

# Hinweise zum Upgrade auf LabVIEW™ 2018

In diesem Dokument wird beschrieben, wie Sie LabVIEW für Windows, macOS und Linux auf LabVIEW 2018 aktualisieren. Bevor Sie auf die neue LabVIEW-Version umsteigen, informieren Sie sich in diesem Dokument zu den 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 2018

## Inhalt

---

Upgrade auf LabVIEW 2018.....	1
1. Sichern der VIs und der Computerkonfiguration.....	2
2. Prüfen der aktuellen Funktionsweise der VIs.....	3
3. Installieren von LabVIEW sowie von 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 2013 oder älteren Versionen.....	6
Upgrade von LabVIEW 2014.....	6
Upgrade von LabVIEW 2015.....	7
Upgrade von LabVIEW 2016.....	7
Upgrade von LabVIEW 2017.....	7
Änderungen und Neuerungen in LabVIEW 2018.....	8
Anpassen eines datentypflexiblen VIs für verschiedene Datentypen.....	8
Ausführen von Operationen über die LabVIEW-Kommandozeilen-Schnittstelle.....	9
Aufrufen von Python-Code über LabVIEW.....	9
Verbesserungen am Application Builder.....	10
Verbesserungen an der Programmoberfläche.....	10
Verbesserungen am Blockdiagramm.....	11
Verbesserungen am Frontpanel.....	12
Neue VIs und Funktionen.....	12
Neue und geänderte Eigenschaften und Methoden.....	13
Änderungen und Neuerungen in vorherigen LabVIEW-Versionen.....	13

## Upgrade auf LabVIEW 2018

---

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 LabVIEW-Versionen, zu denen Standard-Support erhältlich ist, finden Sie auf der NI-Webseite unter [ni.com/info](http://ni.com/info) nach Eingabe des Infocodes `lifecycle`.

## Überblick über den empfohlenen Upgrade-Prozess

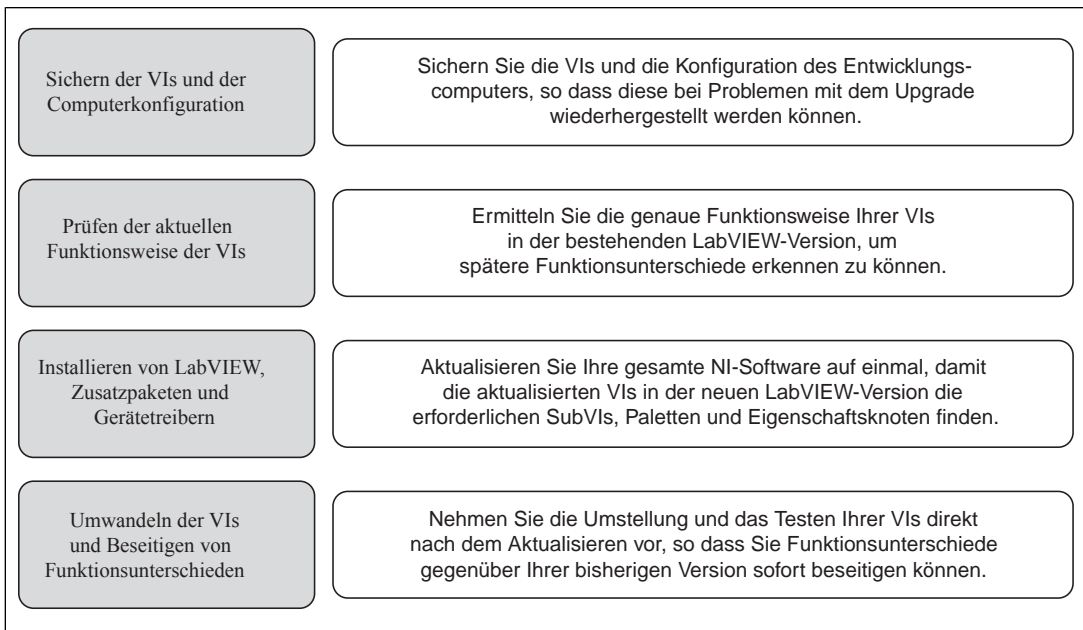


Figure 1.



**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](http://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 2018 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 VI-Version, wenn sich durch die Umstellung verursachte Funktionsunterschiede nicht beheben lassen. Weitere Informationen zur Verwendung von Versionsverwaltungssystemen mit

LabVIEW finden Sie auf der Registerkarte **Inhalt** der *LabVIEW-Hilfe* unter **Grundlagen»Arbeiten mit Projekten und Zielen»Allgemeines»Verwendung der Versionsverwaltung in LabVIEW**.

