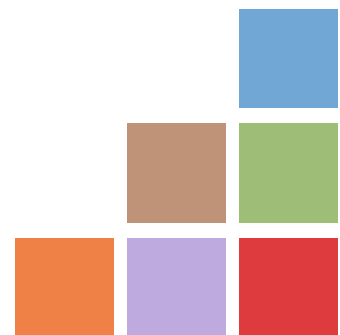




**COBOL-IT® Developer Studio
Getting Started
The Remote System Perspective
Version 2.0**



Contents

ACKNOWLEDGMENT	3
COBOL-IT DEVELOPER STUDIO TOPICS	4
Introduction and License Terms	4
COBOL-IT Developer Studio License terms.....	4
Dependencies	5
Dependencies and Comments	5
The COBOL-IT Developer Studio Distribution	5
THE REMOTE SYSTEMS PERSPECTIVE.....	7
Configuring the Remote Systems Perspective	7
Window>Preferences>General>Network Connections>SSH2.....	7
Window>Preferences>General>Workspace	8
Window>Preferences>COBOL>License file path	8
Window>Preferences>Run/Debug>Perspectives>.....	9
Opening the Remote System Explorer	10
Window>Open Perspective>Other>Remote System Explorer	10
Using The Remote System View	11
Remote System View>Define a connection to remote system	11
Enter Host Name, or IP Address.....	11
Remote System View>[Host Name]>[Dropdown]>Connect	12
View of the Remote Systems view when logged in.....	13
File>New>Project>COBOL>COBOL Project	14
Working in a remote project	15
Remote Run in Debug.....	18
Re-opening a Remote Project	20
FAQ.....	20
Copy and Paste files from a local machine to a remote machine	20

Acknowledgment

Copyright 2008-2020 COBOL-IT S.A.R.L. All rights reserved. Reproduction of this document in whole or in part, for any purpose, without COBOL-IT's express written consent is forbidden.

Microsoft and Windows are registered trademarks of the Microsoft Corporation. UNIX is a registered trademark of the Open Group in the United States and other countries. Other brand and product names are trademarks or registered trademarks of the holders of those trademarks.

Contact Information:

The Lawn
22-30 Old Bath Road
Newbury, Berkshire, RG14 1QN
United Kingdom
Tel: +44-0-1635-565-200

COBOL-IT Developer Studio Topics

Introduction and License Terms

This document describes how to use the **COBOL-IT Developer Studio Remote System Perspective**. **COBOL-IT Developer Studio** is COBOL-IT's eclipse-based development environment, designed to support users of the **COBOL-IT Compiler Suite**.

COBOL-IT Developer Studio License terms

The copyright for the COBOL-IT Developer Studio® is wholly owned by COBOL-IT. Any unauthorized reproduction of the software without the express written consent of COBOLIT is prohibited.

For more information, please contact us at: contact@cobol-it.com

COBOL-IT Corporate Headquarters are located at

The Lawn
22-30 Old Bath Road
Newbury, Berkshire, RG14 1QN
United Kingdom
Tel: +44-0-1635-565-200

COBOL-IT, COBOL-IT Compiler Suite, CitSQL, CitSORT, and COBOL-IT Developer Studio are trademarks or registered trademarks of COBOL-IT.

Eclipse is a trademark of the Eclipse Foundation.

IBM, and AIX are registered trademarks of International Business Machines Corporation. Linux is a registered trademark of Linus Torvalds.

Windows, Visual Studio, and Visual Studio Express are registered trademarks of Microsoft Corporation.

Java and Solaris are registered trademarks of Sun Microsystems, Inc.

UNIX is a registered trademark of The Open Group

HP is a registered trademark of Hewlett Packard, Inc.

Red Hat is a registered trademark of Red Hat, Inc.

All other trademarks are the property of their respective owners.

Dependencies

Dependencies and Comments

Dependency	Comment
“C” compiler	The COBOL-IT Compiler requires a “C” compiler. While most Linux>Unix installations will include a “C” compiler, many Windows installations will not. Windows users can download the Visual Studio from www.microsoft.com .
COBOL-IT Compiler Suite	The COBOL-IT Compiler Suite, Standard Edition can be downloaded at the COBOL-IT Online Portal. For access to the COBOL-IT Online Portal, please contact your sales representative at sales@cobol-it.com .
Java Runtime Environment (JRE)	The COBOL-IT Developer Studio Kepler build can be run with the Java Runtime Environment (JRE) Version 1.6 or greater. The COBOL-IT Developer Studio Neon build can be run with the JRE Version 1.8 or greater.
Eclipse	Eclipse is included with the download of Developer Studio.

The COBOL-IT Developer Studio requires that the COBOL-IT Compiler Suite already be installed on the host platform, and that a “C” compiler be installed on the host platform.

The COBOL-IT Developer Studio is an Eclipse plug-in, and as such, requires that Eclipse be installed on the host platform. Eclipse, in turn, requires that a Java Runtime Environment (JRE) be installed on the host platform.

The COBOL-IT Developer Studio Distribution

For Windows-based installations, the COBOL-IT Developer Studio, Enterprise Edition can be downloaded from the COBOL-IT online portal with a login and password provided by your sales representative.

The COBOL-IT Developer Studio, Enterprise Edition is available with Subscription. The COBOL-IT Developer Studio, Enterprise Edition provides functionality with the installation of several Perspectives:

- Developer Studio Perspective in which users set up and build COBOL projects, using a locally installed version of the COBOL-IT Compiler Suite Enterprise Edition. The Developer Studio Perspective additionally provides access to Code Coverage and Profiling Tools.
- Debugger Perspective providing access to a feature-rich COBOL debugger both locally, and on Remote Systems

- Remote Systems Perspective, allowing use of Compiler, Runtime, and Debugger functionalities installed on remote servers.
- Git and RSEGit Perspectives, providing users with full access to the Git/Github Source Code Control System.
- Data Displayer Perspective, providing access to a tool for browsing and modifying data in indexed, sequential and relative files.
- Planning Perspective, providing access to the Mylyn Task Manager.
- For more information about the usage of Git/RSEGit, Data Displayer, Mylyn Task Manager, and Code Coverage, see the Getting Started with the Developer Studio- The Utilities Manual.
- Using the COBOL-IT Developer Studio requires a license for both the COBOL-IT Compiler Suite Enterprise Edition, and COBOL-IT Developer Suite.

The Remote Systems Perspective

The Remote Systems Perspective allows the user to connect to remote systems and perform all of the Developer Studio functions.

The Remote System Perspective is enabled with the Enterprise Edition of the Developer Studio, Compiler Suite on the local machine, and Compiler Suite on the remote machine.

Eclipse settings for the Remote System Perspective are stored in Window>Preferences>Remote Systems. No changes need to be made.

COBOL-IT uses an SSH Connection to the Remote System.

