

mapp UserX: Benutzerverwaltung mithilfe des Benutzer-Rollen-Systems



mapp UserX dient zum Einrichten einer Benutzerverwaltung. Rollen und Benutzer werden über das Automation Studio Benutzer- Rollen-System, damit auch OPC UA konform, erstellt und dann mithilfe von mapp UserX verwaltet. Dazu gehören Zugriffsrechte, Benutzerdaten, Passwortdefinition (Buchstaben, Zahlen, groß, klein, ...), Login und Logout, sowie eine Verbindung zur Visualisierung.

mapp UserX liefert viele Funktionen, welche das Umsetzen der Bestimmung 21 CFR Teil 11 erleichtern (z.B. werden Passwörter automatisch nach einem definierten Zeitraum ungültig).

mapp UserX kann als lokales oder zentrales User management verwendet werden.



Auf unserem [B&R Tutorial Portal](#) sind Tutorials zum Thema mapp UserX ([GER/ENG](#)) zu finden. (Zugangsdaten)



Auf unserem [Youtube Kanal](#) sind viele Tutorials zum Thema zu finden.

1 Konzept



Allgemeines

- Als Applikationist möchte ich mich nicht um Basisfunktionen wie das Ein- und Ausloggen, sowie die Darstellung über eine Visualisierung kümmern müssen
- Als Applikationist möchte ich meine Benutzer, sowie Benutzergruppen einfach konfigurieren können
- Als OEM möchte ich meinen Kunden beim Erfüllen der FDA Regulierungen unterstützen und mapp UserX bietet viele, dafür notwendige Funktionen out-of-the-box

Export-, und Import-Funktion

- Als Endnutzer möchte ich meine aktuellen Benutzer und Rollen an der bestehenden Maschine exportieren, um diese in anderen Maschinen wieder verwenden zu können.

User management

- Als Applikationist möchte ich Einstellungen für mein User management vornehmen können (z.B. die Definition wie ein Passwort auszusehen hat oder nach welchem Zeitraum ein Benutzer automatisch ausgeloggt wird)
- Als Endnutzer möchte ich Zugriff durch Dritte auf meine Maschine verhindern. Nur Berechtigte besitzen die Möglichkeit meine Maschine zu bedienen
- Als Endnutzer möchte ich Mitarbeiter, welche aus meiner Firma ausgeschieden sind, sperren bzw. löschen können

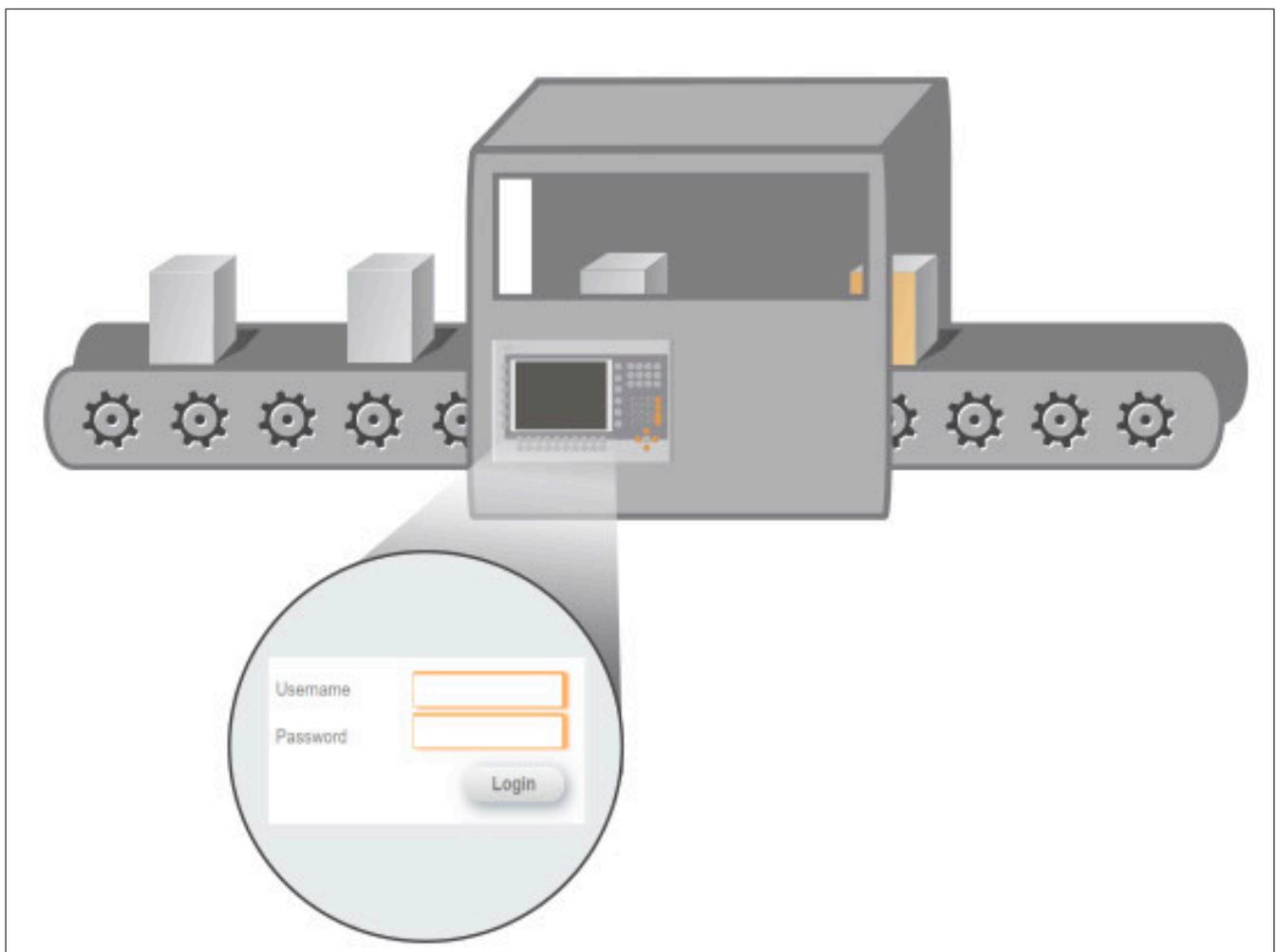
- Als OEM möchte ich einen eigenen Zugang zu meinen Maschinen, um bestimmte Maschinendaten abzufragen bzw. zu verändern
- Als Maschinenbediener möchte ich verhindern, dass jemand ohne mein Wissen Maschinenparameter verändert
- Als Schichtleiter möchte ich meine Rezepte erstellen und verändern können. Ich möchte aber nicht, dass meine Maschinenbediener bzw. Servicetechniker Rezepte verändern können
- Als Endkunde möchte ich, dass sich alle Mitarbeiter über unseren Active Directory Server anmelden, damit alle Benutzerdaten zentral verwaltet werden.

1.1 Glossar

In diesem Abschnitt werden alle wichtigen Begriffe und Definitionen innerhalb mapp UserX aufgelistet und erklärt.

Benutzerverwaltung

Mithilfe einer Benutzerverwaltung kann sich ein Benutzer eindeutig im System identifizieren. Die Identifikation erfolgt durch Eingabe eines Benutzernamens sowie Passworts. Durch mapp UserX kann eine Benutzerverwaltung schnell und einfach realisiert werden. Für mehr Information dazu siehe Abschnitt [Arten der Benutzerverwaltung](#).



User management

Mithilfe des [User management](#) werden Benutzer und Rollen verwaltet. Allgemeine Einstellungen des User managements werden über die [MpUserX Konfiguration](#) festgelegt. Dabei kann beispielsweise festgelegt werden welche Definitionen für das Passwort gelten oder wie viele Login-Versuche ein Benutzer hat.

Name	Value
<ul style="list-style-type: none"> [-] UserMgmtX <ul style="list-style-type: none"> [-] User management system settings <ul style="list-style-type: none"> Type Local [-] General Settings <ul style="list-style-type: none"> Password change interval 0 User expiration time 0 Admin unlock time 3600 Password change required FALSE Password case FALSE Password alphanumeric FALSE Password length 5 Password special characters FALSE Username length 1 Password History Size 0 No delete FALSE Login attempts 3 Signature attempts 3 Edit users with same user-level FALSE User Import Mode Overwrite Existing Items Role Import Mode Overwrite Existing Items Unchecked Import FALSE 	

Das User management bietet zur Laufzeit auch verschiedene Funktionen an, wie das Sperren von Benutzern oder das Exportieren oder Importieren von Benutzern und Rollen. Welche Funktionen verwendet werden können, sind im Abschnitt [User management Funktionen](#) erklärt.

Session

Meldet sich ein Benutzer über mapp UserX im System an, so wird eine Login-Session geöffnet. Die Session kann dabei über die Applikation ("Application Session") oder über eine mapp View Visualisierung ("Visualization Session") geöffnet werden.

```

PROGRAM_CYCLIC

MpUserXLogin_0.Enable := TRUE;
MpUserXLogin_0.MpLink := ADR(gUserXLogin);
MpUserXLogin_0.Username := ADR(Username);
MpUserXLogin_0.Password := ADR>Password);
MpUserXLogin_0();

END_PROGRAM
        
```

Application Session

Eine [Application Session](#) beschreibt den Login-Vorgang auf applikativem Wege. Dazu werden verschiedene Funktionsbausteine bzw. Funktionen verwendet, wie [MpUserXLogin](#).

Visualization Session

Eine [Visualization Session](#) beschreibt das Öffnen einer Session über mapp View. Dabei können verschiedene Login-Widgets verwendet werden.

Für einen erfolgreichen Login-Vorgang wird ein Benutzername sowie das dazugehörige Passwort benötigt.

Benutzer, Rollen und Passwörter

Damit sich ein Benutzer im System anmelden kann müssen vorher Benutzer und Rollen im Benutzer-Rollen-System von Automation Studio definiert werden. Dabei wird der Rollename, Benutzername sowie das dazugehörige Passwort festgelegt. In der [MpUserX Konfiguration](#) werden die zuvor definierten Benutzer und Rollen verlinkt.

Mithilfe des Benutzernamens und dem dazugehörigen Passwort kann sich ein Benutzer identifizieren bzw. wird vom Zielsystem authentifiziert.

Identifizierung / Authentifizierung

Für einen erfolgreichen Login-Vorgang wird eine Identifizierung sowie Authentifizierung benötigt.

Die Identifizierung erfolgt mit dem Benutzernamen. Damit am Zielsystem sichergestellt werden kann, dass es sich wirklich um den korrekten Benutzer handelt wird bei einem Login-Vorgang zusätzlich zum Benutzernamen auch ein Passwort verwendet. Über dieses Passwort kann sich der Benutzer am Zielsystem authentisieren. Am Zielsystem wird der angegebene Benutzername und das dazugehörige Passwort überprüft. Bei erfolgreicher Überprüfung hat das Zielsystem den Benutzer authentifiziert und der Benutzer ist im System angemeldet.

Durch diesen Login-Vorgang können für einen Benutzer definierte Rechte freigeschaltet werden.

Rechte

Meldet sich ein Benutzer am Zielsystem an, so können für diesen Benutzer Rechte freigeschaltet werden. Ein Recht kann beispielsweise das Editieren von Maschinenparametern sein oder das freischalten einer Service-Seite.

Es gilt zu beachten, dass die Rechte an der Rolle des Benutzers geknüpft sind und nicht an den Benutzer selbst.

Abhängig davon, ob eine Application-Session oder Visualization-Session geöffnet wurde, können die Rechte unterschiedlich festgelegt werden. Für mehr Information siehe Abschnitt [Verwalten von Berechtigungen innerhalb einer Application Session](#) bzw. Abschnitt [Verwalten von Berechtigungen innerhalb einer Visualization Session](#).

Zusätzlich zu definierten Rechten kann der angemeldete Benutzer unterschiedliche Funktionen nutzen.

Funktionen

Ist ein Benutzer im System angemeldet, so kann dieser unterschiedliche Funktionen nutzen, wie beispielsweise das Ändern des eigenen Passworts. Dabei wird zwischen [Funktionen der Application Session](#) sowie [Funktionen der Visualization Session](#) unterschieden.

1.2 Arten der Benutzerverwaltung

Mithilfe einer Benutzerverwaltung kann sich ein Benutzer eindeutig im System authentifizieren. Dies erfolgt durch Eingabe eines Benutzernamens sowie Passworts. Dadurch wird der Schutz vor Fremdzugriffen und somit die Vertraulichkeit der Inhalte im jeweiligen System gewährleistet. Zusätzlich können, je nach Benutzer, unterschiedliche Funktionen im System freigeschaltet oder gesperrt werden.

In Automation Studio wird die Benutzerverwaltung mithilfe des Benutzer-Rollen-System realisiert.

Dabei gibt es zwei verschiedene Möglichkeiten das Benutzer-Rollen-System zu verwenden:

Benutzer-Rollen-System in Kombination mit Automation Runtime (AR)

Für diese Kombination wird die ArUser Bibliothek verwendet. Über diese Bibliothek können Benutzer beispielsweise hinzugefügt, gelöscht oder auch importiert werden. Die Benutzerverwaltung muss dabei applikativ selbst erstellt werden. Das bedeutet, Benutzer-Funktionen, wie das Ein-, und Ausloggen, müssen selbst definiert werden. Die Benutzerverwaltung kann auch in dieser Variante applikativ mit einer Visualisierung verbunden werden.

Benutzer-Rollen-System in Kombination mit mapp UserX

mapp UserX basiert auf dem Benutzer-Rollen-System von Automation Studio. Meldet sich ein Benutzer über mapp UserX an, so wird eine Login-Session geöffnet. Diese kann applikativ ("[Application Session](#)") oder über die Visualisierung ("[Visualization Session](#)") gestartet werden. Über diese [Session](#) können Benutzer-Funktionen verwendet werden, wie das Ein-, und Ausloggen oder das Ändern des Passworts.

Mithilfe des [User management](#) werden Benutzer und Rollen verwaltet. Beispielsweise können Benutzer über das User management gesperrt werden, oder ihre Email-Adresse vermerkt werden. Außerdem werden über das User Management Regeln festgelegt, die für alle Sessions gelten, bspw. die Definition wie ein Passwort aussehen muss.

1.3 User management

Das User management wird in der Regel von einem Administrator des Systems verwaltet. Der Administrator vergibt Benutzerkennungen, erteilt Zugriffsberechtigungen auf Systeme, Services oder Anwendungen und entzieht diese bei Bedarf wieder. Eine Auflistung der verfügbaren Funktionen ist im Abschnitt [User management Funktionen](#) zu finden.

Modus des User managements

Die Benutzer können über das User management lokal, zentral oder lokal und zentral verwaltet werden. Der Modus wird dabei über die [MpUserX Konfiguration](#) bestimmt. Für mehr Information siehe Abschnitt [Modus des User managements](#).

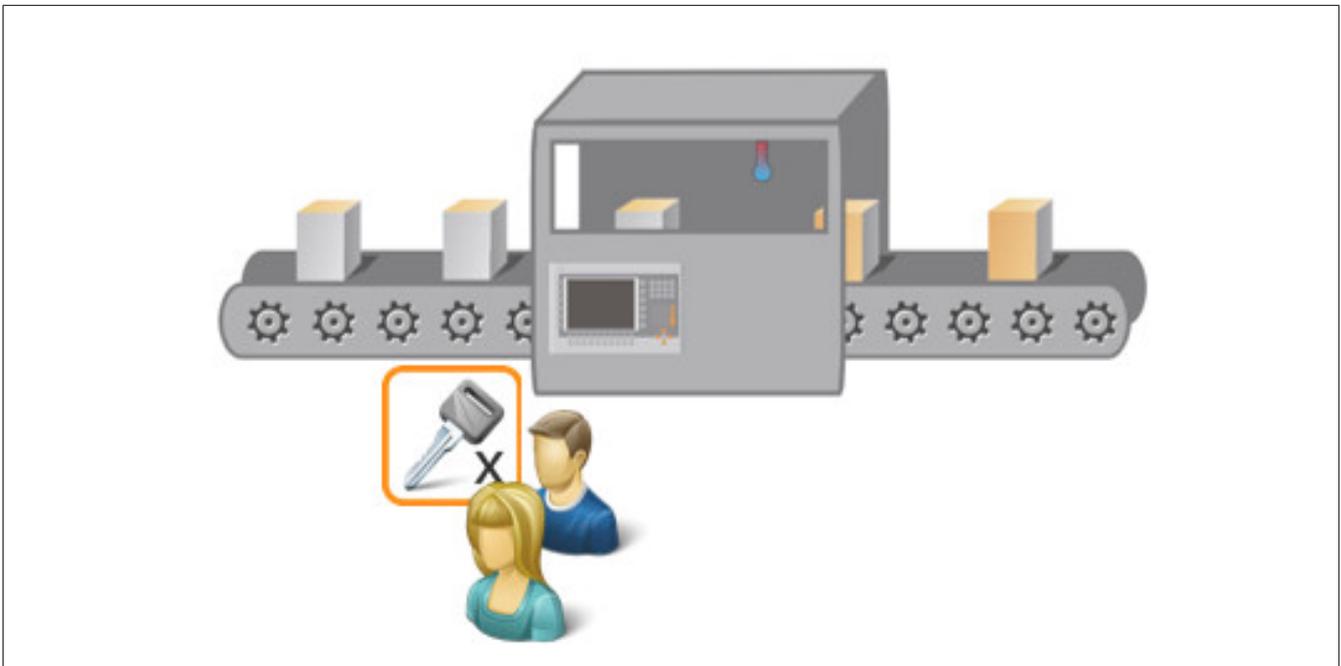
1.3.1 Modus des User managements

In der [MpUserX Konfiguration](#) kann unter "User management system settings" festgelegt werden in welchem Modus das User management arbeiten soll.

Es kann zwischen einem lokalen, zentralen oder lokal und zentralen System gewählt werden.

Lokales User management

Bei der Auswahl des lokalen User managements werden alle Rollen und Benutzer lokal an der Maschine verwaltet. Es können neue Rollen und Benutzer hinzugefügt, bearbeitet, gelöscht, exportiert werden und vieles mehr.



Zentrales User management

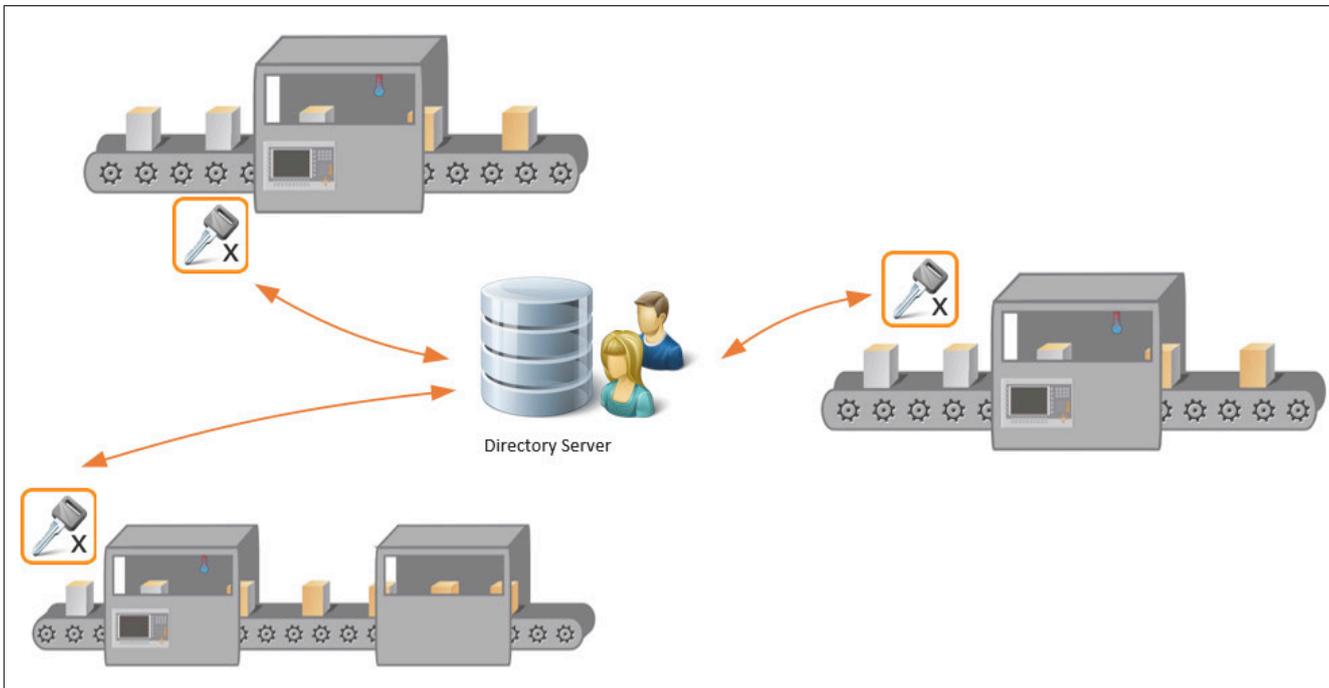
Beim zentralen User Management wird mapp UserX mit einem Directory Server kombiniert. Abhängig vom verwendeten Betriebssystem kann zwischen dem Active Directory Server und dem 389 Directory Server gewählt werden. Bei einem Windows-basierenden System muss der Active Directory Server verwendet werden. Bei einem Linux-basierenden System wird der 389 Directory Server gewählt.

Über diesen Server werden die Rollen und Benutzer verwaltet. mapp UserX kann durch die Verbindung auf die Rollen und Benutzer zugreifen und diese verwenden. Eine Verbindung zum Server muss nur während Aktionen, wie beispielsweise während eines Login-Vorgangs oder einer elektronischen Signatur, bestehen.

Als Kommunikationsprotokoll zwischen dem Zielsystem und dem Directory Server wird [LDAPS](#) ("Lightweight Directory Access Protocol Secure") verwendet.

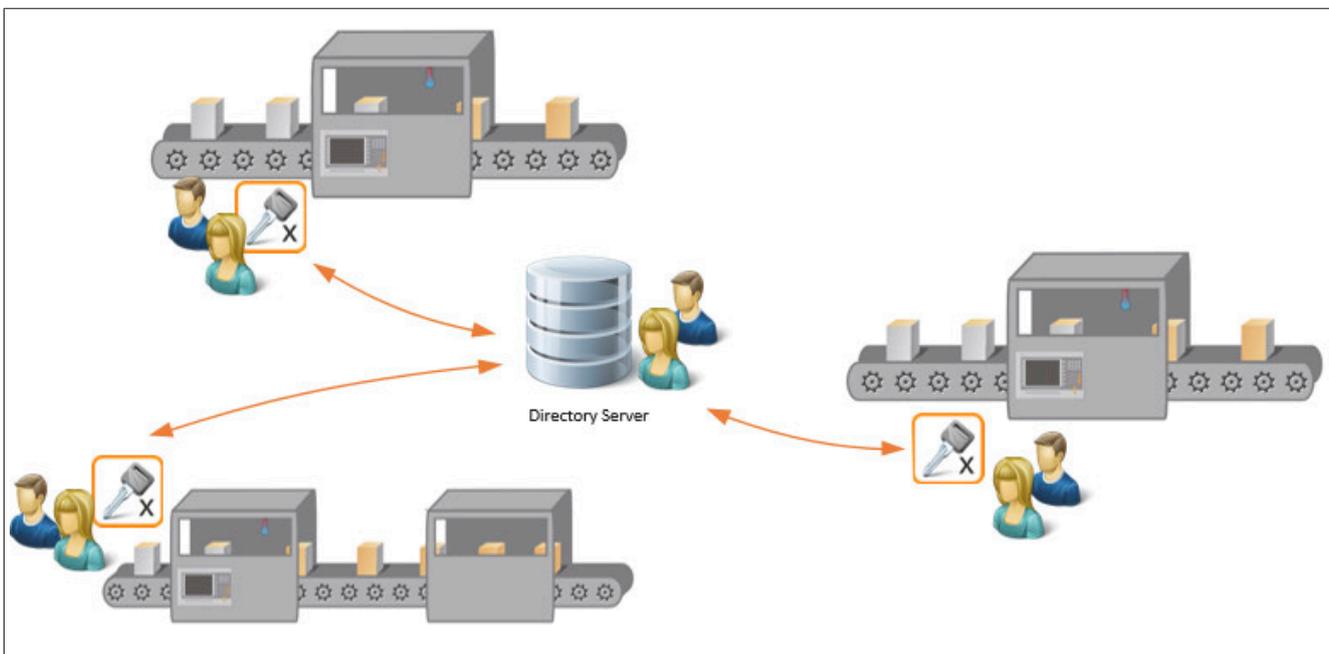
Der Vorteil dabei ist, dass mehrere Maschinen mit einem Directory Server verbunden sind und damit Zugriff auf das damit verbundene User Management haben. Das User Management erfolgt dadurch an einer zentralen Stelle.

Für mehr Information siehe Abschnitt [Zentrales User management](#).



Lokales und zentrales User management

Bei der Verwendung des lokalen und zentralen User managements können sowohl Benutzer eines Directory Servers verwendet werden, sowie lokale Rollen und Benutzer. Dabei gelten die gleichen Rahmenbedingungen wie in den oberen zwei Abschnitten dokumentiert wurde.



Das bedeutet, es können neue lokale Benutzer und Rollen hinzugefügt, gelöscht, exportiert und vieles mehr werden. Für Benutzer welche über einen Directory Server verwaltet werden, gelten die gleichen Rahmenbedingungen wie [hier](#) erklärt wird.

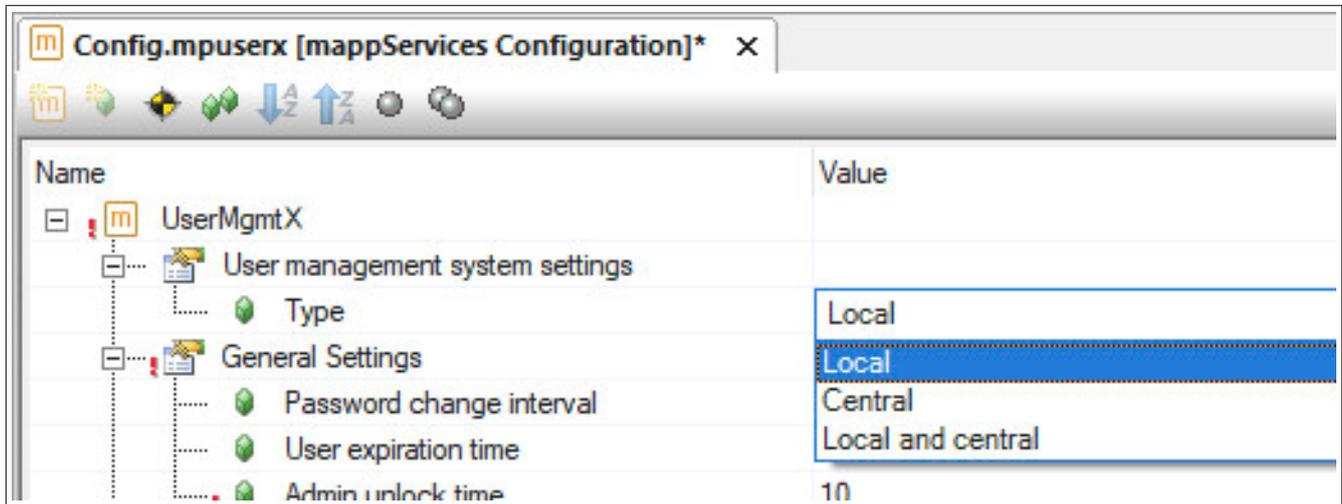
Für mehr Information siehe Abschnitt [Zentrales User management](#).

1.3.1.1 Zentrales User management

Um das zentrale User management verwenden zu können, wird die Automation Runtime \geq F4.44 benötigt.

Das zentrale User management kann innerhalb einer **Application Session** sowie **Visualization Session** verwendet werden.

Damit mapp UserX über ein zentrales User management arbeitet, kann in der [MpUserX Konfiguration](#) unter "User management system settings" zwischen den Typen "Central" und "Local and central" gewählt werden.



Bei der Auswahl einer dieser Typen arbeitet mapp UserX in Kombination mit einem Directory Server. Für mehr Information dazu siehe Abschnitt [Directory Server](#). Auf diesem Server sind alle Gruppen und Benutzer abgespeichert welche verwendet werden können. mapp UserX verbindet sich mit dem Directory Server und kann dadurch die dort angegebenen Gruppen und Benutzer verwenden.

Mappen der Rollen

Damit die Rollen und Benutzer des Directory Servers mit mapp UserX verwendet werden können, muss mithilfe der [MpUserX Konfiguration](#) ein Rollen-Mapping durchgeführt werden. Für mehr Informationen siehe Abschnitt [Mappen der Rollen](#).

Funktionen

Welche Punkte bei der Verwendung vom zentralen bzw. zentralen und lokalen User management zu beachten sind und welche Funktionen verwendet werden können, wird [hier](#) erklärt.

Zertifikat

Um eine sichere Verbindung zum Directory Server zu gewährleisten muss ein [Zertifikat](#) verwendet werden.

Rahmenbedingungen

Bei der Verwendung von mapp UserX in Kombination mit einem Directory Server gilt es die [Rahmenbedingungen](#) zu beachten!

1.3.1.1.1 Directory Server

Mithilfe eines Directory Servers kann ein zentrales User management erstellt werden. Abhängig vom verwendeten Betriebssystem können folgende Directory Server verwendet werden:

- **Active Directory Server:** Der Active Directory Server kommt bei Windows-basierenden Betriebssystemen zum Einsatz. Die Funktion wurde mit dem Windows Server 2019 getestet
- **389 Directory Server:** Für Linux-basierende Betriebssysteme kann der 389 Directory Server verwendet werden. Die Funktion wurde mit der Version 1.4.2.15 getestet

Die Rollen, welche am Directory Server als Benutzergruppen bezeichnet werden, sowie alle Benutzer werden am Server verwaltet.

Die Benutzer werden Benutzergruppen zugeteilt. Für einen Benutzer können zusätzliche Informationen festgelegt werden, wie zum Beispiel die Telefonnummer oder die E-Mail Adresse.

The image shows a Windows dialog box titled "Max Mustermann Properties". At the top, there are several tabs: "Published Certificates", "Member Of", "Password Replication", "Dial-in", "Object", "Security", "Environment", "Sessions", "Remote control", "Remote Desktop Services Profile", "COM+", and "Attribute Editor". The "General" tab is currently selected. Below the tabs, there is a user icon and the name "Max Mustermann". The main area contains several input fields: "First name:" with "Max", "Initials:" with "MM", "Last name:" with "Mustermann", "Display name:" with "Max Mustermann", "Description:", "Office:", "Telephone number:" with "1234567890" and an "Other..." button, "E-mail:" with "mmustermann@example.com", and "Web page:" with an "Other..." button. At the bottom, there are four buttons: "OK", "Cancel", "Apply", and "Help".

Abbildung 1: Festlegen zusätzliche Informationen am Active Directory Server

Als Kommunikationsprotokoll zwischen dem Zielsystem und dem Directory Server wird [LDAPS](#) ("Lightweight Directory Access Protocol Secure") verwendet. Über dieses Protokoll kann mapp UserX eine Verbindung zum Directory Server aufbauen.

Die Benutzer und Rollen am Directory Server können nur am Server selbst bearbeitet werden.

Verbindung zu Directory Server aufbauen

Damit mapp UserX in Kombination mit einem Directory Server arbeiten kann, muss eine Verbindung zum Server während Benutzeraktionen bestehen. Eine Benutzeraktion kann beispielsweise ein Login-Vorgang sein oder die Durchführung einer elektronischen Signatur.

Unter "Server Settings" in der [MpUserX Konfiguration](#) wird festgelegt welcher Directory Server verwendet werden soll. Im Anschluss werden die Verbindungs-Parameter wie Host-Name und Port-Nummer festgelegt. Mehr Information dazu kann [hier](#) unter "Server Einstellungen" gefunden werden. Die Verbindungs-Einstellungen können mithilfe des Funktionsbausteins [MpUserXServerConfig](#) zur Laufzeit geändert werden. Wie sich die restlichen Parameter in der Konfiguration zur Laufzeit auswirken können, ist im Abschnitt [Funktionen](#) erklärt.

Active Directory Server

Bei der Erstellung der Benutzergruppen am Active Directory Server kann festgelegt werden, ob eine Gruppe lokal, global oder universal ("Group scope") verwaltet werden soll. Zusätzlich kann auch ein Gruppentyp ("Group type") definiert werden.

Welche Einstellung dabei verwendet wird hat keinerlei Einfluss auf mapp UserX. Es kann ein beliebiger Geltungsbereich sowie Gruppentyp festgelegt werden.

