

# RELEASE NOTES

# NI TestStand™

## Version 2012

These release notes contain NI TestStand 2012 system requirements, installation instructions, information about new features, and other changes since TestStand 2010 SP1.

## Contents

---

Getting Started .....	1
Recommended System Requirements .....	2
Installation Instructions .....	4
Activating TestStand Licenses .....	5
Recommended Database Client Software .....	6
Installing Additional Software Components .....	7
Installing Multiple Versions of TestStand on the Same Computer .....	7
TestStand and Windows 7/Vista.....	8
Using the TestStand Version Selector .....	9
Directory Relocation.....	9
Migrating to TestStand 2012 .....	9
Migrating to TestStand 2012 and Windows 7/Vista .....	9
Migrating from TestStand 2010 SP1 or TestStand 2010 Using the TestStand Migration Utility.....	10
Manually Migrating from Any Version of TestStand .....	12
Upgrading TestStand 2012 Beta to TestStand 2012.....	16
Behavior Changes.....	16
LabVIEW Support Behavior Changes .....	19
Process Model Incompatibilities .....	21
What's New in TestStand 2012 .....	23
TestStand API and TestStand UI Controls API Additions.....	24
Submitting Feedback on TestStand Documentation .....	24

## Getting Started

---

The best way to familiarize yourself with TestStand is to explore the *Guide to TestStand Documentation* topic in the *NI TestStand Help*, which contains links to all the TestStand documentation in electronic format. Select **Start»All Programs»National Instruments»TestStand»Documentation»NI TestStand Help** or select **Help»NI TestStand Help** in the TestStand Sequence Editor to access the *NI TestStand Help*.

# Recommended System Requirements

National Instruments recommends the system requirements described in Table 1 to run TestStand 2012.

**Table 1.** Recommended and Minimum System Requirements

Requirement	Recommended	Minimum
Processor	Pentium 4/M or equivalent	Pentium III, Celeron 866 MHz, or equivalent
Memory	1 GB	256 MB
Disk space for TestStand	1 GB	—
Disk space for Microsoft .NET Framework Version 4.0, if not already installed	850 MB (x86) or 2 GB (x64)	—
Disk space for device drivers from National Instruments Device Drivers DVD	Amount of space required varies depending on the drivers you install. Install only the drivers for the hardware you will use.	—
Video adapter	—	1024 × 768 resolution

Use the Microsoft Windows operating systems and service packs described in Table 2 with TestStand 2012.

**Table 2.** Recommended and Required Windows Operating System Service Packs

Operating System	Recommended Service Pack	Required Service Pack	Comments
Windows 7 (32- and 64-bit), including Starter Edition	Service Pack 1	—	Refer to the <a href="#">TestStand and Windows 7/Vista</a> section of this document for more information about using TestStand on Windows 7/Vista.
Windows Vista (32- and 64-bit) Business, Enterprise, or Ultimate	Service Pack 2	Service Pack 1	
Windows XP	—	Service Pack 3	—
Windows Server 2008 R2 (64-bit)	—	Service Pack 2	TestStand supports only R2 editions of Windows Server.
Windows Server 2003 R2 (32-bit)	—	Service Pack 2	

TestStand does not support Guest user accounts on Windows. TestStand supports Administrator and Limited accounts on Windows 7/Vista and Administrator, Power Users, and Users accounts on Windows XP.

Use the following Microsoft software with TestStand 2012:

- Microsoft Internet Explorer version 7.0 (version 5.5 minimum)
- Microsoft Visual Studio 2010, 2008, or 2005
- Microsoft .NET Framework 4.0

Refer to the *.NET Support Enhancements* section of the *TestStand 2012 New Features* topic in the *NI TestStand Help* for more information about .NET support in TestStand 2012.

TestStand includes features that automate Visual Studio for debugging and generating and editing code module source code. Visual Studio Express Editions do not support these features.

The .NET and Microsoft Foundation Class (MFC) examples include projects and solutions created in Visual Studio 2005.

Use the National Instruments application development environments (ADEs) listed in Table 3 with TestStand 2012.

**Table 3.** Supported ADE Versions

<b>ADE</b>	<b>Supported</b>	<b>Allowed but Not Supported</b>	<b>Comments</b>
LabVIEW	2012, 2011 SP1, 2010 SP1, or 2009 SP1	8.6.1, 8.5.1, or 8.2.1  TestStand 2012 does not support calling LabVIEW 8.0.1 or earlier VIs.	LabVIEW examples and user interfaces use files created with LabVIEW 2009 SP1. TestStand API support files for LabVIEW use files saved with LabVIEW 8.2.1.  Refer to the National Instruments website at <a href="http://ni.com/info">ni.com/info</a> and enter the Info Code <code>ts1v64</code> to for information about LabVIEW 64-bit compatibility with TestStand.
LabWindows/CVI	2012, 2010 SP1, 2009 SP1, or 9.0.1	2010 and 2009 or earlier	LabWindows/CVI examples, user interfaces, and API use files created with LabWindows/CVI 9.0.1.
Measurement Studio	2009 or 8.6.1	—	Measurement Studio 2010 or later no longer includes C++ MFC classes the TestStand C/C++ DLL Adapter previously supported.

TestStand can execute code modules developed with versions of ADEs other than the listed supported versions, but National Instruments performs only limited testing with earlier versions of ADEs. TestStand might be able to execute code modules developed with versions of ADEs later than the listed supported versions, but National Instruments cannot ensure support for versions released after TestStand. Refer to the National Instruments website at [ni.com/info](http://ni.com/info) and enter the Info Code `tslcp` for more information about the TestStand life cycle policy.

## Installation Instructions

Unless you specify another location during installation, the installation program copies core files to <Program Files>\National Instruments\TestStand 2012.



**Note** You cannot install TestStand to a network path or a mapped network path. You must install TestStand on a local computer.

National Instruments recommends exiting all programs before running the installer. Applications that run in the background, such as virus scanning utilities, might cause the installer to take longer than average to complete.

Complete the following steps to install TestStand.

1. Log on as an administrator or as a user with administrator privileges.
2. Insert the TestStand 2012 installation media and follow the instructions that appear on the screen.

National Instruments recommends installing the complete TestStand program. If you perform a custom installation and do not install all the TestStand features, you can run the installation program again later to install additional features.



**Note** If you do not install some TestStand features and then you later use the standard Windows Control Panel facility for adding and removing programs to install the features, the features might not correctly register and TestStand displays a COM error the next time you launch TestStand. This error occurs even if you restart the computer after you use the Windows Control Panel to install the features. National Instruments recommends that you reactivate TestStand using the TestStand Version Selector after you install the features.