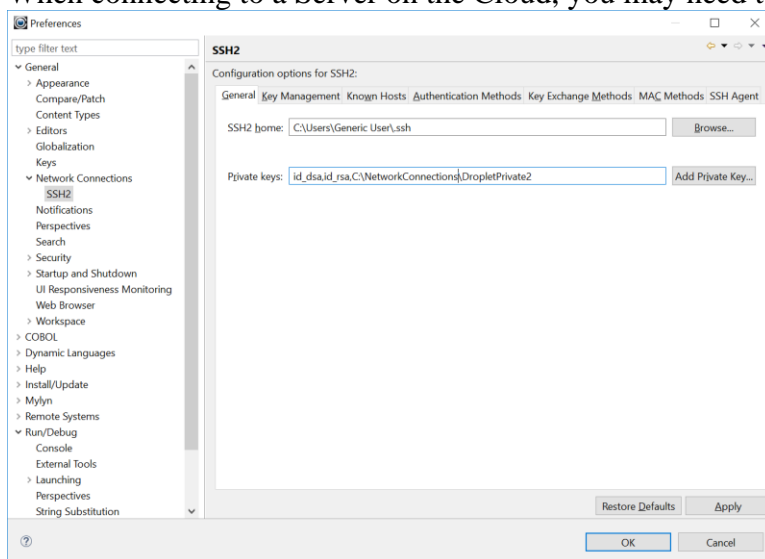
You can store the user-name and password for the login to the remote system in Window>Preferences>Remote Systems>Passwords.

Configuring the Remote Systems Perspective

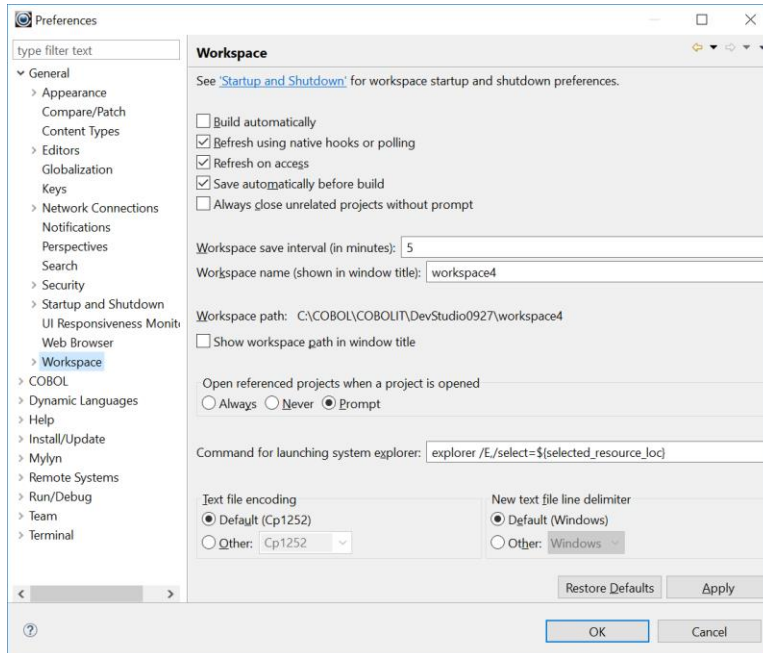
Configuring your development environment using the Window>Preferences>General, Window>Preferences>COBOL, and Window>Preferences>Run/Debug dialog screens causes the settings to be applied at the Workspace level. Settings applied at the Workspace level are applied automatically to all Projects in a Workspace.

Window>Preferences>General>Network Connections>SSH2

When connecting to a Server on the Cloud, you may need to define a Private Key.



Window>Preferences>General>Workspace



The screen print above shows the suggested settings for :

Build automatically (off).

This can also be toggled off at the main menubar
Project>Build Automatically function.

Refresh automatically (on).

When the compile generates output files into the Project folders and subfolders, it is convenient to have the folders auto-refresh, and show the files. If Refresh automatically is set to (off), you must use the manual Refresh, located on the right-click dropdown menu on the Project folder in the Navigator window.

Save automatically before build (on). This can be a matter of personal preference. It is convenient to be able to code, and then compile, without first having manually clicked on a Save function. With this function set to (on), using the Build, or Clean and Build functions will automatically cause the source to be SAVE'd before compiling.

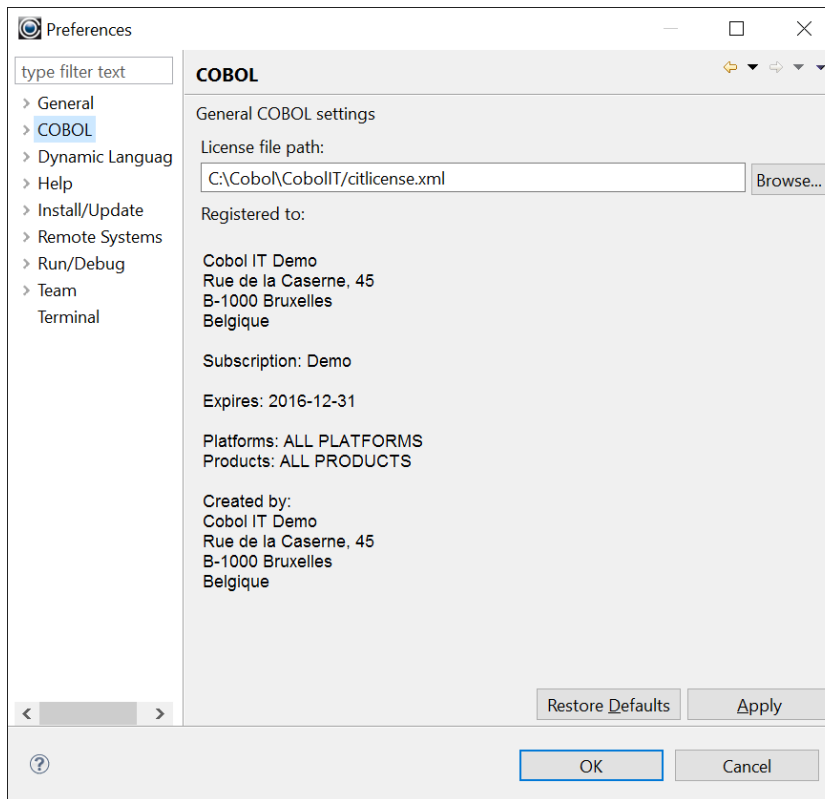
Window>Preferences>COBOL>License file path

The Enterprise Edition License file gives the user access to Remote Compile/Run/Debug capabilities. However, it does not serve as a license file to the remotely-located compiler. The remotely-located compiler is licensed separately, and that license will be validated when the

compiler is launched at the remote location.

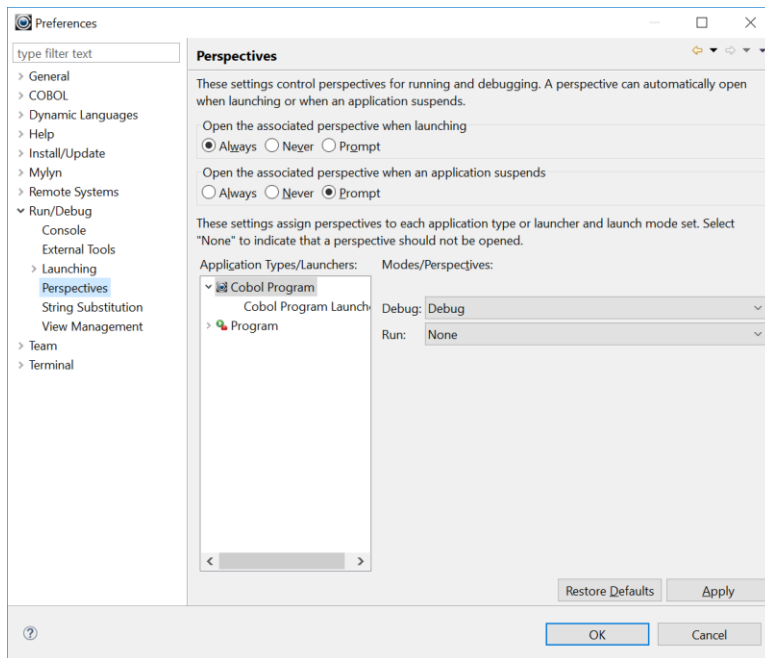
The Enterprise Edition license file is emailed to Customers with Subscriptions. Store the file in a directory of your choice. Then, use the Browse interface in Window>Preferences>COBOL to locate the license file, select it, and click on “Apply”. Click “OK” to close the Window>Preferences>COBOL dialog window.

