



## Inhalt

1	Neue Version von SILworX .....	1
2	Bemerkungen zur Version 3.38.....	2
3	Verbesserungen der Version 3.x gegenüber 2.x.....	3
3.1	Unterstützung neuer Fähigkeiten von HIMax V.3.x .....	3
3.2	Multitasking .....	3
3.3	Benutzerverwaltung für SILworX-Projekte .....	3
3.4	Unterstützung von 3 GB Speicher .....	3
3.5	Veränderte Handhabung der Redundanz von E/A-Modulen .....	4
3.6	Weitere Verbesserungen .....	4
3.7	Systemanforderungen.....	4
3.8	Verbesserungen gegenüber früheren Versionen .....	5
4	Problembehebung der Version 3.x.....	6
4.1	Verbesserungen, bei denen Änderungen des CRCs möglich sind .....	6
4.2	Verbesserungen im FBS-Editor.....	7
4.3	Verbesserungen bei Kommunikation .....	8
4.4	Verbesserungen im Versionsvergleich.....	9
4.5	Weitere Verbesserungen .....	9
5	Einschränkungen der Version 3.38 .....	12
5.1	Einschränkungen beim FBS-Editor .....	12
5.2	Einschränkungen bei Online-Ansicht und -Test .....	12
5.3	Einschränkungen bei der OPC-Konfiguration .....	13
5.4	Einschränkungen bei Kommunikation .....	14
5.5	Einschränkungen bei Struktur-Datentypen.....	14
5.6	Einschränkungen bei Projekten mit HIMax Steuerungen .....	15
5.7	Weitere Einschränkungen .....	15
5.8	Grundsätzliche Einschränkungen.....	18
6	Übergang von einer Vorversion auf Version 3.38.....	20

## 1 Neue Version von SILworX

- **Wichtiger Hinweis:** HIMA empfiehlt dringend, diese Version 3.38 anstelle der Vorversionen 2.x und 3.30 einzusetzen, siehe Kapitel 2.
- Version **3.38** für:
  - HIMax Steuerungen ab Firmware Version 2.0.
  - HIMatrix Steuerungen ab CPU-Betriebssystem Version 7.0 und COM-Betriebssystem Version 12.0.

Dieses Dokument beschreibt die neuen Funktionen der Version 3.38, ihre Einschränkungen und die Verbesserungen gegenüber Vorversionen:

- Kapitel 2, 3 und 4 enthalten die neuen Funktionen und Verbesserungen.
- Kapitel 5 enthält die in V.3.38 bestehenden Einschränkungen.
- Kapitel 6 enthält die Übergangsprozedur von einer Vorversion.

## 2 Bemerkungen zur Version 3.38

- Diese Version beseitigt ein sicherheitskritisches Problem der Vorversionen V.2.36, V.2.46, V.2.52 und V.3.30, das aber nur in sehr seltenen Fällen auftreten kann:

Der Compiler kann die Stackgröße zu niedrig berechnen und somit zuwenig Speicher anlegen. Das Problem wurde beobachtet, wenn auf das Anwenderprogramm oder von diesem aufgerufene Bausteine folgendes zutraf:

- Programm oder Baustein rufen sehr viele Instanzen von Funktionen oder Bausteinen auf.
- Die aufgerufenen Instanzen haben viele Eingänge.

Das Problem führt in den meisten Fällen, in denen es auftritt, zum Fehlerstopp des Anwenderprogramms.

Die Generierung mit der Version V.3.38 kann auch für die Validierung der Projekte aus den Vorversionen genutzt werden. Ergibt die Generierung mit V.3.38 denselben CRC wie die Vorversion, so war das Projekt vom obengenannten Fehler nicht betroffen. [HE17005]

### 3 Verbesserungen der Version 3.x gegenüber 2.x

Dieses Kapitel benennt die Verbesserungen und Erweiterungen der Hauptversion 3 gegenüber der Hauptversion 2.

#### 3.1 *Unterstützung neuer Fähigkeiten von HIMax V.3.x*

SILworX unterstützt die folgenden zusätzlichen Fähigkeiten von HIMax V.3.x:

- Multitasking
- Neue E/A-Module
- Neue Kommunikationsprotokolle
- Reload von E/A-Modulen

#### 3.2 *Multitasking*

In einer HIMax Steuerung mit einem Betriebssystem ab V.3.x können bis zu 32 Anwenderprogramme gleichzeitig ablaufen. SILworX enthält Erweiterungen zur Planung und Bedienung dieser Programme:

- Multitasking-Editor
- Erweiterte Einstellungen bei Ressource und Programm
- Erweiterungen bei Online-Ansichten: Control Panel, Online-Detailansicht, Force-Editor

#### 3.3 *Benutzerverwaltung für SILworX-Projekte*

Es ist möglich, in SILworX eine Benutzerverwaltung zu definieren, so dass nur berechtigte Benutzer ein Projekt bearbeiten können. Diese Benutzerverwaltung kann mit der Benutzerverwaltung der Steuerung zusammenarbeiten, so dass ein Benutzer sich beim Verbindungsaufbau zur Steuerung nicht erneut authentifizieren muss.

#### 3.4 *Unterstützung von 3 GB Speicher*

Die Version 3.x kann 3 GB virtuellen Speicher verwalten. Dadurch ist diese Version in der Lage, sehr große Projekte zu kompilieren und zu drucken. Die 32-bit-Versionen der Betriebssysteme Windows XP und Vista müssen aber dafür konfiguriert werden, dass sie SILworX bis zu 3 GB virtuellen Speicher zur Verfügung stellen.

Windows XP:

- In der Datei boot.ini unter "[operating systems]" die Zeile, die "WINDOWS" enthält, um die Option "/3GB" erweitern.

Windows Vista:

- Boot Configuration Data (BCD) editieren:  
BCDEDIT /Set increaseUserVa 3072.

Hinweis: diese Option wird nur in sehr seltenen Fällen benötigt.

### **Hinweise zur Benutzung von zusätzlichem Speicher**

Die Vergrößerung des Speichers für SILworX verkleinert den für das Betriebssystem verfügbaren Speicher. Dies kann dazu führen, dass Windows einige Treiber nicht mehr lädt oder sogar nicht mehr startet.

Aus diesem Grund sollte der verwendete PC möglichst nur Geräte enthalten, die mit Standardtreibern benutzt werden können.

Gleichzeitig mit SILworX sollten möglichst wenige weitere Programme ablaufen. Diese sollten wenig speicherintensiv sein.

Falls Probleme dieser Art auftreten, ist es möglich, SILworX weniger als 3 GB Speicher zur Verfügung zu stellen.

