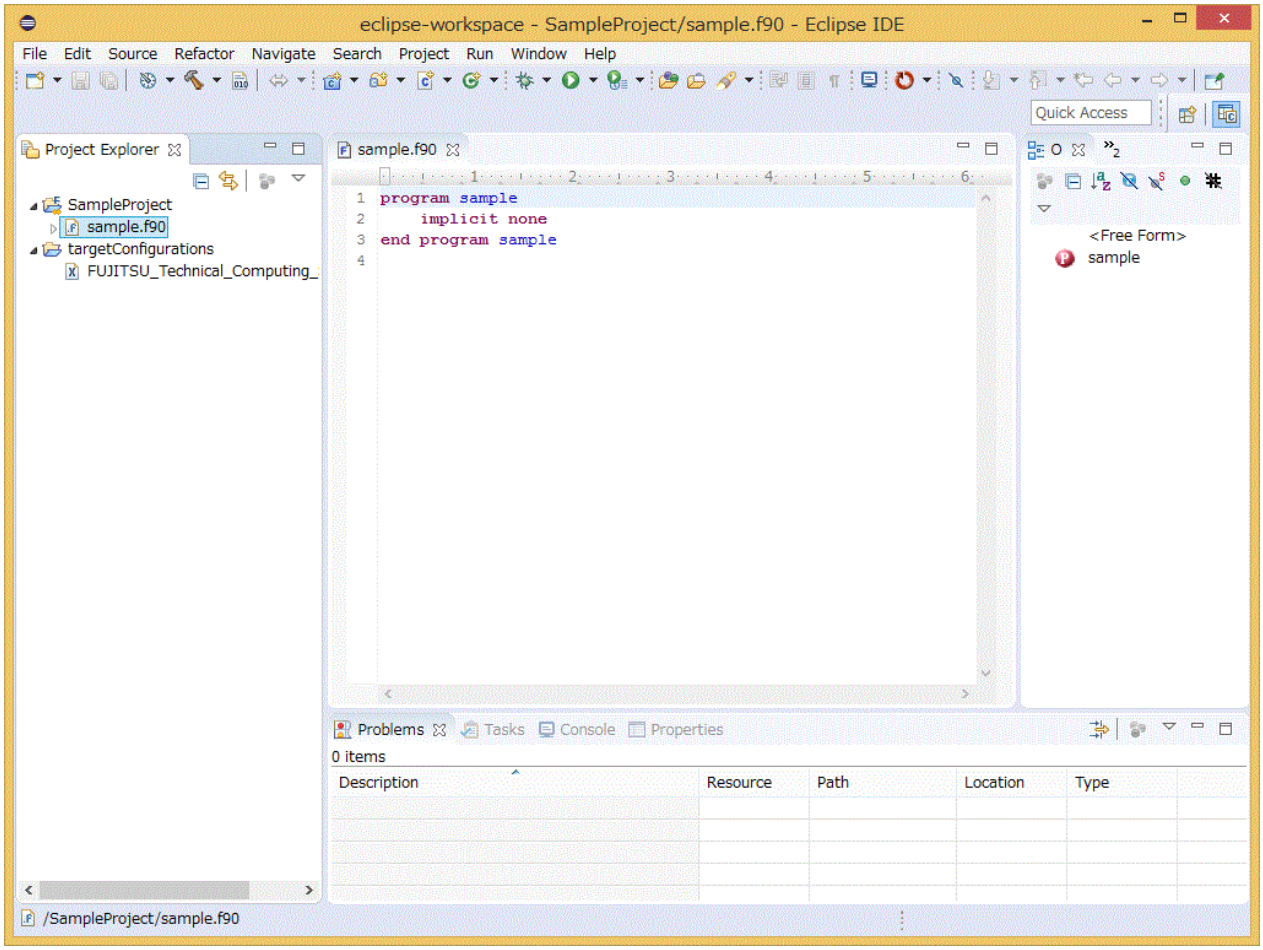


Table 3.2 Settings when creating a new source code

Item Name	Setting Details
Source folder:	Specifies the location to store the source file.
Source file:	Specify a name for the source file.
Template:	Select a template for the source file.

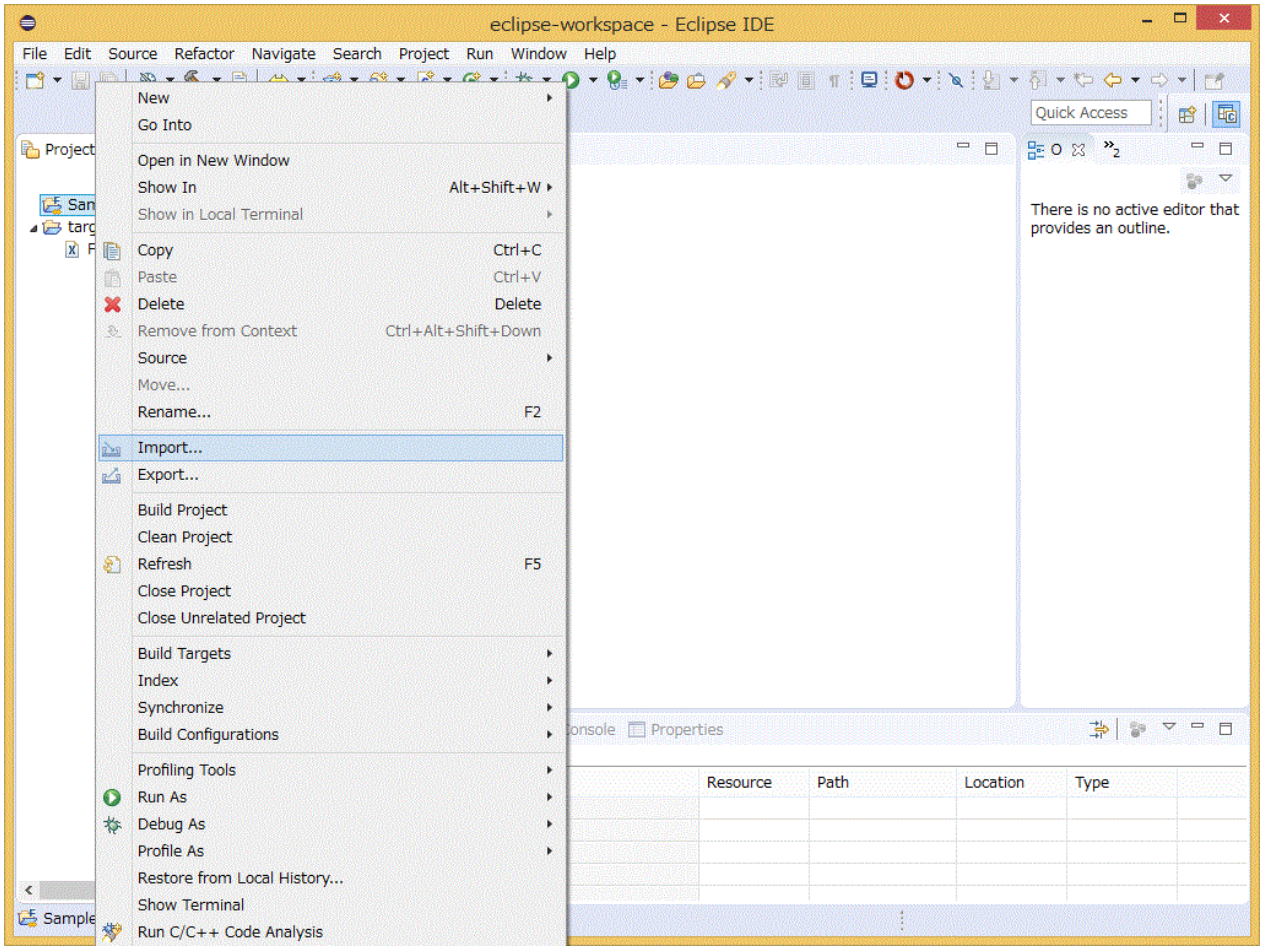
3. The file is added to the project.



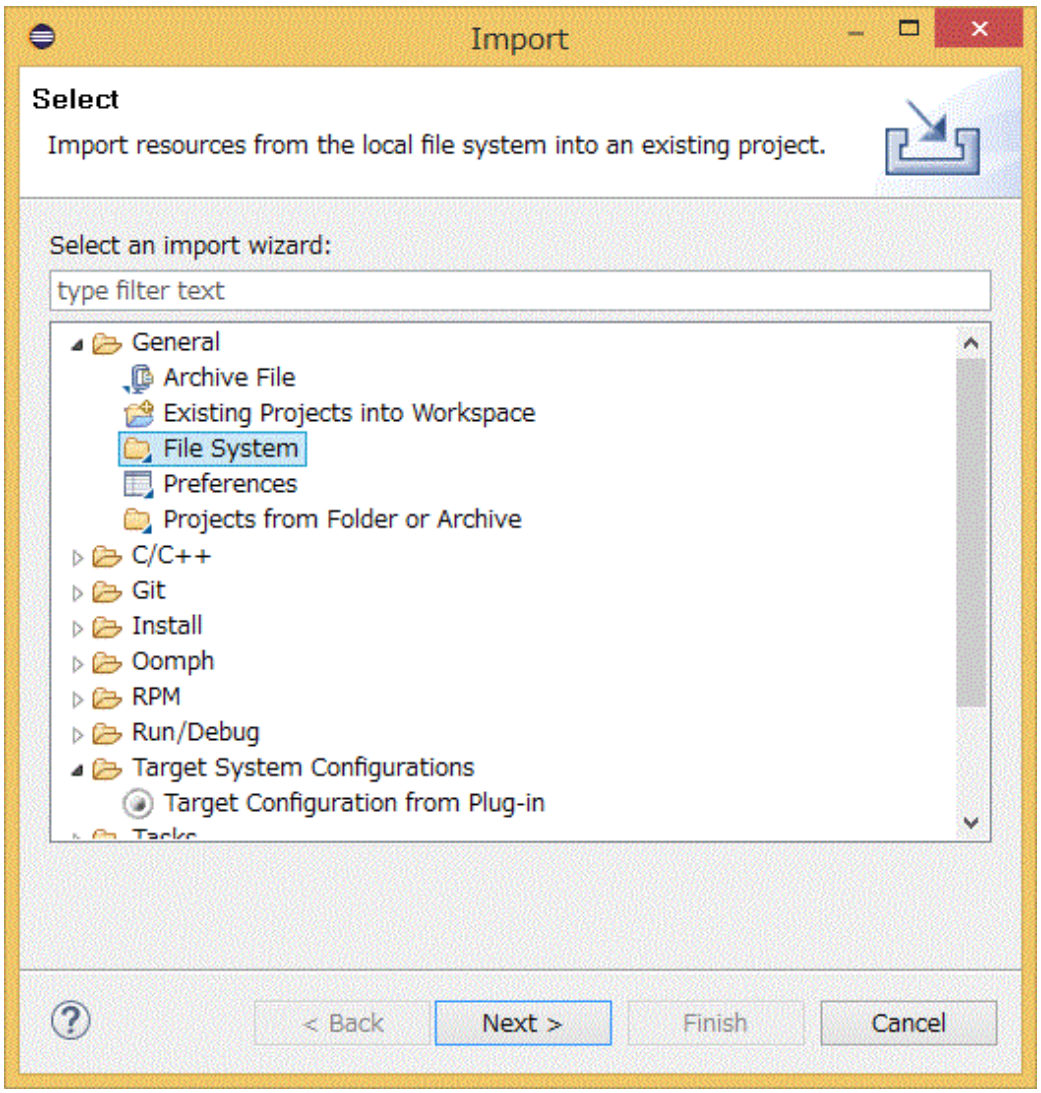
3.2.2 Importing Source Files

This section describes the procedure for importing source files to a project.

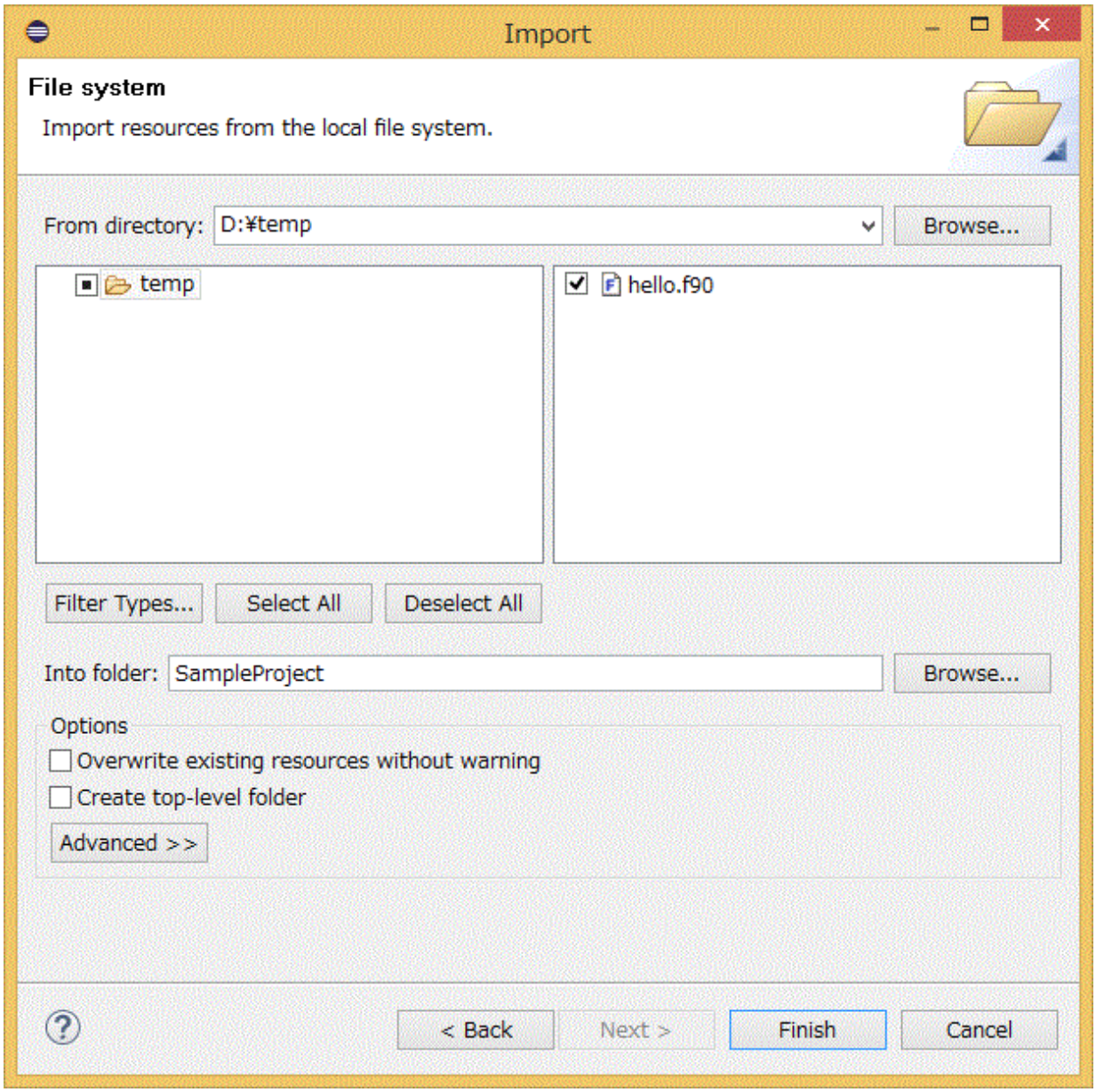
1. Select and right-click the project created in "3.1 Creating a Project". Select [Import...] from the displayed menu.



2. Select [General] - [File System] under [Select an import wizard] in the [Import] window, and click [Next >] button.



3. Click the [Browse...] button in [From directory:], and specify the directory that contains the files you want to import. The files in the directory appear in the box below. Check all the files you want to import. Confirm that the project created in "3.1 Creating a Project" is specified in [Into folder:]. After completing all settings, click the [Finish] button.



Note

If you use the MPI Barrier Analysis function, import the following header located on the login node into your project.

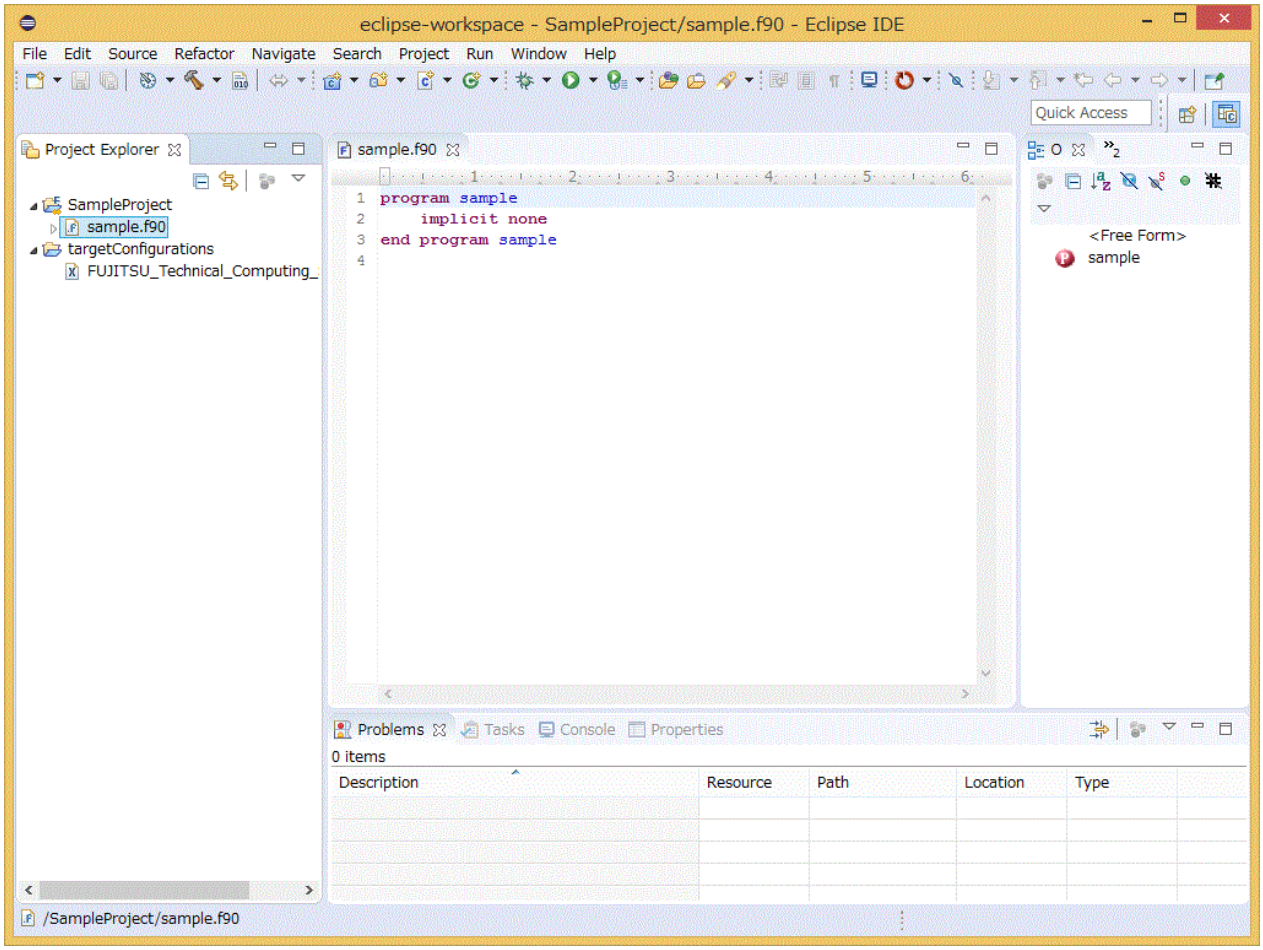
```
/installation_path/include/mpi/fujitsu/mpi.h
```

For details on "*installation_path*", contact the system administrator.

3.3 Editing a Source File

This section describes the procedure for editing a source file.

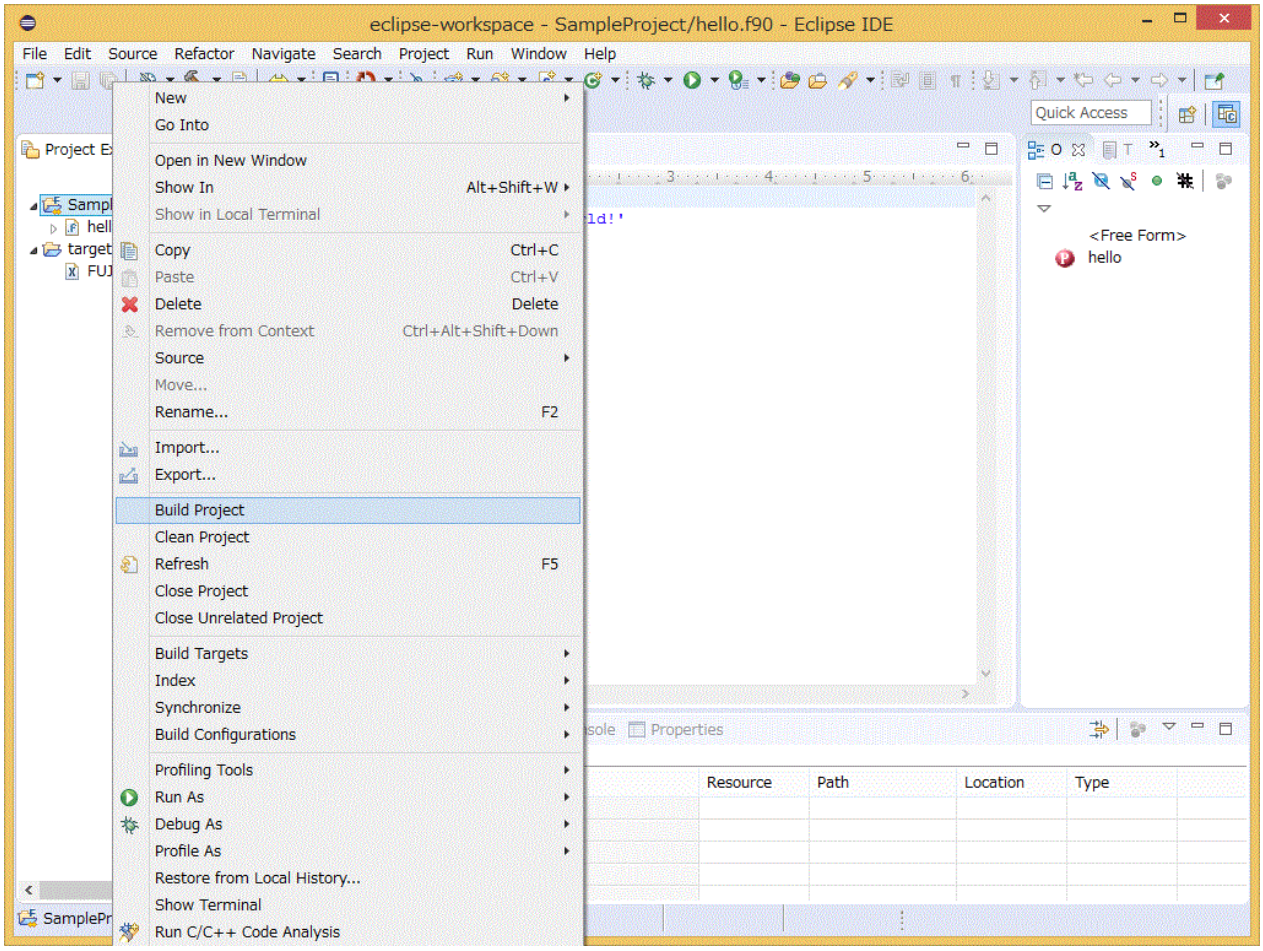
1. From the [Project Explorer] view, select and double-click the file you want to edit. Edit this file when it has opened. To save the edited file, click [File] - [Save] on the menu bar.



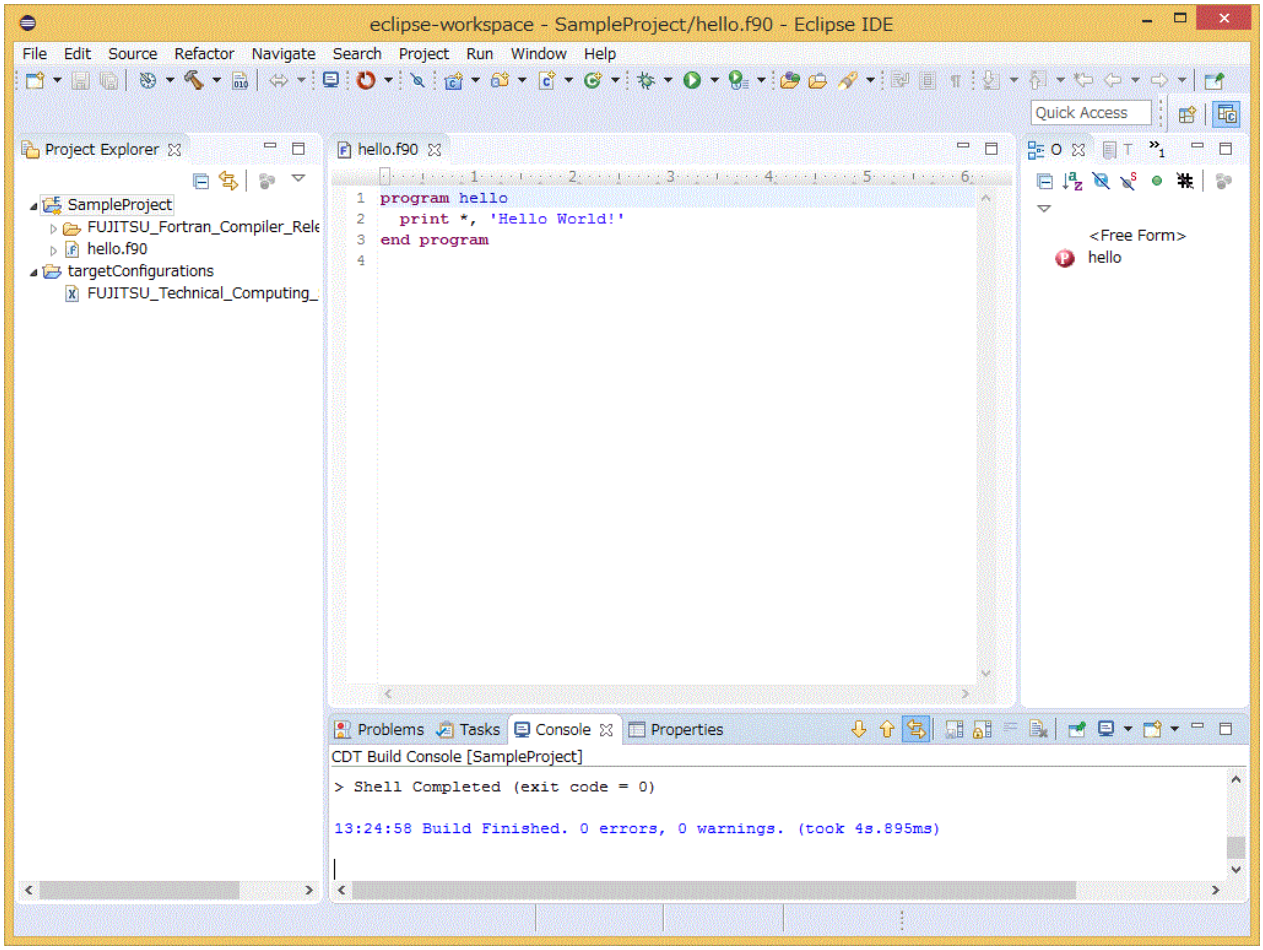
3.4 Building a Project

This section describes the procedure for building a project.

1. Select and right-click the project created in "3.1 Creating a Project". Select [Build Project] from the displayed menu.



2. Execute build. The [Console] view at the bottom right outputs a build log, and the generated files are added to the [Project Explorer] view.



Note

When running [Build Project], build may occur before a file or directory is created in the login node, resulting in a build error. When this occurs, an error message "No such file or directory" is printed. If you receive an error message "No such file or directory", select [Build Project] again.

If you change compiler options before the build, the build results may not reflect the changed options. Check the [Console] view. If the changed options are not reflected, clean the project before executing the build again.

Even though an error occurs, command issues are not terminated. Therefore, multiple errors can occur.

Information

To clean a project, select and right-click the project created in "3.1 Creating a Project", and select [Clean Project] from the displayed menu.

Chapter 4 Using Fujitsu Extended Functions

This chapter describes procedures for using Fujitsu extended functions.

Note

To use Fujitsu extended functions, use the workspace created in "2.3.1 Connecting to the Login Node (Remote System)".

4.1 Build Using Fujitsu Compiler

The following describes the build procedure that uses the Fujitsu compiler. For details on the compiler, see "Fortran User's Guide", "C User's Guide", or "C++ User's Guide".

4.1.1 Setting Environment Variables on the Login Node

The integrated development environment uses compilers that reference environment variables at login to the login node. Therefore, add the environment variables used by the compilers to the login shell of the login node. The following example shows what to add to use bash.

```
export LANG_HOME=installation_path
export PATH="${LANG_HOME}/bin:${PATH}"
export LD_LIBRARY_PATH="${LANG_HOME}/lib64:${LD_LIBRARY_PATH}"
```

For details on "*installation_path*", contact the system administrator.

4.1.2 Creating a Project

For the procedure for creating a project, see "3.1 Creating a Project".

4.1.3 Adding a Source File

For the procedure for creating source code, see "3.2 Adding a Source File".