Enter the license path of the locally-installed Developer Studio. The license file must not be modified in any way. Any modifications to the license file render it invalid.



Window>Preferences>Run/Debug>Perspectives>...

Set “Open the associated perspective when launching” to “Always”. This will cause the Debug Perspective to be opened whenever a Debug command is launched.

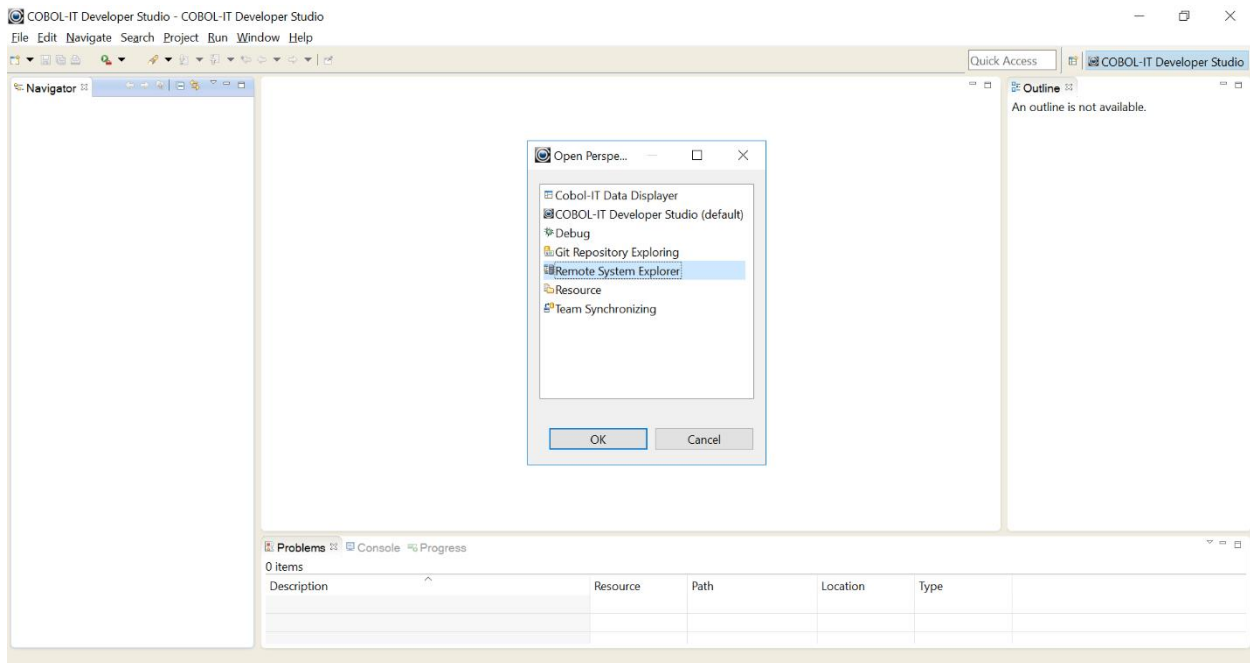


Opening the Remote System Explorer

Window>Open Perspective>Other>Remote System Explorer

Select Window>Open Perspective>Other.

In the Open Perspective dialog screen, select Remote System Explorer.

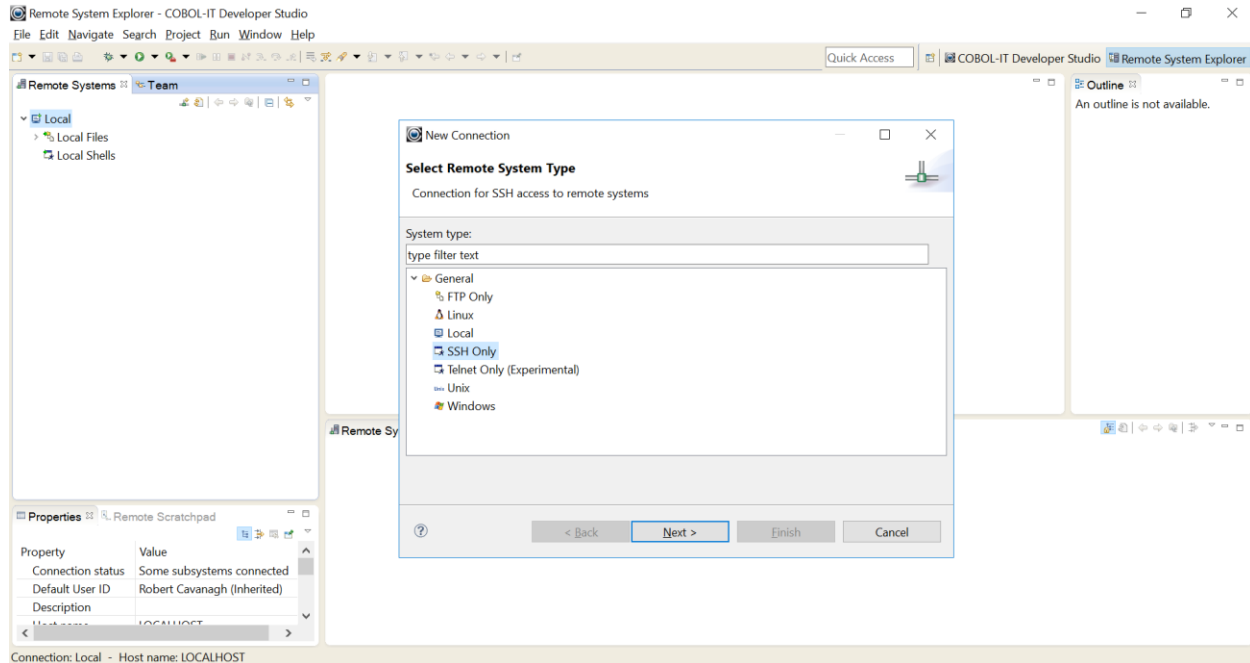


Using The Remote System View

In the Remote System View, click on **Define a connection to remote system** 

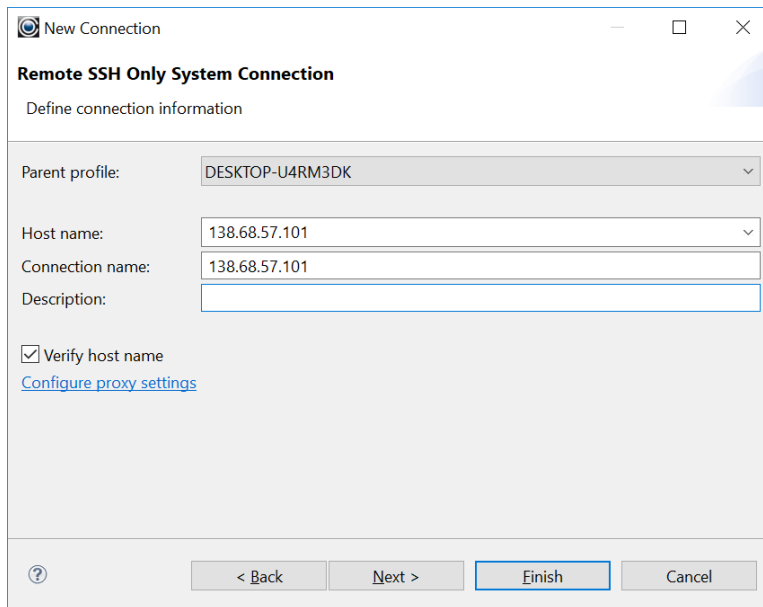
Remote System View>Define a connection to remote system

On the New Connection dialog screen, select SSH Only.
Click on the “Next” button to continue.



