

Hinweise zum Upgrade von LabVIEW™

In diesem Dokument wird beschrieben, wie LabVIEW unter Windows, OS X und Linux auf LabVIEW 2014 aktualisiert wird. Lesen Sie dieses Dokument, bevor Sie mit dem Upgrade beginnen. Es enthält Hinweise zu folgenden Themen:

- die empfohlene Vorgehensweise für das Upgrade von LabVIEW
- Hinweise zu möglichen Kompatibilitätsproblemen beim Laden von VIs, die mit älteren Versionen von LabVIEW erstellt wurden
- Änderungen und Neuerungen in LabVIEW 2014

Inhalt

Upgrade auf LabVIEW 2014.....	2
1. Sichern der VIs und der Computerkonfiguration.....	2
2. Prüfen der aktuellen Funktionsweise der VIs.....	4
3. Installieren von LabVIEW, Zusatzpaketen und Gerätetreibern.....	4
4. Umwandeln der VIs und Beseitigen von Funktionsunterschieden.....	4
Fehlersuche beim Upgrade.....	6
Hinweise zur Kompatibilität beim Upgrade.....	6
Upgrade von LabVIEW 2009 oder älteren Versionen.....	6
Upgrade von LabVIEW 2010.....	6
Upgrade von LabVIEW 2011.....	8
Upgrade von LabVIEW 2012.....	9
Upgrade von LabVIEW 2013.....	12
Änderungen und Neuerungen in LabVIEW 2014.....	14
LabVIEW-Taskleistensymbole mit Versionsangabe und Bitanzahl.....	14
Installieren und Aktualisieren von DataPlugins in LabVIEW.....	15
Verbesserungen am Blockdiagramm.....	15
Verbesserungen am Frontpanel.....	16
Verbesserungen an der Programmoberfläche.....	16
Neue und geänderte VIs, Funktionen und Knoten.....	17
Verbesserungen am Application Builder.....	19
Erweiterter Funktionsumfang durch integrierte Zusatzpakete.....	20
Neue LabVIEW-Suites.....	21
Verbesserungen an LabVIEW-Webdiensten.....	21
Benutzerdefinierte Dokumentation für VIs und Anwendungen.....	21
LabVIEW 2014 (64 Bit) für Mac OS X.....	22
LabVIEW 2014 (64 Bit) für Linux.....	23
Änderungen an den Touchpanel-Funktionen.....	23
Neue und geänderte Toolkits.....	24
Neuerungen und Änderungen in vorherigen LabVIEW-Versionen.....	24

Upgrade auf LabVIEW 2014

Zum Umstellen kleinerer Anwendungen auf eine neue LabVIEW-Version genügt es in der Regel, die VIs in der neuen LabVIEW-Version zu laden. Um durch die Umstellung verursachte Probleme so effizient wie möglich zu erkennen und zu beheben, wird jedoch eine systematischere Vorgehensweise empfohlen.



Tipp Große LabVIEW-Anwendungen zur Steuerung und Überwachung von kritischen Operationen sowie Anwendungen, deren Ausfall kostspielig ist, profitieren insbesondere von der systematischen Umstellung. Das Gleiche gilt für Anwendungen, die mit mehreren Modulen und Toolkits arbeiten oder die in einer nicht unterstützten LabVIEW-Version gespeichert sind. Informationen zu den LabVIEW-Versionen, zu denen Standard-Support erhältlich ist, finden Sie auf der Webseite von National Instruments unter ni.com/info nach Eingabe des Infocodes `lifecycle`.

Überblick über den empfohlenen Upgrade-Prozess

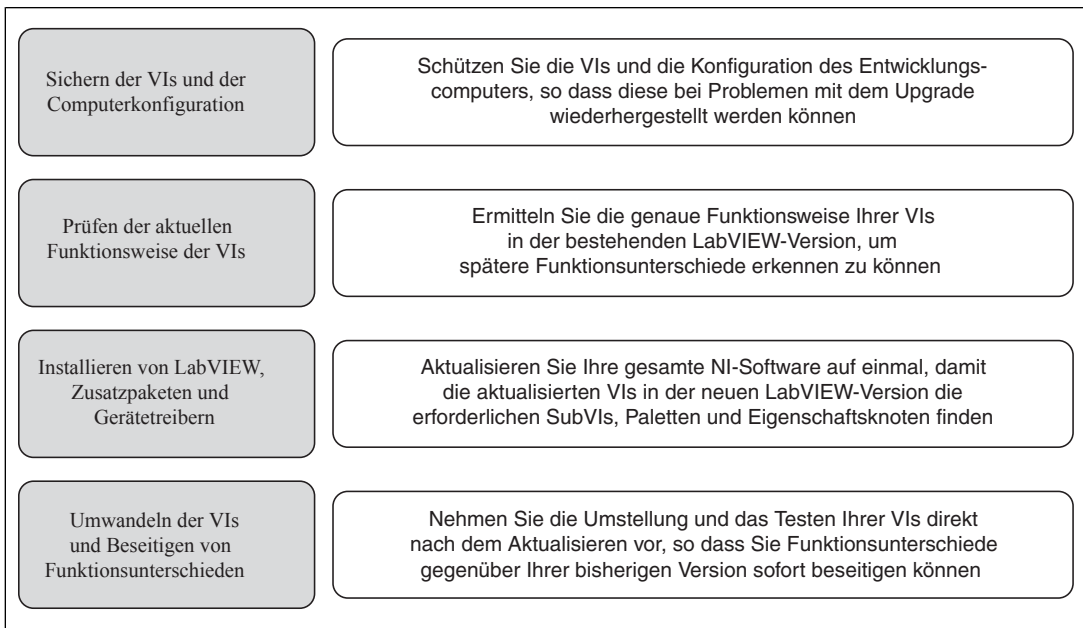


Figure 1.



Hinweis Zum Umstellen Ihrer VIs von einer LabVIEW-Version bis einschließlich 5.1 müssen Sie die VIs zuerst auf eine Zwischenversion von LabVIEW aktualisieren. Weitere Informationen zum Upgrade einer älteren LabVIEW-Version erhalten Sie auf der Website ni.com/info nach Eingabe des Infocodes `upgradeOld`.

1. Sichern der VIs und der Computerkonfiguration

Durch Sichern einer Kopie der VIs und – sofern möglich – der Konfiguration des Computers vor dem Upgrade auf LabVIEW 2014 wird gewährleistet, dass Sie den Upgrade-Prozess bei Bedarf erneut beginnen können.

a. Sichern von VIs

Wenn Sie Ihre VIs vor dem LabVIEW-Upgrade sichern, können Sie die alte Version problemlos wiederherstellen. Ohne die Sicherungskopie können Sie die aktualisierten VIs nicht mehr in älteren LabVIEW-Versionen öffnen. Die VIs müssen dann erst in der neuen Version für eine ältere Version gespeichert werden.

Die VIs können auf folgende Art und Weise gesichert werden:

- **Speichern der VIs in einem Versionsverwaltungssystem**—Ermöglicht das Wiederherstellen der vorherigen Dateiversion, wenn sich durch die Umstellung verursachte Funktionsunterschiede nicht beheben lassen. Weitere Informationen zur Verwendung von Versionsverwaltungssystemen mit LabVIEW finden Sie in der *LabVIEW-Hilfe* auf der Registerkarte **Inhalt** unter dem Thema **Grundlagen»Arbeiten mit Projekten und Zielen»Allgemeines»Verwendung der Versionsverwaltung in LabVIEW**.
- **Erstellen einer Kopie der VIs**—Ermöglicht das Erstellen einer Sicherungskopie. Gehen Sie je nach der ursprünglichen Speichermethode der VIs dazu wie folgt vor:
 - Als Projekt gespeichert—Öffnen Sie das Projekt und speichern Sie über **Datei»Speichern unter** eine Kopie der *.lvproj-Datei und des Projektinhalts. Wählen Sie **Alle Abhängigkeiten einschließen**, um auch Komponenten einzuschließen, von denen das Projekt abhängt.
 - In einer LLB oder einem Verzeichnis gespeichert—Erstellen Sie im Datei-Explorer Ihres Betriebssystems eine Kopie der LLB oder des Verzeichnisses und speichern Sie diese an einer anderen Stelle. Zum Vermeiden von Namenskonflikten sollte die Kopie nicht auf derselben Festplatte gespeichert werden.

b. Sichern der Computerkonfiguration

Beim Upgrade von LabVIEW werden versionsübergreifend verwendete gemeinsame Dateien verändert. Das kann in Ausnahmefällen dazu führen, dass Ihre VIs in älteren Versionen nicht mehr erwartungsgemäß funktionieren. Die vorherige Version der gemeinsam genutzten Dateien kann nach dem Installieren der neuen LabVIEW-Version nicht ohne Weiteres wiederhergestellt werden. Daher sollten Sie die Konfiguration der NI-Software auf Ihrem Entwicklungscomputer mit einer der nachfolgend aufgeführten Methoden sichern. Das gilt insbesondere, wenn Ihre bisher verwendete LabVIEW-Version nicht mehr unterstützt wird oder ein Ausfall Ihrer Anwendungen kostspielig wäre.

- **Erstellen eines Abbilds der Computerkonfiguration**—Erstellen Sie vor dem Umstellen auf die neue Version mit Hilfe von *Disk-Imaging-Software* ein Festplattenabbild des Computers, das die Dateien, die installierte Software sowie sämtliche Benutzereinstellungen enthält. Mit dem Festplattenabbild können Sie den Computer nach dem Upgrade ggf. wieder in den Originalzustand zurückversetzen.
- **Testen Sie den Upgrade-Prozess auf einem Testcomputer**—Obwohl das LabVIEW-Upgrade auf einem Testcomputer gegenüber dem Erstellen eines Festplattenabbilds zeitaufwändiger ist, wird diese Vorgehensweise dennoch dringend für alle ausfallkritischen Systeme in Produktionsprozessen empfohlen. Nach dem Beheben eventueller Upgrade-Probleme auf dem Testcomputer können Sie den Produktionscomputer entweder durch den Testcomputer ersetzen oder den Upgrade-Prozess auf dem Produktionscomputer wiederholen.



Tipp Damit VIs auf dem Testcomputer genauso funktionieren wie auf dem später genutzten Computer, sollte die Konfiguration der Computer (u. a. hinsichtlich CPU, RAM, Betriebssystem und Softwareversionen) möglichst ähnlich sein.