If you install LabVIEW, LabWindows/CVI, or Visual Studio after you install TestStand, launch the sequence editor or a TestStand User interface once so TestStand configures these applications to locate TestStand API files. When you install a newer version of LabVIEW after you install TestStand, run the TestStand Version Selector and select the most recent version of TestStand to ensure the LabVIEW User Interface VIs function properly. Select **Start»All Programs»National Instruments»TestStand»TestStand Version Selector** to launch the TestStand Version Selector application, `TSVerSelect.exe`, located in the <Program Files>\National Instruments\Shared\TestStand Version Selector directory.

Refer to the National Instruments website at [ni.com/info](http://ni.com/info) and enter the Info Code `rddrau` to access the latest software drivers and updates.

# Activating TestStand Licenses

After you install TestStand, you must use the NI Activation Wizard to activate the software or initiate the evaluation period for the software. You must have the serial number and the name of the software kit to activate TestStand. You can find both of these items on the Certificate of Ownership card included in your software kit. Refer to the *Activating Your Software* topic in the *NI TestStand Help* for more information about how to activate TestStand.

National Instruments offers a variety of licenses for the different ways you can use TestStand in development and deployment applications. You can select from the following types of licenses: the TestStand Development System License, the TestStand Custom Sequence Editor License, the TestStand Debug Deployment Environment License, and the TestStand Base Deployment Engine License.

In most cases, when you first install TestStand, you activate a TestStand Development System License. Use the other licenses to activate TestStand on computers to which you deploy TestStand custom sequence editors or user interfaces you develop. Refer to the *Licensing Options for TestStand Systems* section of Chapter 1, *Introduction to TestStand*, of the *Getting Started with TestStand* manual for information about how to use the available licenses when you build a TestStand-based test solution.

Use the following descriptions only as a reference for the licensing options. Refer to [ni.com/activate](http://ni.com/activate) for more information about activating TestStand licenses. Refer to [ni.com/teststand](http://ni.com/teststand) to purchase a TestStand license. Contact a local National Instruments representative for more information or for questions about specific licensing needs.



**Note** This document does not replace the National Instruments Software License Agreement installed in the <Program Files>\National Instruments\Shared\MDF\Legal\license\NIReleased directory.

## TestStand Evaluation Package

When you run TestStand in Evaluation Mode, the software expires after 7 days. You can use an [ni.com](http://ni.com) User Profile to extend the evaluation period for an additional 45 days.

The software runs as a fully functional Development System during the evaluation period. You can activate a license at any point during or after the evaluation period.

## TestStand Development System License (777777-09)

Activate the TestStand Development System License to develop and edit test sequences and to save sequence files within the TestStand Sequence Editor, within a TestStand custom sequence editor, or programmatically using the TestStand API. This license also grants the ability to develop custom sequence editors and operator interfaces. You must have an [ni.com](http://ni.com) User Profile to activate a TestStand Development System license.

## TestStand Custom Sequence Editor License (777775-01)

Activate this license to develop and edit test sequences and sequence files within a TestStand custom sequence editor or programmatically using the TestStand API.

## TestStand Debug Deployment Environment License (779851-09)

The TestStand Debug Deployment Environment License offers the most flexibility for deploying TestStand and LabVIEW-based, LabWindows/CVI-based, and Measurement Studio-based systems.

Activate this license to install the development versions of TestStand, LabVIEW, LabWindows/CVI, Measurement Studio, and any corresponding add-on toolkits on a single test station so you can debug deployed test applications on the test station. This license grants the ability to make minor edits to fix bugs in deployed test applications but does not grant the ability to perform any development tasks using TestStand, LabVIEW, LabWindows/CVI, or Measurement Studio on the test station.

You cannot activate and deactivate the TestStand Debug Deployment Environment License and reuse it on multiple computers. If you need to use a single debug license across multiple computers, contact National Instruments for more information about the Concurrent TestStand Debug Deployment Environment License.

## TestStand Base Deployment Engine License (777774-03)

The TestStand Base Deployment Engine License is the minimum license required for all deployed TestStand-based applications. Activate this license to deploy the TestStand Engine, a TestStand Operator Interface, and sequence files to the single test station to which the license applies. This license does not grant the ability to perform any development tasks using the TestStand Sequence Editor, a TestStand custom sequence editor, or the TestStand API.

## Recommended Database Client Software

Use the following recommended database client software with the database components included with TestStand:

- **Microsoft Access**—Use the Microsoft Jet 4.0 Object Linking and Embedding Database (OLE DB) Provider.
- **Microsoft SQL Server**—Use the SQL Server Native Client or the Microsoft OLE DB Provider for SQL Server provider.
- **Oracle**—Use the latest Oracle Provider for OLE DB and Oracle Client software. You can download the Oracle Provider from the Oracle Web site at [www.oracle.com](http://www.oracle.com).



**Note** National Instruments does not recommend using the Microsoft OLE DB Provider for Oracle because it does not support all the OLE DB features TestStand requires.



**Note** Oracle recommends using Oracle Provider 11.1.0.6.0 or later if you want to read 64-bit integer values stored in NUMBER columns.

- **MySQL**—Use the MySQL Open Database Connectivity (ODBC) Driver 3.51 or later.
- **Sybase SQL Anywhere**—Use the Adaptive Server Anywhere ODBC Driver 9.0 or later.

# Installing Additional Software Components

TestStand includes the device driver software on a DVD. If you require device driver software on CDs, refer to the National Instruments website at [ni.com/info](http://ni.com/info) and enter the Info Code `drivercd`.

TestStand 2012 installs the following additional software components:

- .NET Framework 4.0
- LabVIEW 2011 SP1 and 2009 SP1 RTE
- LabWindows/CVI 2012 RTE
- LabWindows/CVI SQL Toolkit DLL version 2.2, `cvidb32.dll`
- National Instruments Session Manager
- NI Update Service
- NI License Manager



**Note** TestStand can use newer versions of the LabVIEW RTE when you install LabVIEW on a development system. You can include newer versions of the LabVIEW RTE in deployments using the Drivers and Components dialog box of the TestStand Deployment Utility. Refer to the *NI TestStand Help* for more information about the Drivers and Components dialog box.

The National Instruments Device Driver DVD contains the following suggested components:

- National Instruments Measurement & Automation Explorer (MAX)
- Interchangeable Virtual Instrument (IVI) Compliance Package (ICP)

To use IVI, download and install IVI Compliant-specific drivers from the Instrument Driver Network, located at [ni.com/devzone/idnet](http://ni.com/devzone/idnet).

## Installing Multiple Versions of TestStand on the Same Computer

You can install TestStand 2012 on a computer that contains a previous TestStand version, but you cannot install TestStand 2012 over a previous TestStand version. You cannot install TestStand 2012 in the default installation directory for a previous version of TestStand. For example, TestStand 2012 does not install in `<Program Files>\National Instruments\TestStand 2010 SP1`. If you installed a previous version of TestStand in a non-default directory, you can uninstall the previous version of TestStand and install TestStand 2012 to that directory.

