

# FICO® Xpress Workbench

3.3

USER GUIDE

**FICO**® **Decisions**

© 2016–2020 Fair Isaac Corporation. All rights reserved. This documentation is the property of Fair Isaac Corporation ("FICO"). Receipt or possession of this documentation does not convey rights to disclose, reproduce, make derivative works, use, or allow others to use it except solely for internal evaluation purposes to determine whether to purchase a license to the software described in this documentation, or as otherwise set forth in a written software license agreement between you and FICO (or a FICO affiliate). Use of this documentation and the software described in it must conform strictly to the foregoing permitted uses, and no other use is permitted.

The information in this documentation is subject to change without notice. If you find any problems in this documentation, please report them to us in writing. Neither FICO nor its affiliates warrant that this documentation is error-free, nor are there any other warranties with respect to the documentation except as may be provided in the license agreement. FICO and its affiliates specifically disclaim any warranties, express or implied, including, but not limited to, non-infringement, merchantability and fitness for a particular purpose. Portions of this documentation and the software described in it may contain copyright of various authors and may be licensed under certain third-party licenses identified in the software, documentation, or both.

In no event shall FICO or its affiliates be liable to any person for direct, indirect, special, incidental, or consequential damages, including lost profits, arising out of the use of this documentation or the software described in it, even if FICO or its affiliates have been advised of the possibility of such damage. FICO and its affiliates have no obligation to provide maintenance, support, updates, enhancements, or modifications except as required to licensed users under a license agreement.

FICO is a registered trademark of Fair Isaac Corporation in the United States and may be a registered trademark of Fair Isaac Corporation in other countries. Other product and company names herein may be trademarks of their respective owners.

FICO® Xpress Workbench 3.3

Deliverable Version: A

Last Revised: March 27, 2020

# Contents

---

## CHAPTER 1:

### Introducing FICO® Xpress Workbench ..... 7

What is the FICO® Analytic Cloud (FAC) .....	8
Decision Management Platform (DMP) .....	8
Getting Started .....	8
Importing Xpress Insight apps to Xpress Workbench on the FAC Edition .....	9
Converting Xpress Mosel Models on the Desktop Edition .....	9
Open an Existing Xpress Workbench Project .....	10

## CHAPTER 2:

### First Steps ..... 11

Logging in and out of the FAC Edition of Xpress Workbench .....	11
Logging In .....	11
Logging Out .....	12
Installing and Launching the Desktop Edition of Xpress Workbench .....	12
Supported versions .....	13
Creating a Project .....	13
Create a Project with the FAC Edition .....	13
Create a Project with the Desktop Edition .....	13
Opening and Closing Projects with the Desktop Edition .....	14
The Xpress Workbench IDE .....	14
Editing Mosel Files .....	16
Editing Insight App Source Files .....	16
Using the View Designer and VDL to Create Custom Views .....	16
View Designer Tutorial .....	17
View Designer Overview .....	18
Creating a New View .....	20
Publishing to Xpress Insight .....	22
Debugging Xpress Insight Scenarios .....	23
Setting Breakpoints .....	23
Debugging the Scenario .....	23
Finishing the Debug Session .....	24

## **CHAPTER 3:**

### **Working with Projects ..... 25**

Creating Projects .....	25
Creating Projects with the FAC Edition .....	25
Create a Project with the Desktop Edition .....	26
Opening and Closing Projects with the Desktop Edition .....	27
Renaming Projects - FAC Edition Only .....	27
Deleting Projects - FAC Edition Only .....	27
Opening Projects .....	27
Backing up and Restoring Projects - FAC Edition Only .....	28
Displaying the Dashboard - FAC Edition Only .....	28

## **CHAPTER 4:**

### **Editing Source Files ..... 29**

Opening Files .....	29
Opening Files with the FAC Edition .....	29
Opening Files with the Desktop Edition .....	29
Creating Files .....	29
Creating Files from the File Tree .....	30
Creating Files from the File Menu .....	30
Creating Files from a Template .....	30
Creating Files from the View Designer .....	30
Renaming Files .....	31
Uploading Files - FAC Edition Only .....	31
Downloading Files - FAC Edition Only .....	32
Downloading the Project .....	32
Viewing File Revision History - FAC Edition Only .....	32
Coding Assistance .....	33
Finding Text .....	33
Replacing Text .....	34
Finding Text Across Many Files .....	35
Replacing Text Across Many Files .....	35
Highlighting Syntax .....	36
Automatically Completing Words .....	36
Automatically Completing Statements .....	36
Inline Documentation .....	37
Folding Code .....	38
Folding Code with the Mouse .....	38
Folding Code with the Keyboard .....	38
Collaborating with Other Users - FAC Edition Only .....	38
Sending Messages to Users .....	38

## CHAPTER 5: Running and Debugging Mosel Models ..... 41

Running Models	41
Stopping a Model	42
Changing the Working Directory	43
Debugging Models	43
Starting a Debug Session	43
Setting and Removing Breakpoints	44
Suspending and Resuming the Model	45
Stepping through the Model	45
Examining Variables	46
Examining Arrays	46
Saving Array Data as CSV Files	47
Evaluating Expressions	47
Using the Immediate Window	48
Compiling Models	48
Configuring Compiler Options	49
Compiler Options	49
Running Submodels and Parallel Solving	50
Preparing Template Files for Mosel Deployment Wrappers	51
Tuning Mosel Models	51
Starting a Tuning Session	52
Tuner Settings	53
Manually Enabling the Tuner	54
Profiling Mosel Models	55
Starting a Profiling Session	55
Collecting Coverage Data	55
Tracing Model Execution	56

## CHAPTER 6: Working with Xpress Insight ..... 57

Configuring an Xpress Insight Server - Desktop Edition Only	57
Managing Insight Views	57
Managing Views	59
Importing Views	59
Deleting Views	60
Managing View Groups	60
Deleting View Groups	61
Changing the View order	61
Publishing to Xpress Insight	61
Updating Xpress Insight Apps	62

Opening Xpress Insight .....	63
Exporting Xpress Insight Apps .....	63
Debugging Scenarios .....	64
Manually Executing Scenarios .....	64
Cancelling Debug Sessions .....	65
Unlinking Projects from Xpress Insight .....	65

## **CHAPTER 7:**

### **Interacting with Tableau Server ..... 67**

Creating Workbooks .....	67
Editing Workbooks .....	68
Downloading Workbooks into Xpress Workbench .....	69
Viewing the Workbook Status .....	69

## **APPENDIX B:**

### **Contacting FICO ..... 71**

Product Support .....	71
Product Education .....	71
Product Documentation .....	71
Sales and Maintenance .....	72
Related Services .....	72
FICO Community .....	72
About FICO .....	72

## **INDEX.....73**

## CHAPTER 1

# Introducing FICO® Xpress Workbench

---

FICO® Xpress Workbench is a tool for creating, editing, and debugging FICO® Xpress Insight apps and standalone Xpress Mosel models.

Xpress Workbench enables you to:

- Create and edit source files
- Publish Xpress Insight apps to an Xpress Insight server
- Debug an Xpress Insight scenario while it is executing within an Xpress Insight server
- Build an Xpress Insight app archive
- Debug a Mosel model in a standalone instance of Mosel
- Compile a Mosel model into a BIM file for use in FICO® Xpress Executor or another application powered by Mosel

Xpress Workbench is available on the FICO® Analytic Cloud (FAC) and in a desktop version. Some help topics are specific to one version or the other, and the titles of the topics will include *FAC Edition* or *Desktop Edition*. Additional documentation resources that you are likely to need during the course of development are available and listed below.

- [Xpress Insight Mosel Developer Guide](#)  
The guide to Mosel app development with Xpress Insight
- [Xpress Insight Python Developer Guide](#)  
The guide to Python app development with Xpress Insight
- [Xpress Insight Mosel Reference](#)  
The reference for the Mosel interface to Xpress Insight
- [Xpress Insight Python Reference](#)  
The reference for the *xpressinsight* Python package for Xpress Insight
- [Mosel Language Reference Manual](#)  
The main reference for Xpress-Mosel, including details of the Mosel language and all modules supplied with the standard distribution
- [Mosel IO Whitepaper](#)  
A guide to file handling in Mosel

- [Optimizer Reference Manual](#)  
The main reference for FICO® Xpress
- [xpress\\_docs.zip](#)  
The complete documentation for the Xpress Suite

## What is the FICO® Analytic Cloud (FAC)

The FICO® Analytic Cloud (FAC) delivers a comprehensive toolkit of cloud-enabled products, across analytics, decision management and optimization through the FICO® Decision Management Platform (DMP), an integrated platform solution in which you can build, deploy, execute, and scale analytic-powered applications.

The Platform serves as the primary cloud infrastructure for building, hosting, and managing components and apps such as Decision Optimizer. For more visit [www.ficoanalyticcloud.com](http://www.ficoanalyticcloud.com).

## Decision Management Platform (DMP)

The FICO® Decision Management Platform (DMP) is a high-performance execution platform that provides a complete environment for developing, distributing, and using decision management solutions.

It can operationalize and connect analytically-powered applications and decisions to provide centralized control to author, develop, and deploy applications faster than any other solution. For more visit <https://www.fico.com/br/platform/fico-decision-management-platform>.

## Getting Started

If you wish to convert existing Xpress Mosel models to work with corresponding FICO® Xpress Insight apps, or would like to import Xpress Insight apps into FICO® Xpress Workbench, this section contains brief references to resources that will help you get started quickly.

### *Converting Xpress Mosel Models to Work with Xpress Insight apps*

Refer to Chapter 2 and subsequent sections of the [Xpress Insight Developers Guide](#) for full information on how to convert an Xpress Mosel model so that it works with an Xpress Insight app.



## *Importing Xpress Insight apps to Xpress Workbench on the FAC Edition*

It is a straightforward process to import an Xpress Insight app into Xpress Workbench.

### *Importing Xpress Insight apps to Xpress Workbench*

To import an Xpress Insight app into Xpress Workbench to continue its development:

- 1 Log in to Xpress Workbench and create a new project. See [Logging In](#) and [Create a Project with the FAC Edition](#) for guidance if needed.
- 2 Load your new project by clicking on its tile on the **Home** page.
- 3 A new project contains some default files and folders which will not be needed. Using the right mouse-button menu, delete:
  - client\_resources (folder)
  - model\_resources (folder)
  - source (folder)
  - application.xml
- 4 Using your operating system's file explorer, navigate to the root folder of your Xpress Insight app and upload all relevant files and folders into your new, now empty project. Refer to [Uploading Files - FAC Edition Only](#) for further details.

## *Converting Xpress Mosel Models on the Desktop Edition*

If you wish to convert existing Xpress Mosel models and build them into FICO® Xpress Insight apps, or would like to open an existing Xpress Workbench project within the desktop version, this section contains brief references to resources that will help you get started quickly.

### *Converting Xpress Mosel Models to work with Xpress Insight apps*

Refer to Chapter 2 and subsequent sections of the [Xpress Insight Developers Guide](#) for full information on how to convert an Xpress Mosel model so that it works with a Xpress Insight app.

## *Open an Existing Xpress Workbench Project*

To open a project within the desktop version of Xpress Workbench:

- 1 Launch the desktop version of Xpress Workbench.
- 2 Select **Open existing file or folder**.
- 3 Select **Open Project**.
- 4 Using your operating system's file explorer, navigate to the location of your project's root folder and follow the prompts.

## CHAPTER 2

# First Steps

---

This section contains a tutorial that shows you how to use Xpress Workbench to create an Xpress Insight app and publish it to an Xpress Insight server. The following topics are covered:

- Logging in and out
- Creating projects
- Editing source files
- Publishing to Xpress Insight
- Debugging Xpress Insight scenarios

## Logging in and out of the FAC Edition of Xpress Workbench

Before you can use Xpress Workbench you must log in.

This topic provides information on obtaining the URL and logging in.

### *Logging In*

To log in, you must know the following:

- Your login URL - contact your administrator to find out the URL.
- Your FICO Analytic Cloud username and password.

To log into Xpress Workbench, perform the following steps:

- 1 Enter the Xpress Workbench URL into the address bar of your browser.
- 2 Enter your username and password into the **Log in** page.
- 3 Click **Log In**.

The Xpress Workbench Dashboard page is displayed. From here you can open a project or create a new one.

## Logging Out

Log out of Xpress Workbench before closing the browser or navigating to a different URL.

- 1 Click the arrow next to your username in the page header.
- 2 Click **Exit** in the drop-down list.

## Installing and Launching the Desktop Edition of Xpress Workbench

The desktop version of Xpress Workbench is installed alongside Xpress.

If you intend to use Xpress Insight in conjunction with Xpress and Xpress Workbench, ensure that you install Xpress Insight as the final install after Xpress and its automatic Xpress Workbench install.

To launch Xpress Workbench:

- In Windows, open the **Start** menu and click the **Xpress Workbench** icon in the **FICO** folder.
- On a Mac, open the **FICO Xpress** folder in **Applications** and click the **Xpress Workbench** icon.

You will be presented with an initial welcome page offering you the following options:

- **Recent projects** enables you to choose a recent project to open.
- **Create project**
- **Open existing file or folder** enables you to open an existing Workbench project, or open a source file from a folder on your computer.
- **Examples** contains a range of Mosel example files, such as:
  - Introductory examples
  - Insight examples
  - Nonlinear examples
  - Calling R from Mosel
  - Calling Python from Mosel
  - Browse all examples
- **Xpress Documentation**-link to the Mosel documentation on the FICO Community
- **Ask the Xpress Community**-link to the Optimization forum in the FICO Community

## Supported versions

Xpress Workbench supports the following platform, operating system, and processor combinations.

FICO Xpress	Platform	Operating System	Processor
Xpress Optimizer	Windows 64-bit	8, 10	Any AMD64 or Intel EM64T enabled 64-bit CPU
Xpress Mosel		Windows Server 2012 R1 and R2, Windows Server 2016 R1 and R2, Windows Server 2019	
■ IVE editor is available for Mosel on Windows only.			
■ Mosel connectivity to spreadsheets is available on Windows, Linux, and macOS.	Linux 64-bit	RHEL 6, 7	Any AMD64 or Intel EM64T enabled 64-bit CPU
Xpress NonLinear	macOS 64-bit	macOS 10.13 (High Sierra) to macOS 10.15 (Catalina)	Intel EM64T enabled 64-bit CPU
	Solaris 64-bit	Solaris 10 - 11	Any AMD64 or Intel EM64T enabled 64-bit CPU

## Creating a Project

A project contains all the source files that make up an optimization app.