389 Directory Server

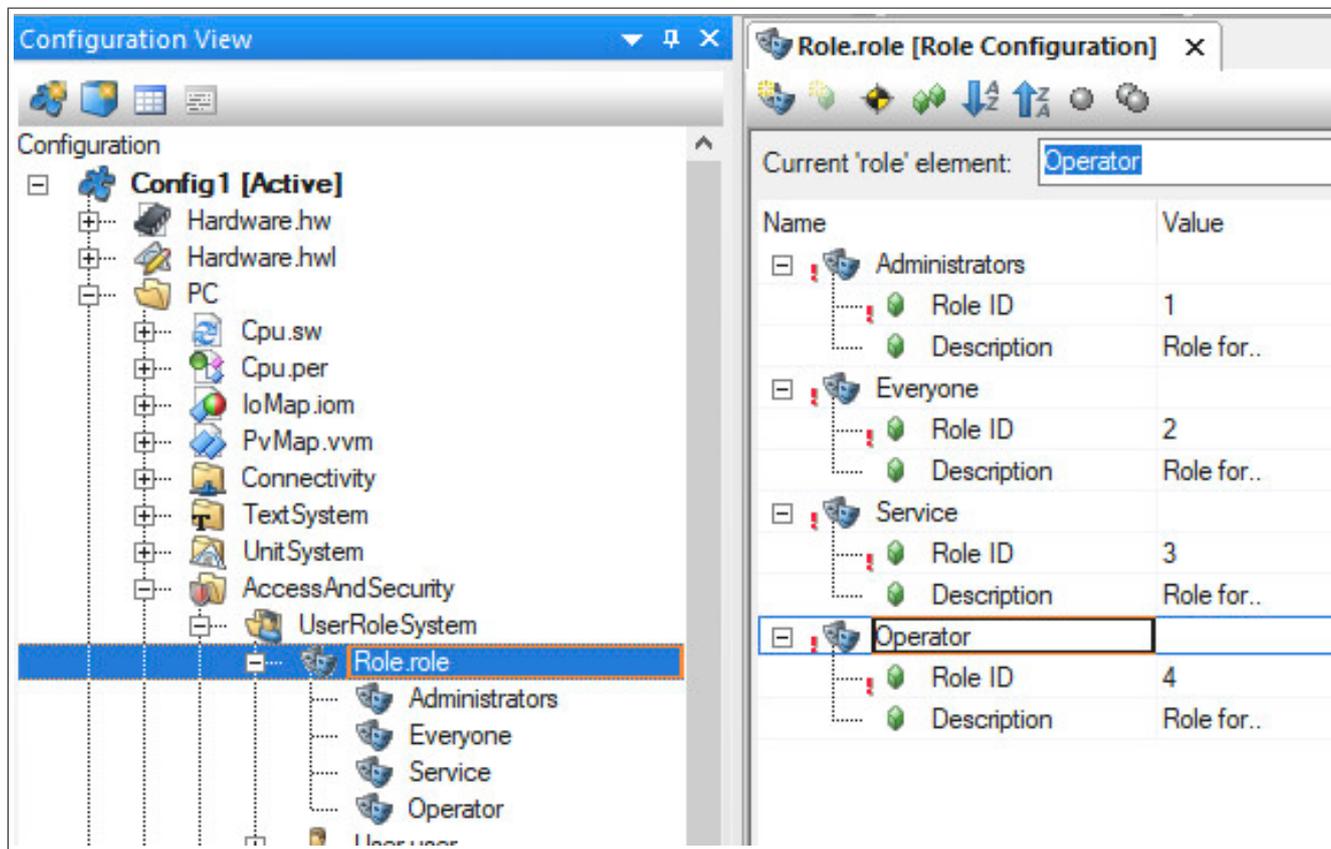
Damit sich im Anschluss Benutzer anmelden können, muss am 389 Directory Server das Plugin "memberOf" aktiviert sein:

1.3.1.1.2 Mappen der Rollen

Rollen und Benutzer werden im Directory Server verwaltet. Damit diese Rollen und Benutzer in mapp UserX verwendet werden können, müssen diese mit den in Automation Studio erstellten Rollen verbunden werden.

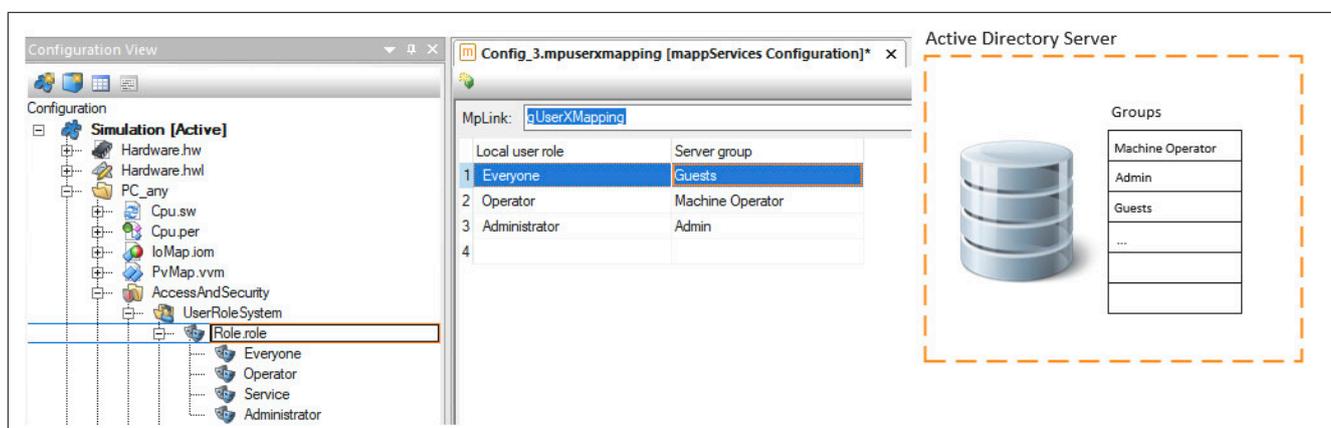
Das bedeutet, für das Zielsystem muss definiert werden, welche Rollen des Directory Servers verwendet werden. Die Benutzer müssen in Automation Studio nicht angegeben werden. Diese werden nur über den Directory Server verwaltet.

Die gewünschten Rollen werden über das Benutzer-Rollen-System in Automation Studio definiert. Der Name der Rolle muss dabei nicht mit den Rollen-Namen des Directory Servers identisch sein.

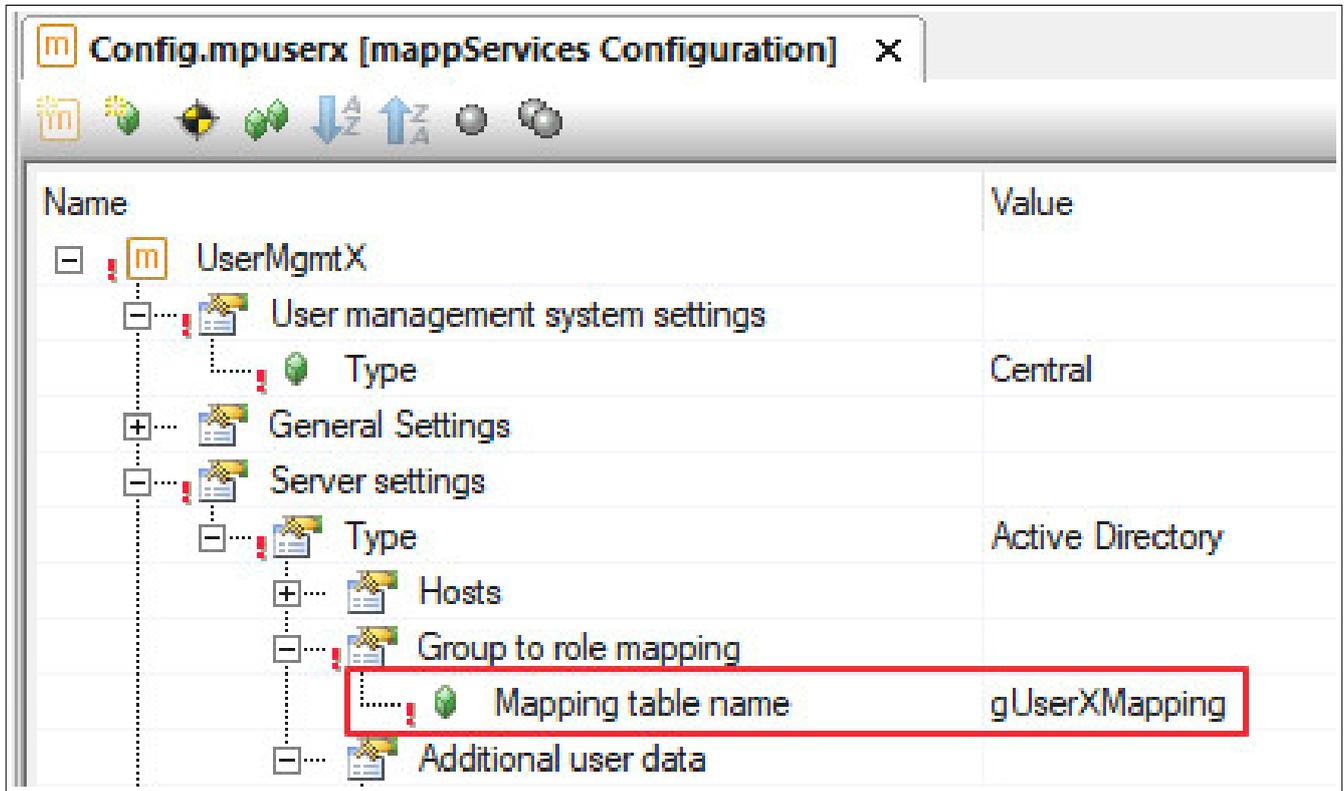


Um nun die Rollen in Automation Studio mit denen des Directory Servers zu verbinden ("mappen") wird die [MpUserXMapping Konfiguration](#) verwendet.

Unter "Local user role" werden die Rollen in Automation Studio angegeben und unter "Server group" jene des Directory Servers.



Diese Mapping-Tabelle muss zum Schluss in der [MpUserX Konfiguration](#) unter "Mapping table name" angegeben werden.

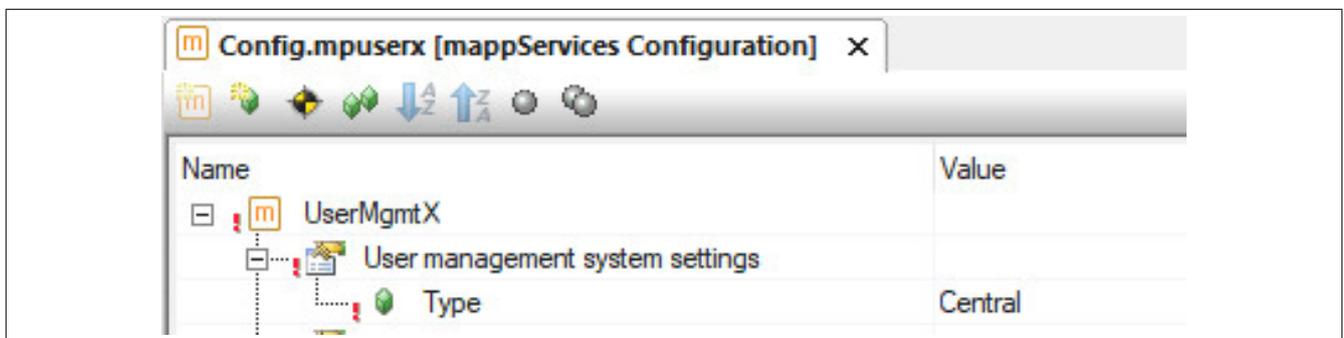


Das Mapping der Rollen kann mithilfe des Funktionsbausteins `MpUserXMappingConfig` zur Laufzeit geändert werden.

1.3.1.1.3 Funktionen

Sind alle Vorbereitungen abgeschlossen kann das User management am Zielsystem verwendet werden. Dabei gibt es bei der Verwendung zwischen dem zentralen und lokal und zentralen User management Unterschiede, welche beachtet werden müssen:

Zentrales User management

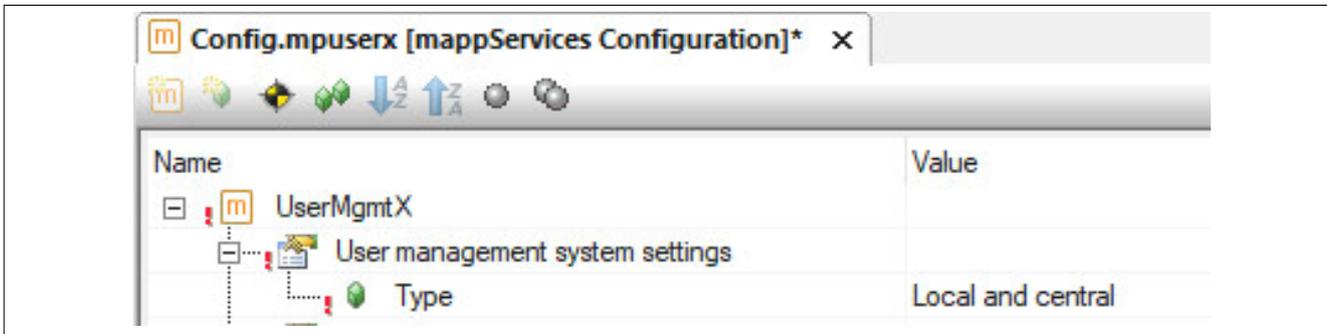


Wird in der `MpUserX Konfiguration` der Typ "Central" ausgewählt, so können nur die Benutzer und Rollen verwendet werden, welche am Directory Server liegen. Das bedeutet, es wird der Benutzername und das Passwort für einen Login-Vorgang verwendet, welches am Directory Server liegt. Ist ein Benutzer angemeldet, so kann nur eine Passwortänderung durchgeführt werden. Der Benutzer kann nicht weiter bearbeitet oder gelöscht werden. Andere Benutzeränderungen werden nur am Directory Server durchgeführt. Die Import-, und Export-Funktion kann bei Benutzern und Rollen, welche am Directory Server verwaltet werden, nicht verwendet werden. Am Zielsystem ist es nicht möglich neue Benutzer anzulegen. Dies ist nur über den Directory Server realisierbar. Wird eine Aktion im zentralen User management ausgelöst welche nicht durchgeführt werden kann, so wird der Fehler "Aktion nicht erlaubt" angezeigt.

Wichtig ist, dass der Directory Server während eines Login-Vorgangs erreichbar ist. Ansonsten kann der Login-Vorgang nicht abgeschlossen werden und es wird der Fehler "Keine Server-Verbindung" angezeigt.

Bei einer Eingabe eines falschen Benutzernamens oder Passworts wird ebenfalls ein Fehler ("Ungültige Eingabe") angezeigt.

Lokales und zentrales User mangements



Wird in der [MpUserX Konfiguration](#) der Typ "Local and central" ausgewählt, so können die Benutzer und Rollen verwendet werden, welche am Directory Server liegen, sowie jene welche in Automation Studio erstellt wurden. Für die Rollen und Benutzer, welche am Directory Server liegen, gelten die gleichen Regeln wie oben im Punkt "Zentrales Benutzermanagement" beschrieben.

Benutzernamen in Automation Studio dürfen nicht mit denen am Directory Server übereinstimmen. Bei Verwendung gleicher Benutzernamen kann es ansonsten zu Authentifizierungsproblemen kommen. Die lokal erstellten Benutzer und Rollen können zur Laufzeit am Zielsystem bearbeitet oder gelöscht werden. Zur Laufzeit können auch neue Benutzer und Rollen hinzugefügt werden. Die Import-, und Export-Funktion kann nur bei lokal erstellten Benutzern und Rollen verwendet werden.

Festlegen von Rechten pro Maschine

Pro Maschine können die Rechte der Rollen individuell in der [MpUserX Konfiguration](#) angepasst werden. Dadurch können die Rechte, beispielsweise abhängig vom verwendeten Maschinentypen, angepasst werden. Für mehr Information, siehe Abschnitt [Verwalten von Berechtigungen](#).

<ul style="list-style-type: none"> UserMgmtX <ul style="list-style-type: none"> User management system settings General Settings Server settings Roles <ul style="list-style-type: none"> Role: Administrators Role: Operator <ul style="list-style-type: none"> Name Level Administrator Rights <ul style="list-style-type: none"> Access right 1 Access right 2 Access right 3 	<ul style="list-style-type: none"> General settings to define which user management system is used Local user management system specific settings Central user management system specific settings Operator 5 FALSE Access rights Full View None Move Axis 1 Setting page Recipe management 	<ul style="list-style-type: none"> Role from URS (User / Role System) User level Defines whether the role has administrator rights
--	--	---

Änderungen zur Laufzeit

Zur Laufzeit können mithilfe der Funktionsbausteine [MpUserXServerConfig](#) und [MpUserXMappingConfig](#) Änderungen an den Verbindungs-Parametern zum Directory Server durchgeführt werden, sowie die [Mapping-Tabelle](#) angepasst werden.

Das bedeutet, zur Laufzeit kann sich das Zielsystem mit einem anderen Directory Server verbinden. Wichtig hierbei ist, dass die [Mapping-Tabelle](#) mit den Rollen des neuen Directory Servers übereinstimmen.

Stimmen die Rollen nicht überein und es wird ein Login-Vorgang mit dem richtigen Benutzernamen und Passwort gestartet, so loggt sich der Benutzer im System ein, hat aber keine Rollen-Zuordnung. Das bedeutet, Rollenbasierte-Rechte werden nicht angewendet. Der Benutzer hat keine Rechte.

Anzeigen von Benutzerinformation

Für einen Benutzer können zusätzliche Benutzerinformationen am Directory Server hinterlegt werden. Dies kann beispielsweise die Telefonnummer, E-Mail Adresse oder das Land in dem der Benutzer ist, sein.

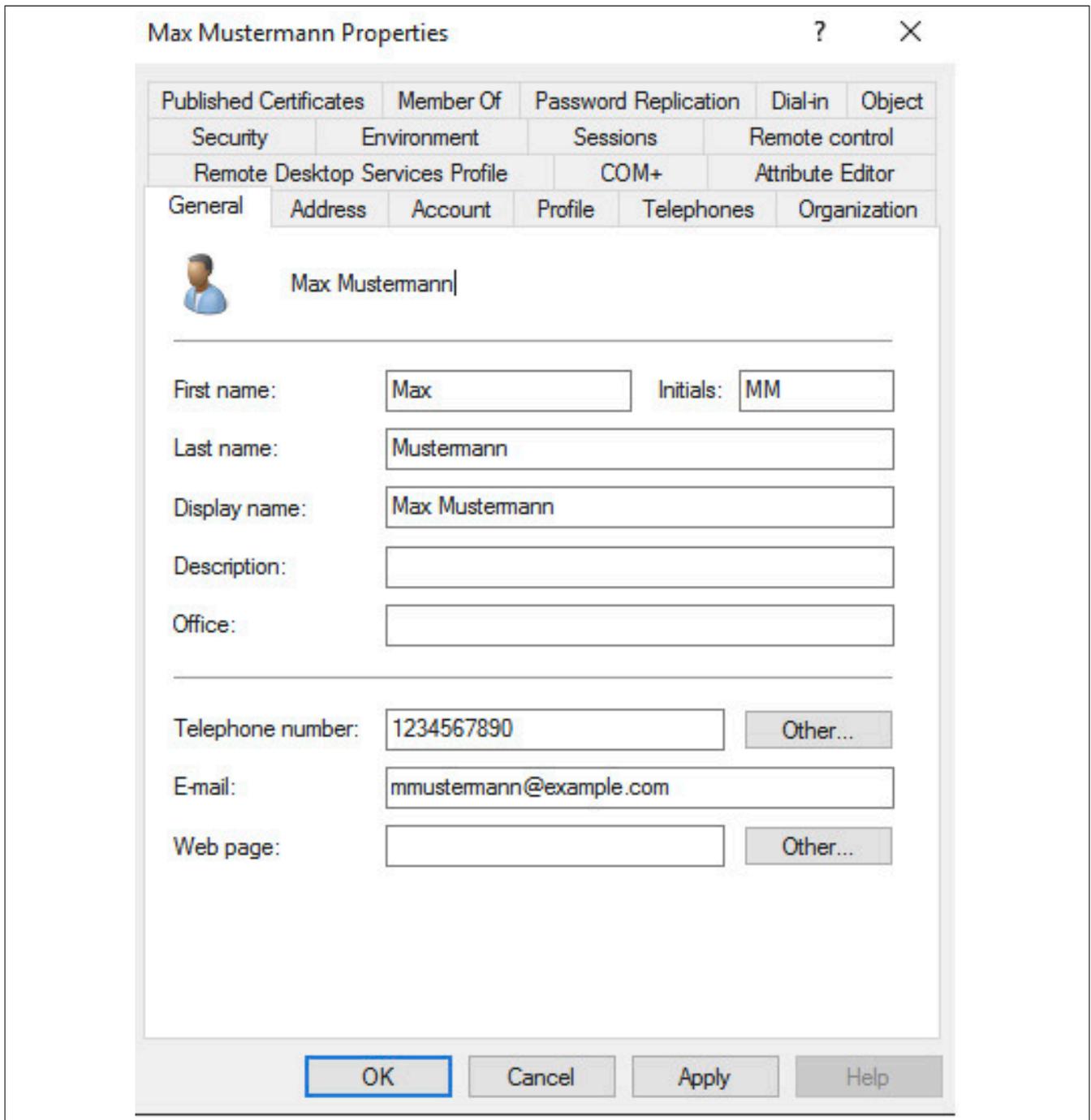
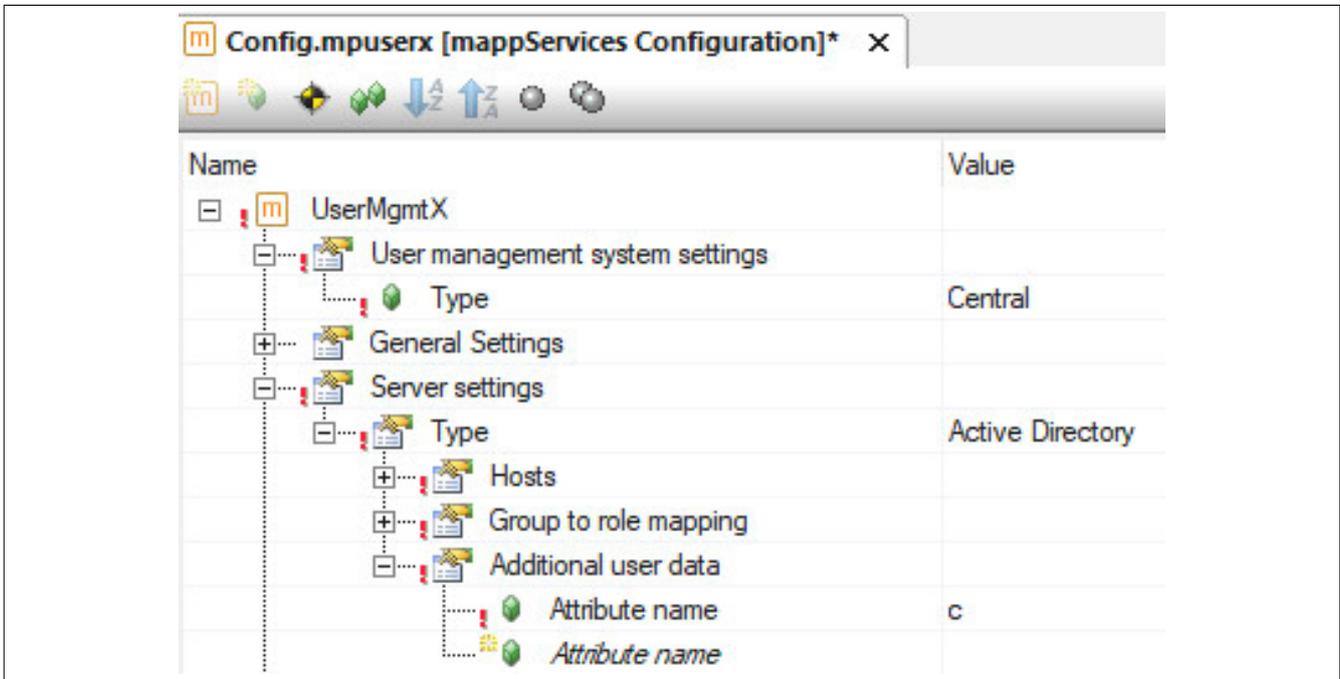


Abbildung 2: Festlegen von zusätzlichen Informationen am Active Directory Server

Um diese zusätzliche Information am Zielsystem verwenden zu können, um diese beispielsweise an einer Visualisierung darzustellen, kann der Parameter "Additional user data" der [MpUserX Konfiguration](#) verwendet werden. Welche zusätzliche Information angezeigt werden soll, wird über "Attribute name" definiert.



Die Verbindung zum Directory Server wird über LDAPS hergestellt. Das bedeutet, es dürfen nur LDAP leserbare Attribute unter "Attribute name" verwendet werden. Welche LDAP Attribute existieren ist [hier](#) zu finden.

Soll also beispielsweise das dem Benutzer zugeordnete Land angezeigt sein, so wird der Attribut-Name "c" angegeben.

Die Funktion würde für den Attribut-Namen "c" zum Beispiel "AT" für Österreich zurückliefern.

Standardmäßig wird für jeden Benutzer die Information der Telefonnummer ("telephoneNumber"), E-Mail-Adresse ("mail") sowie die bevorzugte Sprache ("preferredLanguage") abgefragt.

Das bedeutet, die Funktion `MpUserXGetData` kann diese Information des Benutzers anzeigen, ohne dass diese über "Attribute name" definiert wurde. Wichtig hierbei ist, dass diese Standard-Information des Benutzers am Directory Server festgelegt werden muss.

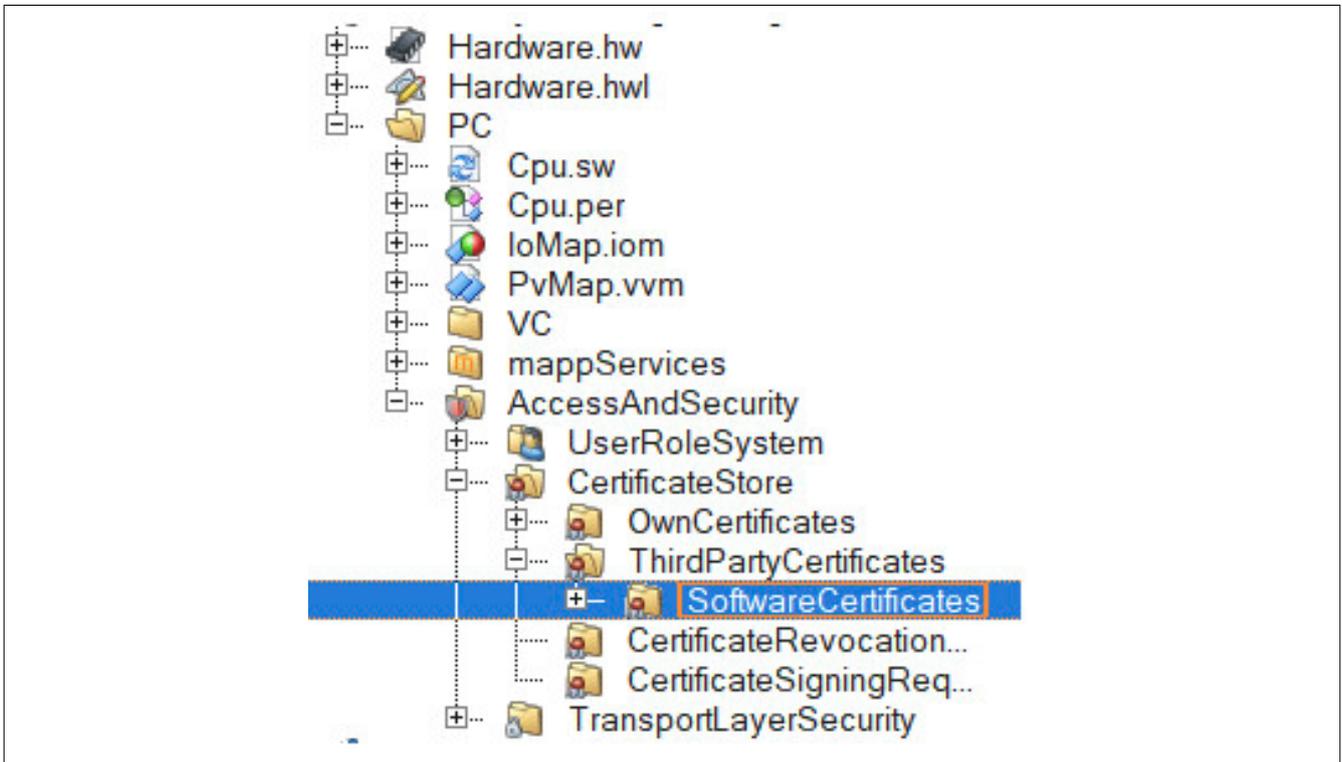
1.3.1.1.4 Zertifikat

Für allgemeine Informationen zu Zertifikaten in B&R Systemen siehe [hier](#).

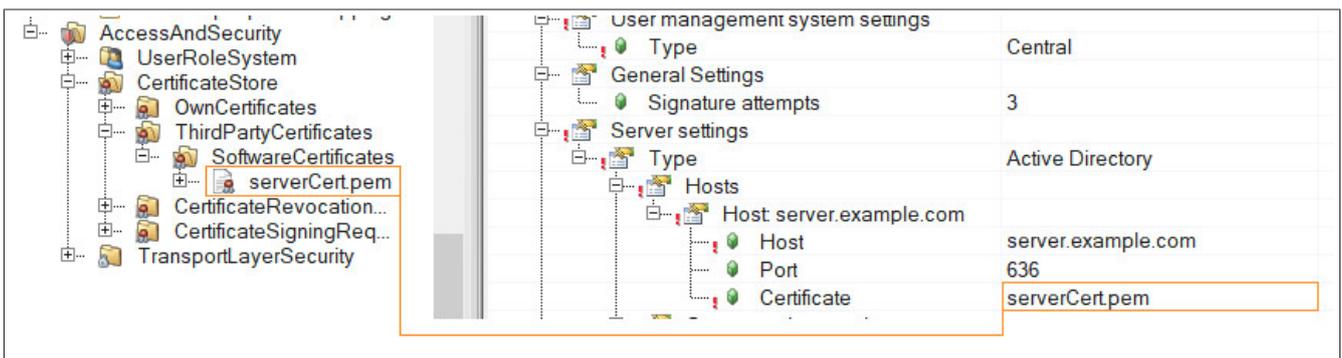
Um eine sichere Verbindung zum Directory Server zu gewährleisten, muss das Root-Authority-Zertifikat des Servers verwendet werden. Wie dieses Zertifikat exportiert werden kann ist [hier](#) erklärt. Es muss ein base64 codiertes X.509 Zertifikat verwendet werden.

Bei Verwendung des Active Directory Servers kann das Zertifikat die Dateierweiterungen ".pem", ".cer" oder ".crt" besitzen. Wird der 389 Directory Server verwendet, so kann nur ein Zertifikat mit der Endung ".pem" angegeben werden.

Das exportierte Zertifikat wird unter "AccessAndSecurity" in Automation Studio abgelegt:



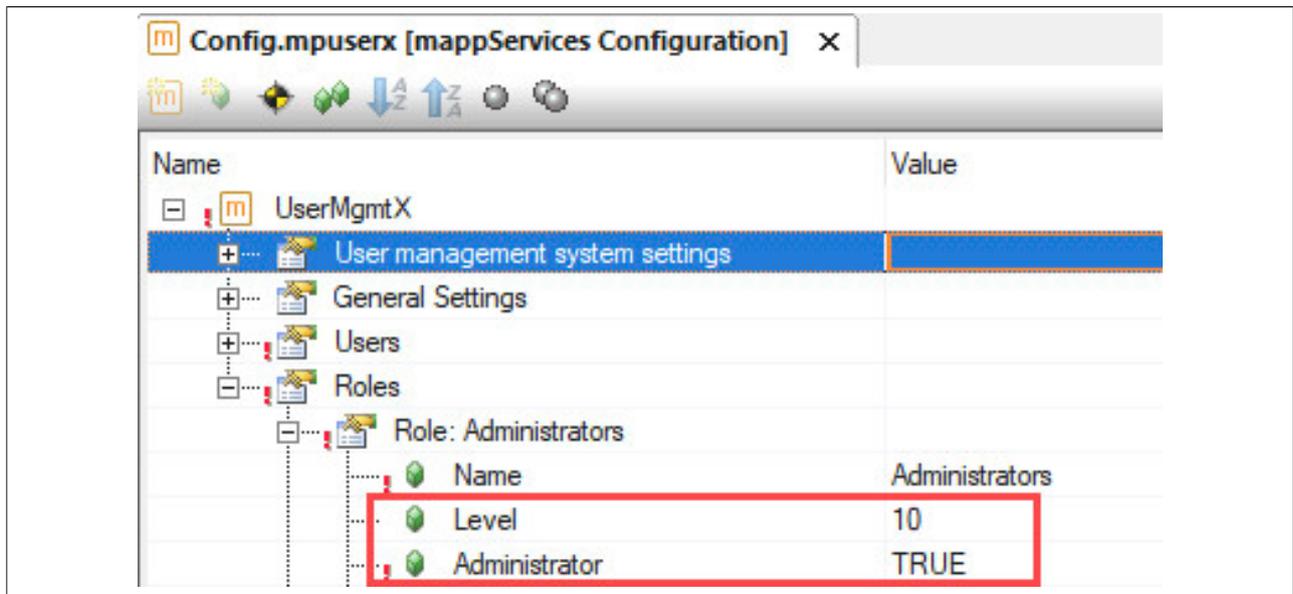
Im Anschluss wird das Zertifikat in der [MpUserX Konfiguration](#) unter "Certificate name" angegeben.



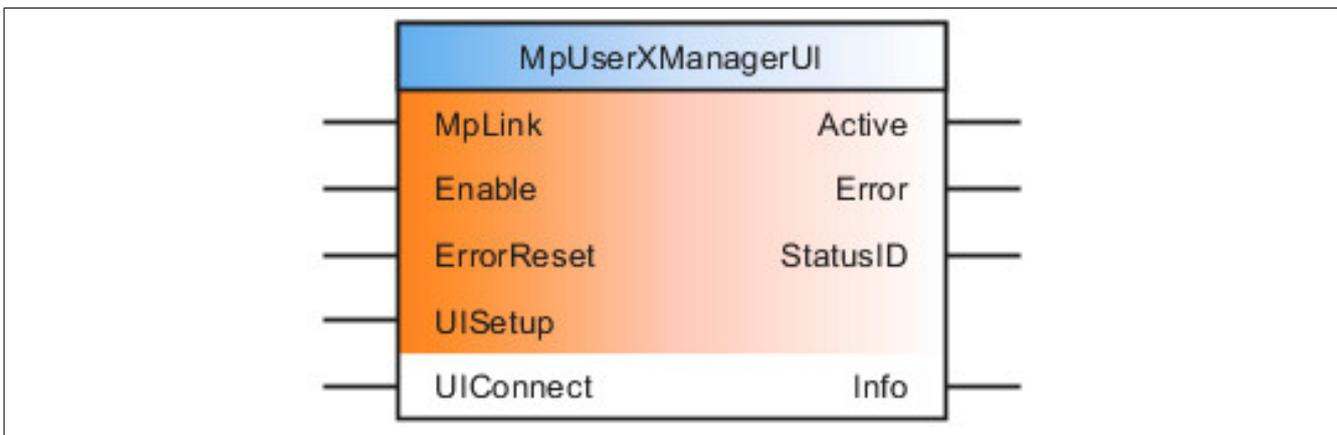
1.3.2 User management Funktionen

Mithilfe des User managements können verschiedene Funktionen verwendet werden. Dabei gibt es zwei verschiedene Möglichkeiten diese zu implementieren. Applikativ, mithilfe den hier beschriebenen Funktionen, oder visuell über eine mapp View Visualisierung. Dabei werden die User management Funktionen über das UserList-Widget verwaltet. Für mehr Information siehe Abschnitt [Anbindung an mapp View](#).

