NATIONAL INSTRUMENTS

T<small>EST</small>S<small>TAND</small>™ R<small>ELEASE</small> N<small>OTES</small>

Version 2.0

These release notes contain system requirements, installation instructions, new features, and updated information to help you begin using TestStand 2.0.

Contents

TestStand Installation

The TestStand setup program installs TestStand on your computer in approximately 10 minutes.

Minimum System Requirements

To run TestStand for Windows 2000/NT/Me/9*x*, you must have the following:

- Windows 2000/NT 4.0 or later, or Windows Me/9*x*. For Windows NT, National Instruments recommends Service Pack 3 or later.

- At least a 266 MHz Pentium class or higher microprocessor

- SVGA resolution or higher video adapter, with a minimum 800×600 video resolution

- Minimum of 64 MB of memory

- 100 MB free hard disk space (250 MB recommended)

- Microsoft-compatible mouse

Installing TestStand

Note National Instruments recommends that you close other applications before you install TestStand.

Caution If you have LabVIEW VIs that you saved with a version of LabVIEW older than 5.1 that call the TestStand API, you must mass compile them in LabVIEW 5.1 or later before installing TestStand 2.0. If you do not do mass compile your VIs, you will have to manually replace every ActiveX diagram node that uses the TestStand API. The TestStand 2.0 installer displays a message box if it detects an existing LabVIEW installation.

Follow these instructions to install TestStand:

1. Verify that your computer and monitor are powered on and that you have installed Windows 2000/NT 4.0 or later, or Windows Me/9*x*.

2. Insert your TestStand Installation CD into your CD-ROM drive.

3. Choose **Run** from the **Start** menu.

4. Type *x*:\tssetup.exe in the input box, where *x* is your CD-ROM drive, and click on **OK**.

5. Follow the instructions that appear in the dialog boxes.

National Instruments recommends that you install the complete TestStand program to take full advantage of all the TestStand capabilities. If you choose to install with options, select the options you want and follow the directions on the screen. If necessary, you can run the setup program again later and install additional files.

Refer to the `<TestStand>\Doc\Readme.txt` file for any additional issues that are too recent to be included in the printed TestStand documentation.

Upgrade Notes

This section contains information on how to upgrade from a previous version of TestStand.

Preserving Your Changes to TestStand Components

TestStand 2.0 includes changes to several components that users may also change. For example, TestStand 2.0 enhances the process model, operator interfaces, and certain step types. If you have made changes to one of these components and placed the changes in the appropriate user directory, TestStand 2.0 will not overwrite your changes. Your modified component will continue to function correctly with TestStand 2.0.

However, if you want both the changes you have made to a component and the changes that TestStand 2.0 provides, you must do one of the following:

- Use a file-differencing tool to determine the changes between the TestStand 2.0 version of the component and the original version of the component that you modified. Apply the TestStand 2.0 improvements to your version of the component.

- Use a file-differencing tool to determine the changes you made to the component. Reapply your improvements to a copy you make of the TestStand 2.0 version of the component.

Which option you choose depends on whether you or TestStand 2.0 made the larger number of changes to the component.

To compare sequence files, use the sequence file diff/merge tool in the TestStand 2.0 sequence editor. To compare text files, use a source code differencing tool such as Microsoft Windiff, or the file diff/merge tool in the LabWindows/CVI editor.

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 2.0 process model with the corresponding sequence from the 1.0.*x* process model without making further modifications.

Installing Multiple TestStand Versions on the Same Machine

National Instruments recommends that you install only one version of TestStand on a particular machine. Although TestStand 2.0 will install on a machine that contains a previous TestStand version, only one version of TestStand can be active at a time. If you must install TestStand 2.0 on the same machine as an earlier TestStand version, you can use the TestStand Version Selection utility to select which TestStand version is active. You can find the Version Selection utility in the following directory, `<Program Files>\NationalInstruments\Shared\TestStandVerSelect\ TSVerSelect.exe`.

If you activate TestStand 2.0 and run an operator interface from your 1.0.*x* installation, the operator interface uses the 2.0 engine, step types, and components. If you activate TestStand 1.0.*x* and run a TestStand 2.0 operator interface or the sequence editor, the application you run will not function correctly.

TestStand 2.0 installs shortcut items to the **Start»Programs»National Instruments»TestStand** menu. Previous versions of TestStand installed shortcut items to the **Start»Programs»National Instruments TestStand** menu. When you install TestStand 2.0 without uninstalling a previous version of TestStand, your start menu can contain both groups of shortcut items. The version switcher utility does not alter start menu items or other application shortcuts. You must ensure that you run the version of the sequence editor or operator interface that you intend.