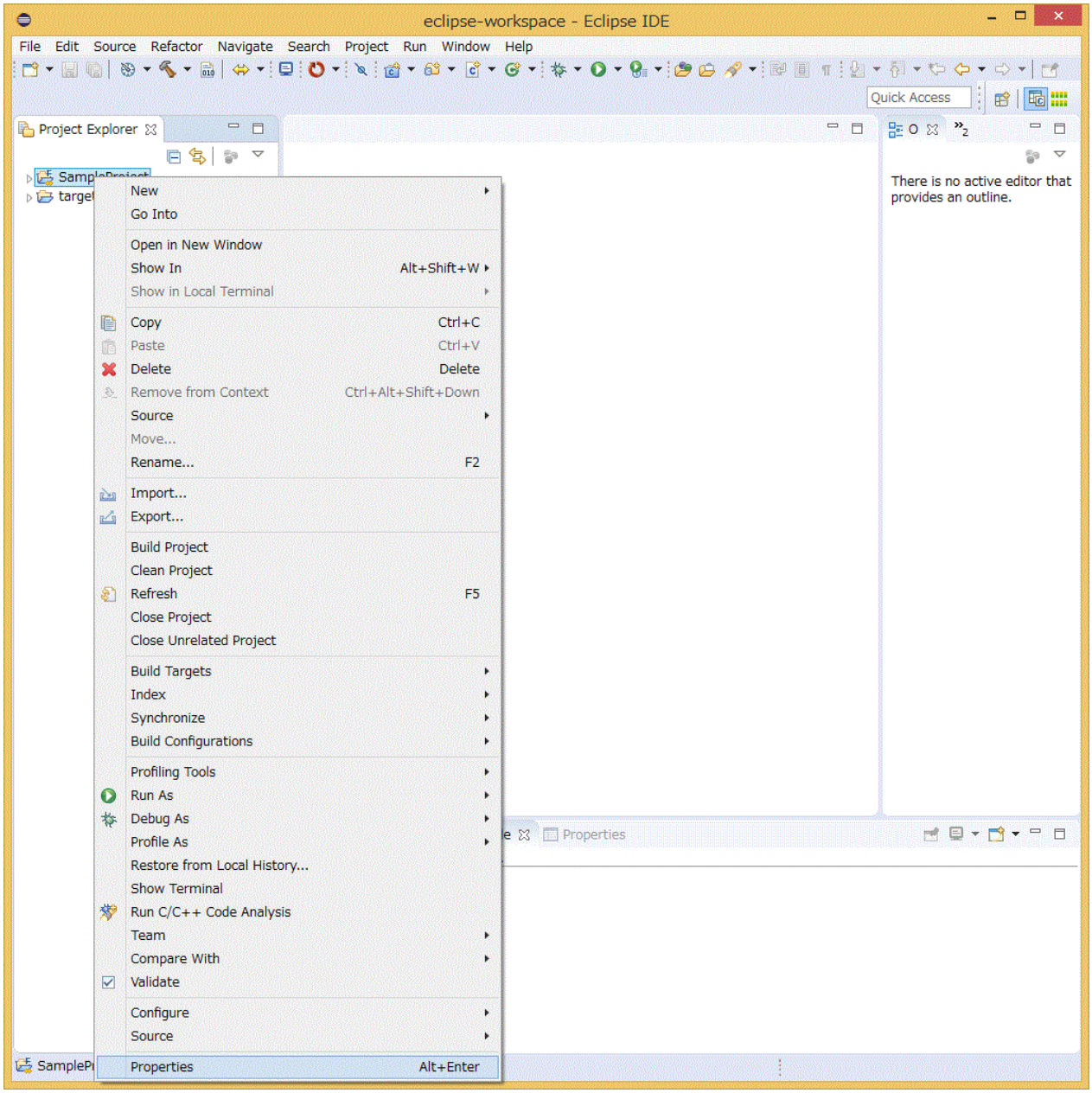
4.1.4 Editing a Source File

For the procedure for editing source code, see "3.3 Editing a Source File".

4.1.5 Specifying Compiler Options

Specify compiler options. For details on the compiler, see "Fortran User's Guide", "C User's Guide", or "C++ User's Guide".

1. Select the project for which you want to specify compiler options, and click [Properties] in the right-click menu.



2. Specify compiler options by referencing "4.1.5.1 Specify Compiler Options (Fortran)", "4.1.5.2 Specify Compiler Options (C)", or "4.1.5.3 Specify Compiler Options (C++)".

4.1.5.1 Specify Compiler Options (Fortran)

Select [Fortran Build] - [Settings]. Make settings for the items in the [Tool Settings] tab. Click the [Apply and Close] button.

Figure 4.1 [Tool Settings] tab - [FUJITSU Fortran Compiler]

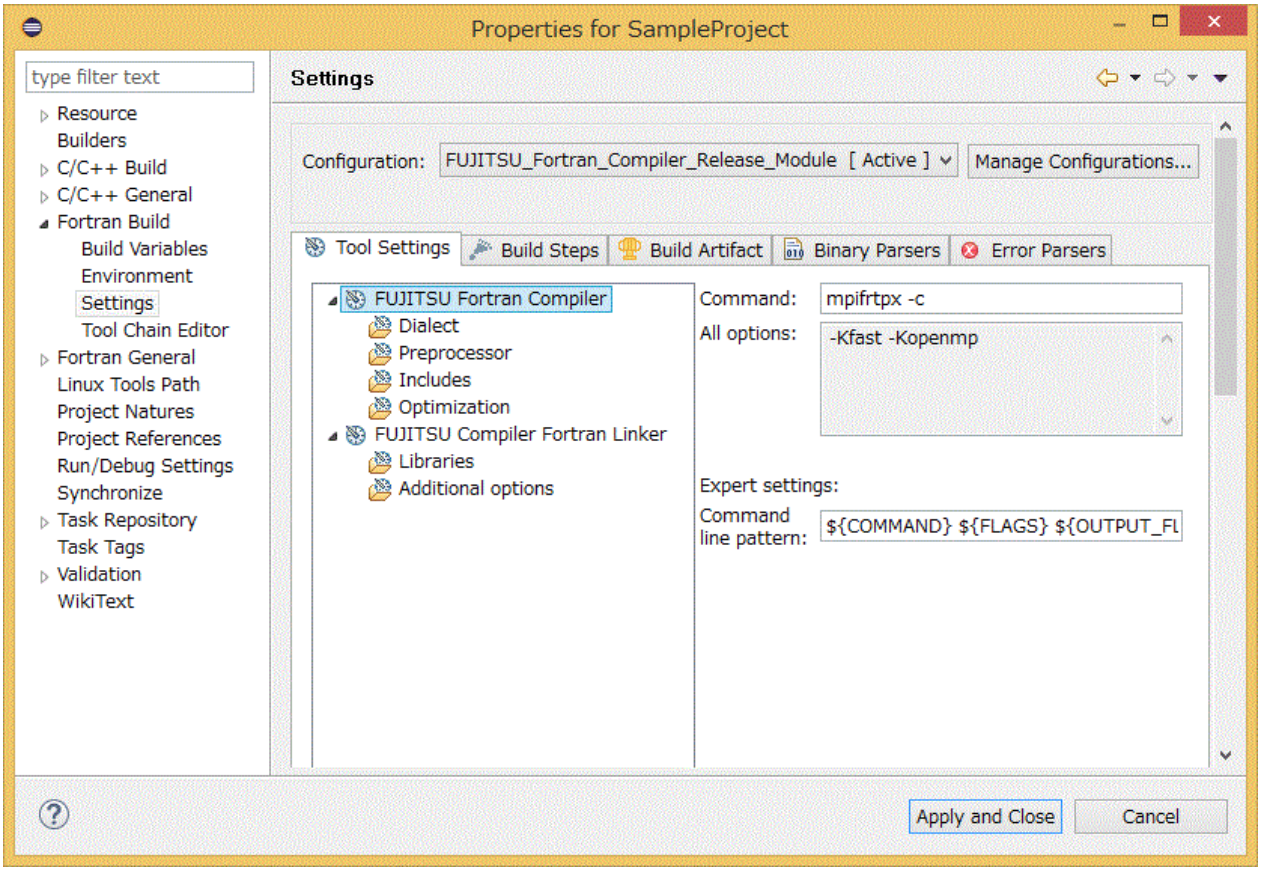


Table 4.1 Settings in [FUJITSU Fortran Compiler]

Item Name	Description
Command:	Adds the command name to be used at the time of compilation. Rewrite if necessary.
All options:	Filled with compiler options to be specified at the time of compilation. The settings made in [Dialect], [Preprocessor], [Includes], and [Optimization] are automatically reflected in this order.
Command line pattern:	Filled with the command line output format.

Figure 4.2 [Tool Settings] tab - [FUJITSU Fortran Compiler] - [Dialect]

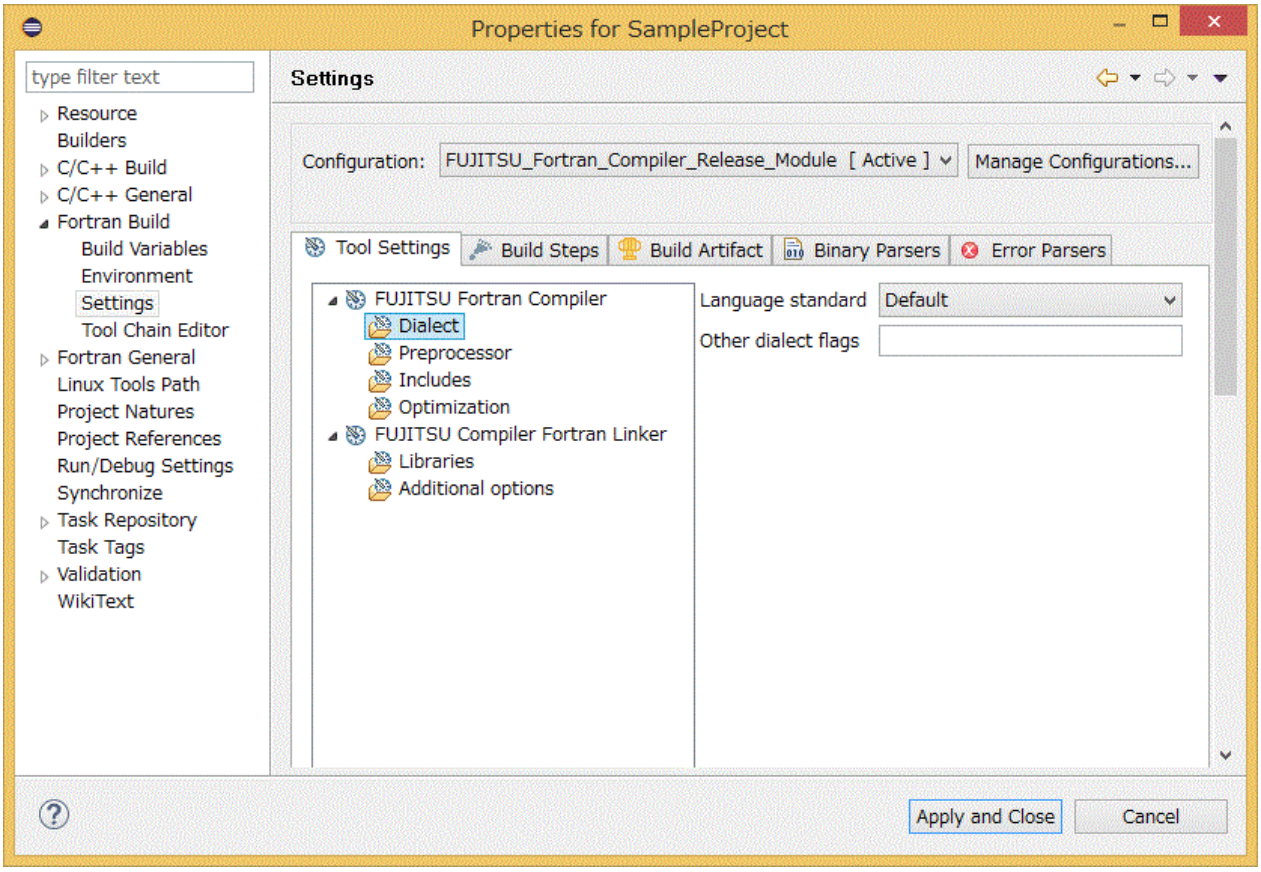


Table 4.2 Settings in [Dialect] (Fortran)

Item Name	Description
Language standard	Selects compiler options for the interpretation of language specifications. If "Default" is selected, no compiler options will be added.
Other dialect flags	Adds an arbitrary compiler option.

Figure 4.3 [Tool Settings] tab - [FUJITSU Fortran Compiler] - [Preprocessor]

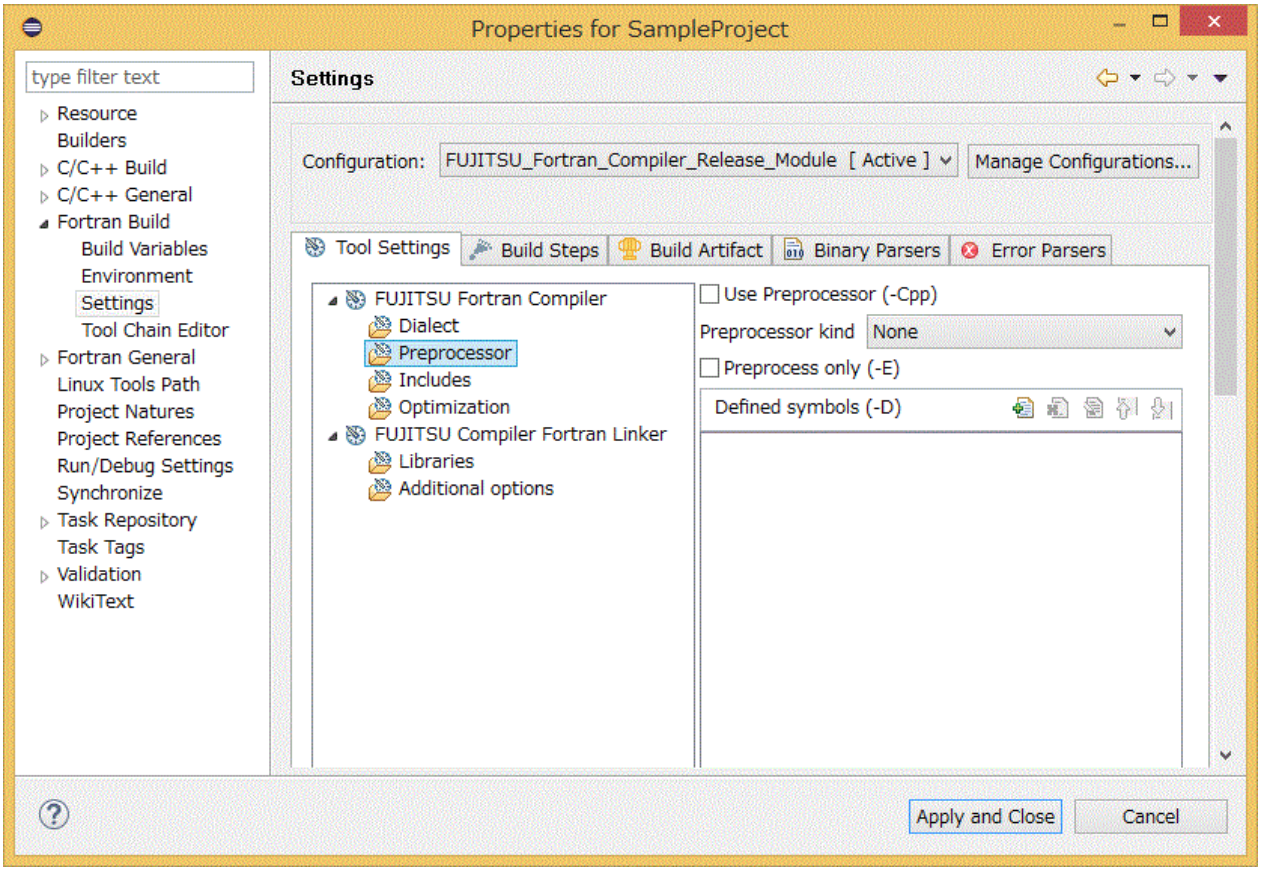


Table 4.3 Settings in [Preprocessor] (Fortran)

Item Name	Description
Use Preprocessor (-Cpp)	Specifies whether to call a preprocessor. If this box is checked, the compiler option -Cpp will be added.
Preprocessor kind	Selects the type of the preprocessor to use. None Does not add compiler options. C language (-Ccpp) Adds the compiler option -Ccpp. Fortran (-Cfpp) Adds the compiler option -Cfpp.
Preprocess only (-E)	If this box is checked, the compiler option -E will be added. Note The preprocessor results are output to the [Console] view.
Defined symbols (-D)	Like the preprocessing directive #define, associates *name* with *tokens*. *name*=*tokens* is added as an argument for the compiler option -D. *name* and *tokens* are arbitrary values. Click the + icon on the right to open the input window. Add *name*=*tokens* and click the [OK] button. Repeat this step if you want to specify more than one.

Figure 4.4 [Tool Settings] tab - [FUJITSU Fortran Compiler] - [Includes]

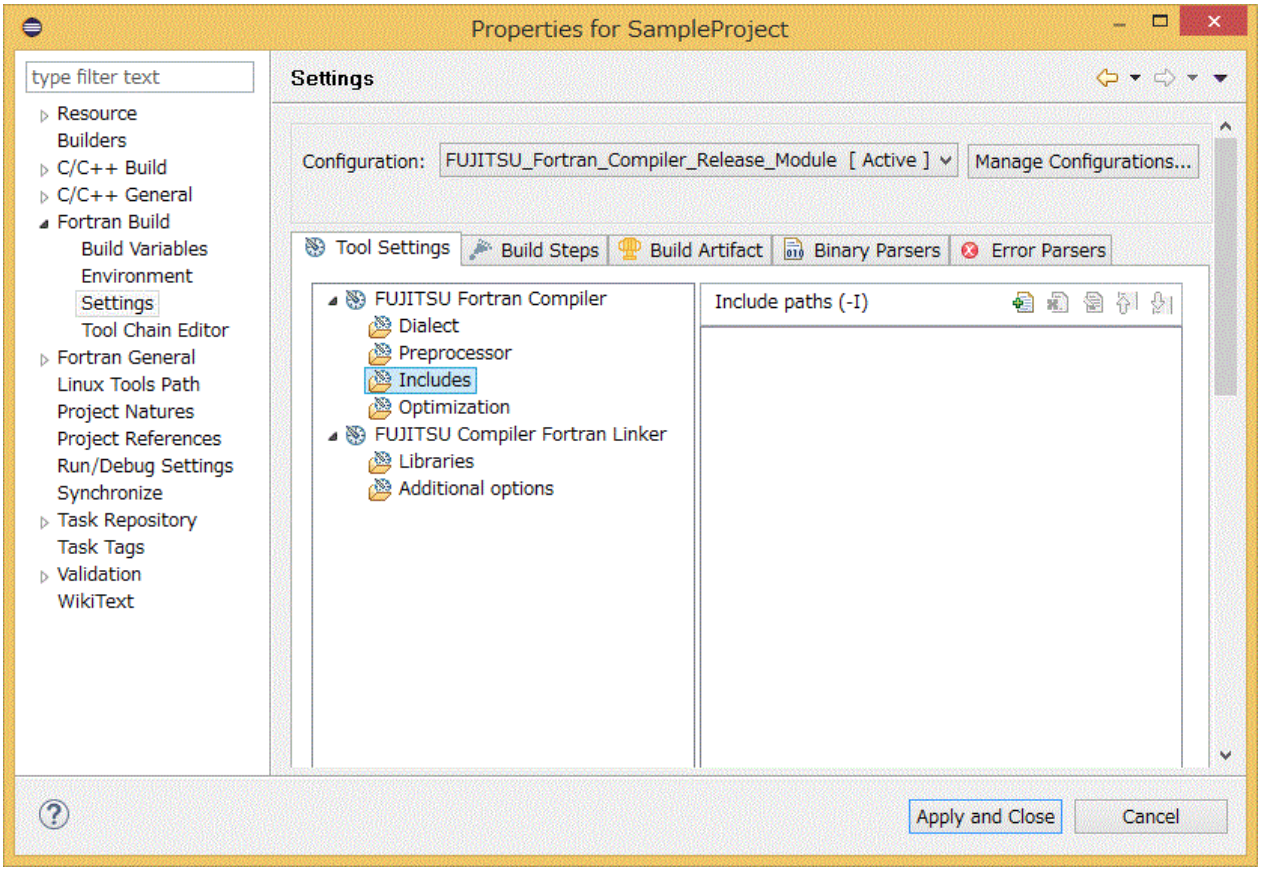


Table 4.4 Settings in [Includes] (Fortran)

Item Name	Description
Include paths (-I)	Specify the path to the header to add to the reference destination at the time of compilation. Add it as an argument for the compiler option -I. Click the + icon on the right to open the input window. Add the path to the header and click the [OK] button. Repeat this step if you want to specify more than one.

Figure 4.5 [Tool Settings] tab - [FUJITSU Fortran Compiler] - [Optimization]

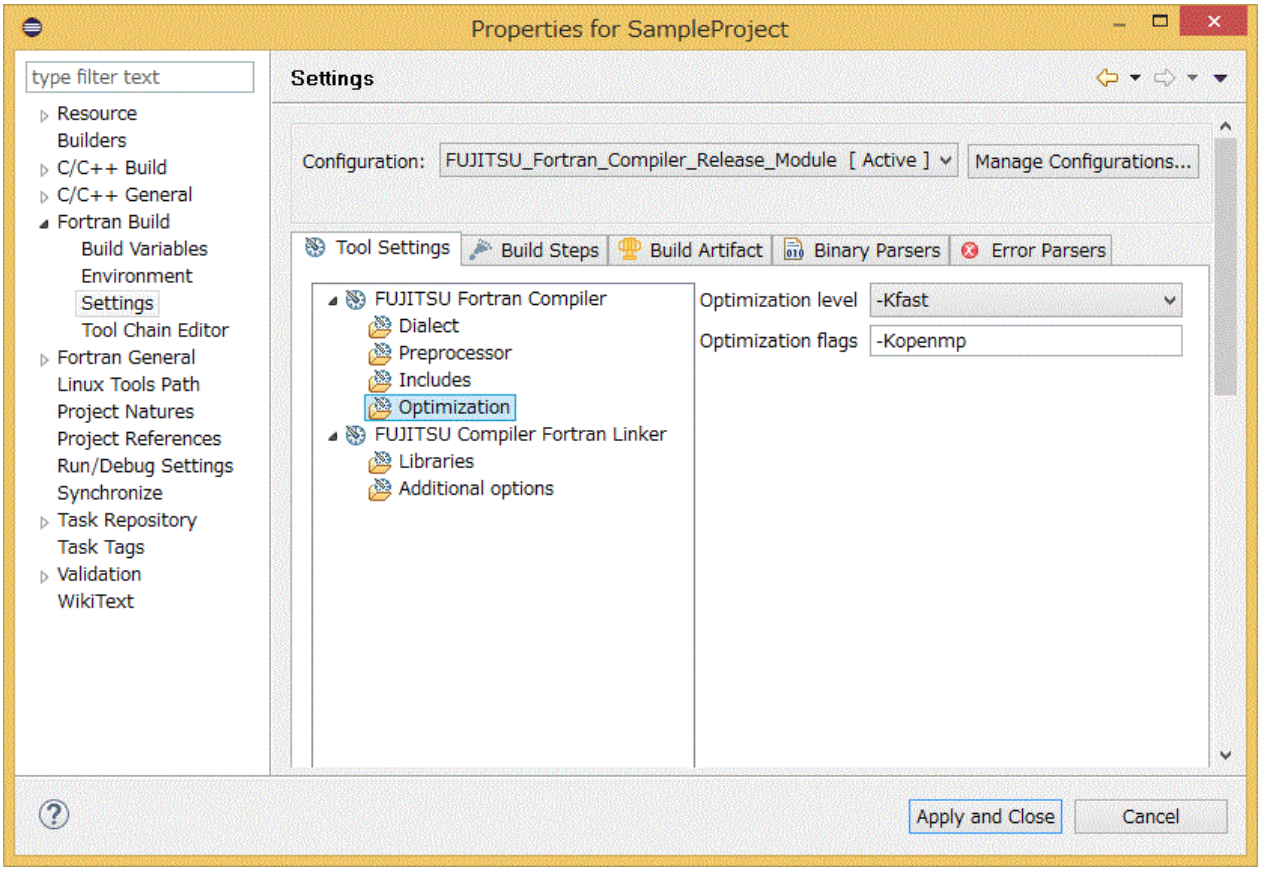


Table 4.5 Settings in [Optimization] (Fortran)

Item Name	Description
Optimization level	Selects compiler options related to the optimization level.
Optimization flags	Adds an arbitrary compiler option.

Figure 4.6 [Tool Settings] tab - [FUJITSU Compiler Fortran Linker]

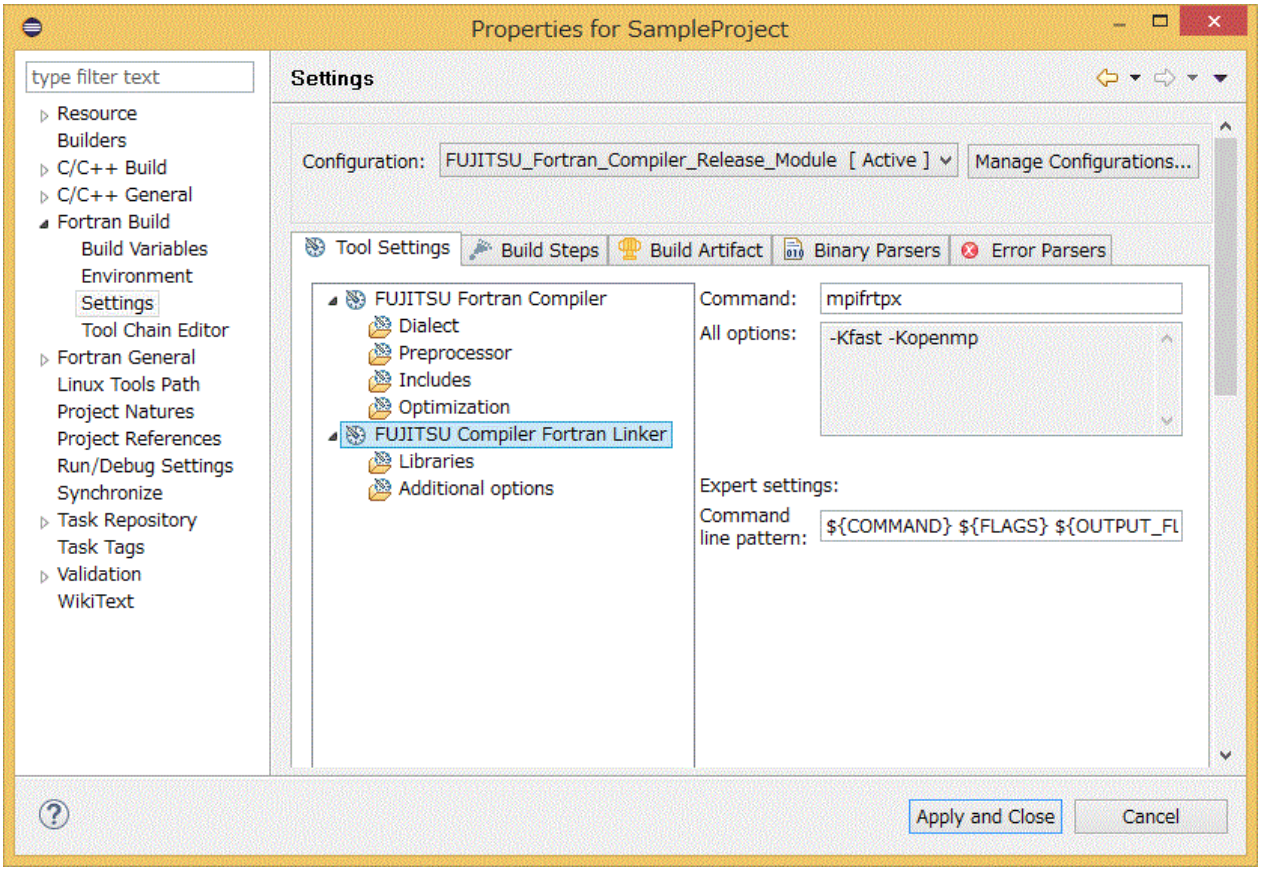


Table 4.6 Settings in [FUJITSU Compiler Fortran Linker]

Item Name	Description
Command:	Adds the command name to be used at the time of linking. Rewrite if necessary.
All options:	Filled with the compiler options to be used at the time of linking. Settings made in [Libraries] and [Additional options] are automatically reflected in this order.
Command line pattern:	Filled with the command line output format.