Enter Host Name, or IP Address

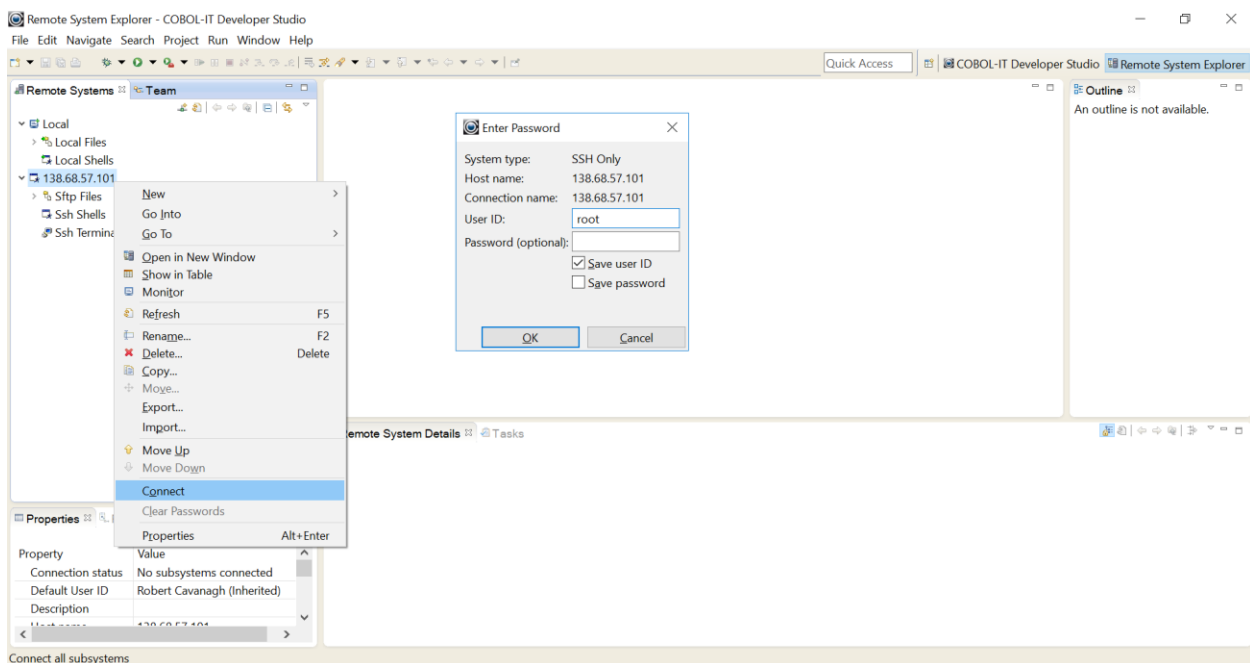
On the New Connection dialog screen, enter the host name of the remote machine.
The host name can be entered either as the name of the machine (for example sun01) or as an IP address.
Click on the Finish button to proceed to the password screen.



Remote System View>[Host Name]>[Dropdown]>Connect

The Remote System View provides you with a tree structure based on the Host Name/IP Address. Select the Host Name/IP Address. Right-click and select Connect from the dropdown menu.

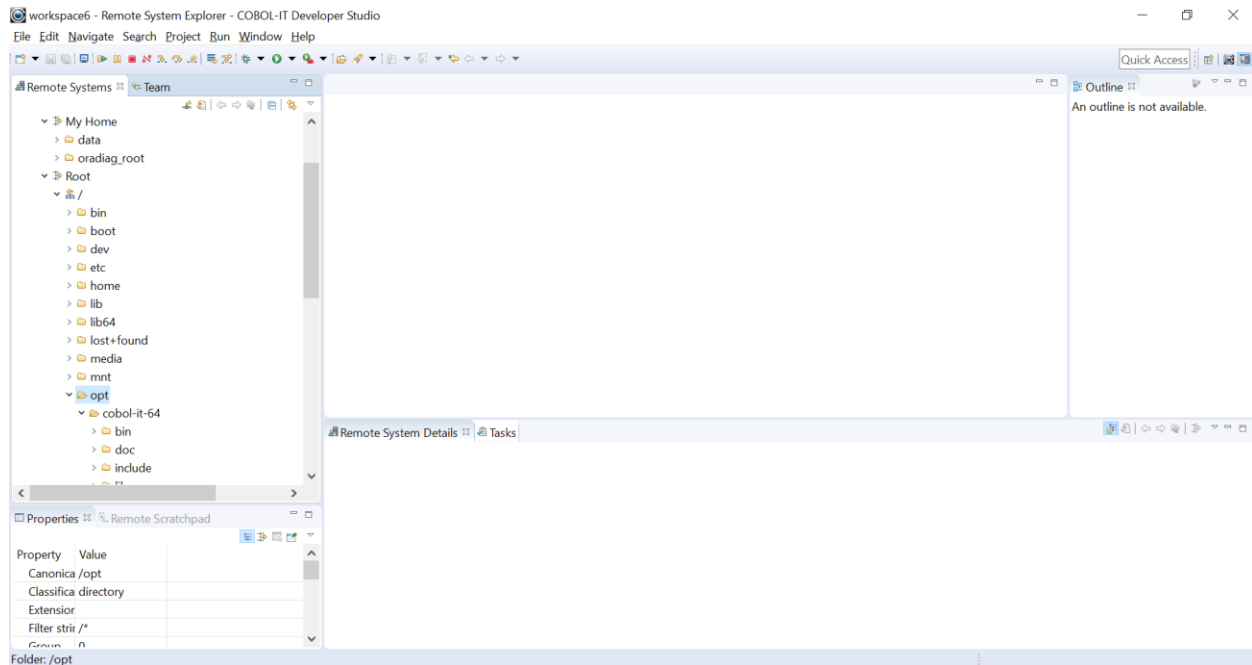
On the Enter Password Dialog Screen, enter the User ID, and Password. Note that the User ID and password can be saved, making future attachments to the remote machine automatic.



Note- This login can be pre-configured in Window>Preferences>Remote Systems>Passwords

View of the Remote Systems view when logged in

Having logged into “My Home”, you see that “My Home” now provides a view of the files and directories in the login directory on the remote system.



File>New>Project>COBOL>COBOL Project

When you have logged into the remote machine, you work in the Developer Studio and Debugger Perspective interfaces. For example, to create a new project, select File>New>Project. In this exercise, we will create a project using existing source code.