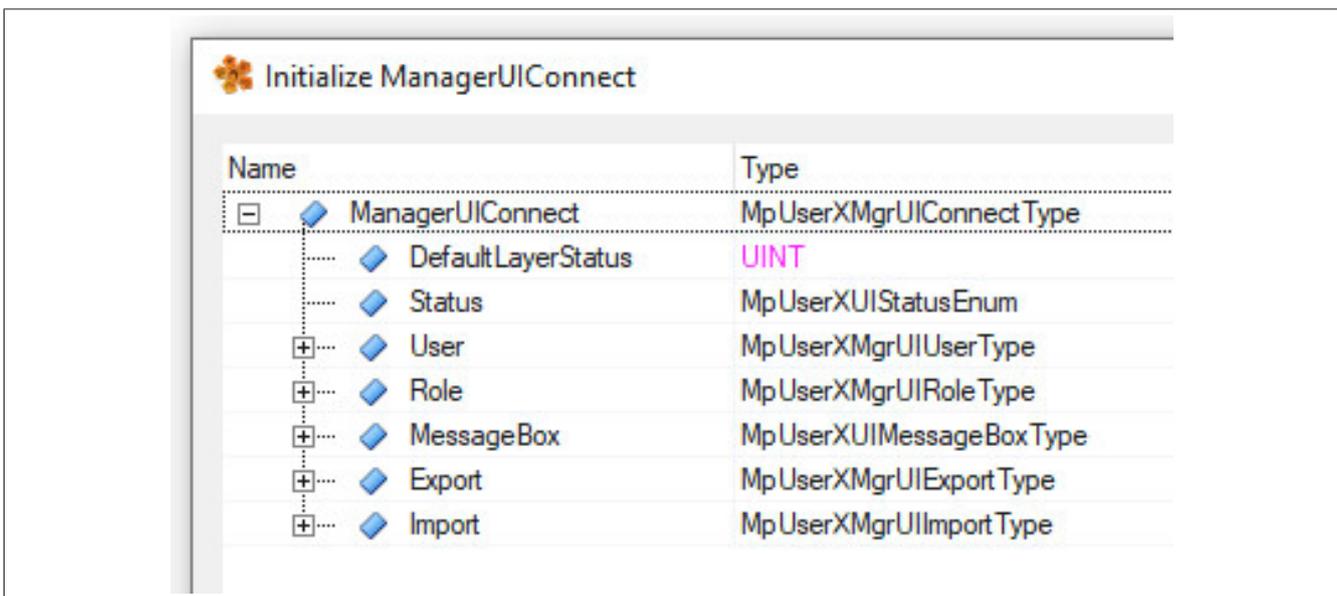
Nur ein Administrator hat die Möglichkeit andere Benutzer und Rollen zu bearbeiten bzw. neue Benutzer und Rollen zu erstellen/löschen. Wichtig hierbei ist, dass das Benutzerlevel des Administrators höher sein muss als die der anderen Benutzer.



Die Funktionen werden mithilfe des Funktionsbausteins `MpUserXManagerUI` realisiert.



Dazu muss am Eingangsparameter "UIConnect" eine Prozessvariable vom Datentypen `MpUserXMgrUIConnectType` angehängt werden.



Über diese Prozessvariable können folgende Funktionen verwendet werden:

Auslesen von Benutzerinformation

Über die "UIConnect" Prozessvariable können Informationen der einzelnen Benutzer angezeigt werden.

Welche Benutzer vorhanden sind wird unter "List" angezeigt. Abhängig von "SelectedIndex" wird unter "Info" die Benutzerinformation des gewünschten Benutzers angezeigt:

The screenshot shows two views of the ManagerUIConnect process variable. The left view shows the 'List' section with a table of users. The right view shows the 'Info' section for the selected user 'Dave'. A red arrow points from the 'SelectedIndex' value of 2 in the left view to the 'Info' section in the right view.

UserNames	Value
UserNames[0]	"Admin"
UserNames[1]	"Anonymous"
UserNames[2]	"Dave"
UserNames[3]	"Jane"
UserNames[4]	""
UserNames[5]	""
UserNames[6]	""
UserNames[7]	""
UserNames[8]	""
UserNames[9]	""
UserNames[10]	""
UserNames[11]	""
UserNames[12]	""
UserNames[13]	""
UserNames[14]	""
UserNames[15]	""
UserNames[16]	""
UserNames[17]	""
UserNames[18]	""
UserNames[19]	""

Property	Value
UserName	"Dave"
FullName	"Dave"
Roles	
Locked	FALSE
Expired	FALSE
RemainingAttempts	3
Language	"
DisplayUnit	"
Creation	DT#2020-04-30-09:02:42
FirstLogin	DT#1970-01-01-00:00:00
LastLogin	DT#2020-04-30-09:02:42
PasswordExpired	DT#2106-02-07-06:28:15
AdditionalData	

Auslesen von Rolleninformation

Über die "UIConnect" Prozessvariable können Informationen der einzelnen Rollen angezeigt werden.

Welche Rollen vorhanden sind wird unter "List" angezeigt. Abhängig von "SelectedIndex" wird unter "Info" die Rolleninformation der gewünschten Rolle angezeigt:

The screenshot shows two views of the ManagerUIConnect process variable. The left view shows the 'List' section with a table of roles. The right view shows the 'Info' section for the selected role 'Administrators'. A red arrow points from the 'SelectedIndex' value of 1 in the left view to the 'Info' section in the right view.

Names	Value
Names[0]	""
Names[1]	"Administrators"
Names[2]	"Everyone"
Names[3]	"Operator"
Names[4]	"Service"
Names[5]	""
Names[6]	""
Names[7]	""
Names[8]	""
Names[9]	""

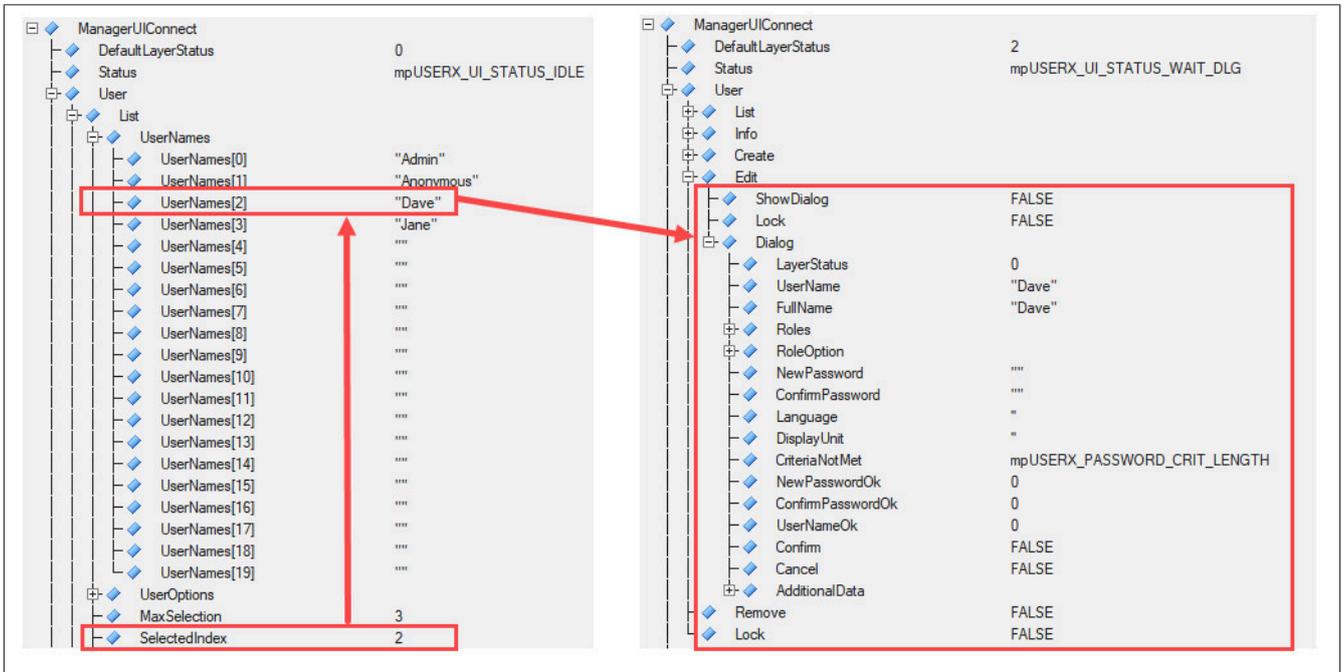
Property	Value
Name	"Administrators"
Index	0
Level	10
Admin	TRUE
AccessRights	

Benutzer editieren

Mithilfe der "UIConnect" Prozessvariable kann ein Benutzer editiert werden. Dadurch kann beispielsweise das Passwort geändert oder die Rollenzuordnung angepasst werden.

Dazu werden die Parameter unter "Edit" verwendet. Die Struktur ist so ausgelegt, dass sie beispielsweise mit einem Dialog verbunden werden kann.

Über "SelectIndex" wird definiert welcher Benutzer editiert werden soll. Durch den Parameter "ShowDialog" wird der Editier-Vorgang gestartet.



Durch "Confirm = TRUE" bzw. "Cancel = TRUE" wird die Editierung bestätigt und geändert bzw. verworfen.

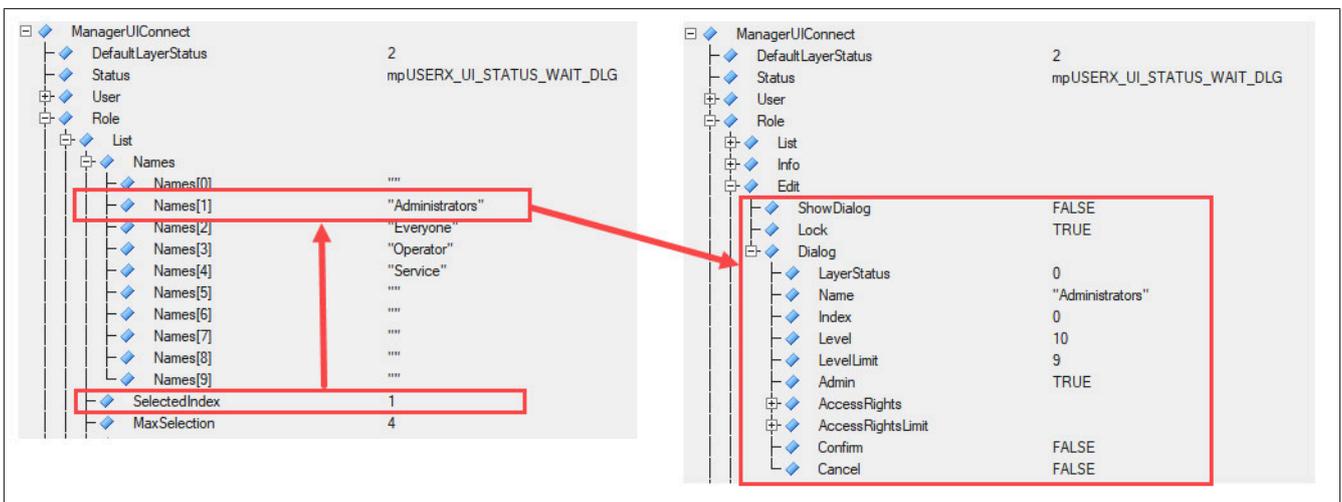
Wie die einzelnen Parameter unter "Edit" verwendet werden können wird [hier](#) erklärt.

Rollen editieren

Mithilfe der "UIConnect" Prozessvariable kann eine Rolle editiert werden. Dadurch kann beispielsweise das Benutzerlevel oder die Zugriffsrechte angepasst werden.

Dazu werden die Parameter unter "Edit" verwendet. Die Struktur ist so ausgelegt, dass sie beispielsweise mit einem Dialog verbunden werden kann.

Über "SelectIndex" wird definiert welche Rolle editiert werden soll. Durch den Parameter "ShowDialog" wird der Editier-Vorgang gestartet.



Durch "Confirm = TRUE" bzw. "Cancel = TRUE" wird die Editierung bestätigt und geändert bzw. verworfen.

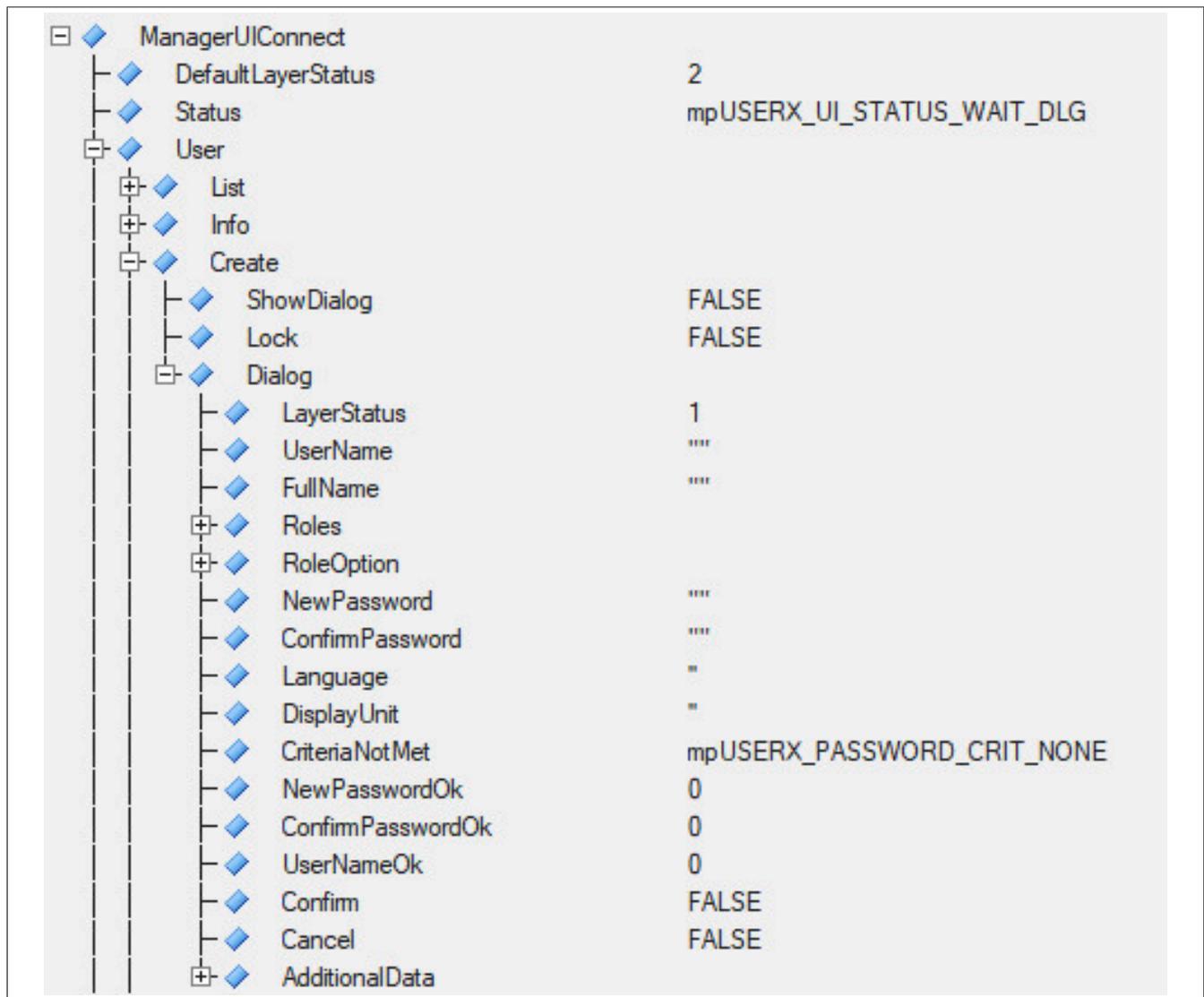
Wie die einzelnen Parameter unter "Edit" verwendet werden können wird [hier](#) erklärt.

Hinzufügen von Benutzern

Mithilfe der "UIConnect" Prozessvariable kann ein Benutzer hinzugefügt werden.

Dazu werden die Parameter unter "Create" verwendet. Die Struktur ist so ausgelegt, dass sie beispielsweise mit einem Dialog verbunden werden kann.

Durch den Parameter "ShowDialog" wird der Erstell-Vorgang gestartet.



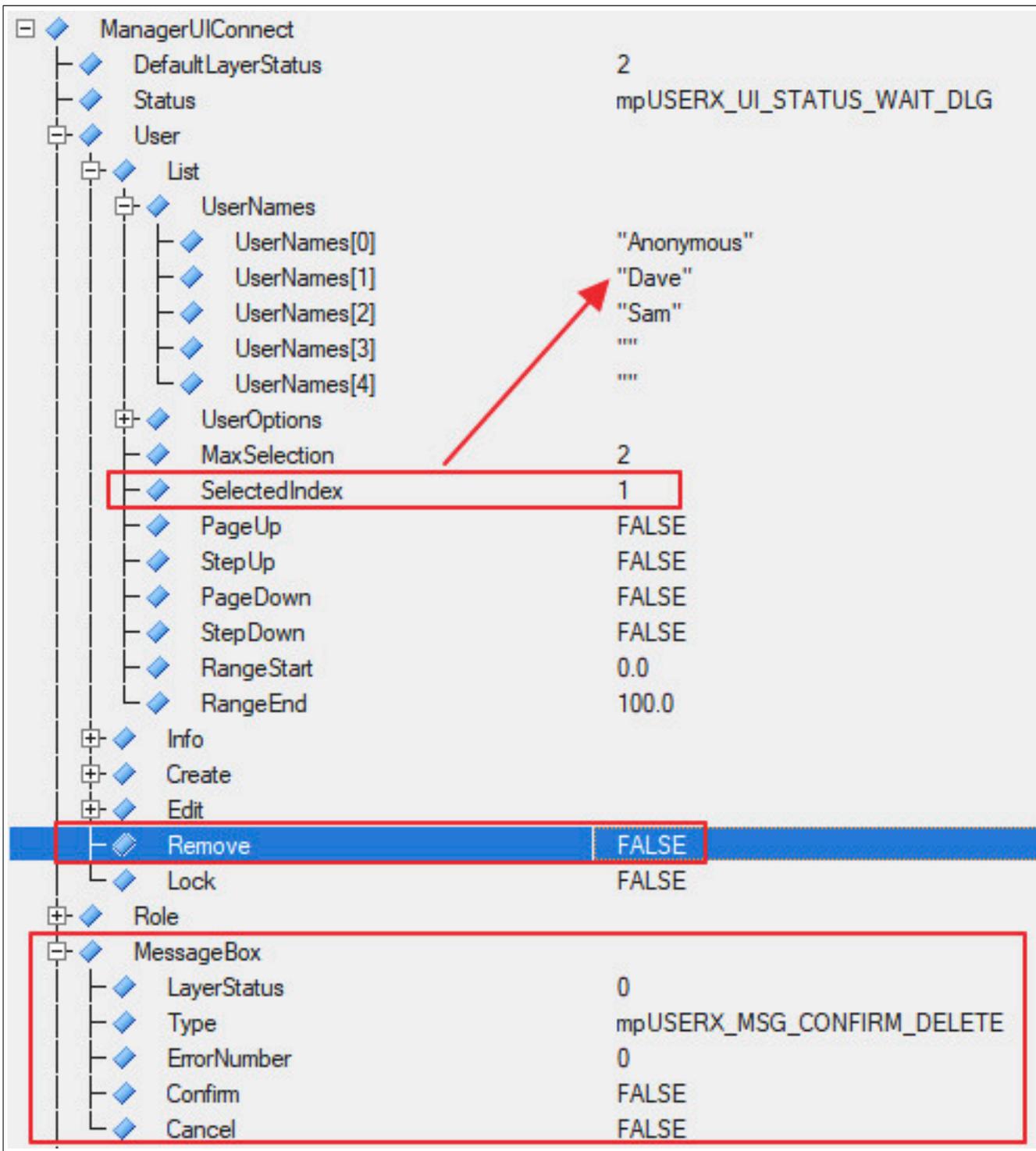
Wie die einzelnen Parameter unter "Create" verwendet werden können wird [hier](#) erklärt.

Durch "Confirm = TRUE" bzw. "Cancel = TRUE" wird die Erstellung des Benutzers bestätigt bzw. verworfen.

Löschen von Benutzern

Mithilfe der "UIConnect" Prozessvariable kann ein Benutzer gelöscht werden.

Über "SelectIndex" wird definiert welcher Benutzer gelöscht werden soll. Dazu wird der Parameter "Remove" verwendet. Unter "ShowMessageBox" muss das Löschen des Benutzers bestätigt werden. Die Struktur ist so ausgelegt, dass sie beispielsweise mit einem Dialog verbunden werden kann.



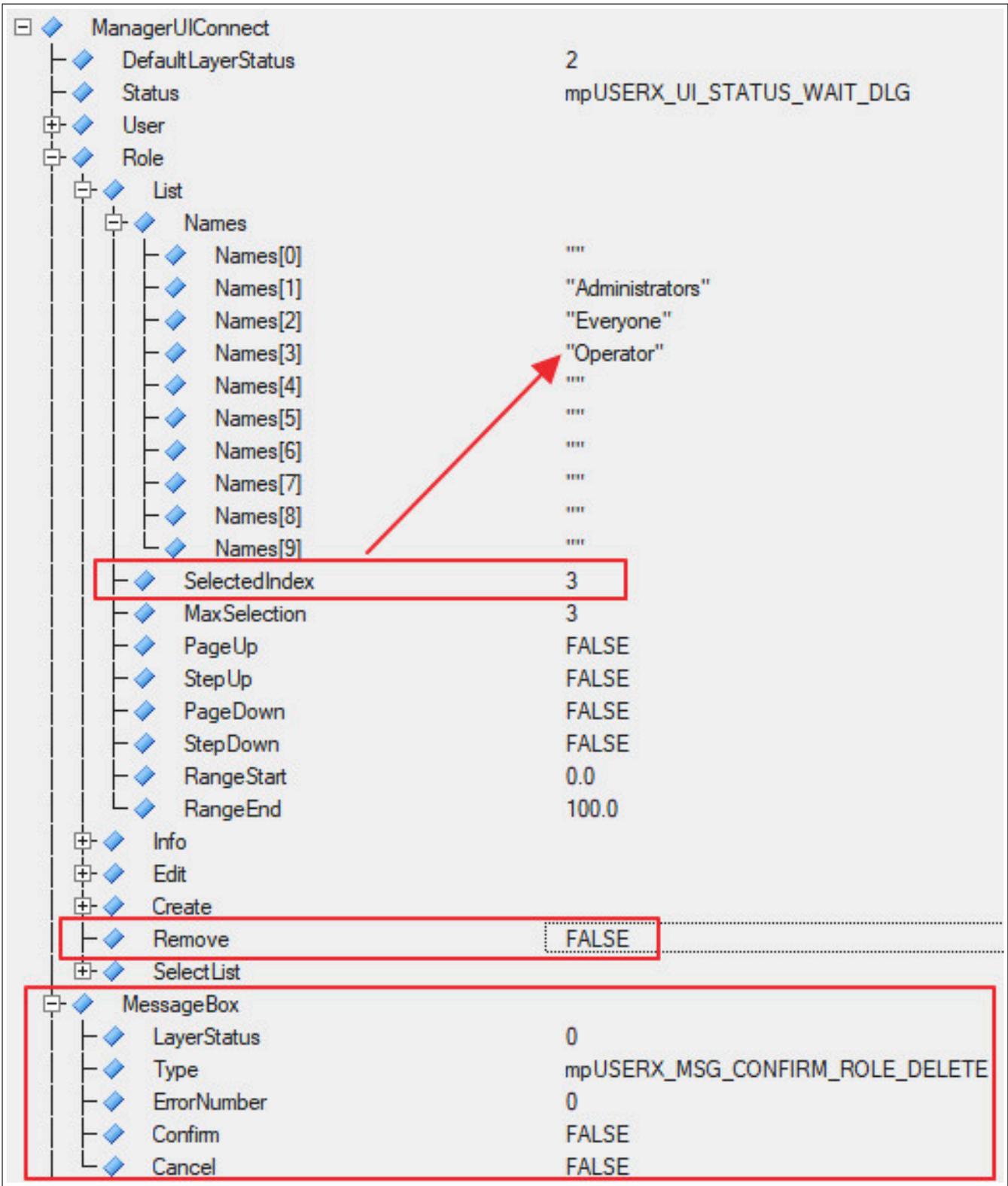
Durch "Confirm = TRUE" bzw. "Cancel = TRUE" wird das Löschen des Benutzers bestätigt bzw. verworfen.

Wird der Parameter "Prevent username reuse = TRUE" in der [MpUserX Konfiguration](#) gesetzt, so kann der Benutzername, welchen der gelöschte Benutzer verwendet hat, nicht mehr vergeben werden. Durch diese Funktion wird sichergestellt, dass Benutzernamen nur einmalig im System verwendet werden können. Wurde also beispielsweise der Benutzer "Dave" gelöscht, so kann kein Benutzer mehr mit diesen Namen erstellt werden. Dadurch wird sichergestellt, dass beispielsweise in Ereignislisten keine Verwechslung im Benutzernamen stattfindet.

Löschen von Rollen

Mithilfe der "UIConnect" Prozessvariable kann eine Rolle gelöscht werden.

Über "SelectIndex" wird definiert welche Rolle gelöscht werden soll. Dazu wird der Parameter "Remove" verwendet. Unter "ShowMessageBox" muss das Löschen der Rolle bestätigt werden. Die Struktur ist so ausgelegt, dass sie beispielsweise mit einem Dialog verbunden werden kann.



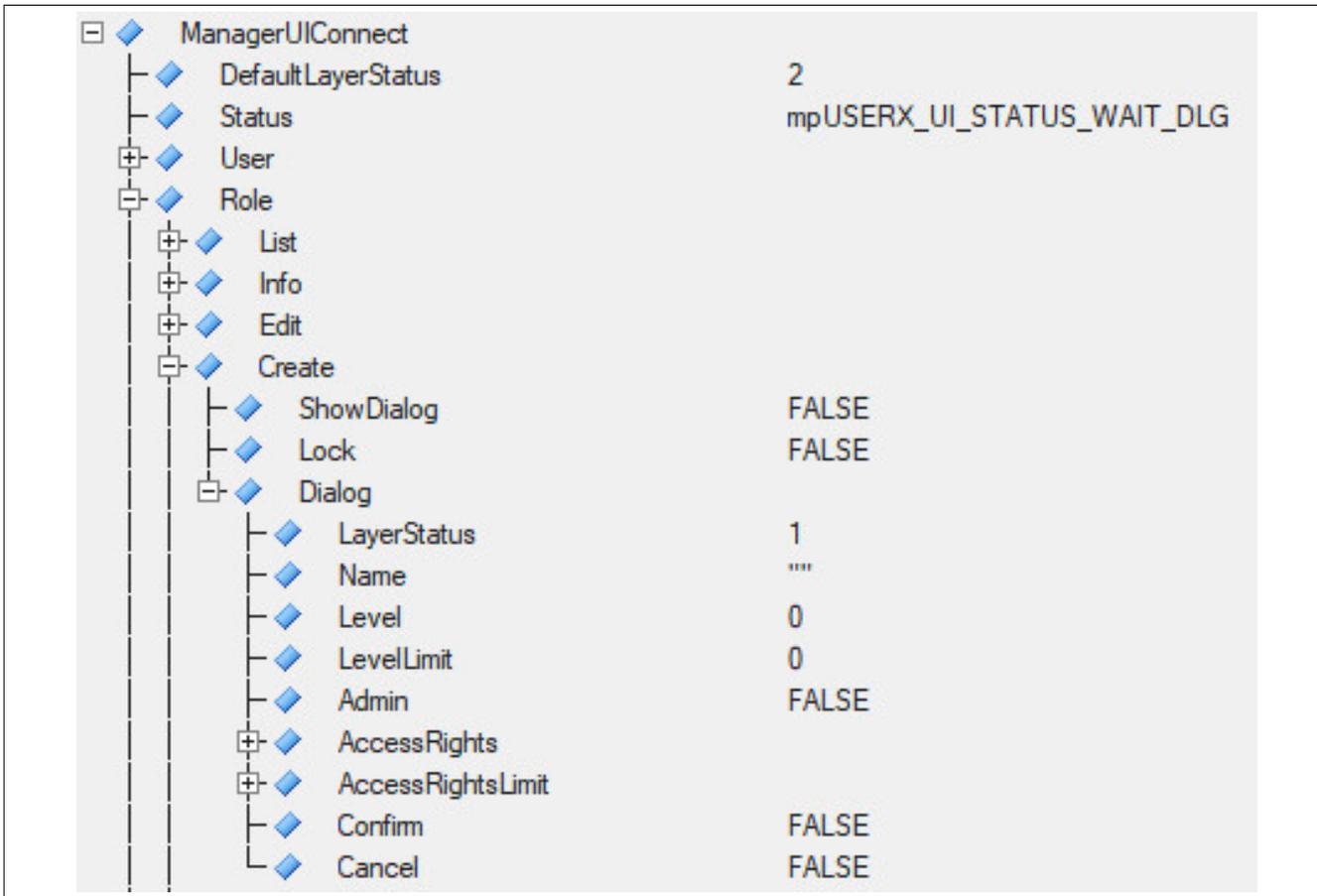
Durch "Confirm = TRUE" bzw. "Cancel = TRUE" wird das Löschen der Rolle bestätigt bzw. verworfen.

Hinzufügen von Rollen

Mithilfe der "UIConnect" Prozessvariable kann eine Rolle hinzugefügt werden.

Dazu werden die Parameter unter "Create" verwendet. Die Struktur ist so ausgelegt, dass sie beispielsweise mit einem Dialog verbunden werden kann.

Durch den Parameter "ShowDialog" wird der Erstell-Vorgang gestartet.

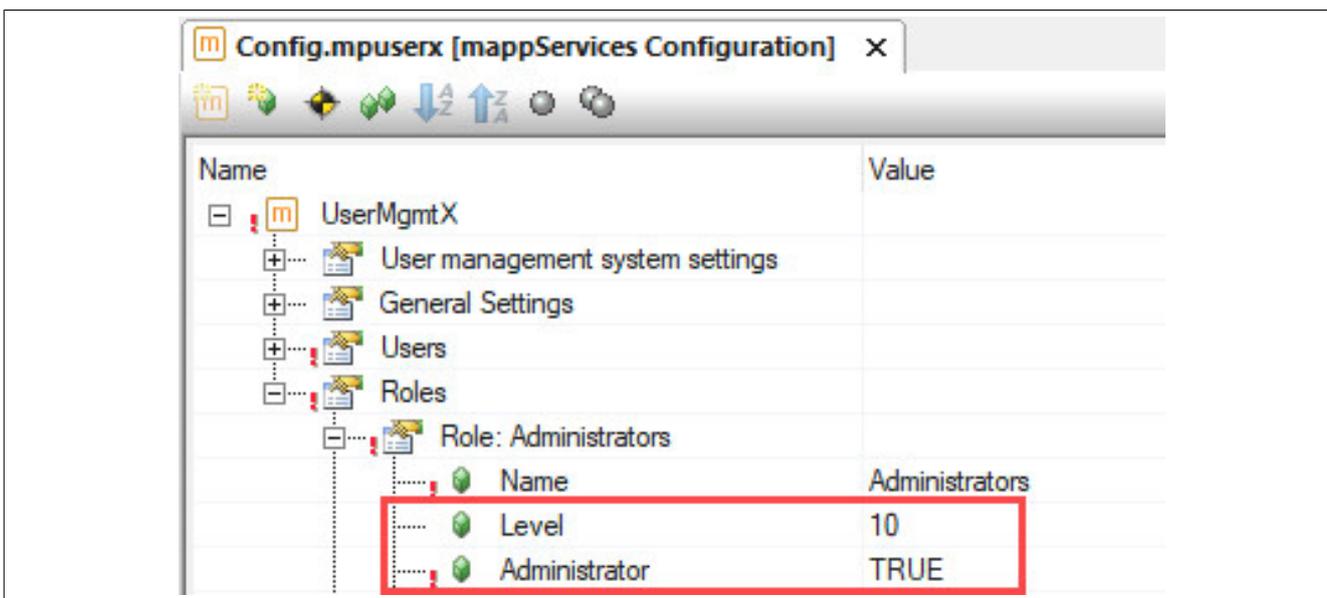


Wie die einzelnen Parameter unter "Create" verwendet werden können wird [hier](#) erklärt.