Refer to the [Migrating to TestStand 2012](#) section of this document for information about using the TestStand 2012 Migration Utility to migrate customizations and configuration settings from TestStand 2010 SP1 or 2010 to TestStand 2012.

## Uninstalling TestStand 2.0.1 or Earlier

Because the uninstallers for TestStand 2.0.1 or earlier remove the <TestStand>\OperatorInterfaces\User directory, you must complete the following steps to safely uninstall the previous TestStand version and preserve all configuration files and files located in the User subdirectories.

1. Move the <TestStand>\OperatorInterfaces\User directory to a location outside the <TestStand> directory.
2. Navigate to the standard Windows Control Panel facility for adding and removing programs and run the TestStand uninstaller. If the uninstaller launches a dialog box that requests confirmation to remove all TestStand configuration files and pre-installed user components, click **No**.
3. When the uninstaller completes, move the OperatorInterfaces\User directory back into the original <TestStand> directory.

You can now install TestStand 2012 into this directory.

## TestStand and Windows 7/Vista

TestStand 2012 runs on Windows 7/Vista.

### Windows Vista or Later Behavior Change

Windows Vista introduced a change in behavior that can cause applications to receive WM\_PAINT messages unexpectedly while performing ActiveX/COM calls from .NET components or .NET applications. National Instruments has tested TestStand 2012 and fixed instances where this behavior change caused the sequence editor and TestStand User Interface (UI) Controls to behave incorrectly. Refer to the National Instruments website at [ni.com/info](http://ni.com/info) and enter the Info Code `rdwttc` for more information about this Windows Vista behavior change and how it might affect applications that you write or modify.

### User Account Control Elevation Prompts

The User Account Control (UAC) security component in Windows 7/Vista requires administrator privileges for some tasks, such as installing software, running certain applications, and changing system settings. If you are logged in as a user with standard privileges, Windows 7/Vista launches a UAC elevation prompt for prohibited tasks. You cannot resolve the UAC elevation prompts programmatically. Refer to Microsoft documentation for more information about UAC prompts.

TestStand 2012 uses a Windows service to automatically handle most TestStand-related UAC prompts and notifications. The NI TestStand Service runs with administrator privileges in the background and is responsible for tasks that require administrator privileges. The TestStand Service does not automatically handle applying settings for remote execution. Performing these actions while logged in to Windows 7/Vista as a user with standard privileges results in a UAC elevation prompt.



## Visual Studio Integration

When you launch Visual Studio from TestStand, Visual Studio runs with the same privileges you used to run TestStand. If you launch TestStand while logged in as a user with standard privileges and you launch Visual Studio from TestStand, you cannot execute tasks in Visual Studio that require administrator privileges.

## Using the TestStand Version Selector

Although you can install TestStand 2012 on a computer that contains a previous TestStand version, only one version of TestStand can be active at a time. If you must install TestStand 2012 on the same computer as a previous version, use the TestStand Version Selector to specify the active version. Select **Start»All Programs»National Instruments»TestStand»TestStand Version Selector** to launch the TestStand Version Selector application, `TSVerSelect.exe`, located in the `<Program Files>\National Instruments\Shared\TestStand Version Selector` directory.

If you activate TestStand 2012 and run a TestStand User Interface from the previous TestStand version, the user interface uses the TestStand 2012 Engine, step types, and components. If you activate the previous TestStand version and run a TestStand 2012 User Interface or the sequence editor, those applications do not function correctly.

## Directory Relocation

To comply with Windows 7/Vista restrictions on writing to the `Program Files` directory and to improve usability for Windows XP users who do not have permission to write to the `Program Files` directory, TestStand 4.1 or later installs some files in different locations from previous versions of TestStand. Refer to the *Directory Relocation* section of the *NI TestStand Version 4.2 Release Notes* for more information about directory location changes. Refer to the National Instruments website at [ni.com/info](http://ni.com/info) and enter the Info Code `ex3ukj` to locate the *NI TestStand Version 4.2 Release Notes*.

## Migrating to TestStand 2012

---

If you are upgrading from TestStand 2010 SP1 or TestStand 2010, you can use the TestStand 2012 Migration Utility to migrate customizations and configuration settings from the earlier version of TestStand. You can also upgrade manually. Refer to the [Migrating from TestStand 2010 SP1 or TestStand 2010 Using the TestStand Migration Utility](#) section of this document for information about using the utility.

If you are upgrading from TestStand 4.2.1 or earlier, you must manually migrate customizations and configuration settings to TestStand 2012. Refer to the [Manually Migrating from Any Version of TestStand](#) section of this document for information about the tasks you need to perform to upgrade manually.

## Migrating to TestStand 2012 and Windows 7/Vista

To comply with Windows 7 and Windows Vista restrictions on writing to the `<Program Files>` directory and to improve usability for Windows XP users who do not have

permission to write to the <Program Files> directory, TestStand 4.1 or later installs some files in different locations from previous versions of TestStand. Refer to the [Directory Relocation](#) section of this document for more information about directory location changes.

If you are migrating from a version of TestStand earlier than TestStand 4.1, refer to the [Manually Migrating from Versions Earlier than TestStand 4.1](#) section of this document for tasks you must complete manually.

## Migrating from TestStand 2010 SP1 or TestStand 2010 Using the TestStand Migration Utility

Use the TestStand 2012 Migration Utility to copy settings, configuration, and public component files from the directories of TestStand 2010 SP1 or TestStand 2010 to TestStand 2012. Select **Start»All Programs»National Instruments»TestStand»Tools»Compatibility»TestStand Migration Utility** to launch the utility.



**Note** When you migrate from TestStand 2010 to TestStand 2012, the utility migrates the report and database options to the TestStand 2012 format for result processing configuration options. Refer to the [Migrating TestStand 2010 SP1 or TestStand 2010 Result Processing Options to TestStand 2012](#) section of this document for more information about using the utility to migrate result processing configuration options.

If you have more than one version of TestStand installed on the computer, the utility launches a dialog box in which you can specify the previous version of TestStand from which you want to migrate. After you specify the previous version of TestStand, the utility checks all the migrateable files on the computer to determine whether the files have changed since installation. The utility displays a status indicator dialog box during this process. When the utility finishes checking the files for changes, the utility main window launches.

Use the **Files to Copy** control to select the files you want to copy. By default, the utility selects files you added or modified in the previous version of TestStand. Expand the directories in the Files to Copy control to view the migrateable files and whether those files have changed since installation. The Conflicts column includes any conflicts that could affect the migration of the files you specify. Click **Migrate Files** to begin the migration.