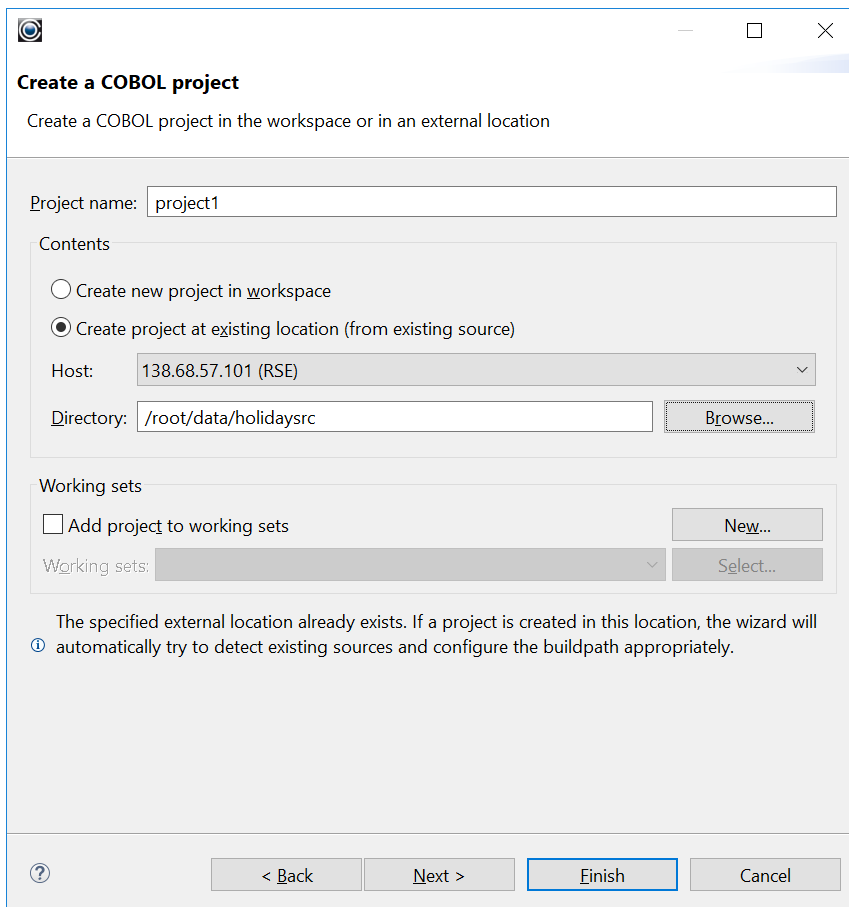
Name the project in the “Project name:” entry field.

Select “Create project at existing location (from existing source).”

Dropdown the Host combobox, and select the [Hostname/IP Address] you are working on.

Click on the Browse button, and browse to the location of the existing source code.

Click Finish.



Create a COBOL project
Create a COBOL project in the workspace or in an external location

Project name:

Contents

Create new project in workspace

Create project at existing location (from existing source)

Host:

Directory:

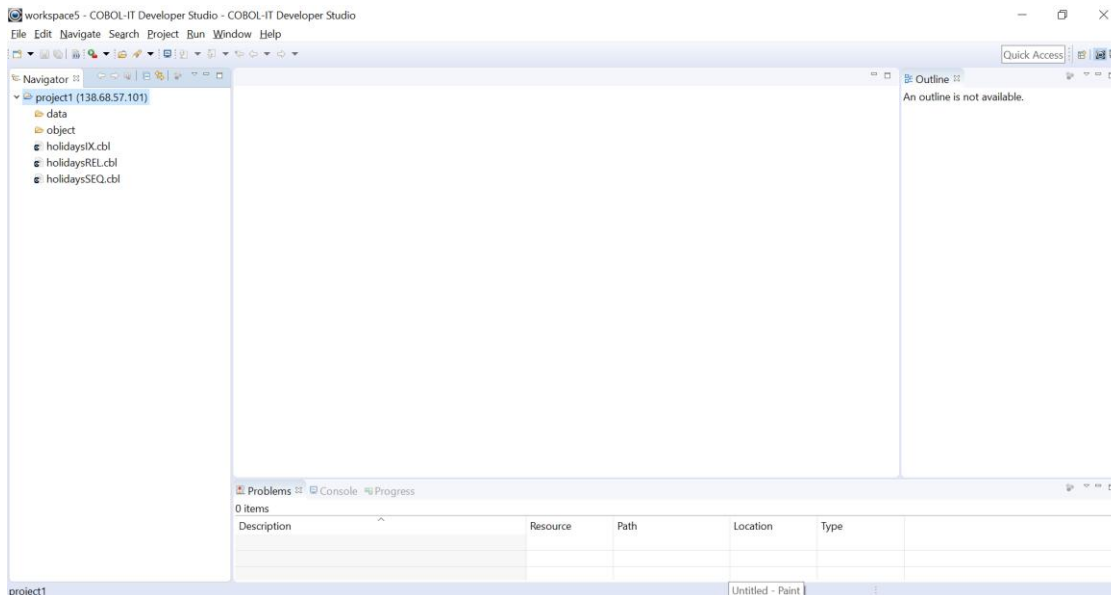
Working sets

Add project to working sets

Working sets:

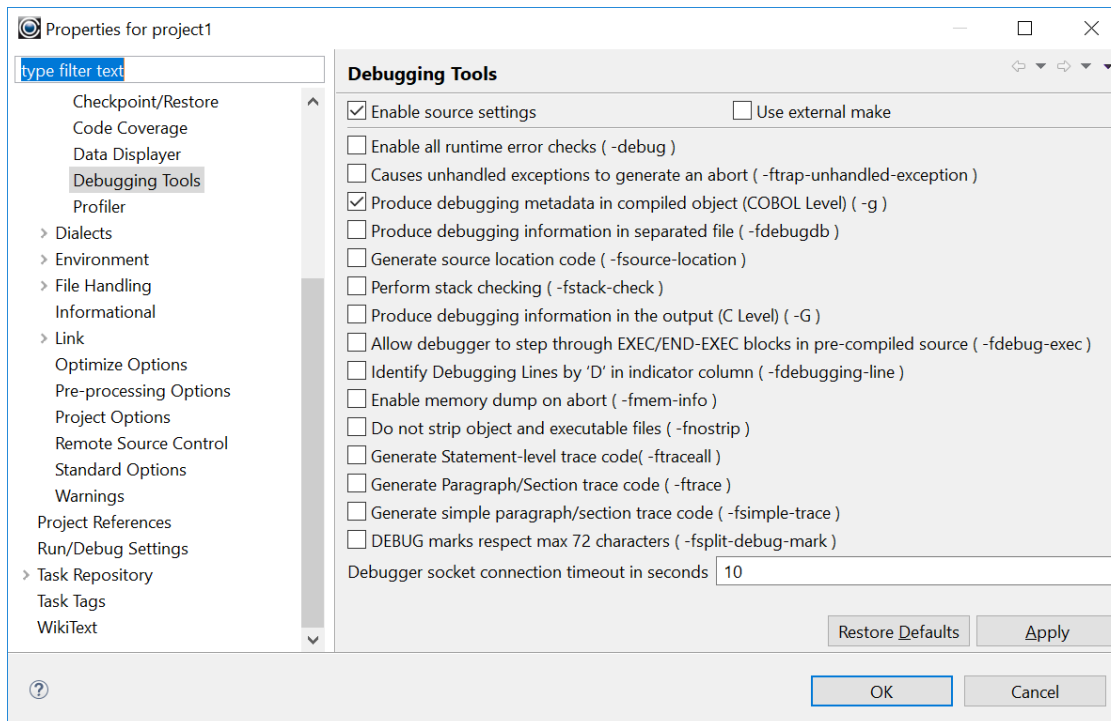
The specified external location already exists. If a project is created in this location, the wizard will automatically try to detect existing sources and configure the buildpath appropriately.