2. Prüfen der aktuellen Funktionsweise der VIs

Die Neuerungen und Änderungen zwischen LabVIEW-Versionen können gelegentlich Funktionsunterschiede von VIs verursachen. Testen Sie die VIs daher in beiden LabVIEW-Versionen. Vergewissern Sie sich, dass Ihnen aktuelle Ergebnisse für die folgenden Tests vorliegen, sofern diese auf Ihre VIs zutreffen:

- Führen Sie eine Massenkompilierung der VIs in der Vorgängerversion von LabVIEW durch und prüfen Sie das dabei erstellte Massenkompilierungsprotokoll auf fehlerhafte VIs. Diese Angaben sind besonders nützlich, wenn mehrere Personen an der VI-Entwicklung beteiligt sind oder wenn Sie vermuten, dass VIs lange nicht mehr kompiliert worden sind. Damit ein Massenkompilierungsprotokoll erstellt wird, aktivieren Sie die Option **Ergebnisse aufzeichnen** im Dialogfeld **Massenkompilierung**. Weitere Informationen dazu finden Sie auf der Registerkarte **Inhalt** der *LabVIEW-Hilfe* unter **Grundlagen»Erstellen von VIs und SubVIs»Anleitung»Speichern von VIs»Massenkompilierung von VIs**.
- Prüfen Sie mit Modultests, ob einzelne VIs ihre vorgesehenen Funktionen erfüllen.
- Prüfen Sie mit Integrationstests, ob SubVIs in einem Projekt oder einer Gruppe erwartungsgemäß zusammenarbeiten.
- Prüfen Sie mit Einsatztests, ob die VIs nach dem Übertragen auf ein Zielsystem (z. B. einen Desktop-Computer oder FPGA-Chip) erwartungsgemäß funktionieren.
- Prüfen Sie mit Leistungstests die CPU-Auslastung, den Speicherbedarf und die Ausführungsgeschwindigkeit des Programmcodes. Im Fenster **Profil - Leistung und Speicher** finden Sie Schätzwerte für die durchschnittliche Ausführungsgeschwindigkeit der VIs.
- Prüfen Sie mit Stresstests, ob die VIs unerwartete Daten korrekt verarbeiten.

Weitere Informationen zum Testen der VIs finden Sie auf der Registerkarte **Inhalt** der *LabVIEW-Hilfe* unter **Grundlagen»Entwicklungsrichtlinien für Anwendungen»Allgemeines»Entwicklung großer Anwendungen»Phasen von Entwicklungsmodellen»Testen von Anwendungen**.



Hinweis Wenn sich VIs nach der Massenkompilierung oder nach Tests geändert haben, sollten Sie zunächst die neuen Versionen der VIs sichern.

3. Installieren von LabVIEW, Zusatzpaketen und Gerätetreibern

a. Installieren von LabVIEW einschließlich Modulen, Toolkits und Treibern

Beim Umsteigen auf eine neue LabVIEW-Version müssen Sie neben dem LabVIEW Development System auch Module, Toolkits und Treiber für die neue LabVIEW-Version installieren. Informationen zur Installationsreihenfolge dieser Komponenten finden Sie in den *Installationshinweisen zu LabVIEW*.

b. Kopieren von user.lib-Dateien

Damit alle benutzerdefinierten Elemente und VIs aus der bisherigen LabVIEW-Version auch für die VIs in LabVIEW 2014 verfügbar sind, müssen Sie das Verzeichnis `labview\user.lib` Ihrer bisherigen Version in das Verzeichnis `labview\user.lib` von LabVIEW 2014 kopieren.

4. Umwandeln der VIs und Beseitigen von Funktionsunterschieden

VIs werden durch eine Massenkompilierung in LabVIEW 2014 auf diese LabVIEW-Version umgestellt. Das dabei erzeugte Fehlerprotokoll enthält Angaben zu fehlerhaften VIs. Werten Sie diese Angaben anhand des Abschnitts *Hinweise zur Kompatibilität beim Upgrade* in diesem Dokument aus, um durch das Upgrade verursachte Funktionsunterschiede zu erkennen.

a. Massenkompilierung von VIs in der neuen Version von LabVIEW

Bei der Massenkompilierung werden die VIs gleichzeitig konvertiert und in LabVIEW 2014 gespeichert. Danach können die VIs jedoch nicht mehr in einer Vorgängerversion von LabVIEW geöffnet werden.

Zum Öffnen der VIs in einer Vorgängerversion muss für jedes VI oder Projekt die Option **Datei»Für vorige Version speichern** gewählt werden. Daher sollten VIs nur dann massenkompiliert werden, wenn sie auf die neue Version umgestellt werden sollen. Zum Erkennen aktualisierungsbezogener Probleme führen Sie eine Massenkompilierung mit Fehlerprotokoll durch, indem Sie im Dialogfeld **Massenkompilierung** die Option **Ergebnisse aufzeichnen** aktivieren.

Informationen zur Massenkompilierung finden Sie in der *LabVIEW-Hilfe* unter folgenden Themen:

- **Grundlagen»Erstellen von VIs und SubVIs»Anleitung»Speichern von VIs»Massenkompilierung von VIs**
- **Grundlagen»Erstellen von VIs und SubVIs»Anleitung»Speichern von VIs»Gängige Fehlermeldungen bei der Massenkompilierung**

b. Korrigieren von Fehlern in VIs

Änderungen und Verbesserungen zwischen Ihrer bisherigen LabVIEW-Version und LabVIEW 2014 können zu VI-Fehlern führen. Zum schnellen Erkennen und Korrigieren fehlerhafter VIs in LabVIEW 2014 führen Sie die folgenden Schritte aus:

1. Um durch das Upgrade verursachte VI-Fehler zu erkennen, vergleichen Sie das im vorherigen Schritt erzeugte Massenkompilierungsprotokoll mit dem Protokoll, das Sie beim Prüfen der aktuellen Funktionsweise der VIs erzeugt haben.
2. Um festzustellen, ob die VI-Fehler durch LabVIEW-Updates verursacht worden sind, lesen Sie den Abschnitt *Hinweise zur Kompatibilität beim Upgrade* in diesem Dokument.

c. Erkennen und Korrigieren von Funktionsunterschieden

Auch wenn National Instruments erhebliche Anstrengungen zur Wahrung der Rückwärtskompatibilität Ihrer VIs unternimmt, kommt es dennoch aufgrund von Verbesserungen oder Fehlerkorrekturen hin und wieder zu Funktionsunterschieden. Mit den folgenden Tools kann auf einfache Weise ermittelt werden, ob Ihre VIs von Änderungen an der neuen LabVIEW-Version betroffen sind:

- **Upgrade VI Analyzer Tests**—Mit Hilfe dieser Tests werden viele der durch die Aktualisierung verursachten Funktionsunterschiede in einer Vielzahl von VIs auf effiziente Weise erkannt. Führen Sie zum Herunterladen und Verwenden dieser Tests folgende Schritte aus:
 1. Laden Sie die Upgrade VI Analyzer Tests für alle Nachfolgerversionen Ihrer aktuellen LabVIEW-Version herunter. Zur Download-Seite für diese Tests gelangen Sie, indem Sie ni.com/info besuchen und den Infocode `analyzevi` eingeben.
 2. Klicken Sie zum Starten der Tests auf **Werkzeuge»VI-Analyseprogramm»VIs analysieren** und starten Sie eine neue VI-Analyse-Aufgabe. Zur Analyse eines gesamten Projekts wählen Sie diesen Menüpunkt aus dem **Projekt-Explorer** aus.
 3. Zum Beheben der Probleme lesen Sie den Abschnitt *Hinweise zur Kompatibilität beim Upgrade* für die LabVIEW-Version, in der die Tests fehlgeschlagen sind. Wenn z. B. mit den LabVIEW 2011 Upgrade VI Analyzer Tests ein potentieller Funktionsunterschied erkannt wird, lesen Sie den Abschnitt *Upgrade von LabVIEW 2010*.
- **Upgrade-Dokumentation**
 - Abschnitt *Hinweise zur Kompatibilität beim Upgrade* in diesem Dokument—Listet Änderungen an LabVIEW auf, die zu Funktionsunterschieden Ihrer VIs führen können. Lesen Sie den Unterabschnitt für jede LabVIEW-Version beginnend bei Ihrer bisherigen Version.



Tipp Zum schnellen Auffinden von veralteten Objekten und anderen Objekten, die im Abschnitt *Hinweise zur Kompatibilität beim Upgrade* erwähnt werden, öffnen Sie Ihre aktualisierten VIs und wählen Sie **Bearbeiten»Suchen und Ersetzen** aus.

- Liste bekannter Probleme in LabVIEW 2014—Führt Programmfehler auf, die bis zur Veröffentlichung von LabVIEW 2014 nicht mehr behoben werden konnten. Die Liste finden Sie auf der Website ni.com/info von National Instruments nach Eingabe des Infocodes `lv2014ki`. Unter *Upgrade - Behavior Change* und *Upgrade - Migration* können Sie sehen, welche Behelfslösungen für die einzelnen Probleme vorgeschlagen werden.
- Dokumentation zu Modulen und Toolkits—Enthält bei manchen Modulen und Toolkits, z. B. beim LabVIEW FPGA und LabVIEW Real-Time Module, Angaben zu Upgrade-Problemen.
- Readme-Dateien für Treiber—Enthalten Listen mit möglichen Upgrade-Problemen. Die Readme-Datei befindet sich auf dem Installationsdatenträger des Treibers.



Tipp Ob Funktionsunterschiede durch das Treiber- oder LabVIEW-Upgrade verursacht wurden, erfahren Sie durch Testen der VIs in Ihrer bisherigen LabVIEW-Version.

- **Ihre eigenen Tests**—Führen Sie in LabVIEW 2014 die gleichen Tests wie in der bisherigen LabVIEW-Version durch. Wenn Ihnen Funktionsunterschiede auffallen, ermitteln Sie anhand der Upgrade-Dokumentation die Ursache dafür.

Fehlersuche beim Upgrade

In der *LabVIEW-Hilfe* finden Sie unter **Upgrade auf LabVIEW 2014»Fehlersuche beim Upgrade** Informationen zum Beheben der folgenden Upgrade-Probleme:

- Suchen nach fehlenden Modul- oder Toolkit-Funktionen
- Suchen nach fehlenden SubVIs, Paletten und Eigenschaftsknoten
- Feststellen, warum sich VIs aus älteren Versionen nicht in LabVIEW 2014 öffnen lassen
- Ermitteln der Versionen installierter Softwareprodukte von National Instruments
- Zurücksetzen von VIs auf eine Vorgängerversion von LabVIEW

Hinweise zur Kompatibilität beim Upgrade

In den folgenden Abschnitten werden versionsspezifische Änderungen an LabVIEW beschrieben, die in bestehenden VIs zu Fehlern oder Funktionsunterschieden führen können.

Bekannte Funktionsprobleme, Kompatibilitätsprobleme sowie kurz vor Veröffentlichung hinzugefügte Funktionen von LabVIEW 2014, die nicht anderweitig dokumentiert sind, werden in der Datei `readme.html` im `labview`-Verzeichnis beschrieben.

Upgrade von LabVIEW 2009 oder älteren Versionen

Informationen zu möglichen Kompatibilitätsproblemen beim Upgrade von LabVIEW 2009 oder älteren Versionen auf LabVIEW 2014 finden Sie auf der Website von National Instruments unter ni.com/info nach Eingabe des Infocodes `oldUpgradeIssues`. Weitere mögliche Upgrade-Probleme sind im vorliegenden Dokument unter *Upgrade von LabVIEW x* beschrieben.

Upgrade von LabVIEW 2010

Beim Upgrade von LabVIEW 2010 auf LabVIEW 2014 können die nachfolgend aufgeführten Kompatibilitätsprobleme auftreten. Sonstige mögliche Probleme beim Upgrade sind in den Abschnitten *Upgrade von LabVIEW 2011*, *Upgrade von LabVIEW 2012* und *Upgrade von LabVIEW 2013* beschrieben.