Figure 4.7 [Tool Settings] tab - [FUJITSU Compiler Fortran Linker] - [Libraries]

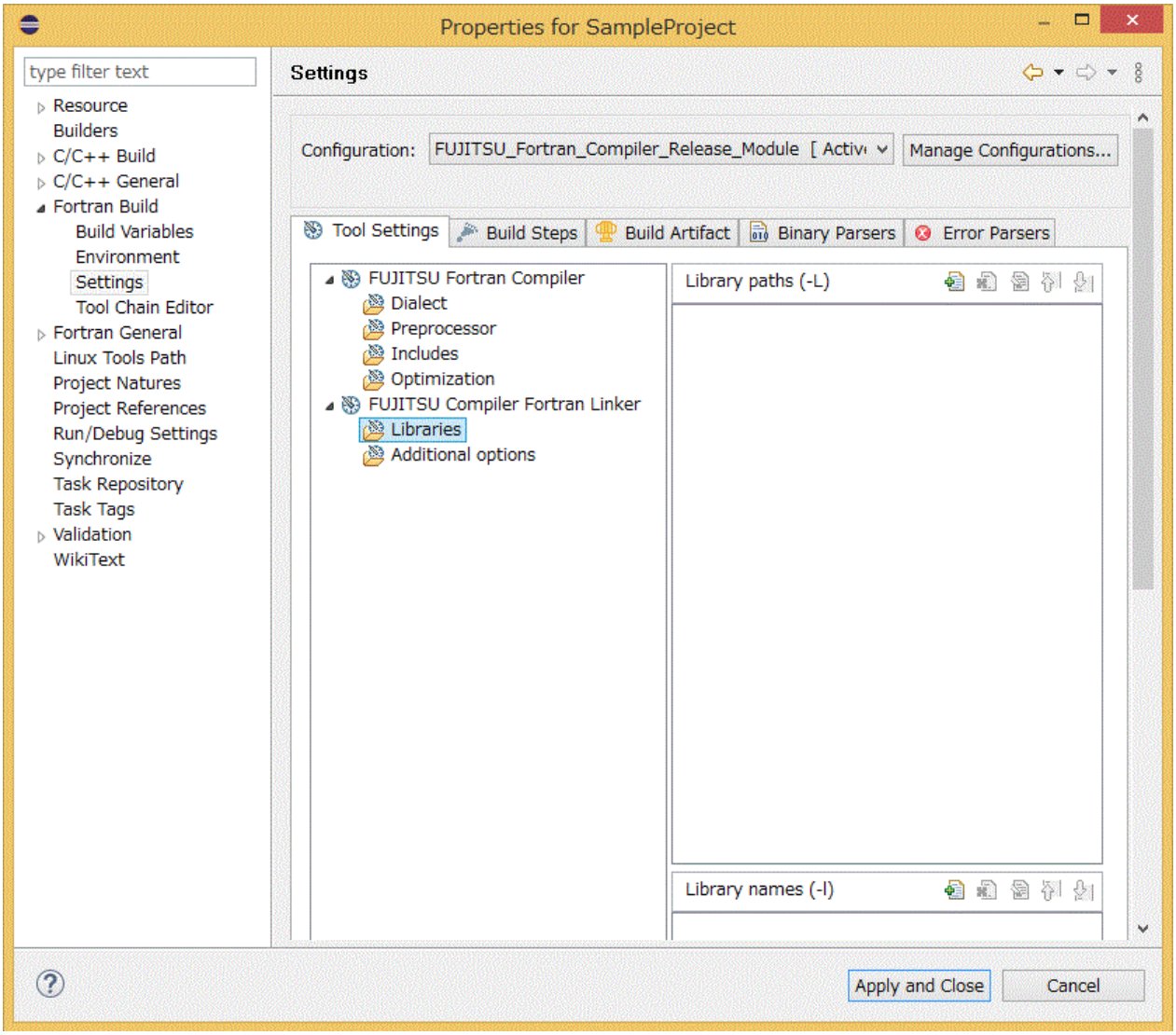


Table 4.7 Settings in [Libraries] (Fortran)

Item Name	Description
Library paths (-L)	Specifies a list of directories to search for a library. The list is specified as an argument for the compiler option -L. Click the + icon on the right to open the input window. Add directory path and click the [OK] button. Repeat this step if you want to specify more than one.
Library names (-l)	Adds the specified library name as a search target. The name is specified as an argument for the compiler option -l. Click the + icon on the right to open the input window. Add library name and click the [OK] button. Repeat this step if you want to specify more than one.

Figure 4.8 [Tool Settings] tab - [FUJITSU Fortran Linker] - [Additional options]

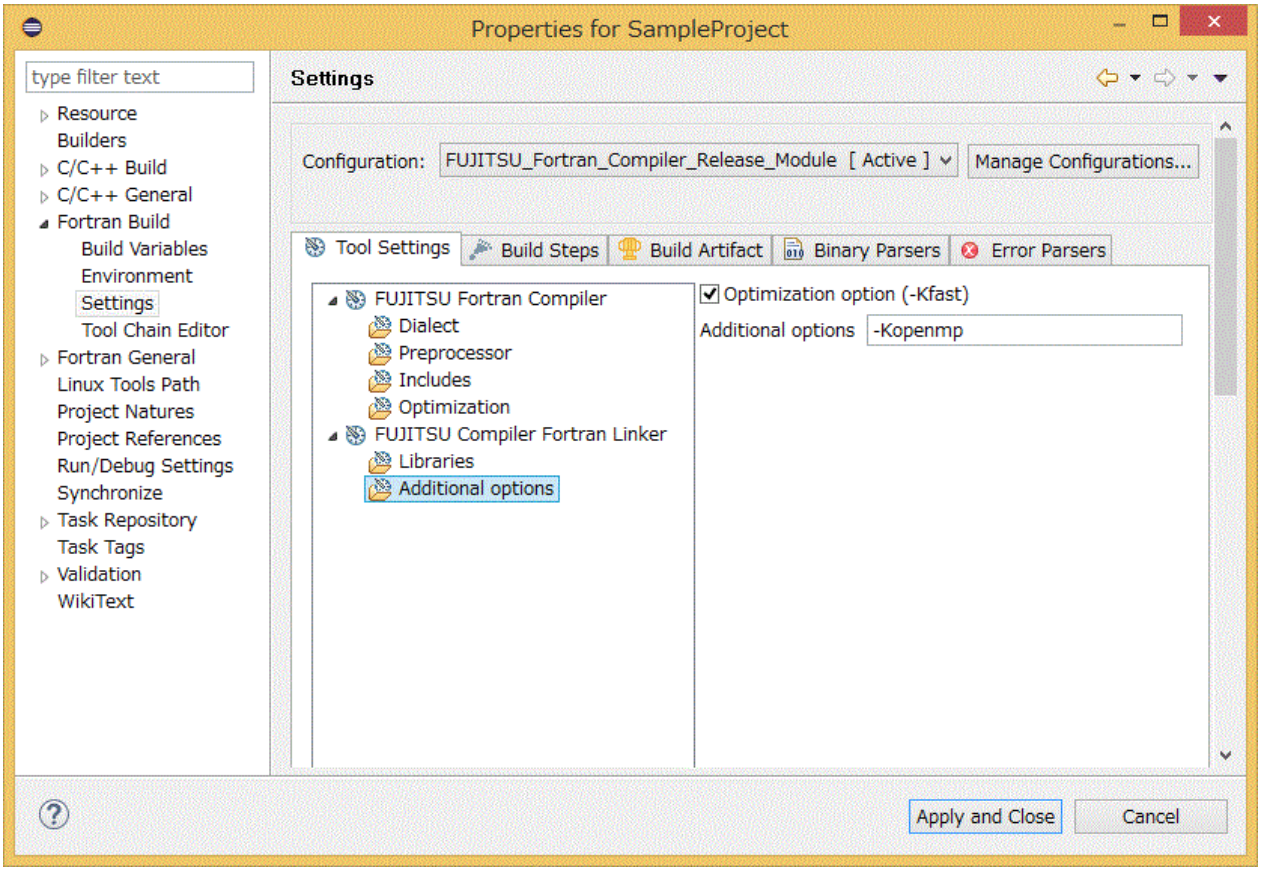


Table 4.8 Settings in [Additional options] (Fortran)

Item Name	Description
Optimization option (-Kfast)	Specifies whether to add the compiler option -Kfast at the time of linking. If this box is checked, the compiler option -Kfast will be added at the time of linking.
Additional options	Adds an arbitrary compiler option.

4.1.5.2 Specify Compiler Options (C)

Select [C/C++ Build] - [Settings]. Make settings for the items in the [Tool Settings] tab. Click the [Apply and Close] button.

Figure 4.9 [Tool Settings] tab - [FUJITSU C Compiler]

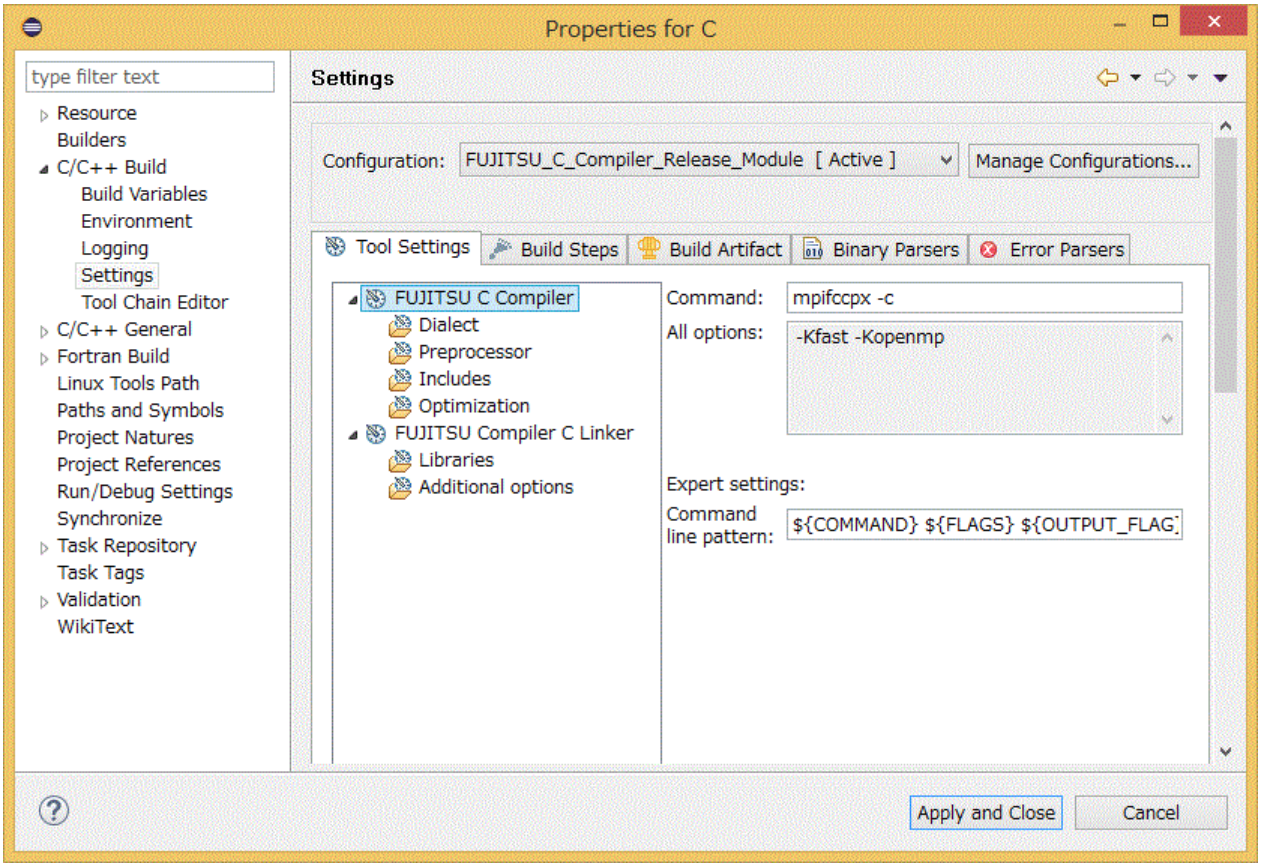


Table 4.9 Settings in [FUJITSU C Compiler]

Item Name	Description
Command	Adds the command name to be used at the time of compilation. Rewrite if necessary.
All options	Filled with compiler options to be specified at the time of compilation. The settings made in [Dialect], [Preprocessor], [Includes], and [Optimization] are automatically reflected in this order.
Command line pattern	Filled with the command line output format.

Figure 4.10 [Tool Settings] tab - [FUJITSU C Compiler] - [Dialect]

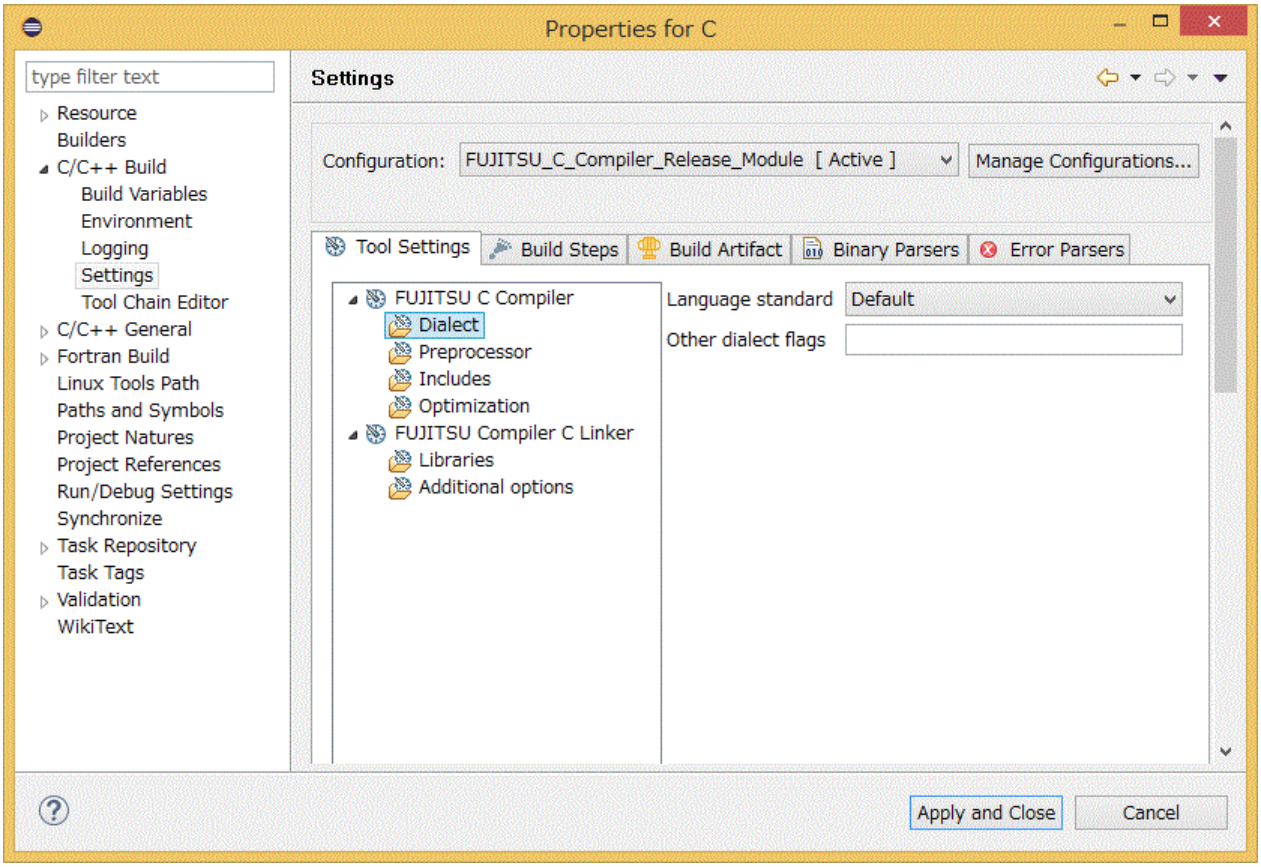


Table 4.10 Settings in [Dialect] (C)

Item Name	Description
Language standard	Selects compiler options for the interpretation of language specifications. If "Default" is selected, no compiler options will be added.
Other dialect flags	Adds an arbitrary compiler option.

Figure 4.11 [Tool Settings] tab - [FUJITSU C Compiler] - [Preprocessor]

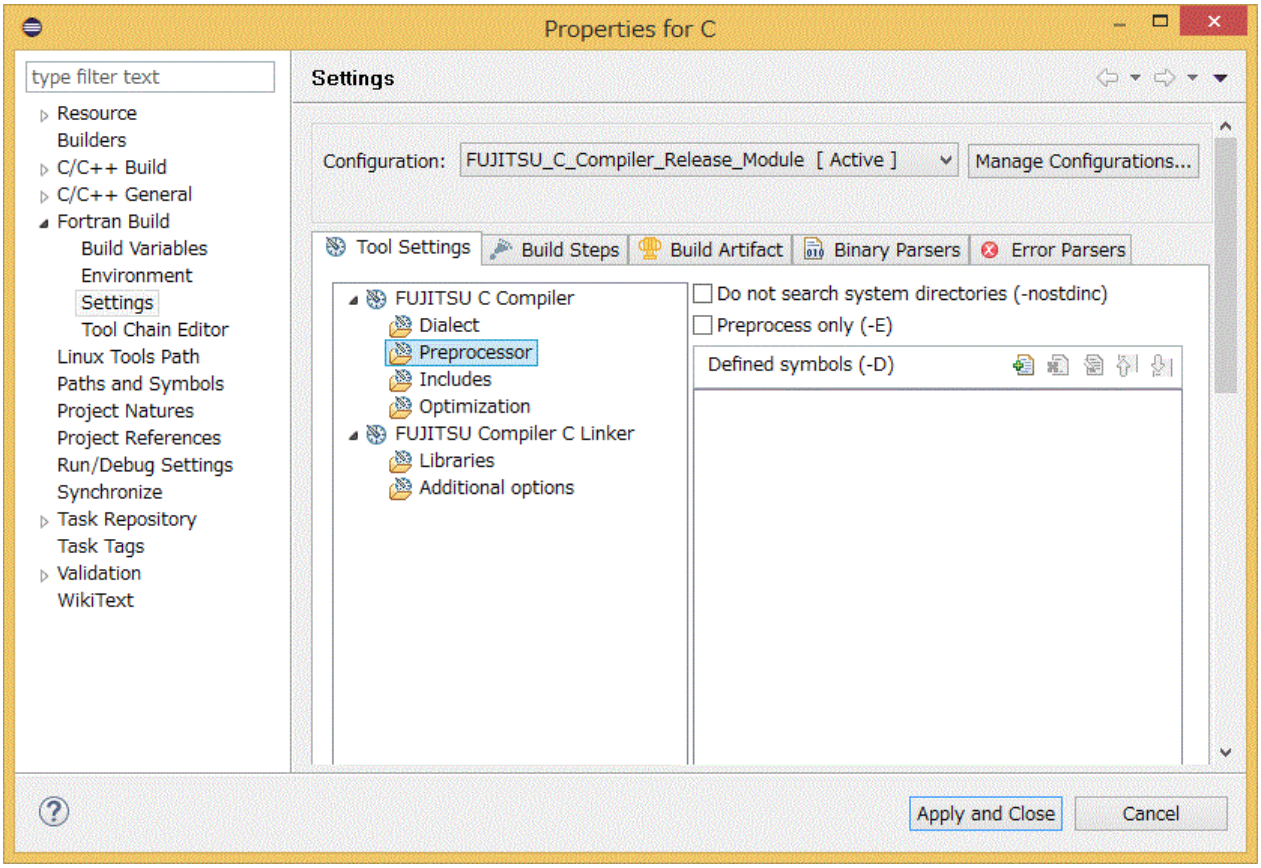


Table 4.11 Settings in [Preprocessor] (C)

Item Name	Description
Do not search system directories (-nostdinc)	Specifies that no standard directories are searched when searching a header. If this box is checked, the compiler option -nostdinc will be added.
Preprocess only (-E)	If this box is checked, the compiler option -E will be added. **Note** The results of the preprocessor will be output to .o file. You will see compile and link errors, but ignore them.
Defined symbols (-D)	Like the preprocessing directive #define, associates *name* with *tokens*. *name*=*tokens* is added as an argument for the compiler option -D. *name* and *tokens* are arbitrary values. Click the + icon on the right to open the input window. Add *name*=*tokens* and click the [OK] button. Repeat this step if you want to specify more than one.

Figure 4.12 [Tool Settings] tab - [FUJITSU C Compiler] - [Includes]

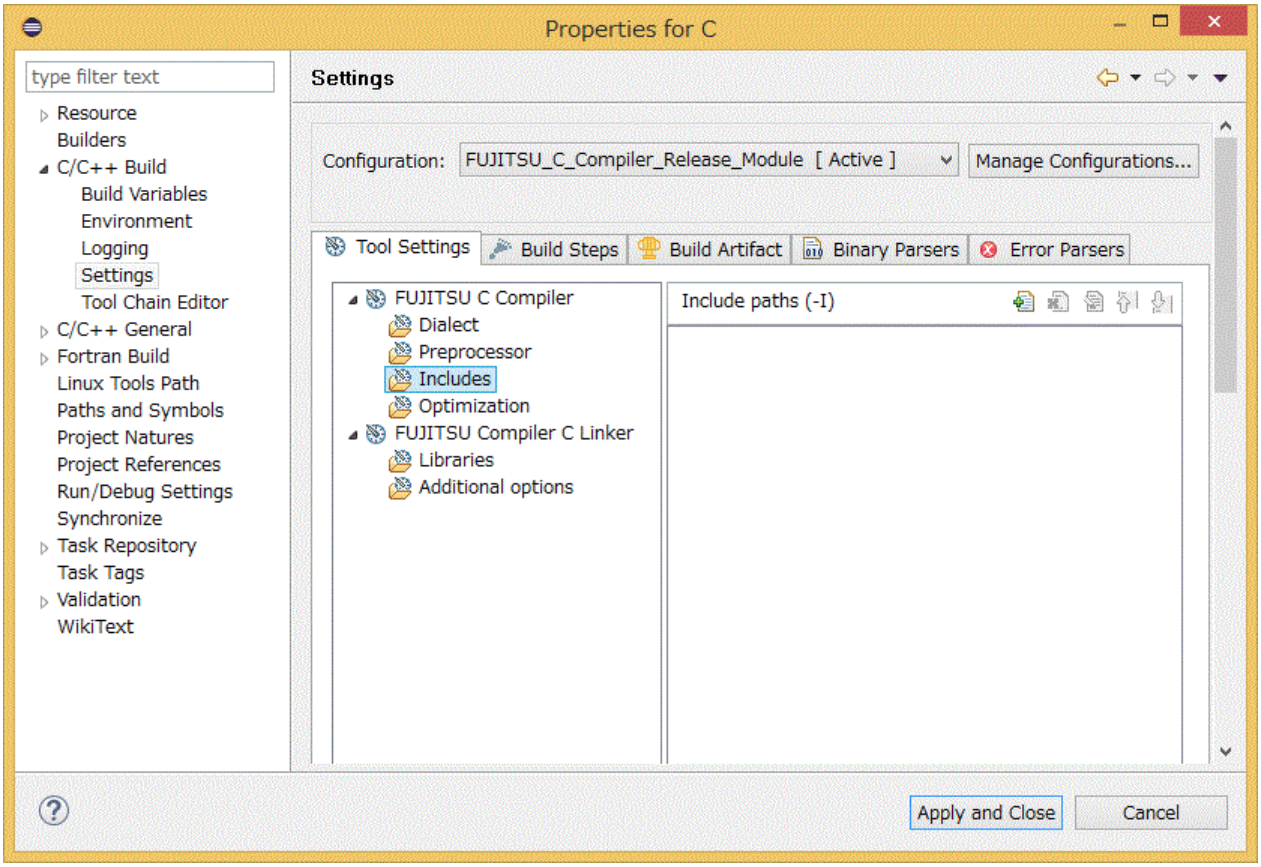


Table 4.12 Settings in [Includes] (C)

Item Name	Description
Include paths (-I)	Specify the path to the header to add to the reference destination at the time of compilation. Add it as an argument for the compiler option -I. Click the + icon on the right to open the input window. Add the path to the header and click the [OK] button. Repeat this step if you want to specify more than one.

Figure 4.13 [Tool Settings] tab - [FUJITSU C Compiler] - [Optimization]

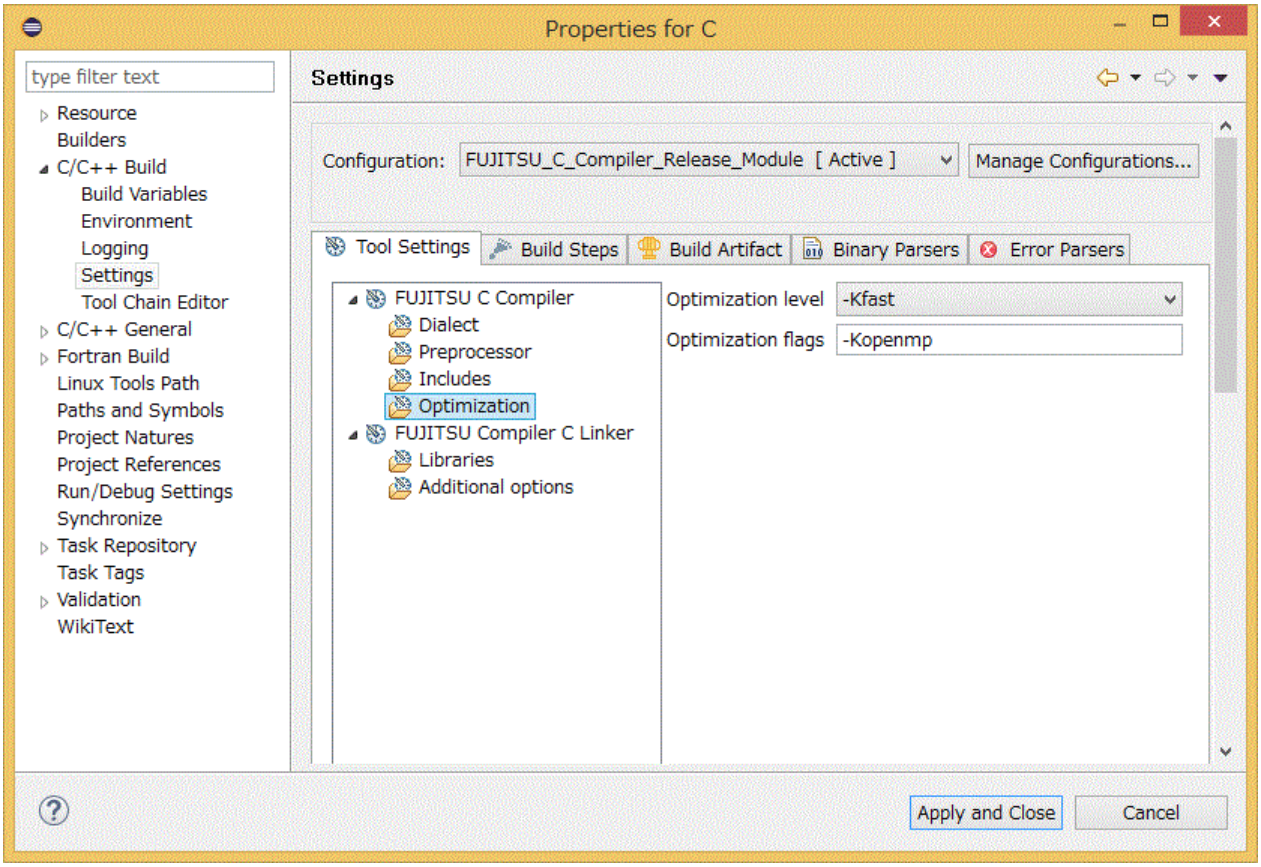


Table 4.13 Settings in [Optimization] (C)

Item Name	Description
Optimization level	Selects compiler options related to the optimization level.
Optimization flags	Adds an arbitrary compiler option.

Figure 4.14 [Tool Settings] tab - [FUJITSU Compiler C Linker]

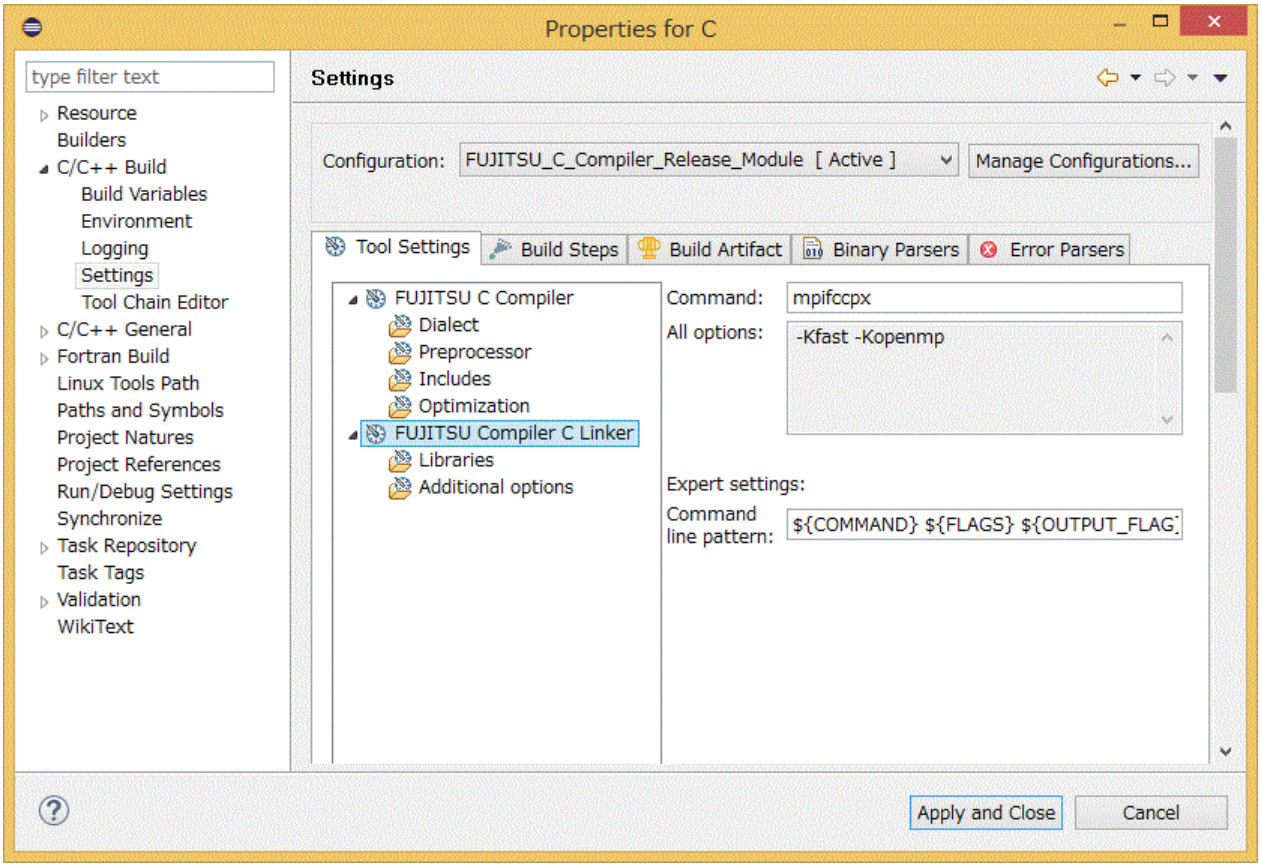


Table 4.14 Settings in [FUJITSU Compiler C Linker]

Item Name	Description
Command:	Adds the command name to be used at the time of linking. Rewrite if necessary.
All options:	Filled with the compiler options to be used at the time of linking. Settings made in [Libraries] and [Additional options] are automatically reflected in this order.
Command line pattern:	Filled with the command line output format.