Working in a remote project



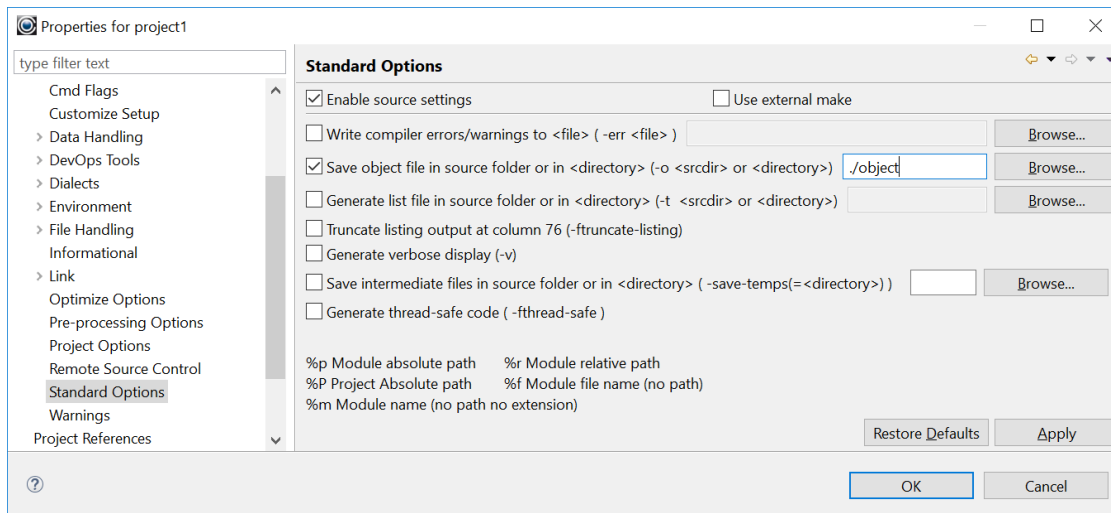
Set Project Properties

In Project>Properties>COBOL Properties>DevOps Tools>Debugging Tools, select the -g compiler flag, to allow debugging.



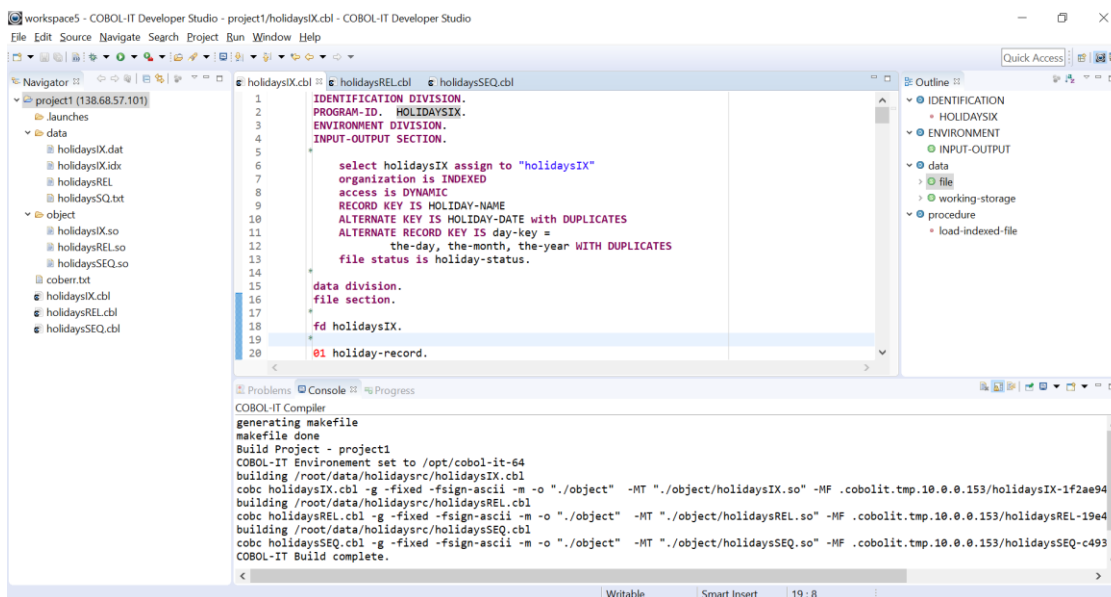
In Project>Properties>COBOL Properties>Standard Options, select the -o compiler flag option,

and enter `./object` into the corresponding entry-field.



Build the Project

Select the project, and right-click to open a drop-down menu. From the drop-down menu, select “Build Project” to build the project.



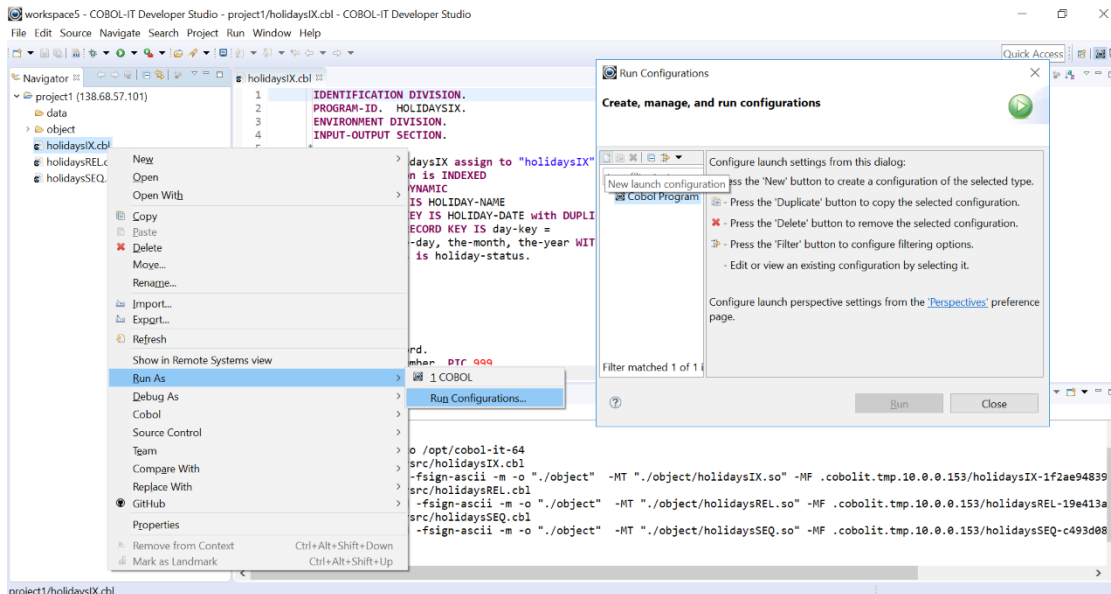
Refresh the Project

After the build is complete, right-click on the Project Name in the Navigator View, and select the Refresh function to refresh the view of the project and view the newly created files.