- **Erstellen einer Kopie der VIs**— Gehen Sie zum Erstellen einer Sicherungskopie je nach der ursprünglichen Speicher­methode der VIs wie folgt vor:
  - Als Projekt gespeichert—Öffnen Sie das Projekt und speichern Sie über **Datei»Speichern** unter eine Kopie der \*.lvproj-Datei und des Projekteinhalts. Wählen Sie **Alle Abhängigkeiten einschließen**, um auch Komponenten einzuschließen, von denen das Projekt abhängt.
  - Als LLB oder in 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 für alle ausfallkritischen Systeme in Produktionsprozessen dringend 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

Aufgrund der Weiterentwicklung von LabVIEW kann es gelegentlich vorkommen, dass ein VI in der aktuellen Version anders funktioniert als in einer Vorgängerversion von LabVIEW. Vergleichen Sie daher, wie ein VI in der bisherigen Version von LabVIEW und in der neuen Version von LabVIEW funktioniert. Führen Sie dazu folgende Tests durch:

- Führen Sie eine Massenkompilierung der VIs in der Vorgängerversion von LabVIEW durch und prüfen Sie das dabei erstellte Massenkompilierungsprotokoll auf fehlerhafte VIs. Eine Massenkompilierung von VIs vor dem Upgrade ist besonders nützlich, wenn mehrere Personen an der VI-Entwicklung beteiligt sind oder wenn Sie vermuten, dass VIs lange nicht mehr kompiliert wurden. Damit ein Massenkompilierungsprotokoll erstellt wird, aktivieren Sie die Option **Ergebnisse aufzeichnen** im Dialogfeld **Massenkompilierung**. Weitere Informationen dazu finden Sie in 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, wie die VIs mit unerwarteten Daten umgehen.

Weitere Informationen zum Testen der VIs finden Sie in 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 sowie von 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.

#### b. Kopieren von user.lib-Dateien

Damit alle benutzerdefinierten Elemente und VIs aus der bisherigen LabVIEW-Version auch für die VIs in LabVIEW 2018 verfügbar sind, müssen Sie das Verzeichnis `labview\user.lib` Ihrer bisherigen Version in das Verzeichnis `labview\user.lib` von LabVIEW 2018 kopieren.

### 4. Umwandeln der VIs und Beseitigen von Funktionsunterschieden

VIs werden durch eine Massenkompilierung in LabVIEW 2018 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.



**Hinweis** Wenn Sie Ihre VIs beim Upgrade auf LabVIEW 2017 massenkompiliert haben, können Sie diesen Schritt überspringen. Alle LabVIEW-Versionen beginning mit LabVIEW 2017 sind in Bezug auf die Runtime-Engine rückwärtskompatibel.

#### a. Massenkompilierung von VIs in der neuen Version von LabVIEW

Bei der Massenkompilierung werden die VIs für die Verwendung in LabVIEW 2018 umgewandelt und 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 von aktualisierungsbezogenen Problemen führen Sie eine Massenkompilierung mit Fehlerprotokoll durch, indem Sie im Dialogfeld **Massenkompilierung** die Option **Ergebnisse aufzeichnen** aktivieren.



**Hinweis** Bei der Massenkompilierung von VIs, die FPGA- oder Real-Time-Ressourcen enthalten, werden die VIs im Dialogfeld **Massenkompilierung** unter Umständen als nicht ausführbar angezeigt. Um die VIs auf Fehler zu prüfen, müssen Sie sie in einem LabVIEW-Projekt mit den erforderlichen FPGA- oder Real-Time-Ressourcen unter dem FPGA- oder Real-Time-Zielsystem öffnen.

Informationen zur Massenkompilierung finden Sie in den folgenden Abschnitten der *LabVIEW-Hilfe*:

- **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 zwischen Ihrer bisherigen LabVIEW-Version und LabVIEW 2018 können gelegentlich zu Fehlern in VIs führen. Zum schnellen Erkennen und Korrigieren fehlerhafter VIs in LabVIEW 2018 gehen Sie wie folgt vor:

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 NI 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:

- **Führen Sie die VI Analyzer Upgrade Tests durch**—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 VI Analyzer Upgrade Tests für alle Nachfolgeversionen Ihrer aktuellen LabVIEW-Version herunter. Zur Download-Seite für diese Tests gelangen Sie, indem Sie [ni.com/info](http://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 den entsprechenden 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 2015 VI Analyzer Upgrade Tests ein potentieller Funktionsunterschied erkannt wird, lesen Sie den Abschnitt *Upgrade von LabVIEW 2014*.
- **Lesen Sie die Dokumentation zum Upgrade**
  - Abschnitt *Hinweise zur Kompatibilität beim Upgrade* dieses Dokuments—Listet Änderungen auf, die die Funktion Ihrer VIs beeinflussen 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 2018—Führt alle Programmfehler auf, die kurz vor der Veröffentlichung von LabVIEW 2018 erkannt wurden. Zu dieser Liste gelangen Sie, indem Sie [ni.com/info](http://ni.com/info) besuchen und den Infocode `lv2018ki` eingeben. Unter *Upgrade - Behavior Change und Upgrade - Migration* (falls vorhanden) können Sie sehen, welche Behelfslösungen für die einzelnen Probleme vorgeschlagen werden.
- Dokumentation für Module und Toolkits—Listet Upgrade-Probleme bei bestimmten Modulen und Toolkits auf, beispielsweise dem LabVIEW FPGA oder LabVIEW Real-Time Module.