Windows XP:

- In der Datei boot.ini unter "[operating systems]" die Zeile, die "WINDOWS" enthält, um die Option "/UserVa=x" erweitern, wobei  $2048 < x < 3072$ .

Windows Vista:

- Boot Configuration Data (BCD) editieren:  
BCDEDIT /Set increaseUserVa x  
wobei  $2048 < x < 3072$ .

### **3.5 Veränderte Handhabung der Redundanz von E/A-Modulen**

Um E/A-Module redundant zu konfigurieren, ist es ab Version 3.x notwendig, zu einem vorhandenen E/A-Modul ein redundantes hinzuzufügen. Die beiden E/A-Module bilden eine Redundanzgruppe. Das redundante Modul kann dabei beliebig platziert werden. Eine Tabelle "Redundanz" in der Objektauswahl liefert einen Überblick über die vorhandenen Redundanzgruppen und ihre Mitglieder.

### **3.6 Weitere Verbesserungen**

- Anzeige des belegten Speichers im Versionsvergleich
- Ersetzen von Variablen durch Drag&Drop verbessert
- Ersetzen von POEs durch Drag&Drop verbessert
- "Gehe zu..." bei Fehlermeldungen
- Anzeige nicht belegter POE-Ausgänge in Auto OLT-Feldern

### **3.7 Systemanforderungen**

Der zur Installation von SILworX verwendete Rechner muss den folgenden Systemanforderungen genügen:

Minimale Anforderungen

- Intel Pentium 4
- 256 MB verfügbarer RAM-Speicher
- 500 MB verfügbarer Festplattenspeicher
- Bildschirmauflösung 1024x768 Pixel
- Ethernet-Schnittstelle
- Windows XP Professional, mindestens SP2
- Windows Vista Ultimate / Business
- (Zu den Systemanforderungen für Windows siehe:  
<http://support.microsoft.com/kb/919183/en-us>)

Empfohlene Anforderungen:

- Intel® Core™ 2 Duo
- 4 GB RAM
- Physical Address Extension (PAE) aktiviert (Benutzung von 3 GB Benutzerspeicher)
- 10 GB verfügbarer Festplattenspeicher
- Bildschirmauflösung 1600x1200 Pixel (true color)
- Ethernet-Schnittstelle
- Windows XP Professional SP3 oder
- Windows Vista Ultimate / Business

### ***3.8 Verbesserungen gegenüber früheren Versionen***

Die Erweiterungen und Verbesserungen der Version 3 gegenüber den Vorversionen V.1.12 und V.1.20 sind den Freigabenotizen zu den Versionen V.2.36 und V.2.46 zu entnehmen. Die Migrationsprozedur von den Versionen V.1.12 und V.1.20 zur V.3.38 ist dieselbe wie in den Freigabenotizen zu den Versionen V.2.36 und V.2.46 beschrieben.

## 4 Problembehebung der Version 3.x

*Dieses Kapitel nennt Probleme der Vorversionen, die in der Version 3.x behoben sind.*

*Der erste Abschnitt listet die Probleme auf, deren Behebung in konvertierten Projekten zu einer Änderung des CRC führen kann.*

### 4.1 Verbesserungen, bei denen Änderungen des CRCs möglich sind

*Die Behebung in V.3.x der in diesem Abschnitt genannten Probleme der Vorversion(en) kann dazu führen, dass sich in konvertierten Projekten der CRC ändert.*

- **Attribut "Retain" bei Funktionen**  
Es ist möglich, eine Funktion mit dem Attribut "Retain" zu versehen. In Vorversionen wurde dies vom Compiler akzeptiert und kompiliert. In V.3.x weist der Compiler dies als Fehler zurück, so dass eine Korrektur notwendig ist. Dadurch ändert sich der CRC. Die Änderung lässt sich per Reload laden. [HE15744]
- **Keine Zustandsvariablen-Parametrierung**  
In Vorversionen wurde bei der Ereignisparametrierung in folgenden Fällen keine Zustandsvariablen-Parametrierung durchgeführt:
  - Wenn als Event-Quelle "CPU" ausgewählt war.
  - Wenn alle Grenzwerte und Hysteresewerte auf 0 gesetzt waren.Die V.3.x führt auch in diesen Fällen eine Zustandsvariablen-Parametrierung durch, so dass die Zustandsvariablen berechnet werden. Dies kann den CPU-Zyklus belasten. Durch diese Korrektur ändert sich der CRC. Die Änderung lässt sich per Reload laden. [HE15168, HE15170]
- **Nicht-redundanter E/A-Kanal kann zusätzlich redundant belegt werden**  
Ist bei einem redundanten E/(A-Modulpaar ein Kanal nicht redundant definiert, so war es bei Vorversionen dennoch möglich, diesem Kanal im Formular für die redundanten Kanäle eine globale Variable zuzuweisen. SILworX verhinderte auch nicht die Aufhebung der Kanal-Redundanz bei einem Kanal, dem bereits eine globale Variable zugewiesen war. Bei der Codegenerierung erzeugte die Vorversion keine Fehlermeldung, ignorierte aber die redundante Belegung. In V.3.x führt eine solche Belegung zu einer Warnung bei der Codegenerierung. Es ist möglich, diese Konfiguration und damit den CRC beizubehalten. Hinweis: Zur Änderung der Belegung muss der Kanal wieder redundant definiert werden. Die Änderung lässt sich per Reload laden, wenn der Kanal nicht zu einem SER-Modul gehört. [HE15558]
- **CRC über Initialwert von benutzerdefinierten Datentypen**  
In Vorversionen umfasste der CRC von benutzerdefinierten Datentypen nicht den Initialwert. In der V.3.x umfasst der CRC auch den Initialwert benutzerdefinierter Datentypen. [HE15821]

- Speicherbedarf des Programms  
In der Vorversion wurde der Speicherbedarf des Programms in seltenen Fällen zu groß berechnet. Die korrekte Berechnung in der V.3.38 kann zu einer CRC-Änderung führen, wenn eine/r der unten aufgeführten Funktionen/Bausteine mit dem Eingang EN verwendet wird. Auch dann ist eine CRC-Änderung unwahrscheinlich.
  - CTU, CTD, CTUD
  - ROL, ROR
  - ABS, CHS, LIMIT
  - SQRT, EXP, EXPT, LN, LOG
  - ASIN, ACOS, ATAN, SIN, COS, TAN
  - CHK\_REAL, TRUNC
  - DIV
  - UNPACK2, UNPACK4, UNPACK8[HE16094]