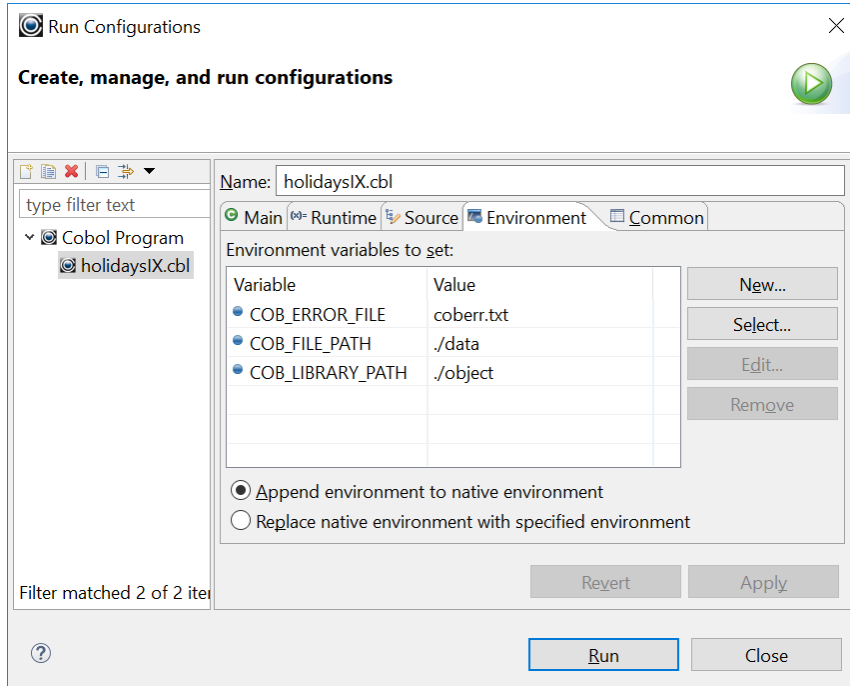
Run the Project

Select `holidaysIX.cbl`, right-click and select Run As... from the drop-down menu to open the

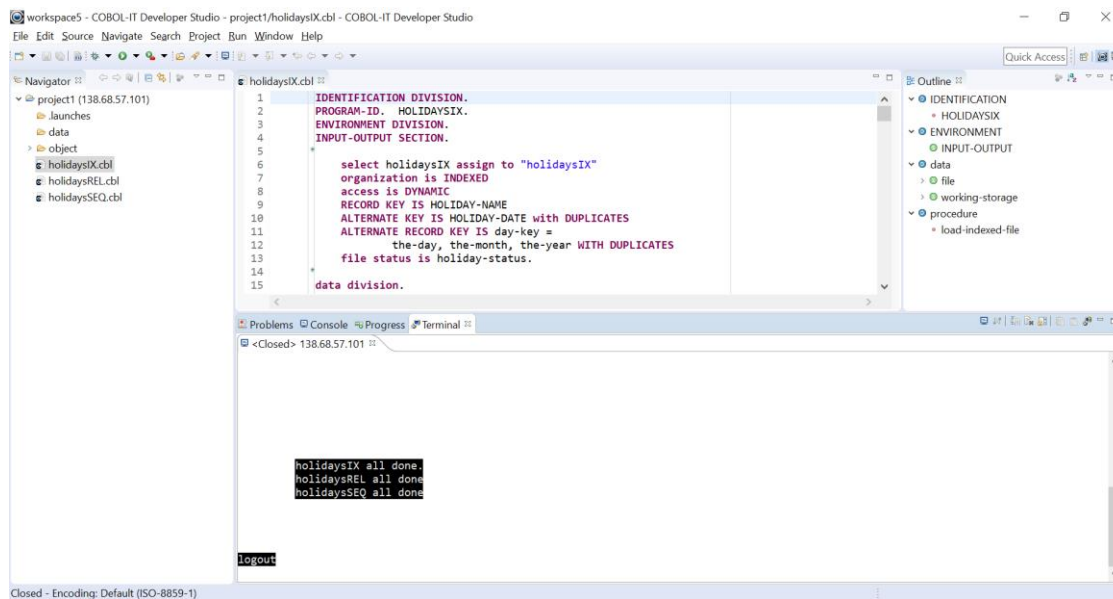
Run Configuration wizard.



We have named our configuration holidaysIX.cbl. On the Main tab, verify that that holidaysIX.cbl in project1 is selected. On the Runtime tab, verify the runtime settings. On the environment tab, set COB_FILE_PATH to ./data and COB_LIBRARY_PATH to ./object. Set COB_ERROR_FILE to coberr.txt. Click Apply, and then click on the Run button.



The program will run, creating indexed, relative and sequential files. We see our output in the Terminal window.



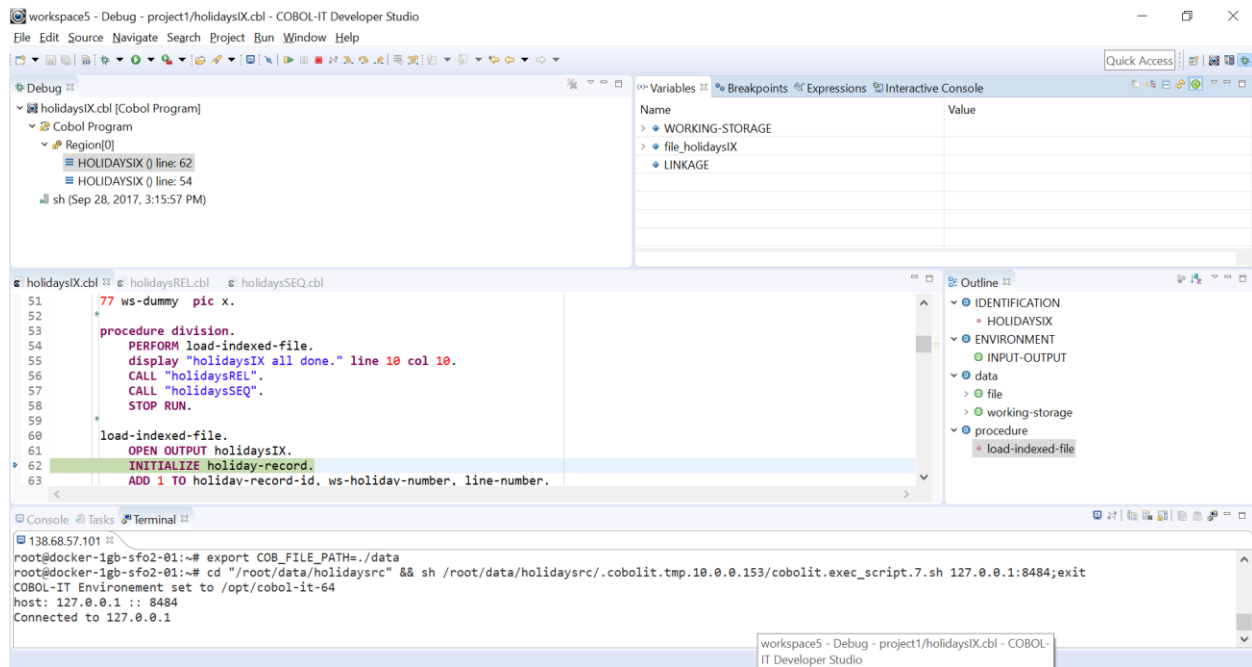
Refresh the Project

After the build is complete, right-click on the Project Name in the Navigator View, and select the Refresh function to refresh the view of the project and view the newly created files.

Remote Run in Debug

Select the program you wish to run in debug mode, and click on the Debug button. In the example below, the program holidaysIX.cbl is run in debug mode. Note that in the original configuration, the debugger was configured to launch automatically when a program was run in debug mode.

For further details on running a program in debug mode see Getting Started with the Developer Studio using the Debugger Perspective.



The screenshot displays the COBOL-IT Developer Studio interface in a remote system perspective. The main window is titled "workspace5 - Debug - project1/holidaysIX.cbl - COBOL-IT Developer Studio".