In this section you will create a new project and open it.

### Create a Project with the FAC Edition

- 1 From the Dashboard page, click the **Create Project** button.
- 2 Enter a name for the project and click **Save**.
- 3 To open the project, click the project's name.  
The project opens in the **Project** page.

### Create a Project with the Desktop Edition


A project contains all the source files that make up an Xpress Insight app or Mosel application.

Create a new Mosel project, Xpress Insight Mosel project, or Xpress Insight Python project.

- 1 To create a new project:
  - To create a simple Mosel project, choose the **Create Project > CREATE MOSEL PROJECT** menu option.

- To create a new Xpress Insight project, you must select to work in either Python or Mosel.
  - To develop using Mosel, choose the **Create Project > CREATE INSIGHT (MOSEL) PROJECT** menu option.
  - To develop using Python, choose the **Create Project > CREATE INSIGHT (PYTHON) PROJECT** menu option.
- 2 Navigate your filesystem to a folder where you wish to store your project and click **OK**.

The project opens—depending on the kind of project you created, it will contain a minimal Xpress Insight app for Python or Mosel, or a minimal Mosel application. You can use a Mosel project to begin the development of a standalone Mosel model. The remainder of this guide focuses, however, on activities relating to Xpress Insight Mosel projects and apps.

 **Note** The [Xpress Insight Python Developer Guide](#) offers information on using Python with Xpress Workbench.

## Opening and Closing Projects with the Desktop Edition

After a project has been created, you can in subsequent Xpress Workbench sessions, open and close it.

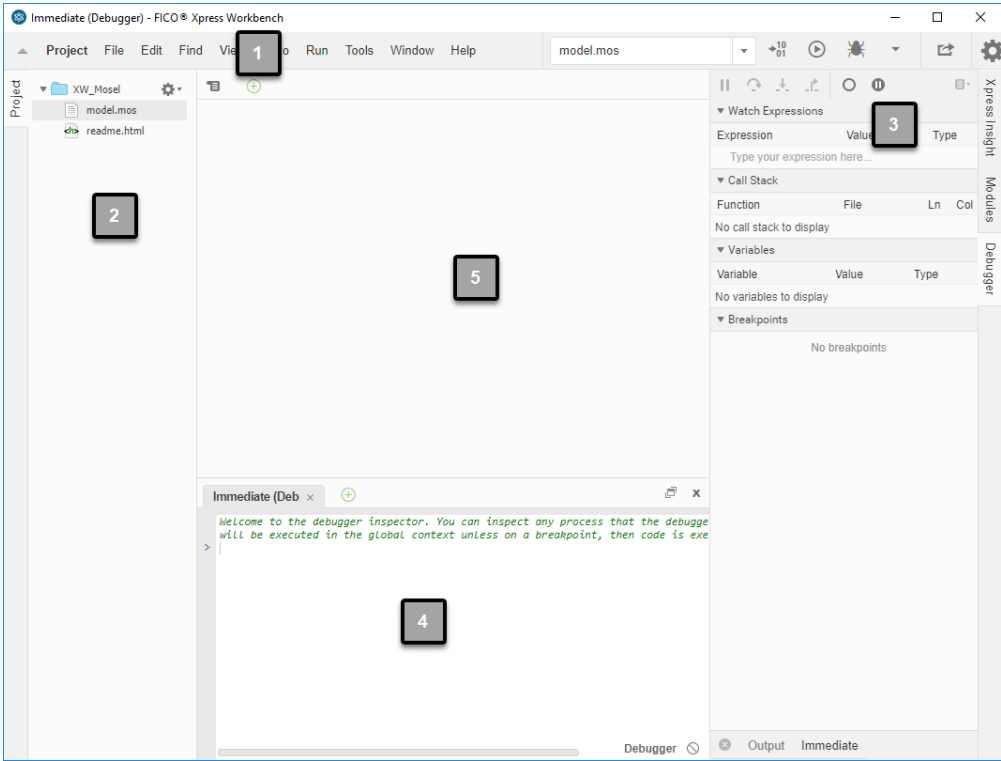
To open an existing project select **Project** from the top-level menu, choose **Open Project**, follow the prompts and use the filesystem navigator to locate the folder in which your existing project is located.

To close an open project, select **Project** from the top-level menu and choose **Close Project**.

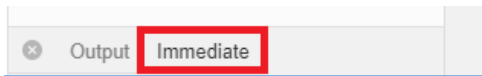
## The Xpress Workbench IDE

The Xpress Workbench Integrated Development Environment contains five major panels.

The Xpress Workbench IDE helps you edit every type of source file that makes up your optimization app, along with many other file types. It also offers an intuitive representative method for editing views (vdl files) in a drag and drop interface.



Item	Description
1	<p>Top Menu</p> <ul style="list-style-type: none"><li>■ On the left is displayed the major command menus that you will use to create and edit your Mosel file.</li><li>■ On the right are the buttons which compile, run, and debug the current file.</li><li>■ There are also buttons that enable editing of the Compiler Options and Project Settings (Preferences).</li></ul>
2	<p>Left sidebar—Tabbed panels allow you to:</p> <ul style="list-style-type: none"><li>■ see a hierarchical view of the current <b>Project</b>, to view and open any file in the current project.</li><li>■ View a list of Xpress Workbench <b>Commands</b>.</li><li>■ When a .vdl file is selected, the <b>Palette</b> displays the elements that can be used to create Views for Xpress Insight.</li></ul> <p>✓ <b>Note</b> The user interface only displays tabs appropriate to the type of file that is open and active in Workbench.</p>
3	<p>Right sidebar—Pop out menus enable you to:</p> <ul style="list-style-type: none"><li>■ Connect to <b>Xpress Insight</b>, to build and publish Insight apps.</li><li>■ View a complete list of <b>Modules</b> available to your Mosel script.</li><li>■ Open the <b>Debugger</b> menu to evaluate expressions.</li><li>■ At the bottom of the pane, the <b>Immediate</b> button opens a tabbed pane which lets you evaluate expressions in an environment similar to a command prompt.</li></ul>

Item	Description
	
4	Bottom tabbed pane <ul style="list-style-type: none"> <li>■ The results from each executed Mosel file opens a tab in this area.</li> <li>■ The <b>Immediate</b> button opens a tab in this area.</li> </ul>
5	File editing pane

## Editing Mosel Files

When you use Xpress Workbench to create a new Mosel project, a correctly structured Mosel model file named `model.mos` is automatically generated in the project folder.

This model file provides the basis for your model creation.

### *Workbench code editing features*

Xpress Workbench supports code folding, auto-completion, and syntax highlighting in the Mosel language. For more see [Coding Assistance](#) on page 33.

## Editing Insight App Source Files

Every new Xpress Insight project contains the files for a minimal Xpress Insight app.

In this section you will edit some of these files to add functionality to the app.

## *Using the View Designer and VDL to Create Custom Views*

The View Designer enables rapid development of custom views by allowing you to associate user interface components with model data for viewing and editing.

Views are created using a Markup Language which is a super-set of HTML and follows the same syntax, along with additional elements defined by the VDL language. The View Designer creates this VDL code, which is extensible markup language that allows you to create reusable components. VDL contains extensive functionality for defining interactive views through configuration only. Content can be augmented with customization via HTML, CSS and JavaScript. This chapter explores the underlying VDL code that makes up the View in greater detail.

You can quickly switch between the code editor and the View Designer using the links displayed in the top right corner of the artboard.





Figure 1: View Designer link

This section provides examples in both the View Designer and code editor when the feature being described is accessible by both methods.

The View Designer integrates with Xpress Workbench entities and scenarios, generating the required VDL code to provide simple, direct bindings to your app's model data, to aid rapid view development.

## VDL Examples App

Your Xpress Workbench installation contains an app that showcases all the elements within VDL, it is located at `<installdir>/examples/insight/vdl/vdl_language.zip`. You can upload it to Xpress Workbench and click on its tile to launch it within the Xpress Workbench environment.

## Open Source Extensions

In addition to the developed VDL components, the View Designer also supports the integration of additional open source components. FICO has collected several examples and documented the installation process for Xpress Workbench, to enable them to be used in Xpress Insight views. The available examples include:

- an alternative to `<vdl-table>` in a FICO Xpress Insight custom view, using `vdlx-datagrid` that offers enhanced performance whilst also enabling customizable Index column positioning; customizable sorting defaults; support for calculated columns, and use of range operators for column filters. This can be downloaded from <https://github.com/fico-xpress/vdlx-datagrid>.
- user interface elements including `vdlx-sidebar`, the main sidebar extension; `vdlx-toolbar`, a simple toolbar at the top of view; `vdlx-dropdown-button`, for a drop-down list; `vdlx-tabs`, to create tabs or pills for inner view navigation; and `vdlx-modal`, a modal dialog wrapper that can contain custom VDL.

These can be downloaded as a single zip archive from <https://github.com/fico-xpress/vdl-custom-extensions>.

## Javascript API

FICO also offers a JavaScript API, for more see [Using the JavaScript API to Create Custom Views](#).

## View Designer Tutorial

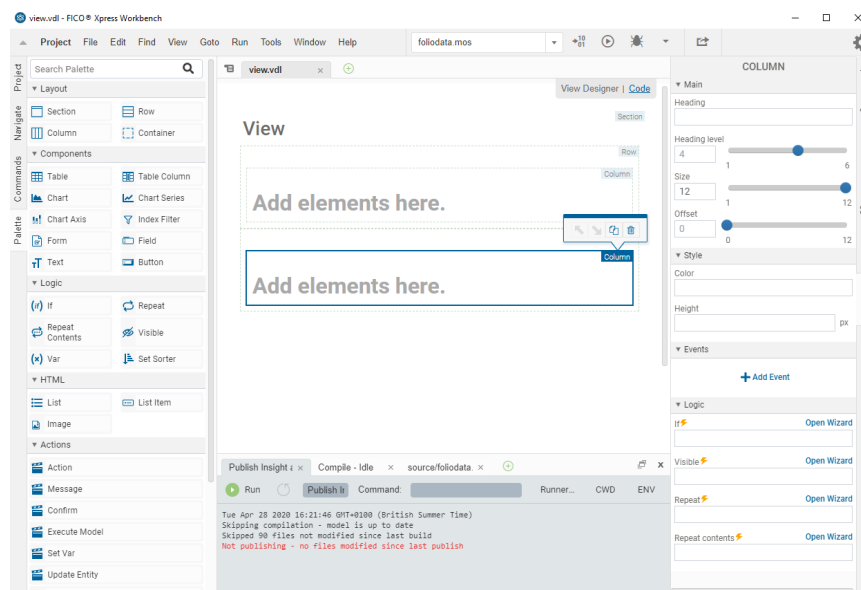
Creating Custom Views using the View Designer is fully explored in the Xpress Insight Developers Guide.

For more on Using VDL to Create Custom Views, see Chapter 5 of the Xpress Insight Developers Guide.

## View Designer Overview

The View Designer is the default editor for a file with a .vdl extension when Xpress Workbench is initially installed—This setting can be changed in the project preferences.

In the View Designer, a **Palette** pane is displayed on the left, the artboard shown in the central area, and an **Attributes** pane on the right.



Views for Xpress Insight apps are composed of various elements: Layouts (such as Section, Row, and Column), Components (such as Tables and Charts), and Logical elements (such as Repeat and Visible). Drag elements from the **Palette** to the artboard to add them to the view.



**Note** Views are regularly automatically saved while they are open in the View Designer.

Hovering over the elements on the artboard will highlight their boundaries. Elements can be selected by a left click.

The most commonly used attributes of the currently selected element are displayed in the **Attributes** pane; The selected element type is shown at the top of the **Attributes** pane.

Elements placed on the artboard can be moved relative to each other, duplicated, or deleted by utilizing the [action toolbar](#) that displays when a placed element is selected. Elements can also be cut, copied, and pasted.

## Using the Palette

Elements are organized on the **Palette** in groups. Click and drag the element you wish to use onto the artboard. As you pass over placed layout elements, they will change color to indicate if the dragged element can be dropped there.

- blue shading indicates a valid position.
- red shading indicates an invalid position and the element will not be accepted at this position.

The five panes in the Palette and the elements they contain are:

LAYOUT	Section	Block of page content with padding top and bottom
	Row	Each row is placed under existing content and can hold up to 12 columns
	Column	Divides a row into vertical blocks with a configurable width
	Container	Defines an area in the view and is used to hold other elements
COMPONENTS	Table	Placeholder for a table of array entities that are defined in its Table Columns
	Table Column	Defines the array entities that provide the content for a Table
	Chart	Placeholder for a chart that displays model entity data or custom data
	Chart Series	Set of entity or data information supplied to the Chart element
	Index-Filter	Filter an index out of a VDL chart or table
	Form	A container for field elements that controls their behavior
	Field	Used to enter text or values in a Form
	Button	Can be placed anywhere in the view to set a value and run the scenario
	Text	Text in the view can be static or updated using expressions
LOGIC	If	Use an expression to control whether an element is displayed
	Repeat	Loop around each element in an array, set, element in a JS array, or active scenario.
	Repeat Contents	Used to repeat a block of content
	Visible	Use an expression to control the visibility of an element
	Var	Defines a variable name and its value
	Set Sorter	Change the way a set is sorted from the default settings.

HTML	List	A container for List Item elements that populate a bulleted list
	List Item	A container for an item in a list, each list item must be populated with content such as a text component or an html image container
	Image	Defines the source and size of an image in the view
ACTIONS	Action	The base action used to call other Actions or Javascript functions
	Message	Display a message
	Confirm	Show a confirmation dialog
	Execute Model	Queue the model for execution
	Set Var	Set a dynamic variable
	Update Entity	Set the value of a model entity
	Remove Entity	Delete the value of a model entity
	Create Attachment	Create a new attachment
	Edit Attachment	Edit a existing attachment
	Manage Attachment	Manage an attachment
	Edit Attachment Properties	Open a dialog to change the properties of an attachment
	Download Attachment	Download an attachment
	Upload Attachment	Upload an attachment

## Creating a New View

Perform the following steps to begin creating the user interface.