- Readme-Dateien zu Treibern—Enthalten Listen mit möglichen Upgrade-Problemen. Die Readme-Datei befindet sich auf dem Installationsdatenträger des Treibers.



**Tipp** Ob Funktionsunterschiede durch ein Treiber-Upgrade oder das LabVIEW-Upgrade auf LabVIEW 2018 verursacht wurden, erfahren Sie durch Testen der VIs in Ihrer bisherigen LabVIEW-Version.

- **Führen Sie Ihre eigenen Tests durch**—Führen Sie in LabVIEW 2018 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 2018»Fehlersuche beim Upgrade** Informationen zu den folgenden Themen:

- Suchen nach fehlenden Modul- oder Toolkit-Funktionen
- Suchen nach fehlenden SubVIs, Paletten und Eigenschaftsknoten
- Feststellen, warum sich VIs aus älteren Versionen nicht in LabVIEW 2018 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 versionspezifische 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 2018, die nicht anderweitig dokumentiert sind, werden in der Datei `readme.html` im `labview`-Verzeichnis beschrieben.

## Upgrade von LabVIEW 2013 oder älteren Versionen

Informationen zu möglichen Kompatibilitätsproblemen beim Upgrade von LabVIEW 2013 oder älteren Versionen auf LabVIEW 2018 finden Sie auf der Website von NI unter `ni.com/info` nach Eingabe des Infocodes `upnote14de`. Weitere mögliche Upgrade-Probleme sind im vorliegenden Dokument unter *Upgrade von LabVIEW x* beschrieben.

## Upgrade von LabVIEW 2014

Beim Upgrade von LabVIEW 2014 auf LabVIEW 2018 können die nachfolgend aufgeführten Kompatibilitätsprobleme auftreten. Alle weiteren potentiellen Probleme beim Upgrade von LabVIEW sind in den Abschnitten *Upgrade von LabVIEW 2015*, *Upgrade von LabVIEW 2016* und *Upgrade von LabVIEW 2017* dieses Dokuments beschrieben.

### Erkennen von Pufferzuteilungen in LabVIEW-Anwendungen

Zu den Neuerungen in LabVIEW 2014 Service Pack 1 und neueren Versionen gehört das Fenster **Pufferzuweisungen anzeigen** mit Informationen zur Puffervergabe innerhalb einer LabVIEW-Anwendung. Klicken Sie zur Anzeige dieses Dialogfelds auf **Werkzeuge»Profil»Pufferzuweisungen anzeigen**.

### Hyperlinks in freien Beschriftungen

LabVIEW 2015 und neuere Versionen erkennen URLs in freien Beschriftungen und kennzeichnen diese automatisch durch blauen unterstrichenen Text. Beim Upgrade von LabVIEW 2014 oder älteren Versionen werden URLs in freien Beschriftungen jedoch nicht automatisch in Hyperlinks umgewandelt. Zum Aktivieren von Hyperlinks in Frontpanel-Beschriftungen klicken Sie mit der rechten Maustaste auf eine

freie Beschriftung und setzen Sie ein Häkchen vor dem Menüpunkt **Hyperlinks aktivieren**. Hyperlinks in Blockdiagrammbeschriftungen können nicht deaktiviert werden.

### Veraltete VIs, Funktionen und Knoten

Folgende VIs werden von LabVIEW 2015 und neueren LabVIEW-Versionen nicht mehr unterstützt:

- **Aus Tabellenkalkulationsdatei lesen**—Verwenden Sie stattdessen das VI "Tabelle mit Trennzeichen lesen".
- **In Tabellenkalkulationsdatei schreiben**—Verwenden Sie stattdessen das VI "Tabelle mit Trennzeichen schreiben".

## Upgrade von LabVIEW 2015

Beim Upgrade von LabVIEW 2015 auf LabVIEW 2018 kann das nachfolgend aufgeführte Kompatibilitätsproblem auftreten. Alle weiteren potentiellen Probleme beim Upgrade von LabVIEW sind in den Abschnitten *Upgrade von LabVIEW 2016* und *Upgrade von LabVIEW 2017* dieses Dokuments beschrieben.

In LabVIEW 2016 und neueren LabVIEW-Versionen enthält das Dialogfeld **Schnelleinfüge-Einstellungen** eine Liste mit Standard-Bedientasten für Frontpanel- und Blockdiagrammobjekte. Wenn Sie von LabVIEW 2015 oder älteren Versionen auf LabVIEW 2016 umsteigen, werden in der älteren Version hinzugefügte Schnell Tasten nicht automatisch in die Liste der neuen Version übernommen.

## Upgrade von LabVIEW 2016

Beim Upgrade von LabVIEW 2016 auf LabVIEW 2018 kann das nachfolgend aufgeführte Kompatibilitätsproblem auftreten. Alle weiteren potentiellen Probleme beim Upgrade von LabVIEW sind im Abschnitt *Upgrade von LabVIEW 2017* dieses Dokuments beschrieben.