Figure 4.15 [Tool Settings] tab - [FUJITSU Compiler C Linker] - [Libraries]

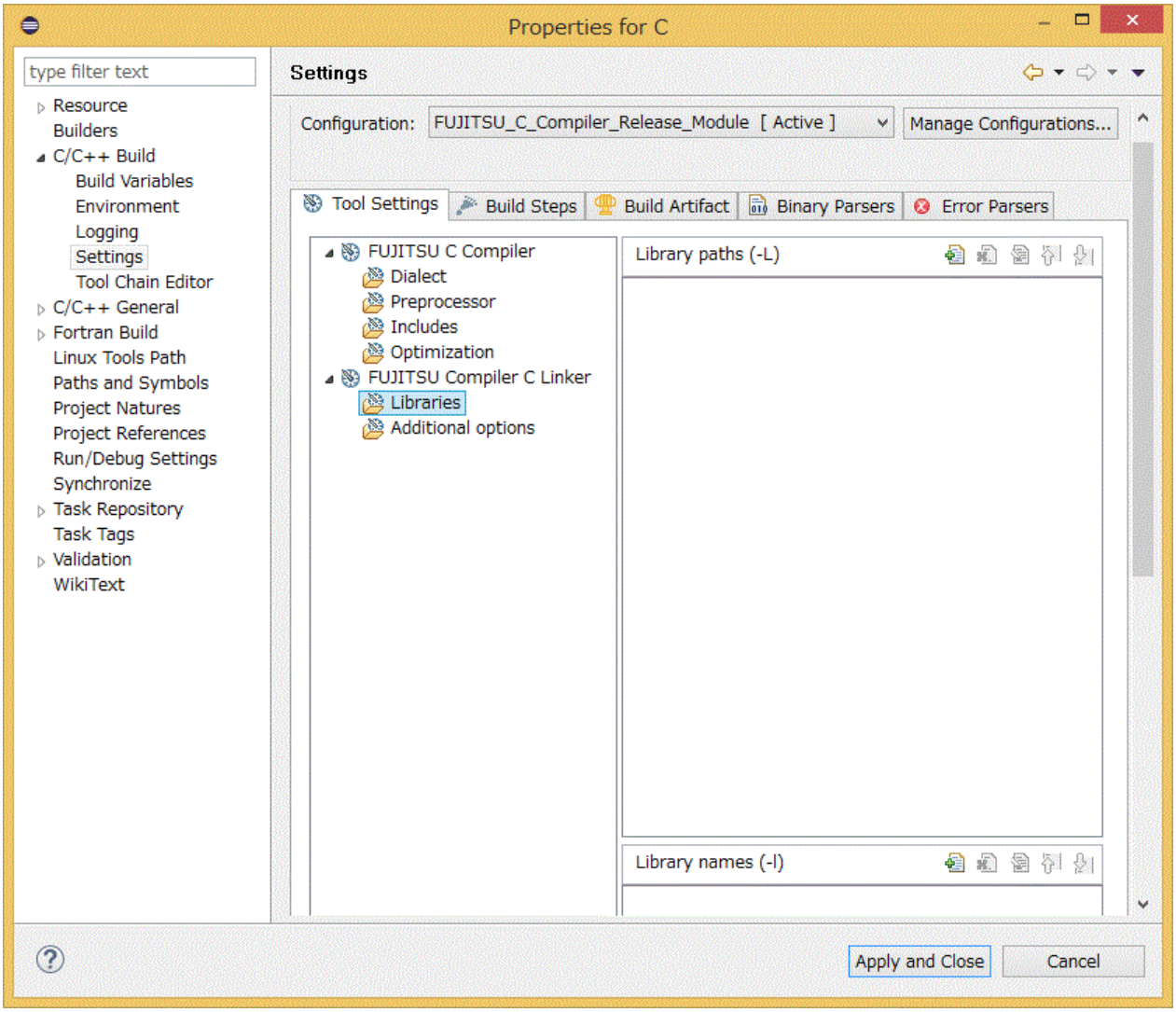


Table 4.15 Settings in [Libraries] (C)

Item Name	Description
Library paths (-L)	Specifies a list of directories to search for a library. The list is specified as an argument for the compiler option -L. Click the + icon on the right to open the input window. Add directory path and click the [OK] button. Repeat this step if you want to specify more than one.
Library names (-l)	Adds the specified library name as a search target. The name is specified as an argument for the compiler option -l. Click the + icon on the right to open the input window. Add library name and click the [OK] button. Repeat this step if you want to specify more than one.

Figure 4.16 [Tool Settings] tab - [FUJITSU Compiler C Linker] - [Additional options]

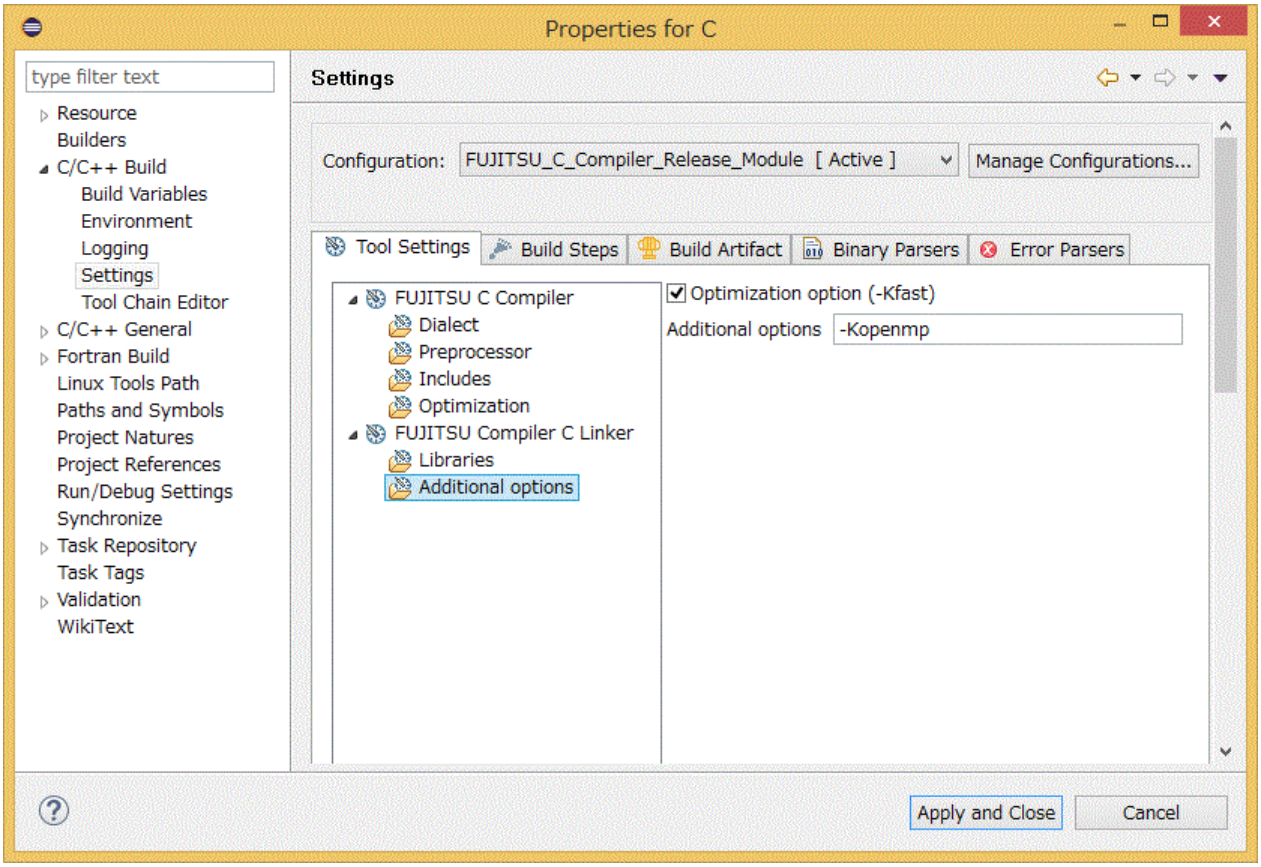


Table 4.16 Settings in [Additional options] (C)

Item Name	Description
Optimization option (-Kfast)	Specifies whether to add the compiler option -Kfast at the time of linking. If this box is checked, the compiler option -Kfast will be added at the time of linking.
Additional options	Adds an arbitrary compiler option.

4.1.5.3 Specify Compiler Options (C++)

Select [C/C++ Build] - [Settings]. Make settings for the items in the [Tool Settings] tab. Click the [Apply and Close] button.

Figure 4.17 [Tool Settings] tab - [FUJITSU C++ Compiler]

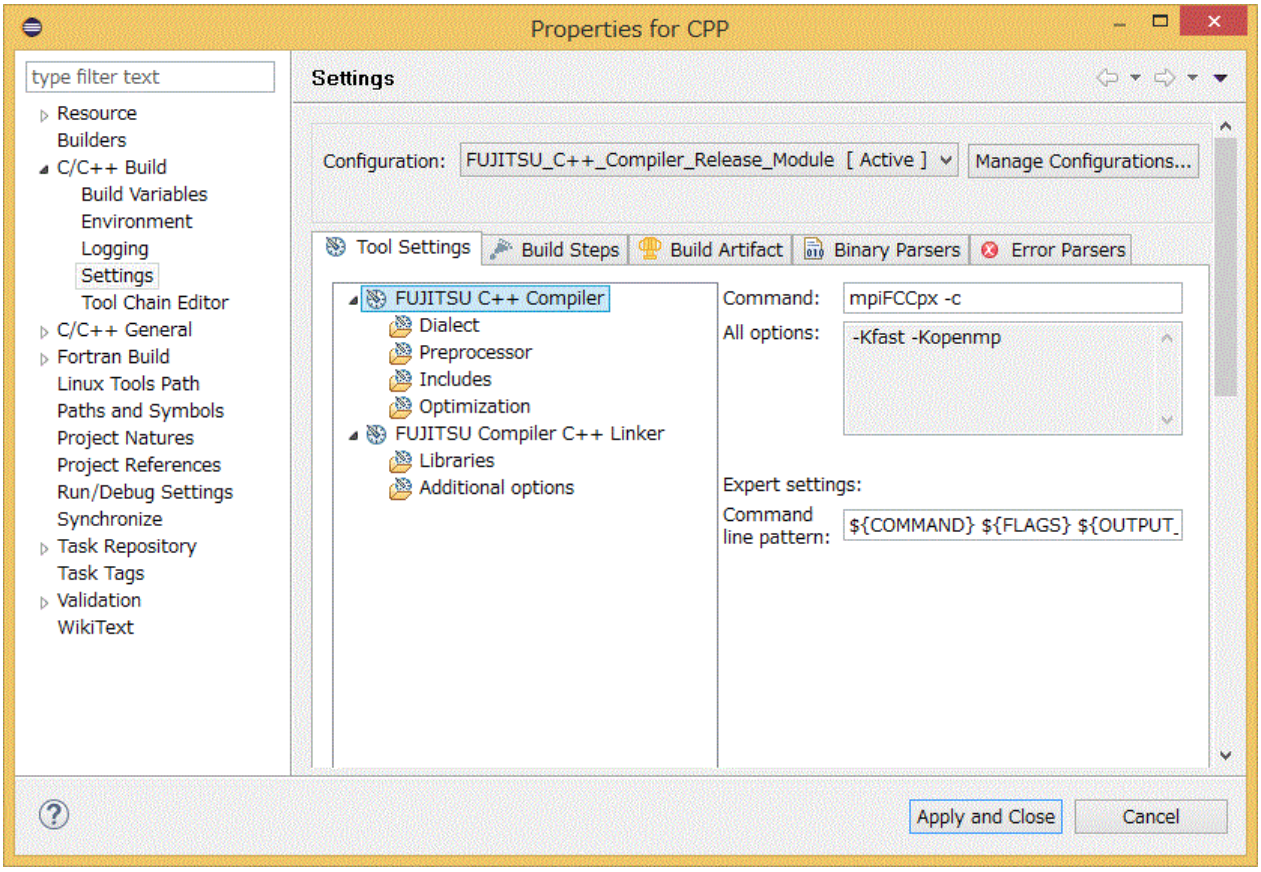


Table 4.17 Settings in [FUJITSU C++ Compiler]

Item Name	Description
Command:	Adds the command name to be used at the time of compilation. Rewrite if necessary.
All options:	Filled with compiler options to be specified at the time of compilation. The settings made in [Dialect], [Preprocessor], [Includes], and [Optimization] are automatically reflected in this order.
Command line pattern:	Filled with the command line output format.

Figure 4.18 [Tool Settings] tab - [FUJITSU C++ Compiler] - [Dialect]

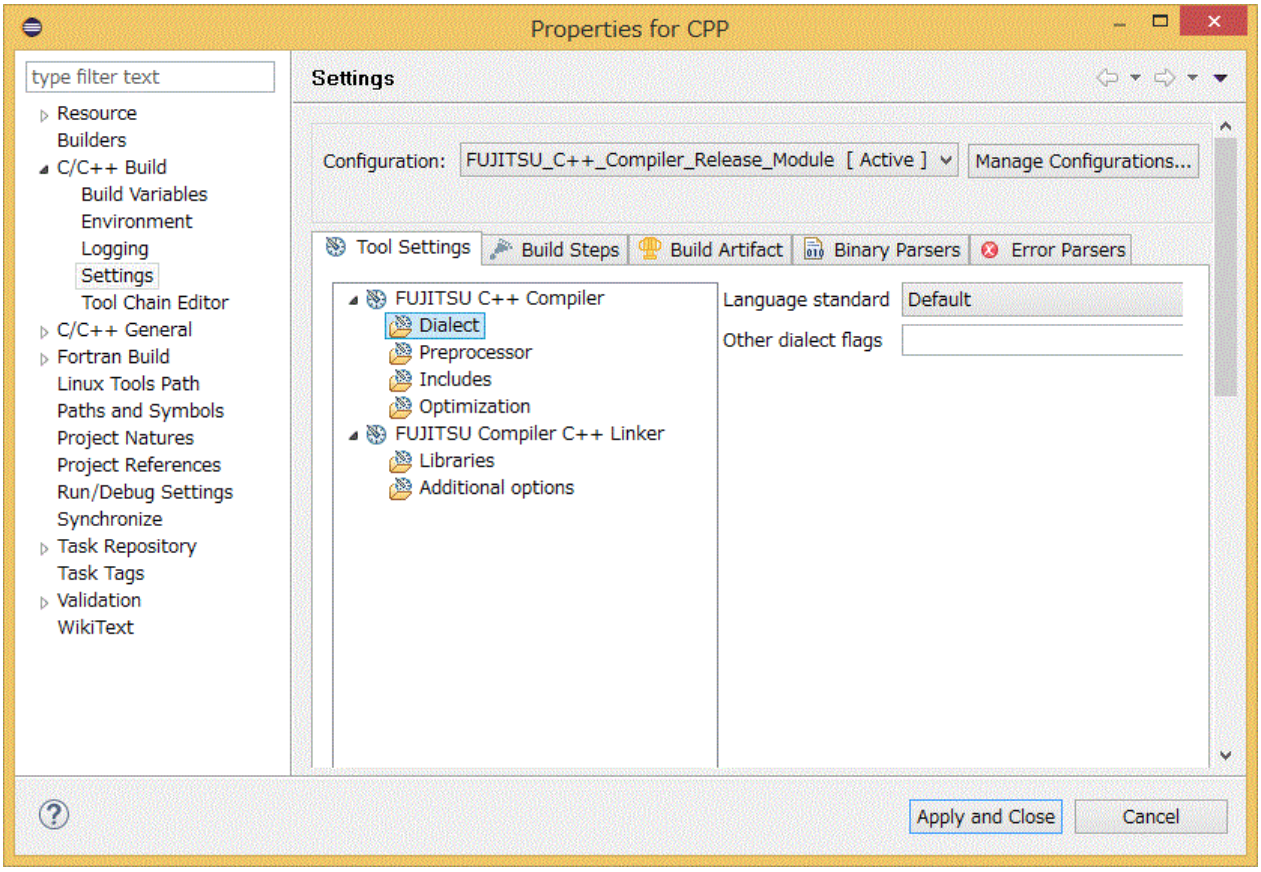


Table 4.18 Settings in [Dialect] (C++)

Item Name	Description
Language standard	Selects compiler options for the interpretation of language specifications. If "Default" is selected, no compiler options will be added.
Other dialect flags	Adds an arbitrary compiler option.

Figure 4.19 [Tool Settings] tab - [FUJITSU C++ Compiler] - [Preprocessor]

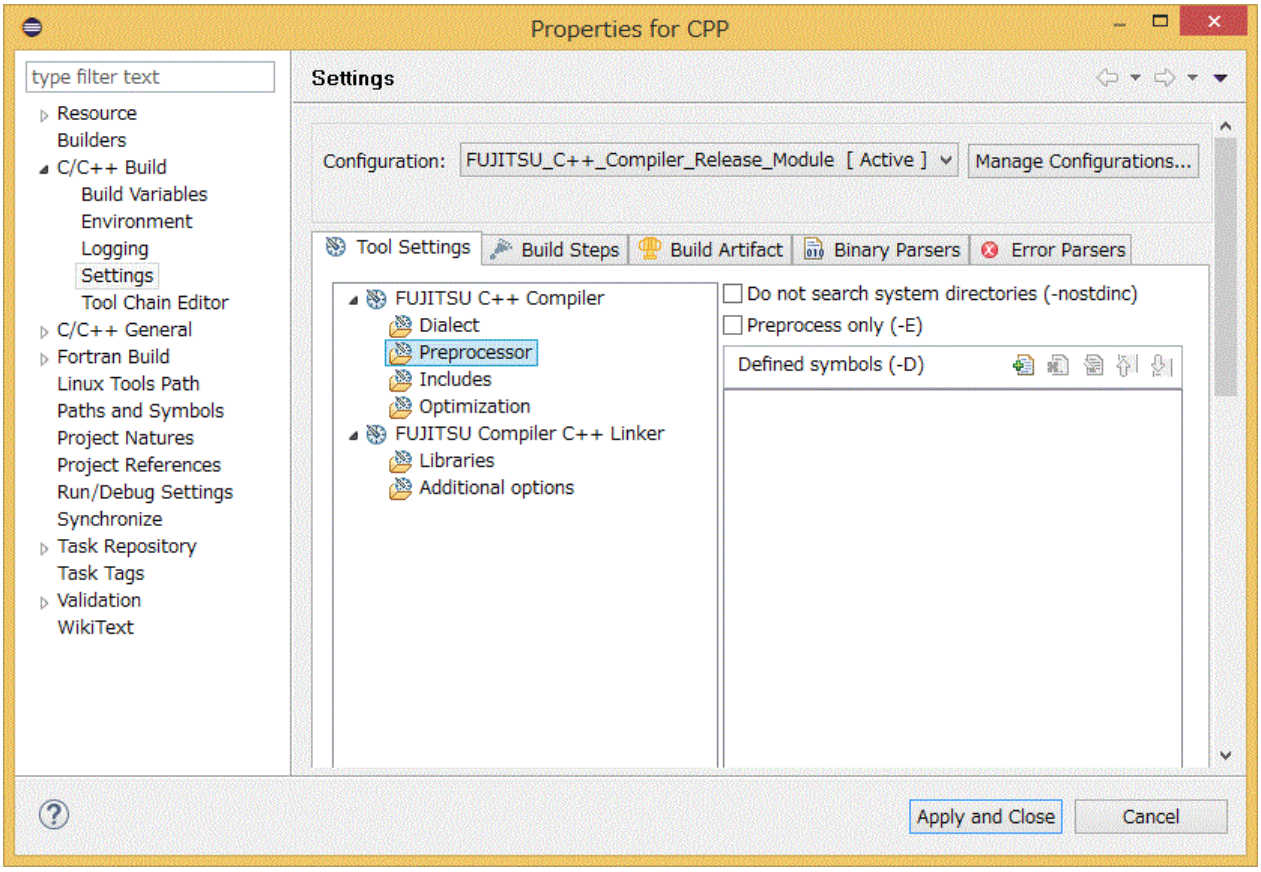


Table 4.19 Settings in [Preprocessor] (C++)

Item Name	Description
Do not search system directories (-nostdinc)	Specifies that no standard directories are searched when searching a header. If this box is checked, the compiler option -nostdinc will be added.
Preprocess only (-E)	If this box is checked, the compiler option -E will be added. Note The results of the preprocessor will be output to .o file. You will see compile and link errors, but ignore them.
Defined symbols (-D)	Like the preprocessing directive #define, associates *name* with *tokens*. *name*=*tokens* is added as an argument for the compiler option -D. *name* and *tokens* are arbitrary values. Click the + icon on the right to open the input window. Add *name*=*tokens* and click the [OK] button. Repeat this step if you want to specify more than one.

Figure 4.20 [Tool Settings] tab - [FUJITSU C++ Compiler] - [Includes]

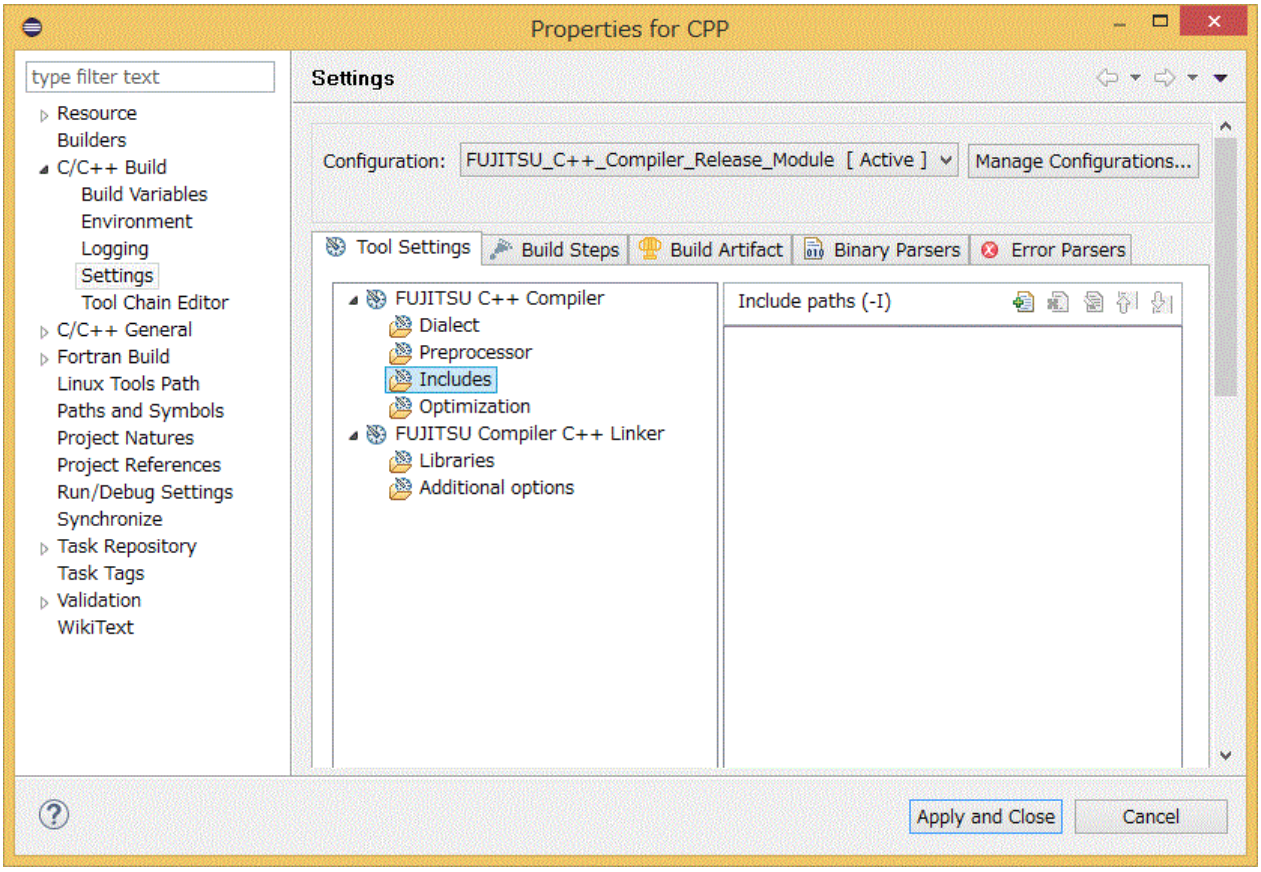


Table 4.20 Settings in [Includes] (C++)

Item Name	Description
Include paths (-I)	Specify the path to the header to add to the reference destination at the time of compilation. Add it as an argument for the compiler option -I. Click the + icon on the right to open the input window. Add the path to the header and click the [OK] button. Repeat this step if you want to specify more than one.

Figure 4.21 [Tool Settings] tab - [FUJITSU C++ Compiler] - [Optimization]

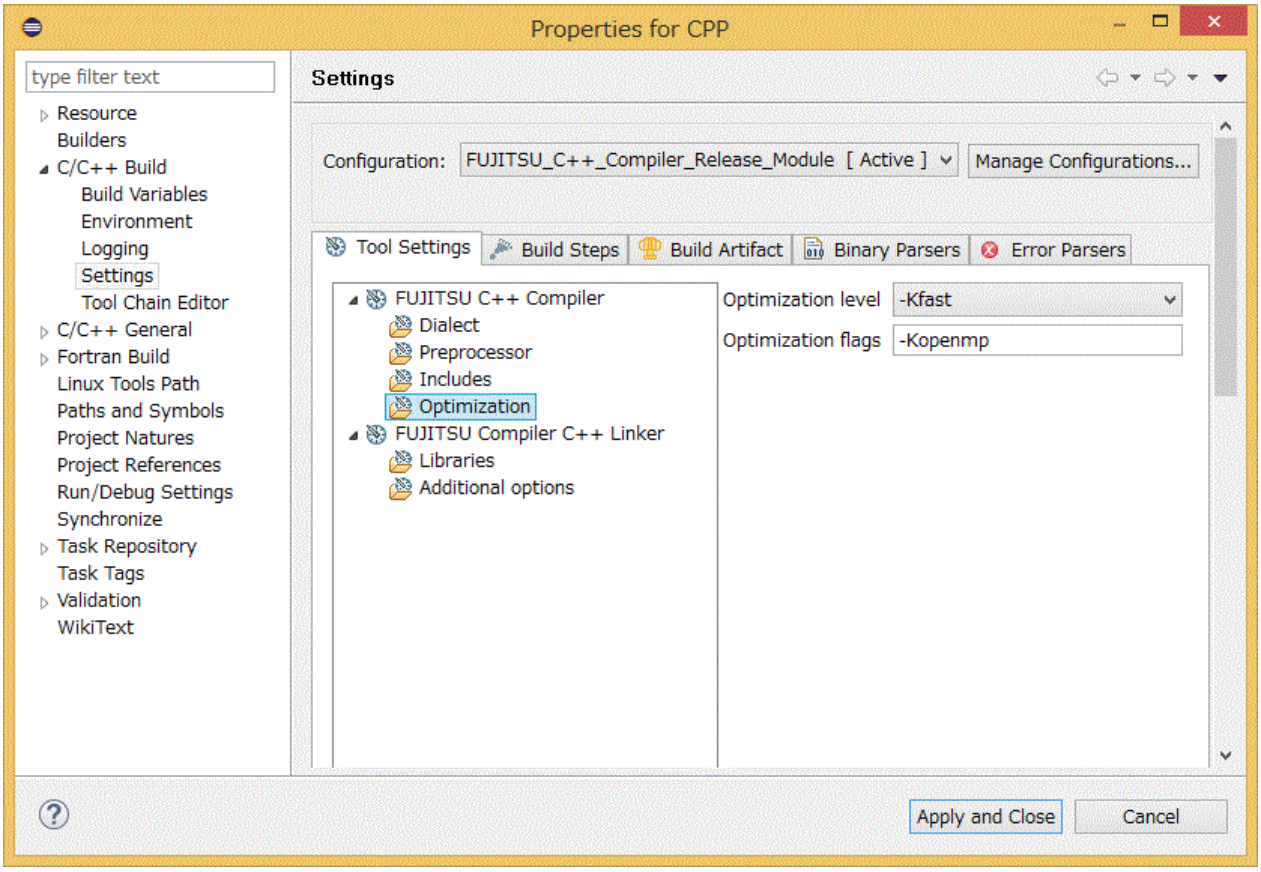


Table 4.21 Settings in [Optimization] (C++)

Item Name	Description
Optimization level	Selects compiler options related to the optimization level.
Optimization flags	Adds an arbitrary compiler option.