Änderungen an der Funktionsweise von VIs und Funktionen

In LabVIEW 2011 und neueren LabVIEW-Versionen funktioniert das VI "UDP: Multicast öffnen" nur, wenn am Eingang **Multicast-Adresse** des VIs eine Adresse angegeben ist. Der Anschluss **Port** wurde außerdem in **Port (Ausgang)** umbenannt.

Veraltete VIs, Funktionen und Knoten

In LabVIEW 2011 und neueren Versionen wurde der Eingang **Init/Kont** aus allen polymorphen Instanzen des VIs "Nullphasenfilter" entfernt. Wenn ein VI aus einer älteren LabVIEW-Version Instanzen des VIs "Nullphasenfilter" enthält, müssen Sie diese daher durch gleichnamige VIs von der Filter-Palette ersetzen.

Änderungen der Funktionsweise von Eigenschaften, Methoden und Ereignissen

Die Funktionsweise folgender Eigenschaften, Methoden und Ereignisse hat sich in LabVIEW 2011 und neueren LabVIEW-Versionen geändert:

- In LabVIEW 2010 löscht die Methode "Cache mit kompilierten Objekten löschen" den Objekt-Cache für ein angegebenes Zielsystem. In LabVIEW 2011 und neueren Versionen löscht die Methode hingegen den gesamten Benutzer-Cache für die ausgeführte LabVIEW-Version. In LabVIEW 2010 erstellte VIs, in denen diese Methode verwendet wird, verursachen zwar in LabVIEW 2011 und neueren Versionen keinen Fehler, aber sie löschen mehr VI-Objekt-Dateien als zuvor. Die betroffenen VIs müssen daher beim Laden neu kompiliert werden.
- In LabVIEW 2010 und älteren LabVIEW-Versionen ignoriert das Ereignisdatenfeld **NeuerBereich** des Ereignisses "Änderung des Skalierungsbereichs" benutzerdefinierte Werte für Offset und Faktor für Graphen und Diagramme. Seit LabVIEW 2011 werden benutzerdefinierte Werte für Offset und Faktor vom Ereignisdatenfeld **NeuerBereich** bei der Ausgabe von Ergebnissen berücksichtigt. Wenn Sie das Problem in LabVIEW 2010 mit Hilfe speziell erstellten Programmcodes korrigiert haben, müssen Sie den Code nach dem Umstellen der LabVIEW-Version ändern.

Veraltete Eigenschaften, Methoden und Ereignisse

Die Methode "Auswahl in Subsystem umwandeln" der Klasse "SimDiagramm" wird seit LabVIEW 2011 nicht mehr unterstützt.

Übertragen von Build-Spezifikationen auf Zielsysteme, die SSE2-Anweisungen nicht unterstützen

Um eine Build-Spezifikation für ein Zielsystem, das SSE2-Anweisungen nicht unterstützt, auf LabVIEW-Versionen ab 2011 umzustellen, müssen Sie die SSE2-Optimierung für die Build-Spezifikation deaktivieren. Wenn die Optimierung nicht deaktiviert wird, kann die Anwendung zwar in LabVIEW erstellt, aber nicht auf dem Zielsystem ausgeführt werden.

Welche Gerätetypen SSE2-Anweisungen unterstützen, erfahren Sie in der *LabVIEW-Hilfe* unter dem Thema **Grundlagen»Erstellen und Verteilen von Anwendungen»Konfigurieren von Build-Spezifikationen»SSE2-Unterstützung des Zielsystems** ermitteln.

Polymorphe VI-Anschlüsse, die 64-Bit-Daten und numerische Daten doppelter Genauigkeit unterstützen

Wenn Sie in LabVIEW 2011 und neueren Versionen Daten mit erweiterter Genauigkeit mit einem polymorphen VI verbinden, das sowohl Daten doppelter Genauigkeit als auch 64-Bit-Integer unterstützt, werden die Werte mit erweiterter Genauigkeit auf doppelte Genauigkeit reduziert.

LabVIEW 8.5 und 8.6 funktionieren auf die gleiche Art und Weise. In LabVIEW 8.0, 8.2, 2009 und 2010 wird allerdings statt des Datentyps mit doppelter Genauigkeit der 64-Bit-Integer-Datentyp ausgewählt.

Verbesserte Fehlererkennung bei bestimmten LabVIEW-DLLs

Wenn ein "Knoten zum Aufruf externer Bibliotheken" in älteren LabVIEW-Versionen eine LabVIEW-DLL aufruft und auf dem Zielsystem nicht alle erforderlichen Ressourcen installiert sind, kann die DLL nicht ausgeführt werden. Eine entsprechende Fehler- bzw. Rückmeldung wird jedoch nicht ausgegeben. In LabVIEW 2011 und neueren Versionen wird bei Aufruf von DLLs ein Fehler ausgegeben, wenn auf dem Zielsystem Ressourcen fehlen. In LabVIEW 2011 und neueren Versionen können daher Fehlermeldungen zu DLLs ausgegeben werden, die in vorherigen Versionen fälschlicherweise ausblieben.

Die verbesserte Fehlererkennung gilt unter anderem für VIs, die LabVIEW-DLLs mit folgenden Merkmalen aufrufen:

- Ein VI in der DLL arbeitet mit lizenzierten Funktionen, die auf dem Zielcomputer nicht installiert sind.
- Ein VI in der DLL arbeitet mit einem Knoten zum Aufruf externer Bibliotheken, der sich auf eine DLL bezieht, die auf dem Zielcomputer nicht installiert ist.
- Die VIs in der DLL wurden mit SSE2-Optimierung kompiliert, aber der Zielcomputer unterstützt keine SSE2-Anweisungen.

Änderungen an den Suchverzeichnissen für Daten der NI-Suchmaschine für Beispiele

In LabVIEW 2011 und neueren Versionen werden im Vergleich zu älteren LabVIEW-Versionen weniger Verzeichnisse nach Dateien der NI-Suchmaschine für Beispiele (*.bin3) durchsucht. Damit LabVIEW die für die NI-Suchmaschine für Beispiele erstellten Beispiel-VIs findet, müssen Sie *.bin3-Dateien in einem der folgenden Verzeichnisse speichern:

- `labview\examples\exbins`—Bei älteren LabVIEW-Versionen konnten Sie die *.bin3-Dateien an einer beliebigen Stelle im Verzeichnis `examples` speichern.
- `labview\instr.lib`
- `labview\user.lib`

Kompatibilitätsprobleme zwischen LabVIEW 2011 und anderer Software von National Instruments

In LabVIEW 2011 muss NI Spy 2.3 (oder eine neuere Version) oder NI I/O Trace 3.0 verwendet werden. NI Spy wurde nach NI Spy 2.7.2. in "NI I/O Trace" umbenannt. NI I/O Trace befindet sich auf dem Datenträger mit den NI-Gerätetreibern.

LabVIEW 2011 unterstützt Measurement Studio 8.0 und neuere Versionen. Wenn Sie auf ni.com/info den Infocode `exd8yy` eingeben, gelangen Sie zum Upgrade Advisor, über den Sie Measurement Studio 8.0 oder eine neuere Version von Measurement Studio erwerben können.

Upgrade von LabVIEW 2011

Beim Upgrade von LabVIEW 2011 auf LabVIEW 2014 können die nachfolgend beschriebenen Kompatibilitätsprobleme auftreten. Sonstige mögliche Probleme beim LabVIEW-Upgrade sind in den Abschnitten *Upgrade von LabVIEW 2012* und *Upgrade von LabVIEW 2013* beschrieben.

Austausch serialisierter Daten zwischen verschiedenen LabVIEW-Versionen

In LabVIEW 2011 und älteren Versionen werden Daten zwischen LabVIEW-Versionen mit Hilfe der Funktionen "Daten serialisieren" und "String deserialisieren" ausgetauscht. In LabVIEW 2012 muss zum Übertragen der Daten das VI "VariantFlattenExp.vi" im Verzeichnis `labview\vi.lib\Utility`

verwendet werden. Das VI "VariantFlattenExp.vi" nimmt Hexadezimal-Integer der LabVIEW-Version an, in welche die Daten übertragen werden sollen.

Veraltete VIs, Funktionen und Knoten

Folgende VIs, Funktionen und Knoten werden von LabVIEW 2012 und neueren LabVIEW-Versionen nicht mehr unterstützt:

- **Polarplot**—Verwenden Sie stattdessen das VI "Polarplot mit Punktoptionen". Das VI wurde außerdem um die Eingänge **Linien/Punkte** und **Größe** erweitert.
- **Rechteck zeichnen**—Verwenden Sie stattdessen das VI "**Rechteck zeichnen**".

Änderungen der Funktionsweise von Eigenschaften, Methoden und Ereignissen

Bei der Methode "Zellenwert setzen" der Klasse "Tabelle" hat sich die Zahlendarstellung von **X-Index** und **Y-Index** von "vorzeichenloser 32-Bit-Integer" in "vorzeichenbehafteter 32-Bit-Integer" geändert.

Veraltete Eigenschaften, Methoden und Ereignisse

Die folgenden Eigenschaften, Methoden und Ereignisse werden von LabVIEW 2012 oder neueren Versionen nicht unterstützt:

- Die Methode "Von Datentyp erstellen" der Klasse "Diagramm". Wenn Sie ein VI auf die aktuelle LabVIEW-Version umstellen, das diese Methode enthält, ruft das VI die Methode "Aus Datentyp erstellen (veraltet)" auf. Ersetzen Sie die veraltete Methode durch die neue Methode "Aus Datentyp erstellen", die nicht mehr den Eingang **Darstellung** enthält.
- Die Eigenschaft "Rahmen[]" der Klasse "ZeitFlachSequenz". Verwenden Sie stattdessen die Eigenschaft "Rahmen[]" der Klasse "FlacheSequenz".
- Die Eigenschaft "Frontpanel-Fenster:Öffnen" der "VI"-Klasse. Verwenden Sie stattdessen die Methoden "Frontpanel:Öffnen" und "Frontpanel:Schließen" oder die Eigenschaft "Frontpanel-Fenster:Status".
- Die Eigenschaft "FPWinOpen" der Klasse "VI" für ActiveX. Verwenden Sie stattdessen die Methoden "OpenFrontPanel" und "CloseFrontPanel" oder die Eigenschaft "FPState".
- Die Eigenschaft "Statische Member-VIs" der Klasse "LV-Klassenbibliothek". Verwenden Sie stattdessen die Eigenschaft "Statische Member-VIs[]".
- Die Eigenschaft "Dynamische Member-VIs" der Klasse "LV-Klassenbibliothek". Verwenden Sie stattdessen die Eigenschaft "Dynamische Member-VIs[]".