To create a new View using **Xpress Workbench 3.1** or higher:

- 1 Click **Open Project**.
- 2 Either:
  - Click **File > New > Insight View (VDL)**.
  - Click the **New View** button on the tab bar.



Figure 2: New View button

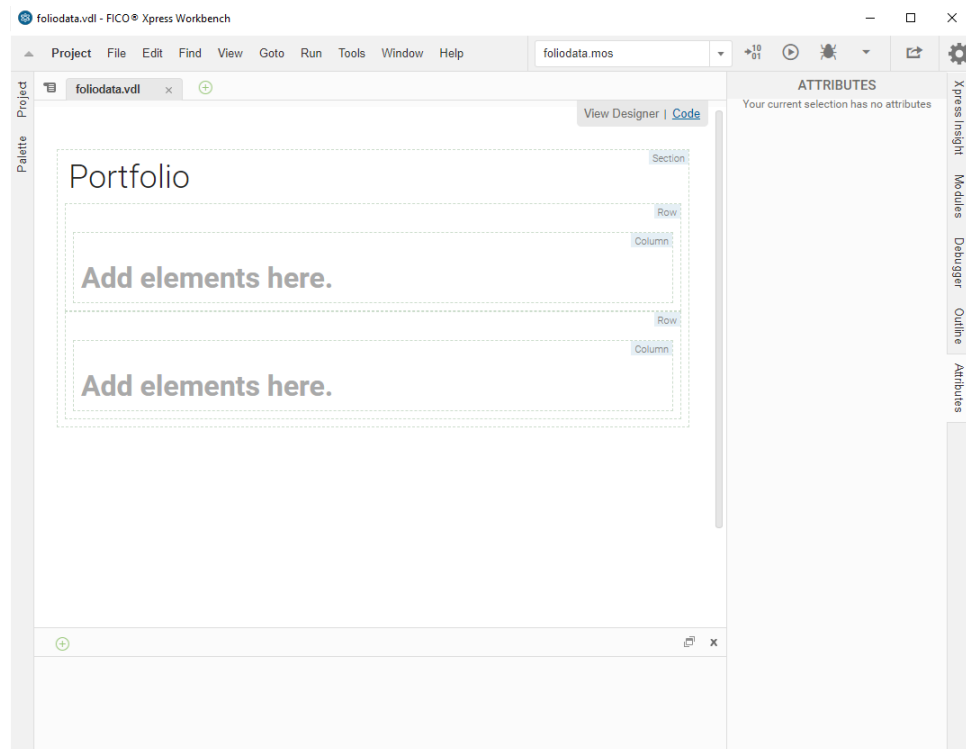
- In the **Xpress Insight** tab, click **Views > VDL View**.

3 Proceed through each page of the **New View** wizard.

- 1 In the first wizard page, enter an appropriate text string in the **View Title** field—This text is displayed as the view label by Insight and sets the default value for the **Filename**. Click **Next** to proceed.
- 2 The second wizard page enables you to select a page layout: the options are a Basic view, Two columns, or Three columns. Click **Next**.
- 3 In the final wizard page, select from the following options:

Menu option	Definition
Requires Scenario To Be Loaded:	This option is enabled by default-It can be disabled if you are creating a view containing static introductory information, or a view used to configure how scenarios are loaded.
Set As Default View:	The view will be displayed in the Xpress Insight web client when a scenario is selected-This will replace any view previously set as default view in your app.
Default When No Scenarios Selected:	The view will be displayed by default until a Scenario is selected.

4 Click **FINISH**. The new view is displayed on the artboard.



- 5 If desired, switch to the code editor view using the link at the top right of the artboard.

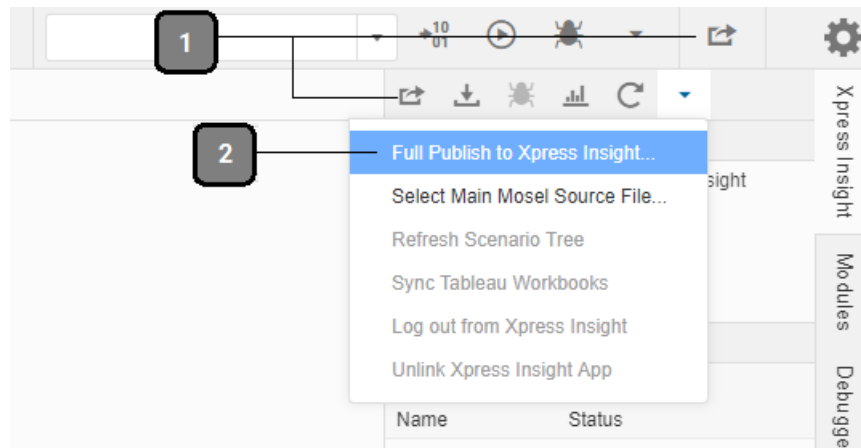
## Publishing to Xpress Insight

In this section you will publish the project to an Xpress Insight server and test it.

If you do not have access to an Xpress Insight server, contact your administrator.

✓ **Note** If you are working with the Desktop Edition and this is the first time you have used Xpress Insight, you will first need to configure a server, see [Configuring an Xpress Insight Server - Desktop Edition Only](#) on page 57.

All Xpress Workbench features that relate to Xpress Insight are accessed from the **Xpress Insight** sidebar. You can find this sidebar on the right side of the Xpress Workbench **Project** page. There are two options when publishing to Xpress Insight, Full or Quick.



- 1 Quick Publish - Use this option to include only the files that have been changed since the last publish
- 2 Full Publish - Always use this option when publishing to Xpress Insight for the first time-additionally it is recommended you use this option when adding a new view, and as the final publish prior to release.

Complete the following steps to publish a project to Xpress Insight for the first time:


- 1 Open the **Xpress Insight** sidebar, if it is not already open, by clicking its name.
  - 2 Click the **Publish** button, which is the first button from the left within the sidebar.
  - 3 Use the **Xpress Insight Server** drop-down to select the server where you want to publish the app.
  - 4 In the list of Xpress Insight apps, select **Create a new app**.
- ✓ **Note** If you are using the FAC Edition on the FICO® Analytic Cloud, the list of Xpress Insight apps may take several seconds to load.
- 5 Click **Publish**.

- 6 Log on to Xpress Insight in a new tab, open the new app, and review the VDL view that you created.

## Debugging Xpress Insight Scenarios

In this section you will debug a Mosel model within Xpress Insight by executing a scenario.

Before continuing, make sure you have followed the steps in the previous section to publish a project to Xpress Insight.

 **Note** For more on Debugging Mosel models, see [Running and Debugging Mosel Models](#) on page 41

### Setting Breakpoints

Before you start debugging, set a breakpoint in the Mosel model. When the model reaches a breakpoint, the scenario being executed pauses. While the scenario is paused, you can use the debugger to inspect the state of the model.

Breakpoints are set by clicking in the breakpoint area, which is the narrow grey area where the line numbers for the source file are displayed. When you hover the mouse to the left of the line numbers, the cursor changes to a hand icon to show that clicking here sets a breakpoint.

- 1 In the file tree, locate the **source** folder and click the arrow icon to its left.
- 2 When the folder expands to reveal **application.mos**, double-click its filename.
- 3 Locate the following line (near line 25):  
`initializations from "input.dat"`
- 4 Click in the breakpoint area for this line. A red disk appears to show that a breakpoint has been set.

### Debugging the Scenario

Complete the following steps to start debugging a scenario:

- 1 Open the **Xpress Insight** sidebar, if it is not already open, by clicking its name.
- 2 Expand the scenario tree and select the scenario you want to debug.
- 3 Click the **Debug** button, which is the third button from the left within the sidebar.

- 4 In the menu that appears, click **Debug Scenario - LOAD**.
  - The app is published to Xpress Insight before the debug session starts.
  - The line with the breakpoint is highlighted to show that model execution is suspended at that line.
  - The **Debugger** sidebar is open, where you can see the call stack and the values of the variables defined by the model.

Now that the debug session is established, you can step through the model and see how the state changes as the execution progresses.
- 5 In the **Debugger** sidebar, find the **Variables** section and click the **arrow** icon to the left of **Globals**.

You see a list of the global variables defined in the model. The value of `MY_INTEGER` is 0.
- 6 Step over the current line by clicking the second button from the left in the **Debugger** sidebar, or by pressing F10.

The next executable line is highlighted to show that model execution has moved forward.
- 7 Expand the **Globals** entry within the **Variables** section again.

The value of `MY_INTEGER` is now 10.

## *Finishing the Debug Session*

Complete the following steps to finish the debug session:

- 1 Resume the model by clicking the first button from the left in the **Debugger** sidebar, or by pressing F8.

The last line of the model is highlighted to show that the model is suspended after having finished executing. You can now review the final state of the model.
- 2 Click the **Resume** button one more time, and click **OK** when prompted.



## CHAPTER 3

# Working with Projects

---

A project is a collection of source files in Xpress Workbench and typically includes:

- Mosel source files
- Input data files for the Mosel model
- The Xpress Insight companion file
- HTML and JavaScript custom views
- VDL custom views
- Tableau workbooks

When you first log into Xpress Workbench, you see the **Dashboard** page. Here you can do the following tasks:

- See the projects on the server
- Create new projects
- Rename projects
- Delete projects
- Open projects

## Creating Projects

In this section you will create a new project and open it.

### *Creating Projects with the FAC Edition*

Complete the following steps to create a project:

- 1 From the **Dashboard** page, click **Create Project**.
- 2 Enter a name for the project.
- 3 (Optional) If you have a ZIP file containing project files that you want to upload, click **Choose File** and select your ZIP file.

**4 Click `Save`.**

- If you uploaded a ZIP file, it is extracted into the new project.
- If you did not upload a ZIP file, your new project contains the following files, which make up a minimal Xpress Insight application:
  - `application.xml` - an Xpress Insight companion file that defines the app name and some custom views
  - `source/application.mos` - a Mosel source file containing the minimal code needed to run within Xpress Insight
  - `model_resources/input.dat` - a data file which the Mosel model will read to initialize its input data
  - `client_resources/introduction.vdl` - a VDL custom view that displays information about the project
  - `client_resources/input.vdl` - a VDL custom view that displays some input data from the scenario
  - `client_resources/results.vdl` - a VDL custom view that displays some result data from the scenario
  - `client_resources/css/application.css` - a CSS file where you can add custom stylesheet rules for all the VDL views

## *Create a Project with the Desktop Edition*

A project contains all the source files that make up an Xpress Insight app or Mosel application.

Create a new Mosel project, Xpress Insight Mosel project, or Xpress Insight Python project.

- 1 To create a new project:
  - To create a simple Mosel project, choose the **Create Project > CREATE MOSEL PROJECT** menu option.
  - To create a new Xpress Insight project, you must select to work in either Python or Mosel.
    - To develop using Mosel, choose the **Create Project > CREATE INSIGHT (MOSEL) PROJECT** menu option.
    - To develop using Python, choose the **Create Project > CREATE INSIGHT (PYTHON) PROJECT** menu option.
- 2 Navigate your filesystem to a folder where you wish to store your project and click **OK**.

The project opens—depending on the kind of project you created, it will contain a minimal Xpress Insight app for Python or Mosel, or a minimal Mosel application. You can use a Mosel project to begin the development of a standalone Mosel model. The remainder of this guide focuses, however, on activities relating to Xpress Insight Mosel projects and apps.



**Note** The [Xpress Insight Python Developer Guide](#) offers information on using Python with Xpress Workbench.

## Opening and Closing Projects with the Desktop Edition

After a project has been created, you can in subsequent Xpress Workbench sessions, open and close it.

To open an existing project select **Project** from the top-level menu, choose **Open Project**, follow the prompts and use the filesystem navigator to locate the folder in which your existing project is located.

To close an open project, select **Project** from the top-level menu and choose **Close Project**.

## Renaming Projects - FAC Edition Only

Complete the following steps to rename a project:

- 1 Hover the mouse over the project you want to rename.
- 2 Click the **pencil** icon that appears.
- 3 Enter a new name for the project.
- 4 Click **Save**.

## Deleting Projects - FAC Edition Only

Complete the following steps to delete a project:

- 1 Hover the mouse over the project you want to delete.
- 2 Click the **cross** icon that appears.
- 3 Click **OK**.

## Opening Projects

**To open a project:**

- If you are using the FAC Edition, when you are logged in to the FAC, click the project's name to open the **Project** page.
- If you are using the Desktop Edition, select **Project** from the top-level menu, choose **Open Project**, follow the prompts and use the filesystem navigator to locate the folder in which your existing project is located.

**To close a project:**

- If you are using the FAC Edition, save your work and close your browser window.
- If you are using the Desktop Edition, select **Project** from the top-level menu and choose **Close Project**.

## Backing up and Restoring Projects - FAC Edition Only

If you are using the FICO® Decision Management Platform, your projects are automatically backed up.

To review the status of your backups, click **Admin** on the main Xpress Workbench menu. The **Backup status** page is displayed for the current DMP component - Xpress Workbench. It presents a list of backups, their **Start times**, their **Status** and their **Durations**.

Backups are only made when it is detected that a project changes. They occur daily for up to the last 3 days, and then weekly for up to the last 5 weeks. If you have been working consistently on several projects, you are likely to see up to eight backups per project.

Click **View logs** to see a detailed log of a backup's execution - if an error has occurred, it will contain further information.

Click **Backup Archive** in the left-hand menu and select your project in the **Project** drop-down list to see a list of backups for a specific project. Click **Download backup** to download the backup file, a gzipped tar file.

Expand the archive in a convenient location on your local machine and the backup files are available for your use.

## Displaying the Dashboard - FAC Edition Only

From the **Project** page, you can return to the **Dashboard** page by clicking the **Home** link in the page header.

## CHAPTER 4

# Editing Source Files

---

You can perform a full set of life-cycle operations on source files.

You can open, edit, and rename files in the Desktop edition. In addition, you can transfer files (upload and download) in the the FICO Analytic Cloud (FAC) edition.

## Opening Files

Complete the following steps to open a file:

- 1 Locate the file you want to open in the file tree.
- 2 Choose one of the following ways to open a file:
  - Double-click the filename.
  - Right-click the filename and click **Open** on the menu that appears.