**Note** The utility only moves files and does not update any other information or settings, such as registry keys.

When the migration completes successfully, the utility generates a text file report that contains the results of the migration, including where the utility stored the file backups and which files were moved.



**Note** The utility overwrites any file you create or modify in TestStand 2012 if the file also exists in the corresponding directory of the previous version of TestStand and if you select the file in the **Files to Copy** control. Although the utility creates a backup copy of all overwritten files, verify that the utility will not overwrite any TestStand 2012 files you intend to keep.



**Note** If the <TestStand Public>\Components directory of the previous version of TestStand includes files installed by an installer other than the TestStand installer, you might need to use that specific installer to reinstall the files to the <TestStand Public>\Components directory of TestStand 2012. National Instruments recommends that you use an installer that targets TestStand 2012 to install any such components instead of using the utility to copy the components. However, some installed components can function properly when copied to another directory and are safe to migrate with the utility.

## Migrating TestStand 2010 SP1 or TestStand 2010 Result Processing Options to TestStand 2012

The TestStand 2012 process models include significant changes to implement a plug-in architecture for model sequence files. TestStand 2012 uses the plug-in mechanism to decouple result processing from the process models. As part of this change, the result processing options move from the `TestStandModelReportOptions.ini` and `TestStandDatabaseOptions.ini` files into a single `ResultProcessing.cfg` file, located in the <TestStand Application Data>\Cfg\ModelPlugins directory, that stores configuration information for all result processing model plug-ins.

The result processing options files from previous versions of TestStand are not compatible with the TestStand 2012 process models. Refer to the *Process Model Result Processing Plug-in Architecture* section of the *TestStand 2012 New Features* topic in the *NI TestStand Help* and *Process Model Incompatibilities* sections of this document for more information about the TestStand 2012 process models. Refer to the *Using the Legacy Model Switcher Tool* section of the *TestStand 2012 New Features* topic in the *NI TestStand Help* for more information about using the Legacy Model Switcher tool to use the equivalent legacy TestStand 2010 process models.

Use the following migration actions in the utility to migrate the result processing configuration options from TestStand 2010 SP1 or TestStand 2010 to TestStand 2012.



**Note** The utility does not support migrating the configuration files from a previous version of TestStand if you modified the ReportOptions or DatabaseOptions data types defined in the default process models.

- **Apply to new models**—Creates a new configuration in a configuration set named **Migrated from TestStand Version**. The utility automatically selects a unique name for the configuration if the utility has already used the default name in a previously completed migration. Refer to the migration report for the specific configuration names the utility uses.
- **Apply to legacy models**—Copies the options files from a previous version of TestStand to the current TestStand directory so you can use the previous configurations with the equivalent legacy TestStand 2010 process models in TestStand 2012. When you migrate database options, the utility renames `TestStandDatabaseOptions.ini` to `TestStandDatabaseSchemas.ini`.
- **Apply to new and legacy models**—Adds a new configuration in the **Migrated from TestStand Version** configuration set and copies the options files from a previous version of TestStand.

If you upgrading from TestStand 4.2.1 or earlier, you can manually migrate result processing options for use with the TestStand 2012 process models or the equivalent legacy TestStand 2010 process models. Refer to the [Manually Migrating Result Processing Options from Any Version of TestStand](#) section of this document for more information about manually migrating result processing options to TestStand 2012.

## Manually Migrating from Any Version of TestStand

If you are upgrading from TestStand 4.2.1 or earlier, you must manually migrate customizations and configuration settings to TestStand 2012, as Table 4 describes.

If you are upgrading from TestStand 2010 SP1 or TestStand 2010, you can manually migrate customizations and configuration settings to TestStand 2012 or you can use the TestStand 2012 Migration Utility to migrate to TestStand 2012. Refer to the [Migrating from TestStand 2010 SP1 or TestStand 2010 Using the TestStand Migration Utility](#) section of this document for more information about using the utility.

**Table 4.** Manually Migrating from Any Version of TestStand

Version Being Upgraded	Items to Migrate	For More Information
Earlier than TestStand 4.1 (manual migration required)	All files or directories affected by directory location changes in TestStand 4.1 or later	<i>Manually Migrating from Versions Earlier than TestStand 4.1</i>
TestStand 2010 SP1 or TestStand 2010 (manual migration optional)	Configuration settings and user components	<i>Manually Migrating User Components from Any Version of TestStand</i>
TestStand 4.2.1 or earlier (manual migration required)	Custom components	<i>Manually Migrating TestStand Changes to Custom Components from Any Version of TestStand</i>
	Result processing options	<i>Manually Migrating Result Processing Options from Any Version of TestStand</i>

### Manually Migrating from Versions Earlier than TestStand 4.1

If you are migrating from a version of TestStand earlier than TestStand 4.1, you must account for the new directory locations in TestStand 4.1 or later. Refer to the [Directory Relocation](#) section of this document for more information about directory location changes.

Complete the following tasks to account for the new directory locations when you migrate to TestStand 2012:

- If you saved files in a relocated directory, copy the files to the new location in TestStand 4.1 or later.

- Update projects and files that reference files and directories that have moved in TestStand 4.1 or later. You can use the `Engine.GetTestStandPath` method to obtain file and directory paths programmatically.

If you use relative paths to reference files in the <TestStand> directory, the paths might break.

Some ADEs might return warnings when you update projects and files, as Table 5 describes.

**Table 5.** ADE Warnings when You Update Projects and Files for Use with TestStand 4.1 or later

ADE	Warning Returned	Resolution
LabWindows/CVI	Some files were not found when you open a project.	Remove the files from the project and re-add them from the <National Instruments>\Shared\CVI\instr\TestStand\API directory in LabWindows/CVI 8.5 or later and from the <CVI>\instr\TestStand\API directory in LabWindows/CVI 8.1.1 or earlier.
Visual Studio	Some TestStand API or adapter support files were not found when you open a project.	Add the \$(TestStand), \$(TestStandAppData), and \$(TestStandPublic) environment variables as needed to the Additional Include Directories control for the project

- Use the TestStand Deployment Utility to update deployable images or installers for deployable images to redirect file destinations to the TestStand 4.1 or later locations. Select **View Destination** in the Distributed Files control and select the new location in the Installation Destination control in the Installer Properties section on the Distributed Files tab.

## Manually Migrating User Components from Any Version of TestStand

You can copy the following directories and files from the previous TestStand installation to the appropriate TestStand 2012 directory to migrate configuration settings and user components. If you are migrating from a version earlier than TestStand 4.1, refer to the [Migrating to TestStand 2012 and Windows 7/Vista](#) section of this document for more information about directory location changes.