Umbenannte Eigenschaften, Methoden und Ereignisse

Folgende Eigenschaften, Methoden und Ereignisse wurden in LabVIEW 2012 und neuer umbenannt:

Klasse	Name in LabVIEW 2011	Name in LabVIEW 2012 und neuer	Typ
Projektobjekt	Unterobjekte[]	Zugehörige Objekte[]	Eigenschaft
Projektobjekt	Elternobjekt	Eigentümer	Eigenschaft
LV-Klassenbibliothek	VorgängerElementeRef	Vorgänger beschränkt Referenzerstellung	Eigenschaft

Upgrade von LabVIEW 2012

Beim Upgrade von LabVIEW 2012 auf LabVIEW 2014 können die nachfolgend beschriebenen Kompatibilitätsprobleme auftreten. Sonstige mögliche Probleme beim LabVIEW-Upgrade sind im Abschnitt *Upgrade von LabVIEW 2013* beschrieben.

Änderungen an der Funktionsweise von VIs und Funktionen

Die Funktionsweise der nachfolgenden VIs und Funktionen hat sich in LabVIEW 2013 geändert.

Webdienst-VIs

Die folgenden VIs auf der Palette "Webdienste" wurden in LabVIEW 2013 umgeschrieben. In diesen VIs wurde der Eingang **httpAnfrageID** durch den Eingang **LabVIEW-Webdienst-Anfrage** ersetzt. Um die neue Funktion zu verwenden, ersetzen Sie jeweils das alte VI durch das gleichnamige neue VI aus der Palette "Webdienste".

- Palette "Webdienste":
 - "Alle Formular Daten lesen"
 - "Alle Abfragevariablen lesen"
 - "Formular Daten lesen"
 - "POST-Daten lesen"
 - "Anfragevariable lesen"
 - "Info zu hochgeladenen Dateien lesen"
- Unterpalette "Ausgabe":
 - "Ausgabe löschen"
 - "ESP-Vorlage rendern"
 - "ESP-Variable festlegen"
 - "HTTP-Header festlegen"
 - "HTTP-Umleitung festlegen"
 - "HTTP-Antwort-Code festlegen"
 - "HTTP-Antwort-MIME-Typ festlegen"
 - "Antwort schreiben"
- Unterpalette "Sicherheit":
 - "Entschlüsseln"
 - "Verschlüsseln"
 - "Auth.-Details ermitteln"
- Unterpalette "Sessions":
 - "Existenz der Session prüfen"
 - "Session erstellen"
 - "Session-Variable löschen"
 - "Session entfernen"
 - "Session-ID-Cookie ermitteln"
 - "Alle Session-Variablen lesen"
 - "Session-Variablen lesen"
 - "Session-Variablen schreiben"

Änderungen am Verhalten des Timeout-Anschlusses der Ereignisstruktur für nicht verarbeitete dynamisch registrierte Ereignisse

Bei der dynamischen Ereignisregistrierung in LabVIEW 2012 und älteren Versionen kann beim Auftreten von Ereignissen, für deren Verarbeitung die Ereignisstruktur nicht konfiguriert ist, der Timeout-Anschluss zurückgesetzt werden. Wenn Sie beispielsweise die Funktion "Ereignisregistrierung" zur Registrierung für die Ereignisse "Maustaste losgelassen", "Maustaste gedrückt" und "Mausbewegung" verwenden, die

Ereignisstruktur jedoch nur für die Verarbeitung von "Maustaste losgelassen" und "Maustaste gedrückt" konfigurieren, wird der Timeout-Anschluss zurückgesetzt, wenn das Ereignis "Mausbewegung" stattfindet.



Hinweis Der Timeout-Anschluss wird nur zurückgesetzt, wenn Sie einen Wert mit diesem Anschluss verbinden.

In LabVIEW 2013 hingegen wird der Timeout-Anschluss der Ereignisstruktur *nicht* zurückgesetzt, wenn ein nicht verarbeitetes dynamisch registriertes Ereignis eintritt.

Änderungen am Standard-.NET-Framework

Zum Erstellen von und Kommunizieren mit .NET-Objekten in LabVIEW 2013 ist .NET Framework 4.0 erforderlich. Mit dem .NET Framework 4.0 können Sie sowohl reine verwaltete Assemblys, die in einer beliebigen Version von .NET Framework erstellt wurden, als auch in .NET 4.0 erstellte Assemblys im gemischten Modus laden. Im Installer für LabVIEW 2013 ist .NET Framework 4.0 enthalten. Wenn Sie .NET Framework 4.0. deinstallieren oder Assemblys aus einer anderen Version von .NET Framework laden, wird möglicherweise beim Erstellen von .NET-Objekten oder bei der Kommunikation mit .NET-Objekten eine Fehlermeldung ausgegeben.

In LabVIEW 2013 wird standardmäßig die Common Language Runtime (CLR) 4.0 verwendet. Sie können jedoch LabVIEW zwingen, Assemblys im gemischten Modus zu laden, die mit CLR 2.0 arbeiten.

Mehr Informationen zum Laden von Assemblys in LabVIEW finden Sie im Thema **Grundlagen»Windows-Konnektivität»Anleitung»NET»Laden von .NET 2.0, 3.0 und 3.5 Assemblies in LabVIEW** der *LabVIEW-Hilfe*.

Änderungen an der Systemschaltfläche

Wenn in LabVIEW 2012 und älteren Versionen die Systemschaltfläche aus der **System**-Palette auf dem Frontpanel abgelegt wird, wird das Element per Voreinstellung durch die Eingabetaste umgeschaltet. In LabVIEW 2013 gibt es keine Standardtaste für die Systemschaltfläche mehr.

Änderungen an den Eigenschaften "Wert" und "Wert (signalisierend)"

Beim Festlegen des Werts eines booleschen Elements mit Latch-Funktion mit der Eigenschaft "Wert" oder "Wert (signalisierend)" wird in LabVIEW 2012 und älteren LabVIEW-Versionen eine Fehlermeldung ausgegeben. Wenn Sie jedoch das boolesche Element mit Latch-Funktion in eine Typdefinition umwandeln, erscheint keine Fehlermeldung. Um in LabVIEW 2013 Laufzeitprobleme zu vermeiden, wird beim Festlegen des Werts eines booleschen Elements mit Latch-Funktion mit Hilfe der Eigenschaften "Wert" oder "Wert (signalisierend)" immer eine Fehlermeldung ausgegeben.

Leistungsverbesserungen an bedingten Tunneln

In LabVIEW 2012 können Sie mit Hilfe der Option **Bedingt** für den Ausgangstunnel einer Schleife festlegen, dass nur bestimmte Werte am Ausgangstunnel einer Schleife ausgegeben werden sollen. Für Abschnitte, bei denen die Ausführungsgeschwindigkeit von Bedeutung ist, wird jedoch empfohlen, mit Alternativen zu dieser Option zu arbeiten. In LabVIEW 2013 wurde die Ausführungsgeschwindigkeit von bedingten Tunneln durch effizientere Speichernutzung erhöht.

Verbinden von benutzerdefinierten Elementen mit einem Unterpanel

Ab LabVIEW 2013 kommt es zu einem Fehler, wenn Sie die Methode "VI einfügen" der Klasse "Unterpanel" mit einem benutzerdefinierten Element verbinden. Um ein benutzerdefiniertes Element mit einem Unterpanel zu verbinden, müssen Sie das Element auf dem Frontpanel eines VIs ablegen und das VI mit dem Unterpanel verbinden.

Verwenden des Utility "NI-Webkonfiguration und Überwachung" mit SSL

In LabVIEW 2012 und älteren Versionen konnten Zertifikate zur SSL-Verschlüsselung (Secure Sockets Layer) und Registrierungsanforderungen über den NI Distributed System Manager eingesehen und bearbeitet werden. Der NI Distributed System Manager unterstützt diese Funktion nicht mehr.

SSL-Zertifikate und CSRs sind jetzt im Utility "NI-Webkonfiguration und Überwachung" einsehbar. Hier können Sie auch CSRs erstellen, bearbeiten und entfernen. Wählen Sie zum Verwalten Ihrer SSL-Zertifikate und CSRs im Utility "NI-Webkonfiguration und Überwachung" die Seite "Webserver-Konfiguration" aus und öffnen Sie die Registerkarte "Verwaltung von SSL-Zertifikaten".

Erstellen und Veröffentlichen von LabVIEW-Webdiensten

In LabVIEW 2013 werden zum Erstellen von Webdiensten und zum Konfigurieren von Eigenschaften für Webdienste (zum Beispiel URL-Mapping) keine Build-Spezifikationen für REST-konforme Webdienste mehr benötigt. Sie können die in LabVIEW 2012 oder in älteren Versionen erstellten Build-Spezifikationen entweder weiterhin verwenden oder in Webdienst-Projektobjekte umwandeln. Das dafür benötigte Umwandlungsprogramm erhalten Sie, indem Sie auf ni.com/info den Infocode `ConvertWS` eingeben.

Um beim Konvertieren eines Webdienstes in das LabVIEW-2013-Format auf einen Großteil der Konfigurationsoptionen für Webdienst-Build-Spezifikationen in LabVIEW 2012 und älteren Versionen zuzugreifen, klicken Sie mit der rechten Maustaste auf das Webdienst-Projekt und wählen Sie **Eigenschaften** aus. Welche Funktionen und Optionen für Webdienste von LabVIEW 2012 und älteren Versionen in LabVIEW 2013 geändert oder entfernt wurden, erfahren Sie in der nachfolgenden Tabelle.

LabVIEW 2012 und älter	LabVIEW 2013
Der Terminus <i>Webmethoden-VI</i> bezieht sich auf VIs, die HTTP-Anfragen von Clients erhalten und Daten an Clients zurücksenden.	Die Webmethoden-VIs wurden in <i>HTTP-Methoden-VIs</i> umbenannt.
Sie können zum Anpassen der URL, die von Clients für den Zugriff auf den Webdienst verwendet wird, einen Dienst-Aliasnamen für den Webdienst festlegen.	Für den Zugriff auf den Webdienst wird der genaue Name des Dienstes benötigt.
Einem Webmethoden-VI können mehrere URLs zugeordnet werden.	Einem HTTP-Methoden-VI kann nur eine URL zugeordnet werden. Um ein VI über mehrere URLs aufrufen zu können, verwenden Sie das VI als SubVI in verschiedenen HTTP-Methoden-VIs mit jeweils eigenem URL-Mapping.
Sie können neue Werte bestimmen, um die Standardwerte der Anschlussfelder des VIs zu überschreiben.	Diese Option wurde entfernt, da einem HTTP-Methoden-VI nicht mehrere URLs zugeordnet werden können. Es ist daher nicht möglich, durch Überschreiben alternative URL-Mappings zu erstellen.
Sie können VIs in einem Projekt als Hilfs-VIs designieren. Hilfs-VIs können Daten mit Webmethoden-VIs austauschen, jedoch nicht mit Clients.	Die Hilfs-VIs wurden in <i>Start-VIs</i> umbenannt. Alle VIs, die sich im Projektelement Start-VIs des Projekts befinden, werden von LabVIEW als Start-VIs betrachtet.
Sie können die "Stand-Alone"-Verteilung für einen Webdienst deaktivieren, was bedeutet, dass der Webdienst nur bei geöffnetem LabVIEW Development System veröffentlicht wird.	Diese Option wurde entfernt.
VIs können so eingestellt werden, dass sie als Schritte vor und nach dem Erstellen des Webdienstes ausgeführt werden.	Diese Funktion ist nicht verfügbar, da Webdienste nicht basierend auf Build-Spezifikationen erstellt werden.