Debug View: Shows the execution state of the program "holidaysIX.cbl [Cobol Program]". The current region is "Region[0]", and the execution is at "HOLIDAYSIX () line: 62". A shell prompt "sh (Sep 28, 2017, 3:15:57 PM)" is visible.

Variables View: A table with columns "Name" and "Value".

Name	Value
WORKING-STORAGE	
file_holidaysIX	
LINKAGE	

Code Editor: Displays the COBOL source code for "holidaysIX.cbl". The code includes a procedure division with a "load-indexed-file" section. Line 62, "INITIALIZE holiday-record.", is highlighted.

```

51 77 ws-dummy pic x.
52
53
54 procedure division.
55     PERFORM load-indexed-file.
56     display "holidaysIX all done." line 10 col 10.
57     CALL "holidaysREL".
58     STOP RUN.
59
60 load-indexed-file.
61     OPEN OUTPUT holidaysIX.
62     INITIALIZE holiday-record.
63     ADD 1 TO holiday-record-id, ws-holiday-number, line-number.
    
```

Outline View: Shows a hierarchical view of the program structure, including sections like IDENTIFICATION, HOLIDAYSIX, ENVIRONMENT, INPUT-OUTPUT, data, file, working-storage, procedure, and load-indexed-file.

Console: Shows the terminal output from the remote system:

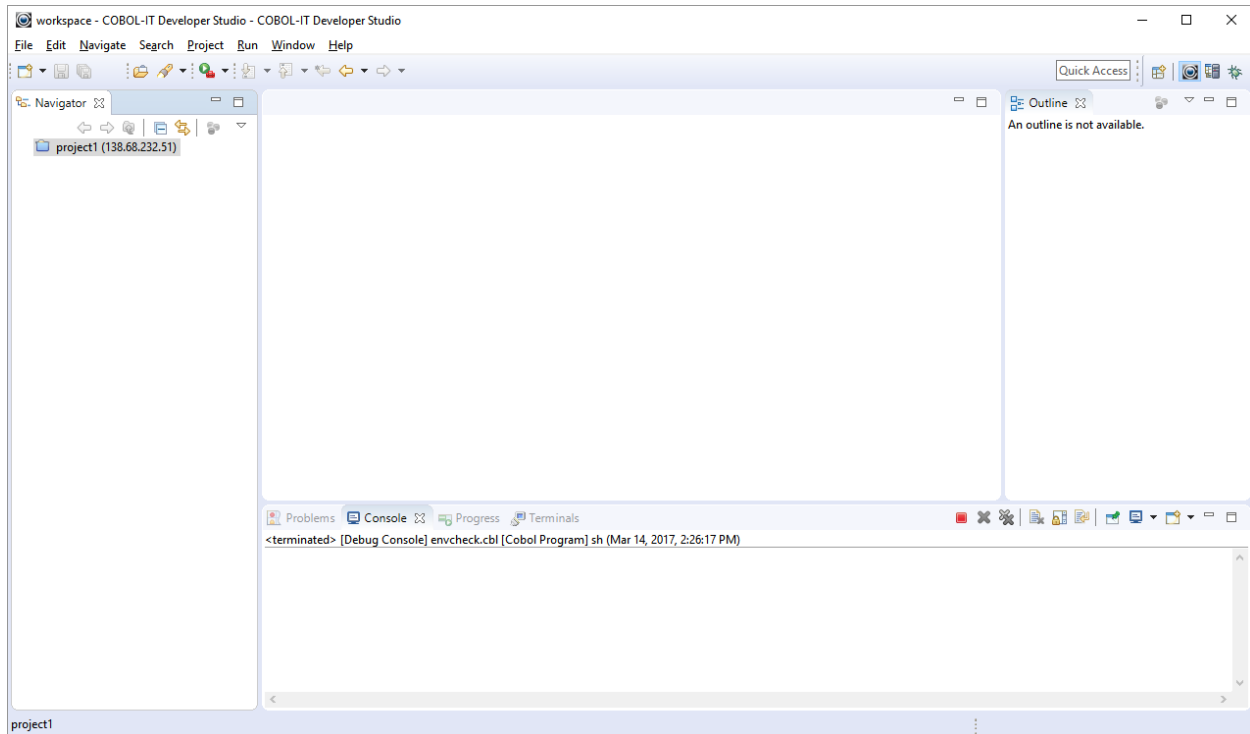
```

138.68.57.101
root@docker-1gb-sfo2-01:~# export COB_FILE_PATH=./data
root@docker-1gb-sfo2-01:~# cd "/root/data/holidaysrc" && sh /root/data/holidaysrc/.cobolit.tmp.10.0.0.153/cobolit.exec_script.7.sh 127.0.0.1:8484;exit
COBOL-IT Environment set to /opt/cobol-it-64
host: 127.0.0.1 :: 8484
Connected to 127.0.0.1
    
```

The status bar at the bottom indicates the current workspace: "workspace5 - Debug - project1/holidaysIX.cbl - COBOL-IT Developer Studio".

Re-opening a Remote Project

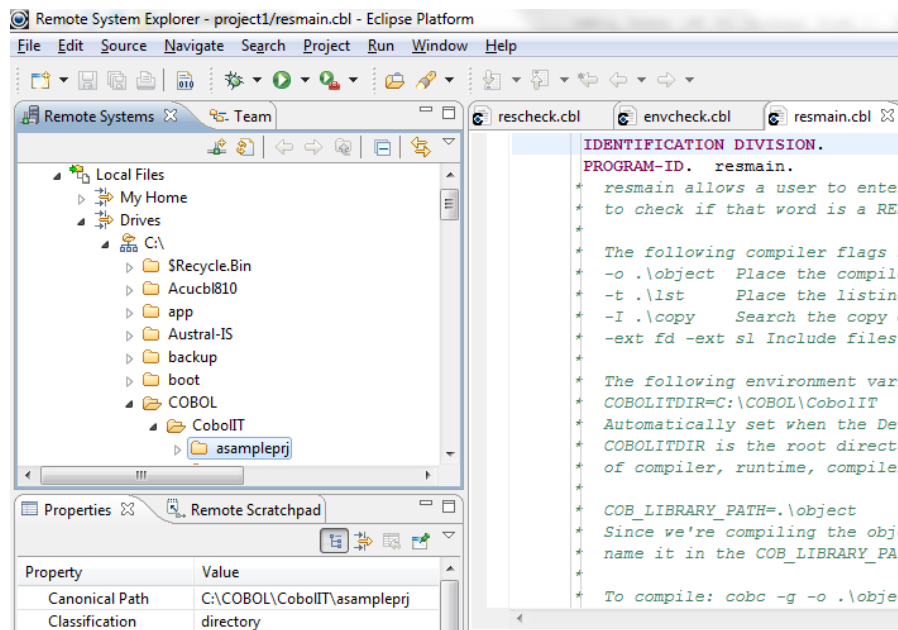
When you re-open a previously created remote project, Eclipse shows the remote project in the Navigator View. To log in to the remote machine, and see the expanded view of the project, double-click on the remote project name.



FAQ

Copy and Paste files from a local machine to a remote machine

In Local Files, select a directory (asampleprj in the image below). Right click on the selected directory, and select “Copy”, or press “Ctrl+C”.



Then, scroll down to the view of the remote machine, and select “My Home”. Right click on the selected directory, and select “Paste”, or press “Ctrl+V”. A pop-up window advises you of the status of the file transfers taking place.



www.cobol-it.com

June, 2020