### Änderungen in der Funktionsweise bei Akteur-Framework-VIs

In LabVIEW-Versionen bis 2016 gibt der Unterakteur für einen fehlgeschlagenen Start aufgrund eines Fehlers in der Methode "Init vor Start" eine Fehlermeldung aus und sendet die Nachricht "Letzte Bestätigung" an den aufrufenden Akteur. In LabVIEW 2017 gibt der Unterakteur einen Fehler aus, ohne die Meldung "Letzte Bestätigung" an den aufrufenden Akteur zu senden.

## Upgrade von LabVIEW 2017

Beim Upgrade von LabVIEW 2017 auf LabVIEW 2018 sind die nachfolgend beschriebenen Gesichtspunkte zu beachten.

### Rückwärtskompatibilität der LabVIEW-Runtime-Engine

Beginnend mit LabVIEW 2017 sind die LabVIEW-Runtime-Engines rückwärtskompatibel. Binärdateien und VIs aus älteren Versionen von LabVIEW können ohne erneute Massenkompilierung bzw. erneutes Erzeugen der entsprechenden Binärdateien in der aktuellen Version von LabVIEW geladen und ausgeführt werden. Alle in künftigen LabVIEW-Versionen erzeugten Binärdateien und VIs können dann beispielsweise in LabVIEW 2017 ohne vorherige Kompilierung geladen werden. Diese Neuerung betrifft EXE-Dateien, DLLs und komprimierte Projektbibliotheken.

Um die Rückwärtskompatibilität für Binärdateien zu aktivieren, markieren Sie das folgende Auswahlfeld auf der Seite **Fortgeschritten** im jeweiligen Dialogfeld (abhängig von Ihren Build-Spezifikationen):

Build-Spezifikation	Dialogfeld	Auswahlfeld
Eigenständige Anwendung (EXE)	Eigenschaften für Anwendung	Künftigen LabVIEW-Runtime-Versionen das Ausführen dieser Anwendung gestatten

Build-Spezifikation	Dialogfeld	Auswahlfeld
Komprimierte Projektbibliotheken	Eigenschaften für komprimierte Bibliothek	Künftigen LabVIEW-Versionen das Laden dieser komprimierten Bibliothek gestatten
DLL	Eigenschaften für DLL	Künftigen LabVIEW-Runtime-Versionen das Laden dieser DLL gestatten

Bei Build-Spezifikationen, die Sie in LabVIEW 2017 oder einer neueren Version erzeugen, sind diese Optionen standardmäßig aktiviert. Sie können diese Optionen deaktivieren, wenn Sie eine Build-Spezifikation an eine bestimmte Version von LabVIEW binden möchten. Das Deaktivieren dieser Optionen verhindert Änderungen an den Leistungsprofilen, wodurch unerwartete aus Compiler-Upgrades resultierende Probleme vermieden werden. Bei Real-Time-Anwendungen werden diese Optionen zwar nicht in den Dialogfeldern angezeigt, die Funktion ist jedoch standardmäßig aktiviert.

### Änderungen in der Funktionsweise bei VI zur Protokollerstellung


In LabVIEW 2018 unterstützen die Protokoll-VIs nicht mehr die Protokollerstellung im Standardprotokollformat. Protokolle können nur noch im HTML-, Word- oder Excel-Format erstellt werden. Folgende VIs sind von der Änderung betroffen:

- **Einfaches Drucken von VI-Panels oder Dokumentationen**—Dieses VI ist veraltet. Verwenden Sie stattdessen das VI **VI-Panels oder Dokumentation drucken**.
- **Einfaches Textprotokoll**—Dieses VI ist veraltet. Verwenden Sie stattdessen das VI **Einfaches Textprotokoll erstellen**.
- **Protokolltyp ermitteln**—Dieses VI ist veraltet. Verwenden Sie stattdessen das VI **Protokolltyp**.
- **Neues Protokoll**—Dieses VI ist veraltet. Verwenden Sie stattdessen das VI **Protokoll erstellen**.
- **Tabulatorbreite für Protokoll festlegen**—Dieses VI ist veraltet.

### Veraltete VIs, Funktionen und Knoten

LabVIEW 2018 und neuere LabVIEW-Versionen unterstützen das VI **Zahl nach Enum** nicht. Verwenden Sie stattdessen die Funktion "Typ erzwingen".

## Änderungen und Neuerungen in LabVIEW 2018

Mit dem Symbol  gekennzeichnete Neuerungen stammen von Vorschlägen aus dem NI Idea Exchange. Zum NI Idea Exchange gelangen Sie über [ni.com/info](http://ni.com/info) nach Eingabe des Infocodes `ex3gus`.

In der Datei `readme.html` im `labview`-Verzeichnis finden Sie Informationen zu bekannten Funktionsproblemen von LabVIEW 2018 sowie zu Fehlerkorrekturen, Kompatibilitätsproblemen und kurz vor Veröffentlichung hinzugefügten Funktionen.