### 4.2 Verbesserungen im FBS-Editor

- Eigenschaft Retain bei Instanzen von Funktionsbausteinen  
Waren in Vorversionen die Interface-Variablen vom Typ VAR\_OUT bei einem Funktionsbaustein auf Retain = TRUE gesetzt, so konnten sie bei einer Instanz dieses Funktionsbausteins auf Retain = FALSE gesetzt werden. Die geänderte Retain-Eigenschaft wurde auf die VAR\_OUT übernommen und änderte die vom POE-Typ vererbte Eigenschaft. Dies hatte aber keine Auswirkungen auf den generierten Code.  
In der V.3.x lässt sich die Retain-Eigenschaft bei Instanzen weiterhin ändern, hat aber keine Auswirkung mehr auf die VAR\_OUT. [HE14182]
- Auffrischung der Anzeige der Abarbeitungsreihenfolge  
In Vorversionen wurde nach einer Änderung an der Logik im Funktionsbaustein-Editor wird die Abarbeitungsreihenfolge so angezeigt, wie sie vor der Änderung gültig war. Die V.3.x löscht die Anzeige der Abarbeitungsreihenfolge bei Änderungen der Logik, so dass der Anwender diese Anzeige - falls gewünscht - neu aufschalten muss. [HE13841]
- Prüfung der Prioritäten von Auswahlverzweigungen in der Ablaufsprache (AS)  
Bei einer AS-Auswahlverzweigung (Zwei Transitionen nach einem Schritt) müssen folgende Bedingungen gelten:
  - Entweder müssen alle Transitionen eine manuell vergebene Priorität haben, oder alle müssen eine automatisch vergebene Priorität haben.
  - Diese Prioritäten müssen unterschiedlich sein.Vorversionen prüften diese Bedingungen bei der Kompilierung nicht. V.3.x prüft die Bedingungen. [HE13911]
- Wert interner Wertfelder  
In Vorversionen nahmen interne Wertfelder von Funktionsbausteinen den eingetragenen Wert nicht an. Stattdessen übernahmen sie den Initialwert aus der Typdefinition. In der V.3.x nehmen sie den eingetragenen Wert an. Die Konver-

tierung nach V.3.x entfernt alle eingetragenen Werte interner Wertfelder, um eine Änderung des CRCs und der Programmsemantik zu verhindern. [HE15544]

- Unsichtbare Rückkopplungsschleifen  
Direktes Verbinden eines Ausgangs mit einem Eingang erzeugte in Vorversionen eine unsichtbare Rückkopplungsschleife, wenn SILworX kein geeignetes Routing finden konnte. Wurde z. B. zu einer Variablen ein Online-Testfeld hinzugefügt, so konnte das Ziehen einer Verbindungslinie zwischen den beiden Anschlusspunkten eine unsichtbare Verbindungslinie erzeugen. Die Verbindungslinie wurde erst nach Löschen der Variablen sichtbar. Die V.3.x zeigt die Verbindungslinie im Vordergrund an. [HE15365]

### 4.3 Verbesserungen bei Kommunikation

- Prüfung der Datenmenge bei MODBUS-Funktionscode 23  
Die Prüfung auf die zulässigen Puffergrößen bei MODBUS-Funktionscode 23 arbeitete in Vorversionen fehlerhaft:
  - Untergrenze für den Lesebuffer wurde nicht geprüft
  - Obergrenze des Schreibpuffers ist 242 Bytes und nicht 240 Bytes, wie die Fehlermeldung angab.
  - War bei einem Funktionscode die maximale Datenmenge überschritten, dann blieb die Fehlermeldung erhalten, auch wenn die Daten auf die zulässige Menge verringert wurden.

Das Problem ist in V. 3.x behoben. [HE12899]

- Löschen eines P2P-verbundenen Signals mit benutzerdefinierten Datentypen  
Vorversionen terminierten beim Löschen einer globalen Variable mit folgenden Eigenschaften:
  - Benutzerdefinierter Datentyp
  - Datenaustausch mit anderer Ressource über **safeethernet**

Das Problem ist in V. 3.x behoben. [HE14271]

- Import von exportierten Proxy-Ressourcen  
Vorversionen konnten die Exportdateien (PRS-Dateien) von Proxy-Ressourcen nur dann importieren, wenn beim Anlegen der exportierten Proxy-Ressourcen die Receive- und Resend-Timeouts auf die Standardwerte gesetzt waren. Das Problem ist in V.3.x behoben. [HE15226]

- Hinzufügen von CUT (ComUserTask) pro Ressource  
Vorversionen erlaubten nur das Hinzufügen einer einzigen CUT zu den Protokollen einer Ressource, obwohl eine CUT pro Kommunikationsmodul existieren kann. Das Problem ist in V.3.x behoben. [HE15624]

- Einstellungen für das **safeethernet** für ein OPC-Server-Set  
Wurden bei der Konfiguration für OPC-Server die Parameter "Priorität Ereignisse" und "Priorität Zustandswerte" innerhalb der **safeethernet**-Einstellungen verändert, wurde in Vorversionen die Konfiguration inkonsistent. Das Problem ist in V. 3.x behoben. [HE14664]



## Programmierwerkzeug SILworX V.3.x

- Codegenerierung für OPC-Server  
Die Codegenerierung für den OPC-Server terminierte in Vorversionen, wenn mehr als eine View verwendet wurde.  
Das Problem ist in V. 3.x behoben. [HE15331]
- Globale Variable in Objektauswahl  
In Vorversionen war es möglich, in der Objektauswahl des safeethernet-Editors für OPC-Server-Sets durch Drücken der Taste "Entf" und anschließender Bestätigung mit "OK" globale Variable zu löschen. Diese Variablen waren dann aus dem Projekt gelöscht und mussten neu angelegt werden, falls nötig.  
Das Problem ist in V. 3.x behoben. [HE14575]

### 4.4 Verbesserungen im Versionsvergleich

- Im Versionsvergleich angezeigte "Programm-Informationen"  
In Vorversionen zeigte der Versionsvergleich unplausible Daten als "Programm-Informationen".  
Die V.3.x zeigt korrekte Daten an und berücksichtigt Multitasking. [HE14533]
- Auflistung eingebetteter Datentypen im Versionsvergleich  
In Vorversionen fehlten in der Auflistung des Versionsvergleichs die "eingebetten" Datentypen, also diejenigen Datentypen, die nur bei Elementen von Arrays oder Strukturen verwendet waren.  
Das Problem ist in V. 3.x behoben. [HE16248]
- Neue Elemente im Versionsvergleich als geändert darstellen  
Vorversionen von SILworX markierten im Versionsvergleich nur gegenüber der alten Projektversion geänderte Elemente (Module, POEs), nicht aber neu hinzugefügte Elemente.  
Das Problem ist in V. 3.x behoben. [HE14685]