Upgrade von LabVIEW 2013

Geänderte Funktionsweise der Funktion "String nach Pfad"

In LabVIEW 2014 gibt die Funktion "String nach Pfad" ungeachtet der Groß- oder Kleinschreibung von `<Kein Pfad>` immer die Konstante `<Kein Pfad>` aus. Sie können also `<kein Pfad>` oder `<Kein`

`Pfad`> am **String**-Eingang angeben. Die Arbeitsweise der Funktion "String nach Pfad" in älteren LabVIEW-Versionen ist in der folgenden Tabelle beschrieben.

LabVIEW 2012 und 2013	LabVIEW 2011 und älter
Die Funktion "String nach Pfad" gibt keine Konstante des Typs <Kein Pfad> aus. Wenn Sie eine Variation des Strings <Kein Pfad> angeben, gibt die Funktion einen Pfad zu einem Verzeichnis namens <Kein Pfad> aus.	Die Funktion "String nach Pfad" ignoriert wie in LabVIEW 2014 die Groß- und Kleinschreibung des Strings <Kein Pfad> und gibt immer die Konstante <Kein Pfad> aus. Unabhängig davon, ob Sie <kein Pfad> oder <Kein Pfad> angeben, gibt die Funktion immer die Konstante <Kein Pfad> aus.

Vergleichen und Aktualisieren von Typdefinitionen

Der Kontextmenüpunkt **Mit Typdefinition vergleichen und aktualisieren** ersetzt den Menüpunkt **Aktualisierung von Typdefinition** in LabVIEW 2013 und früheren Versionen.

Veraltete VIs, Funktionen und Knoten

Folgende VIs, Funktionen und Knoten werden von LabVIEW 2014 nicht unterstützt:

AppleEvent-VIs

LabVIEW 2014 für Mac OS X unterstützt die AppleEvent-VIs nicht mehr. Verwenden Sie für den Datenaustausch mit LabVIEW-externen OS-X-Anwendungen stattdessen das VI "Run AppleScript Code" auf der Palette Libraries & Executables. Wenn Sie in in LabVIEW 2014 für Max OS X ein VI laden, das eines der folgenden AppleEvent-VIs enthält, können Fehler auftreten, und das VI kann evtl. nicht ausgeführt werden:

- AESend Do Script
- AESend Finder Open
- AESend Open
- AESend Open Document
- AESend Print Document
- AESend Quit Application
- Get Target ID
- AESend Abort
- AESend Close
- AESend Open, Run, Close
- AESend Run
- AESend VI Active?
- AECreat Comp Descriptor
- AECreat Descriptor List
- AECreat Logical Descriptor
- AECreat Object Specifier
- AECreat Range Descriptor
- AECreat Record
- AESend

- Make Alias

Akteur-Framework-VIs

LabVIEW 2014 unterstützt das VI "Akteur:Akteur starten" nicht. Verwenden Sie stattdessen das VI "Akteur:Hauptakteur starten" oder das VI "Akteur:Unterakteur starten".


VIs "Eingangs-Port" und "Ausgangs-Port"

Die VIs "Eingangs-Port" und "Ausgangs-Port" werden in LabVIEW 2014 nicht unterstützt.

Veraltete Eigenschaften, Methoden und Ereignisse

LabVIEW 2014 unterstützt die Methode "VI ermitteln:Hilfe-Info (alt)" in der Klasse "Anwendung" nicht. Verwenden Sie stattdessen die Methode "VI ermitteln:Hilfe-Info", um für ein bestimmtes VI Hilfe-Informationen von der Seite **Dokumentation** des Dialogfelds **Eigenschaften für VI** zu erhalten.

Änderungen und Neuerungen in LabVIEW 2014

Mit dem Symbol  gekennzeichnete neue Funktionen stammen von Vorschlägen aus dem LabVIEW Idea Exchange. Zum NI Idea Exchange gelangen Sie über ni.com/info nach Eingabe des Infocodes `ex3gus`.

Bekannte Funktionsprobleme, Kompatibilitätsprobleme, Fehlerkorrekturen sowie kurz vor Veröffentlichung hinzugefügte Funktionen von LabVIEW 2014 werden in der Datei `readme.html` im `labview`-Verzeichnis beschrieben.

LabVIEW-Taskleistensymbole mit Versionsangabe und Bitanzahl

 Neue Taskleistensymbole zeigen die Version und Bitanzahl der installierten LabVIEW-Software an.

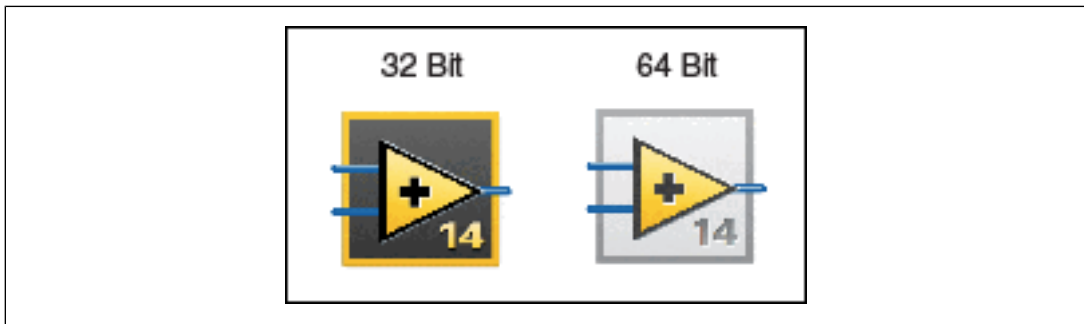


Figure 2.

Durch diese neuen Symbole können Sie die auf Ihrem Computer installierten LabVIEW-Versionen leichter voneinander unterscheiden. Wenn Sie beispielsweise LabVIEW 8.6 und LabVIEW 2014 installiert haben, sind somit Verwechslungen weniger wahrscheinlich. Durch die in den Taskleistensymbolen angezeigte Bitanzahl können Sie außerdem leicht die 32- und die 64-Bit-Version von LabVIEW 2014 auseinanderhalten.



Hinweis Versionspezifische Taskleistensymbole sind allerdings nur unter Windows und OS X verfügbar.

[Idee wurde vom NI-Diskussionsforen-Mitglied "Darren" eingereicht]

Installieren und Aktualisieren von DataPlugins in LabVIEW

In älteren LabVIEW-Version DataPlugins mussten DataPlugins zunächst von ni.com/dataplugins heruntergeladen und manuell installiert werden. In LabVIEW 2014 können Sie direkt mit Hilfe des VIs "Datenspeicher/DataPlugin" nach DataPlugins suchen und diese installieren und aktualisieren.

Verbesserungen am Blockdiagramm

LabVIEW 2014 enthält die nachfolgend aufgeführten Verbesserungen am Blockdiagramm und dem damit verbundenen Funktionsumfang.

Ersetzen eines Tunnels durch einen Case-Selektor

☞ Sie können in LabVIEW 2014 einen Eingangstunnel an einer Case-Struktur durch einen Case-Selektor für die Struktur ersetzen. Wenn Sie den Tunnel mit der rechten Maustaste anklicken und aus dem Kontextmenü **Durch Case-Selektor ersetzen** auswählen, wird der Tunnel in einen Case-Selektor umgewandelt. Durch eine Änderung der Eingangsdaten am Case-Selektor kann sich die Auswahl der Cases ändern, die in die Case-Selektorbeschriftung eingegeben werden können. Wenn Sie einen Tunnel durch einen Case-Selektor ersetzen, wird gleichzeitig der ursprüngliche Case-Selektor in einen Tunnel umgewandelt.

Durch Case-Selektor ersetzen

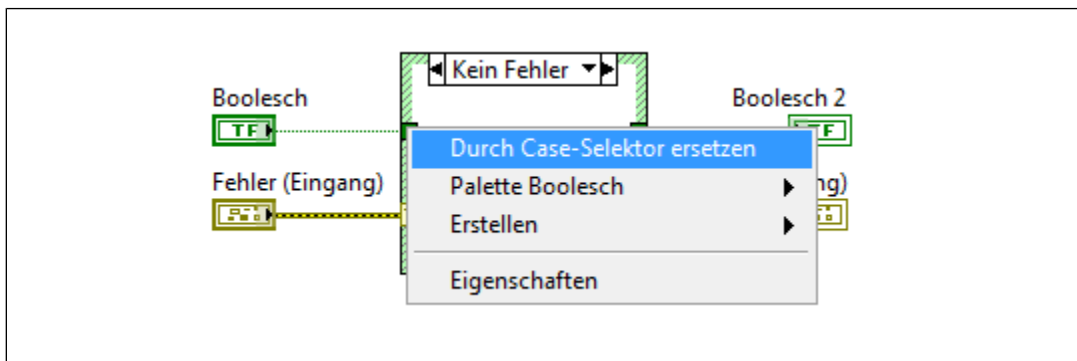


Figure 3.

Nach "Durch Case-Selektor ersetzen"

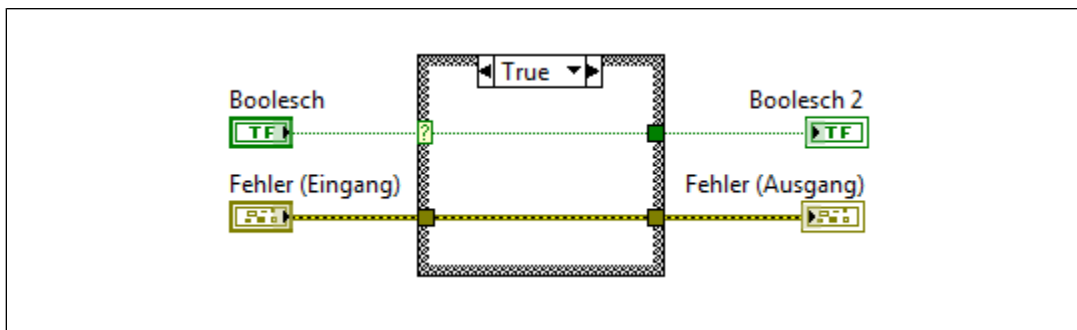


Figure 4.

Sie können einen Tunnel mit der Methode "Durch Case-Selektor ersetzen" auch programmatisch in einen Case-Selektor umwandeln.

[Idee wurde vom NI-Diskussionsforen-Mitglied "NelsonUWP" eingereicht]

Automatisches Verbinden von Objekten in einem VI

In LabVIEW 2014 gibt es **Schnelleinfügetasten** zum einfachen Verbinden mehrerer Blockdiagrammobjekte. Sie können eine Objektreihe oder mehrere parallele Reihen markieren und **<Strg + Leertaste>** drücken, um die Schnelleinfügeliste zu öffnen. Sobald das Dialogfeld **Schnelleinfügeliste** erscheint, drücken Sie **<Strg + W>**. Die markierten Objekte werden automatisch miteinander verbunden. Außerdem können Sie mit **<Strg + Shift + W>** eine Reihe oder mehrere parallele Reihen von Blockdiagrammobjekten verbinden und markierten Programmcode "aufräumen".