Figure 4.22 [Tool Settings] tab - [FUJITSU Compiler C++ Linker]

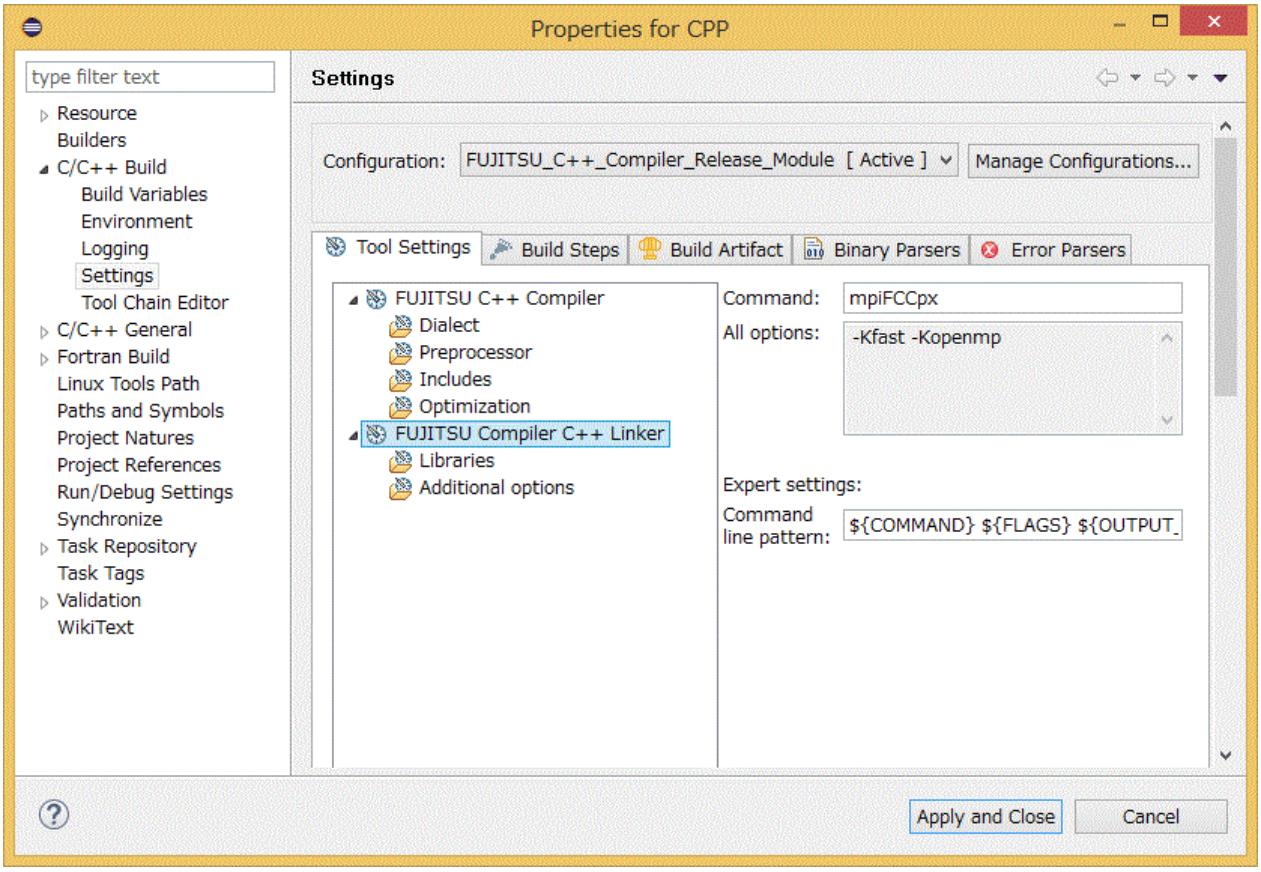


Table 4.22 Settings in [FUJITSU Compiler C++ Linker]

Item Name	Description
Command:	Adds the command name to be used at the time of linking. Rewrite if necessary.
All options:	Filled with the compiler options to be used at the time of linking. Settings made in [Libraries] and [Additional options] are automatically reflected in this order.
Command line pattern:	Filled with the command line output format.

Figure 4.23 [Tool Settings] tab - [FUJITSU Compiler C++ Linker] - [Libraries]

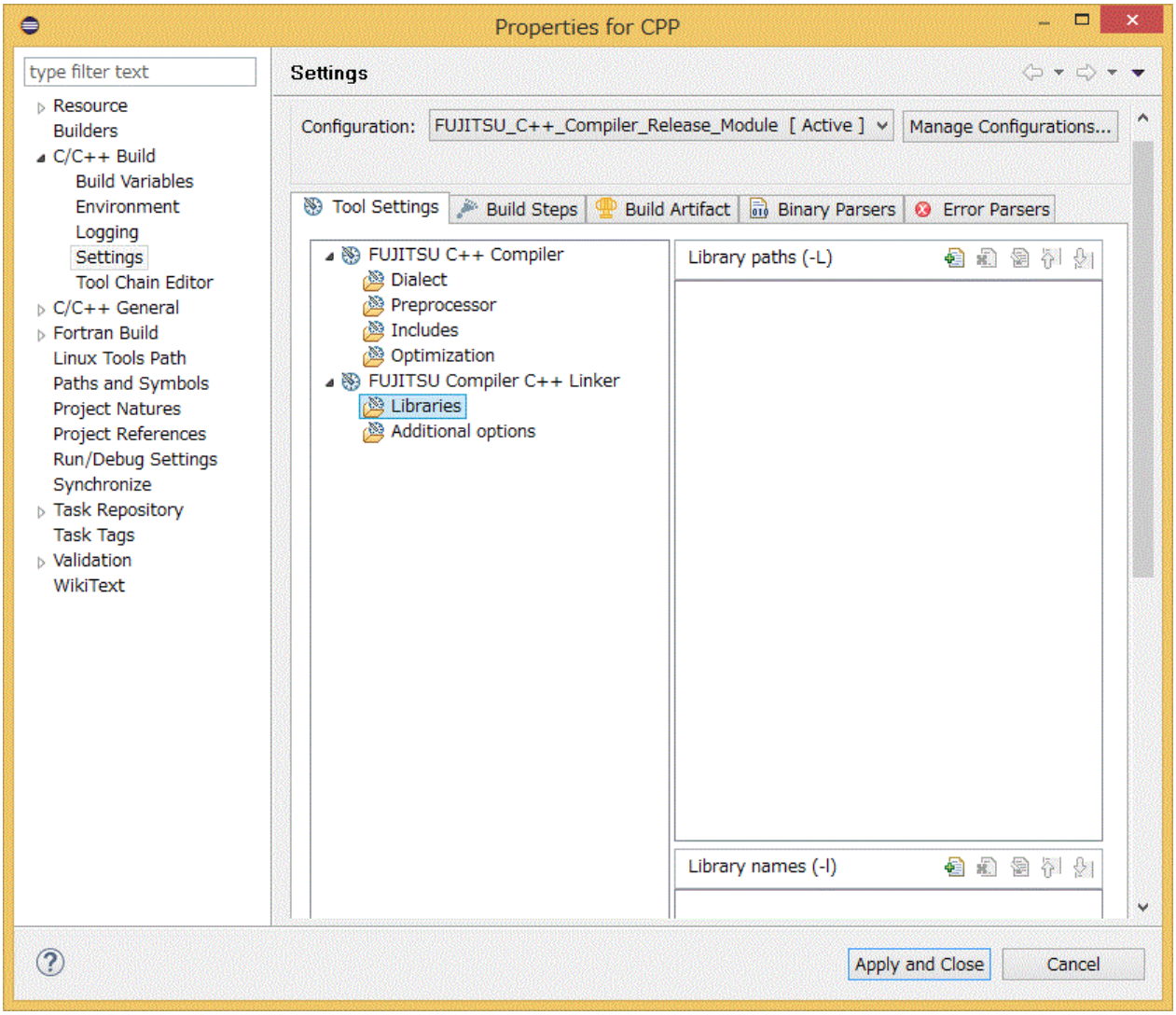


Table 4.23 Settings in [Libraries] (C++)

Item Name	Description
Library paths (-L)	Specifies a list of directories to search for a library. The list is specified as an argument for the compiler option -L. Click the + icon on the right to open the input window. Add directory path and click the [OK] button. Repeat this step if you want to specify more than one.
Library names (-l)	Adds the specified library name as a search target. The name is specified as an argument for the compiler option -l. Click the + icon on the right to open the input window. Add library name and click the [OK] button. Repeat this step if you want to specify more than one.

Figure 4.24 [Tool Settings] tab - [FUJITSU Compiler C++ Linker] - [Additional options]

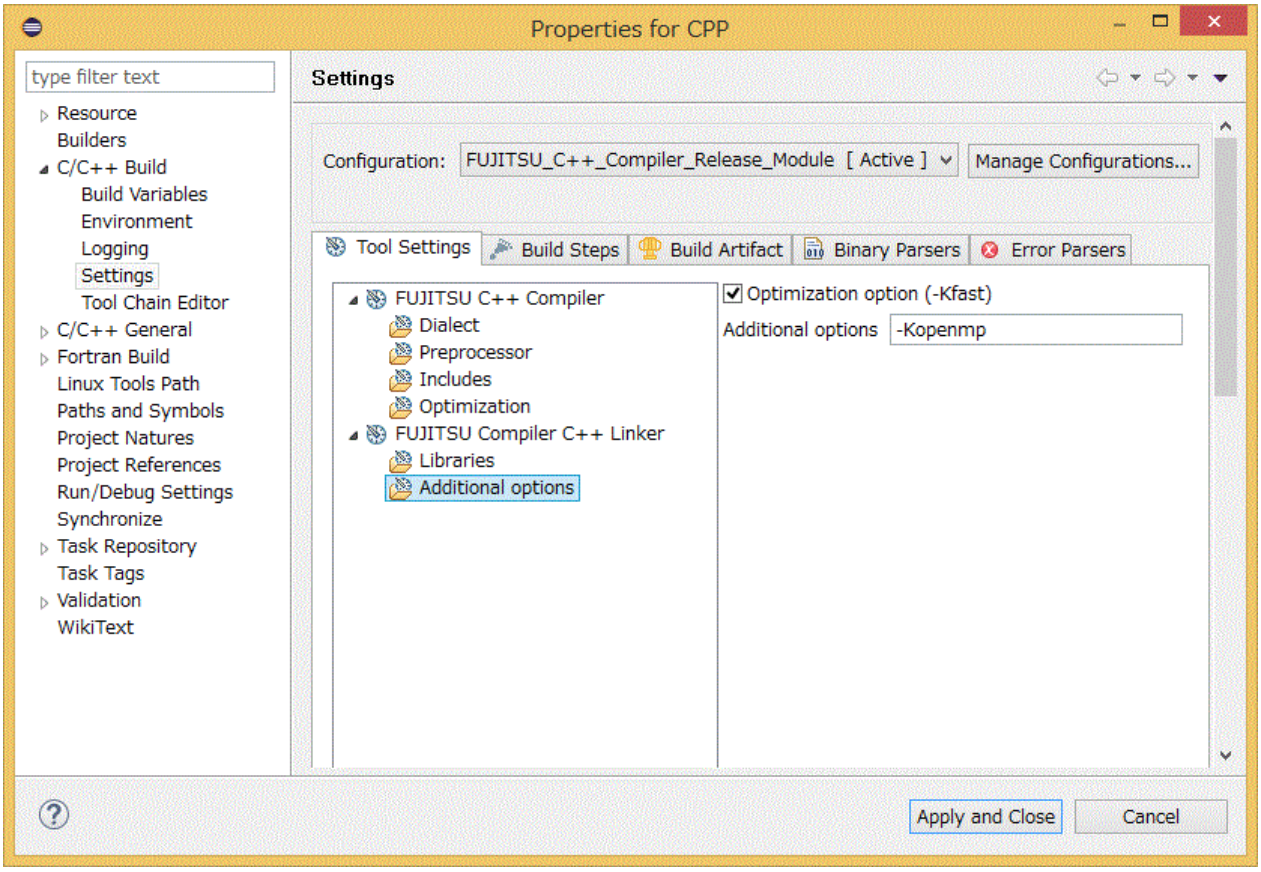


Table 4.24 Settings in [Additional options] (C++)

Item Name	Description
Optimization option (-Kfast)	Specifies whether to add the compiler option -Kfast at the time of linking. If this box is checked, the compiler option -Kfast will be added at the time of linking.
Additional options	Adds an arbitrary compiler option.

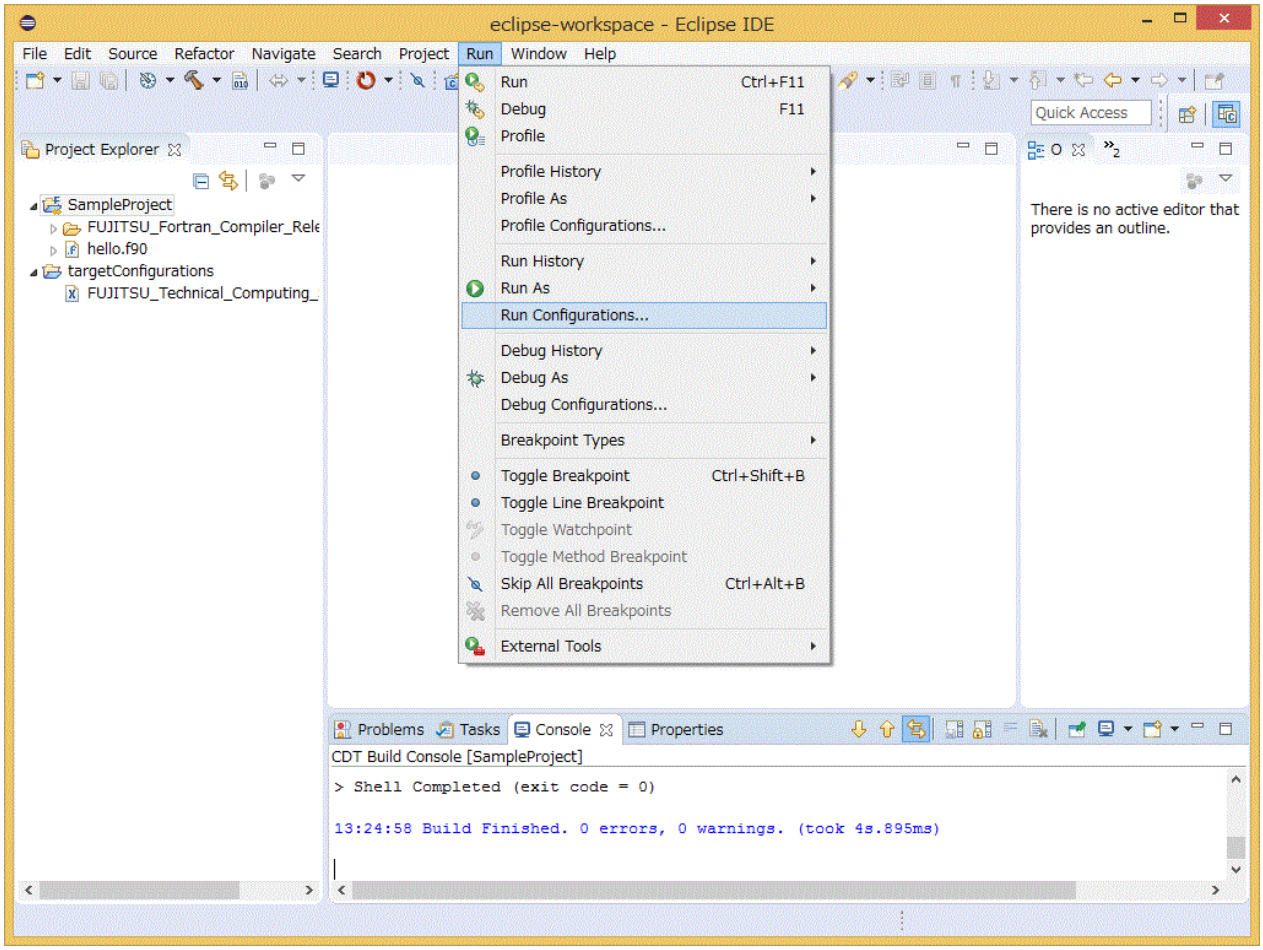
4.1.6 Building a Project

For the procedure for building a project, see "3.4 Building a Project".

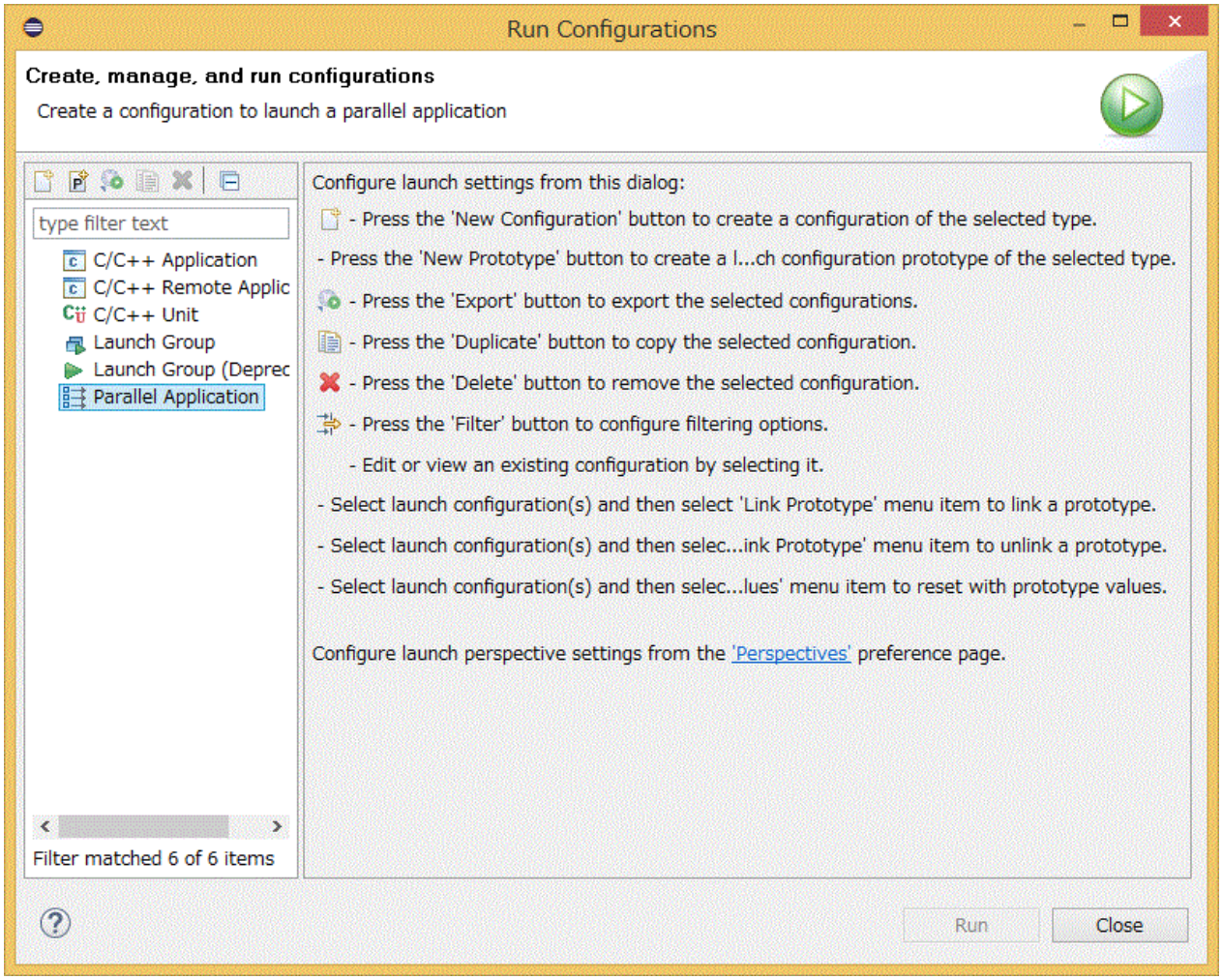
4.2 Job Submission

This section describes the procedure for submitting a job.

1. Click [Run] - [Run Configurations...] on the menu bar.



2. Select [Parallel Application] in the left pane of the [Run Configurations] window, and click the [New launch configuration] icon.



3. A new item (Hereafter referred to as [New Configuration]) is added to [Parallel Application]. Select [New Configuration] to set the contents of the job to be submitted. You can save the set contents. Also, you can register multiple items. The setting items of [New Configuration] are divided across multiple tabs. Set the necessary information, and click the [Run] button. A table below shows details of the necessary settings when submitting a job.

Information

By clicking the [View Script] button, you can check the contents of the job script to be submitted.

Figure 4.25 [New Configuration], [Resources] tab, and [Resources] - [Basic Settings] tab

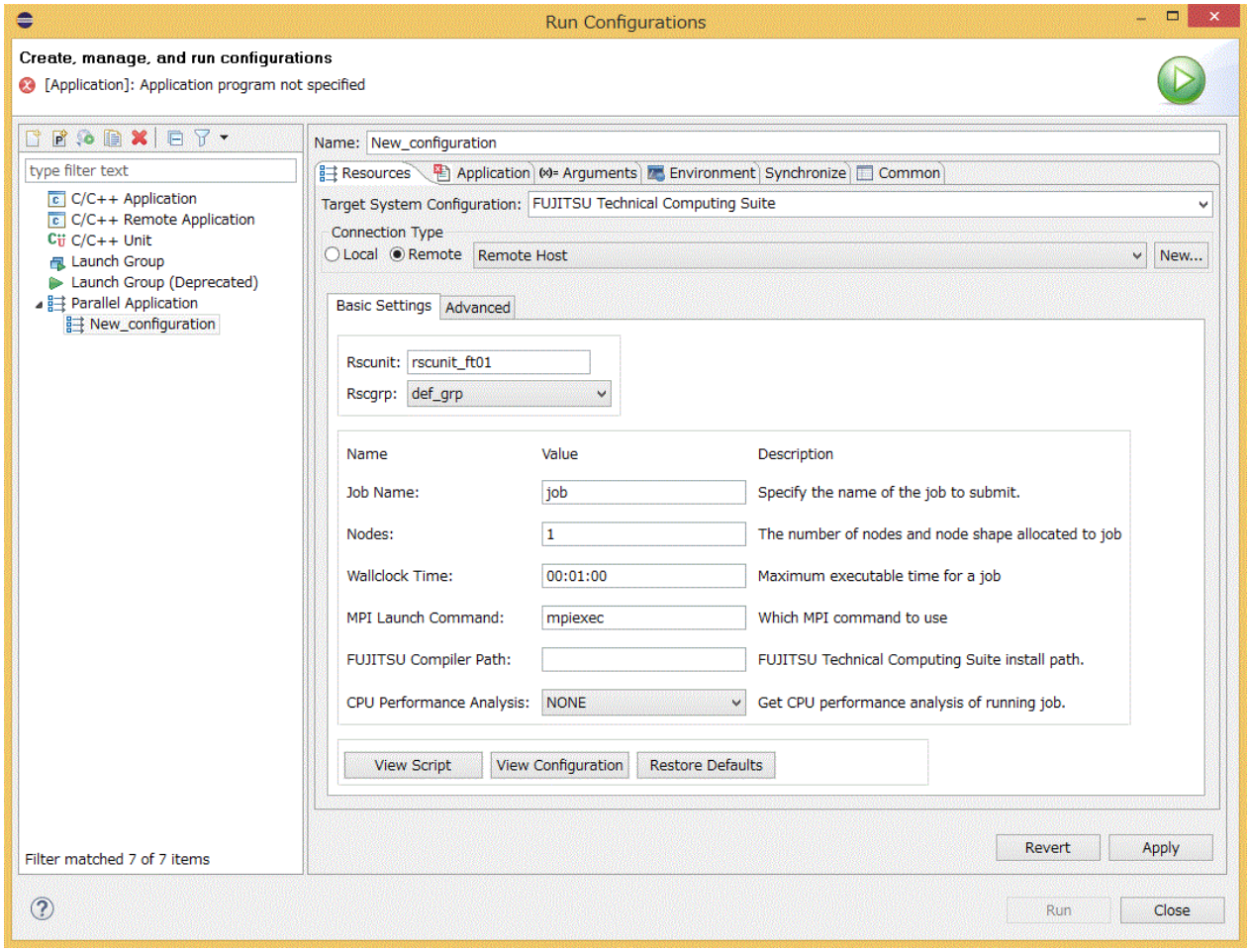


Table 4.25 Settings in [New Configuration]

Item Name	Description
Name:	Specify a name for [New configuration]. Specify an arbitrary name since the name is used for identification by the user.

Table 4.26 Settings in [Resources] tab

Item Name	Description
Target System Configuration:	Select "FUJITSU Technical Computing Suite"
Connection Type	Select [Remote], and then select a name from the pull-down menu. Select the one created for [Connection name] in "2.3.1 Connecting to the Login Node (Remote System)".

Table 4.27 Settings in [Resources] - [Basic Settings] tab

Item Name	Description
Rscunit:	Specify the name of the resource unit to be used.
Rscgrp:	Select the resource group to be used.
Job Name:	Specify the name of the job to be submitted. The name specified here is the file name used in PJM output results.

Item Name	Description
	Note Do not include the yen sign (\), slashes, single quotation marks, or double quotation marks in the specified job name.
Nodes:	Specify the configuration of nodes to be used.
Wallclock time:	Specify the limit on executable time per job in *hh*:*mm*:*ss* format.
MPI Launch Command:	Specify "mpiexec". If you do not want to use the mpiexec command, change it to blank. **Note** To specify the number of parallel processes during MPI program execution, you need to write that after "mpiexec." For example, write the following to specify 2 processes: `mpiexec -n 2`

FUJITSU Compiler Path:	Specify *installation-path*.
	Use *FUJITSU Compiler Path* to add the following environment variables.
	`LANG_HOME : FUJITSU Compiler Path`
	`PATH : $[LANG_HOME}/bin:${PATH}`
	`LD_LIBRARY_PATH : $[LANG_HOME}/lib64:${LD_LIBRARY_PATH}`
	For details on "*installation_path*", contact the system administrator.

CPU Performance Analysis:	Specify the creation of a CPU Performance Analysis Report. For details on CPU Performance Analysis Report, see the "Profiler User's Guide". None Does not specify the creation of a CPU Performance Analysis Report. Single Specifies the creation of a CPU Performance Analysis Report (Single report). Brief Specifies the creation of a CPU Performance Analysis Report (Single report). Standard Specifies the creation of a CPU Performance Analysis Report (Single report). Detail Specifies the creation of a CPU Performance Analysis Report (Single report).

If you want to set items that do not exist in the [Resources] - [Basic Settings] tab, use the [Resources] - [Advanced] tab. The contents specified in the [Resources] - [Advanced] tab are directly reflected in the shell script. If you want to see the shell script, press the [View Script] button.