### 4.5 Weitere Verbesserungen

- Änderung des Modulnamens eines redundanten Moduls  
Bei Änderung des Modulnamens eines redundant verknüpften Moduls gerieten Vorversionen in eine Endlosschleife. Dies wird in V.3.x vermieden. [HE14738]
- Kompilierung von Remote I/Os  
Der Compiler von Vorversionen terminierte bei der Übersetzung der Parameter für Remote I/Os, wenn alle folgenden Bedingungen erfüllt waren:
  - Ein benutzerdefinierter Datentyp (Struct) war definiert.
  - Innerhalb der Parameter für ein Remote I/O gab es mindestens zwei globale Variable dieses Datentyps.
  - Dasselbe Unterelement bei mindestens zwei dieser Variablen wurde benutzt.Das Problem ist in V. 3.x behoben. [HE15607]
- Speicherbedarf beim Drucken von Dokumentation  
Vorversionen konnten nach dem Drucken von Dokumentation beim Starten der Codegenerierung SILworX mit der Fehlermeldung "out of memory" terminieren. V.3.x terminiert in solchen Fällen nicht.

## Programmierwerkzeug SILworX V.3.x

Hinweis: Zur Verwendung von 3 GB Speicherplatz siehe Kap.3.4. [HE15653, HE15676]

- Codegenerierung bei SER Modulen mit E/A-Ereignissen  
In Vorversionen brach der Codegenerator mit Fehler ab, wenn das Projekt SER-Baugruppen enthielt, bei denen E/A-Ereignisse konfiguriert waren. Das Problem ist in V. 3.x behoben. [HE15323]
- Uneingeschränkte Benutzung von SILworX ohne Administrator-Rechte  
Bei Verwendung von SILworX mit einem Windows-Benutzerkonto ohne Administrator-Rechte traten in Vorversionen folgende Probleme auf:
  - Keine Lizenz trotz vorhandener Lizenzierung bei Windows XP. Nur der Demo-Modus war möglich.
  - Die Logdatei des SILworX-Logbuchs wurde bei Windows XP nicht geschrieben.
  - SILworX konnte die INI-Datei kann nicht beschreiben. Deshalb konnte es keine Einstellungen speichern, bei Windows XP und Vista.
  - Bei Verwendung eines U3-USB-Sticks für die Lizenzierung terminierte der OLicenseServer beim Starten. Dadurch hatte SILworX keine gültige Lizenz! Hierfür gab es unter Windows XP keine Abhilfe!

Das Problem ist in V. 3.x behoben. [HE 14880]

- Performance  
In V.3.x ist die Performance an vielen Stellen gegenüber Vorversionen verbessert:
  - Öffnen von Editoren [HE14717, HE14829]
  - Variablen-Import [HE12742, HE13203]
  - Archivieren einer Ressource [HE11954]
  - FBS-Editor [HE13720]
  - Editor für globale Variable [HE14888, HE14911]
  - Löschen von Objekten, z.B. Logik [HE14014]
  - Signaltabelle der safe**ethernet**-Verbindungen für OPC [HE14378]
  - Große Arrays in verschiedene Ressourcen [HE14555]
  - Kopieren von vielen Signalen im Alarm&Event-Editor [HE14646]
  - Codegenerierung [HE15413, HE15648]
  - Wiederherstellen von Bibliotheken [HE15650]
  - Import/Export von Bausteinen [HE15430]
  - Import von Variablen bei Protokollen und Ereignisdefinitionen [HE14874]
- Behandlung von nicht beschriebenen Variablen bei Reload  
Vorversionen behandelten nicht beschriebene Variable unterschiedlicher Typen beim Reload unterschiedlich:
  - Globale Variable, die nicht CONST waren, wurden auf den Initialwert gesetzt.
  - Lokale Variable, die nicht CONST waren, wurden nicht auf den Initialwert gesetzt.

V.3.x setzt nicht beschriebene lokale und globale Variable auf den Initialwert. Ausnahme: V.3.x setzt die Elemente einer globalen Variablen mit Struktur- oder Array-Datentyp nicht auf den Initialwert, wenn eines der Elemente im Anwenderprogramm beschrieben wurde. [HE12058]

- Teilweises Beschreiben strukturierter Variablen  
Vorversionen erkannten das Beschreiben einzelner Elementen von Struktur- und Array-Variablen nicht als Beschreiben der Variable. Dadurch behandelt Reload solche Variablen als unbeschrieben und setzt sie auf den Initialwert zurück.  
Das Problem existiert so nicht mehr in V.3.x, siehe Ausnahme bei HE12058. [HE15560]
- Arrays und Strukturen in Editoren  
Vorversionen konnten bei der Verwendung von Array- und Struktur-Variablen in den Editoren nicht auf einzelne Elemente der Variablen zugreifen (außer dem Globale-Variablen-Editor). Die Elemente mussten als Text eingegeben werden. In der V.3.x ist es in der Objektauswahl möglich, auf die einzelnen Element der Variablen zuzugreifen und diese durch Drag&Drop in den Editor zu ziehen. [HE13722]
- PDF-Datei als Online-Hilfe für benutzerdefinierte POEs  
In Vorversionen war es nicht möglich, als Online-Hilfe für eine benutzerdefinierte POE eine PDF-Datei einzubinden.  
In V.3.x ist eine PDF-Datei als Online-Hilfe für benutzerdefinierte POEs verwendbar. [HE14246]
- Verwendung globaler Variable als Index  
Vorversionen erkannten nicht, dass die Verwendung einer globalen Variablen als Array-Index ein Lesezugriff ist, sowohl in der Querverweisliste als auch bei der Codegenerierung. Dadurch konnte das Forcen dieser Variablen ohne Wirkung sein.  
Die V.3.x erkennt, dass die Verwendung globaler Variable als Index ein Lesezugriff ist. [HE14353]
- "Gehe Zu" aus der Querverweisliste des Globale Variablen-Editors  
In Vorversionen funktionierte "Gehe Zu..." aus der Querverweisliste des Editors nicht für globale Variable, die in einem Protokoll verwendet wurden.  
Das Problem ist in V. 3.x behoben. [HE14410]
- Sprachabhängigkeiten bei Systemvariablen  
In Vorversionen war es wegen Sprachabhängigkeiten bei den Systemvariablen möglich, dass nach einer Sprachumschaltung von SILworX der Versionsvergleich der Ungleichheiten anzeigte und ein Reload des Projekts verweigert wurde.  
In V.3.x sind die Sprachabhängigkeiten beseitigt. [HE15036]
- Nicht existierende **safeethernet**-Variablen führen in V.3.38 zu Fehlermeldungen bei der Codegenerierung  
Löschen einer globalen Variablen, die als **safeethernet**-Variable benutzt wird, führte in den Vorversionen nicht zu einer Fehlermeldung bei der Codegenerierung.  
SILworX V.3.38 generiert Fehlermeldungen für solche Variable. [HE17625]

