



**Systembeschreibung**  
**netPROXY**

**V2.0**

**Hilscher Gesellschaft für Systemautomation mbH**  
**[www.hilscher.com](http://www.hilscher.com)**

DOC160203SD02DE | Revision 2 | Deutsch | 2018-02 | Freigegeben | Öffentlich

# Inhaltsverzeichnis

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Einführung</b>   | <b>3</b>  |
| 1.1      | Was ist netPROXY?   | 3         |
| 1.2      | Änderungsübersicht  | 3         |
| 1.3      | Begriffe, Abkürzungen und Definitionen                                  | 3         |
| 1.4      | Dokumentationsübersicht   | 4         |
| 1.5      | Referenzen  | 5         |
| <b>2</b> | <b>Vorteile des netPROXY-Konzepts</b>                                   | <b>6</b>  |
| 2.1      | Einheitlicher Zugriff auf verschiedene Protokoll-Stacks                 | 6         |
| 2.2      | Vorteile der Benutzung von netPROXY                                     | 7         |
| 2.3      | Auswirkungen auf den Implementationsaufwand                             | 8         |
| 2.4      | netPROXY Einschränkungen  | 10        |
| 2.5      | netX Studio Engineering Tool  | 10        |
| <b>3</b> | <b>netPROXY-Objektmodell</b>  | <b>11</b> |
| 3.1      | Objekte, Instanzen und Elemente   | 11        |
| 3.2      | Globaler Adressbereich für Objekt-IDs                                   | 12        |
| 3.3      | Datentypen für Elemente von netPROXY-Objekten                           | 13        |
| 3.3.1    | Boolean   | 13        |
| 3.3.2    | Integer   | 13        |
| 3.3.3    | Unsigned  | 13        |
| 3.3.4    | Real  | 14        |
| 3.3.5    | String  | 14        |
| 3.4      | Adressierung eines Elements eines Objekts innerhalb eines Geräts        | 16        |
| 3.5      | Zugriff auf ein Objekt von der Applikation                              | 17        |
| <b>4</b> | <b>Gerätestruktur</b>   | <b>18</b> |
| 4.1      | netPROXY Gerät mit getrennten Prozessoren und externer Host-Applikation | 18        |
| <b>5</b> | <b>Anhang</b>   | <b>19</b> |
| 5.1      | Rechtliche Hinweise   | 19        |
|          | <b>Kontakte</b>   | <b>25</b> |

# 1 Einführung

This manual describes the netPROXY application framework for designing communication devices in a protocol-stack-independent way.

## 1.1 Was ist netPROXY?

netPROXY stellt ein objektbasiertes Framework zur Geräteentwicklung mit einer einheitlichen, protokoll-unabhängigen Host-Schnittstelle zur Verfügung. Diese Vorgehensweise erlaubt es, Gerätevarianten mit gleichem Funktionsumfang, aber verschiedenen Kommunikationssystemen extrem effektiv zu entwickeln. Damit wird der Entwicklungsaufwand und die Kosten beträchtlich reduziert.

## 1.2 Änderungsübersicht

| Revision | Datum      | Autor | Änderungen  |
|----------|------------|-------|---|
| 1        | 23.11.2017 | RG    | Alle Abschnitte erstellt.                                       |
| 2        | 2018-02-19 | HH    | Abschnitt <i>netPROXY Einschränkungen</i> [► Seite 10] ergänzt. |

Tabelle 1: Änderungsübersicht

## 1.3 Begriffe, Abkürzungen und Definitionen

| Begriff                 | Beschreibung  |
|-------------------------|---|
| netPROXY                | Ein Application Framework zur effizienten Entwicklung von Netzwerk-Geräten auf der Grundlage von Hilscher's netX Prozessor. Es stellt ein abstraktes Objektmodell für Daten und Dienste zur Verfügung. netPROXY wird auch als allgemeiner Begriff für die netPROXY-Technologie von Hilscher verwendet.  |
| Gruppe                  | Logische Zusammenfassung von verwandten Gerätefunktionen in netPROXY. Die Funktionen können gruppiert werden z.B. anhand ihrer physikalischen Rolle (E/A-Ports, Messkanäle, ...) oder anhand Ihrer funktionalen Rolle (Controller, Monitor, Verarbeitungs-Einheit, Achse eines Antriebs ...)  |
| Objekt                  | Eine definierte Datenstruktur, die aus Datenelementen besteht. Jedes solche Datenelement enthält Daten eines festgelegten Typs.   |
| Instanz (eines Objekts) | Eine Instanz mit der Datenstruktur des zugrundeliegenden Objekts<br>Objekte können eine oder mehrere Instanzen zur Verfügung stellen.<br>Instanzen von Objekten werden dazu verwendet, um Kopien von Bereichen mit derselben Datenstruktur anzulegen.<br>Beispiel: mehrere analoge Messkanäle stellen analoge Werte und Statusinformationen als Datenelemente einer Datenstruktur zur Verfügung. In diesem Fall können die Kanäle eindeutig den Objekt-Instanzen zugeordnet werden. |
| Package                 | Eine Menge von netPROXY Objekten, die eine definierte Funktionalität (Verhalten) zur Verfügung stellen. Ein Package wird innerhalb einer bestimmten Gruppe installiert.   |
| netX Studio             | Eclipse basiertes Tool für netPROXY Projekt-Entwurf, Management usw.  |
| Eclipse                 | Effiziente GUI-basierte Software-Entwicklungsumgebung (IDE), in die netX Studio integriert ist. Siehe <a href="http://www.eclipse.org">http://www.eclipse.org</a> .   |

Tabelle 2: Begriffe und Definitionen

Die folgende Abbildung zeigt die Komponenten und Schnittstellen für die Integration eines netIC IOT Kommunikationsmoduls in ein Host-System. Für diese Geräteentwicklung benötigen Sie das . Die Nummern geben das relevante Dokument an.