- <TestStand Application Data>\Cfg\StationGlobals.ini
- <TestStand Application Data>\Cfg\Templates.ini
- <TestStand Application Data>\Cfg\TestExec.ini
- <TestStand Application Data>\Cfg\TestStandModelModelOptions.ini

- <TestStand Application Data>\Cfg\TestStandPersistedOptions.opt
- <TestStand Application Data>\Cfg\Users.ini
- <TestStand Public>\CodeTemplates
- <TestStand Public>\Components
- <TestStand Public>\UserInterfaces

If you use custom Tools menu items in the previous version of TestStand, complete the following steps to export the items from that installation and import them into TestStand 2012.

1. In the previous version, select **Tools»Customize** in the TestStand Sequence Editor to launch the Customize Tools Menu dialog box.
2. Click the **Export Items to File** button to launch the Export Tools Menu dialog box.
3. Select the menu items to export to a Tools menu file and click **OK**.
4. Create a Setup\ToolMenusToInstall directory in the <TestStand Public> directory of TestStand 2012.
5. Place the Tools menu file you created in step 3 in the new directory.
6. Launch the TestStand 2012 sequence editor. TestStand adds the new menu items to the Tools menu and deletes the Tools menu file.

## Manually Migrating TestStand Changes to Custom Components from Any Version of TestStand

TestStand includes several components you can customize, such as process models, user interfaces, and certain step types. If you made substantial or complex changes to a component, use a file-comparison tool to determine the changes between the TestStand 2012 version of the component and the original version of the component you modified and apply the TestStand 2012 improvements to the custom version of the component.

If you made minor changes to the component, use a file-comparison tool to determine the changes you made to the component and reapply the improvements to a copy of the TestStand 2012 version of the component.



**Note** Subsets of different versions of the same component are not necessarily interoperable without modifications. For example, you cannot replace a single sequence in the TestStand 2012 default process models with the corresponding sequence from older process models without making further modifications. If you customized the process models, you must ensure that TestStand can find all the subordinate components the process models use and that any of those components that are ActiveX servers are registered. The default process model sequence for TestStand uses separate sequences, DLLs, and ActiveX servers to support database logging and report generation features.

You can use the following types of file-comparison tools:

- To compare sequence files, select **Start»All Programs»National Instruments»TestStand»Tools»File Diff and Merge** to launch the Select Files dialog box of the TestStand File Diff/Merge application.

- To compare text files, use a source code comparison tool, such as Microsoft WinDiff. You can also use the Diff command in the Edit menu of the LabWindows/CVI Source window.
- To compare VI files, select **Tools»Compare** in LabVIEW to use the Compare VIs tool. This tool is available only in the LabVIEW Professional Development System.

## Manually Migrating Result Processing Options from Any Version of TestStand

Complete the following steps to migrate result processing options for use with the TestStand 2012 process models.

1. Copy the `TestStandModelReportOptions.ini` and `TestStandDatabaseOptions.ini` files from the `Cfg` directory of the previous version of TestStand to the `<TestStand Application Data>\Cfg` directory of TestStand 2012.
2. Rename `TestStandDatabaseOptions.ini` to `TestStandDatabaseSchemas.ini`.



**Note** This procedure is not supported for migrating TestStand 4.0 or earlier default database schemas. If you want to use TestStand 4.0 or earlier default database schemas in TestStand 2012, click the **Reload NI Schemas** button on the Schemas tab of the Database Options dialog box to launch the Reload NI Schemas dialog box, and click the **2.x to 4.0** button to reload the default database schemas from TestStand 4.0 or earlier.

3. Use the Legacy Model Switcher tool with the `/migration` command-line argument to migrate the report and database options to the TestStand 2012 process models. Confirm that you have enabled the options to migrate report and database options from the legacy models. Refer to the *Using the Legacy Model Switcher Tool* section of the *TestStand 2012 New Features* topic in the *NI TestStand Help* for more information about using the Legacy Model Switcher tool.



**Note** The `/migration` command-line argument disables the type version checking the Legacy Model Switcher tool ordinarily performs and automatically merges the types in the report and database options in the equivalent legacy TestStand 2010 process models with the types in the TestStand 2012 process models.

4. Launch the TestStand 2012 sequence editor.
5. Select **Configure»Result Processing** to launch the Result Processing dialog box.
6. Enable the **Show More Options** option.
7. Verify that the Active Configuration ring control specifies Options from Legacy Models.



**Note** To migrate result processing options to the equivalent legacy TestStand 2010 process models, complete only step 1. However, National Instruments does not recommend using the equivalent legacy TestStand 2010 process models unless you encounter an incompatibility in an existing test system when you use the TestStand 2012 process models or unless other circumstances prevent you from using the TestStand 2012 process models.

Refer to the *Result Processing Improvements* section of the *TestStand 2012 New Features* topic in the *NI TestStand Help* for more information about the Result Processing dialog box and other changes related to result processing.

## Upgrading TestStand 2012 Beta to TestStand 2012

---

You must uninstall TestStand 2012 Beta before you install TestStand 2012. When you uninstall TestStand 2012 Beta, the user interface files remain in the <TestStand Public>\UserInterfaces directory. Installing TestStand 2012 does not overwrite these files. Because the C# and Microsoft Visual Basic .NET user interface executables are strongly bound to the specific assemblies that TestStand 2012 Beta uninstalls, you cannot run the C# or Visual Basic .NET user interface executables after you install TestStand 2012.

To work around this issue, remove the C# and Visual Basic .NET user interfaces before you install TestStand 2012 so TestStand 2012 installs the new user interfaces in the <TestStand Public>\UserInterfaces directory. Alternatively, rebuild the C# and Visual Basic .NET user interfaces after you install TestStand 2012.

Complete the following steps to successfully uninstall TestStand 2012 Beta and remove the user interface files in the <TestStand Public>\UserInterfaces directory before you install TestStand 2012.

1. Navigate to the standard Windows Control Panel facility for adding and removing programs and run the TestStand 2012 Beta and TestStand 2012 Beta Version Selector uninstallers.
2. Navigate to the TestStand 2012 <TestStand Public> directory and delete the UserInterfaces directory.
3. Install TestStand 2012.

## Behavior Changes

---

TestStand introduced the following behavior changes in version 2012:



**Note** Refer to the *NI TestStand Help* for more information about behavior changes between earlier versions of TestStand. Refer to the [TestStand and Windows 7/Vista](#) section of this document for more information about behavior changes in Windows 7/Vista that might affect applications that you write or modify.

- You can no longer debug .NET code using Visual Studio 2008 or 2005. National Instruments recommends that you use Visual Studio 2010 or later to debug .NET code. Refer to the National Instruments website at [ni.com/info](http://ni.com/info) and enter the Info Code `tsvsnet` to access the TestStand support document, *Cannot Debug TestStand 2012 .NET Code Modules in Microsoft Visual Studio 2008 or Earlier*, for more information about why you can no longer use Visual Studio 2008 or 2005 to debug .NET code.
- The Configure menu no longer contains the Report Options or the Database Options items. Select **Configure»Result Processing** to launch the Result Processing dialog box. Click the



icon in the Options column of the Generate Report model plug-in to launch the Report Options dialog box. Click the icon in the Options column of the Database model plug-in to launch the Database Options dialog box.