## 5 Einschränkungen der Version 3.38

*Beim Einsatz von SILworX mit der Version 3.38 sind die folgenden Einschränkungen zu beachten.*

*Bei Beachtung der folgenden Hinweise haben die Einschränkungen keine Auswirkungen auf die Sicherheit und die Verfügbarkeit des Codes, der für eine HIMax Steuerung generiert wird.*

### 5.1 Einschränkungen beim FBS-Editor

- Ausschneiden & Einfügen von Eingangskonnektoren "zerstört" die Logik  
Das Ausschneiden und Einfügen eines Eingangskonnektors führt dazu, dass der Eingangskonnektor seinen Standardnamen annimmt und der zugehörige Ausgangskonnektor verschwindet.  
Abhilfe: statt Ausschneiden & Einfügen Kopieren & Einfügen benutzen, anschließend ursprünglichen Eingangskonnektor löschen. [HE11563]
- SILworX terminiert nach Kopieren und einfügen  
SILworX terminiert nach dem Kopieren eines Logik-Netzwerks, wenn alle folgenden Bedingungen zutreffen:
  - Das Netzwerk enthält ein Wertfeld.
  - Für das Wertfeld existiert ein OLT-Feld und ein zugeordneter Kommentar.Eine weitere Aktion, wie Verschieben der Variablen, führt dann zum Terminieren.  
[HE17261]

### 5.2 Einschränkungen bei Online-Ansicht und -Test

- Wiederverbinden der Diagnoseansicht bewirkt Modul-Login statt System-Login  
Wird bei bestehendem System-Login die Diagnoseansicht geöffnet und die Verbindung getrennt, dann wird beim Versuch, die Verbindung neu aufzubauen, das Modul-Login angeboten.  
Abhilfe: Stellen Sie Verbindungen zum System entweder über die Hardware-Online-Ansicht oder das Control-Panel her.  
Wenn das Modul-Login geöffnet war, schließen Sie alle Ansichten der Baugruppe: Diagnoseansicht und Baugruppenansicht. [HE11926]
- Anzeige des Werts von Array-Elementen im Online-Test nicht möglich  
SILworX kann im Online-Test den Wert einzelner Array-Elemente nicht anzeigen.  
Abhilfe: Force-Editor zur Anzeige der Werte von Array-Elementen benutzen.  
[HE14990]
- Selektion mehrerer Module oder Remote IOs bei Online-Ansicht des Hardware-Editors  
Die Selektion mehrerer Module oder Remote IOs (Multiselektion) arbeitet nicht korrekt. Der Versuch, das Kontextmenü für mehrere selektierte Module oder Remote IOs aufzuschalten, hat zum Ergebnis, dass sich das Kontextmenü irgendeines dieser Objekte öffnet. Fehlbedienungen sind dadurch leicht möglich,

besonders, da nach dem Öffnen der Online-Ansicht alle Remote IOs selektiert sind.

Abhilfe: Nach dem Öffnen der Onlineansicht gezielt das zu bearbeitende Objekt anklicken. [HE16851]

- Fehlerhafte Darstellung beim Online-Test eines nicht geladenen (kopierten) Projekts

Beim Benutzen einer Kopie des in die Steuerung geladenen Projekts für den Online-Test können Objekte fehlerhaft dargestellt werden. Beispiele:

- Falsche Farbe der Linien, die digitale Werte darstellen (Powerflow).
- Online-Test-Felder sind leer.

Außerdem können in diesen Fällen Warnmeldungen erfolgen wie "Die angezeigten Variablennamen können veraltet sein, da sie sich auf eine Ressourcenkonfiguration beziehen, die denselben CRC, aber nicht dieselbe laufende Nummer hat wie die gegenwärtig geladene."

Dies geschieht dann, wenn für den Online-Test nicht das Projekt verwendet wird, das in die Steuerung geladen ist, sondern eine Kopie davon (z.B. Sicherheitskopie).

Abhilfe: Für den Online-Test das Original-Projekt verwenden, falls verfügbar. [HE17649]

- Force-Zustand einer teilweise sichtbaren Variablen wird nicht aktualisiert  
Der Zustand des Force-Schalters einer Variablen wird nicht aktualisiert, wenn diese nur teilweise sichtbar ist.

Auch der Force-Dialog aktualisiert den Zustand des Force-Schalters nicht.

Abhilfe: Bildausschnitt so verschieben, dass die Variable, das Wertfeld oder ein zugehöriges OLT-Feld komplett sichtbar ist. [HE17872]

### 5.3 *Einschränkungen bei der OPC-Konfiguration*

- Mehr als vier OPC-Server je Ressource konfigurierbar  
Es ist möglich, je Ressource mehr als vier OPC-Server für Alarme und Ereignisse zu konfigurieren, und erfolgreich zu kompilieren. Vier dieser OPC-Server können sich auch sicher mit der Ressource verbinden, die übrigen haben Verbindungsabbrüche.  
Abhilfe: Manuell sicherstellen, dass nicht mehr als vier OPC-Server konfiguriert sind. [HE14543]

- Strukturierte globale Variable und OPC-Server  
Der Force-Editor kann in folgenden Fällen für strukturierte globale Variable, die mit dem OPC-Server ausgetauscht werden, einen falschen Variablen-Pfad anzeigen:

- Nach dem Umbenennen der strukturierten globalen Variable.
- Beim Referenzieren anderer globaler Variablen.