Figure 4.26 [Resources] - [Advanced] tab

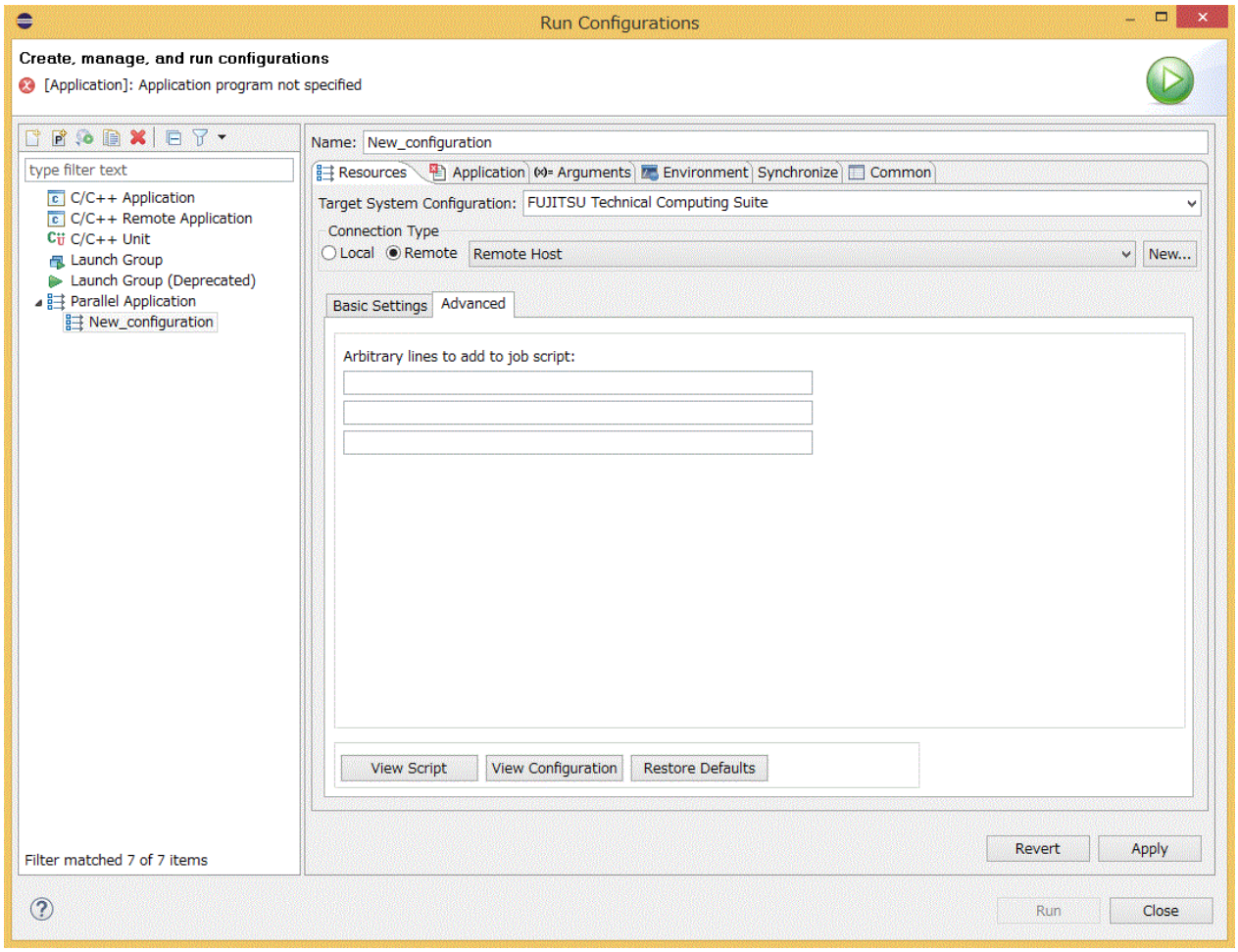


Table 4.28 Settings in [Resources] - [Advanced] tab

Item Name	Description
Arbitrary lines to add to job script:	Write the processing you want added to the job script. **Note** To execute multiple processes within a node during MPI program execution, you need to write in this item. For example, write the following to specify 4 as the number of processes executed within a node: `#PJM --mpi "proc=4"`

Figure 4.27 [Application] tab

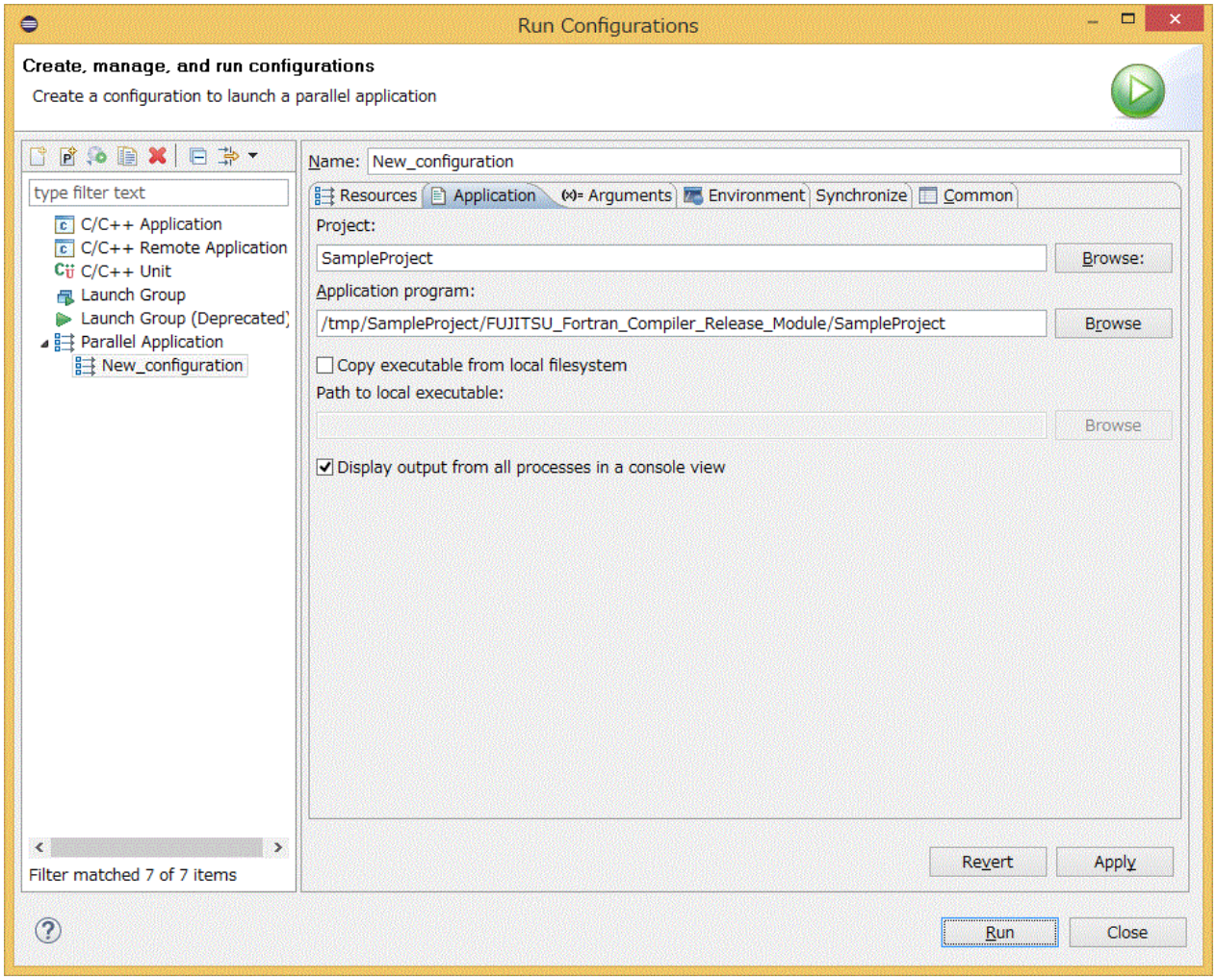


Table 4.29 Settings in [Application] tab

Item Name	Description
Project:	Specify the project to be executed.
Application program:	Specify the program to be executed.
Copy executable from local filesystem	Specify whether to specify a local file as an executable file.
Path to local executable:	Specify the path to local executable files.
Display output from all processes in a console view	Specify whether the [Console] view displays output from all processes.

Figure 4.28 [Arguments] tab

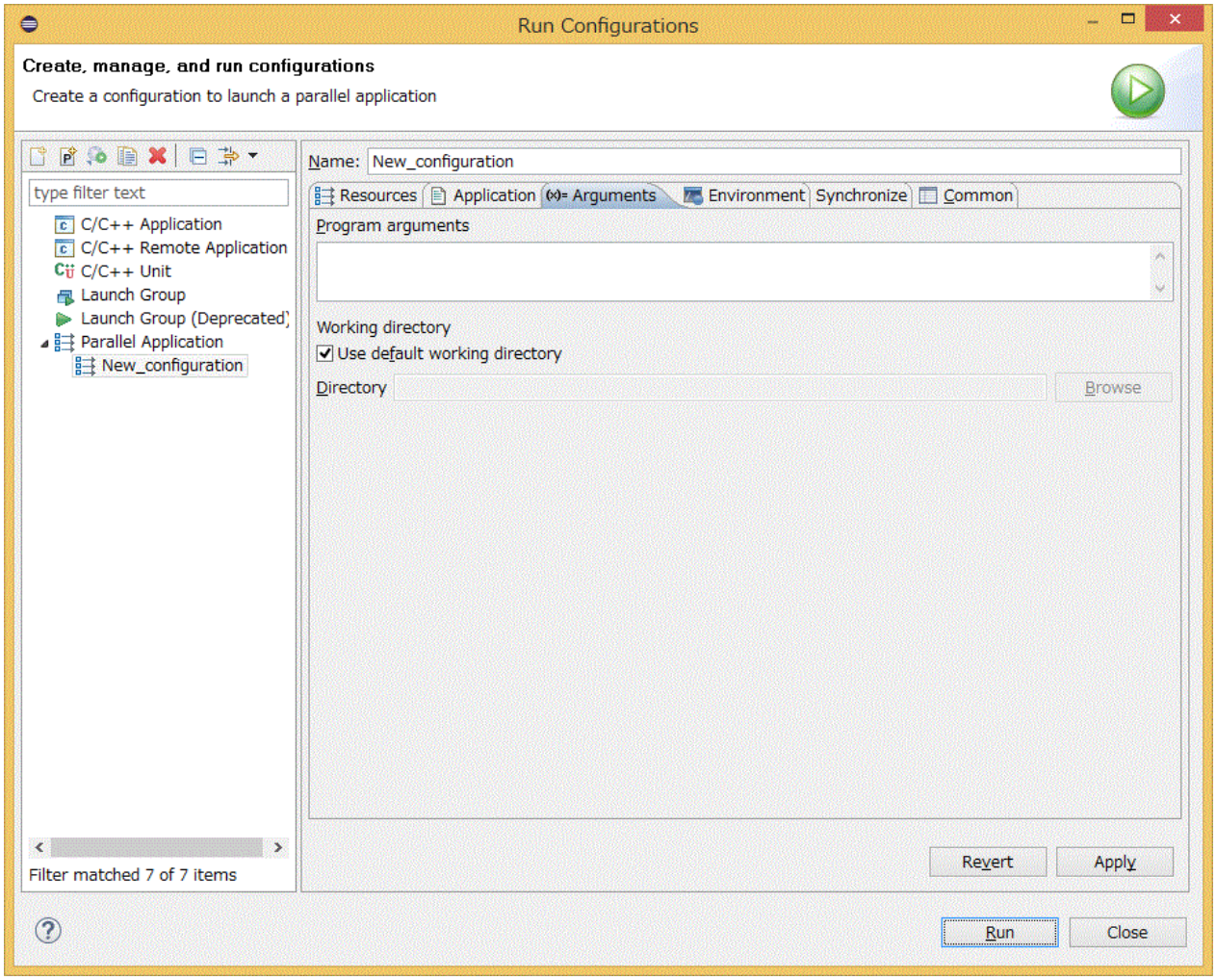


Table 4.30 Settings in [Arguments] tab

Item Name	Description
Program arguments	Specify the arguments to hand over to the program.
Use default working directory	Specifies a working directory. If this box is checked, the directory where the executable file to run exists is used as the working directory.
Directory	Specifies a working directory. The specified path is used as the working directory. If [Use default working directory] is enabled, the setting specified in [Directory] is ignored.

Figure 4.29 [Environment] tab

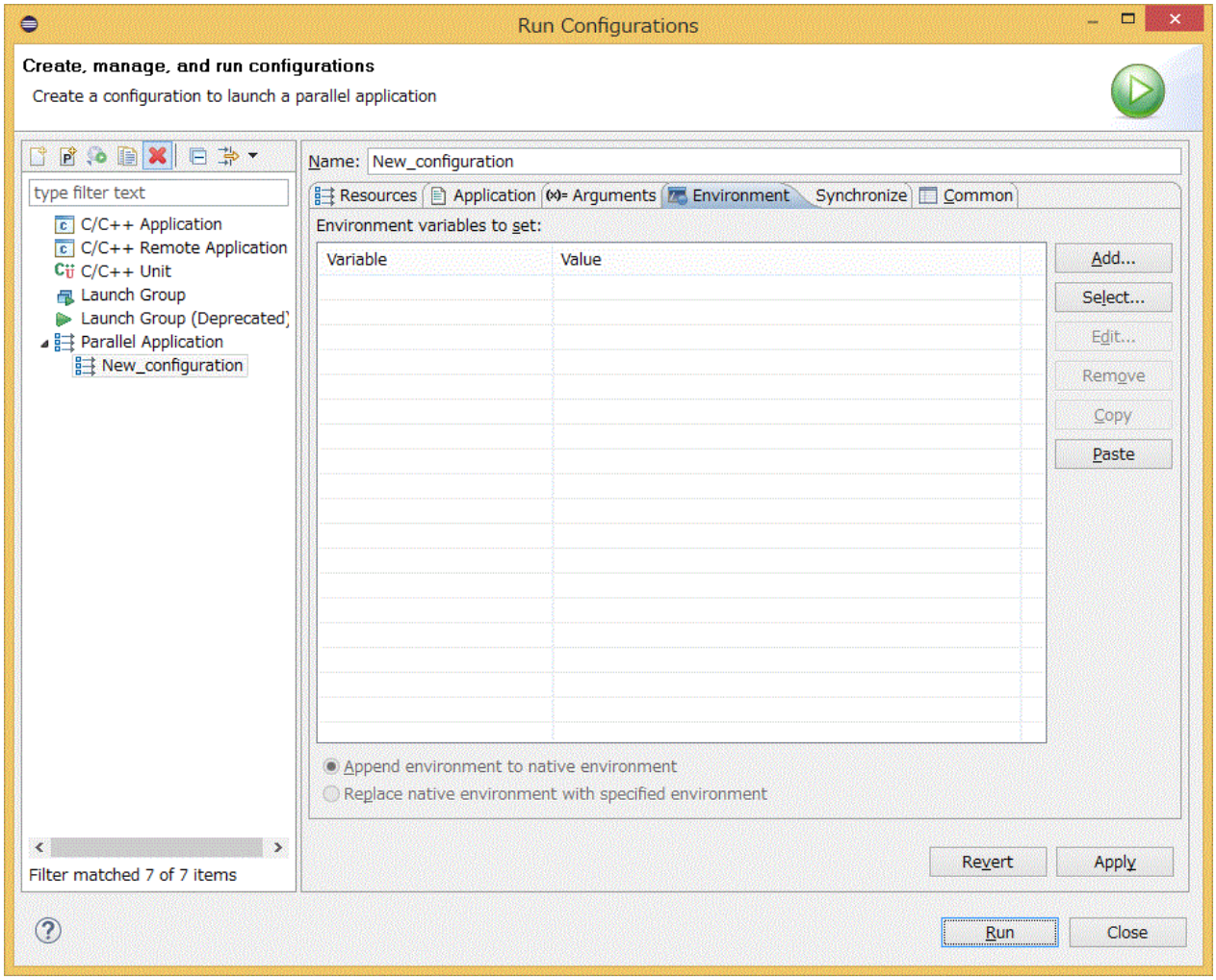
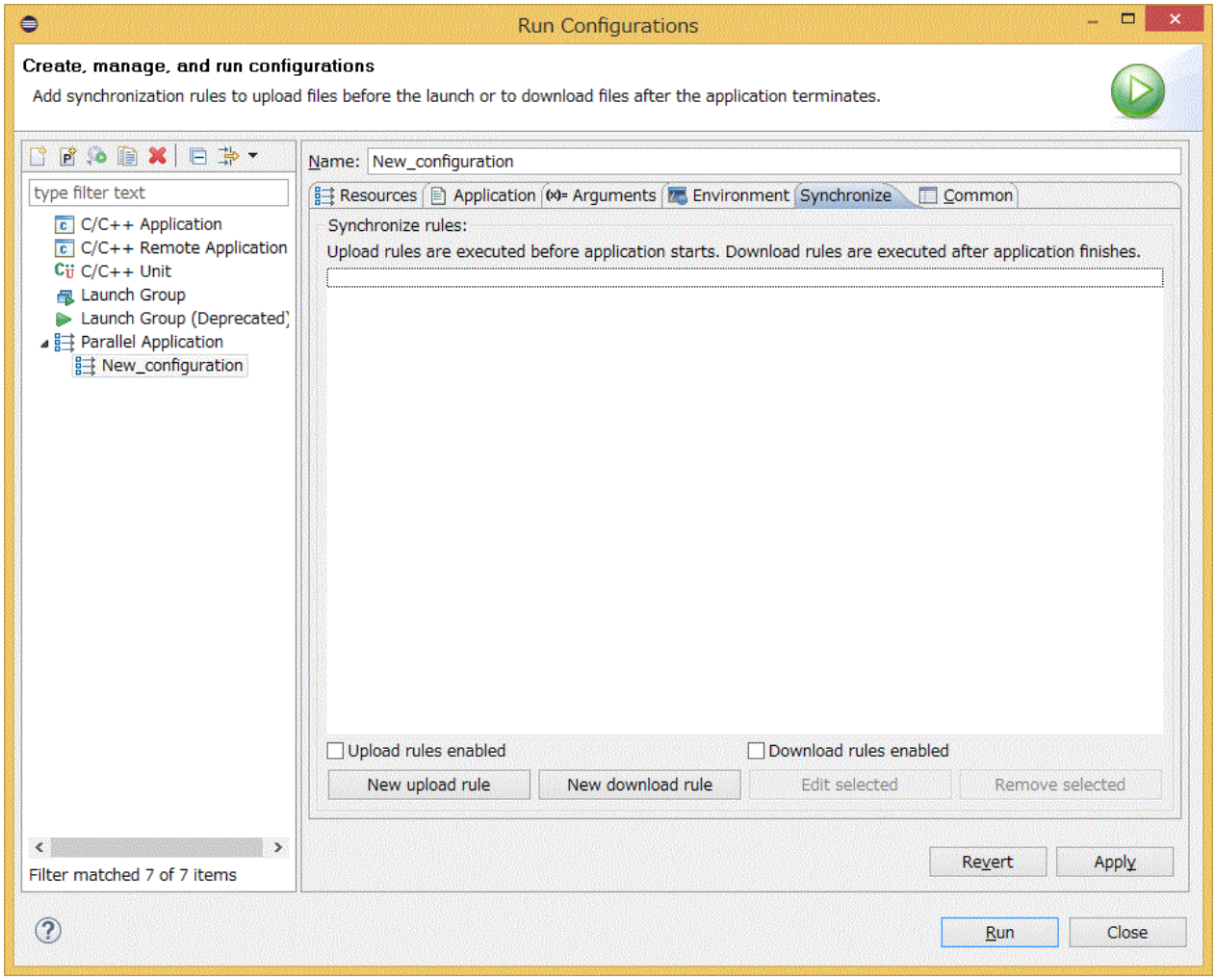


Table 4.31 Settings in [Environment] tab

Item Name	Description
Environment variables to set:	Specify the environment variables to set for the job script.

Figure 4.30 [Synchronize] tab



Make settings related to file sharing between the client machine and the login node. If you use "4.4 CPU Performance Analysis Report Display", you must enable [Download rules enabled]. Another use is that in the client machine, you use the file output on the login node.

Note

Do not repeatedly submit the same job that uses "4.4 CPU Performance Analysis Report Display." Doing so may cause an error due to a file sharing conflict between the client machine and login node. In that case, rerun the job alone.

Table 4.32 Settings in [Synchronize] tab

Item Name	Description
Synchronize rules:	Displays a list of rules created in [New upload rule] or [New download rule], which is described below.
Upload rules enabled	Enables the rules created in [New upload rule] out of the rules displayed in [Synchronize rules].
Download rules enabled	Enables the rules created in [New download rule] out of the rules displayed in [Synchronize rules].
New upload rule	Opens the [Upload Rule] window, where you create rules related to upload from the client machine to the login node.
New download rule	Opens the [Download Rule] window, where you create rules related to download from the login node to the client machine.

Figure 4.31 [Upload Rule] window

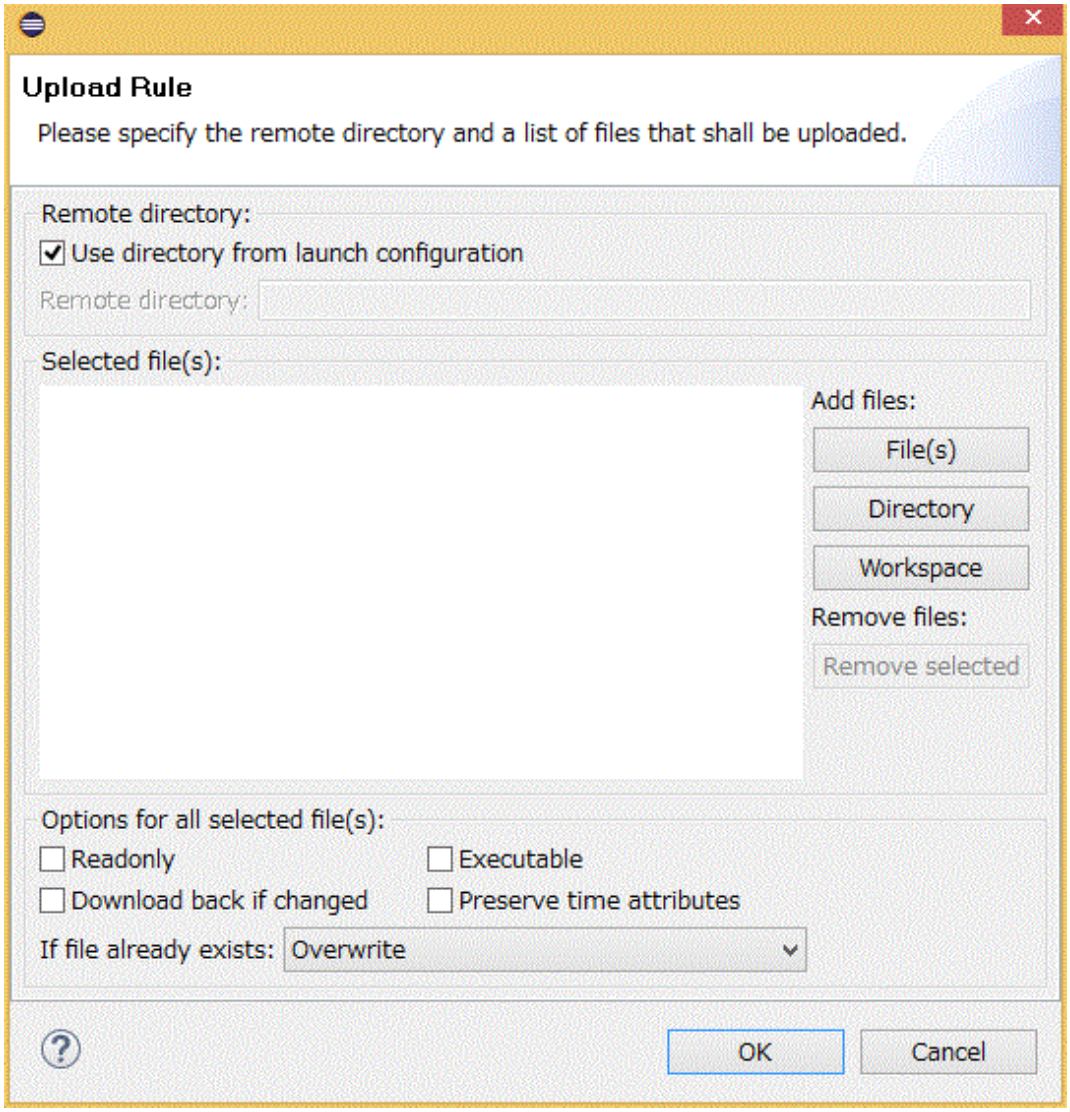


Table 4.33 Settings in [Upload Rule] window

Item Name	Description
Use directory from launch configuration	Specifies the upload destination directory. If this box is checked, the same directory as for the program specified in [Application program:] in the [Application] tab is specified.
Remote directory:	Specifies the upload destination directory. Specify an arbitrary directory. If [Use directory from launch configuration] is enabled, the setting specified in [Remote Directory] is ignored.
Selected file(s):	Selects a file , directory , or workspace to be uploaded.

Figure 4.32 [Download Rule] window

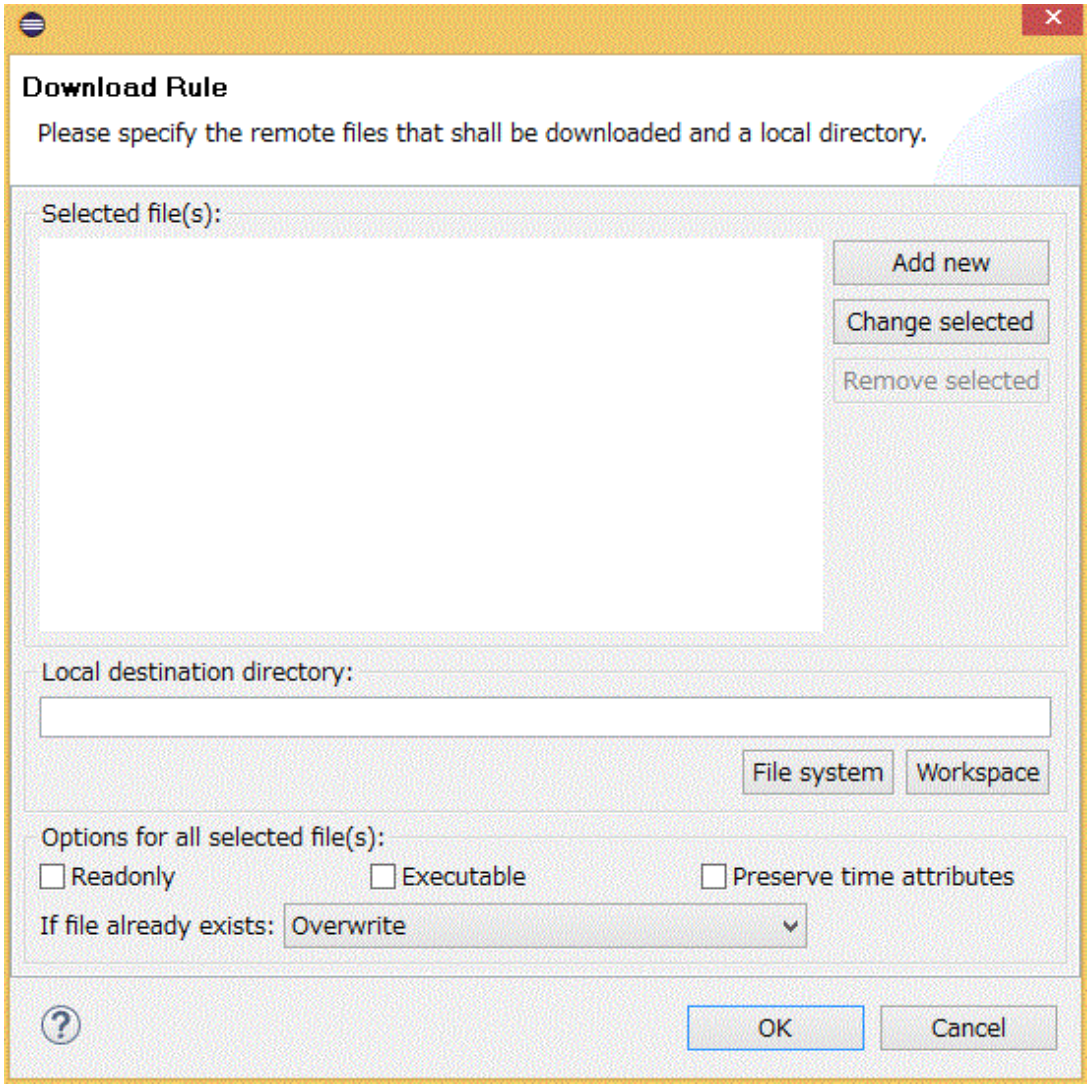


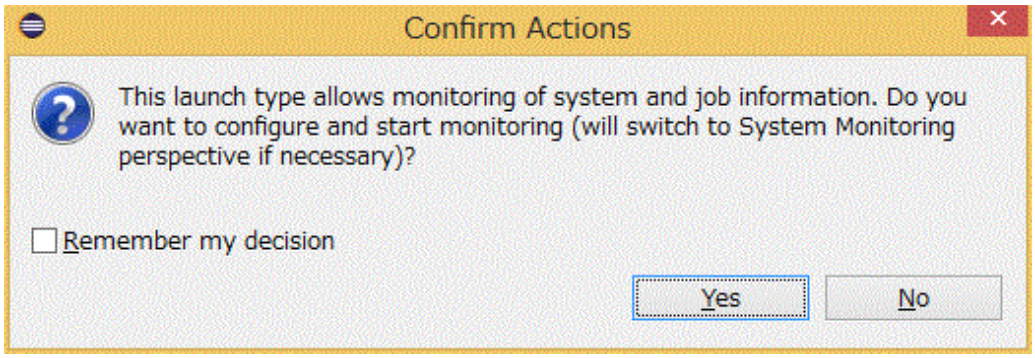
Table 4.34 Setting in [Download Rule] window

Item Name	Description

Selected file(s):	Selects a file, directory, or workspace to be downloaded.
	The following settings are required when using "4.4 CPU Performance Analysis Report Display".
	`/project_storing_directory_on_the_login_node/`
	`FUJITSU_XXX_Compiler_Release_Module/csv (Directory)`
	Directory that contains Profiler results (CSV files) for use with CPU Performance Analysis reports. The *FUJITSU_XXX_Compiler_Release_Module* varies depending on the "4.1.5 Specifying Compiler Options".
	`/installation_path/misc/cpupa/cpu_pa_report.xlsm (File)`
	CPU performance analysis report file. For details on "*installation_path*", contact the system administrator.

Local destination directory:	Specifies the download destination. When using "4.4 CPU Performance Analysis Report Display", press the [Workspace] button to specify the project as the download destination.

4. A confirmation window appears and asks whether to open the [System Monitoring] perspective. Clicking [Yes] button will automatically start the [System Monitoring] perspective after the job is submitted. For details on the [System Monitoring] perspective, see "4.3 Job Status Check and Operation".