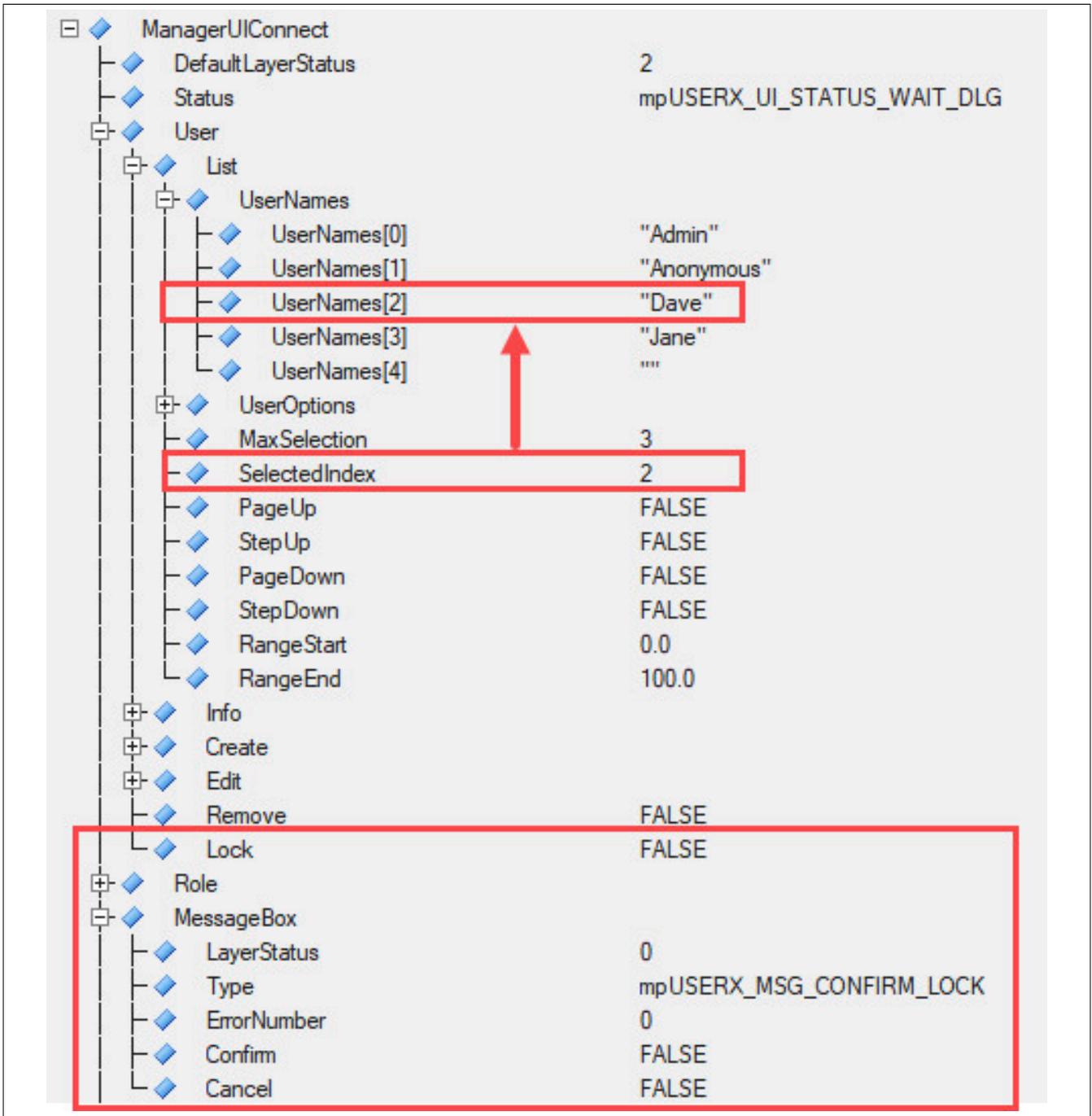
Durch "Confirm = TRUE" bzw. "Cancel = TRUE" wird die Erstellung der Rolle bestätigt bzw. verworfen.

Sperren eines Benutzers

Das Recht einen Benutzer zu sperren bzw. zu entsperren hat nur der Administrator. Wichtig hierbei ist, dass das Benutzerlevel des Administrators höher sein muss als die der anderen Benutzer.



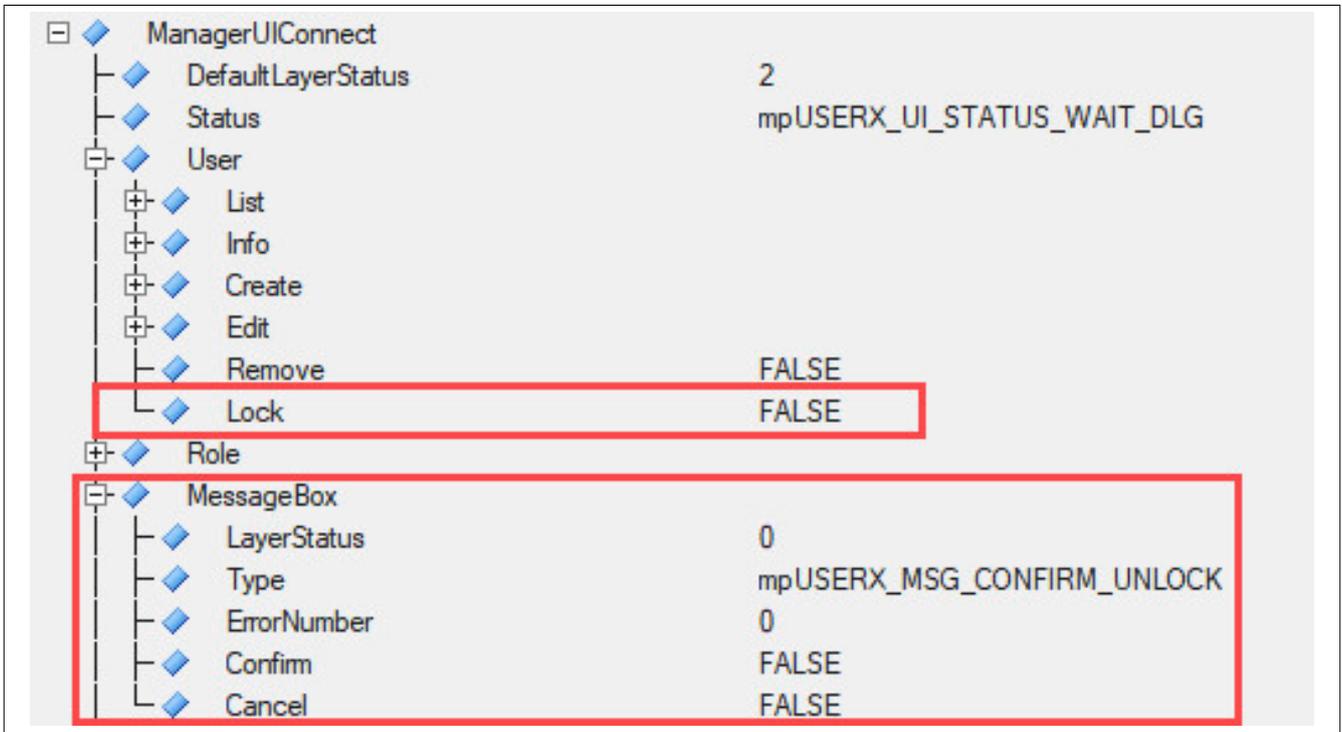
Ein Administrator kann einen Benutzer über die "UIConnect" Prozessvariable sperren. Dazu wird das Kommando "Lock = TRUE" verwendet. Welcher Benutzer gesperrt werden soll, hängt von "SelectedIndex" ab. Die Sperrung kann über "MessageBox" bestätigt ("Confirm = TRUE") oder abgebrochen ("Cancel = TRUE") werden.



Ob ein Benutzer gesperrt ist, wird über "Locked" angezeigt.

ManagerUIConnect	
DefaultLayerStatus	0
Status	mpUSERX_UI_STATUS_IDLE
User	
List	
UserNames	
UserNames[0]	"Admin"
UserNames[1]	"Anonymous"
UserNames[2]	"Dave"
UserNames[3]	"Jane"
UserNames[4]	""
UserOptions	
MaxSelection	3
SelectedIndex	2
PageUp	FALSE
StepUp	FALSE
PageDown	FALSE
StepDown	FALSE
RangeStart	0.0
RangeEnd	100.0
Info	
UserName	"Dave"
FullName	"Dave"
Roles	
Locked	TRUE
Expired	FALSE
RemainingAttempts	3
Language	"
DisplayUnit	"
Creation	DT#2020-04-30-09:02:42
FirstLogin	DT#1970-01-01-00:00:00
LastLogin	DT#2020-04-30-09:02:42
PasswordExpired	DT#2106-02-07-06:28:15
AdditionalData	

Um den Benutzer wieder zu entsperren, wird "Lock" erneut auf "TRUE" gesetzt. Die Entsperrung kann über "MessageBox" bestätigt ("Confirm = TRUE") oder abgebrochen ("Cancel = TRUE") werden.



Gibt ein Benutzer bzw. Administrator bei einem Einlogg-Vorgang das Passwort zu oft falsch ein, so wird der Benutzer bzw. Administrator automatisch gesperrt. Wie oft das Passwort eingegeben werden kann, wird in der [MpUserX Konfiguration](#) über den Parameter "Login attempts" festgelegt.

Wurde der Administrator automatisch gesperrt, so wird dieser standardmäßig nach einer Stunde wieder entsperrt. Diese Zeit kann auch über die [Konfiguration](#) über den Parameter "Admin unlock time" geändert werden. Wird diese Zeit auf 0 gesetzt, so wird der Administrator nicht automatisch entsperrt. Es empfiehlt sich eine "Admin unlock time" festzulegen. Ein gesperrter Administrator kann nur von einem anderem Administrator entsperrt werden. Wichtig hierbei ist, dass dieser Administrator das gleiche oder ein höheres Benutzerlevel besitzt.

Andere gesperrte Benutzer können nur durch den Administrator wieder entsperrt werden. Dazu wird das Kommando "Lock" an der [MpUserXMgrUIUserType](#) Struktur verwendet. Über die Dialogbox-Struktur von [MpUserXMgrUIConnectType](#) ("MessageBox") kann der Entsperr-Vorgang anschließend bestätigt ("Confirm = TRUE") bzw. abgebrochen ("Cancel= TRUE") werden.

Ein Benutzer wird auch gesperrt, wenn dessen Account über längere Zeit nicht benutzt wurde.

Ist ein Benutzer gesperrt, so kann nur der Administrator diesen Benutzer entsperren. Diese Zeit wird in der Konfiguration über den Parameter "User expiration time" definiert. Bei 0 wird diese Funktion nicht genutzt. Der Account kann immer genutzt werden.

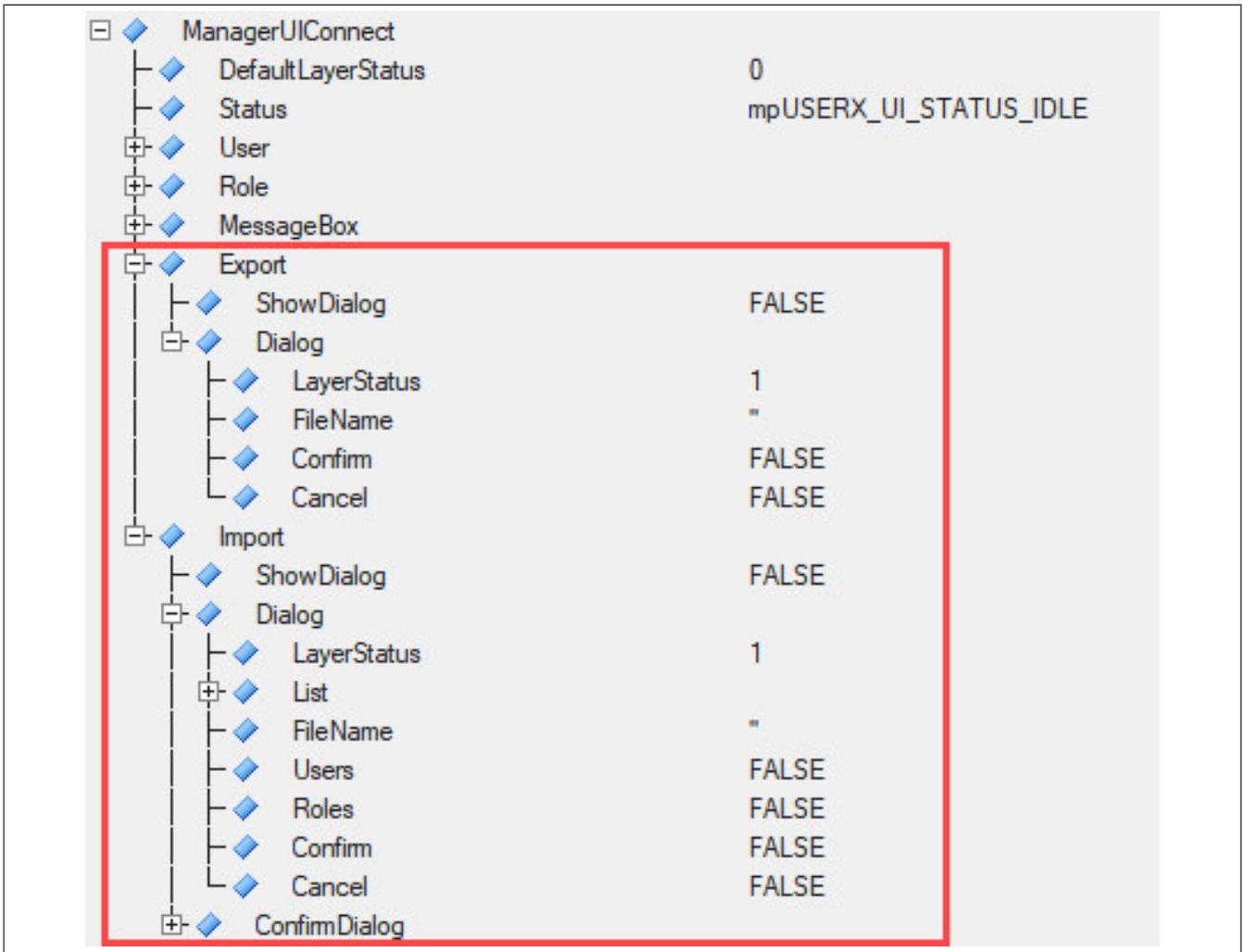
Importieren und Exportieren von Benutzern und Rollen

Mithilfe der "UIConnect" Prozessvariable können Rollen und Benutzer exportiert bzw. importiert werden. Wichtig hierbei ist, dass das verwendete Speichermedium definiert wird, auf dem die Benutzer und rollen exportiert bzw. importiert werden können. Dies wird über den "UISetup" Eingangsparameter an [MpUserXManagerUI](#) festgelegt. Standardmäßig ist das Speichermedium "HD" angegeben.

 Initialize MpUserXManagerUI_0

Name	Type
[-] MpUserXManagerUI_0	MpUserXManagerUI
[-] [+&+] MpLink	MpComIdent Type
[-] Enable	BOOL
[-] ErrorReset	BOOL
[-] [-] UISetup	MpUserXMgrUISetup Type
[-] UserListSize	UINT:=20
[-] ScrollWindow	USINT:=1
[-] FileDevice	STRING[20]:='HD'
[-] FileExtension	STRING[20]:='usr'
[-] RoleListSize	UINT:=9
[-] Confirmation	MpUserXMgrUISetupConfirm Type
[-] [+&+] UIConnect	MpUserXMgrUIConnect Type
[-] Active	BOOL
[-] Error	BOOL
[-] StatusID	DINT
[-] [-] Info	MpUserXInfo Type
[-] [-] Internal	MpUserXInternal Type

Für den Export/Import werden die Parameter unter "Export" bzw. "Import" verwendet. Die Struktur ist so ausgelegt, dass sie beispielsweise mit einem Dialog verbunden werden kann.



Durch den Parameter "ShowDialog" wird der Exportier-, bzw. Importier-Vorgang gestartet.

Wie die Struktur verwendet wird um Benutzer und Rollen zu exportieren wird [hier](#) erklärt. Wie ein Import durchgeführt werden kann, wird [hier](#) erklärt.

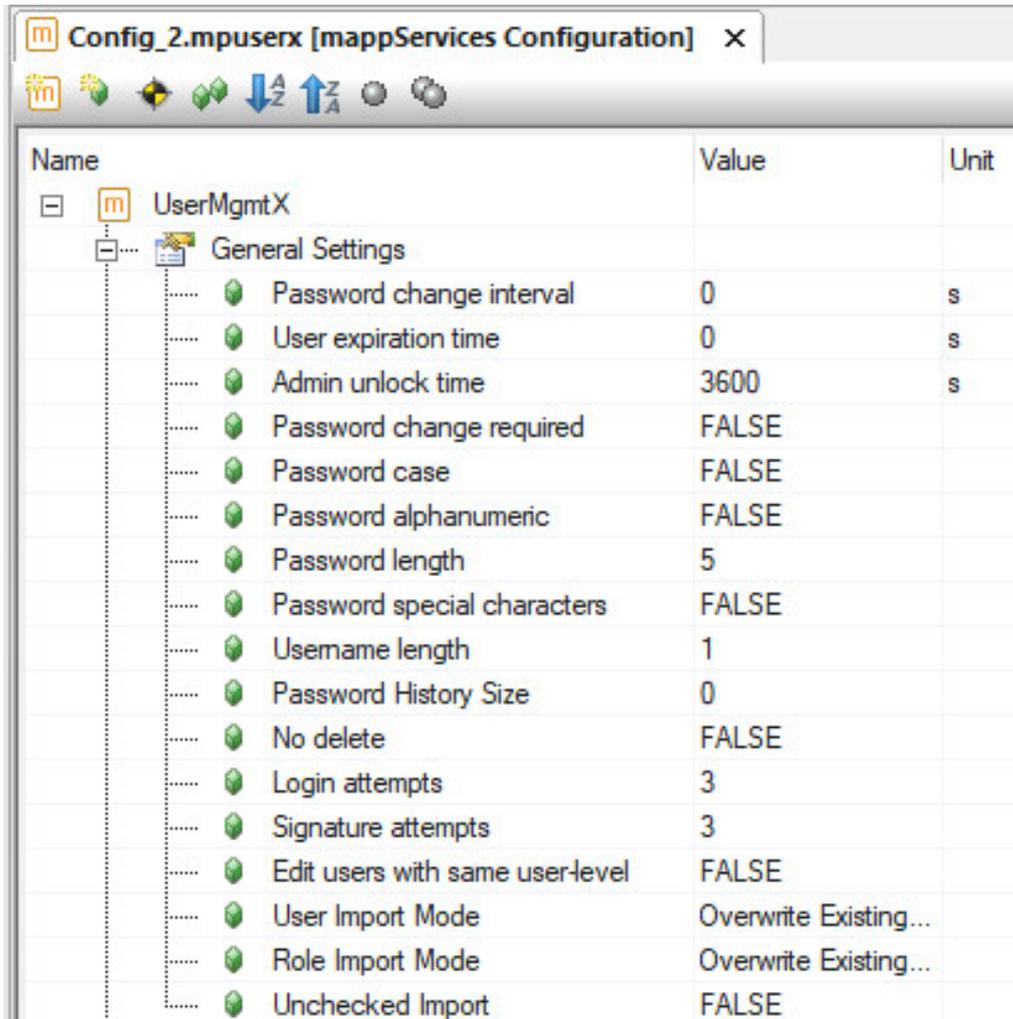
Durch "Confirm = TRUE" werden die Rollen und Benutzer importiert bzw. exportiert. Durch "Cancel = TRUE" wird der Vorgang abgebrochen.

Wie externe Benutzer importiert bzw. exportiert werden können ist [hier](#) erklärt.

Festlegen einer Passwort-Richtlinie

Eine Passwort-Richtlinie legt fest, wie ein Benutzer sein Passwort definieren muss. Es kann zum Beispiel festgelegt werden, dass mindestens 2 Zahlen und ein Großbuchstabe im Passwort vorkommen muss.

Die Passwort-Richtlinie wird in der [MpUserX Konfiguration](#) unter "General Settings" definiert. Standardmäßig werden bereits Parameter gesetzt, wie beispielsweise eine minimale Passwortlänge von 5 Zeichen:



Name	Value	Unit
UserMgmtX		
General Settings		
Password change interval	0	s
User expiration time	0	s
Admin unlock time	3600	s
Password change required	FALSE	
Password case	FALSE	
Password alphanumeric	FALSE	
Password length	5	
Password special characters	FALSE	
Username length	1	
Password History Size	0	
No delete	FALSE	
Login attempts	3	
Signature attempts	3	
Edit users with same user-level	FALSE	
User Import Mode	Overwrite Existing...	
Role Import Mode	Overwrite Existing...	
Unchecked Import	FALSE	

Es empfiehlt sich aus Sicherheitsgründen eine starke Passwort-Richtlinie festzulegen, wie beispielsweise das Verwenden von Groß-, und Kleinbuchstaben ("Password case") oder das Verwenden von Sonderzeichen ("Password special characters").

1.3.2.1 Importieren von extern erstellten Benutzern

In mapp UserX können extern erstellte Benutzer importiert werden.

Folgende Punkte müssen beachtet werden.

Anpassen der Import Datei

Eine zu importierende Datei kann zum Beispiel folgendermaßen aussehen:

```
<?xml version="1.0" encoding="UTF-8"?>
<DATA>
  <Format Type="MpUserX" Version="1.00" Check="1397041096" />
  <Roles>
    <Role ID="Administrators" Check="625992050">
      <URS>
        <Property ID="Description" Value="Role for.. " />
        <Property ID="RoleID" Value="1" />
      </URS>
      <mapp Level="10" Admin="true" />
    </Role>
    <Role ID="Everyone" Check="2822523930">
      <URS>
        <Property ID="Description" Value="Role for.. " />
        <Property ID="RoleID" Value="2" />
      </URS>
    </Role>
    <Role ID="Operator" Check="522675515">
      <URS>
        <Property ID="Description" Value="Role for.. " />
      </URS>
    </Role>
  </Roles>
</DATA>
```

```

        <Property ID="RoleID" Value="3" />
    </URS>
    <mapp Level="5" Admin="false" />
</Role>
</Roles>
<Users>
    <User ID="Admin" Check="2668014190">
        <URS>
            <Property ID="Password" Value="5XQCVvs6f5U05TRPzN31aa88oWbh6jPIOMf0jZ+WqJ0d1pEtEnA=" />
            <Property ID="SystemElementID" Value="Role" />
            <Property ID="UserID" Value="2" />
        </URS>
        <Roles>
            <Role ID="Administrators" />
        </Roles>
        <mapp Name="" Unit="" Language="" />
    </User>
    <User ID="Anonymous" Check="2211814008">
        <URS>
            <Property ID="Password" Value="" />
            <Property ID="SystemElementID" Value="Role" />
            <Property ID="UserID" Value="1" />
        </URS>
        <Roles>
            <Role ID="Everyone" />
        </Roles>
    </User>
    <User ID="test" Check="1205742122">
        <URS>
            <Property ID="Password" Value="Xuql2LeDtyaK66UJVGcoe5VQ6geGiMzmXk81S6Tnj18q3MFV31E=" />
            <Property ID="SystemElementID" Value="Role" />
            <Property ID="UserID" Value="3" />
        </URS>
        <Roles>
            <Role ID="Operator" />
        </Roles>
        <mapp Name="test" Unit="" Language="" />
    </User>
</Users>
</DATA>

```

Soll ein externer Benutzer (ExternUser) hinzugefügt werden, so muss die Importdatei erweitert werden. Dabei muss das <User> Tag um den Eintrag *Ext="1"* erweitert werden. Dadurch wird ein externer Benutzer gekennzeichnet.

```

<?xml version="1.0" encoding="UTF-8"?>
<DATA>
    <Format Type="MpUserX" Version="1.00" Check="1397041096" />
    <Roles>
        <Role ID="Administrators" Check="625992050">
            <URS>
                <Property ID="Description" Value="Role for.. " />
                <Property ID="RoleID" Value="1" />
            </URS>
            <mapp Level="10" Admin="true" />
        </Role>
        <Role ID="Everyone" Check="2822523930">
            <URS>
                <Property ID="Description" Value="Role for.. " />
                <Property ID="RoleID" Value="2" />
            </URS>
        </Role>
        <Role ID="Operator" Check="522675515">
            <URS>
                <Property ID="Description" Value="Role for.. " />
                <Property ID="RoleID" Value="3" />
            </URS>
            <mapp Level="5" Admin="false" />
        </Role>
    </Roles>
    <Users>
        <User ID="Admin" Check="2668014190">

```

```

<URS>
  <Property ID="Password" Value="5XQCVvs6f5UO5TRPzN31aa88oWbh6jPIOMfOjZ+WqJOd1pEtEnA=" />
  <Property ID="SystemElementID" Value="Role" />
  <Property ID="UserID" Value="2" />
</URS>
<Roles>
  <Role ID="Administrators" />
</Roles>
<mapp Name="" Unit="" Language="" />
</User>
<User ID="Anonymous" Check="2211814008">
  <URS>
    <Property ID="Password" Value="" />
    <Property ID="SystemElementID" Value="Role" />
    <Property ID="UserID" Value="1" />
  </URS>
  <Roles>
    <Role ID="Everyone" />
  </Roles>
</User>
<User ID="test" Check="1205742122">
  <URS>
    <Property ID="Password" Value="Xuql2LeDtyaK66UJVGcoe5VQ6geGiMzmXk81S6Tnj18q3MFV31E=" />
    <Property ID="SystemElementID" Value="Role" />
    <Property ID="UserID" Value="3" />
  </URS>
  <Roles>
    <Role ID="Operator" />
  </Roles>
  <mapp Name="test" Unit="" Language="" />
</User>
<User ID="ExternUser" Check="668610876" Ext="1" Pwd="Test99">
  <Roles>
    <Role ID="Operator" />
  </Roles>
  <mapp Name="External User" Unit="iso" Language="en">
    <AdditionalData Key="PhoneNumber" Value="123" />
    <AdditionalData Key="eMail" Value="a.b@c.de" />
    <AdditionalData Key="custom" Value="wasd" />
  </mapp>
</User>
</Users>
</DATA>

```

Anpassen der Konfiguration

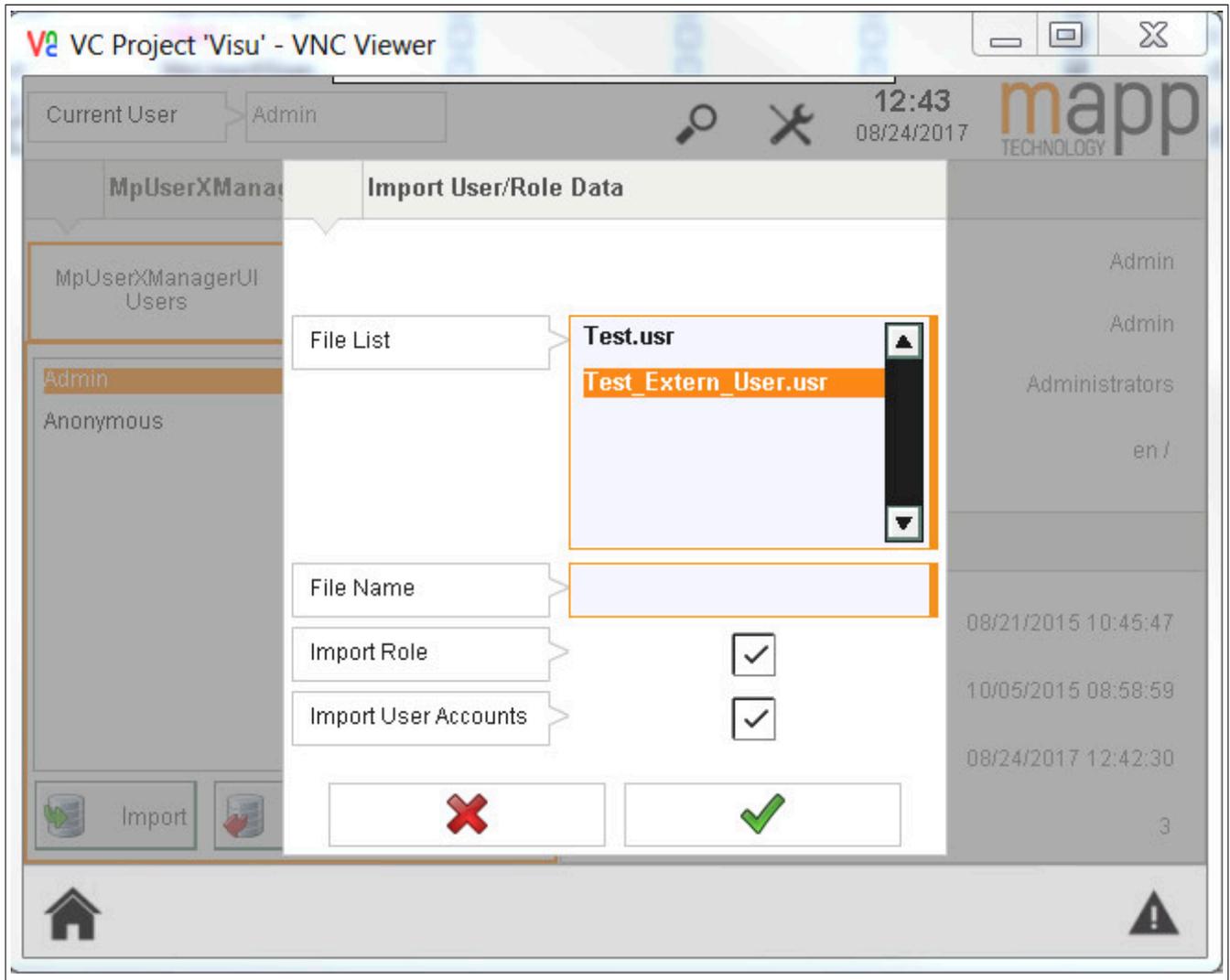
In der [MpUserX Konfiguration](#) muss die Option "UncheckedImport=TRUE" sein. Die Importdatei ist durch eine Prüfsumme geschützt. Ist die Option deaktiviert und die Importdatei wurde um einen Benutzer erweitert, kann die Datei nicht importiert werden.

Wird die Importdatei nicht kontrolliert("UncheckedImport=TRUE"), wird die Prüfsumme nicht beachtet und ein extern erstellter Benutzer kann importiert werden.

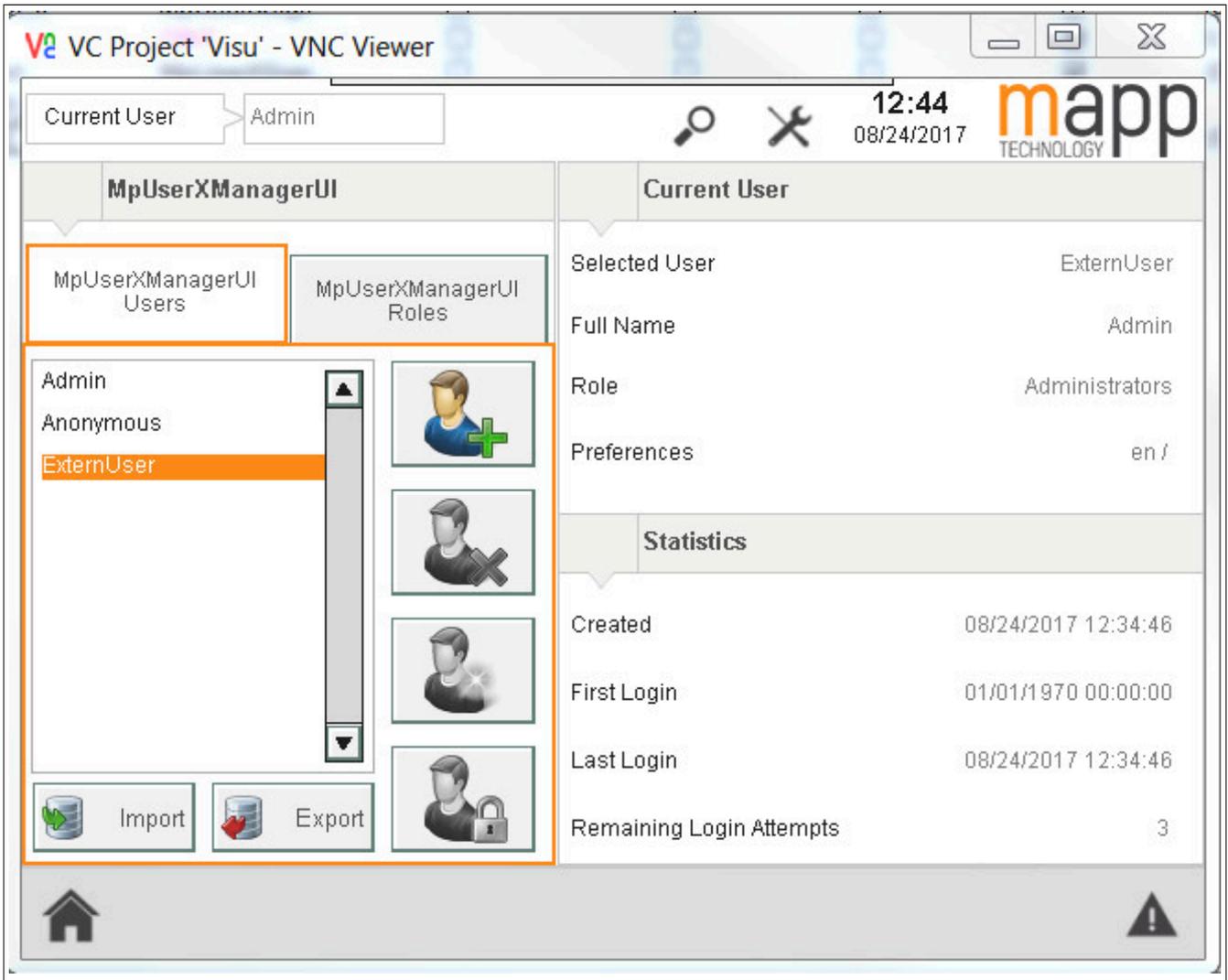
Importieren der Datei

Die Importdatei kann mithilfe von [MpUserXManagerUI](#) über die Visualisierung importiert werden. Es stehen mehrere verschiedene Import-Modi zur Auswahl, welche in der [MpUserX Konfiguration](#) festgelegt werden können.

Die gewünschte Datei wird ausgewählt und importiert:

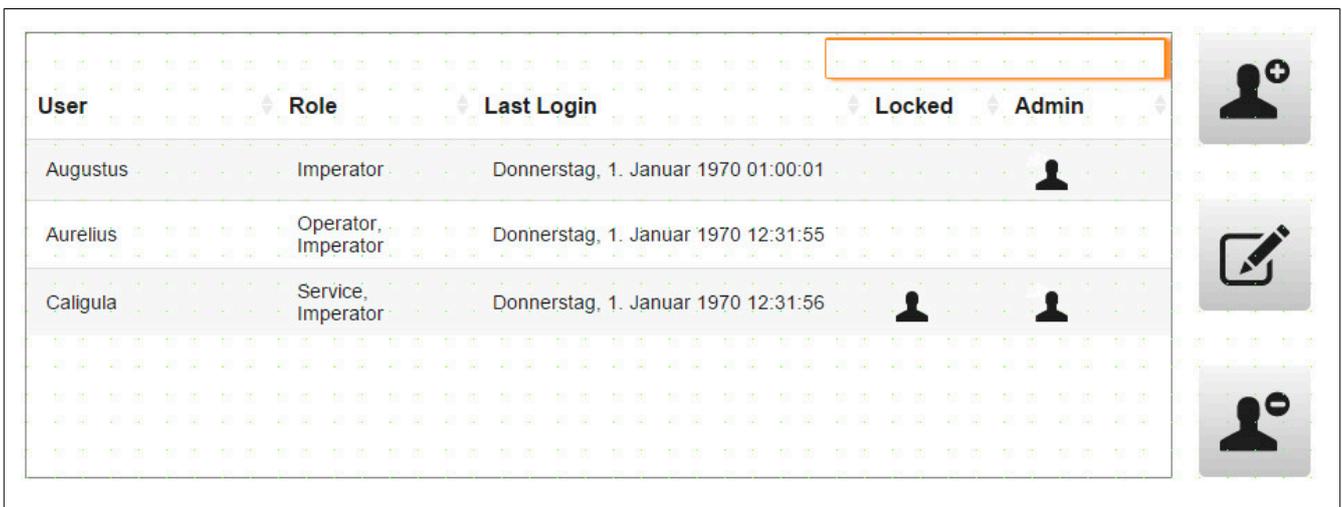


Der extern angelegte Benutzer wird automatisch angelegt.