[HE16768]

## 5.4 Einschränkungen bei Kommunikation

- Strukturen im Modbus-Slave  
Es ist nicht möglich, globale Variable mit Strukturen als Datentyp im Modbus-Slave zu übertragen, da für die Elemente keine korrekten Offsets vergeben werden.  
Abhilfe: einzelne Elemente in den Modbus-Slave übernehmen. [HE16360]
- PROFINET: ARUIDs nach dem Kopieren eines Controllers  
Nach dem Kopieren eines Controllers bei der PROFINET-Konfiguration kann der Fall eintreten, dass die ARUIDs nicht mehr eindeutig sind.  
Abhilfe: Nach dem Kopieren des Controllers die Devices in der Kopie löschen und einzeln kopieren. [HE16453]
- Verwendung von strukturierten globalen Variablen im safeethernet-Editor  
Der safeethernet-Editor übernimmt von einem eingefügten Element einer strukturierten globalen Variablen nur den Elementnamen, so dass der Bezug zur Variablen nicht mehr ersichtlich ist. Das Einfügen eines weiteren Elements mit gleichem Namen ist nicht möglich. [HE16460]
- Variablen-Import in safeethernet-Editor  
Die Import-Funktion für Variable im safeethernet-Editor arbeitet nur dann korrekt, wenn der Editor keine Variablen enthält.  
Abhilfe: Nur dann Variablen importieren, wenn der Editor keine Variablen enthält. [HE16407]
- Beim Import von Modbus-Registern in den Slave falsche Bitadressenvergabe  
Beim Importieren von Modbus-Registern vergibt SILworX bei Datentypen außer BOOL um 8 Bit zu hohe Bitadressen.  
Abhilfe: In der zu importierenden CSV-Datei Adressen in der Form Register.Bit angeben oder *Neue Offsets* erzeugen. [HE15056]

## 5.5 Einschränkungen bei Struktur-Datentypen

- Elemente von Struktur-Variablen als Index  
Es ist nicht möglich, Elemente von Variablen eines Struktur-Datentyps als Index eines Arrays zu benutzen. [HE16159]
- Verschiedene Elemente einer Struktur-Variablen nicht gleichzeitig von unterschiedlichen Quellen beschreibbar  
Es ist nicht möglich, dass sowohl das Anwenderprogramm als auch die Hardware oder die Kommunikation zu gleicher Zeit zwei unterschiedliche Elemente derselben Struktur-Variablen beschreiben.  
Abhilfe: Unterschiedliche Struktur-Variablen für Elemente, in die das Anwenderprogramm schreibt, und für Elemente, in die Hardware oder Kommunikation schreiben. [HE15700]
- Zuordnungen zu Struktur-Variablen und (Namens-)Änderungen  
Änderungen an einem Struktur-Datentyp, z. B. Umbenennen von Elementen, führen dazu, dass Zuordnungen in der Hardware oder in Protokollen zu Variablen dieses Datentyps gelöscht werden. [HE15095]

- Initialwerte von Strukturen  
Inkonsistentes Verhalten bei Initialwerten von Strukturen ist an folgendem Beispiel zu erkennen:  
Definieren eines Datentyps Strukt\_1 mit den Elementen A und B, beide vom Typ BOOL. Zuweisen des Initialwerts TRUE zu Element A. Definieren einer globalen Variablen vom Typ Strukt\_1.  
Dann verhalten sich SILworX und das Zielsystem folgendermaßen:
  - Zur Laufzeit nimmt das Element A der globalen Variablen den Initialwert TRUE korrekt an.
  - Der Globale Variablen-Editor zeigt den Initialwert nicht an.
  - Der Compiler erzeugt eine Warnung, dass A keinen Initialwert hätte. [HE16884]
- Ändern der Reihenfolgenummer von Strukturelementen  
Ändern der Reihenfolgenummer von Strukturelementen nach dem Zuweisen globaler Variable führt bei der Codegenerierung dazu, dass die Initialwerte der Strukturelemente vertauscht sind. [HE17032]

### 5.6 Einschränkungen bei Projekten mit HlMax Steuerungen

- SILworX terminiert nach dem Auflösen einer Gruppe redundanter E/A-Module  
SILworX terminiert beim Speichern im Hardware-Editor, wenn folgende Voraussetzungen vorliegen:
  - Die Online-Ansicht ist geöffnet.
  - Ein Mitglied einer Redundanzgruppe von E/A-Modulen oder die gesamte Redundanzgruppe wurde gelöscht.Abhilfe: Vor dem Speichern die Online-Ansicht schließen. [HE17330]
- Löschen von Basisträgern zerstört Redundanzgruppen von E/A-Modulen  
Beim Löschen von Basisträgern einschließlich der gesteckten Module prüft SILworX nicht, ob die Module mit Modulen auf anderen Basisträgern Redundanzgruppen bilden. Dadurch bleiben im Projekt zerstörte Redundanzgruppen zurück, wodurch das Projekt beschädigt ist.  
Abhilfe: vor dem Löschen von Basisträgern prüfen, ob darauf befindliche Module Redundanzgruppen bilden. Diese Redundanzgruppen ggfs. auflösen. [HE17020]
- Nach Löschen von Basisträgern keine korrekte Neuerstellung von Basisträgern möglich  
Werden mehrere Basisträger nicht in fortlaufender Reihenfolge selektiert und dann gelöscht, so kann in dieser Projektdatei kein neuer Basisträger mehr angelegt werden. Abhilfe: Basisträger nur einzeln löschen. [HE17337]

### 5.7 Weitere Einschränkungen

- Online-Hilfe einer POE nicht ausdrückbar  
Das Dokumentenmanagement kann den Inhalt der Online-Hilfe einer benutzerdefinierten POE nicht ausdrucken. [HE14244]