## *Opening Files with the FAC Edition*

If a file cannot be opened within the Xpress Workbench IDE, you may be offered the option to download it for local editing.

## *Opening Files with the Desktop Edition*

Depending on the file type, you may be presented with an option to open it in an external editor—for example, an XLSX file might be opened in Excel.

You can also open Mosel or VDL source files (MOS or VDL files) in Xpress Workbench directly from your operating system's file explorer—right click on the file and choose **Open**.

## Creating Files

When it comes to creating files, Xpress Workbench offers four options:

- Using the View Designer
- Using the file tree

- Using the file menu
- Using a template

## *Creating Files from the File Tree*

Complete the following steps to create a new file from the file tree:

- 1 In the file tree, locate the folder where you want to create the new file and right-click it. If you want to create a file in the root of the project, right-click the project name in the file tree.
- 2 Click **New** on the menu that appears.
- 3 In the submenu that appears, click the file template that you want to use.
- 4 Type the filename for the file and press **Enter**.

## *Creating Files from the File Menu*

Complete the following steps to create a file from the **File** menu:

- 1 Open the **File** menu and click **New**.
- 2 In the submenu that appears, click the file template that you want to use.
- 3 Open the **File** menu and click **Save**.
- 4 In the **Save As** dialog, type the filename and select the folder where you want to create the new file.
- 5 Click **Save**.

## *Creating Files from a Template*

Complete the following steps to create a file from a predefined template:

- 1 Open the **File** menu and click **New**.
- 2 In the submenu that appears, click the file template that you want to use.
- 3 Open the **File** menu and click **Save**.
- 4 In the **Save As** dialog, type the filename and select the folder where you want to create the new file.
- 5 Click **Save**.

## *Creating Files from the View Designer*

Complete the following steps to create a new file using the New File icon in the View Designer:

The **New File** icon is displayed in the area above the artboard, to the right of the most recent tab.



Figure 3: New File icon

- 1 Click the new file icon.
- 2 Click **New** on the menu that appears, then in the sub menu, select the type of file you wish to create.  
The file is created and displayed in the central pane, surrounded by appropriate context sensitive tabs.
- 3 When you have finished editing the file, click **File > Save**.  
The **Save As** dialog is displayed.
- 4 Select the location for the file and click **SAVE**.

## Renaming Files

Complete the following steps to rename a file:

- 1 Locate the file you want to rename in the file tree.
- 2 Right-click the filename.
- 3 Click **Rename** on the menu that appears.
- 4 Type the new filename.
- 5 Press **Enter**.

## Uploading Files - FAC Edition Only

- ✓ **Note** To upload a large project containing many files, you can put them into a ZIP file and then specify the ZIP file when creating your project in Workbench. See [Create a Project with the FAC Edition](#) on page 13.

To upload a file from your computer:

- 1 Locate the file using your computer's file explorer.
  - On Windows, locate the file using File Explorer.
  - On macOS, locate the file using Finder.
- 2 In the Xpress Workbench file tree, locate the folder where you want to upload the file.
- 3 Drag the file from your computer's file explorer onto the target folder in the Xpress Workbench file tree.

- ✓ **Note** In Google Chrome, you can drag an entire folder from your computer into Xpress Workbench.

## Downloading Files - FAC Edition Only

You can download a single file or multiple files.

Download files in one of the following ways:

- 1 Download a single file or folder:
  - 1 Locate the file or folder you want to download in the file tree.
  - 2 Right-click the filename.
  - 3 Click **Download** on the menu that appears.
- 2 Download several files or folders as a zip or tar.gz archive:
  - 1 Use Ctrl-click to select multiple files. Select each file or folder you want to download in the file tree.
  - 2 When all the files are selected, right-click one of them.
  - 3 Click **Download** on the menu that appears.

On Windows, multi-file selections are downloaded as a zip archive.

On macOS, they are downloaded as a tar.gz archive.

## Downloading the Project

Complete the following steps to download the entire project as a ZIP or tar .gz archive:

- ▶ Open the **File** menu and click **Download Project**.
  - ✓ **Note** On Windows, the project is downloaded as a ZIP archive. On macOS, it is downloaded as a tar .gz archive.
  - ✓ **Note** To create an app archive file that can be imported into Xpress Insight, see [Exporting Xpress Insight Apps](#) on page 63.

## Viewing File Revision History - FAC Edition Only

Xpress Workbench keeps a history of the edits that have been made to a file. Complete the following steps to revert the file to a previous version:

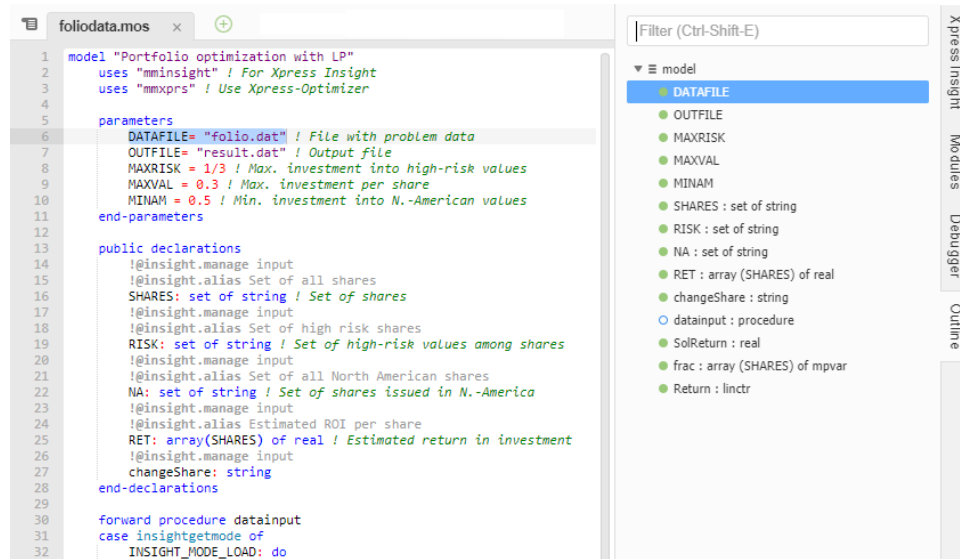
- 1 Open a file by double-clicking its filename.
- 2 Open the **File** menu and click **Show File Revision History**.
- 3 Drag the slider until the correct revision of the file is shown.
- 4 (Optional) Click the **Revert** button.



## Coding Assistance

As a fully integrated IDE, Xpress Workbench offers syntax highlighting, the ability to insert templates for common functions (snippets) via a keyboard shortcut, search for symbols, and is capable of automatically completing keywords and statements.

The structure of an open Mosel model is displayed in the **Outline** pane. Clicking on a displayed element will highlight the associated code in the code editor.



The **Modules** panel in the right-hand menu displays the available Mosel modules, and the subroutines, types, and global symbols that they expose.




These additional features are explained in the following topics:

- [Finding Text](#)
- [Highlighting Syntax](#)
- [Automatically Completing Words](#)
- [Automatically Completing Statements](#)
- [Inline Documentation](#)
- [Folding Code](#)




## Finding Text

The **Find**, **Find in Files**, and **Replace** menu options open a dialog in the text editor pane that contains some common tools.

Item	Description
	Enable Regular expression mode

Item	Description
	Enable case sensitivity
	Match whole words only
	Wrap the search to the start of the document

In addition, each menu option features a unique tool.

Function	Tool icon	Description
<b>Find</b>		Search the selected text only
<b>Find in Files</b>		Show the search results in the console
<b>Replace</b>		Attempt to match the case of the text being replaced

Complete the following steps to find a text string in a given file:

- 1 Open the file by double-clicking its filename.
- 2 Open the **Find** menu and click **Find**.
  - On Windows, you can instead press **Ctrl-F**.
  - On macOS, you can instead press **Cmd-F**.

The **Find** box is displayed below the editor.
- 3 Type the text string you want to find and select options to configure the search.
 

Hover over the options to see the related tool-tip.
- 4 Press **Enter**.
- 5 (Optional) To find the next occurrence click the greater-than symbol to the right of the **Find** box.
  - On Windows, you can find the next occurrence by pressing **Ctrl-K**.
  - On macOS, you can find the next occurrence by pressing **Cmd-K**.
- 6 (Optional) To find the previous occurrence, click the less-than symbol to the right of the **Find** box.

## Replacing Text

Complete the following steps to replace one text string with another:

- 1 Find the text string you want to replace, by following the instructions in the previous section.
- 2 In the right most text box, type a replacement text string.

3 Click **Replace**.

The currently selected occurrence is replaced, and the editor highlights the next occurrence. The currently selected occurrence is replaced, and the editor highlights the next occurrence.

4 (Optional) Repeatedly click **Replace** to continue replacing further occurrences.

5 (Optional) To replace all occurrences in the file, click **Replace All**.

## Finding Text Across Many Files

Complete the following steps to search in all project files for a text string:

1 Open the **Find** menu and click **Find in Files**.

- On Windows, you can instead press **Ctrl-Shift-F**.
- On macOS, you can instead press **Cmd-Shift-F**.

2 In the **Find** box that appears below the editor, type the text string you want to find.

3 (Optional) In the box to the right of the find button, enter a pattern for the filenames you want to match. Several patterns can be separated by a comma, and files can be excluded by beginning a pattern with a minus sign.

By default, all files are matched except those whose names begin with a dot.

4 Click the **Find** button.

The **Search Results** window appears below the editor.

5 Double-click a line in the **Search Results** window to open the source file in the editor.

## Replacing Text Across Many Files

To replace all occurrences of a text string in all project files with another text string:

1 Open the **Find** menu and click **Find in Files**.

- On Windows, you can instead press **Ctrl-Shift-F**.
- On macOS, you can instead press **Cmd-Shift-F**.

2 In the **Find** box that appears below the editor, type the text string you want to find.

3 (Optional) In the box to the right of the find button, enter a pattern for the filenames you want to match. Several patterns can be separated by a comma, and files can be excluded by beginning a pattern with a minus sign.

By default, all files are matched except those whose names begin with a dot.

4 In the **Replace With** box, to the right of the **Find** box, type the string with which you want to replace the found text.

5 Click **Replace**.

## Highlighting Syntax

If Xpress Workbench supports the language of a source file, it highlights the source code according to the syntax of the language. Supported languages include:

- Mosel
- XML
- VDL
- HTML
- JavaScript

The language of a source file is inferred by Xpress Workbench using the filename extension. Complete the following steps to override the language for the open source file:

- 1 Open the **View** menu and click **Syntax**.
- 2 Click the language you want to use for the source file.

## Automatically Completing Words

When editing a source file, Xpress Workbench tries to automatically complete the word that you are typing based on words that have appeared elsewhere in the same source file or in other source files in the project.

Perform the following steps to autocomplete a word:

- 1 Begin typing the name of a variable or subroutine that appears elsewhere in a source file.  
Xpress Workbench displays a list of words that begin with the characters you typed.
- 2 Use the cursor keys to select the word that you wish to autocomplete.
- 3 Press **Enter** or **Tab**.

## Automatically Completing Statements

When editing a source file, Xpress Workbench automatically inserts the closing tag for a typed statement.

Complete the following steps to autocomplete a statement:

- Begin typing the name of a statement. The Xpress Workbench IDE displays a list of words that begin with the characters you typed.
- Use the cursor keys to select the statement that you wish to autocomplete.
- Press **Enter** or **Tab**—the statement is added to the Mosel file.



**Tip** Press **Tab** while the modifying the statement to navigate into the next editable area in the statement.

Xpress Workbench will autocomplete the following Mosel statements:

- model/end-model
- package/end-package
- parameters/end-parameters
- declarations/end-declarations
- record/end-record
- requirements/end-requirements
- initializations/end-initializations
- forall do/end-do
- while do/end-do
- do/end-do
- repeat/until
- if/then/else/end-if
- case/end-case
- procedure/end-procedure
- function/end-function

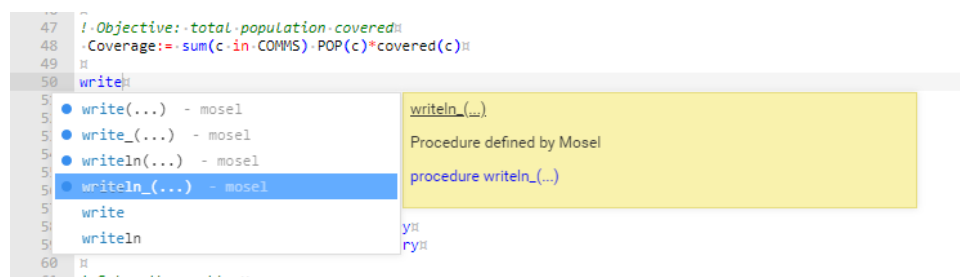
## Inline Documentation

When editing a Mosel file, the Xpress Workbench IDE displays the help text for standard Mosel library subroutines.

In an open Mosel file, when you begin typing one of the available subroutines, the autocomplete dialog will display all the different variations of that subroutine. If you highlight one of the options in this window, either by moving the keyboard cursor or hovering the mouse over an option, the inline documentation window will display any inline help text for the highlighted item.



**Note** If you highlight an item with the keyboard and hover the mouse, the mouse hover has priority.



## Folding Code

In supported source file types, code blocks in the file can be folded into a single line, hiding the contents of the block and making the file easier to read.

### Folding Code with the Mouse

Complete the following steps to fold or unfold a code block in a Mosel source file:

- 1 Hover the mouse in the narrow grey area where the line numbers for the source file are displayed, and look for the small arrows which appear to the right of the line numbers.
- 2 Click an arrow to fold or unfold the code block that begins on that line.

### Folding Code with the Keyboard

Complete the following steps to fold and unfold code blocks using the keyboard:

- 1 Position the cursor anywhere within a code block that you want to fold.
- 2 Press **F2**
- 3 To unfold a code block, position the cursor anywhere on a folded line, and press **F2** again.

## Collaborating with Other Users - FAC Edition Only

Xpress Workbench is a multi-user application with features for collaborating with other users.

When another user opens a project that you have open in your browser, a notification icon appears in the top right of the screen showing you their name. If this user opens a file that you also have open in the project, the position of their cursor within the file is shown on your screen. If the user edits the file, their edits appear on your screen in real time. Xpress Workbench highlights the lines of the file that were edited by each user who is collaborating on the project.