- TestStand no longer includes support for converting a LabVIEW Test Executive or LabWindows/CVI Test Executive Toolkit sequence file to a TestStand sequence file. The Sequence File Converters option is no longer available in the Tools menu. TestStand no longer installs the *Converting from the LabVIEW Test Executive to NI TestStand* and *Converting from the LabWindows/CVI Test Executive Toolkit to NI TestStand* help files, previously located at <TestStand>\Doc\Help\LVTECompatibility.chm and <TestStand>\Doc\Help\LWTECompatibility.chm, respectively. Refer to the National Instruments website at [ni.com/info](http://ni.com/info) and enter the Info Code `exgin9` for more information about converting a LabVIEW Test Executive and LabWindows/CVI Test Executive Toolkit sequence file to a TestStand sequence file.
- Some Sequence Call steps that used a constant amount of memory in previous versions of TestStand might run out of memory when you run the steps in TestStand 2012. This change affects steps for which all the following conditions exist:
  - You use an expression to specify the file path of a sequence file the step calls.
  - The expression evaluates to a different sequence file each time TestStand executes the step.
  - You execute the step in a loop.
  - You disable the **Optimize Non-Reentrant Calls to This Sequence** option on the General tab of the Sequence Properties dialog box for each sequence the step calls.

Previous versions of TestStand kept loaded only the first executed sequence file, not all the different sequence files to which the expression evaluated. This behavior negatively affected performance due to the repeated loading and unloading of sequence files. To avoid the performance degradation, TestStand 2012 keeps all the sequence files loaded, and memory usage increases each time TestStand loads a different sequence file. In most cases, the increase in memory usage is bounded and preferable to the performance degradation.

If increased memory usage is unacceptable, select **Unload after step executes** for the Unload Option on the Run Options panel of the Properties tab of the Step Settings pane for the step. When you select this option, TestStand does not keep any of the sequence files loaded after executing the step.

- The default report generators in TestStand 2010 SP1 or earlier generated HTML reports or transformed ATML and XML reports to HTML that conformed to the HTML 3.2 standard. The default report generators in TestStand 2012 generate HTML that conforms to the HTML 4.0.1 transitional standard.
- TestStand no longer supports saving sequence files for use in TestStand 2.0.1 and TestStand 2.0.
- When you call the `UndoItem.EditedObjects` property in an instance of the new `UndoItem2` class, the `UndoItem.EditedObjects` property returns the objects the `UndoItem2.TopObjects` property returns. If one of the returned objects does not exist, the `UndoItem.EditedObjects` property returns an error that explains that the property the lookup string specifies does not exist.

- Using an expression to access the TestStand API might be faster, after TestStand first evaluates the expression, than using one or more ActiveX/COM Adapter steps to access the TestStand API.
- The Generic Recordset (NI) and TS 2.x–4.0 Generic Recordset (NI) database schemas now use the `TOP 1` clause in the command text for all statements, as the following example shows, to improve performance when logging results to a database by preventing the recordset from accessing existing records in the database:

```
SELECT TOP 1 * FROM <TableName>
```

You might encounter this performance improvement when you are using database files larger than 500 MB, especially when you enable the **Use On-The-Fly Logging** option on the Logging Options tab of the Database Options dialog box.

- When you insert a Message Popup step in a sequence, the default value for the Title Expression control on the Text and Buttons tab of the Step Settings pane is now `NameOf(Step)` instead of "untitled" so that by default the title of the message is the same as the name of the Message Popup step.
- The **Reset Flags in All Loaded Instances of the Type** option in the Edit Flags dialog box is now the **Apply Changes to Flags and Type Flags to All Loaded Instances of the Type** option.

In previous versions of TestStand, when you enabled the **Reset Flags in All Loaded Instances of the Type** option and clicked **OK**, TestStand reset all flags in all instances of the type to the default values the type specified, regardless of which flags you modified in the Edit Flags dialog box.

In TestStand 2012, when you enable the **Apply Changes to Flags and Type Flags to All Loaded Instances of the Type** option and click **OK**, TestStand applies the changes you make to the flags in the Edit Flags dialog box to all instances of the type that are loaded. When TestStand updates the instances, TestStand updates only the flags you modified in the Edit Flags dialog box.



**Note** The values specified in the Instance Default Flags control in the Edit Data Type Flags dialog box specify the initial flag values TestStand uses when it creates a new instance of the type. If you clear a flag in the Edit Flags dialog box but the same flag is set in the Instance Default Flags control in the Edit Data Type Flags dialog box, TestStand sets the flag in all instances of the type that are loaded because the set state of the flag is the state that the flag is configured to have for new instances of the type.

- Executions return a `.NET System.TypeLoadException` error that indicates that TestStand could not load the `NationalInstruments.TestStand.Utility.TSHelper` type from the `NationalInstruments.TestStand.Utility.dll` assembly when all of the following conditions are true:
  - The C# or Microsoft Visual Basic .NET user interface uses the .NET Common Language Runtime (CLR) version 2.0
  - A local copy of `NationalInstruments.TestStand.Utility.dll` from an earlier version of TestStand is located in the same directory as the user interface
  - You create an on-the-fly report or TestStand results file (`.tsr`)

Replace the local copy of `NationalInstruments.TestStand.Utility.dll` with the current version installed in the `<TestStand>\Bin` directory to prevent the error.

## LabVIEW Support Behavior Changes

TestStand introduced the following behavior changes for LabVIEW support in version 2012:

- TestStand 2012 supports using only LabVIEW 2011 SP1 or earlier to execute a step you originally configured using LabVIEW 7.1.1 or earlier. To use LabVIEW 2012 to execute such a step, click the **Load VI Prototype** button on the LabVIEW Module tab for the step or use the Update VI Calls tool to update the step so you can execute the step using LabVIEW 2012 or later. Refer to the [Getting Started](#) section of this document for more information about supported versions of LabVIEW. Refer to the *NI TestStand Help* for more information about the Update VI Calls tool.
- When you select **Binary String** in the Type column of the VI Parameter Table on the LabVIEW Module tab to store a LabVIEW string that contains binary data in a TestStand property, TestStand handles the string data in different ways depending on the version of LabVIEW you are using, as Table 6 describes.