1.3.2.2 Anbindung an mapp View

Werden die User management Funktionen über eine mapp View Visualisierung verwaltet, so wird das User-List-Widget dazu verwendet. Wie das Widget Schritt für Schritt implementiert wird, ist im Abschnitt Beispielhafte Projektierung zu finden.

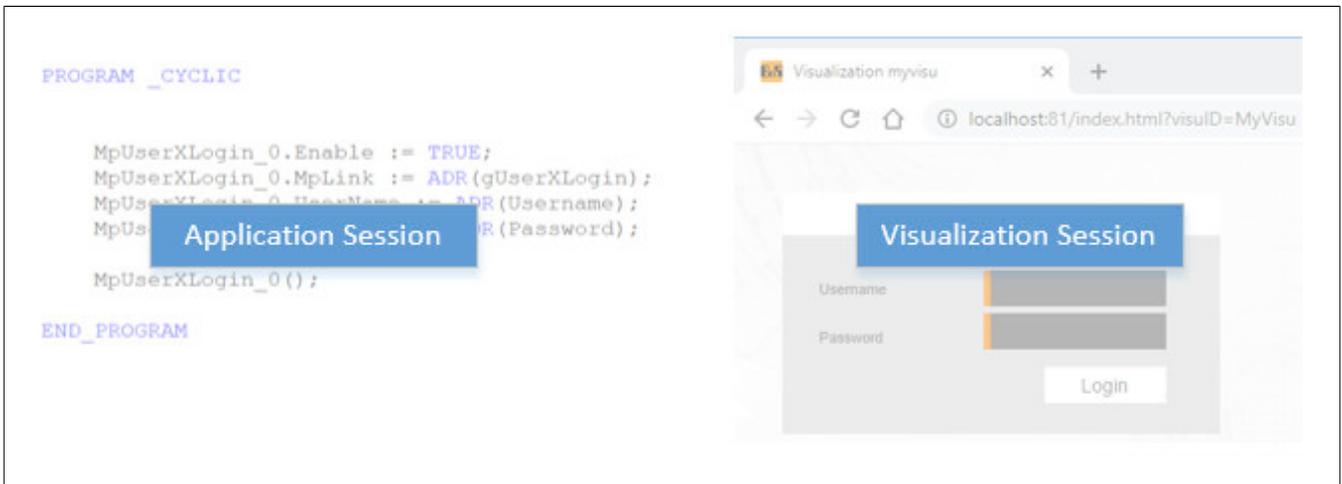


Mithilfe dieses Widgets können Benutzerinformationen angezeigt werden. Zusätzlich kann über verschiedene Aktionen ein Benutzer hinzugefügt, editiert oder gelöscht werden.

1.4 Session Handling

Meldet sich ein Benutzer über mapp UserX im System an, so wird eine Login-Session geöffnet. Die Session kann dabei über die Applikation geöffnet werden oder über eine mapp View Visualisierung. Es wird zwischen einer [Application Session](#) und einer [Visualization Session](#) unterschieden.

Je nachdem welche Session verwendet wird, sind unterschiedliche Vorkehrungen zu treffen. Soll sich ein Benutzer über eine [Application Session](#) anmelden, so müssen Funktionsbausteine implementiert und parametrierbar werden. Bei der Verwendung einer [Visualization Session](#) muss eine mapp View Visualisierung verwendet werden sowie die Login-Widgets eingefügt werden.

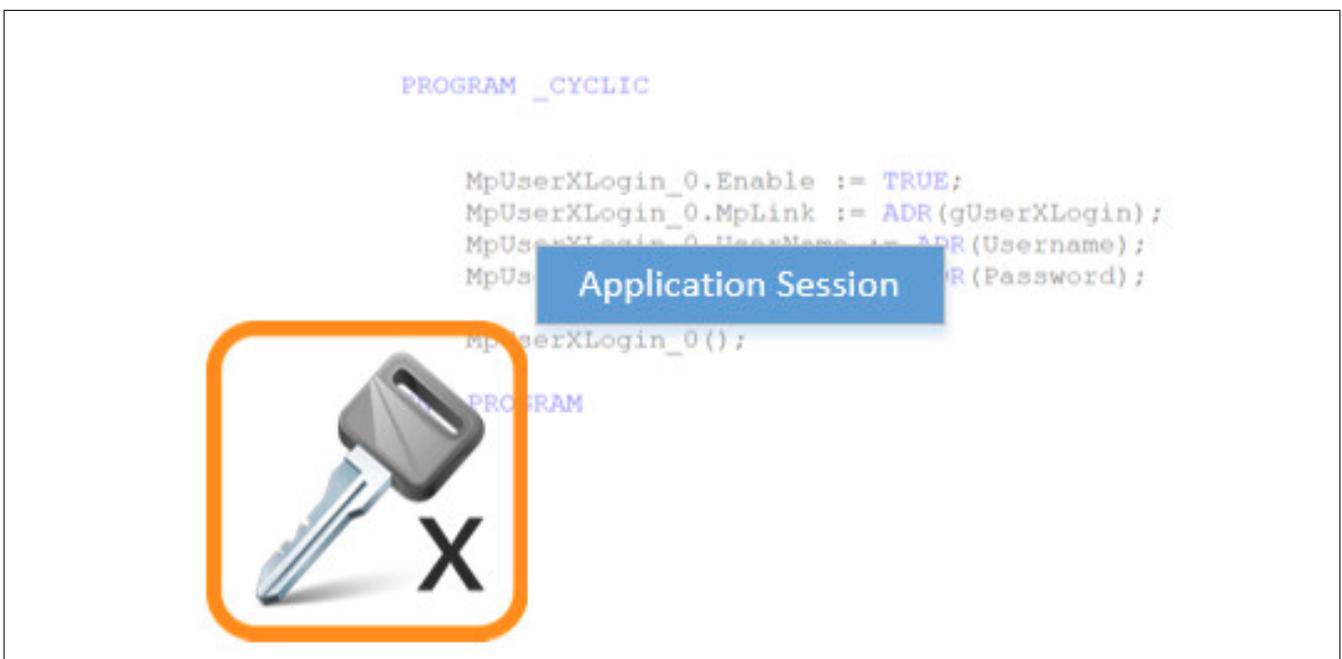


Benutzer-Funktionen

Durch einen Login-Vorgang können verschiedenen Benutzer-Funktionen verwendet werden. Wie die jeweiligen Funktionen in den Sessions angewandt werden, ist im Abschnitt [Application Session](#) bzw. [Visualization Session](#) zu finden.

1.4.1 Application Session

Eine Application Session beschreibt den Login-Vorgang auf applikativem Weg.



Vorkehrungen

Damit sich ein Benutzer über eine Application Session anmelden kann, sind einige Vorkehrungen notwendig. Dazu gehören das Einfügen von Konfigurationen sowie das Einfügen und Parametrieren von Funktionsbausteinen. Für mehr Information siehe [hier](#).

Funktionen

Durch Öffnen der Application Session sind verschiedene Benutzer-Funktionen verfügbar. Welche das sind wird im Abschnitt [Funktionen](#) erklärt.

Verwalten von Berechtigungen

Durch das Öffnen einer Application Session können, je nach angemeldeten Benutzer, verschiedene Berechtigungen freigeschaltet werden. Wie Berechtigungen in der Application Session verwaltet werden, wird [hier](#) erklärt.

mapp UserX in einer modularen Applikation

mapp UserX kann in einer modularen Applikation verwendet werden. Für mehr Informationen siehe Abschnitt [mapp UserX in einer modularen Applikation](#)

1.4.1.1 Vorkehrungen

In diesem Abschnitt wird erklärt, welche Schritte notwendig sind um den Login-Vorgang über eine Application Session abzuwickeln.

Einfügen der Benutzer und Rollen

Im ersten Schritt müssen im Benutzer-Rollen-System die gewünschten Benutzer und Rollen angelegt werden. Standardmäßig existieren die Rollen "Everyone" und "Administrator" und der Benutzer "Anonymous" welchem die Rolle "Everyone" zugeteilt ist.

In diesem Beispiel werden zusätzlich die Rollen "Service" und "Operator" erstellt, sowie die Benutzer "Admin", "Jane" und "Dave":

The screenshot displays three configuration windows in the mapp UserX interface:

- Configuration View:** Shows a tree structure of system components. Under 'AccessAndSecurity', 'UserRoleSystem', and 'Role.role', roles like 'Administrators', 'Everyone', 'Operator', and 'Service' are listed. Under 'User.user', users 'Anonymous', 'Admin', 'Jane', and 'Dave' are listed.
- Role.role [Role Configuration]:** Shows the configuration for the 'Administrators' role. The table below represents the data shown in this window:

Name	Value
Administrators	
Role ID	1
Description	Role for..
Everyone	
Role ID	2
Description	Role for..
Operator	
Role ID	3
Description	Role for..
Service	
Role ID	4
Description	Role for..

- User.user [User Configuration]:** Shows the configuration for the 'Anonymous' user. The table below represents the data shown in this window:

Name	Value
Anonymous	
User ID	1
Password	
Roles	
Assigned Role [1]	Everyone
Assigned Role [2]	
Admin	
User ID	2
Password	*****
Roles	
Assigned Role [1]	Administrators
Assigned Role [2]	
Jane	
User ID	3
Password	*****
Roles	
Assigned Role [1]	Service
Assigned Role [2]	
Dave	
User ID	4
Password	*****
Roles	
Assigned Role [1]	Operator
Assigned Role [2]	

Einfügen der Konfigurationen

Um mapp UserX über eine Application Session verwenden zu können, werden zwei Konfigurationen eingefügt. Die [MpUserLoginX Konfiguration](#) sowie die [MpUserX Konfiguration](#).

MpUserX Konfiguration

Über diese Konfiguration kann mapp UserX vorkonfiguriert werden. Hier kann der [Modus des User managements](#) bestimmt werden sowie Passwort-Richtlinien definiert werden.

Es wird ein lokales User management definiert. Alle vorher angelegten Benutzer und Rollen werden über die Konfiguration angegeben. Die Konfiguration kann nur einmal eingefügt werden, das bedeutet, die hier angelegten Benutzer und Rollen gelten für das gesamte Zielsystem.

Name	Value
UserMgmtX	
User management system settings	
Type	Local
General Settings	
Password change interval	0
User expiration time	0
Admin unlock time	3600
Password change required	FALSE
Password case	FALSE
Password alphanumeric	FALSE
Password length	5
Password special characters	FALSE
Username length	1
Password History Size	0
No delete	FALSE
Login attempts	3
Signature attempts	3
Edit users with same user-level	FALSE
User Import Mode	Overwrite Existing Items
Role Import Mode	Overwrite Existing Items
Unchecked Import	FALSE
Users	
User: Admin	
User: Anonymous	
User: Dave	
User: Jane	
User:	
Roles	
Role: Administrators	
Role: Everyone	
Role: Operator	
Role: Service	

Für die Rollen werden jeweils unterschiedliche Levels definiert. Wichtig hierbei ist, dass für die Administrator-Rolle die Eigenschaft "Administrator = TRUE" festgelegt wird.

Roles		
Role: Administrators		
Name	Administrators	
Level	10	
Administrator	TRUE	
Rights		
Access right 1	Undefined	
Role: Everyone		
Name	Everyone	
Level	0	
Administrator	FALSE	
Rights		
Access right 1	Undefined	
Role: Service		
Name	Service	
Level	8	
Administrator	FALSE	
Rights		
Access right 1	Undefined	
Role: Operator		
Name	Operator	
Level	5	
Administrator	FALSE	

MpUserXLogin Konfiguration

Die **MpUserLoginX Konfiguration** wird für das Öffnen einer Application Session benötigt. Die Konfiguration wird für einen Login-Punkt am Zielsystem verwendet. Das bedeutet, es können mehrere Application Sessions am Zielsystem geöffnet bzw. geschlossen werden. Pro Application Session wird eine **MpUserLoginX Konfiguration** benötigt. Über die Konfiguration kann eine automatische Logout-Zeit festgelegt werden. Dieser Parameter wird für die Funktion "Automatisches Ausloggen bei Inaktivität" verwendet.

Name	Value	Unit
gUserXLogin		
Login		
Auto-logout time	0	s
Alarms	MpAlarm	

Funktionsbaustein einfügen und parametrieren

Um sich über die Applikation anzumelden, wird der Funktionsbaustein **MpUserXLogin** verwendet.

MpUserXLogin

Um sich über die Applikation anzumelden, wird der Funktionsbaustein [MpUserXLogin](#) benötigt. Zusätzlich werden zwei weitere Prozessvariablen eingefügt, welche für den Benutzernamen sowie für das Passwort verwendet werden. Der Funktionsbaustein kann dabei folgendermaßen parametrisiert werden:

```

VAR
    MpUserXLogin_0 : MpUserXLogin := (0);
    Password : WSTRING[80] := "";
    Username: WSTRING[80] := "";

END_VAR

PROGRAM _INIT

    MpUserXLogin_0.Enable := TRUE;
    MpUserXLogin_0.MpLink := ADR(gUserXLogin);
    MpUserXLogin_0.UserName := ADR(Username);
    MpUserXLogin_0.Password := ADR>Password);

END_PROGRAM

PROGRAM _CYCLIC

    MpUserXLogin_0();

END_PROGRAM

PROGRAM _EXIT

    MpUserXLogin_0.Enable := FALSE;
    MpUserXLogin_0();

END_PROGRAM

```

Wie im oberen Abschnitt "MpUserXLogin Konfiguration" erklärt wurde, wird pro Application Session (Login-Punkt am Zielsystem) eine Konfiguration benötigt, das bedeutet, pro Application Session wird ein MpUserXLogin Funktionsbaustein verwendet.

Alle Vorkehrungen sind nun abgeschlossen. Wie sich ein Benutzer nun im System anmelden kann oder beispielsweise das Passwort ändern kann, ist im Abschnitt [Funktionen](#) erklärt.

1.4.1.2 Funktionen

Durch Verwenden der Application Session kann der Benutzer verschiedene Funktionen verwenden. Folgende Funktionen können verwendet werden:

1.4.1.2.1 Login und Logout im System

Wurden alle Schritte welche im Abschnitt [Vorkehrungen](#) aufgelistet sind ausgeführt, so kann ein Login sowie Logout im System mithilfe des Funktionsbausteins [MpUserXLogin](#) durchgeführt werden.

Login

Über die Variablen "Username" und "Password" wird der gewünschte Benutzername sowie das dazugehörige Passwort angegeben. Durch das Login-Kommando am Funktionsbaustein wird der Login-Vorgang gestartet. Wurde der Login-Vorgang erfolgreich abgeschlossen, so wird dies an den Ausgangsparametern angezeigt. Die Application Session wurde erfolgreich geöffnet.

Watch [UserMgmt::Main.st]	
Name	Value
MpUserXLogin_0	
MpLink	94072840
Enable	TRUE
ErrorReset	FALSE
LifeSign	0
Login	TRUE
Logout	FALSE
UserName	97751950
Password	97751788
Active	TRUE
Error	FALSE
StatusID	0
CommandBusy	FALSE
CommandDone	TRUE
CurrentUser	"Jane"
CurrentLevel	8
AccessRights	
Info	
AutoLogoutRemain	T#0ms
Diag	
Internal	
pObject	108139352
pInstance	0
State	3
Username	"Jane"
Password	"jane"

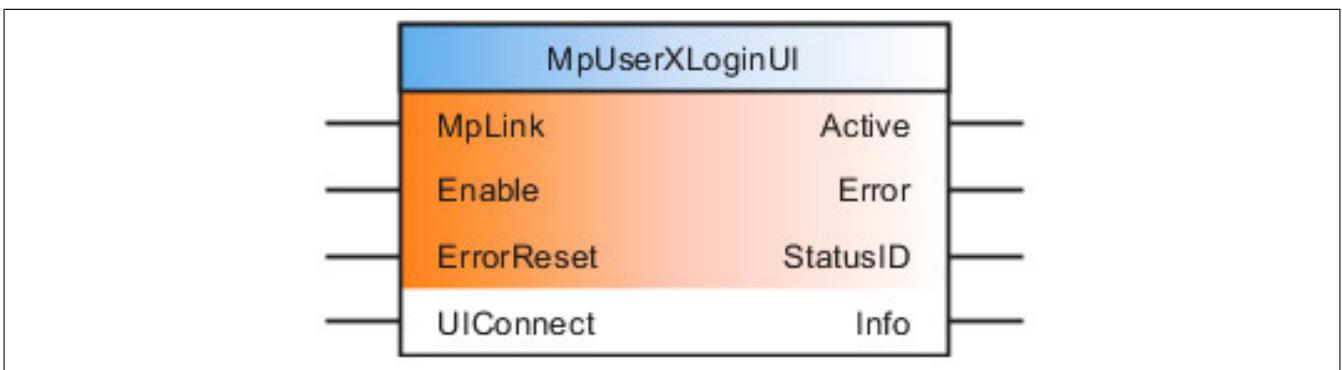
Logout

Um die Session wieder zu schließen, wird das Logout-Kommando am Funktionsbaustein verwendet.

Watch [UserMgmt::Main.st]	
Name	Value
[-] MpUserXLogin_0	
MpLink	94072840
Enable	TRUE
ErrorReset	FALSE
LifeSign	0
Login	FALSE
Logout	TRUE
UserName	97751950
Password	97751788
Active	TRUE
Error	FALSE
StatusID	0
CommandBusy	FALSE
CommandDone	TRUE
CurrentUser	""
CurrentLevel	0
AccessRights	
Info	
AutoLogoutRemain	T#0ms
Diag	
Internal	
pObject	108139352
pInstance	0
State	3
Username	"Jane"
Password	"jane"

MpUserXLoginUI

Eine weitere Möglichkeit eine Application Session zu öffnen bietet der Funktionsbaustein `MpUserXLoginUI`. Dieser wird in Kombination mit `MpUserXLogin` verwendet. Der Funktionsbaustein `MpUserXLoginUI` kann verwendet werden, um die Application Session mit einer Visualisierung zu verbinden. Dafür wird der `UIConnect`-Parameter des `MpUserXLoginUI` Funktionsbausteins verwendet.



An "UIConnect" wird eine Variable angehängt, welche vom Datentypen `MpUserXLoginUIConnectType` ist.

Name	Type
[-] LoginUIConnect	MpUserXLoginUIConnect Type
◆ Status	MpUserXUIStatusEnum
◆ CurrentUser	WSTRING[50]
◆ Language	STRING[20]
◆ DisplayUnit	STRING[20]
◆ LoggedIn	BOOL
◆ UserLevel	DINT
+◆ Login	MpUserXLoginUILogin Type
+◆ ChangePassword	MpUserXLoginUIPwd Type
+◆ MessageBox	MpUserXUIMessageBox Type
◆ DefaultLayerStatus	UINT

Der Login-Parameter bietet die Möglichkeit, den gewünschten Benutzernamen sowie das dazugehörige Passwort festzulegen. Durch "Login = TRUE" wird der Login-Vorgang gestartet. War dieser erfolgreich, so wird der Benutzername, Benutzerlevel etc. ebenfalls an "LoginUIConnect" angezeigt.

[-] ◆ LoginUIConnect	
◆ Status	mpUSERX_UI_STATUS_IDLE
◆ CurrentUser	""
◆ Language	"
◆ DisplayUnit	"
◆ LoggedIn	FALSE
◆ UserLevel	-1
+[-] ◆ Login	
◆ Login	FALSE
◆ Logout	FALSE
◆ UserName	"Jane"
◆ Password	"jane"

Das Login-Kommando sowie der Benutzername und Passwort wird nach erfolgreichem Login automatisch zurückgesetzt. Der Parameter "Login" ist dadurch bereit einen neuen Login durchzuführen. Die Application Session wurde nun erfolgreich geöffnet.

[-] ◆ LoginUIConnect	
◆ Status	mpUSERX_UI_STATUS_IDLE
◆ CurrentUser	"Jane"
◆ Language	"
◆ DisplayUnit	"
◆ LoggedIn	TRUE
◆ UserLevel	8
+[-] ◆ Login	
◆ Login	FALSE
◆ Logout	FALSE
◆ UserName	""
◆ Password	""

Um den Benutzer im System abzumelden (Schließen der Application Session), wird das Logout-Kommando auf TRUE gesetzt.

Treten während eines Login-Vorgangs Probleme auf, wie beispielsweise die Eingabe eines falschen Passworts, so kann eine Fehlerbehandlung mithilfe des Parameters "MessageBox" durchgeführt werden. Die dort angezeigte Statusnummer kann in der Automation Help gesucht werden und dort die vorgeschlagenen Problemlösungen durchgeführt werden.

Wie bereits erwähnt, kann die UIConnect-Struktur verwendet werden, um die Application Session mit einer Visualisierung zu verbinden. Dies kann eine mapp View Visualisierung sein oder auch eine 3rd party Visualisierung.

Ändern des Benutzerpassworts

Das Passwort eines Benutzers kann mithilfe des Funktionsbausteins [MpUserXLoginUI](#) geändert werden. Dieser wird in Kombination mit [MpUserXLogin](#) verwendet. Der Funktionsbaustein [MpUserXLoginUI](#) kann verwendet werden, um die Application Session mit einer Visualisierung zu verbinden. Am Eingangsparameter "UIConnect" wird dafür eine Variable angehängt, welche vom Datentypen [MpUserXLoginUIConnectType](#) ist.

 Initialize LoginUIConnect

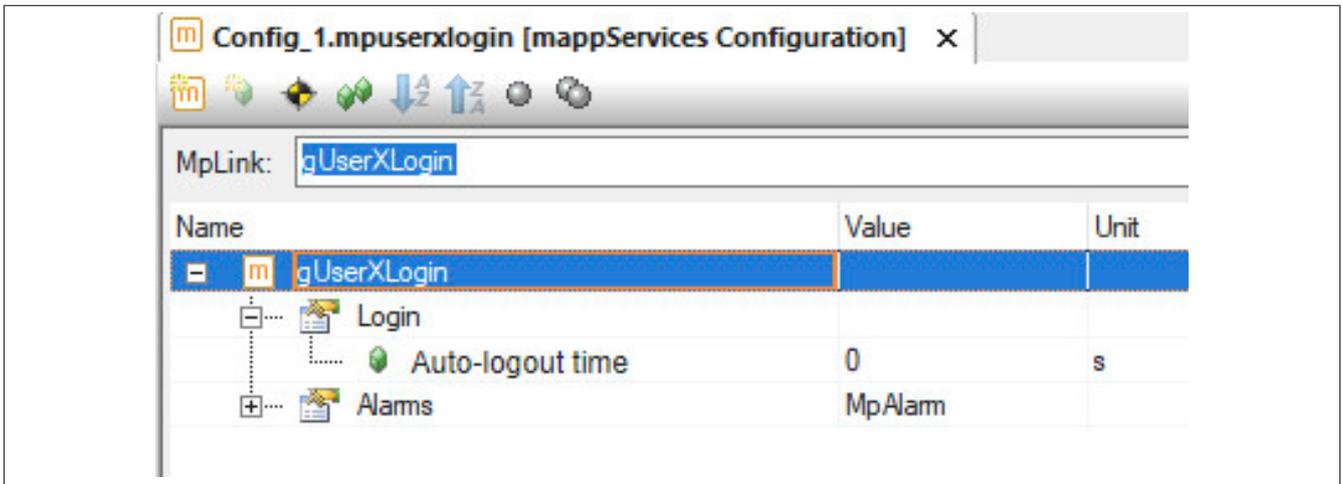
Name	Type	Value
[-] LoginUIConnect	MpUserXLoginUIConnect Type	(0)
◆ Status	MpUserXUIStatusEnum	
◆ Current User	WSTRING[50]	
◆ Language	STRING[20]	
◆ DisplayUnit	STRING[20]	
◆ LoggedIn	BOOL	
◆ UserLevel	DINT	
+◆ Login	MpUserXLoginUILogin Type	
[-] ChangePassword	MpUserXLoginUIPwdType	
◆ ShowDialog	BOOL	
[-] Dialog	MpUserXLoginUIPwdDlgType	
◆ OldPassword	WSTRING[50]	
◆ NewPassword	WSTRING[50]	
◆ ConfirmPassword	WSTRING[50]	
◆ LayerStatus	UINT	
◆ Confirm	BOOL	
◆ Cancel	BOOL	
◆ CriteriaNotMet	MpUserXUIPasswordCriteriaEnum	
◆ NewPasswordOk	UINT	
◆ ConfirmPasswordOk	UINT	
+◆ MessageBox	MpUserXUIMessageBox Type	
◆ DefaultLayerStatus	UINT	

Über "ChangePassword" kann der Benutzer sein Passwort ändern. Dabei muss das alte Passwort sowie das neue Passwort angegeben werden. Über die [MpUserX Konfiguration](#) können die Passwortkriterien definiert werden, wie beispielsweise, dass mindestens 5 Zeichen für das Passwort verwendet werden müssen usw. Ob die Kriterien mit dem neuen Passwort übereinstimmen, wird über den Parameter "CriteriaNotMet" angezeigt. Mehr Information zu den einzelnen Parametern sind in der Struktur [MpUserXLoginUIPwdDlgType](#) erklärt. Bei der Angabe eines Passworts, welches nicht den Kriterien entspricht, wird der Fehler "Passwort nicht sicher genug" am Funktionsbaustein angezeigt.

Stimmen alle Kriterien überein, so kann die Passwort-Änderung durch "Confirm = TRUE" gestartet werden.

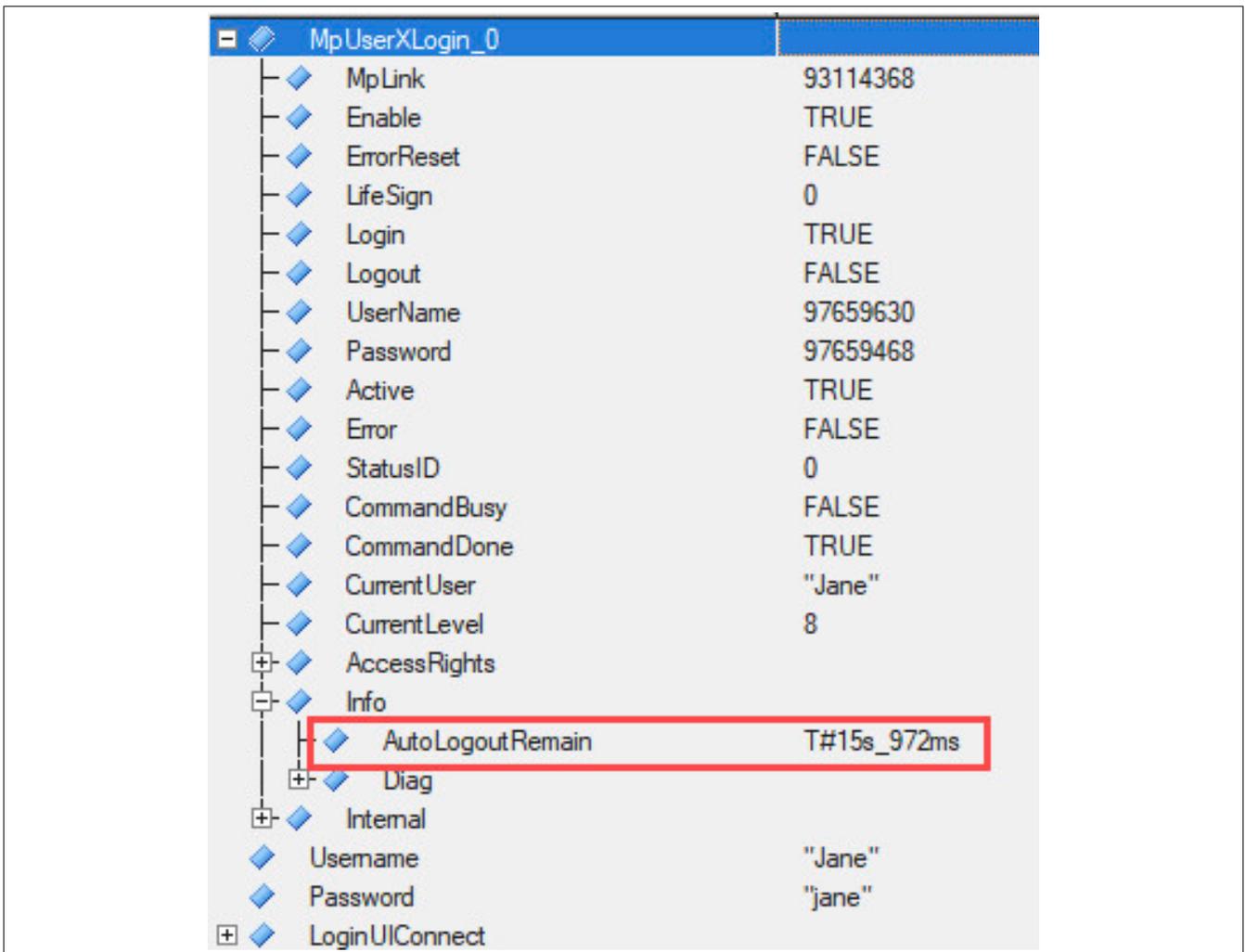
Automatisches Ausloggen bei Inaktivität

In diesem Abschnitt wird erklärt, wie ein angemeldeter Benutzer bei Inaktivität automatisch ausgeloggt wird. Die Auslog-Zeit wird dafür einmal pro Application Session festgelegt und gilt für alle Benutzer. Die Zeit wird in Sekunden in der *MpUserLoginX* Konfiguration unter "Auto-logout time" festgelegt.



Bei einer Angabe von 0 Sekunden ist die Funktion nicht verfügbar, das heißt, es wird nie ein Benutzer bei Inaktivität automatisch ausgeloggt.

Werden beispielsweise 20 Sekunden festgelegt, so wird der Benutzer, sobald dieser im System angemeldet ist, nach 20 Sekunden Inaktivität automatisch ausgeloggt. Die Auslog-Zeit wird dabei am Ausgangsparameter "Auto-LogoutRemain" am Funktionsbaustein *MpUserXLogin* angezeigt.



Die Auslog-Zeit wird wieder auf den Ursprungs-Wert (20 Sekunden) zurückgesetzt, sobald eine positive Flanke am Eingangsparameter "LifeSign" registriert wird. Meldet sich der Benutzer im System an und navigiert oder klickt auf einer Seite, so muss "LifeSign = TRUE" werden.

Benachrichtigung über ablaufendes Passwort

In diesem Abschnitt wird erklärt, wie ein Benutzer über das ablaufende Passwort informiert werden kann.

In der [MpUserX Konfiguration](#) kann über "Password expiration notification" festgelegt werden, wann ein Benutzer über das Ablaufende seines Passwortes benachrichtigt werden soll. Die Eingabe erfolgt in Sekunden. Bei einer Eingabe von beispielsweise 432000 wird der Benutzer 5 Tage bevor sein Passwort abläuft benachrichtigt. Die Benachrichtigung erfolgt dabei über die [Info-Struktur](#) am [MpUserXLogin](#) Funktionsbaustein. Dabei gibt der Parameter "DaysUntilPasswordExpiration" an wie viele Tage bis zum Passwort-Ablauf noch vorhanden sind. Der Benutzer muss innerhalb dieser Zeit das Passwort ändern. Innerhalb dieser definierten Zeit bekommt der Benutzer bei jedem Login-Vorgang die Warnung "Passwort wird ablaufen" am [MpUserXLogin](#) bzw. [MpUserXLoginUI](#) Funktionsbaustein angezeigt. Wurde das Passwort nicht geändert und die Frist ist abgelaufen, so wird bei einem neuen Login-Vorgang die Warnung "Passwortänderung erforderlich" am [MpUserXLogin](#) Funktionsbaustein angezeigt. Wird der Login-Vorgang über [MpUserXLoginUI](#) durchgeführt, so wird der Fehler "Passwortänderung erforderlich" angezeigt.

Ist die Frist zur Passwort-Änderung abgelaufen und der Benutzer ist im System noch angemeldet, so wird der Benutzer nicht ausgeloggt oder verliert Zugriffsrechte. Erst sobald eine neue Anmeldung erfolgt wird die Warnung bzw. Fehlermeldung angezeigt.

Zentral verwaltete Benutzer

Wird als [User management Modus](#) "Zentral" bzw. "Lokal und zentral" ausgewählt, so gilt es, abhängig vom verwendeten Directory Server, bei zentralen Benutzern folgendes zu beachten:

- **Active Directory Server:** Bei Benutzern welche am Active Directory Server verwaltet werden gilt es zu beachten, dass das Ablauf-Datum des Passworts am Server festgelegt wird und nicht in der Konfiguration. Das Ablauf-Datum des Passworts wird nur abgefragt wird wenn eine Verbindung zum Active Directory Server besteht. Dies ist während eines Login-Vorgangs der Fall. Wurde das Passwort innerhalb der Frist nicht geändert und ein neuer Login-Vorgang wird gestartet, so kann sich der zentrale Benutzer nicht mehr an der Maschine anmelden. Es erscheint die Warnung "Passwortänderung erforderlich" am [MpUserXLogin](#) bzw. [MpUserXLoginUI](#) Funktionsbaustein. Die Passwort-Änderung muss nun direkt am Active Directory Server erfolgen. Nachdem das Passwort geändert wurde, kann sich der Benutzer wieder über die Maschine anmelden.
- **389 Directory Server:** Wann für einen Benutzer das Passwort abläuft, wird direkt über den 389 Directory Server bestimmt nicht über die [MpUserX Konfiguration](#). Während eines Login-Vorgangs wird die Information von mapp UserX am 389 Directory Server abgefragt und am "DaysUntilPasswordExpiration" Parameter des [MpUserXLogin](#) Funktionsbausteins angezeigt. Wurde das Passwort innerhalb der Frist nicht geändert und ein neuer Login-Vorgang wird gestartet, so kann sich der zentrale Benutzer nicht mehr an der Maschine anmelden. Es muss eine Passwort-Änderung durchgeführt werden. Dies kann direkt an der Maschine oder über den 389 Directory Server gemacht werden.