| #   | Zielgruppe                                | Dokument   | Inhalt   |
|-----|---|--|--|
| (1) | Produktentwickler,<br>Software-Entwickler | Systembeschreibung: netPROXY,<br>DOC160203SDxxDE, deutsch.   | Beschreibung des objektbasierten<br>Frameworks zur Geräteentwicklung.  |
| (2) | Produktentwickler                         | Bedienerhandbuch: netX Studio Engineering<br>Tool, Geräteentwicklung,<br>DOC160103OIxxDE, deutsch.   | Geräteentwicklung mithilfe von netX<br>Studio Engineering Tool. Beschreibung<br>der OPC UA-, MQTT- und Real-Time-<br>Ethernet-Konfiguration. |
| (3) | Produktentwickler                         | Application note: netX Studio Engineering<br>Tool, Web Builder, DOC160207ANxxDE,<br>deutsch.         | Funktionen und Handhabung des Web<br>Builders zum Gestalten und Verwalten von<br>HTML-Webseiten.   |
| (4) | Hardware-Entwickler,<br>Produktentwickler | Bedienerhandbuch: netIC IOT, Hardware mit<br>netX Studio konfigurieren,<br>DOC160501OIxxDE, deutsch. | Pinbelegung konfigurieren und<br>Schnittstellenparameter einstellen.   |
| (5) | Hardware-Entwickler                       | Design guide: netIC IOT, NIOT-I-IC52-RE,<br>DOC141002DGxxEN, englisch.                               | netIC IOT Hardware in das Host-System<br>integrieren.  |
| (6) | Software-Entwickler                       | Protocol API: netPROXY Host API, Function<br>interface, netPROXY Host API.chm,<br>englisch.          | API für die Host-Anwendung für den<br>Zugriff auf Objekte in einem Gerät.  |
| (7) | Software-Entwickler                       | Technische Referenz: netPROXY<br>Objektreferenz, DOC160204TRxxEN,<br>englisch.                       | Beschreibung der netPROXY-Objekte.   |
| (8) | Software-Entwickler                       | Toolkit manual: cifX/netX Toolkit, DPM,<br>DOC090203TKxxEN, englisch.                                | Beschreibung und Verwendung des cifX<br>C-Toolkits.  |
| (9) | Software-Entwickler                       | Programming reference guide: cifX API,<br>DOC121201PRxxEN, englisch.                                 | Beschreibung und Verwendung der<br>Standard cifX API.  |

Tabelle 3: Dokumentationsübersicht

## 1.5 Referenzen

Dieses Dokument bezieht sich auf die folgenden Dokumente:

1. Hilscher Gesellschaft für Systemautomation mbH: Protocol API: netPROXY Host API, Function interface, netPROXY Host API.chm, englisch, 2018.
2. Hilscher Gesellschaft für Systemautomation mbH: Technical reference, netPROXY object reference, DOC160204TRxxEN, englisch, 2018.

## 2 Vorteile des netPROXY-Konzepts

### 2.1 Einheitlicher Zugriff auf verschiedene Protokoll-Stacks

netPROXY ermöglicht einen einheitlichen Zugriff auf Geräte, die unterschiedliche Protokoll-Stacks für unterschiedliche Feldbus- oder Real-Time-Ethernet-Systeme verwenden. netPROXY hilft, Entwicklungskosten in Projekten mit parallelem Einsatz mehrerer Protokoll-Stacks spürbar zu senken.

netPROXY kann als Zwischenschicht zwischen der Benutzer-Applikation und dem Protokoll-Stack angesehen werden. Die folgende Abbildung zeigt, wie netPROXY eine Abstraktionsebene oberhalb des Protokoll-Stacks mittels transparenten Objektmodell verwendet.



Abbildung 2: Vergleich des einheitlichen netPROXY Ansatzes mit dem Standard-Ansatz in Projekten mit mehreren Protokoll-Stacks

## 2.2 Vorteile der Benutzung von netPROXY

Jedes Netzwerk-System definiert üblicherweise einige verschiedene Dienste, die von der Benutzer-Applikation in besonderer Weise behandelt werden sollen. Dies erfordert fast immer ein weitreichendes Verständnis der Funktion und der Ablaufkontrolle in dem jeweiligen Netzwerk-System. Deswegen muß jedes mal, wenn der Schritt zu einer neuen Netzwerktechnologie gemacht wird, erheblicher Aufwand investiert werden.

Die netPROXY-Technologie zielt auf die Erreichung von Synergien bei der Entwicklung der Benutzer-Applikation auf der Grundlage verschiedener Protokoll-Stacks ab. Auf diese Weise können erhebliche Vorteile der netX Automation-Plattform nicht nur auf der Hardware-Seite durch die einheitliche Verwendung des netX-Prozessors, sondern auch durch die Wiederverwendung der Applikations-Software beim Einsatz verschiedener Kommunikationssysteme erzielt werden.

Die Abstraktionsschicht oberhalb des Protokoll-Stacks verbirgt die Komplexität der verschiedenen Kommunikationssysteme und stellt grundlegende Dienste zur Verfügung, wie zyklischen und azyklischen Datenaustausch, Konfiguration usw. Der Anwender kann sich mehr auf seine eigene Applikation konzentrieren statt sich mit protokoll-spezifischen Details auseinandersetzen zu müssen. Das intelligente Tool vereinfacht die Anwendungsentwicklung. Es hilft bei der netzwerk-spezifischen Zuordnung und verwendet sogar die Benutzereingaben, um folgendes zu erzeugen

- Quellcodes für die Applikation
- Netzwerk-spezifische Gerätebeschreibungsdateien
- Dokumentation
- Webseiten für den integrierten Web-Server

## 2.3 Auswirkungen auf den Implementationsaufwand

### Konventionelle Vorgehensweise

Die industriellen Kommunikationssysteme und -Protokolle wurden im Hinblick auf spezielle Anwendungsgebiete in der Automatisierung entwickelt und stellen jeweils ein wohldefiniertes Objektmodell und Dienste zur Verfügung, um verschiedene Anwendungsprofile abzudecken.