### Anpassen eines datentypflexiblen VIs für verschiedene Datentypen

Die Palette "Vergleich" enthält die neue Unterpalette "Auf Typ überprüfen". Mit der Funktion und den VIs zur Typüberprüfung können Sie ein datentypflexibles VI (`.vim`) so konfigurieren, dass es nur mit Datentypen arbeitet, die bestimmte Kriterien erfüllen. Mit der Typspezialisierungsstruktur können Sie Programmabschnitte in einem datentypflexiblen VI für bestimmte Datentypen anpassen.



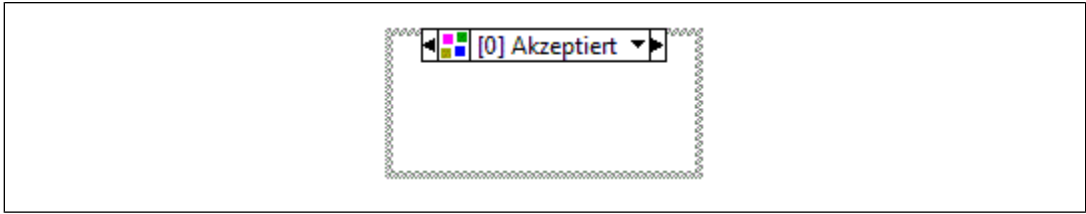


Figure 2.

Ein Beispiel für die Verwendung der Typspezialisierungsstruktur zum Anpassen von Programmabschnitten in einem datentypflexiblen VI finden Sie unter `labview\examples\Malleable VIs\Type Specialization Structure\Malleable VIs - Type Specialization Structure.lvproj`.

## Ausführen von Operationen über die LabVIEW-Kommandozeilen-Schnittstelle

In LabVIEW 2018 können Sie vordefinierte Operationen mit Hilfe von Befehlen über die LabVIEW-Kommandozeilen-Schnittstelle (LabVIEW-CLI) aufrufen. Verwenden Sie die Kommandozeile beispielsweise, um den Build-Prozess von LabVIEW-Anwendungen zu automatisieren. Die LabVIEW-Kommandozeilen-Schnittstelle unterstützt die folgenden Operationen:

- **MassCompile**—Führt eine Massenkompilierung der Dateien im angegebenen Verzeichnis aus.
- **ExecuteBuildSpec**—Erstellt mit Hilfe der Einstellungen in den angegebenen Build-Spezifikationen eine Anwendung, eine Bibliothek oder eine Bitdatei und gibt den Pfad der Ausgabedateien aus.
- **RunVI**—Führt ein VI mit der vordefinierten Anschlussfeld-Schnittstelle aus und gibt die Ausgabewerte des VIs oder (im Fehlerfall) Angaben zu Fehlern aus.
- **CloseLabVIEW**—Schließt LabVIEW ohne Eingabeaufforderungen.
- **(VI Analyzer Toolkit) RunVIANalyzer**—Führt die angegebene VI-Analyse-Aufgabe im LabVIEW VI Analyzer Toolkit aus und speichert das Testprotokoll am angegebenen Speicherort.
- **(Unit Test Framework Toolkit) RunUnitTests**—Führt Tests für die angegebenen Dateien im Unit Test Framework Toolkit durch und speichert die JUnit-Datei am angegebenen Speicherort.



**Hinweis** Um diese Operation in LabVIEW auszuführen, müssen Sie die UTF-Junit-Protokollbibliothek unter Verwendung des VI Package Managers (VIPM) von JKI installieren. Weitere Informationen zum Umgang mit der VIPM-Software erhalten Sie, indem Sie auf der Registerkarte **Inhalt** der *LabVIEW-Hilfe* auf **Toolkits»Zugriff auf LabVIEW-Zusatzpakete über den VI Package Manager (VIPM)** klicken.

Sie können auch benutzerdefinierte Operationen erstellen, die in LabVIEW aufgerufen werden können.

Weitere Informationen zur LabVIEW-CLI finden Sie in der *LabVIEW-Hilfe* unter **Grundlagen»Ausführen von Operationen über die LabVIEW-Kommandozeilen-Schnittstelle**.

## Aufrufen von Python-Code über LabVIEW

Die Palette "Konnektivität" enthält eine neue Unterpalette namens "Python", mit der Sie Python-Code von LabVIEW-Code aus aufrufen können. Die Python-Palette umfasst folgende Funktionen:

- **"Python-Session öffnen"**—Öffnet eine Python-Session mit einer bestimmten Version von Python.
- **"Python-Knoten"**—Ruft eine Python-Funktion direkt auf.
- **"Python-Session schließen"**—Schließt eine Python-Session.



**Hinweis** Für die Arbeit mit Python-Funktionen in LabVIEW muss Python 2.7 oder 3.6 installiert werden. Obwohl nicht unterstützte Versionen mit den Python-Funktionen in LabVIEW funktionieren können, wird empfohlen, nur mit den unterstützten Versionen von

Python zu arbeiten. Besuchen Sie [ni.com/info](http://ni.com/info) und geben Sie den Infocode `python` ein, um mehr über das Installieren von Python zu erfahren.