5. The job is submitted.

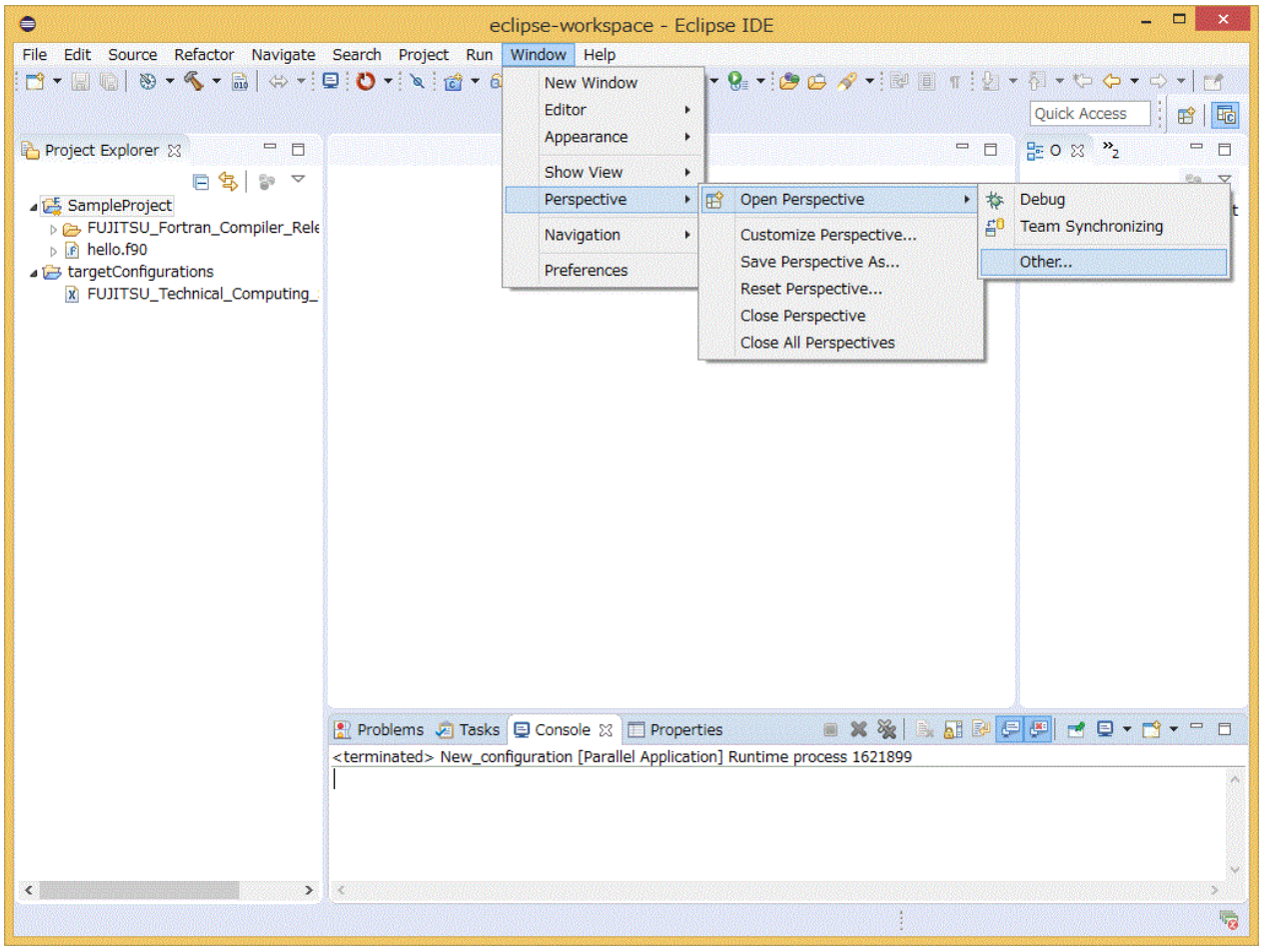
Note

The settings on the [Common] tab are unavailable even when they have set values.

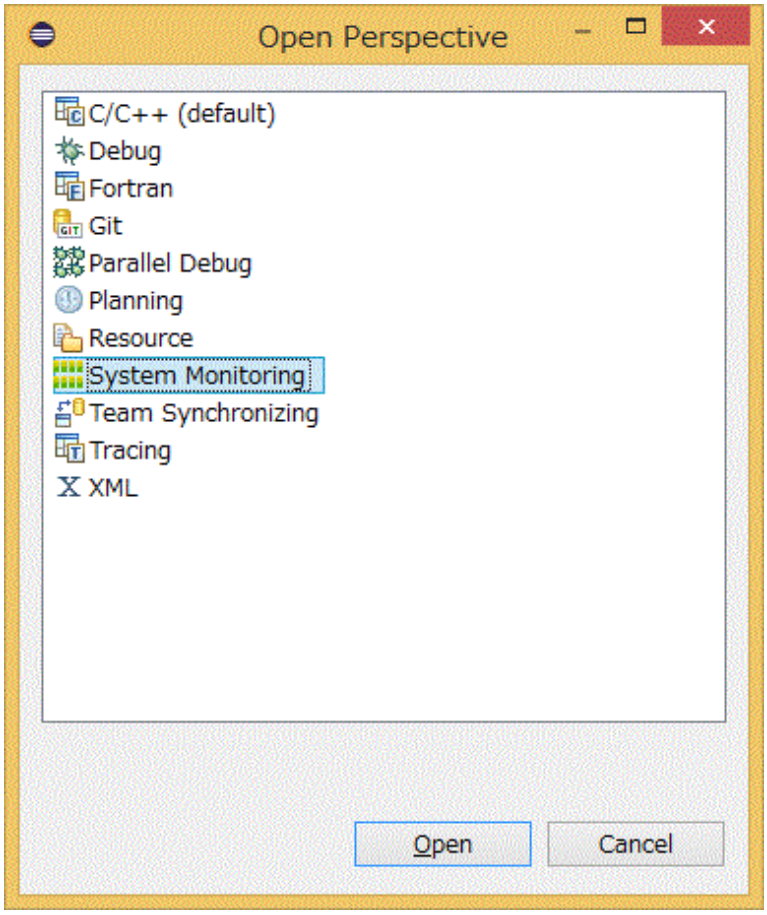
4.3 Job Status Check and Operation

Use the [System Monitoring] perspective to display the status of a job on a compute node and the node status.

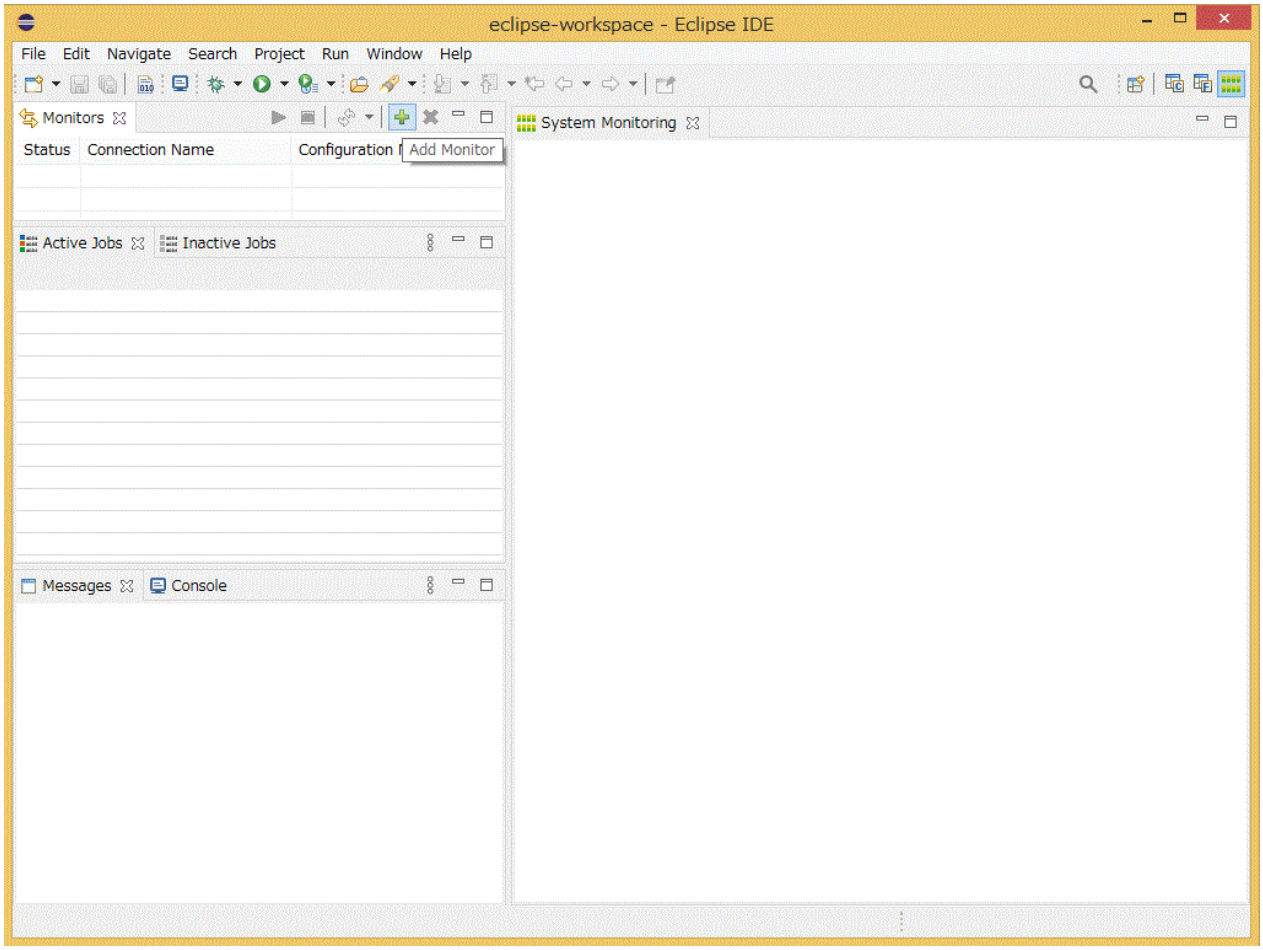
1. Click [Window] - [Perspective] - [Open Perspective] - [Other...] on the menu bar.



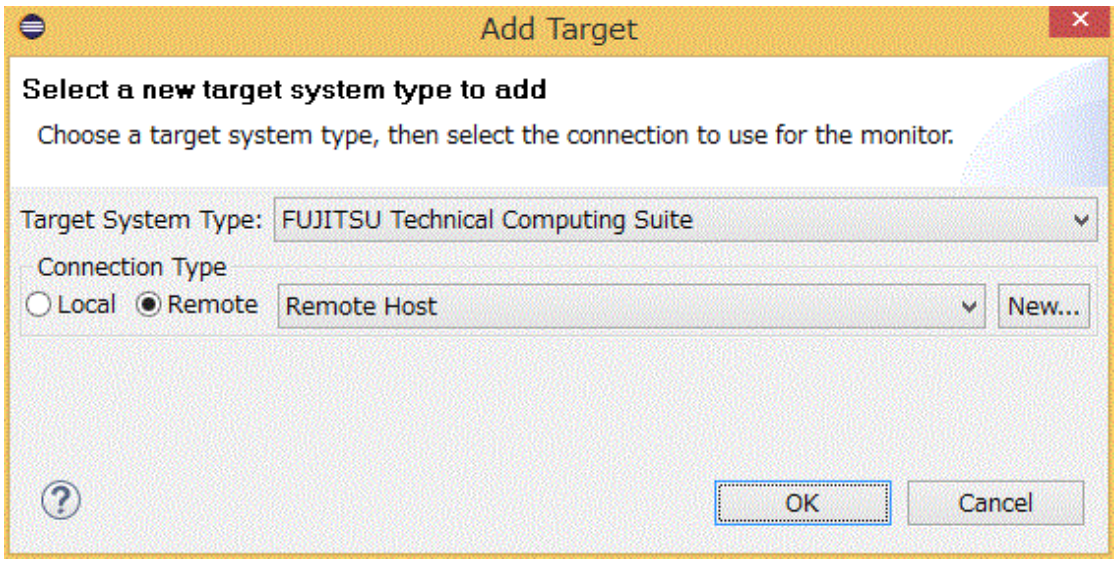
2. Select [System Monitoring] from the [Open Perspective] window, and click [Open] button.



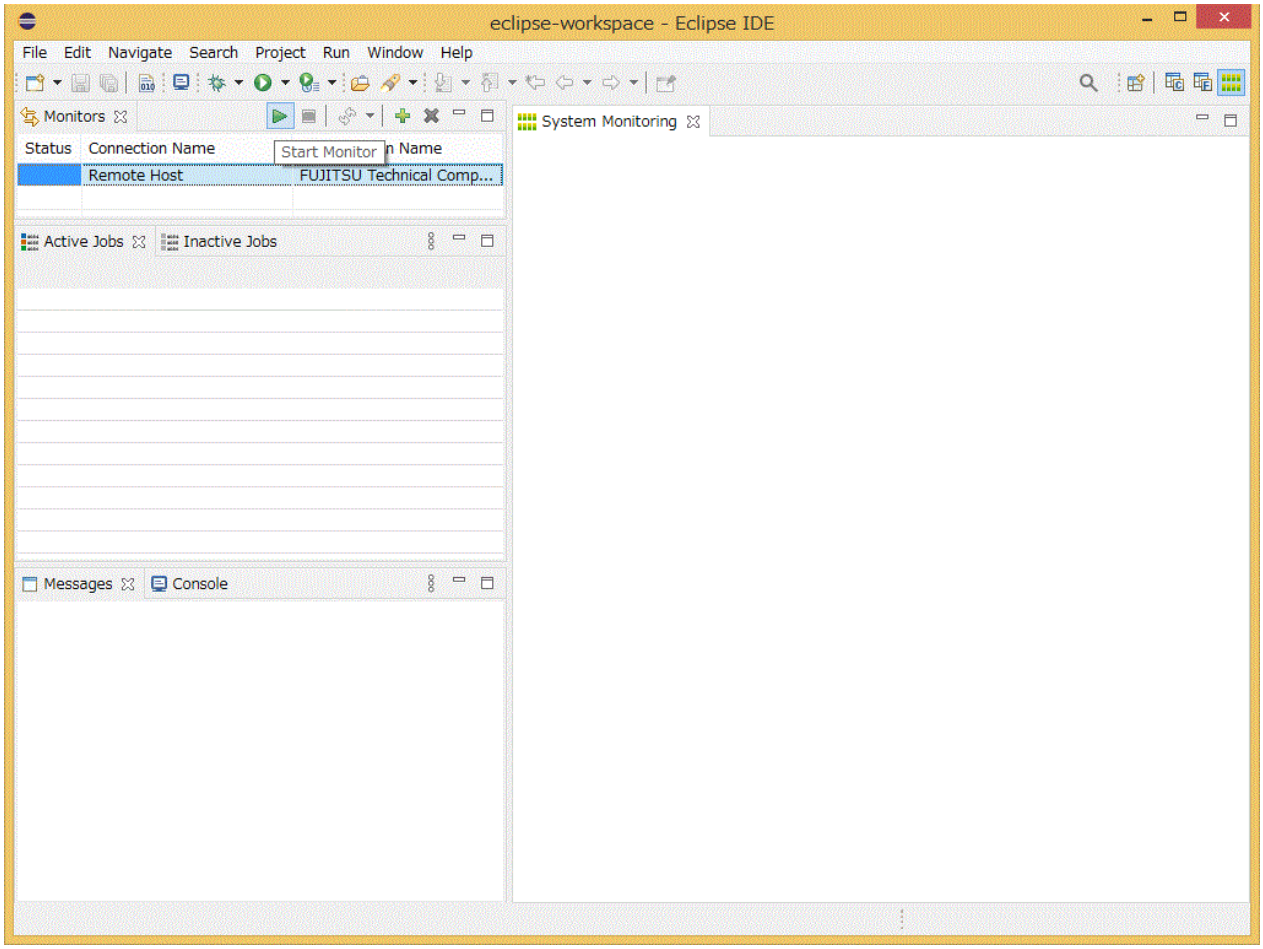
3. The workbench view switches to the [System Monitoring] perspective. Click the green plus-icon button (Add Monitor).



4. Select "FUJITSU Technical Computing Suite" from the [Target System Type] pull-down menu in the [Add Target] window. Select [Remote], and then select a name from the pull-down menu. Select the one created for [Connection name] in "2.3.1 Connecting to the Login Node (Remote System)". After completing all the settings, click [OK] button.



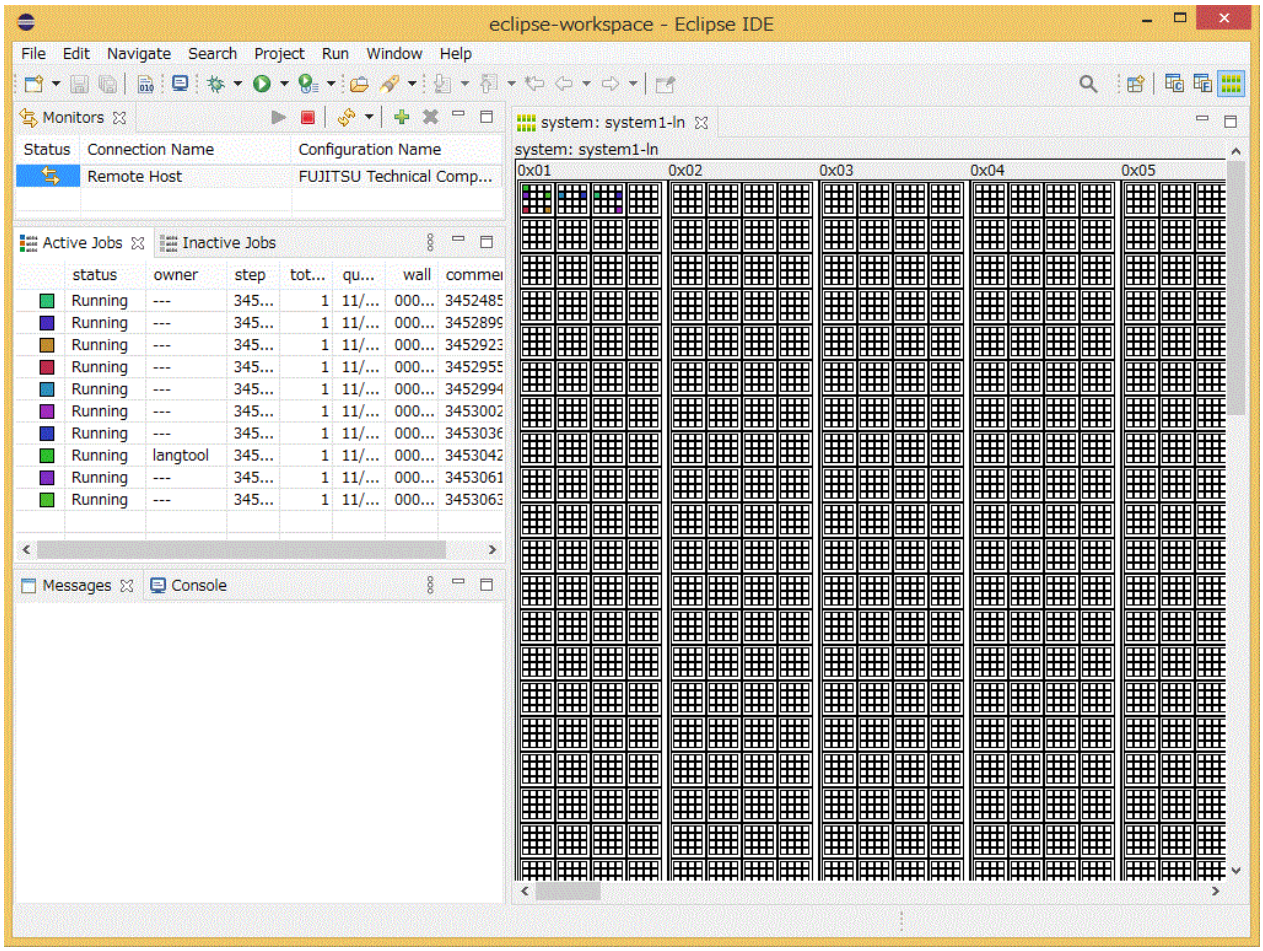
5. The setting is added to the [Monitors] view. While the setting you want enabled is selected, click the green play button (Start Monitor). The "Status" column displays a bidirectional arrow icon when a connection has been correctly established.



6. The [system] view displays monitoring results. Also, when a job is selected in the [Active Jobs] view, the [Messages] view and [Console] view display information on the selected job.

Note

The layout of the [System Monitoring] perspective depends on the contents of "layout _ default _ TC _ SUITE.xml".



Note

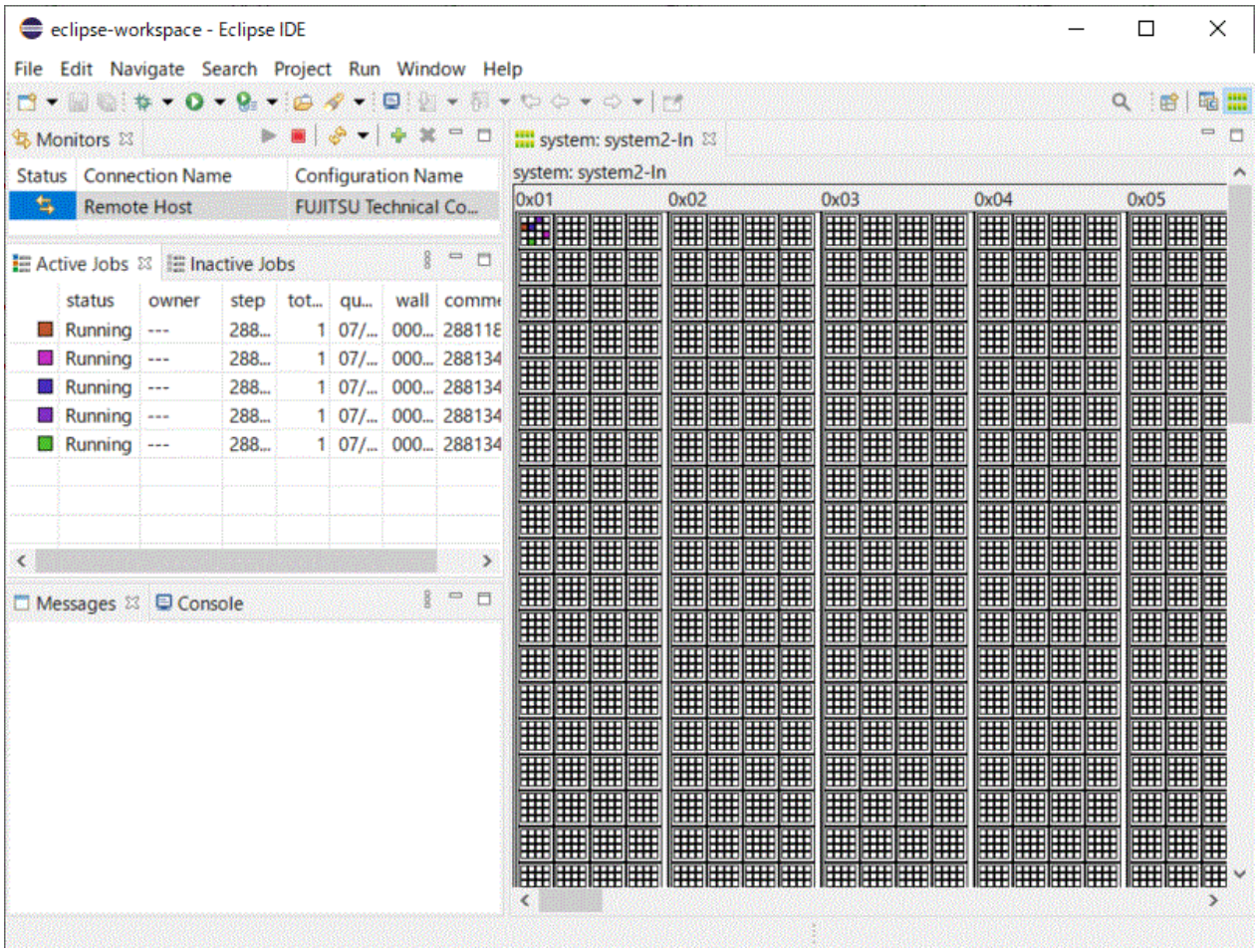
The perspective displays information obtained from the Job Operation Software. However, depending on the timing, "?" may be displayed for information that cannot be obtained. To solve this problem, refresh the [System Monitoring] perspective.

From the viewpoint of privacy protection, "---" is displayed for the owner, group, and name of other users' jobs.

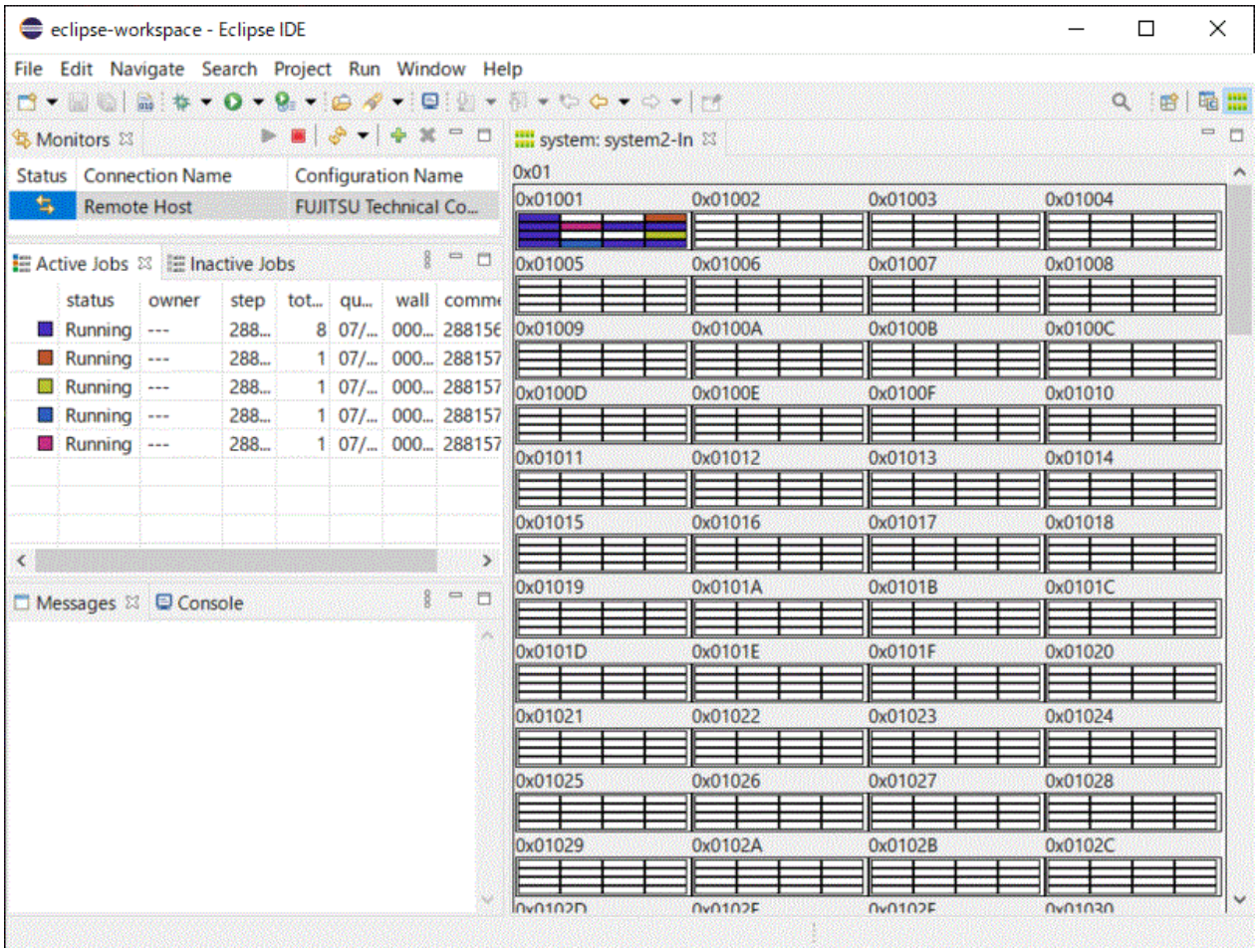
4.3.1 Displaying the Usage of Computational Resources

The [System Monitor] view virtually displays nodes or cores within nodes in color based on the usage of computational resources. You can drill down on the screen at four levels to visually check the usage of computational resources.

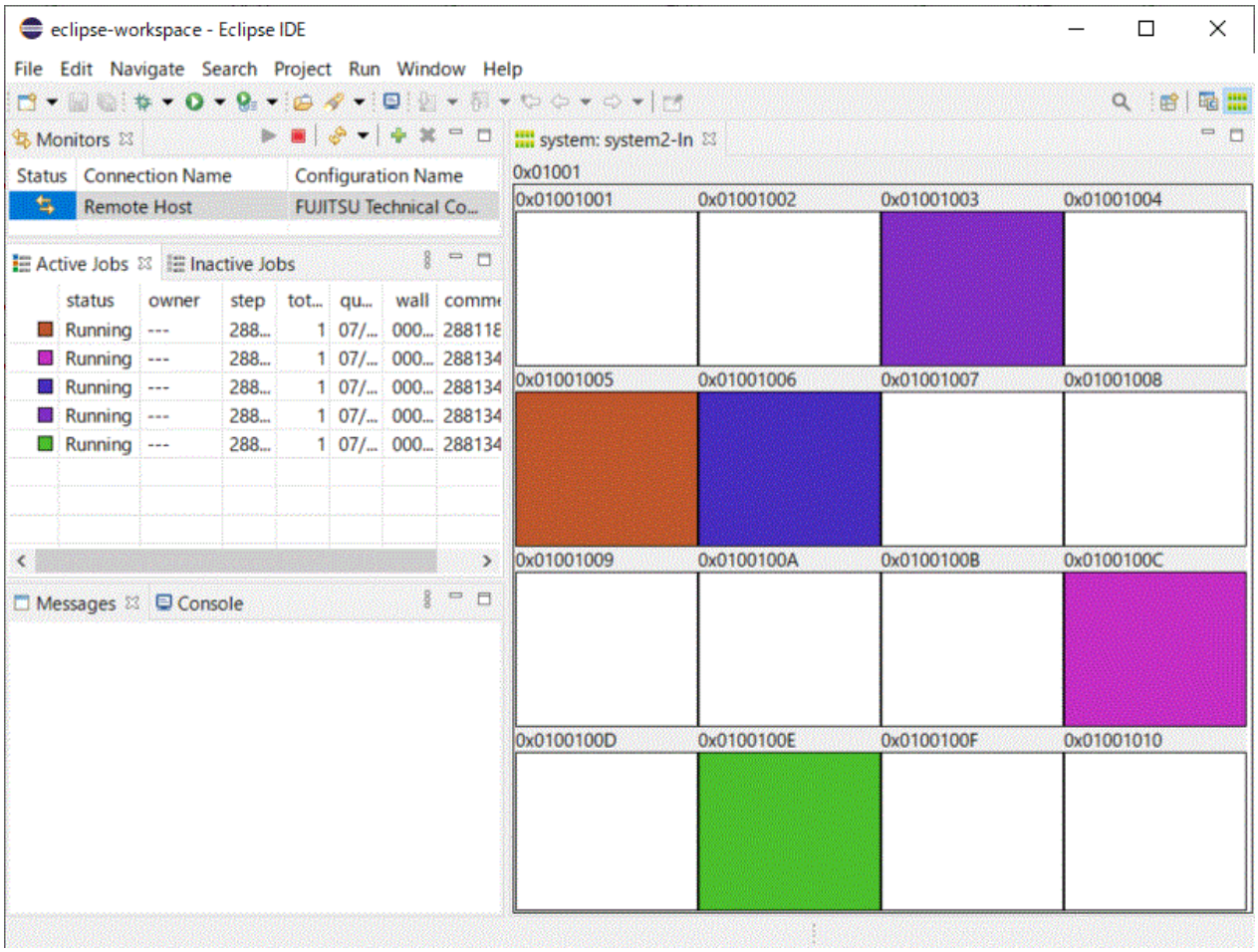
1. In the [System Monitor] view displaying the first level, you can check the usage of nodes throughout the system. Click a title bar displaying a number to switch to the second level of display for the selected number. Click the title bar at the top of the [System Monitor] view to switch to the initial screen of the first level.