Conventional Approach

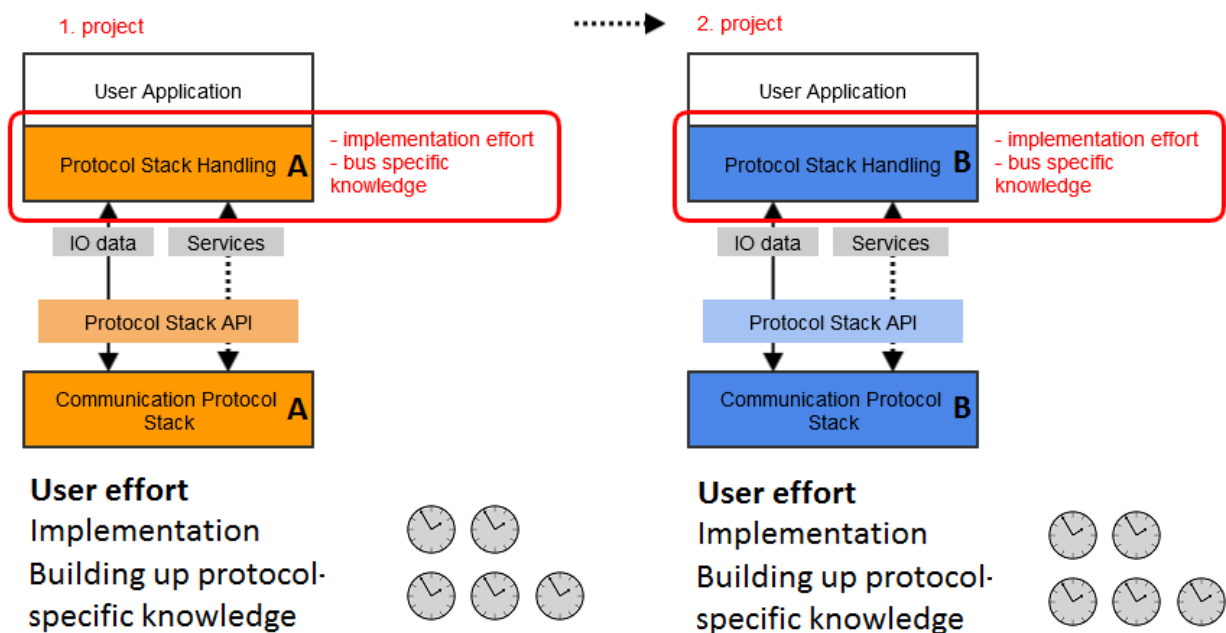


Abbildung 3: Konventionelle Vorgehensweise

Die konventionelle Vorgehensweise unter Anwendung protokoll-stack-spezifischer Dienste führt zu beachtlichem Implementationsaufwand. Dieser Aufwand wird hervorgerufen durch das notwendige Fachwissen über die Funktionsweise der speziellen Bus-/Netzwerk-Systeme und der zugehörigen Dienste in der jeweiligen API Implementation des Protokoll-Stacks.



## Vorgehensweise bei netPROXY

netPROXY wurde von Hilscher entwickelt, um die Realisierung von netzwerk-basierten Geräten auf der Basis des netX Controllers erheblich zu vereinfachen.

Der Vorteil von netPROXY liegt darin, dass netPROXY die Möglichkeit bietet, die Gerätedaten der Applikation einheitlich auf der Basis eines einfachen Objektmodells zu beschreiben. Dies braucht nur einmal in der Entwurfsphase durchgeführt zu werden, um die spezifischen Gerätefunktionen abzudecken, um diese anschließend beim Einsatz weiterer Netzwerk-Protokolle wiederzuverwenden.

netPROXY setzt das Objektmodell nicht nur bei der Verwaltung und dem Zugriff auf die Daten ein, sondern führt auch automatische Umwandlungen zwischen den Netzwerkdiensten der netPROXY Dienste durch. Manche dieser Transformationen können generisch durchgeführt werden. Andere dagegen erfordern konfigurierbare Transformationsregeln, um eine hohe Flexibilität zu erreichen.

Der nächste Vorteil von netPROXY liegt darin, dass Dienste in derselben Weise wie der Zugriff auf ein Datenobjekt zur Verfügung gestellt werden. Dies vermeidet die Einführung neuer Service API-Funktionen bei jeder Erweiterung und führt zu einer sehr schlanken API. Diese ist einfach zu erlernen und zu benutzen.

Wenn netPROXY erst einmal zur Beschreibung der Geräte-Applikationsdaten und zum Zugriff auf das Netzwerk-Protokoll eingeführt worden ist, können alle anderen Netzwerke einfach automatisch unter Anwendung der Default-Mapping-Regeln für das jeweilige System integriert werden.

### netPROXY Approach

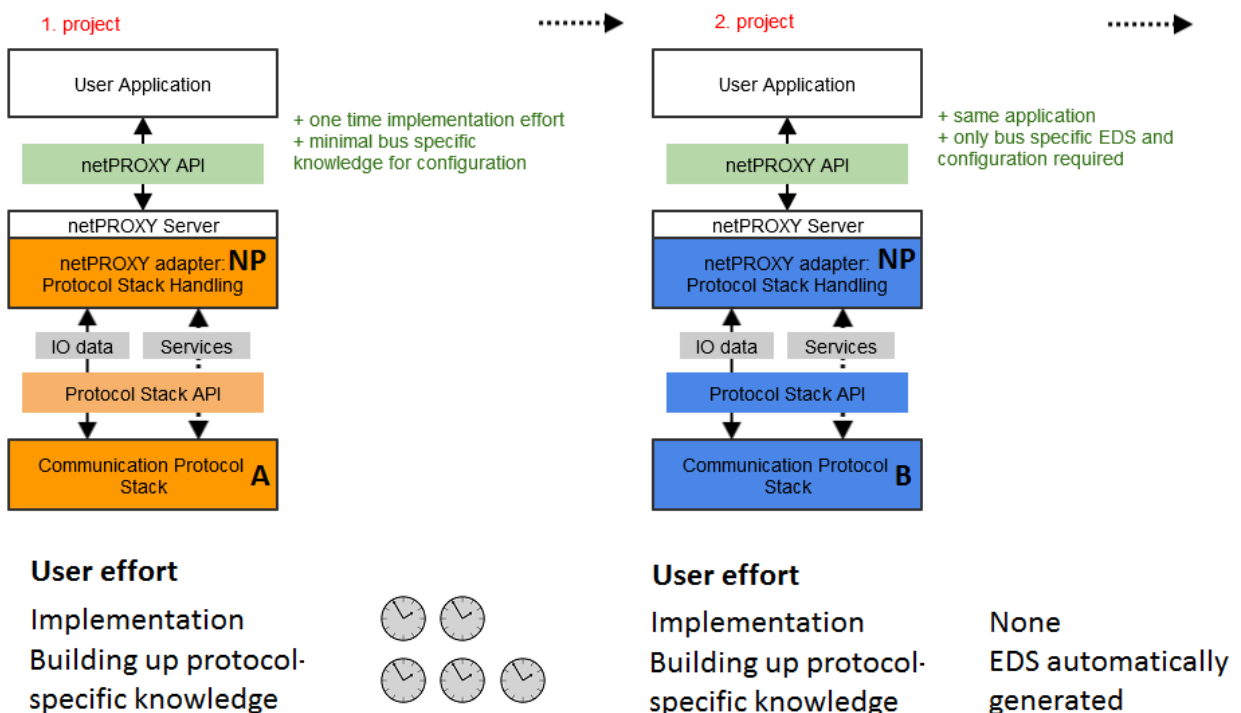


Abbildung 4: Vorgehensweise bei netPROXY

## 2.4 netPROXY Einschränkungen

netPROXY verbirgt die Komplexität der verschiedenen Kommunikationssysteme und stellt grundlegende Dienste zur Verfügung, wie zyklischen und azyklischen Datenaustausch. Als Ergebnis dieser Abstraktion ergeben sich folgende Einschränkungen für die Anwendung:

- eine Profil-Anwendung ist nicht möglich,
- ersetzen vorhandene Produkte ist nicht möglich aufgrund speziell benötigtes Verhalten des Gerätes,
- eine Anwendung mit hohen Anforderungen an die Echtzeit-Kommunikation ist nicht möglich.

Für diese Anforderungen muss die Anwendung die protokoll-spezifischen Dienste verwenden.

Falls spezielle Anforderungen an das Verhalten eines Gerätes benötigt werden, prüfen Sie diese Anforderungen hinsichtlich Realisierbarkeit anhand der Beschreibung der netPROXY Objektreferenz.