**Table 6.** TestStand Behavior for Handling Binary Data in a LabVIEW String

LabVIEW Version	TestStand Behavior
<ul style="list-style-type: none"><li>• LabVIEW 2012 or later Development System on a non-multibyte operating system</li><li>• LabVIEW 2009 SP1 or later Development System on a multibyte operating system</li><li>• LabVIEW 2009 SP1 or later RTE on any operating system</li></ul>	TestStand compresses the binary data, encodes the compressed data, and stores the compressed data using only printable ASCII characters. To pass a compressed string to a VI, select <b>Binary String</b> in the Type column. TestStand unencodes and decompresses the binary data the string stores before passing it to the VI.
<ul style="list-style-type: none"><li>• LabVIEW 2011 SP1 or earlier Development System on a non-multibyte operating system</li><li>• LabVIEW 2009 or earlier Development System on a multibyte system</li><li>• LabVIEW 2009 or earlier RTE on any operating system</li></ul>	TestStand escapes the string before storing it and substitutes hexadecimal codes for the unprintable characters, such as the NUL character, in the string. To pass an escaped string to a VI, select <b>Binary String</b> in the Type column. TestStand unescapes the string before passing it to the VI and substitutes the correct character values for the hexadecimal values in the escaped string.



**Note** To retrieve the raw data in TestStand when the binary data is compressed, call the `PropertyObject.GetValBinary` method of the TestStand API on the variable that stores the binary string.

- When you use LabVIEW 2012 or later with TestStand 2012 or later on any operating system to pass a string in a LabVIEW variant data type, TestStand stores the string as either a string variable or an array of numbers. When you used TestStand 2010 SP1 or earlier and LabVIEW 2009 SP1 or later, TestStand stored the string as a string variable only on a non-multibyte operating system or as an array of numbers only on a multibyte operating system.
- TestStand 2012 includes the following changes to LabVIEW user interfaces:
  - The `Load_Top-Level_VI.vi`, located in the `<TestStand Public>\UserInterfaces\Full-Featured\LabVIEW` directory, now launches the LabVIEW user interface in the context of a LabVIEW project.
  - If you use the files located in the `<TestStand Public>\UserInterfaces\Full-Featured\LabVIEW` or `<TestStand Public>\UserInterfaces\Simple\LabVIEW` directories as a starting point for creating a user interface, National Instruments recommends that you always run the user interface in the context of its LabVIEW project to avoid name collisions. If you expect the user interface to run outside the context of the LabVIEW project, you must modify the names of the VIs in the user interface to ensure the names will not collide with the names of code module VIs or its subVIs.
  - A LabVIEW project library now provides a namespace for the LabVIEW simple and full-featured user interface examples. If you load any VI from the user interface by name, you must update the code that loads the VI to include the namespace.
  - When you run the `TestExec.exe` build specification to generate an executable, LabVIEW prefixes VIs listed under the Dependencies node in the LabVIEW project with `Simple UI -` or `Full UI -` for the simple and full-featured user interfaces, respectively, to avoid name collisions. If you load any VI from the user interface by name, you must update the code that loads the VI to include VI prefixes.

Complete one of the following options for all dynamic calls listed under the Dependencies node, including VIs that you call from top-level files but that you have not explicitly added to the My Computer target in the LabVIEW project, to resolve this issue:

  - Add the `Simple UI -` or `Full UI -` prefix to the path you pass to the Open VI Reference function when the caller VI runs in the LabVIEW RTE. You can use the `Application:Kind` property in LabVIEW to determine whether the VI is running in the development system or the LabVIEW RTE.
  - Add the VIs listed under the Dependencies node to the My Computer target to prevent the name change when you build the executable. LabVIEW prefixes only the files listed under the Dependencies node. National Instruments recommends that you prefix these VIs manually to avoid potential name collisions that might prevent the user interface from running VIs that have conflicting names.

# Process Model Incompatibilities

The TestStand 2012 process models are highly compatible with sequences written for previous versions of the TestStand process models. However, the following incompatibilities exist:

- The equivalent legacy TestStand 2010 Batch process model uses Sequence Call steps to call sequences that implement controller and test socket synchronization. The TestStand 2012 Batch process model implements the same functionality in the `ControllerAndSocketSynchronizationManager` class in a .NET code module, `ManagedModelSupport.dll`, located in the `<TestStand>\Components\Models\TestStandModels\ManagedModelSupport` directory.
- If at run time you add or remove properties or resize arrays that the UUT data type or any `ModelData` data type contains, you must select **No** in the New Thread column in the Result Processing dialog box for all model plug-ins. Enable the **Show More Options** option in the Result Processing dialog box to display the New Thread column. You must also not use the Offline Results File result processing plug-in. If you do not follow these instructions, incorrect behavior might occur. You can use the equivalent legacy TestStand 2010 process models to avoid this issue. This restriction might be removed in a future version of TestStand.
- The Single Pass Execution entry point in the TestStand 2012 process models now calls the `PreUUT`, `PostUUT`, `PreUUTLoop`, `PostUUTLoop`, `PreBatch`, `PostBatch`, `PreBatchLoop`, and `PostBatchLoop` callbacks.
- TestStand 2012 changes the type of the **ModelData** parameter of the `PreBatch`, `PostBatch`, and `BatchReport` callbacks from `NI_BatchModelData` to `NIBatchModelData2`. `NIBatchModelData2` no longer contains database- or reporting-related fields.

If you used a previous version of TestStand to create these batch callbacks and the callbacks access any of the removed fields, you must change the **ModelData** parameter type in Batch callbacks to `NI_BatchModelData2` and update any references to the deleted database logging and reporting fields of the **ModelData** parameter. TestStand now uses the **ModelPluginConfiguration** parameter to pass the equivalent database logging and reporting information.

National Instruments recommends updating Batch callbacks in a test system in which you can edit callbacks.

- Because the structure of the model files has changed, you might need to update sequences that make specific assumptions about the implementation of the process models. For example, TestStand 2012 renames and moves some variables and significantly changes some support sequences in `ModelSupport.seq`, located in the `<TestStand>\Components\Models\TestStandModels` directory.
- You can now invoke Report Generation and Database Logging callbacks in separate threads asynchronously relative to the UUT loop thread. You must update existing custom callbacks that rely on being called synchronously relative to the UUT loop thread to support the new asynchronous result processing option.
- The result processing options files from previous versions of TestStand are not compatible with TestStand 2012. The result processing options move from the `TestStandModelReportOptions.ini` and `TestStandDatabaseOptions.ini` files into a single `ResultProcessing.cfg` file, located in the `<TestStand Application Data>\Cfg\ModelPlugins` directory, that stores

configuration information for all result processing model plug-ins. The database schemas in the `TestStandDatabaseOptions.ini` file move to the `TestStandDatabaseSchemas.ini` file.

Refer to the [Migrating TestStand 2010 SP1 or TestStand 2010 Result Processing Options to TestStand 2012](#) section of this document for information about using the TestStand 2012 Migration Utility to migrate the result processing options from previous versions of TestStand installed on the computer to TestStand 2012. Refer to the [Manually Migrating Result Processing Options from Any Version of TestStand](#) section of this document for information about manually migrating the result processing options from previous versions of TestStand installed on the computer to TestStand 2012.