**Note** In this version of **Xpress Workbench**, all projects are public and can be opened and edited by any other user who has an account on the server.

## Sending Messages to Users

To send a message to another user who has the same project open:

- 1 Open the **Collaborate** sidebar by clicking its name.
- 2 Type your message into the text box at the bottom of the sidebar.

- 3 Press **Enter** to send the message.

All other users who have the project open are notified with the message.

When a user opens the **Collaborate** sidebar, they see a history of the messages that were sent by all users of the project.





# Running and Debugging Mosel Models


---

Xpress Workbench allows you to run Mosel models in a standalone environment. Mosel models are compiled, loaded, and run when they are executed in Workbench. When a Mosel model is compiled, the syntax of the model and the libraries implemented in Mosel are checked but no operation is executed. The result of the compilation is a Binary Model (BIM) that is saved in a second file. In this form, the model is ready to be executed and the source file is not required any more. To actually 'run' the model, the BIM file must be read in again by Mosel and then executed.

Running a Mosel Xpress Insight app in Workbench, outside of Xpress Insight, can be faster and simpler than publishing the complete app to an Xpress Insight server whenever you want to test it. When you run the Mosel model in Workbench, it is also re-compiled if the source has changed.

## Running Models

Complete the following steps to run a model in Mosel:

- 1 In the **Run Configurations** drop-down list, select the Mosel source file that you want to run.  
The **Run Configurations** drop-down list is located to the right of the main menu bar, near the top of the screen.
  - 2 Click the **Run** button, which is the second button to the right of the **Run Configurations** drop-down list.
  - 3 Check the **Output** window that opens at the bottom of the screen to see the output printed by the Mosel model. If the model expects some input, you can type in the output window, too.
-  **Note** Xpress Workbench compiles the Mosel source file and any packages that it imports before running the Model. Any syntax errors are highlighted in the code window, and compilation errors are displayed in the **Output** window. Packages can be compiled but not run, as a debugging step.

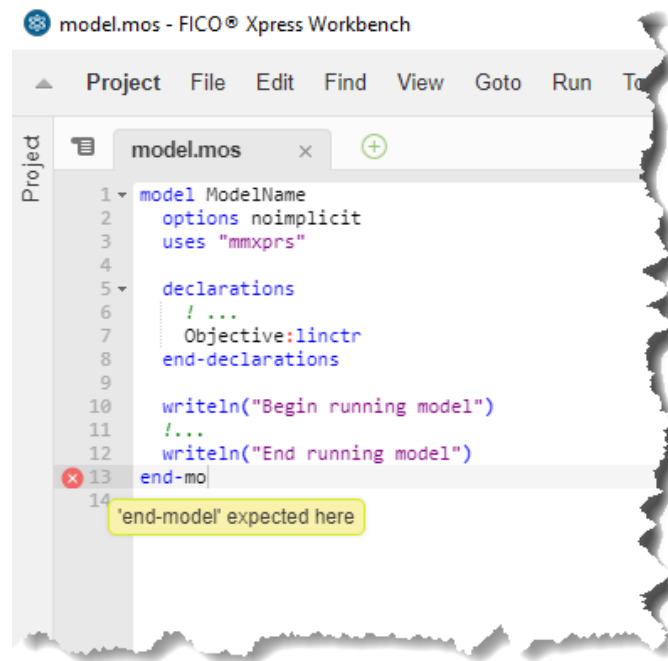


Figure 4: Highlighted syntax error

- 4 Run the model, choosing one of two methods:
    - Click the **Run** button and then click the menu item for the Mosel source file you want to run.
    - After selecting the Mosel source file in the **Run Configurations** drop-down list, press CTRL - F5.
- ✓ **Note** Only source files that declare a Mosel model can be executed. Source files that declare a Mosel package cannot be run directly; instead, run the main Mosel source file, which begins and ends with the **model** and **end-model** keywords.

## Stopping a Model

A model that is executing in Mosel can be stopped at any time. Complete the following steps to stop a model:

- 1 Click the **Stop** button in the **Output** window.
  - 2 Wait for the model to finish executing.
- ✓ **Note** If a model is being debugged, this stops the debug session immediately. You cannot review the final state of the model before it exits.

## Changing the Working Directory

By default, the working directory of the running Mosel model is the directory containing the source file. If the model reads any input files or writes any output files, they are read from or written to this directory.

Complete the following steps to change the working directory:

- 1 Run the model by following the instructions above.
- 2 Either wait for the model to finish executing or click the **Stop** button to interrupt it.
- 3 When the model has finished executing, click the **CWD** button in the **Output** window.
- 4 Choose the working directory for the model and click **Select**.
- 5 Run the model again. This time it runs in the newly selected working directory.



**Note** If the project contains a top-level directory named **model\_resources**, the working directory is set to this directory by default. This makes the runtime environment for an Xpress Insight app similar to how it is when the app runs on an Xpress Insight server, when Xpress Insight copies the contents of the **model\_resources** directory into the working directory.

## Debugging Models

Xpress Insight provides a fully-featured Mosel debugger. The debugger allows you to:

- Set and clear breakpoints.
- Step over the next program statement.
- Step into a subroutine call at the current statement.
- Step out of the current subroutine call.
- View the current call stack.
- View the values of local variables within the current subroutine call.
- View the values of global variables in the main model.
- View the values of global variables defined in each package.
- Evaluate expressions as you step through the program.


## Starting a Debug Session

Complete the following steps to run a model in debug mode:

- 1 In the **Run Configurations** drop-down list, select the Mosel source file that you want to run.

- 2 Click the **Debug** button, which is third button to the right of the **Run Configurations** drop-down list.


Alternatively, press F5.

 **Note** Only one Mosel model can be debugged at any given time. If you run a second Mosel model while the first one is still active, the debugger does not attach to the second model and the **Run in Debug Mode** button is deselected by default in its output window.

When the model has finished executing, the **Debugger** sidebar opens and the last line of the model is highlighted to show that the model is suspended in the debugger after it has finished executing. Xpress Workbench suspends the model execution here so that you can review the final state of the model.

- 3 When you are ready, you can terminate the model execution by clicking the **Resume** button in the **Debugger** sidebar, clicking **OK** when prompted.


Mosel exits


 **Note** Once Mosel has exited, you can no longer see the values of any model variables. Make sure you have finished reviewing the model data before you terminate Mosel.

## Setting and Removing Breakpoints

Breakpoints are set by clicking in the breakpoint area, which is the narrow grey area where the line numbers for the source file are displayed. When you hover the mouse to the left of the line numbers, the cursor changes to a hand icon to show that clicking in that spot sets a breakpoint.

- 1 To set a breakpoint, click in the breakpoint area to the left of the line where you want to set the breakpoint.  
A red disk appears to show that a breakpoint was set.
- 2 To remove a breakpoint, click on the red disc icon in the breakpoint area to the left of the line where you want to remove the breakpoint.  
The red disc icon disappears from the breakpoint area.
- 3 To disable a breakpoint, open the **Debugger** sidebar if it is not already open, by clicking on its name and in the breakpoints section, clear the checkbox for the breakpoint that you want to disable.  
The debugger no longer suspends at this breakpoint, until you enable it again.
- 4 To disable all breakpoints in the model, open the **Debugger** sidebar if it is not already open, by clicking on its name and click the **Toggle Breakpoints** button, which is the fifth button from the left in the **Debugger** sidebar.

 **Note** Conditional breakpoints are not supported in this version of Xpress Workbench.

 **Note** Breaking automatically on exceptions is not supported in this version of Xpress Workbench.

## *Suspending and Resuming the Model*

The model can be suspended at any time while it is executing, allowing you to inspect its state. Complete the following steps to suspend the model:

- 1 Open the **Debugger** sidebar, if it is not already open, by clicking on its name.
- 2 Click the **Pause** button, which is the first button from the left in the **Debugger** sidebar.  
While the model is executing, this button has a pause icon.
- 3 To resume execution, open the **Debugger** sidebar, if it is not already open, by clicking on its name.
- 4 Click the **Resume** button, which is the first button from the left in the **Debugger** sidebar.

Alternatively, press F8.

While the model is suspended, this button has a **play** icon.

## *Stepping through the Model*

While the model is suspended, you can step through to the next statement, depending on which action you choose.

- 1 Open the **Debugger** sidebar if it is not already open, by clicking on its name.
- 2 Step through a model using any of the following steps:
  - To step to the next statement in the current subroutine, click the **Step Over** button, which is the second button from the left in the **Debugger** sidebar.  
Alternatively, press F10.
  - If the current statement contains a call to a subroutine, you can step into the subroutine. The debugger suspends at the first statement of the subroutine. Click the **Step Into** button, which is the third button from the left in the **Debugger** sidebar.  
Alternatively, press F11.
  - If the current statement is within a subroutine, you can step out of the subroutine. The debugger suspends at the statement after the calling statement. Click the **Step Out** button, which is the fourth button from the left in the **Debugger** sidebar.  
Alternatively, press Shift-F11.

## Examining Variables

While the model is suspended, you can inspect its data using the **Variables** section in the **Debugger** sidebar.

The variables section contains the following entries:

- An entry for each local variable defined in the current subroutine  
If the current statement is not within a subroutine, no local variables are available.
- Global symbols  
Expand this entry to see variables that are declared at the top level of the model.
- Parameters  
Expand this entry to see runtime parameters defined in the model.
- Constants  
Expand this entry to see any constant declarations in the model.
- An entry for each namespace defined in the model  
Expand these entries to see the global variables defined in each namespace.

The **Variables** section shows the name, type, and value of each variable. If a variable holds structured data, such as a set, array, list, or user-defined record, the entry for that variable can be expanded. Complete the following steps to expand a variable:

- 1 Click the arrow to the left of its name.  
The entries within the data structure appears.
- 2 Continue to expand the entries until you see the data that you require.



**Note** You can also see the value of a variable by hovering the mouse over a reference to the variable in the active source file.

## Examining Arrays

When inspecting array data in the debugger, Workbench shows a maximum of 20 array entries in the variables tree.


Complete the following steps to view the full contents of the array in a table:


- 1 While suspended at a breakpoint, open the **Debugger** sidebar.
- 2 In the **Variables** tree, find the array you want to inspect and right-click its name.  
Sets and lists can also be inspected in tables.
- 3 In the menu that appears, click **Inspect in table**.

A new tab opens containing a table. The table has a row for every entry in the array, a column for each index set of the array, and a column for the array values.

#### 4 You can inspect the table in the following ways:

- Click a column header to sort the table by that column.
- Filter the table by entering a filter value in one or more of the filter cells, which appear at the top of the table directly under the column headings.
- To inspect the array elements in more detail, hover the mouse over the table cell until a tooltip appears.
  - If the table cell contains a record, you can expand the tooltip to show its member values.
  - If the table cell contains a nested array, you can expand the tooltip to show its elements.

 **Note** You can increase the number of array elements displayed in the variables tree by changing the **Maximum Collection Items to Show in Debugger** setting in your user preferences.

 **Note** Arrays with more 10,000 entries cannot be filtered or sorted for performance reasons. To sort or filter large arrays you must save the data as a CSV file and open it in another application. The limit can be increased by changing the **Maximum Collection Table with Sorting/Filtering** setting in your user preferences, but this may have a significant impact on the performance of the application.

## Saving Array Data as CSV Files

Array data can be saved as a CSV file so that it can be analyzed using other applications.

To export an array as a CSV file:

- 1 Open the array in a table.  
See [Examining Arrays](#) on page 46.
- 2 Click **Save as CSV**.
- 3 Choose where you would like to save the CSV file and click **Save**.

If you are using the FICO Analytic Cloud edition of Workbench, your browser may download the file automatically without prompting you to choose where to save it.

## Evaluating Expressions

In the **Watch Expressions** section of the **Debugger** sidebar, you can evaluate any number of variables and expressions in the context of the current subroutine.

The values are updated each time you step through the program.  
You can add or remove watch expressions.

- 1 Open the **Debugger** sidebar, if it is not already open, by clicking on its name.

- 2 Complete the following steps to add a watch expression while the debugger is suspended:
  - a Double-click on the text **Type your expression here** in the **Watch Expressions** section.
  - b Enter the name of a variable or an expression involving one or more variables.
  - c Press **Enter**.
- 3 Complete the following steps to remove a watch expression:
  - a Right-click the watch expression you want to remove.
  - b Click **Remove Watch Expression** on the menu that appears.

## Using the Immediate Window

The **Immediate** window lets you evaluate expressions in an environment similar to a command prompt. Complete the following steps to open the **Immediate** window and evaluate an expression:

- 1 Open the **Debugger** sidebar, if it is not already open, by clicking on its name.
- 2 At the bottom of the sidebar, click **Immediate**.
- 3 In the window that opens, type an expression involving one or more variables, and press **Enter**.

The value of the expression is printed on the next line.

## Compiling Models

Complete the following steps to compile a Mosel source file to a BIM file:

- 1 From the **Run Configurations** drop-down list (located to the right of the main menu bar, near the top of the screen), select the Mosel source file that you want to compile.
- 2 Click the **Compile** button, which is the button directly to the right of the **Run Configurations** drop-down list.  
Alternatively, press **Ctrl-B**.
- 3 Check the **Output** window for any compilation errors.

If compilation is successful, a BIM file appears with the source file in the file tree.



**Note** If the source file loads any packages, Xpress Workbench compiles the packages before compiling the selected source file. Intermediate BIM files are placed in a folder named out by default.



## Configuring Compiler Options

Complete the following steps to change the Mosel options which are used when compiling source files:

- 1 Click **Run Tools** (the fourth button to the right of the **Run Configurations** drop-down list).
- 2 In the menu that appears, click **Compiler Options**.
- 3 In the **Compiler Options** dialog box, configure the Mosel options that should be used when compiling source files.  
See [Compiler Options](#) on page 49 for more information.

## Compiler Options

You can configure how Mosel compiles source files in the **Compiler Options** dialog box.