## 2.5 netX Studio Engineering Tool

netPROXY wird mit einem intelligenten Engineering Tool ausgeliefert, das die Entwicklung der Applikation vereinfacht und heißt netX Studio Engineering Tool.

Das Engineering Tool netX Studio basiert auf Eclipse und bietet XML-basiertes Datenhandling. Es vereinfacht die Erstellung des Gerätedatenmodells und erleichtert die richtige Konfiguration von netPROXY. Es enthält integrierte Generatoren zur automatischen Erstellung von:

- eine C-Header-Datei mit kundenspezifischen Objekten für die Applikation
- die Gerätebeschreibungsdatei
- Webseiten für den integrierten Webserver
- Dokumentation
- Benutzer-Management.

## 3 netPROXY-Objektmodell

netPROXY modelliert Geräte anhand von Objekten. Das Objekt ist das zentrale Element von netPROXY, jedes Datum oder jeder Dienst, der verwendet wird, um ein Gerät mit Hilfe der netPROXY-Technologie zu entwerfen, wird mit Hilfe eines Objekts abstrahiert. Üblicherweise stellt ein Objekt eine bestimmte Eigenschaft des zu modellierenden Geräts dar.

Dieses Kapitel erklärt die Beziehungen zwischen Objekten und Ihren Instanzen und Elementen.

Es klärt auch, wie mehrere Objekte zu Gruppen und Packages zusammengefasst werden können, um Geräte strukturiert zu modellieren.

### 3.1 Objekte, Instanzen und Elemente

Objekte, Instanzen und ElementeDieser Abschnitt konzentriert sich auf Objekte, ihre Instanzen und Elemente. Innerhalb der netPROXY-Architektur haben Objekte eine zentrale Rolle. Folgende Grundeigenschaften gelten für alle Objekte:

#### Definition und Identifizierung

- Ein Objekt hat immer eine Nummer, über die es identifiziert werden kann, nämlich die Objekt ID. Regeln für die Objekt ID finden Sie im Abschnitt *Globaler Adressbereich für Objekt-IDs* [► Seite 12].
- Ein Objekt hat einen Namen.

#### Beschreibung

- Die Beschreibung eines Objekts enthält Informationen über seine interne Struktur.

#### Beziehung zwischen Objekt, und Instanz

- Ein Objekt hat mindestens eine Instanz (die Instanz 0). Wenn mehr als eine Instanz existiert, werden die weiteren Instanzen in aufsteigender Reihenfolge nummeriert (Instanz 1,2, ...).
- In jeder Instanz können Sie den Elementen eines Objekts einen unterschiedlichen Satz von Werten zuweisen.

#### Beziehung zwischen Objekt und Element

- Jedes Objekt hat mindestens ein Element (Element 0). Wenn mehr als ein Element existiert, werden die weiteren Elemente in aufsteigender Reihenfolge nummeriert (Element 1,2, ...).
- Jedes Objekt gehört zu einem definierten Datentyp. Für die Elemente stehen die folgenden Datentypen zur Verfügung:

| Datentyp           | Verfügbar für             |                |
|--------------------|---------------------------|----------------|
|                    | Kundendefinierte Elemente | Systemelemente |
| Boolean            | Verfügbar                 | Available      |
| Binary             | Nicht verfügbar           | Verfügbar      |
| Integer            | Verfügbar                 | Verfügbar      |
| Unsigned (integer) | Verfügbar                 | Verfügbar      |
| Real               | Verfügbar                 | Verfügbar      |
| String             | Verfügbar                 | Verfügbar      |
| Object reference   | Nicht verfügbar           | Verfügbar      |
| Variable binary    | Nicht verfügbar           | Verfügbar      |

Tabelle 4: Datentypen und Verfügbarkeit

Für weitere Informationen siehe folgenden Abschnitt *Datentypen für Elemente von netPROXY-Objekten* [► Seite 13].

### Programmzugriff

Programm-Code kann auf netPROXY-Objekte über das netPROXY host API zugreifen. Für weitere Informationen zu den Funktionen des netPROXY host API siehe separates Dokument netPROXY Function interface, Protocol API (Document ID: DOC160205APIXXEN) (Referenz [1]).

## 3.2 Globaler Adressbereich für Objekt-IDs

Die Objekt-ID ist eine eindeutige 32-Bit-Zahl, die zu einer speziellen Objektdefinition gehört. Die folgende Tabelle zeigt den globalen Adressbereich für Objekt-IDs.

| Objekt-ID               | Kategorie                                    |
|-------------------------|--|
| 0x00000000              | NULL   |
| 0x00000001 - 0x0FFFFFFF | netPROXY                                     |
| 0x10000000 - 0x1FFFFFFF | System<br>(z. B. Geräte-ID)                  |
| 0x20000000 - 0x2FFFFFFF | Kommunikationsprotokoll<br>(z. B. Parameter) |
| 0x30000000 - 0x3FFFFFFF | Hilscher-Objekte                             |
| 0x40000000 - 0x4FFFFFFF | Kundenspezifische Objekte                    |
| 0x50000000 - 0xFFFFFFFF | Reserviert                                   |

Tabelle 5: Globaler Adressbereich für Objekt-IDs

## 3.3 Datentypen für Elemente von netPROXY-Objekten

Die folgenden Datentypen sind für Elemente von netPROXY-Objekten verfügbar:

### 3.3.1 Boolean

Dieser Datentyp kann nur die Werte "True" und "False" annehmen.

- Der Wert ist "False", wenn alle Bits den Wert "0" haben.
- Wenn mindestens ein Bit gesetzt ist, ist der Wert "True".
- Mögliche Größen sind: 1, 2, 4 or 8 Byte.

### 3.3.2 Integer

Ein Integer-Wert mit Vorzeichen wird als Binärzahl (nach Zweierkomplement-Methode) dargestellt. Die Größe der Zahl, die von diesem Element abgespeichert werden kann, hängt von der Elementgröße ab.

Momentan können nur die folgenden Größen benutzt werden:

| Größe | Minimaler Wert   | Maximaler Wert     | Darstellung des Datentyps C/C++ |
|-------|------------------|--------------------|---------------------------------|
| 1     | -128             | 127                | int8_t                          |
| 2     | -32768           | 32767              | int16_t                         |
| 4     | -2147483648      | 2147483647         | int32_t                         |
| 8     | -2 <sup>63</sup> | 2 <sup>63</sup> -1 | int64_t                         |

Tabelle 6: Integer-Typen

### 3.3.3 Unsigned

Ein vorzeichenloser Integer-Wert ist eine Binärzahl. Die Größe der Zahl, die von diesem Element abgespeichert werden kann, hängt von der Elementgröße ab.