## Verbesserungen am Application Builder

In LabVIEW 2018 wurden folgende Verbesserungen am LabVIEW Application Builder und an den Build-Spezifikationen vorgenommen:

### Erstellen von Paketen unter Windows und auf Linux-Real-Time-Zielsystemen

Sie können Pakete erstellen und diese mit dem NI-Paketmanager oder mit SystemLink anderen Personen bereitstellen. Im NI-Paketmanager und in SystemLink können Pakete für Dateien jeder Art (z. B. Quellcodepakete, komprimierte Projektbibliotheken, DLLs, .NET-Assemblies oder ausführbare Dateien) erzeugt werden.

**(Windows 64 Bit)** Sie können NI-Pakete (\*.nibpkg) erstellen, indem Sie im **Projekt-Explorer** mit der rechten Maustaste auf **Build-Spezifikationen** klicken und **Neu»Package** auswählen. Die Empfänger können mit Hilfe des Paketmanagers oder SystemLink einen Feed abonnieren, mit dem die Pakete gesucht und installiert werden können.

**(NI Linux Real-Time)** Bei installiertem LabVIEW Real-Time Module können opkg-Pakete (\*.ipk) auch auf NI-Linux-Real-Time-Zielsystemen erstellt werden. Empfänger können Pakete über SystemLink oder über die Kommandozeile auf dem NI-Linux-Real-Time-Zielsystem installieren. Der Paketmanager unterstützt keine \*.ipk-Dateien.

Weitere Informationen zum Erstellen von Paketen finden Sie, indem Sie auf der Registerkarte **Inhalte** in der *LabVIEW-Hilfe* zu **Grundlagen»Erstellen und Bereitstellen von Anwendungen»Erstellen von Build-Spezifikationen»Erstellen von Paketen zu Distributionszwecken** navigieren.

### Unterstützung der Rückwärtskompatibilität für in LabVIEW erzeugte .NET-Assemblies

Mit der Unterstützung für Rückwärtskompatibilität können .NET-Interop-Assemblies nicht mehr nur in der LabVIEW-Version geladen werden, in der sie erstellt wurden, sondern auch mit der neuesten Version der auf dem Computer installierten LabVIEW-Runtime-Engine. So kann beispielsweise eine in LabVIEW 2018 erstellte .NET-Interop-Assembly mit Nachfolgerversionen der LabVIEW-Runtime-Engine geladen und ausgeführt werden, ohne dass die Assembly dazu neu kompiliert werden muss.


Zum Wahren der Rückwärtskompatibilität für eine .NET-Assembly aktivieren Sie im Dialogfeld **Eigenschaften für .NET-Interop-Assembly** unter **Fortgeschritten** die Option **Künftigen LabVIEW-Versionen das Laden dieser .NET-Assembly gestatten**.

Bei Build-Spezifikationen, die Sie in LabVIEW 2018 oder einer neueren Version erzeugen, ist diese Option standardmäßig aktiviert. Sie können diese Option deaktivieren, wenn Sie eine Build-Spezifikation an eine bestimmte Version von LabVIEW binden möchten. Beim Deaktivieren dieser Option werden Änderungen an den Leistungsprofilen verhindert, wodurch aus Compiler-Upgrades resultierende unerwartete Probleme vermieden werden. Bei Real-Time-Anwendungen wird diese Option nicht im Dialogfeld angezeigt, weil die Funktion standardmäßig aktiviert ist.

## Verbesserungen an der Programmoberfläche

In LabVIEW 2018 gab es folgende Verbesserungen an der Programmoberfläche:

### Verbesserungen beim Erstellen von Typdefinitionen

 In LabVIEW 2018 stehen Ihnen im Vergleich zu bisherigen LabVIEW-Versionen mehr Möglichkeiten zum Erstellen einer Typdefinition zur Verfügung. Bei einer Typdefinition werden alle Exemplare eines

benutzerdefinierten Bedien- oder Anzeigeelements mit einer gespeicherten Datei verknüpft. Typdefinitionen lassen sich nach einer der folgenden Vorgehensweisen erstellen:

- Wählen Sie **Datei»Neu** und wählen Sie unter **Andere Dateien** den Unterpunkt **Typdefinition** aus.
- Klicken Sie mit der rechten Maustaste im **Projekt-Explorer** auf **Mein Computer** und wählen Sie **Neu»Typdefinition** aus dem Kontextmenü aus.

*[Idee wurde vom NI-Diskussionsforen-Mitglied Mathis\_B. eingereicht.]*

### Tastenkombinationen für das Formatieren von Text

☞ Folgende Tastenkombinationen stehen beim Bearbeiten von Text in der LabVIEW-Umgebung für den Schriftschnitt zur Verfügung:

- <Strg + B>—Stellt den Text fettgedruckt dar.
- <Strg + I>—Stellt den Text kursiv dar.
- <Strg + U>—Stellt den Text unterstrichen dar.

*[Idee wurde vom NI-Diskussionsforen-Mitglied vt92 eingereicht]*

## Verbesserungen am Blockdiagramm

In LabVIEW 2018 wurden die nachfolgend aufgeführten Verbesserungen am Blockdiagramm vorgenommen.