Configure the following settings:

- **BIM File Output Folder:**  
Configures the folder within the project where intermediate BIM files will be placed during compilation.
- **Debugging Information:**  
Determines the information which is included in the generated BIM file. Can be one of the following options:
  - Remove private [option->object] names.  
Names of private variables, constraints, and subroutines are not saved into the BIM file.
  - Add debugging information.  
The **-g** command line flag is passed to the Mosel compiler. This option is required to locate a runtime error.
  - Add debugging and tracing information.  
The **-G** command line flag is passed to the Mosel compiler. This option is required to run the model with the debugger.
- **Private Declarations in Mosel are stripped on compilation by default.**
  - Declarations are private by default unless explicitly prefixed with the public keyword. Any entities in your Mosel code that need to be visible to Xpress Insight must be declared public.

- **Implicit Declarations:** Determines the behavior of the Mosel compiler when it encounters a symbol which was not explicitly declared in a declarations block. Can be one of the following options:
  - Allow implicit declarations.  
No command line flags are passed to the Mosel compiler. Implicit declarations are silently accepted.
  - Forbid implicit declarations.  
The **-ni** command line flag is passed to the Mosel compiler. Implicit declarations cause the compilation to fail with an error.
  - Warn about implicit declarations.  
The **-wi** command line flag is passed to the Mosel compiler. Implicit declarations are accepted but the compiler will print a warning.
- **Treat Warnings as Errors.**  
If checked, the **-we** command line flag is passed to the Mosel compiler. Compiler warnings are handled like errors such that any warning will make the compilation fail.
- **Generate Doc Annotations.**  
If checked, the **-D** command line flag is passed to the Mosel compiler. Documentation annotations are generated in the resulting BIM file. (By default, documentation annotations are ignored.)
- **Enable Xbim Extension.**  
If checked, the **-I** command line flag is passed to the Mosel compiler. The xbim extension is enabled, causing additional symbol information to be stored in the generated BIM file (in particular array index names).
- ✓ **Note** When a model is run from Xpress Workbench, the **-g** option is automatically added so that stack traces can be displayed if an error occurs. When a model is debugged from Xpress Workbench, the **-G** option is automatically added.

For more information about Mosel compiler options, see the [Mosel Reference Manual](#).

## Running Submodels and Parallel Solving

Xpress Workbench supports running Mosel models that load and run submodels. Before your master model can load and run a submodel, you must ensure that the submodel has been compiled. Complete the following steps to run a model that uses submodels:

- 1 Open the submodel source file by double-clicking its filename.
- 2 Open the **Run** menu and click **Build**.  
If compilation is successful, a BIM file appears alongside the source file in the file tree.
- 3 Ensure that the submodel BIM file is located in the working directory of the master model.

- 4 Select the master model source file in the **Run Configurations** drop-down list.
- 5 Click the **Run** button, which is the second button to the right of the **Run Configurations** drop-down list.
- 6 If a submodel encounters a runtime error, the master model will be suspended on whichever line was being executed when the error occurred. Click **Resume** to continue executing the master model.

If you change the submodel source file, make sure you recompile it before running the master model.

✓ **Note** Manually compiling the submodel is not needed if your master model compiles it programmatically using the **compile** function in Mosel.

✓ **Note** You cannot debug a submodel while it is running within the master model. Breakpoints set in submodel source files are ignored. To debug a submodel in standalone mode, select the main source file for the submodel in the **Run Configurations** drop-down list and run it as usual.

✓ **Note** (FAC edition only) Remote instances of Mosel are not supported. Submodels must be loaded and run in the main Mosel process.

## Preparing Template Files for Mosel Deployment Wrappers

It is straightforward to execute Mosel models from applications written in C, C#, Java, VB.NET and VBA.

To assist with this, Xpress Workbench contains—for each of those languages—template source files that can load and execute a pre-compiled Mosel model.

To deploy a Mosel model in an application:

- 1 Compile your Mosel model in Xpress Workbench.
- 2 Open the **File** menu, click **New** then choose the appropriate Mosel deployment file template.
- 3 Find the constant with the value `test.bim` near the top of the file and change its value to the name of your BIM file.
- 4 Save the file then copy this source file and your BIM file into your project in Microsoft Visual Studio, Eclipse IDE or other development environment.

✓ **Note** Xpress Workbench cannot compile or run source files written in C, C#, Java, VB.NET or VBA. To compile and run the generated source file, use an IDE tool which supports these languages, such as those listed above.

For more information on how to interact with Mosel from C, C#, Java, VB.NET and VBA, see the *Mosel User Guide* and the *Mosel Libraries Reference Manual*.

## Tuning Mosel Models

For a given optimization problem, setting suitable control parameters frequently results in improved performance such as solution time reduction. The FICO® Xpress

built-in tuner can help a user to identify such a set of control settings that allows the Xpress solver to solve problems faster than by using defaults.

See section *Using the Tuner* in the [Xpress Optimizer Reference Manual](#) for more information on how to tune optimization problems.

## Starting a Tuning Session


Complete the following steps to start tuning a Mosel model:

- 1 In the **Run Configurations** drop-down list, select the Mosel source file that you want to tune.
- 2 Click the **Run Tools** button (the fourth button to the right of the **Run Configurations** drop-down list).
- 3 In the menu that appears, click **Show Run Dialog**.
- 4 In the **Run Model** dialog, ensure that the **Run Mode** field is set to **Tune the Model** and configure the settings for the tuning session.  
See [Tuner Settings](#) on page 53 for more information.
- 5 Click **Run**.

The tuner logs from will appear in an **Output** window at the bottom of the screen.

- 6 When you see the message **New tuner results** at the top of the screen, click the link.

A new tab will open, displaying the tuner results in a table.

 **Note** Tuner result files and log files are written to a directory named `tuneroutput` within the project. You can view the tuner results table for a previous tuner session by locating the XTR file in the Xpress Workbench file tree and double-clicking it.

The columns of the tuner results table are as follows:

Column	Description
Problem	The name of the tuning session, as specified in the <b>Tune Model</b> dialog.
Type	The type of problem that was solved, e.g. MIP.
Xpress Version	The version of the Xpress that was used for the tuning session.
Timestamp	The date and time of each tuner run, in the form <code>yyyymmdd.hhmmss.micros</code> .
Logs	Click in this cell to open the Xpress log file for the tuner run.
Tuner Threads	The number of times the tuner solved the problem as part of each run, including permutations.
one or more columns...	A column for each Xpress control which the tuner tried during the session.
Code	Click this cell to see the Mosel code which sets the Xpress controls as they were for this tuner run.

Column	Description
Status	The status of the tune run, e.g. whether or not the problem was solved within the time limit.
Run Time	The total time taken to solve the problem.
Objective	The value of the objective function at the end of the tuner run.

For MIP problems, the following columns are also present:

Column	Description
Bound	The best bound at the end of the tuner run.
Gap	The gap at the end of the tuner run, as a percentage.
Integral	The integral at the end of the tuner run.
Solutions	The number of solutions found during the tuner run.
Nodes	The number of tree nodes searched during the tuner run.

## Tuner Settings

In the **Run Model** dialog box, you can configure the following settings that affect the tuner's behavior:

- **Session Name:** A name for the tuning session. It will determine the directory where tuner result files are written. The name will also appear in the tuner table. If you plan to combine the data from several tuner sessions, the name can serve to differentiate the results from different sessions.
- **Method File:** A path to a tuner method file. A tuner method consists of a list of controls for the tuner to try with. You can create a template tuner method file by opening the **File** menu, clicking **New** and choosing **Tuner Method**.
- **Tuner Time Limit:** The maximum time in seconds to spend tuning the model. If set to zero, no time limit will be imposed.
- **Permutations:** The number of times that the problem will be randomly permuted and solved while tuning it. This can improve stability by avoiding lucky controls that show improved performance by coincidence rather than structural reasons. See section *Tuner with Problem Permutations* in the [Xpress Optimizer Reference Manual](#).
- **Tuner Threads:** The number of different control settings that will be run simultaneously.

## Manually Enabling the Tuner

In some cases, the **Run Model** dialog is not sufficient, and you must modify the source code of your model to enable the tuner:

- If your model executes a submodel using Mosel's run subroutine, the tuner will not be enabled for the submodel.
- If your model solves more than one optimization problem, the tuner will be enabled for all problems defined in the model, but the custom tuner settings specified in the **Run Model** dialog will only apply to the top-level problem.

There are two ways to manually enable the tuner. The first is to pass the XPRS\_TUNE option to the minimize or maximize call. For example, to tune a submodel when it is executed from a master model, the submodel could contain the following code:

```
minimize(XPRS_TUNE, Objective)
```

The second way is to set the xprs\_tunermode parameter:

```
setparam("xprs_tunermode", 1)
minimize(Objective)
```

If you want to tune the main problem in your Mosel model, but disable the tuner for the subproblems, you can set xprs\_tunermode to zero when solving the subproblem, and then set it to 1 when solving the main problem.

```
with subprob do
  setparam("xprs_tunermode", 0) ! Disable the tuner
  minimize(SecondaryObjective)
done
setparam("xprs_tunermode", 1) ! Enable the tuner
minimize(PrimaryObjective)
```

To customize the tuner settings when tuning a subproblem, the relevant parameters must be set inside the with block which defines the subproblem. For example:

```
with subprob do
  setparam("xprs_tunermethodfile", "tunermethod.xtm")
  setparam("xprs_tunerthreads", 4)
  minimize(SecondaryObjective)
done
```

For more information about parameters that affect the tuner's behavior, see *Using the Tuner* in the [Optimizer Reference Manual](#). See the [Mosel Language Reference Manual](#) for more information about working with multiple problems in Mosel.

## Profiling Mosel Models

Running a model with the profiler provides detailed information on what part of the code is actually executed and how much time each statement requires. This information may be helpful for optimizing the model (by locating hot spots where the code is using a great deal of computer time) and also for building test suites (by checking whether the data sets used in the test set exercise all statements of a given model).


### *Starting a Profiling Session*

Complete the following steps to start profiling a Mosel model:

- 1 In the **Run Configurations** drop-down list, select the Mosel source file that you want to profile.
- 2 Click **Run Tools** (the fourth button to the right of the **Run Configurations** drop-down list).
- 3 In the menu that appears, click **Show Run Dialog**.
- 4 In the **Run Model** dialog box, set the **Run Mode** field to **Profile the Model**.
- 5 Click **Run**.

When the model has finished running, a file with the extension `.prof` is written into the same directory as the source file. Each line in this file contains:

- The number of times the corresponding statement has been executed.
- The total amount of time (in seconds) spent on this particular line (this measure is not valid if the statement is a recursive call).
- The elapsed time (in seconds) between the beginning of the execution and the last time the line was executed.
- The text of the model source.

 **Note** If the model runs additional submodels via `mmjobs`, the `.prof` files are written similarly to the way the main model is written.

## Collecting Coverage Data

The profiler can also collect coverage data. The difference between this and profiling the model is that the generated files have the extension `.cov` and they only indicate the number of times each statement has been executed.

Additionally, existing `.cov` files are updated instead of being replaced, such that iteration counts of each statement are added up.

After execution, the total execution time and source file coverage information is displayed in the output pane and an annotated version of each source file is written containing the number of times each statement has been executed. The coverage logs will tell you whether two executions of a model take the same path through the

source code, independent of the optimization results which might vary if the solver version changes.

- 1 Complete the following steps to collect coverage data for a Mosel model:
  - 1 In the **Run Configurations** drop-down list, select the Mosel source file for which you want to collect coverage data.
  - 2 Click **Run Tools**, the fourth button to the right of the **Run Configurations** drop-down list.
  - 3 In the menu that appears, click **Show Run Dialog**.
  - 4 In the **Run Model** dialog box, set the **Run Mode** field to **Collect Coverage Data**.
  - 5 Click **Run**.

## Tracing Model Execution

Mosel can execute a model in trace mode. When executing in trace mode, the line number of each executed line is written to a trace file named `sourcefile-timestamp.trac`. This feature can be used to identify the point of failure if a model terminates unexpectedly. Trace mode output can be controlled via annotations, for more see [Mosel Command: Tracing Mode](#).

Complete the following steps to execute a Mosel model in trace mode:

- 1 In the **Run Configurations** drop-down list, select the **Mosel source file** for which you want to collect coverage data.
- 2 Click **Run Tools**, the fourth button to the right of the **Run Configurations** drop-down list.
- 3 In the menu that appears, click **Show Run Dialog**.
- 4 In the **Run Model** dialog box, set the **Run Mode** field to **Trace Model Execution**.
- 5 Click **Run**.



## CHAPTER 6

# Working with Xpress Insight

---

Xpress Workbench can assist you with many tasks during the development of Xpress Insight apps, including:

- Publishing your project to Xpress Insight
- Building an Xpress Insight app archive
- Debugging scenarios running on an Xpress Insight server
- Downloading new and modified workbooks from Tableau Server into your project
- Autocompletion in VDL source files

## Configuring an Xpress Insight Server - Desktop Edition Only

You need to configure an Xpress Insight server before you can publish to or otherwise interact with it.

The first time you try to publish to Xpress Insight, you will encounter a dialog box prompting you to provide configuration information.

When it appears, review the list in the left-hand pane and select the appropriate server. If your planned server is not in the list, click **Add** and follow the prompts to add a new server.

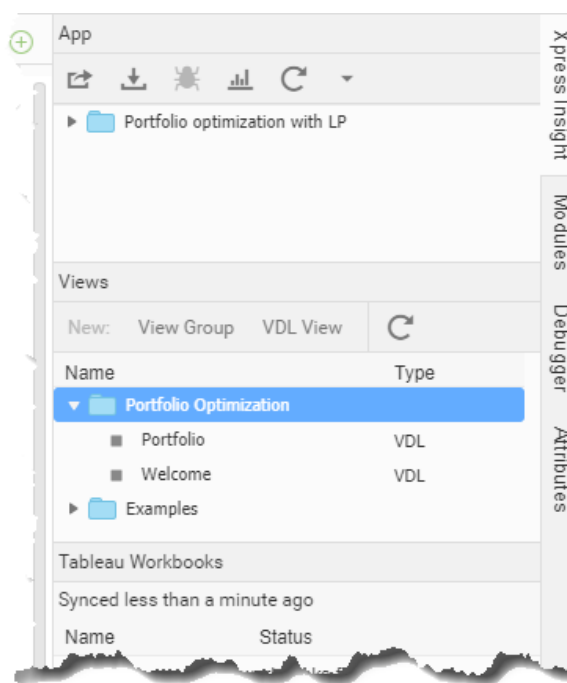
Complete the other fields and click **Save** to save your settings. By default, a locally-installed instance of Xpress Insight is active at the URL <http://localhost:8860>. Trial version user credentials are `admin/admin123` - contact your system administrator for details of other login options.