Nutzerpräferenzen

In der [MpUserX Konfiguration](#) kann im Abschnitt "Users" für einen Benutzer festgelegt werden, welche Sprache ("Language") und welches Einheitensystem ("Measurement system") bevorzugt verwendet werden soll. Wurde eine Sprache bzw. ein Einheitensystem für einen Benutzer festgelegt und dieser Benutzer meldet sich an, so muss eine Sprach-, bzw. Einheitenumschaltung applikativ durchgeführt werden. Ein Wechsel des Systems erfolgt nicht automatisch. Die gewünschte Sprache bzw. Einheit kann mithilfe des [MpUserXManagerUI](#) Funktionsbausteins gelesen werden. Dazu wird der Eingangsparameter "UIConnect.User.Info.Language" bzw. "UIConnect.User.Info.DisplayUnit" verwendet.

UserMgmtX		
+ ...	User management system settings	
+ ...	General Settings	
- ...	Users	
- ...	User: Jane	
- ...	UserName	Jane
- ...	FullName	Jane Johnson
- ...	Language	en
- ...	Measurement system	imperial
+ ...	Additional Data	

Zur Laufzeit können diese Nutzerpräferenzen mithilfe von [MpUserXConfig](#) geändert werden.

Bei der Angabe ungültiger Daten wie der Angabe eines nicht existierenden Einheitensystems oder der Angabe von keinen Daten werden die Daten trotzdem von mapp UserX gespeichert. mapp UserX überprüft die Daten nicht auf Gültigkeit.

Es gilt zu beachten, dass die Nutzerpräferenzen nur für [lokale Benutzer](#) gespeichert werden kann!

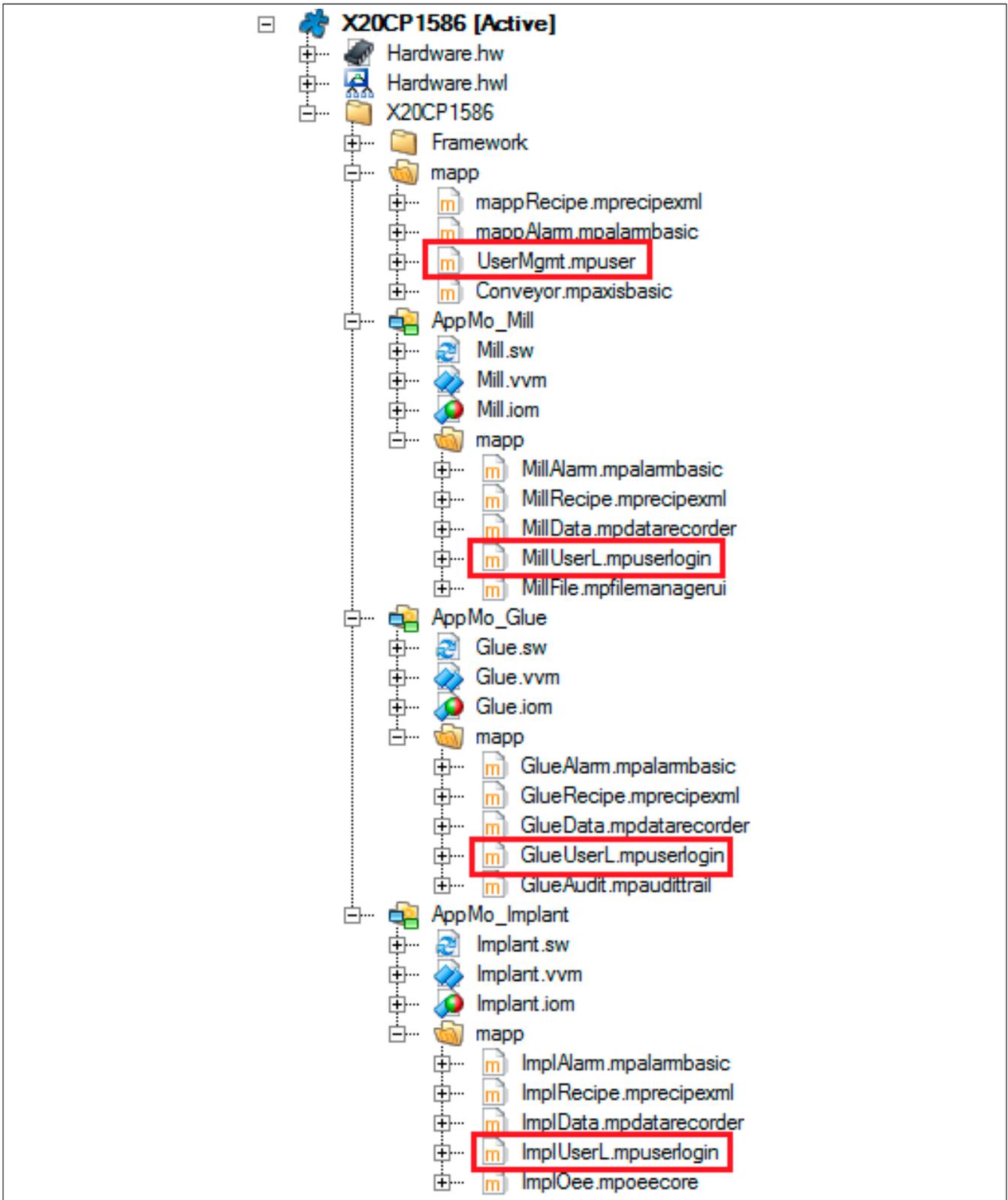
1.4.1.3 mapp UserX in einer modularen Applikation

In diesem Abschnitt wird erklärt was man bei der Implementierung von mapp UserX in modularen Applikationen beachten muss. Allgemeine Infos zu mapp in Applikationsmodulen / einer Maschinengruppe sind Bilden von modularen Applikationen

Verhalten von mapp UserX in einer modularen Applikation

Die [MpUserX Konfiguration](#) gilt für die komplette Maschine und sollte daher nicht einer speziellen Maschinengruppe bzw. einem speziellen Applikationsmodul zugewiesen sein. Bei Verwendung von Applikationsmodulen sollte die [MpUserX Konfiguration](#) im Framework Projekt liegen.

Die [MpUserLoginX Konfiguration](#) kann mehrfach eingefügt werden. Die Konfiguration beschreibt einen Loginpunkt in der Maschine und kann in den verschiedenen Applikationsmodulen / Maschinengruppen eingefügt werden. Über jeden Loginpunkt kann sich ein Benutzer anmelden und somit eine Application Session öffnen. Pro Applikationseinheit / Maschinengruppe ist nur ein Loginpunkt erlaubt.



1.4.1.4 Verwalten von Berechtigungen

Meldet sich ein Benutzer im System an, so werden, abhängig von der Rolle des Benutzers, unterschiedliche Berechtigungen aktiviert. Das bedeutet manche Funktionen im System werden gesperrt bzw. freigeschaltet.

Beispiel:

Meldet sich im System der Benutzer Jane an, welche die Rolle "Service" besitzt, so erhält diese die Berechtigung auf die Service-Seite zu navigieren und dort unterschiedliche Parameter zu regulieren. Meldet sich der Benutzer Dave an, welche die Rolle "Operator" besitzt, so ist die Navigation auf die Service-Seite gesperrt. Es können keine Änderungen durchgeführt werden.

Die Realisierung dieses Beispiels ist dabei auf zwei unterschiedliche Arten möglich: Die Abfrage der definierten Zugriffsrechte oder das kontrollieren des Benutzerlevels.

Es gilt zu beachten, dass Berechtigungen immer an die Rolle und nicht an dem Benutzer geknüpft sind.

Berechtigung über Zugriffsrechte

Berechtigungen können über Zugriffsrechte bestimmt werden. Die Zugriffsrechte werden in der [MpUserX Konfiguration](#) unter "Rights" festgelegt:

Name	Value	Description
UserMgmtX		
User management system settings		General settings to define which user management system is used
Type	Local	Defines the type of user management system
General Settings		Local user management system specific settings
Users		Local users
Roles		
Role: Administrators		
Role: Everyone		
Role: Service		
Name	Service	Role from URS (User / Role System)
Level	8	User level
Administrator	FALSE	Defines whether the role has administrator rights
Rights		Access rights
Access right 1	Full	Navigate to Service Page
Access right 2	Undefined	
Role: Operator		
Name	Operator	Role from URS (User / Role System)
Level	5	User level
Administrator	FALSE	Defines whether the role has administrator rights
Rights		Access rights
Access right 1	None	Navigate to Service Page
Access right 2	Undefined	

Welche Zugriffsrechte dabei existieren sollen, können selbst definiert werden. Für mehr Information dazu siehe Abschnitt [Rechte und Levels](#).

Um das Zugriffsrecht der jeweils angemeldeten Rolle zu kontrollieren, kann der Funktionsbaustein [MpUserXLogin](#) sowie die Funktion [MpUserXAccessRight](#) verwendet werden.

MpUserXLogin

Meldet sich ein Benutzer im System an, so werden die Zugriffsrechte am Funktionsbaustein [MpUserXLogin](#) an "AccessRights" angezeigt. Auf Basis dieser Ausgabeparameter kann die Service-Seite gesperrt oder freigeschalten werden.

MpUserXLogin_0	
MpLink	94382968
Enable	TRUE
ErrorReset	FALSE
LifeSign	0
Login	TRUE
Logout	FALSE
UserName	99033838
Password	99033676
Active	TRUE
Error	FALSE
StatusID	0
CommandBusy	FALSE
CommandDone	TRUE
CurrentUser	"Jane"
CurrentLevel	8
AccessRights	
AccessRights[0]	mpUSERX_ACCESS_FULL
AccessRights[1]	mpUSERX_ACCESS_UNDEFINED
AccessRights[2]	mpUSERX_ACCESS_UNDEFINED

MpUserXAccessRight

Mithilfe der [MpUserXAccessRight](#) Funktion kann überprüft werden, ob ein aktuell angemeldeter Benutzer die Berechtigung auf ein definiertes Zugriffsrecht besitzt. Das bedeutet, das in der Konfiguration definierte Zugriffsrecht "Access Right 1" kann über [MpUserXAccessRight](#) überprüft werden. Dafür wird das zu überprüfende Zugriffsrecht an "Right" angegeben:

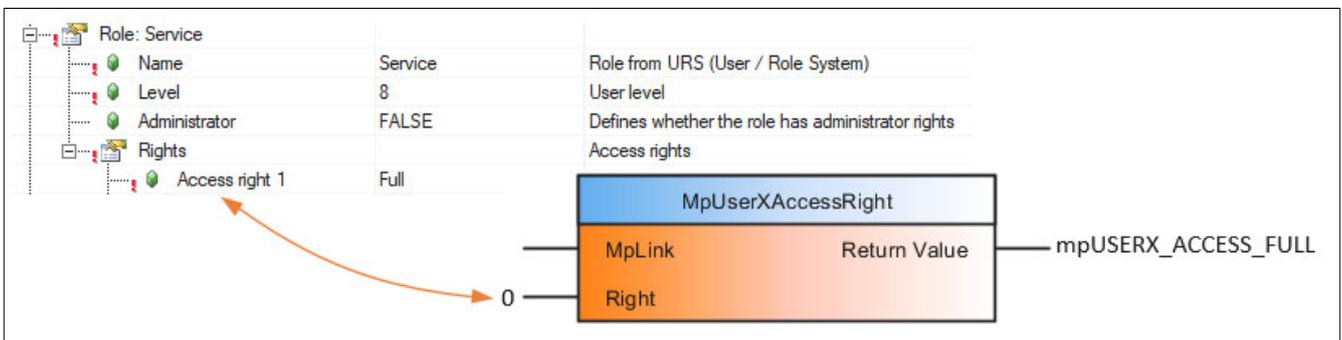


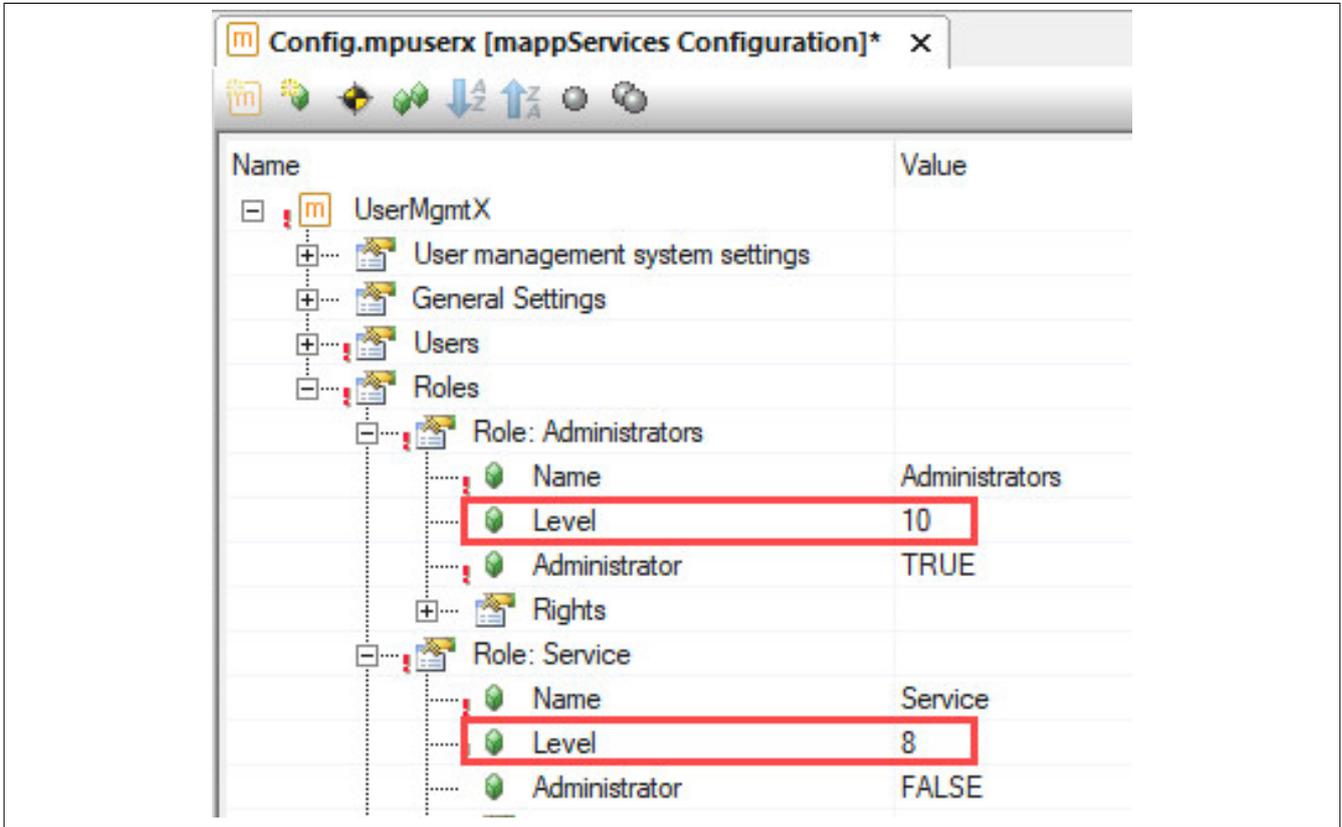
Abbildung 3: Access Right 1 = Right 0

Der Rückgabewert der Funktion gibt an, welche Berechtigung für das Zugriffsrecht existiert. Dieser Rückgabewert kann dazu verwendet werden, die Service-Seite freizuschalten oder zu sperren.

Berechtigung über Benutzerlevel

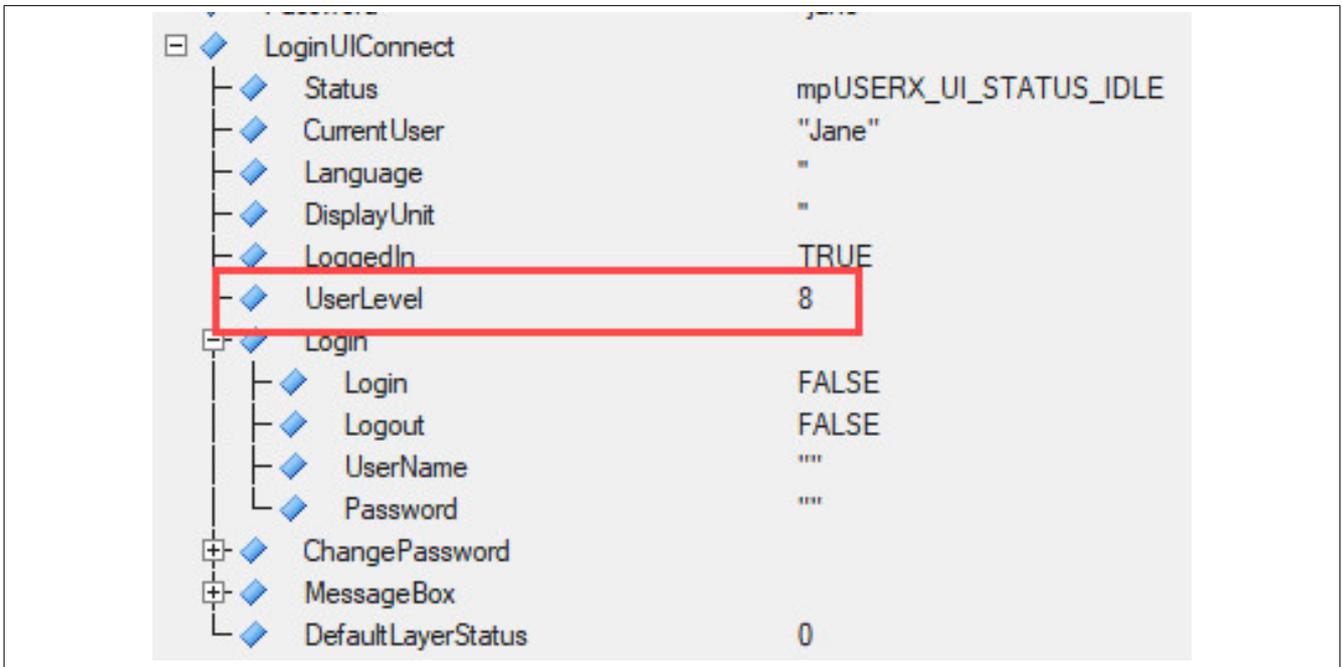
In diesem Abschnitt wird erklärt, wie Berechtigungen über das Benutzerlevel verwaltet werden können.

In der [MpUserX Konfiguration](#) kann für eine Rolle ein Level zugewiesen werden. Es gilt die Regel, je höher das Level ist, desto mehr Berechtigungen hat diese Rolle.



Meldet sich nun ein Benutzer an, so hat dieser Benutzer das Level welches für die ihm zugeordnete Rolle festgelegt wurde. Das Benutzerlevel wird dabei beispielsweise am Ausgangsparameter "CurrentLevel" des Funktionsbausteins [MpUserXLogin](#) angezeigt.

Auch über den UIConnect-Parameter von [MpUserXLoginUI](#) wird das Benutzerlevel angezeigt.

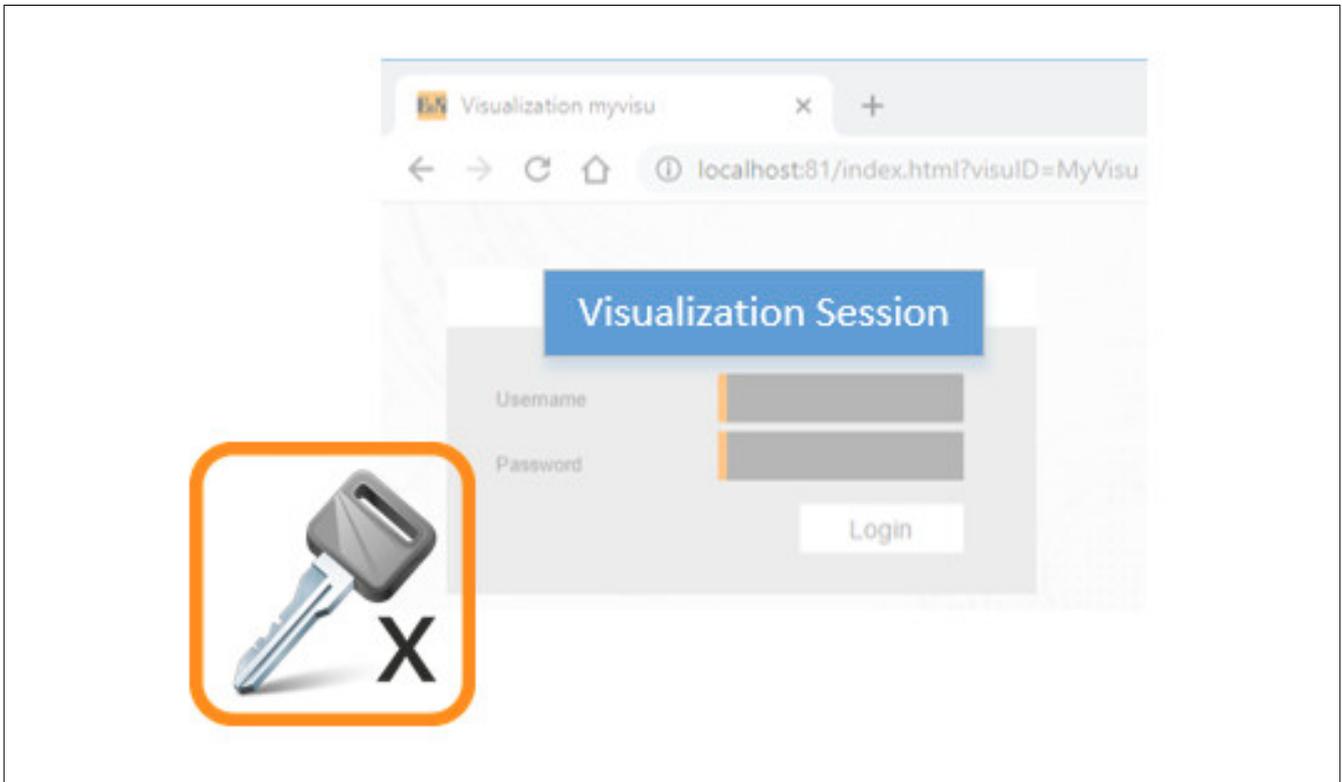


Dieser Werte können verwendet werden, um die Service-Seite freizuschalten oder zu sperren.

Zusätzlich kann die Funktion [MpUserXLevel](#) verwendet werden. Mithilfe dieser Funktion kann direkt das Benutzerlevel des aktuell angemeldeten Benutzers abgefragt werden. Auch dies kann als Basis dafür dienen, die Service-Seite freizuschalten oder zu sperren.

1.4.2 Visualization Session

Eine Visualization Session beschreibt den Login-Vorgang über mapp View.



Vorkehrungen

Damit sich ein Benutzer über eine Visualization Session anmelden kann, sind einige Vorkehrungen notwendig. Dazu gehören das Definieren von Rollen und Benutzern sowie das Einfügen und Konfigurieren von Widgets. Für mehr Information siehe [hier](#).

Funktionen

Durch Öffnen der Visualization Session sind verschiedene Benutzer-Funktionen verfügbar. Welche das sind wird im Abschnitt [Funktionen](#) erklärt.

Verwalten von Berechtigungen

Durch das Öffnen einer Visualization Session können, je nach angemeldeten Benutzer, verschiedene Berechtigungen freigeschaltet werden. Wie Berechtigungen in der Visualization Session verwaltet werden, wird [hier](#) erklärt.

1.4.2.1 Vorkehrungen

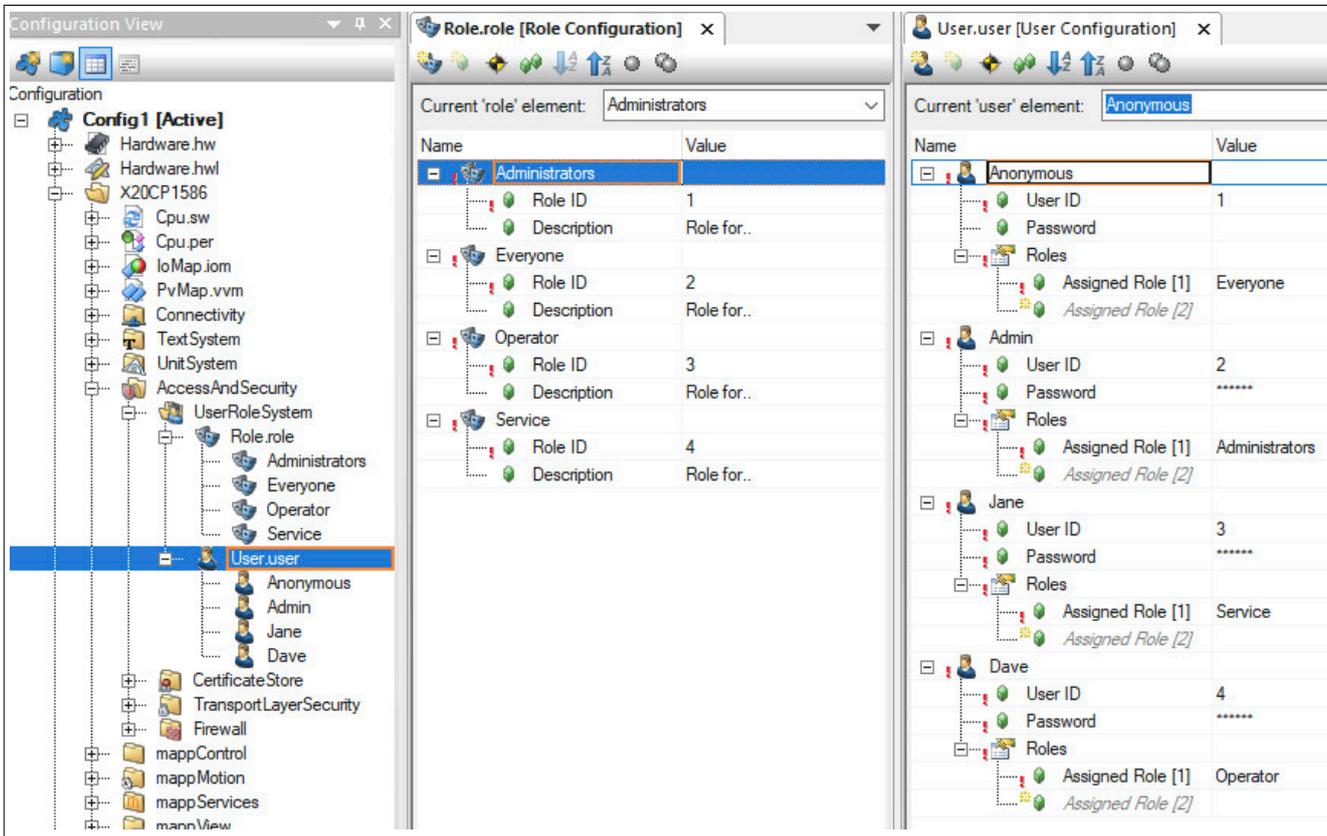
In diesem Abschnitt wird erklärt, welche Schritte notwendig sind um den Login-Vorgang über eine Visualization Session abzuwickeln.

Einfügen der Benutzer und Rollen

Im ersten Schritt müssen im Benutzer-Rollen-System die gewünschten Benutzer und Rollen angelegt werden.

Standardmäßig existieren die Rollen "Everyone" und "Administrator" und der Benutzer "Anonymous" welcher die Rolle "Everyone" zugeteilt hat. Es gilt zu beachten, dass für die Funktion der Benutzer "Anonymous" mit der Rolle "Everyone" benötigt wird. Dieser Benutzer darf nicht verändert werden.

Es werden die Rollen "Service" und "Operator" erstellt, sowie die Benutzer "Admin", "Jane" und "Dave":



Einfügen der Konfigurationen

Um mapp UserX über eine Visualization Session verwenden zu können, wird die **MpUserX Konfiguration** eingefügt.

MpUserX Konfiguration

Über diese Konfiguration kann mapp UserX vorkonfiguriert werden. Hier kann der **Modus des User managements** bestimmt werden sowie beispielsweise Passwort-Richtlinien definiert werden.

Alle vorher angelegten Benutzer und Rollen werden über die Konfiguration angegeben. Die Konfiguration kann nur einmal eingefügt werden, das bedeutet, die hier angelegten Benutzer und Rollen gelten für das gesamte Zielsystem.

Name	Value
UserMgmtX	
User management system settings	
Type	Local
General Settings	
Password change interval	0
User expiration time	0
Admin unlock time	3600
Password change required	FALSE
Password case	FALSE
Password alphanumeric	FALSE
Password length	5
Password special characters	FALSE
Username length	1
Password History Size	0
No delete	FALSE
Login attempts	3
Signature attempts	3
Edit users with same user-level	FALSE
User Import Mode	Overwrite Existing Items
Role Import Mode	Overwrite Existing Items
Unchecked Import	FALSE
Users	
User: Admin	
User: Anonymous	
User: Dave	
User: Jane	
User:	
Roles	
Role: Administrators	
Role: Everyone	
Role: Operator	
Role: Service	

Für die Rollen werden jeweils unterschiedliche Levels definiert. Wichtig hierbei ist, dass für die Administrator-Rolle die Eigenschaft "Administrator = TRUE" festgelegt wird.

Roles		
Role: Administrators		
Name	Administrators	
Level	10	
Administrator	TRUE	
Rights		
Access right 1	Undefined	
Role: Everyone		
Name	Everyone	
Level	0	
Administrator	FALSE	
Rights		
Access right 1	Undefined	
Role: Service		
Name	Service	
Level	8	
Administrator	FALSE	
Rights		
Access right 1	Undefined	
Role: Operator		
Name	Operator	
Level	5	
Administrator	FALSE	

Vorbereiten der Visualisierung

Für eine Visualization Session wird eine mapp View Visualisierung verwendet.



Wie eine mapp View Visualisierung erstellt wird ist hier erklärt.

Es gilt zu beachten, dass in der mapp View Konfiguration als Authentifizierungs-Art "MpUserX" definiert werden muss. Nur so können die Benutzer und Rollen von mapp UserX innerhalb von mapp View verwendet werden.

Name	Value	Unit
MappViewConfiguration		
Server configuration		
Protocol	HTTP	
Port Number	81	
Maximal client connect...	1	
Maximal B&R client co...	0	
Authentication mode	MpUserX	
OPC-UA system		
Server connection time...	MpUserX	ms
Samplina rate groups		

Verwenden von Login-Widgets

Um die Visualization Session zu öffnen können verschiedene Login-Widgets verwendet werden. In diesem Beispiel wird das Login-Widget verwendet. Zusätzlich wird ein LogoutButton-Widget eingefügt.



Alle Vorkehrungen sind nun abgeschlossen. Wie sich ein Benutzer nun im System anmelden kann ist im Abschnitt [Funktionen](#) erklärt.

1.4.2.2 Funktionen

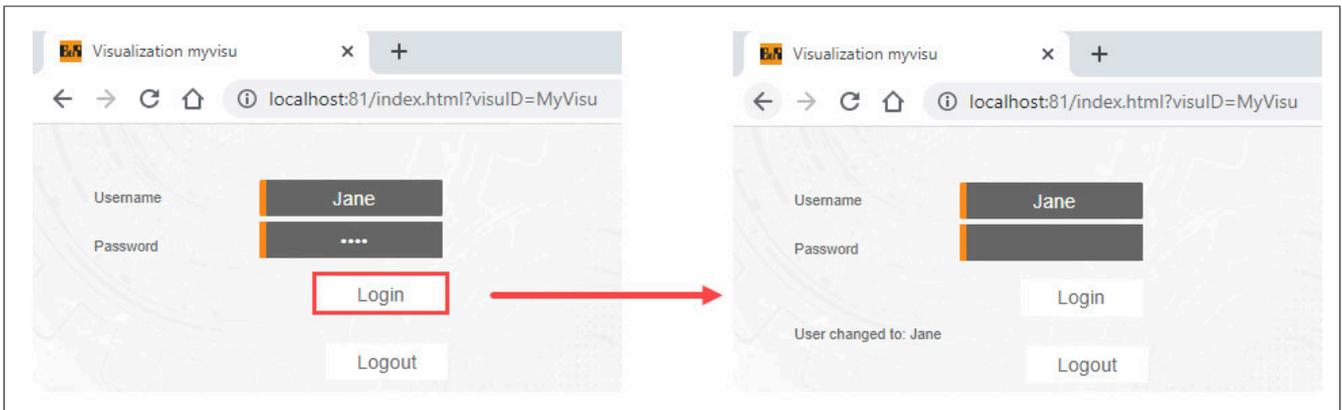
Durch verwenden einer Visualization Session kann folgende Funktion verwendet werden:

Login und Logout im System

Wurden alle Schritte welche im Abschnitt [Vorkehrungen](#) aufgelistet sind ausgeführt, so kann ein Login sowie Logout im System durchgeführt werden.

Login:

Zur Laufzeit wird über den Browser der gewünschte Benutzername sowie das dazugehörige Passwort angegeben. Durch einen Klick auf "Login" meldet sich der Benutzer im System an. Die Visualization Session wurde erfolgreich geöffnet. Der aktuell angemeldete Benutzer wird direkt an der Visualisierung angezeigt.