### Verbesserungen beim Umgang mit Fehlern in parallelen For-Schleifen

In LabVIEW 2018 gibt es erstmals Fehlerregister. Mit Fehlerregistern wird das Reagieren auf Fehler bei einer For-Schleife mit parallelen Iterationen vereinfacht. Fehlerregister werden anstelle von Schieberegistern für Fehler-Cluster in einer parallelen Schleife verwendet, wie in der nachfolgenden Abbildung dargestellt.

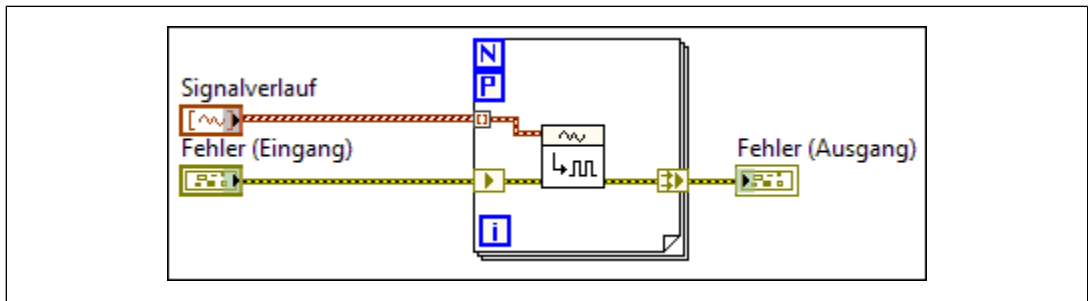


Figure 3.

Fehlerregister fassen automatisch Fehler aus parallel ausgeführten Iterationen zusammen. Im Einklang mit empfohlenen Methoden der Programmierung werden Fehler durch Schieberegister geleitet, wobei Schieberegister automatisch in Fehlerregister umgewandelt werden, wenn für eine For-Schleife parallele Iterationen aktiviert sind.

Die Funktionsweise von Fehler- und Schieberegistern ist unterschiedlich. Der linke Anschluss des Fehlerregisters funktioniert wie ein nicht indizierter Eingangstunnel und erzeugt bei jeder Iteration den gleichen Wert. Der rechte Anschluss des Fehlerregisters fasst die Werte aller Iterationen zusammen, so dass Fehler oder Warnungen von der frühesten Iteration (nach Index) den Ausgangswert des Fehlerregisters bilden. Wird die For-Schleife null Mal wiederholt, wird der mit dem linken Tunnel verbundene Wert an den Ausgang im rechten Tunnel weitergeleitet.

## Verbesserungen beim Entfernen und erneuten Verbinden von Objekten

Wenn Sie eine Auswahl von Blockdiagrammobjekten entfernen und neu verbinden, werden auch alle Gestaltungselemente entfernt, die sich innerhalb des markierten Bereichs befinden. Das gilt auch für freie Beschriftungen. Um Objekte zu entfernen und neu zu verbinden, ziehen Sie Auswahlrechteck um die Blockdiagrammobjekte auf, klicken Sie die Auswahl mit der rechten Maustaste an und wählen Sie **Entfernen und neu verbinden** aus. Sie können auch mit Hilfe der Schnellasten der **Schnelleinfügeliste**, <Strg + Leerzeichen> und <Strg + R>, die Objekte nach dem Auswählen entfernen und erneut verbinden.

## Verbesserungen am Frontpanel

### Bedien- und Anzeigeelemente in NXG-Darstellung

Auf der Elementepalette gibt es die neue Unterpalette **NXG-Darstellung** für Frontpanel-Bedien- und -Anzeigeelemente. Mit den Bedien- und Anzeigeelementen in NXG-Darstellung können Sie Frontpanel mit dem gleichen Look wie in LabVIEW NXG erstellen. Die Darstellung dieser Elemente ändert sich je nach Plattform, auf der das VI ausgeführt wird. Durch Verwendung dieser Bedien- und Anzeigeelemente wird das Umstellen der VIs auf LabVIEW NXG erleichtert.

## Neue VIs und Funktionen

LabVIEW 2018 umfasst folgende neue VIs und Funktionen:

- Der Palette "Konvertierung" wurde eine neue Unterpalette mit dem Namen "Auf Typ überprüfen" hinzugefügt, die folgende VIs und Funktionen enthält:
  - "Array-Dimensionsanzahl überprüfen"
  - "Array-Dimensionsgrößen überprüfen"
  - "Auf komplexen Zahlentyp überprüfen"
  - "Fehler-Cluster-Typ überprüfen"
  - "Auf Festkommatyp überprüfen"
  - "Auf Fließkommatyp überprüfen"
  - "Auf Dezimalzahlentyp überprüfen"
  - "Auf Integer-Typ überprüfen"
  - "Auf reellen Fließkommatyp überprüfen"
  - "Auf reellen Zahlentyp oder Signalverlaufstyp überprüfen"
  - "Auf reellen Zahlentyp überprüfen"
  - "Auf identischen oder abgeleiteten Typ überprüfen"
  - "Auf Skalar- oder Signalverlaufstyp überprüfen"
  - "Auf Skalartyp überprüfen"
  - "Auf vorzeichenbehafteten Integer-Typ überprüfen"
  - "Strukturtyp-Übereinstimmung überprüfen"
  - "Auf vorzeichenlosen Integer-Typ überprüfen"
  - Typspezialisierungsstruktur
- Der Palette "Konnektivität" wurde eine neue Unterpalette mit dem Namen "Python" hinzugefügt, die folgende Funktionen enthält:
  - "Python-Session öffnen"
  - "Python-Knoten"
  - "Python-Session schließen"
-  Die Palette "Konvertierung" enthält die neue Funktion "Typ erzwingen". Die Funktion wandelt die Eingangsdaten unter Beibehaltung ihres Werts in den angegebenen Datentyp um. Im Gegensatz