Momentan können nur die folgenden Größen benutzt werden:

| Größe | Minimaler Wert | Maximaler Wert     | Darstellung des Datentyps C/C++ |
|-------|----------------|--------------------|---------------------------------|
| 1     | 0              | 255                | uint8_t                         |
| 2     | 0              | 65.535             | uint16_t                        |
| 4     | 0              | 4.294.967.295      | uint32_t                        |
| 8     | 0              | 2 <sup>64</sup> -1 | uint64_t                        |

Tabelle 7: Vorzeichenlose Integer-Typen

### 3.3.4 Real

Fließkommazahlen werden gemäß IEEE 754 dargestellt.

Momentan können nur die folgenden Größen benutzt werden:

| Größe | Darstellung des Datentyps C/C++ |
|-------|---------------------------------|
| 4     | float                           |
| 8     | double                          |

Tabelle 8: Real-Typen

### 3.3.5 String

Alle Strings sind null-terminiert. Die maximale Stringlänge, die das Element aufnehmen kann, muß einschließlich Begrenzungsbyte im Bereich zwischen 1 und 65535 liegen. Die folgenden Stringformate werden unterstützt:

#### Datenformate

#### Druckbarer ASCII-String

Nur die folgenden Zeichen sind erlaubt:

- 32(20h) – 126(7Eh)

#### ASCII Dezimalzahl

Nur die folgenden Zeichen sind erlaubt:

- 48(30h) – 57(39h)

#### ASCII Time

Diese Angabe benutzt das standardisierte Format ISO 8601:2006-09 / EN 28601 (Erweitertes Format).

YYYY= Jahreszahl im Vierzeichenformat

MM= Monatszahl im Zweizeichenformat (01=Januar, ...)

DD= Tageszahl im Zweizeichenformat (01 bis 31)

hh= Stunde im Zweizeichenformat (00 bis 23)

mm= Minute im Zweizeichenformat (00 bis 59)

ss= Sekunde im Zweizeichenformat (00 bis 59)

TZD= Zeitzonenbezeichner (Z oder +hh:mm oder -hh:mm)

Die Angaben hh, mm und ss können auch optional einen Nachkommaanteil haben.

Beispiel für eine vollständige Datumsangabe mit Stunden, Minuten, Sekunden und dezimalen Sekundenbruchteilen.

- YYYY-MM-DDThh:mm:ss,sTZD (e.g.  
1997-07-16T19:20:30,45+01:00)

**ASCII Version**

Prefix= Versionsnummer ('V' = Freigegeben, 'T' = Test, oder jeder Großbuchstabe)

Major= Hauptversionsnummer 1-5 Stellen (0 bis  $2^{16}-1$ )

Minor= Nebenversionsnummer 1-5 Stellen (0 bis  $2^{16}-1$ )

Revision= Revisionsnummer 1-5 Stellen (0 bis  $2^{16}-1$ )

Build= Buildnummer 1-10 Stellen (0 bis  $2^{32}-1$ )

Alle Teile ohne Prefix, werden an einem Punkt t (2Eh) aufgespalten. der Punkt ist optional und kann weggelassen werden, z.B. bei hardware-relevanter Versionsinformation

Beliebiger Text kann zur Versionsinformation hinzugefügt werden und muss durch ein einzelnes Leerzeichen (20h) abgetrennt werden.

Beispiel für eine vollständige Versionsinformation:

- PrefixMajor.Minor.Revision ( "V1.0.0", "T1.0.0 debug code" )
- PrefixMajor.Minor.Revision.Build ( "V1.2.3.4", "V1.2.3.4 null series" )

### 3.4 Adressierung eines Elements eines Objekts innerhalb eines Geräts

Four numbers address a specific element of an object within a device:

- Gruppen-ID
- Objekt-ID
- Instanznummer
- Elementnummer

#### Wertebereich

| ID oder Nummer | Beschreibung   |
|----------------|--|
| Gruppen-ID     | Die netPROXY-Gruppen-ID adressiert die Gruppe (Objektsammlung) innerhalb eines Gerätes.<br>Ein Gerät enthält oft nur eine Gruppe. In diesem Fall ist die Gruppen-ID immer 0.   |
| Objekt-ID      | Die netPROXY-Object-ID adressiert das Objekt.<br>Das netPROXY-Objekt 0x00001000 enthält eine Liste aller Objekte eines Gerätes inklusive der Gruppen-ID, der das Objekt zugeordnet ist.  |
| Instanznummer  | Die netPROXY-Instanznummer adressiert die Instanz eines Objektes, da ein Objekt mehrfach existieren kann.<br>Wert 0 adressiert die erste Instanz eines Objektes. Wert 1 adressiert die zweite Instanz (falls verfügbar) eines Objektes. Die N Instanzen eines Objektes werden mit 0, 1, ..., N-1 adressiert. |
| Elementnummer  | Die netPROXY-Elementnummer adressiert einen Teil eines Objektes.<br>Wert 0 adressiert das erste Element eines Objektes. Wert 1 adressiert das zweite Element (falls verfügbar) eines Objektes. Die N Elemente eines Objektes werden mit 0, 1, ..., N-1 adressiert.   |

Tabelle 9: Wertebereich zum Adressieren eines Elementes



### 3.5 Zugriff auf ein Objekt von der Applikation

Die Applikation kann auf das Objekt über zwei API-Funktionen, nämlich `read()` und `write()`, zugreifen.

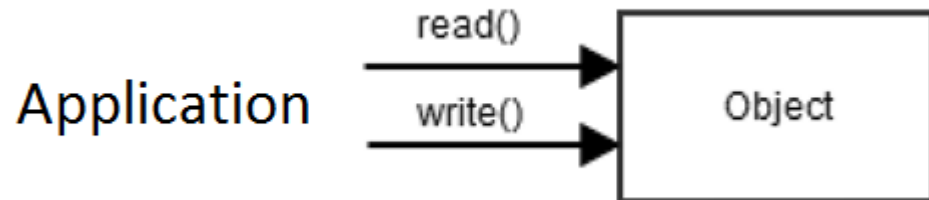


Abbildung 5: Einfaches Package (Ohne Verhalten)

Beim Zugriff auf ein Objekt über die API- Funktionen `read()` und `write()` kann dieses über die vier folgenden Angaben referenziert werden:

| Feldname | Elementarer Datentyp | Beschreibung       |
|----------|----------------------|--------------------|
| GroupID  | UINT32               | netPROXY Gruppe    |
| ObjectID | UINT32               | netPROXY Objekt ID |
| Instance | UINT32               | netPROXY Instanz   |
| Element  | UINT32               | netPROXY Element   |

Tabelle 10: Objektreferenzierung

## 4 Gerätestruktur

Es gibt zwei mögliche Typen von Gerätestrukturen, um ein netPROXY-Gerät aufzubauen.