What Is New and Different in TestStand 2.0

New Features

TestStand 2.0 contains the following new features:

- Two new process models, Batch Model and Parallel Model
- Synchronization step types
- Built-in step synchronization properties
- Asynchronous sequence calls
- HTBasic adapter
- API additions
- Workspaces, projects, and source code control
- Sequence file diff/merge tool
- IVI step types
- Instrument session manager

- Report enhancements
  - Automatic reporting
  - Batch reports
  - XML reports example
- Database logging
  - New database logger
  - TestStand 2.0 schema
  - Measurement logging
- Combined property loader
- Numeric Limit Test step enhancements
  - Units
  - Numeric Format
- Multiple Numeric Limit Test step
- Message Popup step enhancements
  - Timeouts
  - Positioning
  - Modality
- DLL adapter enhancements
  - Structure passing
  - Enumeration parameters
  - Function call editing
- New engine callbacks
- Declare variable in the expression browser
- Save as version 1.0.*x*
- Step type enhancements
  - Multiple type palettes
  - Unlimited substeps
  - Combining step types
  - Step type menu editor
  - Version properties
  - Apply changes to all step instances
- Update sequence files tool
- Various performance improvements
- Variable propagation
- Localized decimal point support

- Special floating point numbers: NAN, IND, INF
- New examples
- List view tip strips
- Installable tool menu items
- Operator interface manual source
- Failure chain in reports
- Documentation generator options
- File dialog box directory history list
- Improvements to the Assemble Test VIs

Changes Between TestStand 2.0 and TestStand 1.0.*x*

The following are changes in behavior between TestStand 2.0 and TestStand 1.0.*x*.

Numeric Limit Step

In TestStand 1.0.*x*, the numeric limit step provides a radix selection ring that specifies the numeric format of limit values in the Edit Limits dialog box. In TestStand 2.0, these radix selections are a subset of a more general set of numeric formatting options. Unlike TestStand 1.0.*x*, the numeric format option now applies to the limit and measurement values for the step in the test report.

Creating Subproperties

TestStand 2.0 no longer allows you to create subproperties under a property that is not a container. To conform to this change, TestStand has adjusted the structure of the User data type as follows:

- Each privilege group is a container property instead of a Boolean.
- Each privilege group has a Boolean subproperty named GrantAll. Instead of setting the Boolean value of a privilege group, you now set the GrantAll property to specify whether a user has all privileges within a privilege group.

Applications that check privileges with the CurrentUserHasPrivilege function will continue to function correctly with no changes. If your application depends on the ability to create subproperties within non-container properties, you can set the NoSubPropertiesAllowedIn NonContainers entry in TestExec.ini to False. However, to ensure compatibility with future versions of TestStand, National Instruments recommends that you contact TestStand support for assistance in restructuring your application to avoid the use of this obsolete feature.

Executing Steps

When an execution is terminating in TestStand 1.0.*x* and a step in a cleanup step group calls a subsequence, the steps in the main step group of the subsequence do not execute. TestStand 2.0 correctly executes the steps in the main step group of the subsequence.

TestStand 1.0.*x* incorrectly checks the step run mode for `Force Pass` and `Force Fail` before evaluating the step precondition. TestStand 2.0 checks only the run mode for `Skip` before evaluating the step precondition. TestStand 2.0 checks for `Force Pass` and `Force Fail` after evaluating the step precondition. Refer to Table 6-5, *Order of Actions That a Step Performs*, for more information.

LabVIEW Palette

In TestStand 2.0, the TestStand API Enumerations.vi is combined with TestStand API Numeric Constants.vi. TestStand 2.0 does not have the VIs Get Property Value (Variant).vi and TestStand Set Property Value (Variant).vi because of a problem with LV variant indicators. For more information, refer to the LabVIEW Related Problems in `<TestStand>\Doc\Readme.txt`. TestStand 2.0 has three new VIs: `GetSession.vi`, `TestStand - Start Modal Dialog.vi`, and `TestStand - End Modal Dialog.vi`. The VIs `InitializeTerminationMonitor.vi`, `GetMonitorStatus.vi`, and `CloseTerminationMonitor.vi` replace `Check for Stopping Execution.vi`. The VIs `IVI Refnum to session.vi`, `Session to IVI refnum.vi`, `VISA Refnum to session.vi`, and `Session to VISA refnum.vi` replaced `Refnum to session.vi` and `Session to refnum.vi`.

Changes Between TestStand 2.0 and TestStand 2.0 Beta

The following are changes in behavior between TestStand 2.0 and TestStand 2.0 Beta:

- The TestStand 2.0 Beta software specifies that primary key columns that store GUID values use a string data type of size 32 characters. In the final TestStand 2.0 release, the data type for Access is GUID and the data type for SQL Server is UNIQUEIDENTIFIER.