zur Funktion "Typumwandlung" ist bei dieser Funktion ein Uminterpretieren der Eingangsdaten ausgeschlossen. Verwenden Sie diese Funktion in den folgenden Fällen:

- zum Eliminieren eines Typumwandlungspunkts
- zum Umwandeln von Daten ohne Typdefinition in eine kompatible Typdefinition (oder umgekehrt)
- zum Umbenennen von Daten an der Verbindung, z. B. von Benutzerereignisreferenzen

*[Idee wurde vom NI-Diskussionsforen-Mitglied JackDunaway eingereicht.]*

- Die Palette "Timing" enthält das neue VI "Hochauflösendes Warten (Polling)". Dieses VI pausiert die Blockdiagrammausführung um die angegebene Anzahl von Sekunden, wobei die Timer-Auflösung dieses VIs höher als bei der Funktion "Warten (ms)" ist.

## Neue und geänderte Eigenschaften und Methoden

LabVIEW 2018 umfasst die nachfolgend aufgeführten neuen und geänderten Eigenschaften und Methoden:

- Die Klasse "Linkeschieberegister" enthält die neue Eigenschaft "Ist ein Fehlerregister". Mit Hilfe dieser Eigenschaft können Sie feststellen, ob ein Schieberegister ein Fehlerregister ist. Ein Fehlerregister ist eine spezielle Form eines Schieberegisters. Fehlerregister werden in For-Schleifen erzeugt, bei denen parallele Iterationen aktiviert sind, wenn der Datentyp des Schieberegisters der Fehler-Cluster ist.
- Die Klasse "VI" enthält die neue Methode "Panel in Hauptebene als ausgeblendet konfigurieren". Verwenden Sie diese Methode, um das Frontpanel eines VIs auszublenden und das VI optional aus der Taskleiste auszublenden, wenn das VI als Haupt-VI ausgeführt wird. Diese Methode kann beispielsweise dazu dienen, das Frontpanel von Start-VIs von in LabVIEW erstellten eigenständigen Anwendungen auszublenden.
- Die Klasse "Deaktivierungsstruktur" enthält die neue Eigenschaft "Deaktivierungsart". Mit dieser Eigenschaft kann ermittelt werden, ob es sich bei der Struktur um eine Diagrammdeaktivierungsstruktur, eine bedingte Deaktivierungsstruktur oder um eine Typspezialisierungsstruktur handelt.
- Der Parameter **Deaktivierungsart** der Methode "Deaktivierungsart ändern" (Klasse: "Deaktivierungsstruktur") umfasst die neue Option **Type Specialization Style**. Mit dieser Option können Sie die Deaktivierungsstruktur in eine Diagrammdeaktivierungsstruktur, eine bedingte Deaktivierungsstruktur oder eine Typspezialisierungsstruktur umwandeln.

## Änderungen und Neuerungen in vorherigen LabVIEW-Versionen

---

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

- *Hinweise zum Upgrade auf LabVIEW 2014*—[upnote14de](#)
- *Hinweise zum Upgrade auf LabVIEW 2015*—[upnote15de](#)
- *Hinweise zum Upgrade auf LabVIEW 2016*—[upnote16de](#)
- *Hinweise zum Upgrade auf LabVIEW 2017*—[upnote17de](#)

Weitere Informationen zu Marken von NI finden Sie in den *NI Trademarks and Logo Guidelines* auf [ni.com/trademarks](http://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 NI 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](http://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 NI sowie zu Bezugsquellen für relevante HTS-Codes, ECCNs und andere Import-/Exportangaben finden Sie auf [ni.com/legal/export-compliance](http://ni.com/legal/export-compliance) unter der Überschrift *Export Compliance Information*. NI übernimmt weder ausdrücklich noch stillschweigend irgendeine Gewährleistung für die Richtigkeit oder Vollständigkeit der in diesem Dokument enthaltenen Informationen und haftet nicht für Fehler. Kunden aus US-Regierungsbehörden: Die in diesem Dokument enthaltenen Informationen wurden auf private Kosten entwickelt und unterliegen den anwendbaren beschränkten Rechten und beschränkten Datenrechten, die in FAR 52.227-14, DFAR 252.227-7014 und DFAR 252.227-7015 dargelegt sind.