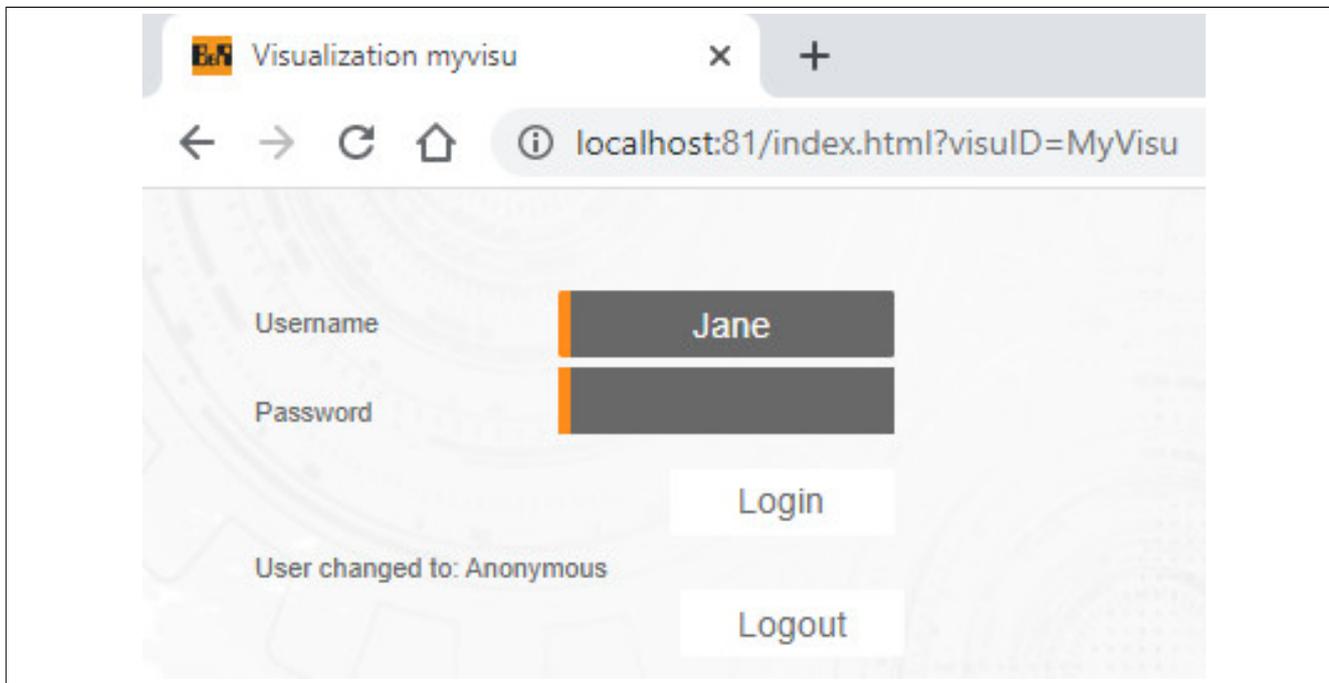


Schlägt der Login-Vorgang fehl, weil beispielsweise ein falsches Passwort verwendet wurde, so wird der Benutzer direkt in der Visualisierung darüber informiert.

Pro verbundenen Client kann eine eigene Visualization Session geöffnet werden.

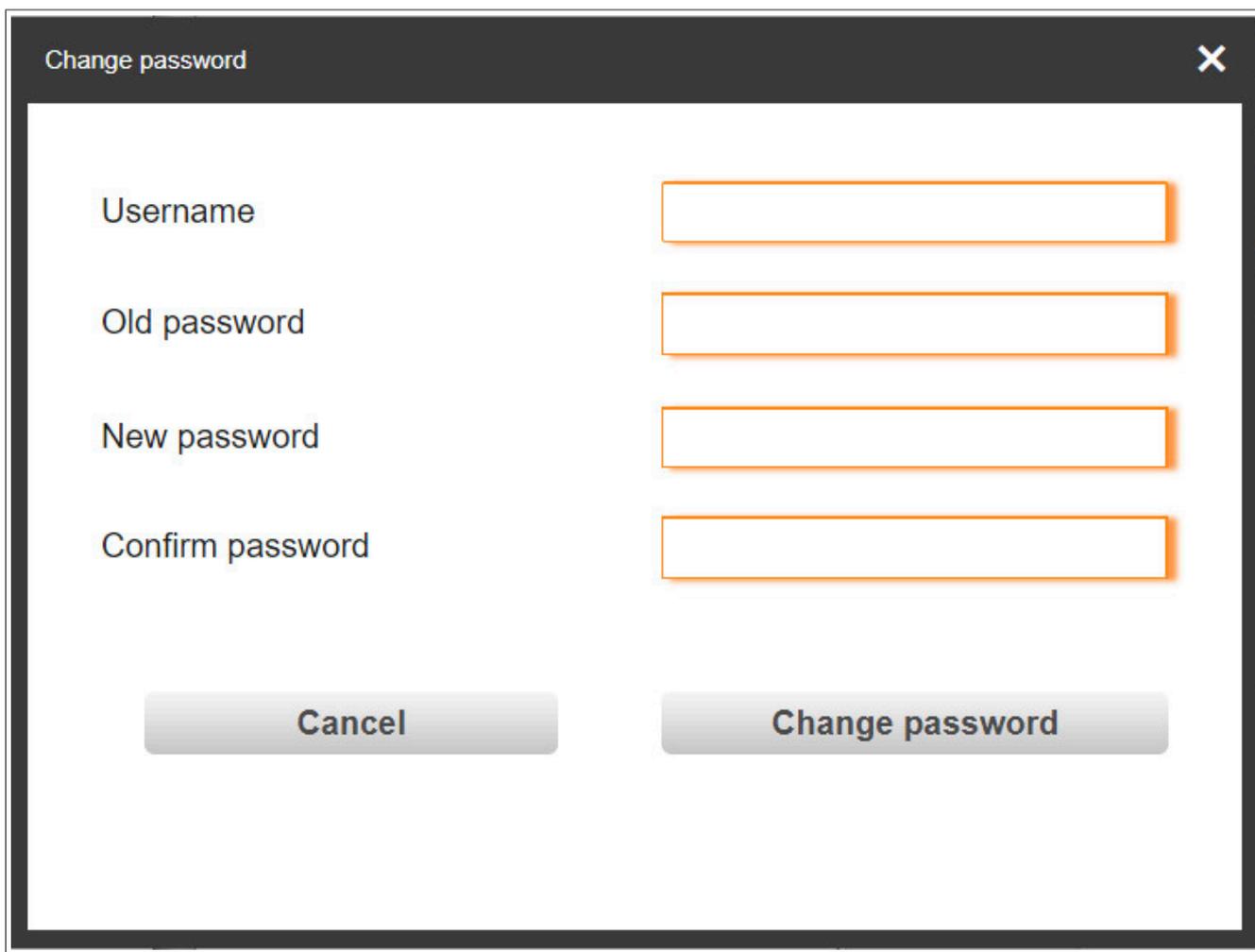
Logout:

Mithilfe des Logout-Button meldet sich der Benutzer im System ab. Die Visualization Session wurde geschlossen.



Ändern des Benutzerpassworts

Das Passwort eines Benutzers kann mithilfe des Systemdialogs `OpenChangePasswordDialog` in `mapp View` geändert werden. Dabei muss der Benutzername, das alte Passwort, sowie ein neues Passwort eingegeben werden.



Automatisches Ausloggen bei Inaktivität

In diesem Abschnitt wird erklärt, wie ein angemeldeter Benutzer bei Inaktivität automatisch ausgeloggt wird. Die Auslog-Zeit wird dafür einmalig festgelegt und gilt für alle Benutzer. Die Zeit wird in Sekunden in der [MpUserX Konfiguration](#) unter "Auto-logout time" festgelegt.

Bei einer Angabe von 0 Sekunden ist die Funktion nicht verfügbar, das heißt, es wird nie ein Benutzer bei Inaktivität automatisch ausgeloggt.

Werden beispielsweise 20 Sekunden festgelegt, so wird der Benutzer, sobald dieser im System angemeldet ist, nach 20 Sekunden Inaktivität automatisch ausgeloggt.

Die Auslog-Zeit wird wieder auf den Ursprungs-Wert (20 Sekunden) zurückgesetzt, sobald eine Aktivität (activity-Count) auf der mapp View Visualisierung festgestellt wurde, wie beispielsweise das Klicken auf einen Button.

Benachrichtigung über ablaufendes Passwort

In diesem Abschnitt wird erklärt, wie ein Benutzer über das ablaufende Passwort informiert werden kann.

In der [MpUserX Konfiguration](#) kann über "Password expiration notification" festgelegt werden, wann ein Benutzer über das Ablauf seines Passwortes benachrichtigt werden soll. Wie die Zeit definiert werden kann, ist im Abschnitt [Angabe der Zeit](#) erklärt. Bei einer Eingabe von "5d" wird der Benutzer 5 Tage bevor sein Passwort abläuft benachrichtigt. Die Benachrichtigung erfolgt dabei über das Client-System PasswordExpirationNotification Ereignis. Die verbleibende Zeit kann auch von der Systemvariable `::SYSTEM.clientInfo.passwordExpirationTime` gelesen werden. Dabei erfolgt die Angabe in Tagen.

Ist die Frist zur Passwort-Änderung abgelaufen und der Benutzer ist im System noch angemeldet, so wird der Benutzer nicht ausgeloggt oder verliert Zugriffsrechte. Erst sobald eine neue Anmeldung erfolgt wird wieder das Client-System PasswordExpirationNotification Ereignis ausgelöst.

Zentral verwaltete Benutzer

Wird als [User management Modus](#) "Zentral" bzw. "Lokal und zentral" ausgewählt, so gilt es, abhängig vom verwendeten Directory Server, bei zentralen Benutzern folgendes zu beachten:

- **Active Directory Server:** Bei Benutzern welche am Active Directory Server verwaltet werden gilt es zu beachten, dass das Ablauf-Datum des Passworts am Server festgelegt wird und nicht in der Konfiguration. Das Ablauf-Datum des Passworts wird nur abgefragt wenn eine Verbindung zum Active Directory Server besteht. Dies ist während eines Login-Vorgangs der Fall. Wurde das Passwort innerhalb der Frist nicht geändert und ein neuer Login-Vorgang wird gestartet, so kann sich der zentrale Benutzer nicht mehr an der Maschine anmelden. Das Client-System PasswordExpirationNotification Ereignis wird ausgelöst. Die Passwort-Änderung muss nun direkt am Active Directory Server erfolgen. Nachdem das Passwort geändert wurde, kann sich der Benutzer wieder über die Maschine anmelden.
- **389 Directory Server:** Wann für einen Benutzer das Passwort abläuft, wird direkt über den 389 Directory Server bestimmt nicht über die [MpUserX Konfiguration](#). Während eines Login-Vorgangs wird die Information von mapp UserX am 389 Directory Server abgefragt und intern von mapp UserX verwendet. Die verbleibende Zeit kann dadurch auch von der Systemvariable `::SYSTEM.clientInfo.passwordExpirationTime` gelesen werden. Wurde das Passwort innerhalb der Frist nicht geändert und ein neuer Login-Vorgang wird gestartet, so kann sich der zentrale Benutzer nicht mehr an der Maschine anmelden. Es muss eine Passwort-Änderung durchgeführt werden. Dies kann direkt an der Maschine oder über den 389 Directory Server gemacht werden.

Nutzerpräferenzen

In der [MpUserX Konfiguration](#) kann im Abschnitt "Users" für einen Benutzer festgelegt werden, welche Sprache ("Language") und welches Einheitensystem ("Measurement system") bevorzugt verwendet werden soll. Wird eine Sprache und ein Einheitensystem für einen Benutzer festgelegt und dieser Benutzer meldet sich an, so wird automatisch, wenn benötigt, ein Sprach-, bzw. Einheiten-Wechsel in mapp View durchgeführt.

UserMgmtX		
+ ...	User management system settings	
+ ...	General Settings	
- ...	Users	
- ...	User: Jane	
	UserName	Jane
	FullName	Jane Johnson
	Language	en
	Measurement system	imperial
+ ...	Additional Data	

Zur Laufzeit können diese Nutzerpräferenzen mithilfe von `MpUserXConfig` geändert werden. Bei der Angabe ungültiger Daten, wie ein nicht existierendes Einheitensystem oder der Angabe von keinen Daten, wird die Sprache bzw. das Einheitensystem von mapp View nicht übernommen. Es wird die in mapp View zuletzt verwendete Sprache bzw. Einheit verwendet. Benutzerdefinierte Einheitensysteme werden von mapp View nicht übernommen.

Es gilt zu beachten, dass die Nutzerpräferenzen nur für **lokale Benutzer** gespeichert werden kann!

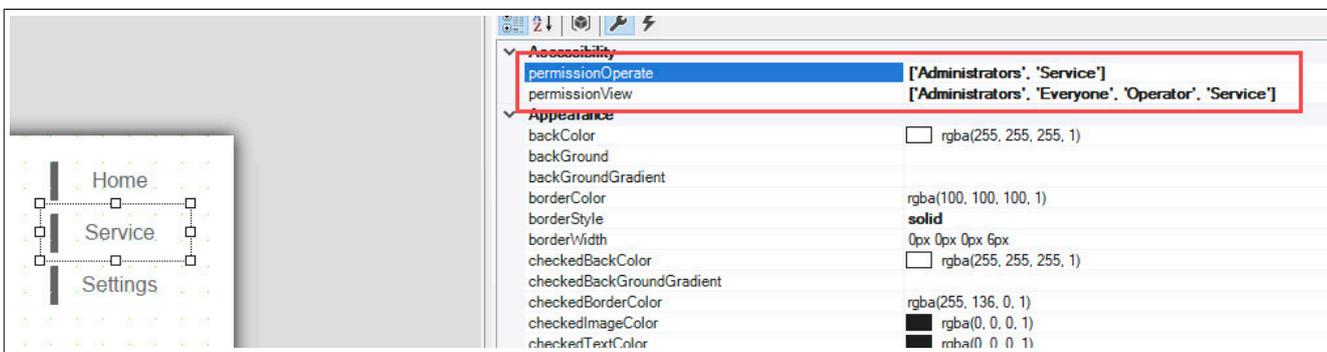
1.4.2.3 Verwalten von Berechtigungen

Meldet sich ein Benutzer im System an, so werden, abhängig von der Rolle des Benutzers, unterschiedliche Berechtigungen aktiviert. Das bedeutet manche Funktionen im System werden gesperrt bzw. freigeschaltet. Dies wird innerhalb von mapp View mit den Eigenschaften "permissionView" und "permissionOperate" gelöst.

Beispiel:

Meldet sich im System der Benutzer Jane an, welche die Rolle "Service" besitzt, so erhält diese die Berechtigung auf die Service-Seite zu navigieren und dort unterschiedliche Parameter zu regulieren. Meldet sich der Benutzer Dave an, welche die Rolle "Operator" besitzt, so ist die Navigation auf die Service-Seite gesperrt. Es können keine Änderungen durchgeführt werden. Die Navigation wird mithilfe von NavigationButton-Widgets gelöst.

Dieses Beispiel kann mithilfe der Eigenschaften "permissionOperate" und "permissionView" in mapp View realisiert werden.



Durch "permissionOperate" werden die Rollen definiert, welche das Recht haben sollen auf die Service-Seite zu navigieren. Durch "permissionView" wird festgelegt welche Rollen das Recht haben den Navigations-Button zu sehen.

1.5 Elektronische Signatur

Die elektronische Signatur ermöglicht das Zeichnen einzelner **Aktionen**. Möchte der Benutzer eine bestimmte Aktion durchführen, dann öffnet sich auf der Visualisierung ein Dialogfenster, welches Benutzernamen und Passwort fordert. Nach Eingabe der Nutzerdaten wird die Aktion, wenn die Berechtigung vorhanden ist, durchgeführt.

Die Elektronische Signatur kann innerhalb einer Application Session sowie Visualization Session verwendet werden.

Ein Aktion kann beispielsweise sein:

- Parameter über die Visualisierung ändern (z.B. Solltemperatur)
- Reinigungsprozess an der Maschine starten
- Aktuellen Produktionsprozess abbrechen

Damit kann für jede Aktion, welche an der Maschine möglich ist, eine elektronische Signatur verlangt werden. Das Zeichnen wird über ein Ereignis im Audit Trail vermerkt. Das Ereignis wird erzeugt unabhängig davon, ob der Benutzer die entsprechenden Berechtigung besitzt.

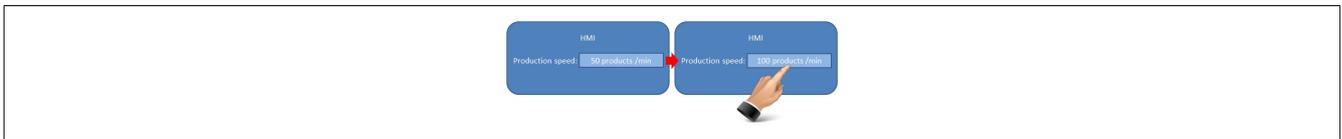
Die Verbindung zur Visualisierung erstellt das Benutzermanagement über den Funktionsblock [MpUserXSignatureUI](#). Über diesen Funktionsblock wird die Anzeige des Dialogfensters geregelt. Die Prüfung, ob der Benutzer die Berechtigung für eine Aktion besitzt erfolgt über [MpUserXSignature](#). Dieser Funktionsblock informiert dann, ob die Signatur erfolgreich war, oder nicht.

Ablauf

Ein möglicher Ablauf beim Zeichnen einer Aktion könnte folgendermaßen aussehen:

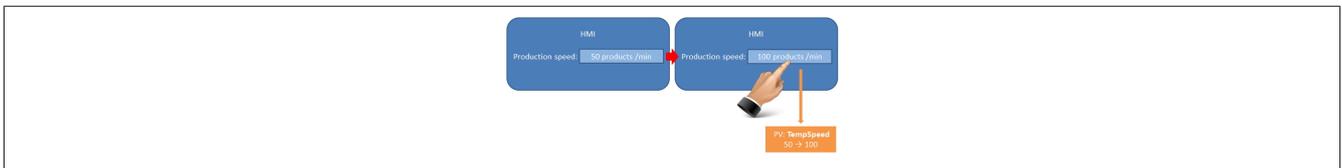
1. Eingabe eines Parameters

Der Benutzer gibt über die Visualisierung einen neuen Wert für die Sollgeschwindigkeit ein. Der Wert wird von 50 Produkten / Minute auf 100 Produkte / Minute erhöht.



2. Temporäre Lagerung des neuen Parameters

Aufgrund der Eingabe wird eine PV in der Applikation auf den neuen Wert gesetzt. Diese ist eine temporäre PV, welche noch keinen direkten Einfluss auf den Prozess nimmt. Gleichzeitig wird der *Completion Datenpunkt* des Eingabefeldes auf einen bestimmten Wert gesetzt. Dieser Wert entspricht der "ActionID", welche in der [MpUserXSign Konfiguration](#) festgelegt wurde. Beispielweise ist in der Konfiguration eine Aktion mit der "ActionID" 1 definiert, welche ein Benutzerlevel von 100 voraussetzt. In unserem Beispiel wird als der *Completion Datenpunkt* auf 1 gesetzt.



3. Dialog zur elektronischen Signatur

Erkennt [MpUserXSignatureUI](#), dass der *Completion Datenpunkt* ungleich null ist, dann wird der Dialog zur elektronischen Signatur geöffnet.



In diesem Dialog wird der Benutzer aufgefordert seinen Benutzernamen, sowie sein Passwort einzugeben. Optional kann auch ein Kommentar angegeben werden, welcher angibt warum eine bestimmte Aktion durchgeführt werden soll. Der Benutzer hat dann die Wahl den Prozess durch *OK* zu Bestätigen, oder durch *Cancel* abbrechen.

4. Prozess zur Verifizierung der eingegebenen Daten

Nun startet ein Prozess zur Überprüfung der Benutzereingaben. Im Bild unten zeigen die rot markierten Objekte einen unterbrochenen Signier - Prozess an. Das kann mehrere Ursachen haben:

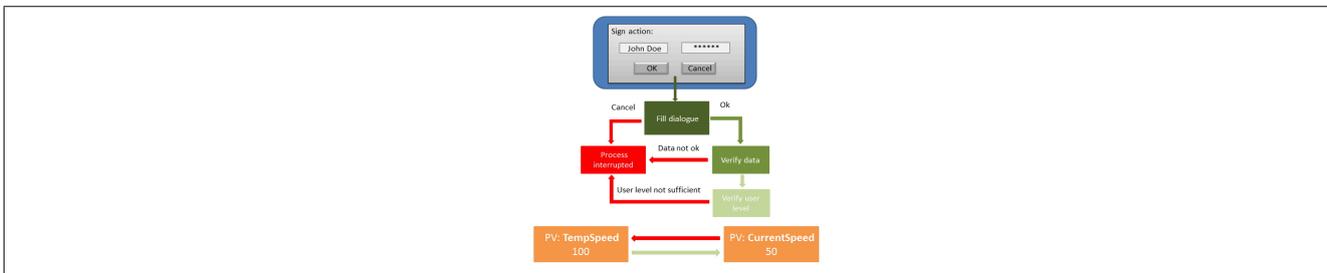
- Der Benutzer drückt *Cancel*, um den Prozess bewusst abbrechen.
Wird der Signaturprozess vom Benutzer abgebrochen, so wird der Ausgang "Rejected" an [MpUserXSignature](#) für einen Zyklus gesetzt
- Der Benutzer hat eine falsche Eingabe gemacht, heißt entweder Benutzername oder Passwort stimmt nicht. In diesem Fall wird auch ein Login-Versuch ("LoginAttempts") verwendet. Das bedeutet, wenn der Benutzername oder das Passwort zu oft falsch eingegeben werden, wird der Benutzer gesperrt. Die Anzahl an Login-Versuchen wird in der [MpUserX Konfiguration](#) definiert. Werden alle Signatur-Versuche aufgebraucht, so wird der Signatur-Vorgang abgebrochen. Der Signatur-Vorgang kann wieder neu gestartet

werden, dabei steht wieder die gleiche Anzahl an Signatur-Versuchen zur Verfügung. Die Login-Versuche werden nicht wieder erhöht.

- Der Benutzer hat nicht das notwendige Benutzerlevel, um die geforderte Aktion durchzuführen, heißt sein Benutzerlevel ist kleiner, als das für die Durchführung der Aktion benötigte Benutzerlevel.

Schlägt der Signaturprozess fehlt (siehe die beiden letzten Punkte) kann der Vorgang wiederholt werden. Wird der **maximal erlaubte Wert** an fehlgeschlagenen Versuchen erreicht, dann wird der Ausgang "Rejected" an **MpUserXSignature** für einen Zyklus gesetzt

- In der Applikation wird in diesem Fall der Wert der temporären PV wieder auf den Wert der aktiven Prozessvariable gesetzt. In unserem Beispiel wird der, zur Produktion verwendete, Geschwindigkeitswert von 50 Produkten / Minute auf die temporäre PV geschrieben



Der grüne Zweig zeigt Schritt für Schritt die erfolgreiche Durchführung einer elektronischen Signatur an:

- Zuerst bestätigt der Benutzer, dass er eine Aktion durchführen möchte durch das Drücken der OK Taste
- **MpUserXSignature** überprüft nun, ob Benutzername, sowie Passwort stimmen
- Stimmen die Nutzerdaten, wird im nächsten Schritt überprüft, ob der Benutzer die angegebenen Berechtigungen besitzt (Benutzerlevel). In unserem Beispiel wird geprüft, ob der Benutzer mindestens ein Benutzerlevel von 100 besitzt
- Falls alle Punkte gegeben sind war die elektronische Signatur erfolgreich. Der Ausgang "Released" von **MpUserXSignature** wird für einen Zyklus gesetzt
- In der Applikation wird in diesem Fall der Wert der aktiven Prozessvariable auf den Wert der temporären PV gesetzt. In unserem Beispiel wird der, über die Visualisierung eingegebene und signierte, Geschwindigkeitswert von 100 Produkten / Minute auf die zur Produktion verwendete PV geschrieben

1.6 Auditing von mapp UserX

Ereignisse in mapp UserX können über mapp Audit aufgezeichnet werden. Für mehr Information wie ein Eintrag erstellt werden kann, siehe Abschnitt Definieren des Ereignistexts.

In diesem Abschnitt werden die verschiedenen Benutzer-Ereignisse aufgelistet:

Benutzermanagement Ereignis

Durch diesen Ereignistyp können Ereignisse des Benutzermanagements, wie zum Beispiel das Löschen eines Benutzers oder Hinzufügen eines Benutzers, aufgezeichnet werden.

Ereignis ID	Komponenten
2	MpUser Komponenten MpUserX Komponenten
Ereignisse	
op	Name des Benutzers welcher beim Auftreten des Ereignisses eingeloggt war
ev	Ereignis ID

Tabelle 1: Benutzermanagement Ereignis

Ereignis ID 2	Komponenten MpUser Komponenten MpUserX Komponenten																																																
idx	Liefert den Index des aktuell generierten Ereignisses. Der Wert wird bei jedem neuen Ereignis inkrementiert. Das erste generierte Ereignis in der Ereignisliste hat somit den Wert 1, das fünfzigste Ereignis hat den Index 50, das hundertste den Index 100 und so weiter.																																																
map	Name der mapp Komponente, welcher das Ereignis über den Mplink zugeordnet ist																																																
usr	Benutzer, welcher von dem Ereignis betroffen ist (z.B. Benutzer der gelöscht wurde)																																																
act	Aktion, welche in einer mapp Komponente ausgeführt wurde. Die ID für die Aktion wird zurückgegeben (z.B. 10 für "Ungültiger Benutzername wurde eingegeben"). Für mehr Informationen siehe folgende Tabelle:																																																
	<table border="1"> <thead> <tr> <th>Ereignis</th> <th>Identifikationsnummer (wird über %act zurückgegeben)</th> <th>Zusatzinformation</th> </tr> </thead> <tbody> <tr> <td>Unbekannte Aktion</td> <td>0</td> <td></td> </tr> <tr> <td>Benutzer hat sich eingeloggt</td> <td>1</td> <td>• {&op}/{%op} - Benutzer</td> </tr> <tr> <td>Benutzer hat sich ausgeloggt</td> <td>2</td> <td>• {&op}/{%op} - Benutzer</td> </tr> <tr> <td>Benutzer wurde automatisch ausgeloggt</td> <td>3</td> <td>• {&op}/{%op} - Benutzer</td> </tr> <tr> <td>Benutzer wurde hinzugefügt</td> <td>4</td> <td>• {&op}/{%op} - welcher Benutzer hat die Operation durchgeführt • {&usr}/{%usr} - welcher Benutzer wurde hinzugefügt</td> </tr> <tr> <td>Benutzer wurde gelöscht</td> <td>5</td> <td>• {&op}/{%op} - welcher Benutzer hat die Operation durchgeführt • {&usr}/{%usr} - welcher Benutzer wurde gelöscht</td> </tr> <tr> <td>Benutzer wurde gesperrt</td> <td>6</td> <td>• {&op}/{%op} - welcher Benutzer hat die Operation durchgeführt • {&usr}/{%usr} - welcher Benutzer wurde gesperrt</td> </tr> <tr> <td>Benutzer wurde entsperrt</td> <td>7</td> <td>• {&op}/{%op} - welcher Benutzer hat die Operation durchgeführt • {&usr}/{%usr} - welcher Benutzer wurde entsperrt</td> </tr> <tr> <td>Benutzer wurde gesperrt, weil er zu oft ein falsches Passwort eingegeben hat</td> <td>8</td> <td>• {&op}/{%op} - welcher Benutzer hat die Operation durchgeführt • {&usr}/{%usr} - welcher Benutzer wurde gesperrt</td> </tr> <tr> <td>Benutzer hat das Passwort geändert</td> <td>9</td> <td>• {&op}/{%op} - Benutzer</td> </tr> <tr> <td>Ungültiger Benutzername wurde eingegeben</td> <td>10</td> <td></td> </tr> <tr> <td>Ungültiges Passwort wurde eingegeben</td> <td>11</td> <td></td> </tr> <tr> <td>Rolle hinzugefügt</td> <td>12</td> <td>• {&op}/{%op} - welcher Benutzer hat die Operation durchgeführt • {&usr}/{%usr} - welche Rolle wurde hinzugefügt</td> </tr> <tr> <td>Rolle gelöscht</td> <td>13</td> <td>• {&op}/{%op} - welcher Benutzer hat die Operation durchgeführt • {&usr}/{%usr} - welche Rolle wurde gelöscht</td> </tr> <tr> <td>Ungültige Anmeldeinformationen wurden eingegeben</td> <td>14</td> <td></td> </tr> </tbody> </table>	Ereignis	Identifikationsnummer (wird über %act zurückgegeben)	Zusatzinformation	Unbekannte Aktion	0		Benutzer hat sich eingeloggt	1	• {&op}/{%op} - Benutzer	Benutzer hat sich ausgeloggt	2	• {&op}/{%op} - Benutzer	Benutzer wurde automatisch ausgeloggt	3	• {&op}/{%op} - Benutzer	Benutzer wurde hinzugefügt	4	• {&op}/{%op} - welcher Benutzer hat die Operation durchgeführt • {&usr}/{%usr} - welcher Benutzer wurde hinzugefügt	Benutzer wurde gelöscht	5	• {&op}/{%op} - welcher Benutzer hat die Operation durchgeführt • {&usr}/{%usr} - welcher Benutzer wurde gelöscht	Benutzer wurde gesperrt	6	• {&op}/{%op} - welcher Benutzer hat die Operation durchgeführt • {&usr}/{%usr} - welcher Benutzer wurde gesperrt	Benutzer wurde entsperrt	7	• {&op}/{%op} - welcher Benutzer hat die Operation durchgeführt • {&usr}/{%usr} - welcher Benutzer wurde entsperrt	Benutzer wurde gesperrt, weil er zu oft ein falsches Passwort eingegeben hat	8	• {&op}/{%op} - welcher Benutzer hat die Operation durchgeführt • {&usr}/{%usr} - welcher Benutzer wurde gesperrt	Benutzer hat das Passwort geändert	9	• {&op}/{%op} - Benutzer	Ungültiger Benutzername wurde eingegeben	10		Ungültiges Passwort wurde eingegeben	11		Rolle hinzugefügt	12	• {&op}/{%op} - welcher Benutzer hat die Operation durchgeführt • {&usr}/{%usr} - welche Rolle wurde hinzugefügt	Rolle gelöscht	13	• {&op}/{%op} - welcher Benutzer hat die Operation durchgeführt • {&usr}/{%usr} - welche Rolle wurde gelöscht	Ungültige Anmeldeinformationen wurden eingegeben	14	
Ereignis	Identifikationsnummer (wird über %act zurückgegeben)	Zusatzinformation																																															
Unbekannte Aktion	0																																																
Benutzer hat sich eingeloggt	1	• {&op}/{%op} - Benutzer																																															
Benutzer hat sich ausgeloggt	2	• {&op}/{%op} - Benutzer																																															
Benutzer wurde automatisch ausgeloggt	3	• {&op}/{%op} - Benutzer																																															
Benutzer wurde hinzugefügt	4	• {&op}/{%op} - welcher Benutzer hat die Operation durchgeführt • {&usr}/{%usr} - welcher Benutzer wurde hinzugefügt																																															
Benutzer wurde gelöscht	5	• {&op}/{%op} - welcher Benutzer hat die Operation durchgeführt • {&usr}/{%usr} - welcher Benutzer wurde gelöscht																																															
Benutzer wurde gesperrt	6	• {&op}/{%op} - welcher Benutzer hat die Operation durchgeführt • {&usr}/{%usr} - welcher Benutzer wurde gesperrt																																															
Benutzer wurde entsperrt	7	• {&op}/{%op} - welcher Benutzer hat die Operation durchgeführt • {&usr}/{%usr} - welcher Benutzer wurde entsperrt																																															
Benutzer wurde gesperrt, weil er zu oft ein falsches Passwort eingegeben hat	8	• {&op}/{%op} - welcher Benutzer hat die Operation durchgeführt • {&usr}/{%usr} - welcher Benutzer wurde gesperrt																																															
Benutzer hat das Passwort geändert	9	• {&op}/{%op} - Benutzer																																															
Ungültiger Benutzername wurde eingegeben	10																																																
Ungültiges Passwort wurde eingegeben	11																																																
Rolle hinzugefügt	12	• {&op}/{%op} - welcher Benutzer hat die Operation durchgeführt • {&usr}/{%usr} - welche Rolle wurde hinzugefügt																																															
Rolle gelöscht	13	• {&op}/{%op} - welcher Benutzer hat die Operation durchgeführt • {&usr}/{%usr} - welche Rolle wurde gelöscht																																															
Ungültige Anmeldeinformationen wurden eingegeben	14																																																

Tabelle 1: Benutzermanagement Ereignis

Wie der Ereignistext angegeben werden kann ist in Abschnitt Verwendung des neuen Formats erklärt.

Beispiel neues Format:

- **'{&op} - Aktion: Benutzer {&usr} wurde gelöscht'** ausgewertet: 'Admin Aktion: Benutzer Mike wurde gelöscht', was bedeutet der Benutzer Admin hat den Benutzer Mike gelöscht

Beispiel altes Format:

- **'[%op] - Aktion: Benutzer [%usr] wurde gelöscht'** ausgewertet: 'Admin Aktion: Benutzer Mike wurde gelöscht', was bedeutet der Benutzer Admin hat den Benutzer Mike gelöscht

Elektronische Signatur Ereignis

Durch diesen Ereignistyp können Ereignisse rund um die Elektronische Signatur aufgezeichnet werden.

Ereignis ID 3	Komponenten MpUserSignature MpUserSignatureUI MpUserXSignature MpUserXSignatureUI
Ereignisse	
cmt	Kommentar, welcher vom Benutzer bei der Signatur angegeben wurde
id	Name der mapp Komponente, welcher das Ereignis über den Mplink zugeordnet ist

Tabelle 2: Elektronische Signatur Ereignis

Ereignis ID 3	Komponenten MpUserSignature MpUserSignatureUI MpUserXSignature MpUserXSignatureUI		
idx	Liefert den Index des aktuell generierten Ereignisses. Der Wert wird bei jedem neuen Ereignis inkrementiert. Das erste generierte Ereignis in der Ereignisliste hat somit den Wert 1, das fünfzigste Ereignis hat den Index 50, das hundertste den Index 100 und so weiter.		
err	Fehlernummer des Benutzermanagements (kann beispielsweise verwendet werden, um anzuzeigen ob der Benutzername oder das Passwort falsch waren)		
act	Liefert die Aktions-ID welche in der Konfiguration für eine Signatur angegeben werden muss		
typ	Aktion, welche in einer mapp Komponente ausgeführt wurde. Die ID für die Aktion wird zurückgegeben (z.B. 2 für "Elektronische Signatur wurde abgebrochen"). Für mehr Informationen siehe folgende Tabelle:		
	Ereignis	Identifikationsnummer (wird über typ zurückgegeben)	Ereignis als Text (wird über typt zurückgegeben)
	Elektronische Signatur wurde erfolgreich durchgeführt	0	OK
	Elektronische Signatur wurde beendet aufgrund einer falschen Eingabe (Benutzername oder Passwort)	1	FAIL
	Elektronische Signatur wurde vom Benutzer abgebrochen	2	ABORT
	Elektronische Signatur wurde beendet, da das Limit für falsche Eingaben (siehe Konfiguration des Benutzermanagements) überschritten wurde	3	FATAL

Tabelle 2: Elektronische Signatur Ereignis

Wie der Ereignistext angegeben werden kann ist in Abschnitt Verwendung des neuen Formats erklärt.