If you deploy a new test system that uses the TestStand 2012 process models to an existing system that uses the equivalent legacy TestStand 2010 process models, you must use the Legacy Model Switcher tool to migrate the report or database options to the TestStand 2012 process models. Use the TestStand Deployment Utility to create an installer that contains the default process models to install the Legacy Model Switcher on the target computer. Refer to the [Using the Legacy Model Switcher Tool](#) section of the *TestStand 2012 New Features* topic in the *NI TestStand Help* for more information about the tool.

- The TestReport callback includes a new **ReportSection** parameter of the Object Reference type. The value of the parameter must be `Nothing` or must refer to a `ReportSection` object. The TestStand 2012 process models use this parameter instead of the **Report** parameter to improve performance. The equivalent legacy TestStand 2010 process models continue to use the **Report** parameter and pass the value `Nothing` for the **ReportSection** parameter. Refer to the *NI TestStand Help* for more information about the TestReport callback.
- TestStand now calls the ModelOptions callback before calling the ProcessSetup and PreUUTLoop callbacks.
- TestStand now calls the PostBatch callback after, instead of before, calling the PostUUT callback in each socket.

If you encounter an incompatibility in an existing test system when you use the TestStand 2012 process models or other circumstances prevent you from using the TestStand 2012 process models, you can use the equivalent legacy TestStand 2010 process models. However, National Instruments recommends using the TestStand 2012 process models in most cases.



**Note** If you migrate a custom legacy process model file to TestStand 2012, you must also migrate the corresponding `ModelSupport.seq` file from the earlier version of TestStand, or you must use the equivalent legacy TestStand 2010 process models, which include a legacy `ModelSupport.seq` file.

You can manually select the equivalent legacy TestStand 2010 process models to use, or you can use the Legacy Model Switcher tool to replace the TestStand 2012 process models with equivalent legacy TestStand 2010 process models located in the `<TestStand>\Components\Models\TestStandModels` directory or vice versa. The Legacy Model Switcher tool does not affect any custom process models. You can also use the Legacy Model Switcher tool to migrate the result processing options from the new TestStand 2012 process

models to the equivalent legacy TestStand 2010 process models or vice versa. You must restart all TestStand applications for the new settings to take effect.

Select **Configure»Station Options** to launch the Station Options dialog box, click the **Model** tab, and use the **Station Model** control to browse to the <TestStand>\Components\Models\TestStandModels\TestStand 2010 Process Models (Legacy) directory and select the equivalent legacy TestStand 2010 process model you want to use. You can also select **Edit»Sequence File Properties** to launch the Sequence File Properties dialog box, click the **Advanced** tab, select **Require Specific Model** in the **Model Option** control, and use the **Model File** control to browse to the equivalent legacy TestStand 2010 process model you want to use.

Select **Start»All Programs»National Instruments»TestStand»Tools»Compatibility»Legacy Model Switcher** or navigate to <TestStand>\Components\Models\TestStandModels\TestStand 2010 Process Models (Legacy)\LegacyModelSwitcher.exe to launch the Legacy Model Switcher tool. You can also use the /NEW or /LEGACY command-line arguments to programmatically select which TestStand process models to use.



**Note** This document reflects only the TestStand 2012 process models. If you are using the equivalent legacy TestStand 2010 process models, refer to the TestStand 2010 documentation for information about those process models.

## What's New in TestStand 2012

---

TestStand 2012 includes the following new features and other changes since TestStand 2010 SP1. Refer to the *TestStand 2012 New Features* topic in the *NI TestStand Help* or refer to the National Instruments website at [ni.com/info](http://ni.com/info) and enter the Info Code `ts2012new` for more information about each new feature.

- Process model result processing plug-in architecture
  - Result processing improvements
  - TestStand Offline Results Processing Utility
  - Legacy Model Switcher tool
- TestStand report functionality improvements
  - On-the-fly report enhancements
  - ATML Test Results report enhancements
  - Other report improvements
- Database logging changes
  - Database options configuration changes
  - Controlling database logging memory usage
- Password-protecting types
- .NET support enhancements
- LabVIEW Adapter enhancements
  - Specifying a VI or class member to call for a step
  - Selecting a LabVIEW class or class member

- LabVIEW ring control support
- Normalizing numeric values you pass to enumeration parameter values
- Edit LabVIEW VI Call dialog box enhancements
- Calling DLLs that use a side-by-side version of the LabWindows/CVI RTE
- TestStand Deployment Utility enhancements
- Dragging and dropping files to create steps
- Excluding hidden subdirectories of search directories
- Expression language enhancements
- Displaying code module preload progress
- Manipulating array elements and array element values
- Documentation and example programs enhancements
- TestStand Sequence Analyzer Rules Additions

## TestStand API and TestStand UI Controls API Additions

---

Refer to the *TestStand 2012 API and UI Controls Additions and Changes* topic in the *NI TestStand Help* for more information about new API classes, methods, and properties. Additionally, some API methods and properties are now obsolete. The *NI TestStand Help* also documents API changes and additions for some earlier versions of TestStand.



**Note** Although some components, such as the sequence editor, expose .NET assemblies, TestStand does not support the use of undocumented assembly API.

## Submitting Feedback on TestStand Documentation

---

TestStand 2012 includes changes to the *NI TestStand Help*, such as a reorganized table of contents and a new *Fundamentals* book. National Instruments welcomes feedback on the changes to the *NI TestStand Help*. You can use the new feedback link located on the bottom of each help topic to submit feedback about that specific help topic or other help-related information.

CVI, LabVIEW, National Instruments, NI, ni.com, NI TestStand, the National Instruments corporate logo, and the Eagle logo are trademarks of National Instruments Corporation. Refer to the *Trademark Information* at [ni.com/trademarks](http://ni.com/trademarks) for other National Instruments trademarks. The mark LabWindows is used under a license from Microsoft Corporation. Windows is a registered trademark of Microsoft Corporation in the United States and other countries. Other product and company names mentioned herein are trademarks or trade names of their respective companies. For patents covering National Instruments products/technology, refer to the appropriate location: **Help»Patents** in your software, the `patents.txt` file on your media, or the *National Instruments Patents Notice* at [ni.com/patents](http://ni.com/patents). You can find information about end-user license agreements (EULAs) and third-party legal notices in the NI TestStand 2012 Readme. Refer to the *Export Compliance Information* at [ni.com/legal/export-compliance](http://ni.com/legal/export-compliance) for the National Instruments global trade compliance policy and how to obtain relevant HTS codes, ECCNs, and other import/export data.

© 2004–2012 National Instruments. All rights reserved.