Weitere Informationen zu Schnelleinfügetasten finden Sie in der *LabVIEW-Hilfe* unter **Grundlagen»LabVIEW-Umgebung»Anleitung»Tastenkombinationen»Tastenkombinationen für die Schnelleinfügeliste**.

Verbesserungen am Frontpanel

LabVIEW 2014 enthält die nachfolgend aufgeführten Verbesserungen am Frontpanel und dem damit verbundenen Funktionsumfang.

Verwenden von Elementen der Palette "Silber"

In LabVIEW 2014 können Sie über die neue Unterpalette "Gestaltungselemente" leicht auf Gestaltungselemente im Silber-Design zugreifen.

Verbesserungen an der Programmoberfläche

In LabVIEW 2014 gab es folgende Verbesserungen an der Programmoberfläche:

Informationen zu Fehlermeldungen über fehlende SubVIs

In LabVIEW 2014 können Sie den Verbleib fehlender SubVIs über das Fenster **Fehlerliste** oder das Fenster **Kontexthilfe** herausfinden. In älteren Versionen von LabVIEW zeigt das Fenster **Fehlerliste** zwar an, dass ein SubVI fehlt, aber es hilft dem Benutzer nicht beim Auffinden des VIs. In LabVIEW 2014 werden im Fenster **Fehlerliste** der Treiber, das Toolkit oder Modul angezeigt, die das fehlende SubVI enthalten, und es werden Informationen zur Fehlerbehebung zur Verfügung gestellt. Außerdem ist der Pfad zum fehlenden SubVI in der **Kontexthilfe** aufgelistet.

Vergleichen und Aktualisieren von Instanzen von Typdefinitionen

Wenn in LabVIEW 2013 und älteren Versionen Änderungen von einer Typdefinition auf Instanzen dieser Typdefinition übertragen werden, kann es vorkommen, dass die Standardwerte für die zu aktualisierenden Instanzen verloren gehen oder fälschlicherweise beibehalten werden.

In den meisten Fällen behält LabVIEW beim Aktualisieren von der Typdefinition automatisch die Standardwerte jeder Instanz bei. Wenn eine Instanz nicht automatisch aktualisiert werden kann, wird sie in einen ungelösten Zustand versetzt, bis sie über das Dialogfeld **Mit Typdefinition vergleichen und aktualisieren** manuell aktualisiert wird.

Sie können das Dialogfeld **Mit Typdefinition vergleichen und aktualisieren** öffnen, indem Sie mit der rechten Maustaste auf eine in ungelöstem Zustand befindliche Instanz klicken und **Mit Typdefinition vergleichen und aktualisieren** auswählen. Der Kontextmenüpunkt **Mit Typdefinition vergleichen und aktualisieren** ist in LabVIEW 2014 an die Stelle des bisherigen Menüpunkts **Aktualisierung von Typdefinition** getreten.

Weitere Informationen zum manuellen Aktualisieren von Instanzen von Typdefinitionen finden Sie im Buch **Grundlagen»Erstellen des Frontpanels»Anleitung»Erstellen von benutzerdefinierten Bedien- und Anzeigeelementen und Typdefinitionen** auf der Registerkarte **Inhalt** in der *LabVIEW-Hilfe*.

Verbesserungen an Dialogfeldern

In LabVIEW 2014 wurden die nachfolgend aufgeführten Verbesserungen an Dialogfeldern vorgenommen.

Informationen zu Fehlermeldungen über die Fehlerdialoge

In LabVIEW 2014 erhalten Sie über das Dialogfeld **Fehler beschreiben** und das Dialogfeld **Einfacher Fehlerbehandler** Informationen zum Beheben des angezeigten Fehlers. Durch Klicken auf den Hyperlink **Auf ni.com recherchieren** wird das Ergebnis einer Fehlersuche auf ni.com im Standard-Webbrowser angezeigt. Der Hyperlink erscheint standardmäßig auch im Dialogfeld des VIs **Allgemeiner Fehlerbehandler**. Der Hyperlink wird nur in der Entwicklungsumgebung angezeigt.



Figure 5.

Neue und geänderte VIs, Funktionen und Knoten

Neue VIs, Funktionen und Knoten

LabVIEW 2014 umfasst folgende neue VIs, Funktionen und Knoten:

Programmatisches Ermitteln oder Festlegen von Versionsangaben für eine Build-Spezifikation

Die Palette "Application Builder" enthält folgende neue VIs:

- "Version der Build-Spezifikation ermitteln"
- "Version der Build-Spezifikation festlegen"

Verwenden Sie diese VIs zum programmatischen Ermitteln oder Festlegen von Versionsangaben für Build-Spezifikationen mit Versionsnummern.

Kommunikation mit externen Anwendungen - LabVIEW für OS X

LabVIEW 2014 für OS X enthält die folgenden Paletten mit neuen VIs für den Datenaustausch mit LabVIEW-externen OS-X-Anwendungen:

- In der Palette "Libraries & Executables" befindet sich das VI "Run AppleScript Code". Dieses VI ermöglicht den Datenaustausch zwischen LabVIEW und externen OS X-Anwendungen. In LabVIEW 2013 und älteren Versionen fand der Datenaustausch mit externen OS-X-Anwendungen über die AppleEvent-VIs statt. In LabVIEW 2014 können Sie mit Hilfe des VIs "Run AppleScript Code" mit LabVIEW-externen OS-X-Anwendungen kommunizieren.
- Die String-Palette enthält das VI "Normalize End Of Line". Dieses VI konvertiert das Zeilenende des angegebenen Strings in das angegebene Zeilenendformat. Wenn Sie kein Format für das Zeilenende festlegen, wandelt das VI die Zeilenenden des Strings in das Zeilenende um, das von

der Befehlszeile der aktuellen Plattform erwartet wird. Mit diesem VI können Sie Strings für verschiedene Plattformen oder für die Kommandozeile der aktuellen Plattform lesbar machen.

- Die Palette "Path/Array/String Conversion" enthält die folgenden VIs:
 - **"Path to Command Line String"**—Wandelt einen Pfad in einen String mit der Pfadangabe um. Dieses VI formatiert den Pfad-String mit dem Pfad im Standardformat der Kommandozeile der aktuellen Plattform. Sie können mit diesem VI Pfade formatieren, bevor Sie Befehle an die Kommandozeile der aktuellen Plattform senden.
 - **"Command Line String to Path"**—Wandelt den angegebenen String in einen Pfad um. Der Eingangs-String muss den Pfad im Standardformat der Kommandozeile der aktuellen Plattform angeben. Formatieren Sie mit Hilfe dieses VIs Pfade zur Verwendung mit LabVIEW, die Sie von der Kommandozeile der verwendeten Plattform erhalten.

Akteur-Framework-VIs

In LabVIEW 2014 sind die folgenden neuen VIs auf der Palette "Akteur-Framework" enthalten:

- **"Akteur:Hauptakteur starten"**—Startet ein asynchron ausgeführtes VI, das für den Akteur Aufgaben ausführt und Nachrichten bearbeitet. Dieses VI gibt eine Referenz auf einen Enqueueer aus, über den Nachrichten an den neu gestarteten Akteur gesendet werden können.
- **"Akteur:Unterakteur starten"**—Startet ein asynchron ausgeführtes VI, das für den Unterakteur Aufgaben ausführt und Nachrichten bearbeitet. Mit Hilfe dieses VIs lassen sich Unterakteure starten. Dieses VI gibt eine Referenz auf den Enqueueer aus, über den Nachrichten an den neu gestarteten Akteur gesendet werden können.
- **"Akteur: Nachricht 'Unterakteur starten' senden"**—Sendet eine Nachricht, die einen Akteur enthält, an einen anderen Akteur. Der Akteur, der die Nachricht empfängt, startet dann den Nutzdatenakteur als Unterakteur. Dieses VI hat den Zweck, dass Akteure sich selbst Nachrichten zusenden können, und sollte nur dafür verwendet werden.

Geänderte VIs und Funktionen

Die folgenden VIs und Funktionen wurden in LabVIEW 2014 geändert

Datenspeicher/DataPlugin-VIs

- **"Datenspeicher öffnen"**—Die Schaltfläche **Weitere DataPlugins von ni.com/dataplugins herunterladen** im Konfigurationsdialogfeld dieses Express-VIs wurde durch die Schaltfläche **DataPlugins installieren/aktualisieren** ersetzt. Durch Anklicken von **DataPlugins installieren/aktualisieren** öffnet sich das Dialogfeld **DataPlugins installieren/aktualisieren**, über das Sie DataPlugins auf ni.com/dataplugins suchen und von dort aus installieren oder aktualisieren können.
- **"DataPlugins auflisten"**—Dieses VI enthält den neuen Eingang **Quelle**, mit dessen Hilfe angegeben wird, ob LabVIEW DataPlugins vom lokalen Computer oder von der Website von National Instruments auflisten soll.
- **"DataPlugin registrieren"**—Dieses VI enthält die neue Instanz "DataPlugin nach Namen installieren". Sie können mit Hilfe dieser Instanz DataPlugins von ni.com/dataplugins auf dem lokalen Computer installieren.

Allgemeine Änderungen an VIs und Funktionen

In LabVIEW 2014 gibt es folgende Änderungen an VIs und Funktionen:

- **"Klassenhierarchie mittels Klassennamen ermitteln"**—Gibt ein Array mit Klassennamen in absteigender Ordnung unter Verwendung eines Klassennamens aus, der von "Allgemein" abgeleitet ist. Wenn der Klassenname beispielsweise "WhileLoop" lautet, gibt dieses VI das folgende Array aus:

```
[Generic, GObject, Node, Structure, Loop, WhileLoop]
```

- **"Hohe Auflösung - Relative Sekunden"**—Gibt den Wert des Timers aus. Das VI ähnelt der Funktion "Timer-Wert (ms)", jedoch mit dem Unterschied, dass dieses VI einen Zeitstempel mit wesentlich höherer Auflösung bietet. "Hohe Auflösung - Relative Sekunden" eignet sich zur hochpräzisen Leistungsbemessung von Code.
- **"Ist Pfad und nicht leer?"**—Das VI gibt TRUE aus, wenn **Pfad** weder leer ist, noch <Kein Pfad> lautet. Ansonsten wird FALSE ausgegeben.
- **Variantekonstante**—Mit Hilfe dieser Konstante wird ein leerer Variant an das Blockdiagramm übergeben. Bei Verwendung dieser Konstante legt LabVIEW immer einen leeren Variant ab. Der Wert des Variants kann nicht bearbeitet werden.
- **"Fehlermeldungsnachricht:Fehlerprotokoll senden"**—Mit Hilfe dieses VIs wird ein Fehler an einen Akteur gesendet. Der Fehler wird dann vom Fehlerbehandlungs-VI des Akteurs verarbeitet. Wenn der Fehler dort nicht verarbeitet wird, funktioniert der Akteur nicht mehr.
- **"TDMS: Eigenschaften festlegen"**—Diese Funktion enthält Verbesserungen an der Eigenschaft `NI_MinimumBufferSize`. Die minimale Puffergröße kann für eine *.tdms-Datei jetzt auf Gruppen- oder Dateiebene festgelegt werden.
- **"Fehler aufheben"**—Dieses VI enthält den neuen Eingang **Aufzuhebender Fehlercode**, an dem Sie einen spezifischen Fehler angeben können, der ignoriert werden soll. Über den neuen Ausgang **Spezifischer Fehler aufgehoben?** wird angezeigt, ob der unter **Aufzuhebender Fehlercode** angegebene Fehler erfolgreich aufgehoben wurde.