### 4.1 netPROXY Gerät mit getrennten Prozessoren und externer Host-Applikation

Diese Geräte benutzen Embedded Modules (z.B. NIC52-RE mit netPROXY) mit einer netPROXY-Firmware. Üblicherweise hat ein solches Gerät zwei CPUs, die miteinander über eine Host-Schnittstelle verbunden sind. Alle Kommunikation zwischen der Host-CPU und der netX-CPU des netPROXY-Geräts läuft über die Host-Schnittstelle.

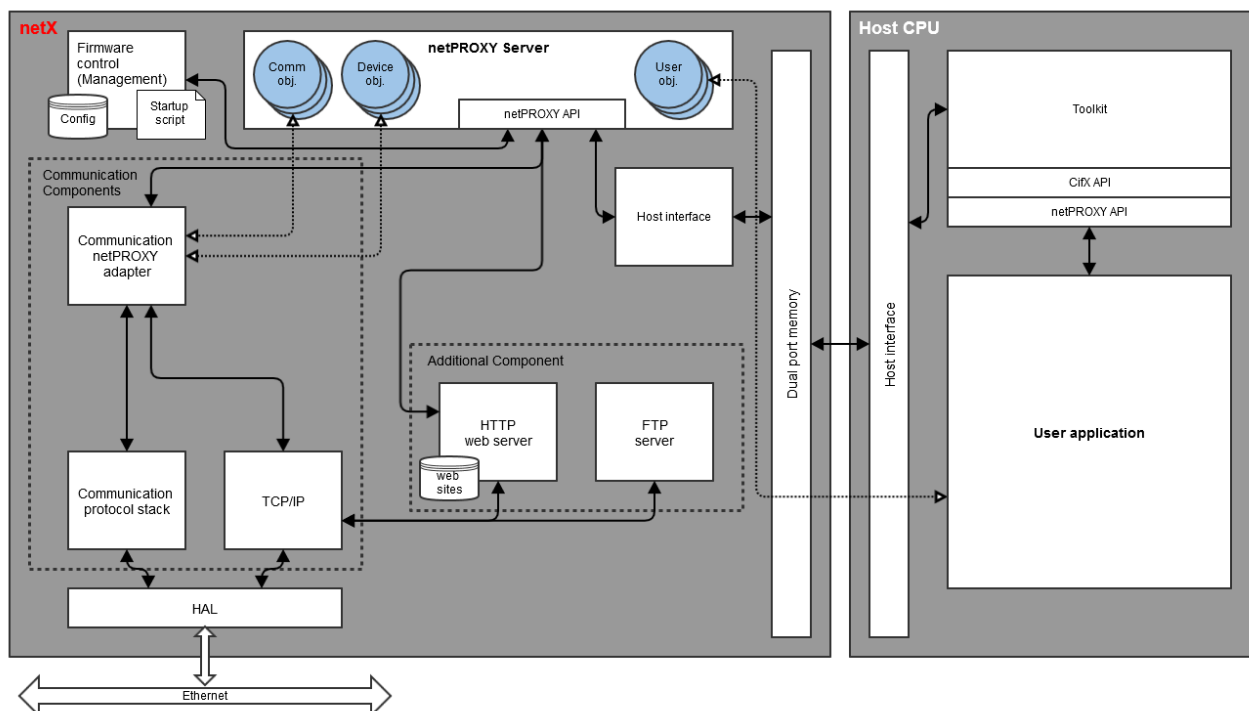


Abbildung 6: netPROXY Gerät mit spezieller Host-Applikation

## 5 Anhang

### 5.1 Rechtliche Hinweise

#### **Copyright**

© Hilscher Gesellschaft für Systemautomation mbH

Alle Rechte vorbehalten.

Die Bilder, Fotografien und Texte der Begleitmaterialien (in Form eines Benutzerhandbuchs, Bedienerhandbuchs, Statement of Work Dokument sowie alle weiteren Dokumenttypen, Begleittexte, Dokumentation etc.) sind durch deutsches und internationales Urheberrecht sowie internationale Handels- und Schutzbestimmungen geschützt. Sie sind ohne vorherige schriftliche Genehmigung nicht berechtigt, diese vollständig oder teilweise durch technische oder mechanische Verfahren zu vervielfältigen (Druck, Fotokopie oder anderes Verfahren), unter Verwendung elektronischer Systeme zu verarbeiten oder zu übertragen. Es ist Ihnen untersagt, Veränderungen an Copyrightvermerken, Kennzeichen, Markenzeichen oder Eigentumsangaben vorzunehmen. Darstellungen werden ohne Rücksicht auf die Patentlage mitgeteilt. Die in diesem Dokument enthaltenen Firmennamen und Produktbezeichnungen sind möglicherweise Marken bzw. Warenzeichen der jeweiligen Inhaber und können warenzeichen-, marken- oder patentrechtlich geschützt sein. Jede Form der weiteren Nutzung bedarf der ausdrücklichen Genehmigung durch den jeweiligen Inhaber der Rechte.

#### **Wichtige Hinweise**

Vorliegende Dokumentation in Form eines Benutzerhandbuchs, Bedienerhandbuchs sowie alle weiteren Dokumenttypen und Begleittexte wurden/werden mit größter Sorgfalt erarbeitet. Fehler können jedoch nicht ausgeschlossen werden. Eine Garantie, die juristische Verantwortung für fehlerhafte Angaben oder irgendeine Haftung kann daher nicht übernommen werden. Sie werden darauf hingewiesen, dass Beschreibungen in dem Benutzerhandbuch, den Begleittexte und der Dokumentation weder eine Garantie, noch eine Angabe über die nach dem Vertrag vorausgesetzte Verwendung oder eine zugesicherte Eigenschaft darstellen. Es kann nicht ausgeschlossen werden, dass das Benutzerhandbuch, die Begleittexte und die Dokumentation nicht vollständig mit den beschriebenen Eigenschaften, Normen oder sonstigen Daten der gelieferten Produkte übereinstimmen. Eine Gewähr oder Garantie bezüglich der Richtigkeit oder Genauigkeit der Informationen wird nicht übernommen.

Wir behalten uns das Recht vor, unsere Produkte und deren Spezifikation, sowie zugehörige Dokumentation in Form eines Benutzerhandbuchs, Bedienerhandbuchs sowie alle weiteren Dokumenttypen und Begleittexte jederzeit und ohne Vorankündigung zu ändern, ohne zur Anzeige der Änderung verpflichtet zu sein. Änderungen werden in zukünftigen Manuals berücksichtigt und stellen keine Verpflichtung dar; insbesondere besteht kein Anspruch auf Überarbeitung gelieferter Dokumente. Es gilt jeweils das Manual, das mit dem Produkt ausgeliefert wird.

Die Hilscher Gesellschaft für Systemautomation mbH haftet unter keinen Umständen für direkte, indirekte, Neben- oder Folgeschäden oder Einkommensverluste, die aus der Verwendung der hier enthaltenen Informationen entstehen.