## Managing Insight Views

The Xpress Insight sidebar in Xpress Workbench features a **Views** pane that enables you to create, re-arrange, group, and delete views.

### *Views pane*

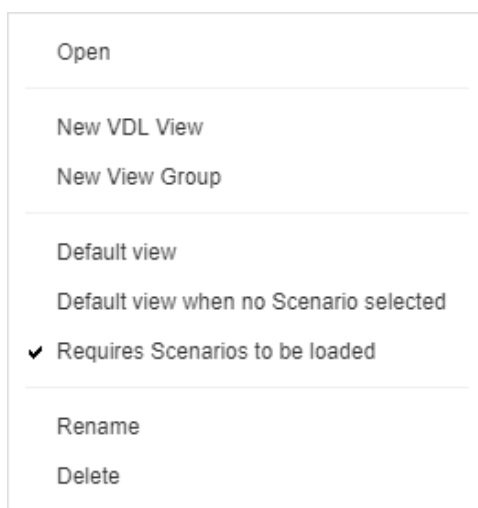
The **Views** pane is displayed when the Xpress Insight sidebar is open.



You can double click on any file displayed in the Views pane to open it. There are three buttons at the top of the **Views** pane:

- **View Group**—Create a new View Group, for more see [Managing View Groups](#).
- **VDL View**—Add a new VDL View, for more see [Managing Views](#).
- **Refresh (icon)**—Refresh the display in the **Views** pane

You can perform several view management tasks by right clicking on a folder or view in the Views pane—The menu content reflects the options available to the item you clicked on.



## Managing Views

The process of creating Views is also described in [Creating a New View](#) on page 20.

There are three ways to create a new view:

- In Xpress Workbench, you can click **File > New > Insight View(VDL)**.
- In the View Designer, click the green plus icon in the tab area.
- In the View Designer, **Xpress Insight > Views** pane, click **VDL View**.

The **New View Wizard** is displayed. Proceed through each page of the New View wizard before clicking **Finish**. The new view is displayed on the artboard.

## Importing Views

Existing Views can be imported into the project using the Import Insight View option.

- 1 Copy a suitable source file into the Project folder, either through the operating system explorer software, or by dragging and dropping the file onto the Project pane in Workbench.

You can import .html or .vdl into the client\_resources folder, or .twb or .twbx into the tableau folder.

- 2 Right-click on the file to open the **Import Insight View** Wizard-This wizard is only displayed if an appropriate file type is selected.

- 3 Enter a Title for the View, and select the View Group. Click Next.
- 4 Select the View Options.

Menu option	Definition
Requires Scenario To Be Loaded:	This option is enabled by default-It can be disabled if you are creating a view containing static introductory information, or a view used to configure how scenarios are loaded.

Menu option	Definition
Set As Default View:	The view will be displayed in the Xpress Insight web client when a scenario selected- This will replace any view previously set as default view in your app.
Default When No Scenarios Selected:	The view will be displayed by default until a Scenario is selected.

5 Click **FINISH**.

The new view is displayed on the artboard.

## Deleting Views

To delete a View:

- In the **Views** pane, right click on the View and select **Delete**—This will remove the file from the view. Confirm the deletion in the next dialog.  
You will also be given the option to delete any associated source files from the project.

## Managing View Groups

View Groups provide a way to organize the views when they are displayed in Xpress Insight.

Creating a View Group will create a new category in the **View Navigator** drop down selector in Xpress Insight.

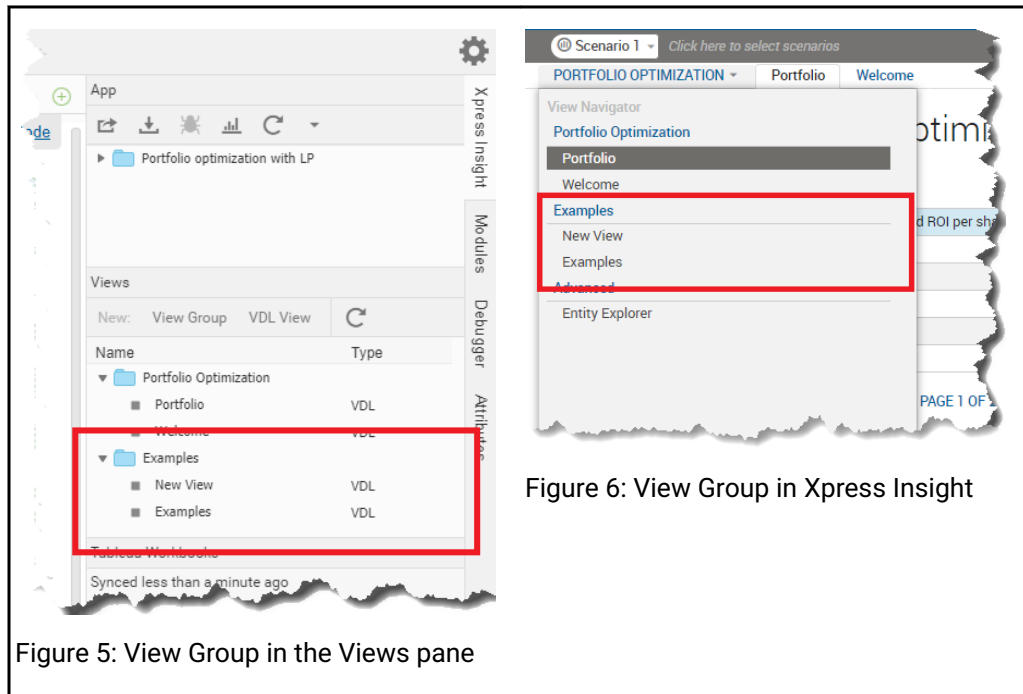


Figure 5: View Group in the Views pane

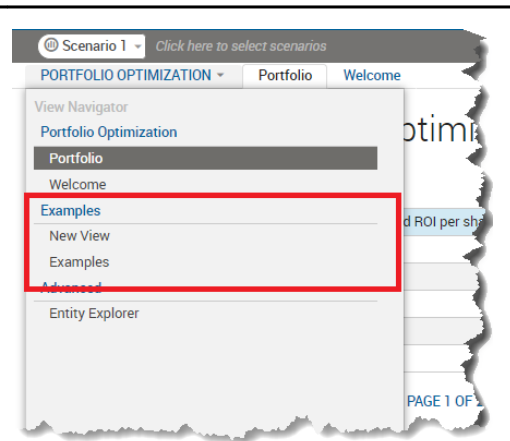


Figure 6: View Group in Xpress Insight

There are two ways to create a View Group. In the **Xpress Insight > Views** pane:

- 1 Click the **View Group** button.
- 2 Right click in the list of Views and select **New View Group**.

A new View Group is created, shown as a folder in the Views list.

## Deleting View Groups

To delete a View Group:

- In the Views pane, right click on the View Group and select **Delete**. The View Group and all Views in that group are deleted.

## Changing the View order

When views are published to Xpress Insight they are displayed across the browser in the order they are arranged in the Views pane.

To change the display order, open the Xpress Insight sidebar. In the **Views** pane, click and drag the views and folders displayed in the list to re-order them—The Views in the list are displayed from left to right in Xpress Insight.



**Note** View Groups cannot be nested.

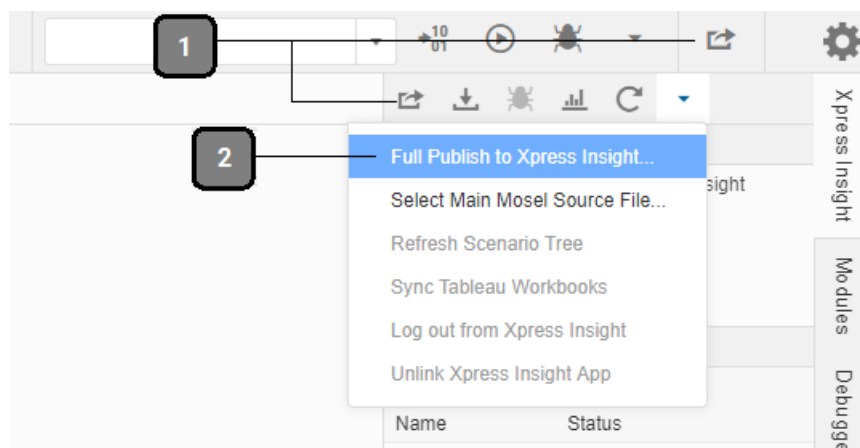
## Publishing to Xpress Insight

Creating or updating an app in Xpress Insight involves packaging the app files into a ZIP file and uploading the ZIP file using the Xpress Insight user interface. Xpress Workbench simplifies the process by performing these tasks automatically.

Before publishing to Xpress Insight, you must ensure that your project follows the structure of an Xpress Insight app:

- It must contain a Mosel source file that uses the **mminight** module.
- It should contain a companion file at the top level of the project, containing the configuration for an Xpress Insight app, including the custom views exposed by the app. For more, see [Editing the Companion File](#).
- Any custom view files should be placed in a **client\_resources** folder at the top level of the project.
- Any input files needed by the Mosel model should be placed in a **model\_resources** folder at the top level of the project.
- Any shared attachments needed by the app can be placed in a **attachments** folder at the top level of the project.

All Xpress Workbench features that relate to Xpress Insight are accessed from the **Xpress Insight** sidebar. You can find this sidebar on the right side of the Xpress Workbench **Project** page. There are two options when publishing to Xpress Insight, Full or Quick.



- 1 Quick Publish - Use this option to include only the files that have been changed since the last publish
- 2 Full Publish - Always use this option when publishing to Xpress Insight for the first time-additionally it is recommended you use this option when adding a new view, and as the final publish prior to release.

Complete the following steps to publish your project to Xpress Insight:

- 1 Open the **Xpress Insight** sidebar, if it is not already open, by clicking its name.
- 2 Use the required **Publish** option.
- 3 Use the **Xpress Insight Server** drop-down to select the server where you want to publish the app.
- 4 Select the Xpress Insight app that you want to update or select **Create a new app**.
- 5 Click **Publish**.
- 6 Check the **Output** window, which opens at the bottom of the screen, to see any errors or warnings.

✓ **Note** On the FICO® Analytic Cloud, the list of Xpress Insight app may take several seconds to load.

A warning is shown if there are files or folders in the project that are not permitted within an Xpress Insight app archive.

## Updating Xpress Insight Apps

Xpress Workbench will remember the Xpress Insight server and app that you last published to. To publish again and update the same app:

- 1 Open the **Xpress Insight** sidebar, if it is not already open, by clicking its name.
- 2 (Optional) If you want to change the main Mosel source file for the app, click **Settings**, which is the rightmost button in the **Xpress Insight** sidebar, and click **Select Main Mosel Source File**.

- 3 Publish an app by doing one of the following:
  - Click **Publish** button which is the first button from the left within the sidebar.
  - If you want to publish to a different Xpress Insight server or app, click **Settings**, which is the rightmost button in the **Xpress Insight** sidebar, and click **Publish to Xpress Insight**.
- 4 Check the **Output** window, which opens at the bottom of the screen, to see any errors or warnings.


## Opening Xpress Insight

Once a project is published to Xpress Insight, it is linked to the Xpress Insight app, and the app's folders and scenarios are displayed in the **Xpress Insight** sidebar.

Complete the following steps to open the app in Xpress Insight:

- 1 Open the **Xpress Insight** sidebar if it is not already open, by clicking its name.
- 2 In the scenario tree, right-click the Xpress Insight app name.
- 3 Click **Open in Xpress Insight** on the menu that appears.

A new browser tab is opened, showing the app within Xpress Insight.

 **Note** To refresh the list of folders and scenarios, click the **Refresh Scenario Tree** button in the rightmost position of the Xpress Insight toolbar.

## Exporting Xpress Insight Apps


Xpress Workbench can automatically publish a project to any Xpress Insight server within the same network environment.

To publish to an Xpress Insight server that is located elsewhere, you must create an app archive file that you can then upload using the Xpress Insight user interface. Complete the following steps to create an app archive file from your project:

- 1 Open the **Xpress Insight** sidebar, if it is not already open, by clicking its name.
- 2 Click the **Download** button, which is the second button from the left within the sidebar.
- 3 Check the **Output** window that opens at the bottom of the screen to see any errors or warnings.

A warning is shown if there are files or folders in the project that are not permitted within an Xpress Insight app archive.

If there are no errors, the app archive file is downloaded by your browser.

 **Note** The downloaded Xpress Insight app archive file only includes those files and folders that are allowed by Xpress Insight. To download all of the files and folders in the project, open the **File** menu and click **Download Project**. The archive downloaded using this method may not be a valid Xpress Insight app archive.

## Debugging Scenarios

Xpress Workbench can be used to debug scenarios while they are executing on an Xpress Insight server.

First you must have published the project to an Xpress Insight server. You can then start a debug session within Xpress Workbench. You can debug any type of scenario execution, whether the scenario is being loaded or run, or is executing in a custom run mode.

Complete the following steps to debug a scenario:

- 1 Open the **Xpress Insight** sidebar, if it is not already open, by clicking its name.
- 2 Expand the scenario tree and select the scenario you want to debug.
- 3 Click **Debug**, which is the third button from the left within the sidebar.
- 4 In the menu that appears, select the execution mode that you want to use for the debug session.

The app is published to Xpress Insight before the debug session starts.

- 5 You can now debug the model as usual. See [Debugging Models](#) on page 43 for more information.

Note that:

- If you set a breakpoint, the line with the breakpoint is highlighted to show that model execution is suspended at that line.
- Otherwise, the model runs to completion and suspends at the last line, so you can review the final state of the model.
- The **Debugger** sidebar opens, where you can see the call stack and the values of the variables defined by the model.

Once you have debugged a scenario, it will be added to the **Run Configurations** drop-down list in the main menu bar. You can quickly repeat the same debug session again by clicking the **Debug** button to the right of the **Run Configurations** drop-down list.