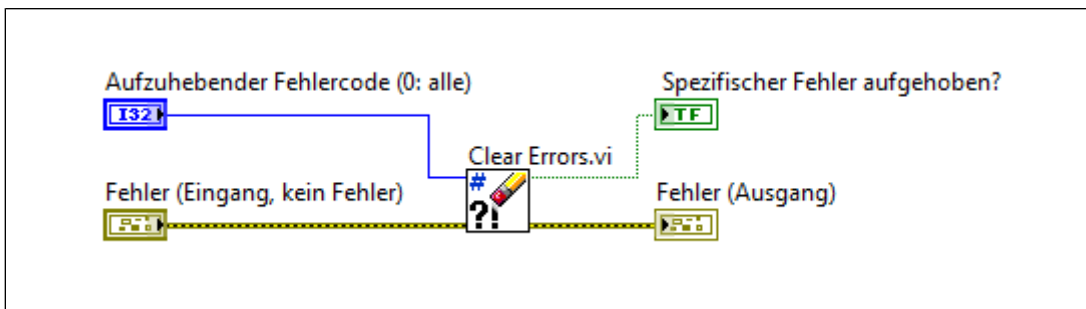


Figure 6.

Verbesserungen am Application Builder

In LabVIEW 2014 gab es folgende Verbesserungen am LabVIEW Application Builder und an den Build-Spezifikationen:

Übertragen von Installern auf Windows-Embedded-Standard-Zielsysteme

Zum Übertragen eines Installers auf ein Zielsystem, das mit Windows Embedded Standard arbeitet: Klicken Sie im Projekt-Explorer unter dem Windows-Embedded-Standard-Zielsystem mit der rechten Maustaste auf die Build-Spezifikationen und wählen Sie **Verteilen** oder **Installieren** aus.

Verbesserungen beim Laden komprimierter Projektbibliotheken und DLLs mit derselben Bibliotheksversion

Um das Laden von DLLs von VIs oder von komprimierten Projektbibliotheken aus derselben Version zu vereinfachen, wird in LabVIEW 2014 das Dialogfeld **Ladewarnungen (Zusammenfassung)** unterdrückt, das in älteren LabVIEW-Versionen bei einer Querverlinkung zwischen DLLs oder komprimierten Projektbibliotheken angezeigt wird. Ein solches "Cross-Linking" entsteht durch das Laden einer zweiten DLL, wenn sich eine zuvor geladene DLL gleichen Namens aus einer anderen komprimierten Projektbibliothek oder einem VI noch im Speicher befindet. Beim Laden einer zweiten

DLL verknüpft LabVIEW diese mit dem Speicherort der zuerst geladenen DLL. DLLs oder komprimierte Projektbibliotheken müssen eine identische Versionsnummer haben, um derselben Version hinzugerechnet werden zu können.

Ausschließen abhängiger komprimierter Projektbibliotheken und DLLs aus einer Build-Spezifikation

Sie können die Anzahl der kopierten Dateien beim mehrfachen Erstellen von Builds mit identischen Bibliotheken in LabVIEW 2014 reduzieren. LabVIEW behält für abhängige komprimierte Projektbibliotheken und DLLs relative Verknüpfungen zum ursprünglichen Speicherort bei, wenn Sie festlegen, dass diese Bibliotheken aus einer Build-Spezifikation ausgeschlossen werden sollen. Anderenfalls werden die komprimierten Projektbibliotheken und DLLs dupliziert und in alle folgenden Builds kopiert. Aktivieren Sie zum Ausschließen dieser Bibliotheken **Abhängige komprimierte Bibliotheken ausschließen** und **Abhängige DLLs ausschließen** auf der Seite **Zusätzliche Ausschließungen** des Dialogfelds **Eigenschaften** für die Build-Spezifikation.

Festlegen von Zielverzeichnissen für in Build-Spezifikationen enthaltene komprimierte Projektbibliotheken und DLLs

In LabVIEW 2014 können Sie die Zielverzeichnisse für abhängige komprimierte Projektbibliotheken und DLLs festlegen, die in der Konfiguration einer Build-Spezifikation enthalten sind. Um die Zielverzeichnisse für abhängige komprimierte Projektbibliotheken festzulegen, wählen Sie auf der Seite **Einstellungen der Quelldateien** einer Build-Spezifikation **Abhängigkeiten** aus. Aktivieren Sie die Optionen **Zielverzeichnis für alle enthaltenen Dateien** und **Zielverzeichnis für komprimierte Bibliotheken und DLLs** und wählen Sie die Zielverzeichnisse aus dem Dropdown-Menü aus.

Verwalten von kompiliertem Code für ein Quellcodepaket

In LabVIEW 2013 und älteren Versionen können Sie über das Auswahlfeld **Kompilierten Code entfernen** auf der Seite **Zusätzliche Ausschließungen** im Dialogfeld **Eigenschaften** die Größe eines Quellcodepakets reduzieren. In LabVIEW 2014 haben Sie die folgenden zusätzlichen Optionen zum Verwalten kompilierten Codes:

- **Kompilierten Code beibehalten**—Behält den kompilierten Code für alle Dateien bei.
- **Dateieinstellungen für VIs und Bibliotheken beibehalten**—Behält die gespeicherten Einstellungen für alle Dateien bei.

Weitere Informationen zum Beibehalten oder Trennen von kompiliertem Code finden Sie im Thema **Vereinfachen der Versionsverwaltung durch Trennen des kompilierten Codes von VIs und anderen Dateitypen** unter **Grundlagen»Arbeiten mit Projekten und Zielen»Allgemeines»Verwendung der Versionsverwaltung in LabVIEW** in der *LabVIEW-Hilfe*.

Erweiterter Funktionsumfang durch integrierte Zusatzpakete

Durch die Integration der folgenden Zusatzpakete steht Ihnen in LabVIEW 2014 zusätzlicher Funktionsumfang zur Verfügung:

- Das LabVIEW 2014 Full Development System und das LabVIEW Professional Development System enthalten den vollständigen Funktionsumfang des LabVIEW PID and Fuzzy Logic Toolkits, allerdings mit Ausnahme des Express-VIs "PID (FPGA)", das nur im LabVIEW 2014 FPGA Module verfügbar ist.
- Im LabVIEW 2014 Professional Development System sind folgende Toolkits inbegriffen:
 - LabVIEW Database Connectivity Toolkit
 - LabVIEW Desktop Execution Trace Toolkit
 - LabVIEW Report Generation Toolkit
 - LabVIEW Unit Test Framework Toolkit

- LabVIEW VI Analyzer Toolkit

Folgende Toolkits wurden zusammengefasst:

- Das LabVIEW Adaptive Filter Toolkit wurde in das LabVIEW 2014 Digital Filter Design Toolkit integriert.
- Das LabVIEW System Identification Toolkit wurde in das LabVIEW 2014 Control Design and Simulation Module und das LabVIEW 2014 Advanced Signal Processing Toolkit integriert.
- Das FPGA Compile Farm Toolkit – jetzt unter dem Namen FPGA Compile Farm Server – und der FPGA IP Builder wurden in das LabVIEW FPGA Module integriert.
- Der Real-Time Trace Viewer wurde in das LabVIEW 2014 Real-Time Module integriert.

Neue LabVIEW-Suites

Die LabVIEW-Suites enthalten zusätzlich zu unseren beliebtesten Anwendungssoftware-Produkten und den meistgenutzten Zusatzpaketen auch die LabVIEW Professional Edition, um bei der Systemerstellung auch höchste Anforderungen in Bezug auf die Darstellung und Analyse von Daten sowie die Entwicklung und Verteilung von Software zu unterstützen. Sie können zwischen folgenden drei LabVIEW-Suites für unterschiedliche Anwendungsbereiche wählen:

- LabVIEW 2014 Automated Test Suite
- LabVIEW 2014 Embedded Control and Monitoring Suite
- LabVIEW 2014 HIL and Real-Time Test Suite

Weitere Informationen zu den neuen LabVIEW-Suites finden Sie auf ni.com/labview/suites.

Verbesserungen an LabVIEW-Webdiensten

LabVIEW 2014 enthält die nachfolgend aufgeführten Verbesserungen an den LabVIEW-Webdiensten und dem damit verbundenen Funktionsumfang.

Erweiterte Sicherheit für LabVIEW-Webdienste

Webdienste in LabVIEW 2014 arbeiten jetzt mit dem Sicherheitsprotokoll OpenSSL, Version 1.0.1g.

Benutzerdefinierte Dokumentation für VIs und Anwendungen

In LabVIEW 2014 können Sie die **Kontexthilfe** eines VIs mit Dokumentation im Web oder auf dem lokalen Computer verknüpfen. (**Windows**) Für das Erstellen benutzerdefinierter Dokumentation eignen sich alle Dateierarten mit Ausnahme von Textdateien. (**OS X und Linux**) Für benutzerdefinierte Dokumentation können nur HTML-, HTM- und PDF-Dateien verwendet werden.



Hinweis (Linux) Installieren Sie einen der folgenden PDF-Viewer, um benutzerdefinierte Dokumentation im PDF-Format verwenden zu können:

- Acroread
- Xpdf
- KPDF
- GGV

Sie können in LabVIEW 2014 außerdem benutzerdefinierte Dokumentation über das **Hilfe**-Menü eines VIs oder einer Anwendung zur Verfügung stellen.

Verknüpfen von VIs mit Dokumentation im Web

Um über die **Kontexthilfe** eine Verknüpfung zwischen einem VI und Dokumentation im Web herzustellen, wählen Sie aus dem Dropdown-Menü der Seite **Dokumentation** im Dialogfeld **VI-Einstellungen** die Option **Webhilfe** aus. Geben Sie die URL des zu verknüpfenden Themas in das Textfeld **Hilfe-URL** ein.

Verknüpfen von VIs mit lokaler benutzerdefinierter Dokumentation

Um über die **Kontexthilfe** eine Verknüpfung zwischen einem VI und lokaler Dokumentation herzustellen, wählen Sie aus dem Dropdown-Menü der Seite **Dokumentation** im Dialogfeld **VI-Einstellungen** die Option **Lokale Hilfedatei** aus. Über das Textfeld **Hilfepfad** können Sie zur benutzerdefinierten Dokumentation auf dem lokalen Computer navigieren. Zum Verknüpfen des VIs mit einem Thema in einer kompilierten Hilfedatei oder zum Verknüpfen des VIs mit einem Lesezeichen besteht auch die Möglichkeit, das Textfeld **Hilfe-Tag** zu verwenden. Anderenfalls kann das Textfeld **Hilfe-Tag** leer bleiben.

Wenn Sie den Pfad für eine kompilierte Hilfedatei namens `My Custom Help.chm` im Textfeld **Hilfepfad** festlegen, können Sie `GettingStarted.html#Intro` im Textfeld **Hilfe-Tag** eingeben, um so über den Link **Ausführliche Hilfe** und die Schaltfläche **Ausführliche Hilfe** im Fenster **Kontexthilfe** eine direkte Verknüpfung mit den "Getting Started"-Informationen in Ihrer benutzerdefinierten kompilierten Hilfedatei herzustellen.