### **Haftungsausschluss**

Die Hard- und/oder Software wurde von der Hilscher Gesellschaft für Systemautomation mbH sorgfältig erstellt und getestet und wird im reinen Ist-Zustand zur Verfügung gestellt. Es kann keine Gewährleistung für die Leistungsfähigkeit und Fehlerfreiheit der Hard- und/oder Software für alle Anwendungsbedingungen und -fälle und die erzielten Arbeitsergebnisse bei Verwendung der Hard- und/oder Software durch den Benutzer übernommen werden. Die Haftung für etwaige Schäden, die durch die Verwendung der Hard- und Software oder der zugehörigen Dokumente entstanden sein könnten, beschränkt sich auf den Fall des Vorsatzes oder der grob fahrlässigen Verletzung wesentlicher Vertragspflichten. Der Schadensersatzanspruch für die Verletzung wesentlicher Vertragspflichten ist jedoch auf den vertragstypischen vorhersehbaren Schaden begrenzt.

Insbesondere wird hiermit ausdrücklich vereinbart, dass jegliche Nutzung bzw. Verwendung von der Hard- und/oder Software im Zusammenhang

- der Luft- und Raumfahrt betreffend der Flugsteuerung,
- Kernschmelzungsprozessen in Kernkraftwerken,
- medizinischen Geräten die zur Lebenserhaltung eingesetzt werden
- und der Personenbeförderung betreffend der Fahrzeugsteuerung

ausgeschlossen ist. Es ist strikt untersagt, die Hard- und/oder Software in folgenden Bereichen zu verwenden:

- für militärische Zwecke oder in Waffensystemen;
- zum Entwurf, zur Konstruktion, Wartung oder zum Betrieb von Nuklearanlagen;
- in Flugsicherungssystemen, Flugverkehrs- oder Flugkommunikationssystemen;
- in Lebenserhaltungssystemen;
- in Systemen, in denen Fehlfunktionen der Hard- und/oder Software körperliche Schäden oder Verletzungen mit Todesfolge nach sich ziehen können.

Sie werden darauf hingewiesen, dass die Hard- und/oder Software nicht für die Verwendung in Gefahrumgebungen erstellt worden ist, die ausfallsichere Kontrollmechanismen erfordern. Die Benutzung der Hard- und/oder Software in einer solchen Umgebung geschieht auf eigene Gefahr; jede Haftung für Schäden oder Verluste aufgrund unerlaubter Benutzung ist ausgeschlossen.

### **Gewährleistung**

Die Hilscher Gesellschaft für Systemautomation mbH übernimmt die Gewährleistung für das funktionsfehlerfreie Laufen der Software entsprechend der im Pflichtenheft aufgeführten Anforderungen und dafür, dass sie bei Abnahme keine Mängel aufweist. Die Gewährleistungszeit beträgt 12 Monate beginnend mit der Abnahme bzw. Kauf (durch ausdrückliches Erklärung oder konkludent, durch schlüssiges Verhalten des Kunden, z.B. bei dauerhafter Inbetriebnahme).

Die Gewährleistungspflicht für Geräte (Hardware) unserer Fertigung beträgt 36 Monate, gerechnet vom Tage der Lieferung ab Werk. Vorstehende Bestimmungen gelten nicht, soweit das Gesetz gemäß § 438 Abs. 1 Nr. 2 BGB, § 479 Abs.1 BGB und § 634a Abs. 1 BGB zwingend längere Fristen vorschreibt. Sollte trotz aller aufgewendeter Sorgfalt die gelieferte Ware einen Mangel aufweisen, der bereits zum Zeitpunkt des Gefahrübergangs vorlag, werden wir die Ware vorbehaltlich fristgerechter Mängelrüge, nach unserer Wahl nachbessern oder Ersatzware liefern.

Die Gewährleistungspflicht entfällt, wenn die Mängelrügen nicht unverzüglich geltend gemacht werden, wenn der Käufer oder Dritte Eingriffe an den Erzeugnissen vorgenommen haben, wenn der Mangel durch natürlichen Verschleiß, infolge ungünstiger Betriebsumstände oder infolge von Verstößen gegen unsere Betriebsvorschriften oder gegen die Regeln der Elektrotechnik eingetreten ist oder wenn unserer Aufforderung auf Rücksendung des schadhafte Gegenstandes nicht umgehend nachgekommen wird.

### **Kosten für Support, Wartung, Anpassung und Produktpflege**

Wir weisen Sie darauf hin, dass nur bei dem Vorliegen eines Sachmangels kostenlose Nachbesserung erfolgt. Jede Form von technischem Support, Wartung und individuelle Anpassung ist keine Gewährleistung, sondern extra zu vergüten.

### **Weitere Garantien**

Obwohl die Hard- und Software mit aller Sorgfalt entwickelt und intensiv getestet wurde, übernimmt die Hilscher Gesellschaft für Systemautomation mbH keine Garantie für die Eignung für irgendeinen Zweck, der nicht schriftlich bestätigt wurde. Es kann nicht garantiert werden, dass die Hard- und Software Ihren Anforderungen entspricht, die Verwendung der Hard- und/oder Software unterbrechungsfrei und die Hard- und/oder Software fehlerfrei ist.

Eine Garantie auf Nichtübertretung, Nichtverletzung von Patenten, Eigentumsrecht oder Freiheit von Einwirkungen Dritter wird nicht gewährt. Weitere Garantien oder Zusicherungen hinsichtlich Marktgängigkeit, Rechtsmangelfreiheit, Integrierung oder Brauchbarkeit für bestimmte Zwecke werden nicht gewährt, es sei denn, diese sind nach geltendem Recht vorgeschrieben und können nicht eingeschränkt werden.

## **Vertraulichkeit**

Der Kunde erkennt ausdrücklich an, dass dieses Dokument Geschäftsgeheimnisse, durch Copyright und andere Patent- und Eigentumsrechte geschützte Informationen sowie sich darauf beziehende Rechte der Hilscher Gesellschaft für Systemautomation mbH beinhaltet. Er willigt ein, alle diese ihm von der Hilscher Gesellschaft für Systemautomation mbH zur Verfügung gestellten Informationen und Rechte, welche von der Hilscher Gesellschaft für Systemautomation mbH offen gelegt und zugänglich gemacht wurden und die Bedingungen dieser Vereinbarung vertraulich zu behandeln.

Die Parteien erklären sich dahin gehend einverstanden, dass die Informationen, die sie von der jeweils anderen Partei erhalten haben, in dem geistigen Eigentum dieser Partei stehen und verbleiben, soweit dies nicht vertraglich anderweitig geregelt ist.