- Several methods in the final TestStand 2.0 API have changed since the 2.0 Beta. To ensure correct operation of a component you built with a 2.0 Beta API, you must recompile the component with the release version of the TestStand 2.0 API and correct any compile errors. Beta components that you do not build, such as the TestStand sequence editor, do not run with the release API. You must use the release versions of these components when you use the release API.

Changes to the TestStand User Manual

This section contains information on several changes and additions to the *TestStand User Manual*.

Chapter 2—Sequence Editor Options

Workspace Window

The icons in the Workspace window have changed. Replace the section on page 2-9 with the following text:

The following icons appear in the Workspace window.

- Project file.
- Workspace file.
- Sequence file.
- Any file that does not have a file association.
- File is a code module or folder contains code modules.
- File not found on disk.
- File is currently checked into the source code control system.
- File is currently checked out of the source code control system.

Multiline Edit Control in Dialog Boxes

Add the following Note to page 2-18, before the Controlling Sequence Flow section.

Note In TestStand 2.0, pressing <Enter> in a comment control in any dialog box closes that dialog box. To insert a carriage return in a comment control, press <Ctrl-Enter>.

Chapter 6—Sequence Execution

Table 6-5 on page 6-25 has changed. Replace the table and text with the following:

Table 6-5. Order of Actions That a Step Performs

Action Number	Description	Remarks
1	Enter batch synchronization section	If option is set
2	Allocate step result	—
3	Check run mode for `Skipped`	—
4	Evaluate precondition	—

Table 6-5. Order of Actions That a Step Performs (Continued)

Action Number	Description	Remarks
5	Acquire step lock	If option is set
6	Check run mode for `Force Pass` and `Force Fail`	—
7	Load module if not already loaded	—
8	Evaluate Loop Initialization expression	Only if looping
9	Evaluate Loop While expression, skip to action 23 if `False`	Only if looping
10	Allocate loop iteration result	Only if looping
11	Call Pre Step engine callbacks	—
12	Evaluate Pre expression	—
13	Call Pre Step substeps for step type	—
14	Call module	—
15	Call Post Step substeps for step type	TestStand calls Post Step substeps even if the user code module generates a run-time error. This enables Post Step substeps to perform error handling, if appropriate.
16	Evaluate Post expression	—
17	Evaluate Status expression	—
18	Call Post Step engine callbacks	—
19	Call Post Step failure engine callbacks	Only if loop iteration fails
20	Fill out loop iteration result	Only if looping
21	Call PostResultListEntry engine callbacks	Only if looping
22	Evaluate Loop Increment expression, return to action 9	Only if looping
23	Evaluate Loop Status expression	Only if looping
24	Unload module if required	—

Table 6-5. Order of Actions That a Step Performs (Continued)

Action Number	Description	Remarks
25	Update sequence failed state	
26	Call Post Step failure engine callbacks	Only if step fails
27	Execute post action	—
28	Release step lock	If option is set
29	Fill out step result	—
30	Call PostResultListEntry engine callbacks	—
31	Exit batch synchronization section	If option is set

Usually, a step performs only a subset of these actions, depending on the configuration of the step and the test station. When TestStand detects a run-time error, it usually proceeds to action 29. If a run-time error occurs in a loop iteration, TestStand performs action 20 before it performs action 29.

Chapter 11—Synchronization Step Types

Add the following note to the *Mismatched Sections* section on page 11-58.

Note Typically, you set the step batch synchronization property to Parallel for any step that can cause a branch to a previous step, for example, a goto step or a step with a goto post action. This ensures that the branching step executes concurrently for all sequence instances, even when the step is nested within another synchronized section. Executing the backward branch concurrently prevents unintended mismatched sections that can result from one thread reaching a previous section while another thread that has yet to branch back waits at a different section before the branch.

Chapter 18—Databases

Add the following bullet below New Data Link in the Database Viewer *File Menu* section on page 18-33.

- **Save Data Link**—Prompts the user for a filename and saves the data link of the active window to a .udl file. This option is only visible when a TableView or ExecuteSQL window is active.

Additional Support Information

For additional information on TestStand, refer to the National Instruments Web site at `ni.com/support/teststand/`. The Web site contains many electronic support services such as a KnowledgeBase database, National Instruments Developer Zone (NIDZ), example programs, and software libraries. These resources can help you in your development.

National Instruments also offers an e-mail-based TestStand technical discussion forum. You can join the forum by sending an e-mail to `listserv@listserv.tamu.edu` with the following text in the body of the note:

```
SUBSCRIBE TESTEXEC <first name> <last name>
```

You will receive an e-mail response with further instructions on sending messages to the forum and unsubscribing from the forum.