Programmatisches Verknüpfen eines VIs mit einer Webhilfe oder lokalen Hilfedatei

Mit der Eigenschaft "Hilfe:Web-URL verwenden" der Klasse "Eigenschaften" können Sie angeben, dass ein VI mit einer Hilfedatei im Web verknüpft werden soll. Um die URL für die Hilfedatei anzugeben, verwenden Sie die Eigenschaft "Hilfe:Dokument-Web-URL" der Klasse "Eigenschaften".

Sie können ein VI mit einer lokalen Hilfedatei verknüpfen, indem Sie eine VI-Referenz mit der Eigenschaft "Hilfe:Dokumentpfad" verbinden und den Speicherort der Hilfedatei auf dem lokalen Computer angeben. Außerdem können Sie mit Hilfe der Eigenschaft "Hilfe:Dokument-Tag" ein VI mit einem individuellen HTML-Thema in der kompilierten Hilfe verknüpfen oder eine HTML-Datei mit einem Lesezeichen versehen.

Benutzerdefinierte Hilfe über das Hilfe-Menü

In LabVIEW 2014 können Sie benutzerdefinierte Hilfe auch über das **Hilfe**-Menü bereitstellen.

(Windows) Für das Erstellen benutzerdefinierter Dokumentation eignen sich alle Dateiarten mit Ausnahme von Textdateien. **(OS X und Linux)** Es können nur HTML-, HTM- und PDF-Dateien verwendet werden.



Hinweis (Linux) Installieren Sie einen der folgenden PDF-Viewer, um benutzerdefinierte Dokumentation im PDF-Format verwenden zu können:

- Acroread
- Xpdf
- KPDF
- GGV

Gehen Sie wie folgt vor, um Ihre benutzerdefinierte Hilfe im **Hilfe**-Menü anzuzeigen:

1. Fügen Sie die benutzerdefinierte Hilfedatei in das Verzeichnis `labview\help` ein.
2. Fügen Sie dann dem `labview\help`-Verzeichnis eine Textdatei hinzu, die den gleichen Namen hat wie die benutzerdefinierte Hilfedatei. Wenn die Datei beispielsweise `My Custom Help.html` heißen soll, muss die Textdatei mit dem Menütitel `My Custom Help.txt` heißen. Die Textdatei enthält den Titel der Hilfe, die im **Hilfe**-Menü angezeigt werden soll.



Hinweis Die Textdatei muss den Titel der benutzerdefinierten Hilfe enthalten, die im **Hilfe**-Menü angezeigt werden soll.

LabVIEW 2014 (64 Bit) für Mac OS X

LabVIEW 2014 gibt es als 64-Bit-Version des LabVIEW Development Systems für Mac OS X. Die Funktionen der Entwicklungsumgebung LabVIEW 2014 (64 Bit) für Mac OS X sind nahezu identisch

mit LabVIEW 2014 (32 Bit) für Mac OS X. Die Unterschiede zwischen LabVIEW 2014 (32 Bit) und LabVIEW 2014 (64 Bit) für Mac OS X werden in der folgenden Tabelle aufgezeigt.

Um Ihre VIs auf die nachfolgend aufgeführten Kompatibilitätsprobleme hin zu prüfen, stehen auf der Website von National Instruments automatisierte Tests bereit. Die Download-Seite für die Tests erreichen Sie über ni.com/info nach Eingabe des Infocodes `analyzevi`.

Unterschied	LabVIEW 2014 (32 Bit)	LabVIEW 2014 (64 Bit)
Speicherort der Anwendung	32-bit/Applications/National Instruments/LabVIEW 2014/LabVIEW.app	64-bit/Applications/National Instruments/LabVIEW 2014 64-bit/LabVIEW.app
Pfad-String-Format	Mac Classic (Macintosh HD:Users:john:Desktop)	POSIX (/Users/johnd/Desktop)
Zeilenende-Konstante	CR (Wagenrücklauf)	LF (Zeilenvorschub)
CIN	Unterstützt	Nicht unterstützt
Textkodierung	Mac-Classic-Systemkodierung	UTF-8
Druckvorschau	Druckvorschau keine systemeigene Funktion	Druckvorschau des Druckdialogfelds



Hinweis LabVIEW 2014 (64 Bit) für Mac OS X unterstützt eine begrenzte Anzahl von Treibern, Modulen und Toolkits von National Instruments. Weitere Informationen finden Sie in der KnowledgeBase auf ni.com unter *Kompatibilität von NI-Produkten mit LabVIEW (64 Bit) für Mac OS X*.

LabVIEW 2014 (64 Bit) für Linux

Mit LabVIEW 2014 ist eine 64-Bit-Version des LabVIEW Development Systems für Linux verfügbar. LabVIEW 2014 (64 Bit) für Linux unterstützt die gleichen Funktionen und Zusatzpakete wie LabVIEW 2014 (32 Bit) für Linux mit Ausnahme der folgenden:

- Suchmaschine für NI-Gerätetreiber
- LabVIEW Control Design and Simulation Module
- LabVIEW VI Analyzer Toolkit

Änderungen an den Touchpanel-Funktionen

Schutz vor Datenänderungen mit Hilfe von Schreibfilter-VIs

In LabVIEW 2014 wurde die Palette "Fortgeschrittene Dateifunktionen" um Schreibfilter-VIs ergänzt. Schreibfilter schützen Daten vor ungewünschten Änderungen, indem sie Schreiboperationen umleiten. Sie können Laufwerke mit Hilfe der VIs für den erweiterten Schreibfilter (EWF) schützen und Schreiboperationen auf ein anderes Laufwerk oder in den RAM umleiten. Verwenden Sie dateibasierte Schreibfilter (FBWF), um Dateien und Ordner auf einem Datenträger zu schützen und Schreiboperationen in den Cache umzuleiten.

In älteren Versionen war für das Arbeiten mit Schreibfilter-VIs das LabVIEW Touch Panel Module erforderlich. In LabVIEW 2014 stehen die Schreibfilter-VIs beim Entwickeln einer Touchpanel-Anwendung im **Projekt-Explorer** zur Verfügung. Zur Unterstützung für Touchpanel-Anwendungen ist der LabVIEW Application Builder erforderlich, der im LabVIEW Professional Development System inbegriffen ist.

Verwenden der Touchpanel-Projektvorlage und Touchpanel-VI-Vorlage

Mit Hilfe der Touchpanel-Projektvorlage können Sie Projekte erstellen, die auf Touchpanels mit Windows Embedded Standard 7 ausgeführt werden können. Die Projektvorlage enthält eine VI-Vorlage, die Sie für Ihre Touchpanel-Anwendung anpassen können. Wählen Sie **Datei»Projekt erstellen** und navigieren Sie zur Touchpanel-Projektvorlage. Im Dialogfeld **Projekt erstellen** finden Sie die Einstellungen für das Touchpanel-Zielsystem und die VI-Vorlage. Weitere Informationen zum Bearbeiten des Projekts finden Sie auch im Ordner `Project Documentation` im **Projekt-Explorer**.

Sie können außerdem eine Touchpanel-VI-Vorlage zu einem vorhandenen Touchpanel-Zielsystem hinzufügen. Es stehen Vorlagen im Hoch- und Querformat mit unterschiedlichen Benutzeroberflächen für spezifische Arten von Touchpanels zur Verfügung. Diese Vorlagen enthalten häufig in Touchpanel-Anwendungen verwendete Bedien- und Anzeigeelemente. Klicken Sie mit der rechten Maustaste auf ein Touchpanel-Zielsystem und wählen Sie **Neue VI-Vorlage** aus, um eine Touchpanel-VI-Vorlage zum Zielsystem hinzuzufügen. Daraufhin wird das Dialogfeld **Vorlage auswählen** angezeigt. Wählen Sie die gewünschte VI-Vorlage aus.

Arbeiten mit Touchpanel-Zielsystemen

In LabVIEW 2014 können Sie Anwendungen für Touchpanels entwickeln, die mit Windows Embedded Standard 7 arbeiten, und diese Anwendungen auf Fehler untersuchen. Bisher war dafür das LabVIEW Touch Panel Module erforderlich. In LabVIEW 2014 werden Touchpanel-Anwendungen auf dem Host-Computer entwickelt und auf Fehler untersucht und können dann direkt auf ein Touchpanel übertragen werden. Zur Unterstützung von Touchpanel-Zielsystemen ist der LabVIEW Application Builder erforderlich. Der Application Builder ist nur im LabVIEW Professional Development System enthalten.

Neue und geänderte Toolkits

LabVIEW Third Party Licensing and Activation Toolkit

Über das LabVIEW Third Party Licensing and Activation Toolkit 2014 können Sie Benutzern eine Deaktivierungsoption für ein Zusatzpaket zur Verfügung stellen, wenn das Zusatzpaket über das Toolkit lizenziert ist. Klicken Sie zum Deaktivieren des Zusatzpakets auf **Hilfe»Zusatzpakete aktivieren**, um den Aktivierungsassistenten für Zusatzpakete von Drittanbietern zu öffnen. Wählen Sie ein Zusatzpaket aus, klicken Sie auf **Deaktivieren** und fahren Sie wie vom Assistenten angegeben fort.

Neuerungen und Änderungen in vorherigen LabVIEW-Versionen

Alle neuen Funktionen, die seit Ihrer bestehenden LabVIEW-Version hinzugekommen sind, finden Sie in den Upgrade-Hinweisen der jeweiligen Versionen. Zum Öffnen dieser Dokumente besuchen Sie die Seite ni.com/info und geben Sie einen der folgenden Infocodes ein:

- *Hinweise zum Upgrade von LabVIEW 2010*—[upnote10de](#)
- *Hinweise zum Upgrade von LabVIEW 2011*—[upnote11de](#)
- *Hinweise zum Upgrade von LabVIEW 2012*—[upnote12de](#)
- *Hinweise zum Upgrade von LabVIEW 2013*—[upnote13de](#)

Weitere Informationen zu Marken von National Instruments finden Sie in den *NI Trademarks and Logo Guidelines* auf ni.com/trademarks. Sonstige hierin erwähnte Produkt- und Firmenbezeichnungen sind Marken oder Handelsnamen der jeweiligen Unternehmen. Nähere Informationen über den Patentschutz von Produkten/Technologien von National Instruments finden Sie unter **Hilfe»Patente** in Ihrer Software, in der Datei `patents.txt` auf Ihrem Datenträger oder unter *National Instruments Patent Notice* auf der Website ni.com/patents. Informationen zu Endbenutzer-Lizenzverträgen (EULAs) und Rechtshinweisen von Drittanbietern finden Sie in der Readme zu Ihrem Produkt von National Instruments. Informationen zu den Global-Trade-Compliance-Richtlinien von National Instruments sowie zu Bezugsquellen für relevante HTS-Codes, ECCNs und andere Import-/Exportangaben finden Sie auf ni.com/legal/export-compliance unter der Überschrift *Export Compliance Information*.