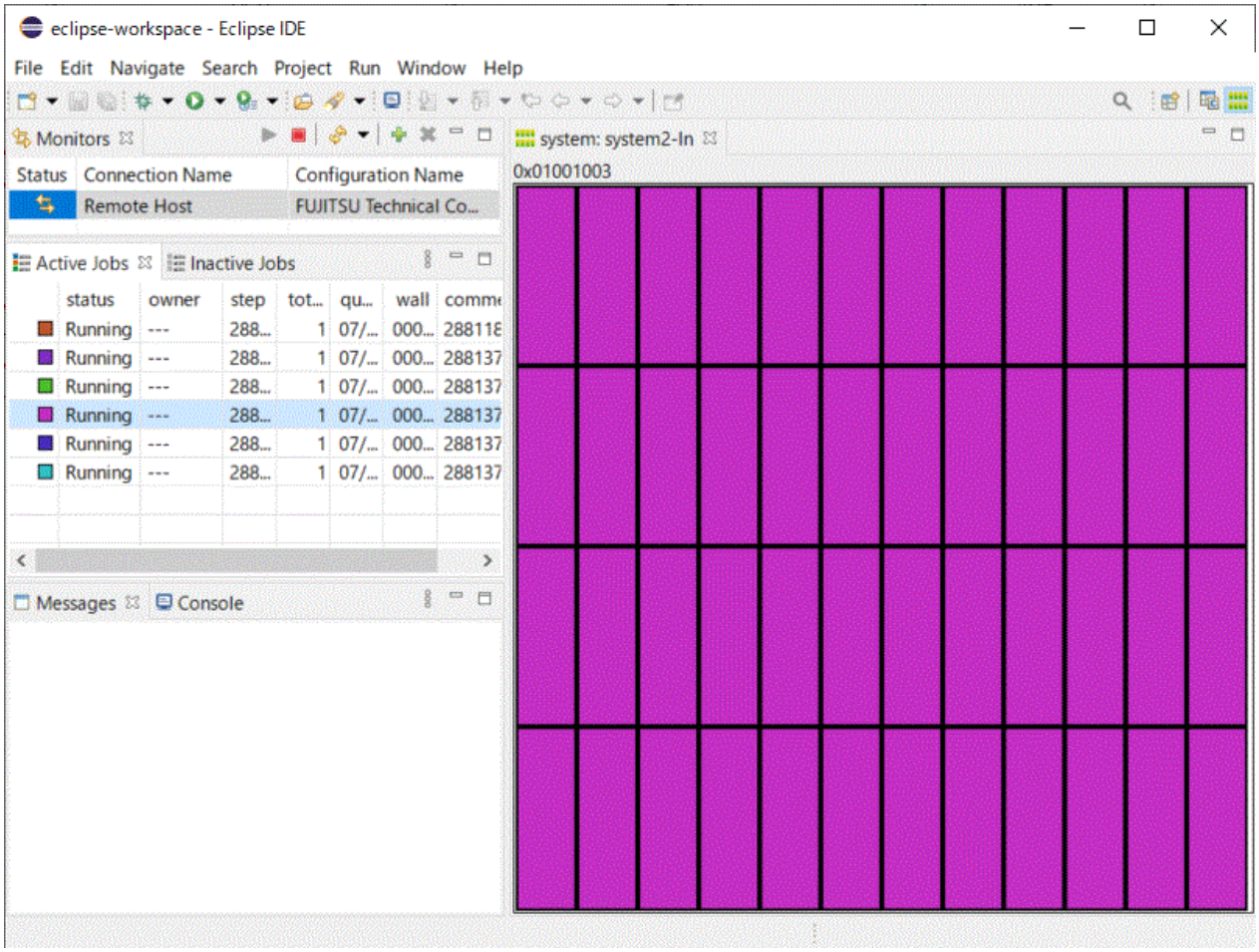
2. In the [System Monitor] view displaying the second level, you can check the usage of nodes in multiple node clusters (16 nodes combined into one). Click a title bar displaying a number to switch to the third level of display for the selected number. Click the title bar at the top of the [System Monitor] view to switch to the first level of display.



3. In the [System Monitor] view displaying the third level, you can check the usage of the 16 nodes in the node cluster selected in the second level. Click a title bar displaying a number to switch to the fourth level of display for the selected number. Click the title bar at the top of the [System Monitor] view to switch to the second level of display.



4. In the [System Monitor] view displaying the fourth level, you can check the usage of the 48 cores in the node selected in the third level. Click the title bar at the top of the [System Monitor] view to switch to the third level of display.



Note

In the [System Monitor] view displaying the fourth level, the node is colored based on how many of the 48 cores in the node are in use. Therefore, when multiple jobs are running on the same node, the number of cores actually assigned to each job may differ from the number of cores displayed as colored.

4.3.2 Operating a Job in a Job View

You can operate jobs displayed in the [Active Jobs] or [Inactive Jobs] view.

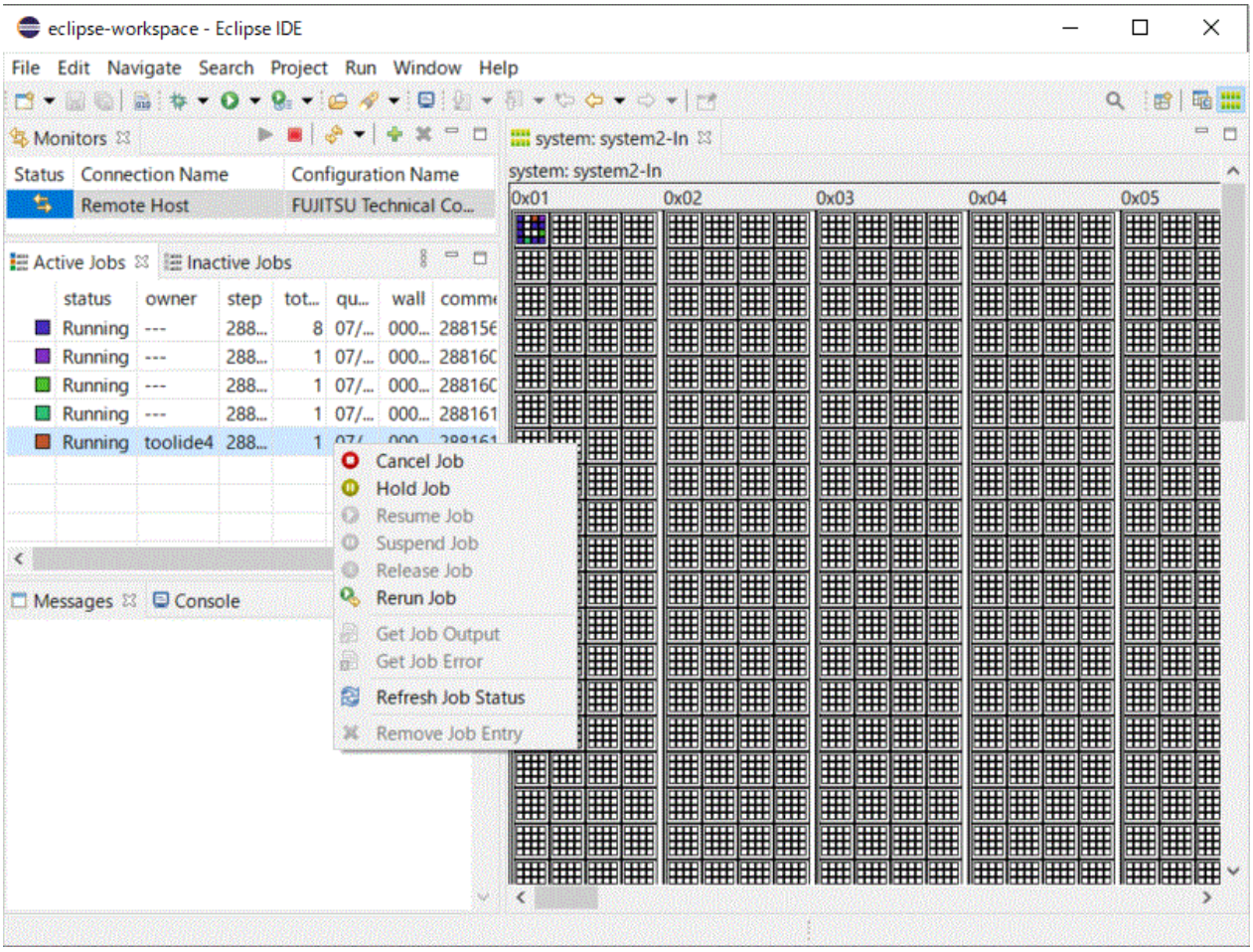
Note

The job views display your jobs preferentially. For this reason, other users' jobs may be omitted from the display.

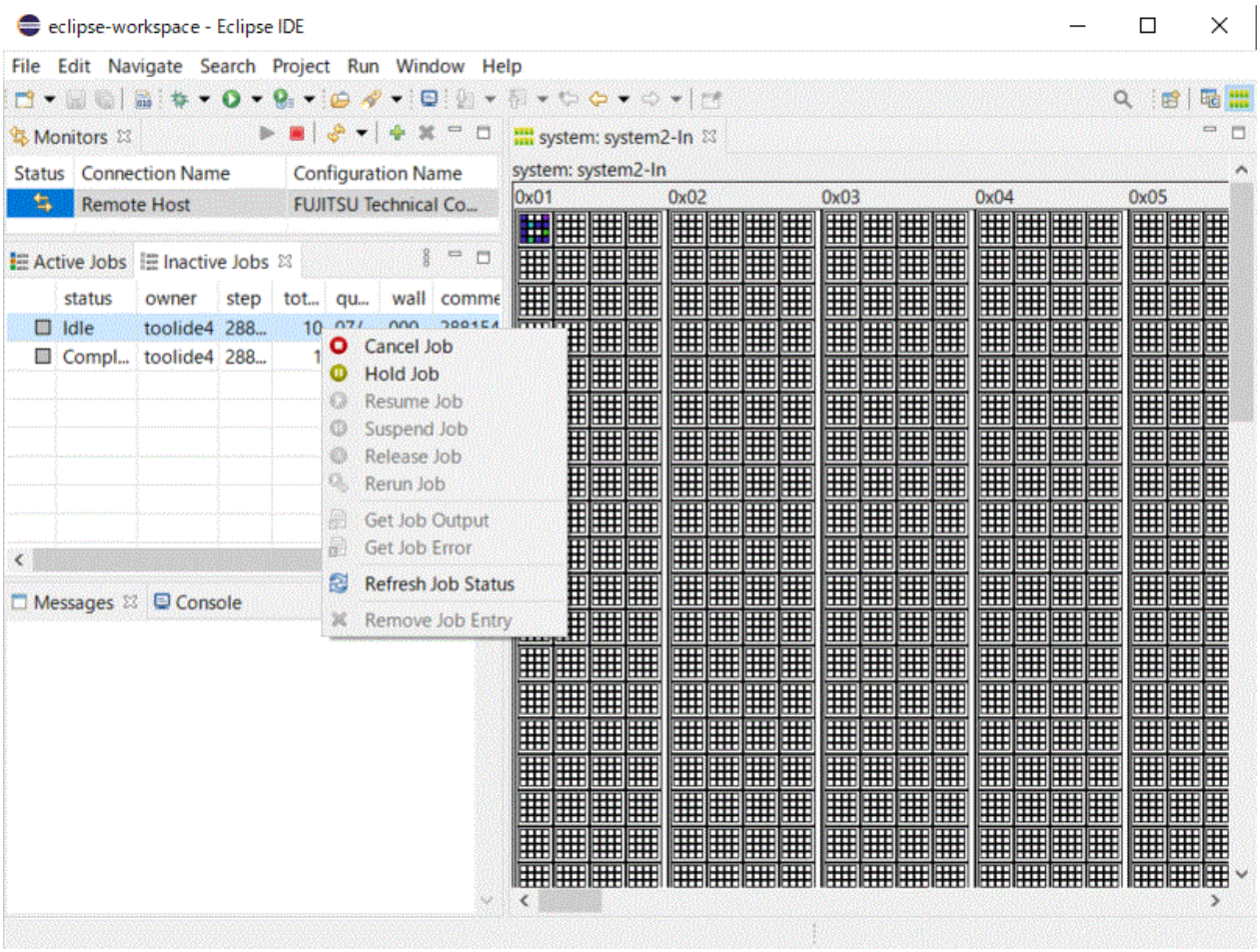
The [Active Jobs] view displays jobs whose job status is "Running." Right-click a job displayed in the [Active Jobs] view to enable operations possible for the job. The job operations cover only the jobs executed from the same Eclipse client.

Note

The job operations that are not enabled when you right-click a job are unavailable. The job operation of the [Rerun Job] button, even when clicked, does nothing. Depending on the timing, the enabled job operations other than [Rerun Job] may also do nothing, even when the job operations are selected. In such cases, wait a moment, and operate the job again.



The [Inactive Jobs] view displays jobs that are owned by you and have a job status other than "Running" and also jobs that are owned by other users and have the job status of "Idle." Right-click a job displayed in the [Inactive Jobs] view to enable operations possible for the job. The job operations cover only the jobs executed from the same Eclipse client.



4.3.2.1 Canceling the Execution of a Job

Cancel the execution of a job. Right-click the job on the [Active Jobs] or [Inactive Jobs] tab, and select [Cancel Job].

4.3.2.2 Holding a Job

Hold a job. Right-click the job on the [Active Jobs] or [Inactive Jobs] tab, and select [Hold Job].

4.3.2.3 Releasing a Job

Release a job. Right-click the job on the [Inactive Jobs] tab, and select [Release Job].

4.3.2.4 Refreshing the Job Status

Update the internal status of a job. Right-click the job on the [Active Jobs] or [Inactive Jobs] tab, and select [Refresh Job Status].

Note

The operation of refreshing the job status is usually unnecessary. Only when the operation that can be normally selected cannot be selected, refresh the job status.

4.3.2.5 Removing a Job Entry

Hide a completed job. Right-click the job on the [Inactive Jobs] tab, and select [Remove Job Entry].

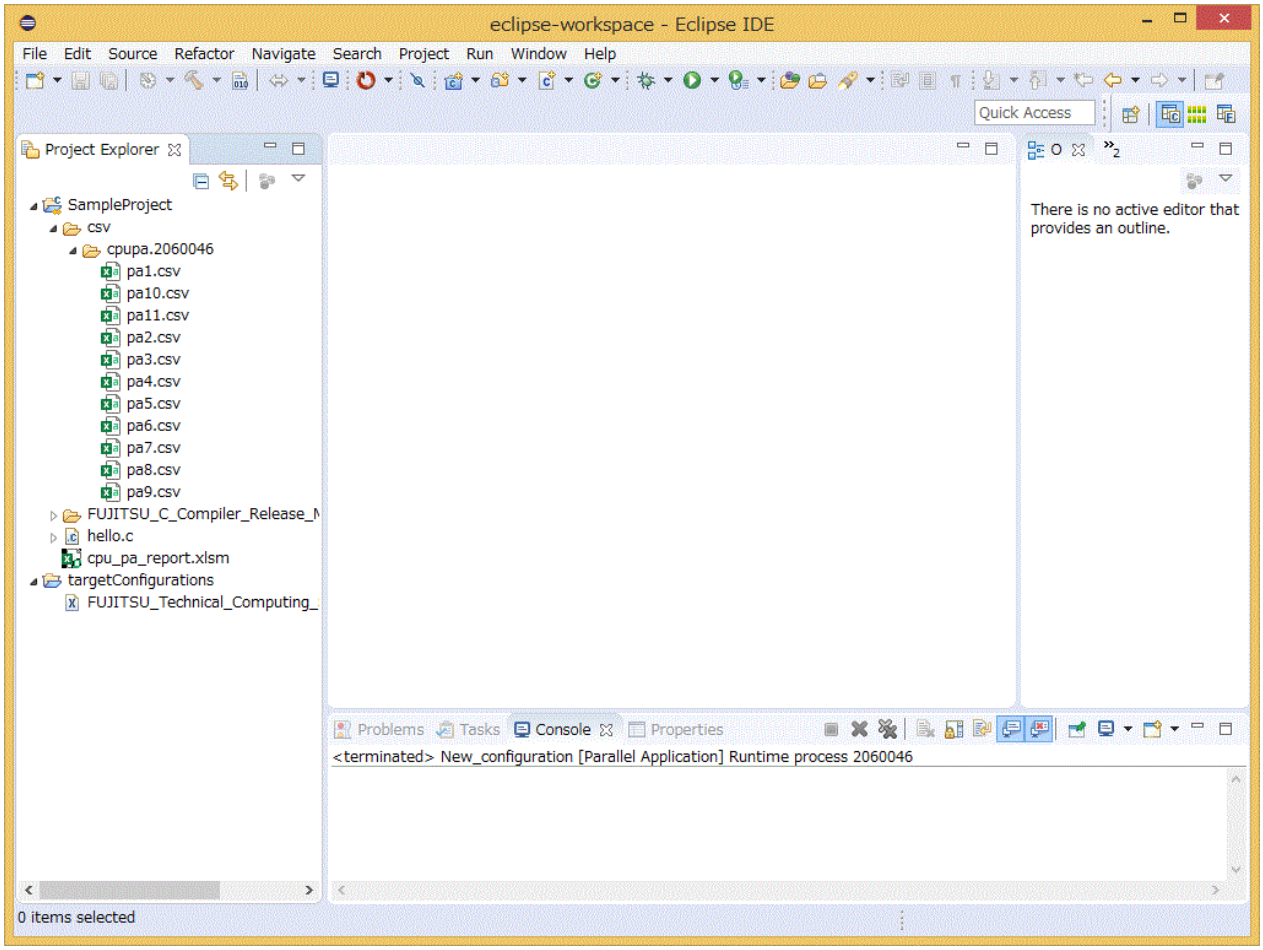
4.4 CPU Performance Analysis Report Display

The following describes the procedure for displaying a CPU Performance Analysis Report from Eclipse. For details on CPU Performance Analysis Report, see the "Profiler User's Guide".

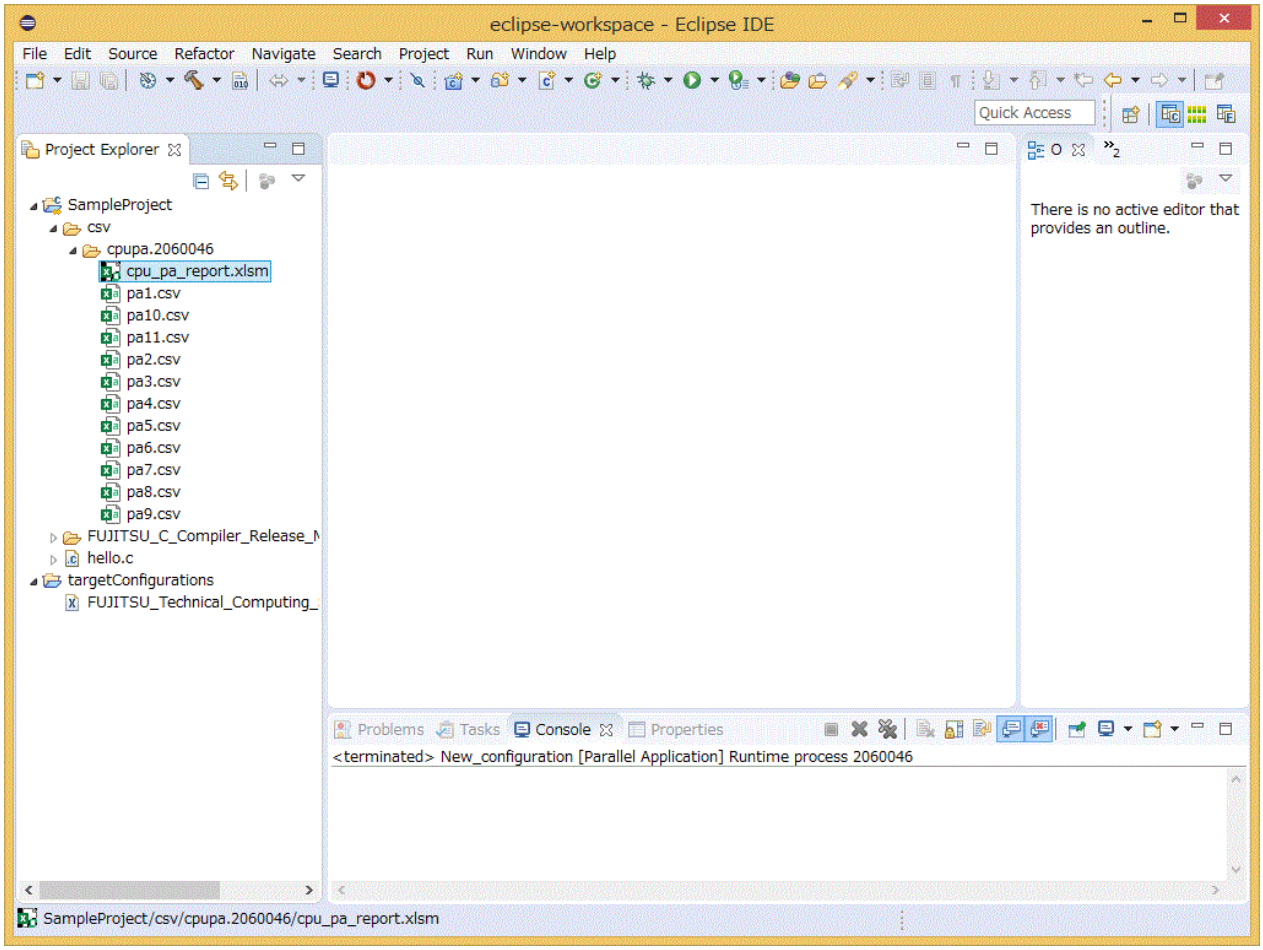
Note

To display a CPU Performance Analysis Report, Microsoft Excel must be installed on the client machine where Eclipse is running. For other cautions when using CPU Performance Analysis report, see "Profiler User's Guide".

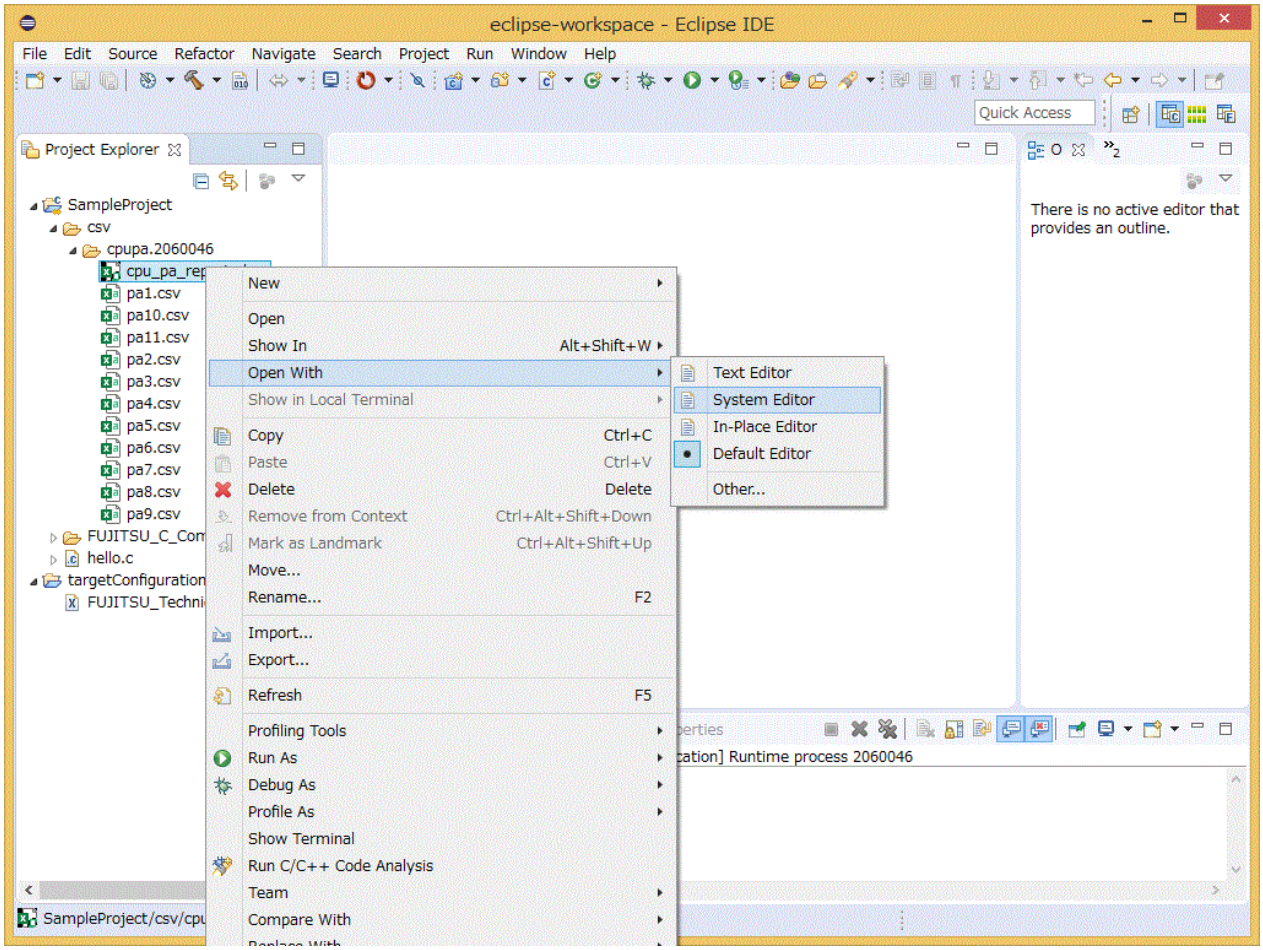
1. From "4.2 Job Submission," execute the program for which you want to create a CPU Performance Analysis Report. Make settings for [CPU Performance Analysis] in "Table 4.27 Settings in [Resources] - [Basic Settings] tab" according to the type of the CPU Performance Analysis Report you want to create. In order to transfer the execution results onto the client machine, be sure to make settings in "Table 4.32 Settings in [Synchronize] tab".
2. Check the [step] value of the submitted job from the [Active Jobs] tab of the [System Monitoring] Perspective. (The CSV file used for CPU performance analysis reports is output to a directory named "cpupa.[*step value*]".)
3. Display [Project Explorer] when the job is finished. The directories and files set in "Table 4.32 Settings in [Synchronize] tab" are added. Refresh the display to add the directories and files set in "Table 4.32 Settings in [Synchronize] tab".



4. Drag and drop the CPU Performance Analysis Report file (cpu_pa_report.xlsm) into the directory where the CSV file of the measurement results is stored. The directory name shall be csv/cpupa.[*step value*] .



5. Select and right-click the CPU performance analysis report file (cpu_pa_report.xlsm) in [Project Explorer]. Select [Open With] - [System Editor] from the displayed menu.



6. The CPU Performance Analysis Report is displayed. For information about using the CPU Performance Analysis report, see "Profiler User's Guide".

Glossary

This section provides explanations of the Eclipse terms used in this manual. For details on terms and descriptions not covered in the section, see Help for Eclipse, the official site of the Eclipse Foundation, commercially available instruction manuals, and other sources.

Workbench

Workbench refers to the entire main screen of Eclipse.

View

View refers to a subwindow displayed on the workbench. Some views have toolbars or menus specific to them. An operation performed by using a view-specific toolbar or menu affects only items of the relevant view.

Perspective

Perspectives are definitions of sets and layouts of views displayed on the workbench.

Workspace

A workspace is a storage location for development assets and the user's work status. It retains the status of work performed by the user, workbench setting information, etc. You can create multiple workspaces but can open only one workspace on the workbench at a time. Development assets are managed within a workspace in units called projects. You can create multiple projects in a workspace.

Toolchain

A collection of tools that generates binaries from a source code. This collection consists of an editor, a compiler, an assembler, a linker, etc. In this document, Toolchain is also used to refer to the configuration information of tools used for a project.

Synchronized project

A synchronized project consists of files mirrored in the local system and one or more remote systems. The files are edited on the local system and synchronized with active remote systems at the timing when each file is changed, created, or deleted. Since a remote system corresponds to a login node in this document, it is hereinafter referred to as a login node. The terminal that corresponds to the local system is referred to as the client machine.