## Manually Executing Scenarios

Because there are so many ways to run a scenario, and they are often specific to the Xpress Insight app, Xpress Workbench can also start a debug session without automatically executing the scenario. After starting the debug session, you must open Xpress Insight and run the scenario manually.

Complete the following steps to debug a scenario manually:

- 1 Open the **Xpress Insight** sidebar, if it is not already open, by clicking its name.
- 2 Expand the scenario tree and select the scenario you want to debug.
- 3 Click the **Debug** button, which is the third button from the left within the sidebar.



- 4 In the menu that appears, select **Debug Scenario Manually**.  
The app is published to Xpress Insight before the debug session starts.
- 5 When you see the message `Click to open Xpress Insight`, click the link.  
If the message disappears before you click it, you can open Xpress Insight manually.  
Alternatively, you can right-click the scenario in the **Xpress Insight** sidebar and click **Open in Xpress Insight** in the menu that appears.  
A new browser tab opens showing the app in Xpress Insight, and the scenario appears on the shelf.  
 **Note** You may need to log in to Xpress Insight before the app is opened.
- 6 In Xpress Insight, execute the scenario. This could be done using the scenario menu or by using a button on a custom view.  
The scenario appears to be continually executing
- 7 Switch back to Xpress Workbench.
- 8 You can now debug the model as usual. See [Debugging Models](#) on page 43 for more information.  
 **Note** When a debug session has started and Xpress Workbench is waiting for the scenario to be executed, the debugger only attaches if the scenario is run by the same user who started the debug session. If another user logs into Xpress Insight and runs the scenario, the debugger does not attach.

## Cancelling Debug Sessions

A debug session can be cancelled, either before the scenario is executed or during execution.

If the session is cancelled before the scenario is executed, the scenario executes as normal when it is next run, and the debugger does not attach.

Complete the following steps to cancel a debug session:

- 1 Click **Stop** in the **Output** window.
- 2 Wait for the model to finish executing  
The debug session stops immediately. You cannot review the final state of the model before it exits.

## Unlinking Projects from Xpress Insight

Xpress Workbench will remember the Xpress Insight server and app that you last published to. When you do not wish to continue publishing the project to this app, you can unlink the project from Xpress Insight.

Complete the following steps to unlink the project from the Xpress Insight app:

- 1 Click the **Settings** button, which is the rightmost button in the **Xpress Insight** sidebar.

- 2 Click **Unlink Xpress Insight App**.

# Interacting with Tableau Server

---

This section describes how to use Xpress Workbench to create and edit Tableau workbooks. The following instructions assume that your project is already linked to an Xpress Insight app. See [Publishing to Xpress Insight](#) on page 61 for more information.

## Creating Workbooks

Tableau workbooks are created and edited using Tableau Server. After creating a workbook, it must be associated with the Xpress Insight app by adding an entry in the companion file.

Before creating a Tableau workbook, ensure that your companion file defines at least one mirror table and data source. The following example defines a mirror table containing a single entity, and a data source containing the mirror table:

```
<database-mirror table-prefix="example_">
  <mirror-tables>
    <mirror-table name="my_table">
      <entity name="my_entity"/>
    </mirror-table>
  </mirror-tables>
  <data-sources>
    <data-source name="my_data_source">
      <mirror-table-ref name="my_table"/>
    </data-source>
  </data-sources>
</database-mirror>
```



**Note** A Tableau workbook can only utilize a single data source. This data source may contain several mirror tables, but the tables may have overlapping index sets so that they can be joined together into a single SQL query. See the [Xpress Insight Developers Guide](#) for more information about how to configure mirror tables and data sources.

Complete the following steps to create a new Tableau workbook:

- 1 Open the companion file by double-clicking its filename.
- 2 Ensure that the companion file contains a valid Tableau data source.
- 3 Choose a name for the new workbook and add a new `tableau-workbook` entry to the companion file, specifying your chosen workbook name and setting the managed attribute to `true`.


See the [Xpress Insight Developers Guide](#) for more information about how to define custom views in the companion file.

- 4 Save the companion file and publish the project to Xpress Insight.
  - 5 Open the Xpress Insight sidebar by clicking its name.
  - 6 Click **Tableau**, which is the fourth button from the left in the sidebar.

The Tableau Server launch page opens in a new browser tab.  
You may be prompted to log into Xpress Insight before this page appears.  
If the mirror database is empty, you are prompted to mirror a scenario. This is to ensure that the workbook data source contains some data.
  - 7 (If prompted) Select the scenario to mirror, and click **Continue**.

Tableau Server is launched.
  - 8 Open the Default project in Tableau Server.
  - 9 Click the **Data Sources** tab.

A list of data sources appears. The data sources defined in the companion file is prefixed with the mirror table prefix for the app.
  - 10 Click the name of the data source that you want to use in the workbook. Only data sources that belong to the current app should be used.

A list of workbooks linked to the data source appears.
  - 11 Click **New Workbook** to the right of the datasource name.
  - 12 Create the content of the workbook.
  - 13 Click **Save As** in the page header.
  - 14 Enter the workbook name exactly as you entered it in the companion file.
  - 15 Click **Save**.
  - 16 The workbook is downloaded to Xpress Workbench.
-  **Important** New workbooks created in Tableau Server will not appear in Xpress Insight until you log out of Xpress Insight and log back in. Log out by clicking your username in the top right of the screen and then clicking **Log Out**.

## Editing Workbooks

Complete the following steps to edit an existing Tableau workbook:

- 1 Open the Xpress Insight sidebar by clicking its name.
- 2 Click the **Tableau** button, which is the fourth button from the left in the sidebar.

The **Tableau Server** launch page opens in a new browser tab.  
If the mirror database is empty, you are prompted to mirror a scenario. This is to ensure that the workbook data source contains some data.
- 3 (If prompted) Select the scenario to mirror, and click **Continue**.

Tableau Server is launched.
- 4 Open the Default project in Tableau Server.
- 5 Click the workbook you want to edit.

- 6 After your changes are saved, the workbook is downloaded to Xpress Workbench.



**Note** Tableau Server can only be used to edit workbooks whose data sources are defined externally, in the companion file. If the Xpress Insight app contains any workbooks with embedded data sources, the system will prevent Tableau Server from being launched.

## Downloading Workbooks into Xpress Workbench

Xpress Workbench automatically downloads workbooks that are created or modified in Tableau Server. This happens in the background.

- 1 After saving a workbook in Tableau Server, switch back to the Xpress Workbench tab.
- 2 Wait up to 60 seconds for Xpress Workbench to synchronize the workbooks. If there were warnings while synchronizing the workbooks, a message appears in Xpress Workbench.

## Viewing the Workbook Status

Complete the following steps to see the status of the workbooks that Xpress Workbench is synchronizing:

- 1 Open the Xpress Workbench sidebar by clicking its name.
- 2 Look in the **Tableau Workbooks** section, below the scenario tree.

The status of each workbook appears.



**Note** Xpress Workbench only synchronizes managed workbooks, which are configured in the companion file with the managed attribute set to true. Without this setting, the workbook will be ignored by Xpress Workbench.



## APPENDIX B

# Contacting FICO

---

FICO provides clients with support and services for all our products. Refer to the following sections for more information.

## Product Support

FICO offers technical support and services ranging from self-help tools to direct assistance with a FICO technical support engineer. Support is available to all clients who have purchased a FICO product and have an active support or maintenance contract. You can find support contact information and a link to the Customer Self Service Portal (online support) on the Product Support home page ([www.fico.com/en/product-support](http://www.fico.com/en/product-support)).

The FICO Customer Self Service Portal is a secure web portal that is available 24 hours a day, 7 days a week from the Product Support home page. The portal allows you to open, review, update, and close cases, as well as find solutions to common problems in the FICO Knowledge Base.

## Product Education

FICO Product Education is the principal provider of product training for our clients and partners. Product Education offers instructor-led classroom courses, web-based training, seminars, and training tools for both new user enablement and ongoing performance support. For additional information, visit the Product Education home page ([www.fico.com/en/product-training](http://www.fico.com/en/product-training)) or email [producteducation@fico.com](mailto:producteducation@fico.com).

## Product Documentation

FICO continually looks for new ways to improve and enhance the value of the products and services we provide. If you have comments or suggestions regarding how we can improve this documentation, let us know by sending your suggestions to [techpubs@fico.com](mailto:techpubs@fico.com).

Please include your contact information (name, company, email address, and optionally, your phone number) so we may reach you if we have questions.

## Sales and Maintenance

If you need information on other FICO® Xpress Optimization products, or you need to discuss maintenance contracts or other sales-related items, contact the Xpress Optimization team by:

- Phone: +44 (0)207 940 8718 or +1 (408) 535-1500
- Web: <http://www.fico.com/en/analytics/optimization> and use the available contact forms

## Related Services

**Strategy Consulting:** Included in your contract with FICO may be a specified amount of consulting time to assist you in using Xpress Workbench to meet your business needs. Additional consulting time can be arranged by contract.

**Conferences and Seminars:** FICO offers conferences and seminars on our products and services. For announcements concerning these events, go to [www.fico.com](http://www.fico.com) or contact your FICO account representative.

## FICO Community

The FICO Community is a great resource to find the experts and information you need to collaborate, support your business, and solve common business challenges. You can get informal technical support, build relationships with local and remote professionals, and improve your business practices. For additional information, visit the FICO Community ([community.fico.com/welcome](http://community.fico.com/welcome)).

## About FICO

FICO (NYSE:FICO) powers decisions that help people and businesses around the world prosper. Founded in 1956 and based in Silicon Valley, the company is a pioneer in the use of predictive analytics and data science to improve operational decisions. FICO holds more than 165 US and foreign patents on technologies that increase profitability, customer satisfaction, and growth for businesses in financial services, telecommunications, health care, retail, and many other industries. Using FICO solutions, businesses in more than 100 countries do everything from protecting 2.6 billion payment cards from fraud, to helping people get credit, to ensuring that millions of airplanes and rental cars are in the right place at the right time. Learn more at [www.fico.com](http://www.fico.com).



# Index

---

## A

- apps
  - updating 62
- array data
  - CSV files 47
- arrays 46
- autocomplete 36, 37

## B

- breakpoints 23, 44

## C

- cancelling debug sessions 65
- closing projects 14, 27
- code
  - folding with the keyboard 38
  - folding with the mouse 38
- code editing features 16
- code editor 16
- collaborating 38
- compiler
  - configuring options 49
- compiler options for Mosel 49
- compiling models 48
- configuring an xpress insight server 57
- coverage data 55
- Create a new view 59
- creating
  - files 30
  - projects 13, 25, 26
- Creating a View Group 60
- Creating Custom Views 17
- Creating projects 25

## D

- dashboard
  - displaying 28
- debug session 23, 24
- debug sessions
  - cancelling 65

- debugging models 41, 43
- debugging scenarios 64
- Default mos file 16
- Delete a view 59
- deleting projects 27
- Deleting View Groups 60
- displaying the dashboard 28
- downloading
  - projects 32
- downloading files 32
- downloading workbooks 69

## E

- editing source files 16
- evaluating expressions 47
- examining arrays 46
- expressions 47

## F

- FICO
  - about 72
  - community 72
  - conferences and seminars 72
  - consulting 72
  - documentation 71
  - education 71
  - maintenance 72
  - product support 71
  - professional services 72
  - sales 72
  - support 71
  - training 71
- file menu 29
- file tree 29
- files
  - creating from File menu 30
  - creating from file tree 30
  - creating from template 30
  - creating from View Designer 30
  - downloading 32

- opening 29
- renaming 31
- reverting to previous version 32

finding text 33, 35  
folding code 38

## G

getting started 9, 11

## H

highlighting syntax 36

## I

IDE 36  
immediate window 48  
index term 8, 10, 11, 13  
Inline Documentation 37  
inline help 37  
Insight tab 20  
Installing Xpress Workbench 12  
introduction 7

## L

Launching Xpress Workbench 12  
logging in 11  
logging out 12

## M

manually executing scenarios 64  
messages

- sending 38

model execution 56  
models

- compiling 48
- debugging 41, 43
- resuming 45
- running 41
- running models 41
- stepping through 45
- stopping 42
- suspending 45
- tuning 51

Modules panel 33  
mosel

- profiling 55

Mosel

- compiler options 49
- profiling models 55

mosel deployment wrappers 51

mosel projects 13, 26

## N

New View wizard 20

## O

opening files 29  
opening projects 14, 27

## P

Palette 14  
plus icon 59  
previous version 32  
project backup 28  
Project pane 14  
project restore 28  
projects 13, 25–27, 32

- creating 13, 25, 26
- deleting 27
- downloading 32
- opening 27
- renaming 27
- working with 25

publishing to

- Xpress Insight 62

publishing to Xpress Insight 22, 61

## R

renaming files 31  
renaming projects 27  
replacing text 34, 35  
resuming models 45  
reverting files 32  
running models 41

## S

sending message 38  
source files 16  
starting a debug session 43  
statement 36  
stepping through models 45  
stopping models 42  
submodels

- parallel solving 50
- running 50

suspending models 45  
syntax

- highlighting 36

**T**

- Tableau Server
  - interacting with 67
- template files 51
- text
  - finding 33, 35
  - replacing 34, 35
- trace mode 56
- tuner settings 53, 54
- tuning models 51
- tuning mosel models 52

**U**

- unlinking projects 65
- uploading files 31
- Using a template 29

**V**

- variables
  - examining 46
- VDL
  - custom views 16
  - orientation 16
- VDL View 57
- View Designer 14, 16, 18
  - artboard 18
  - Attributes Pane 18
  - custom views 16
  - orientation 16
    - Palette
- COMPONENTS 18

- HTML 18

- LAYOUT 18

- LOGIC 18

- View group 57

- View Groups 61

- Views pane 57, 60, 61

**W**

- words
  - automatically completing 36
- Workbench IDE 14
- workbook
  - viewing status 69
- workbooks
  - creating 67
  - downloading 69
  - editing 68
- working directory 43
- working with projects 25

**X**

- Xpress Insgiht
  - publishing to 61
- Xpress Insight 22, 57, 62, 63
  - exporting apps 63
  - opening 63
  - publishing to 22
  - updating app 62
- xpress insight projects 13, 26
- Xpress Workbench
  - description 7