## Programmierwerkzeug SILworX V.3.x

- Versionsvergleich "Modulvergleich" liefert falsche Informationen  
Beim Modulvergleich liefert der Versionsvergleich für die Kanäle des digitalen Eingangsmoduls X-DI 32 02 die Bezeichnungen "AI-Kanal...", da dieses Modul intern analog arbeitet. [HE14494]
- Hardware-Editor kann keine Basisträger kopieren  
Der Hardware-Editor kann Basisträger weder einzeln noch in Gruppen kopieren.  
Abhilfe: Neuen Basisträger anlegen und Module einzeln kopieren. [HE15972]
- Aufruf des Dokumentations-Editors dauert zu lange.  
Beim Aufschalten des Dokumentations-Editors bereitet SILworX zunächst die Dokumentation ohne eine für den Anwender erkennbare Reaktion auf, und öffnet erst danach den Editor. Bei großen Projekten kann diese Wartezeit ca. 30 min dauern. Während dieser Zeit ist der PC stark belastet und SILworX nicht bedienbar. [HE16047]
- Offline-Simulation aus V.1.20 konvertierter Projekte nicht möglich  
Beim Versuch einer Offline-Simulation von Projekten, die aus V.1.20 konvertiert wurden, öffnet sich der FBS-Editor nicht.  
Abhilfe: Zwischenkonvertierung über V.2.36, 2.46, oder 2.52. [HE16255]
- In Excel geöffnete CSV-Datei nicht zeitgleich importierbar  
Es ist nicht möglich, eine in Microsoft Excel geöffnete CSV-Datei zu gleicher Zeit in SILworX zu importieren. [HE16134]
- PES-Passwortlänge bei Konvertierung SILworX V.3.x nach V.2.x  
SILworX V.3.x akzeptiert die Eingabe eines PES-Passworts mit einer Länge von 32 Zeichen. Bei der Konvertierung des Projekts in V.2.x bleibt ein solches Passwort erhalten, ebenso in der Steuerung. Da in SILworX V.3.x das Eingabefeld für das Passwort nur 31 Zeichen zulässt, ist es nicht mehr möglich, mit einem entsprechenden Benutzerkonto das Projekt zu öffnen oder sich auf der Steuerung anzumelden.  
Abhilfe: Vor der Konvertierung das Passwort auf eine Länge von  $\leq 31$  Zeichen ändern. Falls nötig, Konvertierung wiederholen. [HE16838]
- Darstellung von Systemvariablen bei Offline-Simulation  
Die Offline-Simulation zeigt den Wert von digitalen Systemvariablen des Programms in der falschen Farbe an.  
Abhilfe: Systemvariable auf Wertfeld zuweisen. Ab dem Wertfeld zeigt SILworX korrekte Farben an. [HE17033]
- Retain-Attribut von Timer-Bausteinen ist nicht sicher  
Timer-Bausteine lassen sich nicht sinnvoll mit dem Retain-Attribut einsetzen. Beim Einsatz eines Timer-Bausteins mit dem dem Retain-Attribut kann der Zeitähler nach einem Warmstart einen beliebigen Wert annehmen.  
Abhilfe: Timer-Bausteine nicht mit dem Retain-Attribut verwenden. [HE17252]
- Zugriff auf Funktionen und Funktionsbausteine anderer Ressourcen möglich  
Nach dem Verschieben eines Funktions-/Funktionsbaustein-Typs von einer Ressource zu einer anderen ist dieser bei der ursprünglichen Ressource noch verwendbar, obwohl die Ressource außerhalb des Geltungsbereichs liegt. Dies kann z. B. in folgendem Fall zu Fehlern führen:



- Ein Bausteintyp ist in einer Bibliothek der Konfiguration definiert.
- In einer Ressource A wird ein Baustein dieses Typs verwendet.
- Der Bausteintyp wird in eine andere Ressource B verschoben (durch Ausschneiden und Einfügen).

Dann referenziert die Ressource A weiterhin den ursprünglichen Bausteintyp. Ein gleichnamiger, für Ressource B gültiger Typ bleibt bei der Bearbeitung und Codegenerierung außer Betracht, wenn er nach dem Verschieben neu angelegt wird.

Abhilfe: Verschieben von Bausteintypen nur durch Kopieren, Einfügen und Löschen. [HE17533]

- Zugriff auf globale Variablen anderer Ressourcen möglich  
Bei globalen Variablen, die als VAR\_EXTERNALs in Funktionsbausteinen verwendet werden, prüft SILworX nicht den Geltungsbereich.  
Dies kann z. B. in folgendem Fall zu Fehlern führen:

- Eine globale Variable ist in einer Ressource A definiert und wird innerhalb der Ressource in einem Funktionsbaustein verwendet.
- Die globale Variable wird in eine andere Ressource B verschoben (durch Ausschneiden und Einfügen des globalen-Variablen-Containers).

Dann kompiliert SILworX die ursprüngliche Ressource A weiterhin fehlerfrei, obwohl die globale Variable nicht mehr definiert ist.

Abhilfe: Verschieben des globalen-Variablen-Containers durch Kopieren, Einfügen und Löschen. [HE17784]

- Keine Prüfung bei Auswahl- und Simultan-Knoten nacheinander  
Die Verwendung von Auswahl- und Simultan-Knoten innerhalb derselben Hierarchie führt zu Deadlocks. SILworX gibt keine Hinweise auf dieses Problem.  
Abhilfe: Auswahl- und Simultan-Knoten nicht innerhalb derselben Hierarchie verwenden. [HE17716]

- Kein Code generiert für AS-Netzwerk mit Schleife  
SILworX generiert keinen Code für Schritte, die am Eingang und am Ausgang mit derselben Transition verbunden sind und somit eine Schleife bilden.  
[HE17743]

- Dialogbox für Sicherheitsparameter zeigt immer TRUE statt aktuellen Wert  
Die Dialogbox zum Einstellen der Sicherheitsparameter eines Anwenderprogramms zeigt für den Parameter *Start erlaubt* immer den Standardwert TRUE anstelle des Wertes in der Steuerung. Dieses Verhalten kann folgende Konsequenzen haben:
  - Den Versuch, den Parameter auf TRUE zu setzen (bei alter Einstellung FALSE), beantwortet SILworX mit der Meldung "Bitte ändern Sie mindestens einen Parameter...".  
Abhilfe: Zusätzlich einen weiteren Parameter ändern und, falls erforderlich, danach wieder auf den bisherigen Wert setzen.
  - Beim Ändern eines anderen Parameters setzt SILworX den Parameter *Start erlaubt* auf TRUE, wenn er in der Steuerung den Wert FALSE hat!  
Abhilfe: Beim Ändern von Parametern immer *Start erlaubt* auf den gewünschten Wert setzen.[HE17746]

### 5.8 Grundsätzliche Einschränkungen

Die Einschränkungen in diesem Kapitel sind grundsätzlicher Art. Sie liegen in den Eigenschaften von Mitteln begründet, die zum Ablauf von SILworX benötigt werden, z. B. dem PC-Betriebssystem. Daher ist auf absehbare Zeit keine Behebung möglich.