Der Kunde darf dieses Know-how keinem Dritten zur Kenntnis gelangen lassen und sie den berechtigten Anwendern ausschließlich innerhalb des Rahmens und in dem Umfang zur Verfügung stellen, wie dies für deren Wissen erforderlich ist. Mit dem Kunden verbundene Unternehmen gelten nicht als Dritte. Der Kunde muss berechnigte Anwender zur Vertraulichkeit verpflichten. Der Kunde soll die vertraulichen Informationen ausschließlich in Zusammenhang mit den in dieser Vereinbarung spezifizierten Leistungen verwenden.

Der Kunde darf diese vertraulichen Informationen nicht zu seinem eigenen Vorteil oder eigenen Zwecken, bzw. zum Vorteil oder Zwecken eines Dritten verwenden oder geschäftlich nutzen und darf diese vertraulichen Informationen nur insoweit verwenden, wie in dieser Vereinbarung vorgesehen bzw. anderweitig insoweit, wie er hierzu ausdrücklich von der offen legenden Partei schriftlich bevollmächtigt wurde. Der Kunde ist berechnigt, seinen unmittelbaren Rechts- und Finanzberatern die Vertragsbedingungen dieser Vereinbarung unter Vertraulichkeitsverpflichtung zu offenbaren, wie dies für den normalen Geschäftsbetrieb des Kunden erforderlich ist.

## **Exportbestimmungen**

Das gelieferte Produkt (einschließlich der technischen Daten) unterliegt gesetzlichen Export- bzw. Importgesetzen sowie damit verbundenen Vorschriften verschiedener Länder, insbesondere denen von Deutschland und den USA. Das Produkt/Hardware/Software darf nicht in Länder exportiert werden, in denen dies durch das US-amerikanische Exportkontrollgesetz und dessen ergänzender Bestimmungen verboten ist. Sie verpflichten sich, die Vorschriften strikt zu befolgen und in eigener Verantwortung einzuhalten. Sie werden darauf hingewiesen, dass Sie zum Export, zur Wiederausfuhr oder zum Import des Produktes unter Umständen staatlicher Genehmigungen bedürfen.

## **Geschäftsbedingungen**

Beachten Sie auch die Hinweise zu weiteren relevanten rechtlichen Themen auf unserer netIOT-Website unter <http://www.netiot.com/netiot/netiot-edge/terms-and-conditions/>.

## Abbildungsverzeichnis

|              |   |    |
|--------------|---|----|
| Abbildung 1: | netIC IOT Dokumentationsübersicht .....   | 4  |
| Abbildung 2: | Vergleich des einheitlichen netPROXY Ansatzes mit dem Standard-Ansatz in<br>Projekten mit mehreren Protokoll-Stacks ..... | 6  |
| Abbildung 3: | Konventionelle Vorgehensweise .....   | 8  |
| Abbildung 4: | Vorgehensweise bei netPROXY.....  | 9  |
| Abbildung 5: | Einfaches Package (Ohne Verhalten) .....  | 17 |
| Abbildung 6: | netPROXY Gerät mit spezieller Host-Applikation.....   | 18 |

## Tabellenverzeichnis

|             |  |    |
|-------------|--|----|
| Tabelle 1:  | Änderungsübersicht .....                           | 3  |
| Tabelle 2:  | Begriffe und Definitionen .....                    | 3  |
| Tabelle 3:  | Dokumentationsübersicht .....                      | 4  |
| Tabelle 4:  | Datentypen und Verfügbarkeit .....                 | 12 |
| Tabelle 5:  | Globaler Adressbereich für Objekt-IDs .....        | 12 |
| Tabelle 6:  | Integer-Typen .....                                | 13 |
| Tabelle 7:  | Vorzeichenlose Integer-Typen .....                 | 13 |
| Tabelle 8:  | Real-Typen .....                                   | 14 |
| Tabelle 9:  | Wertebereich zum Adressieren eines Elementes ..... | 16 |
| Tabelle 10: | Objektreferenzierung .....                         | 17 |



# Kontakte

## HAUPTSITZ

### Deutschland

Hilscher Gesellschaft für  
Systemautomation mbH  
Rheinstrasse 15  
65795 Hattersheim  
Telefon: +49 (0) 6190 9907-0  
Fax: +49 (0) 6190 9907-50  
E-Mail: [info@hilscher.com](mailto:info@hilscher.com)

### Support

Telefon: +49 (0) 6190 9907-99  
E-Mail: [de.support@hilscher.com](mailto:de.support@hilscher.com)

## NIEDERLASSUNGEN

### China

Hilscher Systemautomation (Shanghai) Co. Ltd.  
200010 Shanghai  
Telefon: +86 (0) 21-6355-5161  
E-Mail: [info@hilscher.cn](mailto:info@hilscher.cn)

### Support

Telefon: +86 (0) 21-6355-5161  
E-Mail: [cn.support@hilscher.com](mailto:cn.support@hilscher.com)

### Frankreich

Hilscher France S.a.r.l.  
69500 Bron  
Telefon: +33 (0) 4 72 37 98 40  
E-Mail: [info@hilscher.fr](mailto:info@hilscher.fr)

### Support

Telefon: +33 (0) 4 72 37 98 40  
E-Mail: [fr.support@hilscher.com](mailto:fr.support@hilscher.com)

### Indien

Hilscher India Pvt. Ltd.  
Pune, Delhi, Mumbai  
Telefon: +91 8888 750 777  
E-Mail: [info@hilscher.in](mailto:info@hilscher.in)

### Italien

Hilscher Italia S.r.l.  
20090 Vimodrone (MI)  
Telefon: +39 02 25007068  
E-Mail: [info@hilscher.it](mailto:info@hilscher.it)

### Support

Telefon: +39 02 25007068  
E-Mail: [it.support@hilscher.com](mailto:it.support@hilscher.com)

### Japan

Hilscher Japan KK  
Tokyo, 160-0022  
Telefon: +81 (0) 3-5362-0521  
E-Mail: [info@hilscher.jp](mailto:info@hilscher.jp)

### Support

Telefon: +81 (0) 3-5362-0521  
E-Mail: [jp.support@hilscher.com](mailto:jp.support@hilscher.com)

### Korea

Hilscher Korea Inc.  
Seongnam, Gyeonggi, 463-400  
Telefon: +82 (0) 31-789-3715  
E-Mail: [info@hilscher.kr](mailto:info@hilscher.kr)

### Schweiz

Hilscher Swiss GmbH  
4500 Solothurn  
Telefon: +41 (0) 32 623 6633  
E-Mail: [info@hilscher.ch](mailto:info@hilscher.ch)

### Support

Telefon: +49 (0) 6190 9907-99  
E-Mail: [ch.support@hilscher.com](mailto:ch.support@hilscher.com)

### USA

Hilscher North America, Inc.  
Lisle, IL 60532  
Telefon: +1 630-505-5301  
E-Mail: [info@hilscher.us](mailto:info@hilscher.us)

### Support

Telefon: +1 630-505-5301  
E-Mail: [us.support@hilscher.com](mailto:us.support@hilscher.com)