- Stützpunkte eines Analogwerts werden als REAL gelesen  
SILworX liest die eingegebenen Werte für Stützpunkte eines Analogwerts (bei 4 mA und 20 mA) als REAL. Die Weiterverarbeitung erfolgt dagegen als LREAL. Auch im Anwenderprogramm kann LREAL benutzt werden. Die Einschränkung hat aber nur bei sehr großen oder sehr kleinen Stützpunkt-Werten Auswirkungen. [HE16388]
- Logische Verknüpfungen von BOOL-Variablen, deren Werte aus der nicht sicherheitsgerichteten Kommunikation stammen, können andere als die erwarteten Ergebnisse liefern.  
Die Ursache ist die Codierung der BOOL-Werte im externen System, die von der in HIMax abweicht.  
Es gibt zwei Möglichkeiten der Abhilfe:
  - das externe System liefert definiert nur die Werte 0 für FALSE und 1 für TRUE
  - ins Anwenderprogramm wird für alle entsprechenden BOOL-Variablen eine Korrekturschaltung eingefügt, die den Wert auf 0 bzw. 1 normiert:  
*unnormierte Variable* -> Baustein AtoByte -> Baustein AtoBOOL -> *normierte Variable* [HE13042, Restriction]
- Änderungen in einem Editor nicht speicherbar?  
Nach bestimmten Änderungen in einem Editor erscheint beim Speichern die Meldung „Die Änderungen konnten nicht gespeichert werden ...“. Nach dem Bestätigen dieser Meldung können die Änderung aber doch gespeichert werden.  
Wird danach der Editor geschlossen, und versucht, ihn wieder zu öffnen, dann erscheint die Meldung „Die angeforderten Daten werden im Moment bearbeitet“.  
Änderungen, bei denen dieses Problem auftritt, sind z.B das zyklische Vertauschen von Variablennamen ( $A \rightarrow B, B \rightarrow C, C \rightarrow A$ ).  
Abhilfe: Vertauschen von Namen vermeiden.  
SILworX ggfs. neu starten. [HE11613, Restriction]
- Schwankungen der Zykluszeit bei LREAL-Berechnungen  
Bei Berechnungen mit Variablen vom Datentyp LREAL können die Zykluszeiten stark schwanken. Für die Bemessung der Watchdog-Zeit ist es notwendig, die Zykluszeit unter realistischen Bedingungen zu bestimmen.  
[HE12115, Restriction]
- Ablaufsprache: Schritt-interner TON startet einen Zyklus später als normaler TON  
Ein Reload mit folgenden Änderungen findet statt:
  - Ein neuer Schritt wird eingefügt, der nach dem Reload unmittelbar aktiv sein soll.
  - Ein TON-Baustein mit dem Eingang fest auf TRUE wird eingefügt.

Dann startet der schritt-interne TON um einen Zyklus später als der TON-Baustein in der Programmlogik. [HE16288, Restriction]

- Benutzung von Hardlocks  
Die Möglichkeit zur Lizenzierung von SILworX mit Hilfe von Hardlocks (U3 USB-Sticks) unterscheidet sich zwischen den Betriebssystemen:
  - Unter Windows XP ist die Lizenzierung mit Hilfe von Hardlocks nur für Benutzer mit Administrator-Rechten möglich, nicht aber für Haupt- oder Standardbenutzer.
  - Unter Windows Vista ist die Lizenzierung mit Hilfe von Hardlocks für alle Benutzer möglich.

Abhilfe für Windows XP: Benutzung von Softlock-Lizenzen. [HE17056, Restriction]

- DIV\_TIME mit REAL-Typisierung meldet für Divisor := +/-INF über ENO einen Fehler  
Die Funktion DIV\_TIME aus der Standardbibliothek setzt fälschlich den Fehlerausgang ENO auf FALSE und meldet somit einen Fehler unter folgenden Bedingungen:
  - Der Eingang IN2 (Divisor) ist vom Typ REAL.
  - Der Wert von IN2 ist +/-INF. [HE15199]
- Behandlung von VAR\_INPUT-Variablen bei benutzerdefinierten Funktionsbausteinen  
SILworX behandelt VAR\_INPUT-Variablen bei benutzerdefinierten Funktionsbausteinen je nach Beschaltung der Eingänge unterschiedlich:
  - Bei Beschaltung der Eingänge mit Variablen von Standard-Datentypen übergibt SILworX den Wert der Variablen an eine baustein-lokale Kopie (Call by value). Änderungen an der ursprünglichen Variablen haben keinen Einfluss auf diese Kopie.
  - Bei Beschaltung der Eingänge mit Variablen von benutzerdefinierten Datentypen übergibt SILworX die Referenz auf die Variable (Call by reference).  
Handelt es sich bei der Variablen um eine globale Variable, dann ist zu beachten, dass diese im Funktionsbaustein verändert werden kann. Eine globale Variable kann in einem Funktionsbaustein zusätzlich als VAR\_EXTERNAL verwendet werden. Änderungen einer solchen Variablen werden beim Lesen der entsprechenden VAR\_INPUT-Variablen im Funktionsbaustein unmittelbar wirksam. Der Wert kann sich also während des Ablaufs des Funktionsbausteins ändern. [HE17740]

## 6 Übergang von einer Vorversion auf Version 3.38

Der Übergang von einer Version ab V.2.36 auf V.3.38 ist folgendermaßen durchzuführen:

- Vor der Konvertierung alle Ressourcen generieren. Dadurch lassen sich eventuelle Abweichungen bei der Generierung nach der Konvertierung feststellen.
- Vor der Konvertierung das Projekt sichern, z. B. auf einem Datenträger.
- Projekt in V.3.38 öffnen und konvertieren.
- Da die Konvertierung sehr umfangreich ist, nach der Konvertierung die Projektintegrität prüfen.
- In V.3.38 eine Codegenerierung durchführen, um festzustellen, ob Fehler auftreten und/oder sich CRCs ändern.
- Erkannte Fehler beseitigen und erneut Code generieren, um CRC-Änderungen festzustellen.
- Liegen keine CRC-Änderungen vor, ist der Übergang erfolgreich abgeschlossen.
- Liegen CRC-Änderungen vor, prüfen, ob diese akzeptabel sind.
- Sind die Änderungen akzeptabel, ist der Übergang erfolgreich abgeschlossen.
- Falls diese inakzeptabel sind, mit der entsprechenden Vorversion weiterarbeiten.

### Hinweise zur Konvertierung:

- Die Konvertierung erfolgt für die Versionen V.2.46 und V.2.52 in einer Weise, dass die generierten Konfigurationen gleich bleiben. Ausnahmen sind in Kapitel 4.1 beschrieben. Die Änderungen von der V. 2.36 zur V.2.46 sind in den Releasenotes der V.2.46 beschrieben.
- Die Konvertierung kann je nach Projektgröße bis zu mehreren Stunden dauern.
- Enthält eine der Ressourcen der zu konvertierenden Projekts eine Benutzerverwaltung, dann fügt die Konvertierung eine PADT-Benutzerverwaltung in das Projekt ein und legt für jeden Benutzer einer Ressource einen PADT-Benutzer und eine PADT-Benutzergruppe mit allen Rechten an.  
Ist dies nicht gewünscht, dann sind vor der Konvertierung alle PES-Benutzerverwaltungen des Projekts zu löschen. Dies führt jedoch dazu, dass nach dem nächsten Generieren von Code kein Reload möglich ist.