Beispiel neues Format:

- 'Signatur ID {&act}: {&typ} ({&typt}) ' ausgewertet: 'Signatur ID 3: Elektronische Signatur wurde erfolgreich durchgeführt (OK)', was bedeutet, dass die Signatur mit der ID 3 erfolgreich durchgeführt wurde

Beispiel altes Format:

- 'Signatur ID [%act]: [%typ] ([%typt]) ' ausgewertet: 'Signatur ID 3: Elektronische Signatur wurde erfolgreich durchgeführt (OK)', was bedeutet, dass die Signatur mit der ID 3 erfolgreich durchgeführt wurde

1.7 Konfiguration während der Laufzeit

Die Konfigurations Parameter können zur Laufzeit über [MpUserXConfig](#), [MpUserXLoginConfig](#), [MpUserXMappingConfig](#) oder [MpUserXServerConfig](#) verändert werden. Dabei ist für die unterschiedlichen Parameter jedoch zu beachten in welchem Zustand sich die Komponente befinden muss um eine Änderung zuzulassen. Folgende Zustände sind möglich:

-  : Dieses Symbol bedeutet, dass dieser Parameter in jedem Zustand geändert werden kann
-  : Dieses Symbol bedeutet, dass sich der Benutzer ab- und wieder anmelden muss damit die Änderungen übernommen werden

Jeder Parameter der Konfigurations Struktur ([MpUserXConfigType](#), [MpUserXLoginConfigType](#), [MpUserXMappingConfigType](#) und [MpUserXServerConfigType](#)) ist mit einem dieser Symbole ausgestattet.

1.8 Rahmenbedingungen

In diesem Abschnitt werden Rahmenbedingungen von mapp UserX aufgelistet.

Allgemein

- In einem Projekt kann nur mit mapp UserX oder mit mapp User gearbeitet werden. Nicht mit beidem parallel. Beide Komponenten dienen zum Einrichten eines Benutzermanagements. In mapp User werden die Benutzer in Gruppen mit unterschiedlichen Berechtigungen unterteilt. In mapp UserX werden Rollen und Benutzer über das Automation Studio Benutzer- Rollensystem, damit auch OPC UA konform, erstellt und verwaltet.
- Wird zur Laufzeit ein neuer Benutzer erstellt, so wird diesem Benutzer automatisch auch die Rolle "Everyone" zugewiesen.
- Alle Rollen und Benutzer, welche in der Konfiguration mapp UserX angelegt werden, werden als Default User bzw Default Rolle bezeichnet

- Beginnend mit der mapp Services Version 5.14.0 ändert sich das Transferverhalten der mapp UserX Komponente. Für weitere Information siehe "Ändern der mapp Services Version auf < 5.14 bzw. > 5.14" im Abschnitt Rahmenbedingungen.
- In der [MpUserX Konfiguration](#) können Zeiten mithilfe von Kürzel angegeben werden, wie beispielsweise "1d 12m". Für mehr Information siehe [hier](#) im Abschnitt "Angabe der Zeit". Diese Angabe der Zeit kann ab der Automation Studio Version >= 4.7 verwendet werden. Für die Zeitangabe wird ein DINT-Wert verwendet. Maximale Zeitangabe wäre dadurch "24855d 3h 14m 7s".

User management

Allgemein

- Es können maximal für 20 unterschiedliche Funktionen Rechte definiert werden

Zentrales User management

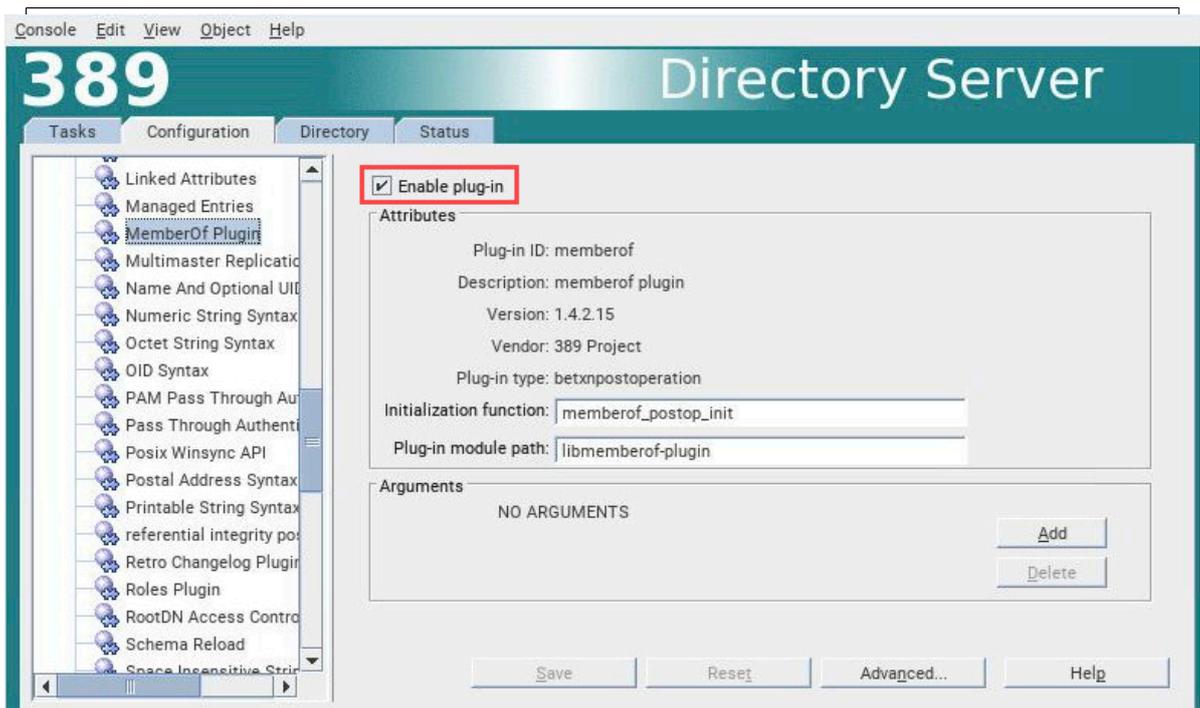
Wird mapp UserX als [Zentrales User management](#) verwendet, so arbeitet die Komponente mit einem Directory Server zusammen. Dabei gilt es folgendes zu beachten:

Software

- Um das zentrale User management verwenden zu können, wird die Automation Runtime Version >= F4.44 benötigt
- Wird ein Windows-basierendes Betriebssystem verwendet, so muss der Active Directory Server in der [MpUserX Konfiguration](#) verwendet werden. Die Funktion wurde mit dem Windows Server 2019 getestet
- Wird ein Linux-basierendes Betriebssystem verwendet, so muss der 389 Directory Server in der [MpUserX Konfiguration](#) angegeben werden. Die Funktion wurde mit der Server Version 1.4.2.15 getestet

Funktionen

- Der Import/Export von Rollen und Benutzern funktioniert nicht für Rollen und Benutzer welche am Directory Server verwaltet werden
- Am Zielsystem können keine Benutzer/Rollen erstellt, bearbeitet oder gelöscht werden, welche über den Directory Server verwaltet werden. Änderungen sind nur am Directory Server möglich. Es ist nur möglich eine Passwortänderung am Zielsystem durchzuführen. Dazu muss der Benutzer angemeldet sein. Rollen und Benutzer welche lokal an der Maschine verwaltet werden, können beliebig editiert und importiert/exportiert werden
- Damit sich Benutzer über den 389 Directory Server anmelden können, muss am Server das Plugin "memberOf" aktiviert sein:



Zertifikat

- Es muss ein [Zertifikat](#) angegeben werden. Dabei muss ein base64 codiertes X.509 Zertifikat verwendet werden. Im Zertifikat-Namen dürfen folgende Sonderzeichen nicht verwendet werden: \ / : * ? " < > |
- Es muss sichergestellt sein, dass die Systemzeit des Zielsystems korrekt ist. Wurde die Systemzeit auf ein älteres Datum zurückgestellt, so kann es sein, dass das angegebene [Zertifikat](#) nicht mehr gültig ist.
- **Active Directory Server:** Wird ein Active Directory Server (Windows-basierendes Betriebssystem) verwendet, so kann das Zertifikat die Endungen ".pem", ".cer" or ".crt". besitzen
- **389 Directory Server:** Wird ein 389 Directory Server (Linux-basierendes Betriebssystem) verwendet, so muss das Zertifikat die Endung ".pem" besitzen.

Allgemein

- Meldet sich ein Benutzer an, so hat der Benutzer nicht nur die ihm zugewiesene Rolle, wie beispielsweise "Operator", sondern besitzt standardmäßig auch die Rolle "Everyone"
- Bei der Verwendung des [zentralen und lokalen User managements](#) muss darauf geachtet werden, dass die lokal erstellten Benutzernamen nicht identisch mit denen am Directory Server sind. Dies kann ansonsten zu Authentifizierungs-Problemen führen
- Bei einem Login-Vorgang muss bei zentralen Benutzern darauf geachtet werden, dass der korrekte Benutzername verwendet wird. Wurde beispielsweise nur der UPN Name im Active Directory Server vergeben, so kann sich der Benutzer nur durch Verwendung des Formats "username@domainname.com" anmelden. Wurde nur der SamAccountName vergeben, so muss das Format "domainname\username" verwendet werden. Wurde UPN sowie SamAccountName vergeben können beide Formate verwendet werden.

- Bei der Verwendung von zentralen Benutzern muss darauf geachtet werden, dass die Rolle des Benutzers am Active Directory Server nicht als "Primary Group" Rolle markiert wurde. "Primary Group" Rollen können nicht zusammen mit mapp UserX verwendet werden.

Member Of

- In der Konfiguration des Zielsystems müssen, damit das zentrale User management funktioniert, die DNS Parameter aktiviert werden. Dabei muss der DNS suffix vergeben werden. Im unteren Bild wurde beispielsweise der Suffix "br-automation.co.at" verwendet. Dieser muss, je nach verwendeten Netzwerk, dementsprechend definiert werden. Wird der DNS von einem DHCP Server bezogen, so muss die Option "Get DNS from DHCP server" aktiviert werden.

Name	Value
X20CP1586	
Configuration ID	mappUserX_Config2
Configuration version	1.0.0
Module system on target	
Simulation	
Memory configuration	
System	
Reboot	
Communication	
Timing	
Resources	
File devices	
Time synchronization	
Internet file system	
Ethernet parameters	
DNS parameters	
Activate DNS service	on
DNS suffix	br-automation.co.at
DNS server	
Get DNS from DHCP server	on
Online parameters	

In Simulation ist diese Einstellung nicht notwendig, nur wenn das zentrale User management auf einer Hardware läuft. Mehr Information zu den DNS Parametern kann hier gefunden werden.

2 Konfiguration

2.1 MpUserLoginX Konfiguration

Die Daten dieser Konfiguration gelten nur für einen Login - Punkt (Panel, an welchem sich jemand über Benutzernamen und Passwort einloggen kann). Ein Login - Punkt bezeichnet einen Station, an welcher der Benutzer die Maschine / einen Teil der Maschine steuern kann. Für jeden Login - Punkt den es für eine Maschine gibt, wird eine dieser Konfigurationen benötigt. Zumindest eine ist notwendig, sonst kann sich niemand einloggen. Bei den unterschiedlichen Funktionen und Funktionsblöcken, welche MpUserX bietet, wird immer mit dem MpLink dieser Konfiguration gearbeitet.

Die Konfiguration kann zur Laufzeit mithilfe des Funktionsbausteins [MpUserXLoginConfig](#) angepasst werden.

Der Parameter "AutoLogout" wird automatisch auf den angegebenen Wert zurückgesetzt sobald sich der Eingang "LifeSign" von MpUserXLogin erhöht. Verändert sich der Wert an "LifeSign" während der angegebenen Ausloggzeit nicht, wird der Benutzer automatisch ausgeloggt.

2.2 MpUserX Konfiguration

Diese Konfiguration wird immer benötigt, wenn ein Benutzermanagement über mapp realisiert werden soll. Die Daten in dieser Konfiguration gelten für die komplette Maschine. Deshalb darf nur eine solche Konfiguration eingefügt werden pro Maschine (bzw. pro Maschinenkonfiguration in der Configuration View in Automation Studio).

Alle Benutzergruppen und Benutzer werden über das Benutzer-Rollensystem erstellt und können dann hier noch erweitert werden (z.B. Administratorrechte vergeben). Alle Benutzer / Rollen welche über das Benutzer-Rollensystem erstellt werden, gelten an jedem Logging - Punkt (Panel, an welchem man sich über Benutzername und Passwort einloggen kann). Der MpLink dieser Konfiguration wird für keine(n) Funktion bzw. Funktionsblock von MpUserX verwendet.

Die Konfiguration kann zur Laufzeit durch den Funktionsbaustein **MpUserXConfig** angepasst werden.

Allgemeine Einstellungen

Im ersten Abschnitt der Konfiguration können allgemeine Einstellungen festgelegt werden.

- **Password change interval:** Intervall, in dem das Passwort geändert werden muss
- **User expiration time:** Gibt die Verfallszeit für einen Benutzer an. Nach Ablauf der Zeit ist der Benutzer gesperrt. Der Benutzer kann nur von einem Admin entsperrt werden. Bei der Angabe von "0" wird die Funktion nicht verwendet.
- **Password expiration notification:** Legt fest, wann der Benutzer über den Ablauf des Passworts benachrichtigt werden soll. Die Angabe erfolgt hier in Sekunden. Wird beispielsweise 432000 eingegeben, so wird der Benutzer 5 Tage vor Ablauf des Passworts informiert. Bei der Eingabe von 0 wird die Benachrichtigungs-Funktion deaktiviert.
- **Admin unlock time:** Nach Ablauf der angegebenen Zeit wird der gesperrte Administrator entsperrt. (Einheit: [sec]) Wird 0 angegeben, wird der Administrator nicht automatisch entsperrt. Die Entsperr-Funktion ist bei 0 nicht verfügbar
- **Auto logout time:** Durch "Auto logout time" wird der Benutzer bei Inaktivität automatisch nach der angegebenen Zeit ausgeloggt. In einer **Application Session** wird der Parameter "Auto logout time" automatisch auf den angegebenen Wert zurückgesetzt sobald sich der Eingang "LifeSign" von **MpUserXLogin** erhöht. Verändert sich der Wert an "LifeSign" während der angegebenen Ausloggzeit nicht, wird der Benutzer automatisch ausgeloggt. In einer **Visualization Session** wird der Parameter "Auto logout time" automatisch auf den angegebenen Wert zurückgesetzt sobald eine Benutzer-Aktion an der Visualisierung durchgeführt wurde.

Wird als automatische Ausloggzeit 0 Sekunden angegeben, so wird die Funktion deaktiviert.

- **Password change required:** Passwortänderung beim ersten Login notwendig Ändert ein Administrator das Passwort eines Benutzers muss dieser sein Passwort beim nächsten Login ändern. Solange der Benutzer das Passwort nicht ändert hat dieser keine Zugriffsrechte. Um dieses Verhalten zu ermöglichen muss dieser Parameter auf **TRUE** gesetzt werden.
- **Password case:** Passwort muss Klein- und Großbuchstaben enthalten
- **Password alphanumeric:** Passwort muss alphanumerisch sein
- **Password length:** Minimale Passwortlänge. Es muss mindestens ein Zeichen angegeben werden. Es ist nicht erlaubt eine Passwort-Länge von 0 zu haben.
- **Password special characters:** Bestimmt, ob ein Passwort Sonderzeichen enthalten muss. Folgende Sonderzeichen können verwendet werden: `!#$%&'()*+,-./:;<=>?@[\\]^_`{|}~`
- **Username length:** Minimale Länge bei der Eingabe eines neuen Benutzernamens
- **Password History Size:** Gibt an wie viele vorherigen Passwörter nicht wiederverwendet werden dürfen Es kann ein Wert zwischen 0 und 5 eingegeben werden Bei der Angabe von 0 wird die Funktion nicht verwendet Bei der Angabe von zum Beispiel 2, darf das neu eingegebene Passwort nicht mit den zwei vorherigen Passwörtern übereinstimmen
- **Prevent username reuse:** Definiert das Lösch-Verhalten innerhalb des Benutzermanagements. Wird der Parameter auf **TRUE** gesetzt, so kann der Benutzername, welcher ein gelöschte Benutzer verwendet hat, nicht mehr vergeben werden. Durch diese Funktion wird sichergestellt, dass Benutzernamen nur einmalig im System verwendet werden können. Wurde also beispielsweise der Benutzer "Dave" gelöscht, so kann kein Benutzer mehr mit diesen Namen erstellt werden. Dadurch wird sichergestellt, dass beispielsweise in Ereignislisten keine Verwechslung im Benutzernamen stattfindet.
- **Login attempts:** Maximale Anzahl an möglichen Login - Fehlversuchen, bis der Benutzer gesperrt wird. Bei der Eingabe von 0 wird der Benutzer niemals gesperrt. Bei Benutzergruppen mit Adminrechten werden diese maximal eine Stunde gesperrt, bzw. sind Benutzergruppen mit Adminrechten nach einem Warmstart der Maschine nicht mehr gesperrt.
- **Signature attempts:** Maximale Anzahl an möglichen Signatur- Fehlversuchen, bis der Signatur-Vorgang abgebrochen wird. Die Login-Versuche ("LoginAttempts") und die Signatur-Versuche hängen miteinander zusammen. Bei jeder falschen Benutzereingabe während eines Signatur-Versuchs wird auch ein Log-

in-Versuch verwendet. Gibt es 2 Signatur-Versuche und 1 Login-Versuch, so wird der Benutzer bei einem fehlgeschlagenen Signatur-Versuch (Eingabe eines falschen Passworts/Benutzernamens) bereits gesperrt.

- **Edit users with same user-level:** Bestimmt, ob ein Admin Benutzer mit dem gleichem Benutzerlevel editieren darf
- **User Import Mode:** Modus für Benutzerimport. Für mehr Information siehe "Import Modi".
- **Role Import Mode:** Modus für Rollenimport. Für mehr Information siehe "Import Modi".
- **Unchecked Import:** Ignorieren der Check-Summe in der Importdatei. Für mehr Information siehe "Import Modi".

Angabe der Zeit

Ab der Automation Studio Version >= 4.7 können Zeitangaben in der Konfiguration einfacher definiert werden. Mithilfe von Kürzeln kann die Zeit in Tagen, Stunden, Minuten und Sekunden angegeben werden. Folgende Kürzel können dabei verwendet werden:

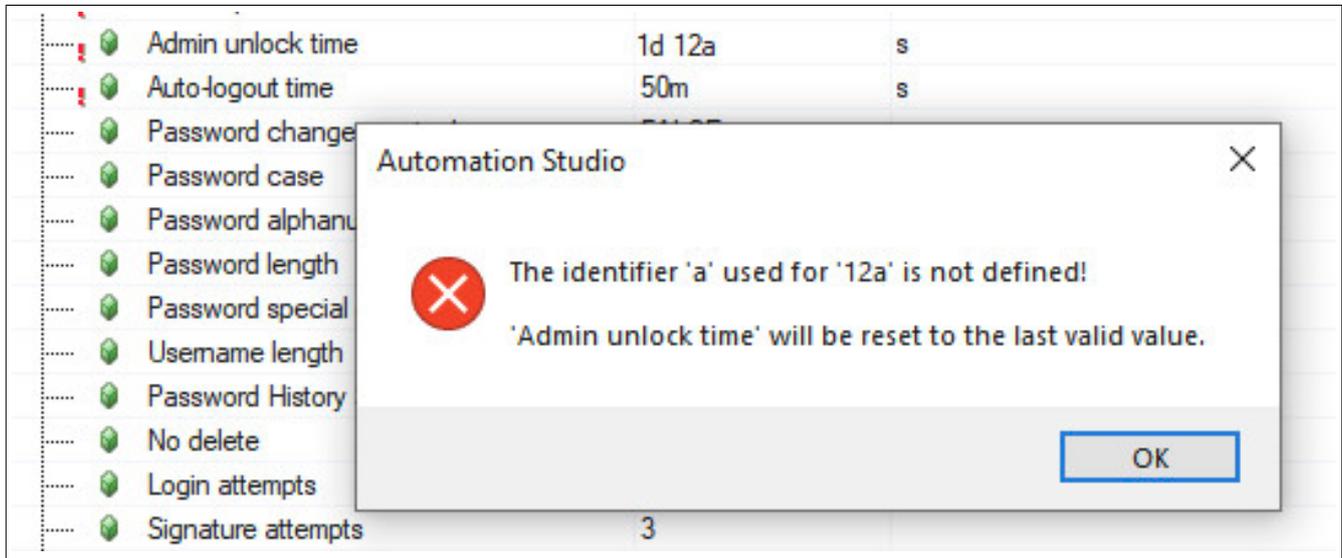
- **d:** Mithilfe von "d" können Tage definiert werden
- **h:** Durch "h" werden Stunden angegeben
- **m:** "m" definiert die Minuten
- **s:** Mithilfe von "s" werden Sekunden angegeben

Es können ein oder mehrere Kürzel miteinander kombiniert werden:

Name	Value
UserMgmtX	
User management system settings	
Type	Local
General Settings	
Password change interval	30d
Password expiration notification	5d 8h 20m 10s
User expiration time	50d
Admin unlock time	30m 15s
Auto-logout time	50m

Wird kein Kürzel angegeben, so wird die Zeit in Sekunden definiert. Eine Angabe von "3600" wäre 3600 Sekunden, also eine Stunde. Die Anordnung der Kürzel muss nicht beachtet werden, eine Zeit kann auch mit "2h 3d" definiert werden, in der Konfiguration wird die Anordnung automatisch angepasst.

Bei einer ungültigen Angabe, wie beispielsweise einer Angabe von "1d 12a", erscheint ein Pop-up-Fenster, welches die ungültige Eingabe erklärt. Beim Schließen des Pop-up-Fensters wird automatisch der zuletzt gültige Wert in der Konfiguration eingetragen.



Server-Einstellungen

Bei der Auswahl "Central" oder "Local and central" in den Benutzermanagement Einstellungen ("User management system settings") werden die Server-Einstellungen sichtbar. Das Benutzermanagement wird dadurch zentral bzw. lokal und zentral verwaltet. Das bedeutet mapp UserX ist mit einem Server verbunden, welcher Benutzer zentral verwaltet.

Um das zentrale User management verwenden zu können, wird die Automation Runtime \geq F4.44 benötigt.

Abhängig vom verwendeten Betriebssystem kann zwischen zwei Servern gewählt werden.

Werden die zentralen Benutzer auf einem Linux-basierenden System verwaltet, so muss der 389 Directory Server ausgewählt werden.

Werden die zentralen Benutzer auf einem Windows-basierenden System verwaltet, so wird der Active Directory Server ausgewählt.

Für mehr Information zu den Servern siehe Abschnitt [Modus des User managements](#).

Allgemeine Einstellungen

Unabhängig vom verwendeten Server müssen die Parameter "Host", "Port" sowie "Certificate name" angegeben werden.

Um eine Verbindung herzustellen muss der Host und der Port des Servers angegeben werden. Unter "Host" wird der DNS Name des Servers angegeben. Dies kann beispielsweise "myserver.mydomain.com" sein oder "SR-V1.br-automation.at" sein. Die Angabe einer IP-Adresse ist nicht möglich.

"Port" bestimmt die Portnummer des Servers.

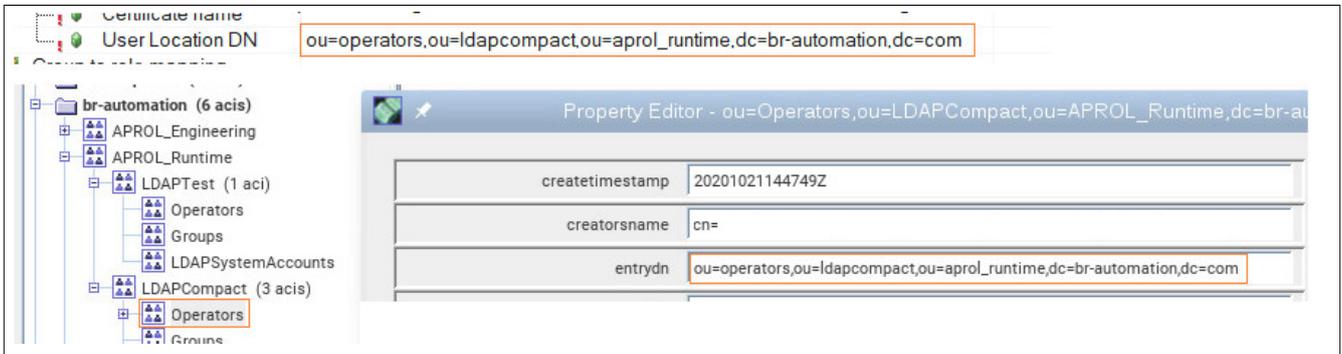
Um eine sichere Verbindung zum Server zu gewährleisten, muss ein Zertifikat zur Server-Verifizierung verwendet werden. Das Zertifikat muss dabei unter "Certificate name" angegeben werden. Als Zertifikat wird ein base64 X.509 Zertifikat verwendet. Das Zertifikat kann die Endungen ".pem", ".cer" oder ".crt" besitzen.

Active Directory spezifische Parameter

Bei der Aktivierung der erweiterten Parameters kann die "BaseDN" definiert werden. Diese gibt den Punkt an an dem der Server die Suche nach Rollen und Benutzern startet. Die "BaseDN" folgt automatisch dem Format des angegebenen Hosts. Bei einer Abweichung kann die "BaseDN" aber auch selbst definiert werden. Diese kann beispielsweise "DC=br-automation,DC=at" sein.

389 Directory Server spezifische Parameter

Unter "User Location DN" muss der vollqualifizierte Name der Organisationseinheit, in der sich der Nutzer am Server befindet, angegeben werden. Es gilt zu beachten, dass nur Benutzer sich anmelden können, welche sich in der angegebenen Organisationseinheit befinden. Benutzer, die sich in untergeordneten oder übergeordneten Organisationseinheiten befinden, können sich nicht anmelden. Am 389 Directory Server wird die "User Location DN" als "entrydn" bezeichnet.



Mapping table

Unter "Mapping table" muss eine [MpUserXMapping Konfiguration](#) angegeben werden. In dieser Konfiguration werden die Active Directory Rollen auf die Automation Studio Rollen gemappt. Für mehr Information dazu siehe [hier](#).

Additional user data

Über "Additional data" kann eine zusätzliche Benutzerinformation abgefragt werden. Welche Information das sein soll wird über "Attribute name" definiert. Für mehr Information siehe "Anzeigen von zusätzlicher Benutzerinformation" im Abschnitt [Funktionen](#).

Rollen

Im Abschnitt "Roles" werden die im Benutzer-Rollen-System angegebenen Rollen ausgewählt. "Level" bestimmt das Benutzerlevel der jeweiligen Rolle. Durch "Administrator" kann festgelegt werden, ob die Rolle ein Administrator sein soll oder nicht. Über "Rights" werden die Rechte für die Rollen definiert. Für mehr Information siehe [Rechte und Levels](#).

Benutzer

Im Abschnitt "Users" werden die im Benutzer-Rollen-System angegebenen Benutzer ausgewählt. Über "FullName" kann der Name des Benutzers angegeben werden. Durch "Language" und "Measurement system" kann die vom Benutzer bevorzugte Sprache bzw. Einheitensystem angegeben werden. Es gilt zu beachten, dass die Nutzerpräferenzen für [zentrale Benutzer](#) nicht gespeichert werden können. Für weitere Information siehe [hier](#) bzw. [hier](#).

Unter "Additional Data" kann für den Benutzer zusätzliche Information hinterlegt werden. Wie zum Beispiel die Telefonnummer (telNr) und die E-Mail Adresse (eMail). Es können aber auch selbst definierte zusätzliche Informationen sein, wie zum Beispiel Lieblingsfarbe etc.

Die zusätzlichen Informationen eines Benutzers, wie zum Beispiel Einheit oder Sprache, werden in der [Konfiguration](#) und nicht im Benutzer-Rollen-System gespeichert.

Aus diesem Grund werden, bei einer Neuübertragung des Projekts auf das Zielsystem, die Benutzerdaten auf dem Zielsystem, mit denen aus der Konfiguration überschrieben. Um die zusätzlichen Informationen, welche zur Laufzeit vielleicht bearbeitet wurden, nicht zu verlieren, empfiehlt es sich die Benutzerdaten vorher zu exportieren.

Nachdem das Projekt auf das Zielsystem übertragen wurde, kann die Benutzerinformation wieder importiert werden. Somit ist sichergestellt, dass keine Information verloren geht.

Folgende Parameter sind davon betroffen:

- Full name
- Language
- Measurement system
- Additional data

id_Mgr_PasswordChangeReq

Ändert ein Administrator das Passwort eines Benutzers muss dieser sein Passwort beim nächsten Login ändern. Solange der Benutzer das Passwort nicht ändert hat dieser keine Zugriffsrechte. Um dieses Verhalten zu ermöglichen muss dieser Parameter auf *TRUE* gesetzt werden.

id_Mgr_PasswordHistory

Der Parameter gibt an wieviele Passwörter für jeden User gespeichert werden und nicht wiederverwendet werden dürfen. Der Wert darf zwischen 0 und 5 liegen. Bei 0 ist die Funktion ausgeschaltet.

Import Modi

Mithilfe von [MpUserXManagerUI](#) ist es möglich Benutzerdaten zu importieren bzw. zu exportieren.

In der Konfiguration kann unter dem Abschnitt *User Import Mode* und *Role Import Mode* definiert werden, wie die Daten importiert werden sollen. Dabei gibt es 5 verschiedene Modi:

- **Skip (Do not import):** Es werden keine Daten importiert
- **Ignore Existing Items:** Es werden alle neuen Daten (Benutzer/Rollen) importiert. Bereits bestehende Daten werden nicht überschrieben.
- **Overwrite Existing Items:** Es werden alle Daten importiert. Bereits bestehende Daten werden überschrieben. Default Benutzer, welche in der Konfiguration MpUserX angelegt wurden, werden dabei auch überschrieben und sind somit nicht mehr in der Konfiguration als Default Benutzer vorhanden.
- **Overwrite Items Only:** Es werden alle bereits existierenden Daten überschrieben. Neue Daten werden nicht importiert. Default Benutzer, welche in der Konfiguration MpUserX angelegt wurden, werden dabei auch überschrieben und sind somit nicht mehr in der Konfiguration als Default Benutzer vorhanden.
- **Remove Existing Items:** Es werden alle Daten importiert. Bereits vorhandene Daten, welche aber nicht in der importierten Datei vorkommen, werden gelöscht.

Wird eine Datei importiert, welcher bestehende Benutzer/Rollen überschreibt, dann kann dieser Vorgang abgefragt werden. Diese Abfrage erfolgt in Form eines Dialogfensters. Es wird auch angezeigt wieviele Dateien sich in einem Konflikt befinden. Der Überschreibvorgang kann auch für alle Dateien bestätigt werden.

Diese Option wird in der Struktur [MpUserXMgrUISetupType](#) über die Parameter "ConfirmOverwriteUser" und "ConfirmOverwriteRole" aktiviert.

The screenshot shows a dialog box titled "Import: Confirm user". It contains the following elements:

- A text input field labeled "Name" with the placeholder text "NAME OF USER/ROLE".
- A text input field labeled "Number of conflicts" with the value "0".
- A checkbox labeled "Apply for all" which is currently unchecked.
- At the bottom, there are two buttons: one with a red "X" icon (cancel) and one with a green checkmark icon (confirm).

2.2.1 Rechte und Levels

In diesem Abschnitt wird erklärt, wie die Rechte und Levels in der Konfiguration definiert werden können.

Die Rechte und Levels können nur innerhalb einer [Application Session](#) verwendet werden.



Levels

Das Level gibt Auskunft über die Zugriffsberechtigungen auf die Maschine. Je höher das Level, desto mehr Berechtigungen besitzt ein Benutzer. Das Level ist in der Benutzergruppe festgelegt. Jeder Benutzer aus einer Gruppe besitzt dasselbe Level. Wenn gewünscht, können die verschiedenen Funktionen der Maschine und der Visualisierung direkt über das Level gesperrt oder freigegeben werden. Folgende Tabelle dient als Beispiel:

Benutzergruppe	Level	Adminrechte	Kommentar
Maschinenbesitzer	100	ja	Kann die Maschine ohne Einschränkungen bedienen und auch die Benutzer verwalten
Prozesstechniker	70	nein	Besitzt hohe Zugriffsrechte, kann jedoch keine Benutzer verwalten, da er keine Adminrechte besitzt
Schichtführer	50	ja	Der Schichtführer kann neue Benutzer anlegen oder editieren. Jedoch nur, falls diese ein niedrigeres Level besitzen
Maschinenbediener	30	nein	Arbeitet täglich mit der Maschine
Wartungstechniker	30	nein	Bedient die Maschine während der Wartung
Besucher	3	nein	Gruppe, die lesenden Zugriff auf die Visualisierung hat, jedoch keine weiteren Funktionen nutzen kann

Falls es mehrere Gruppen auf demselben Level gibt, können sich diese durch verschiedene Rechte unterscheiden.

Funktionen und Rechte

Die Maschine, bei der das Benutzermanagement implementiert wird, besitzt unterschiedliche Funktionen. Dazu können Rezeptverwaltung, Prozesssteuerung, Alarmverwaltung und/oder das Ändern der Prozessparameter gehören. Auf diese (und alle anderen) Funktionen können unterschiedliche Rechte vergeben werden:

- **Keine Rechte:** Der Benutzer hat für diese Funktion keine Rechte
- **Lesezugriff:** Der Benutzer kann Daten einsehen, aber nicht bearbeiten
- **Start-/Stopzugriff:** Der Benutzer kann eine Funktion Starten und Stoppen, jedoch keinen weiteren Einfluss auf den Prozess nehmen (z.B. Ändern der Prozessparameter)
- **Voller Zugriff:** Der Benutzer hat Lese- und Schreibrechte für eine bestimmte Funktion
- **Undefiniert:** Falls kein spezielles Recht definiert wurde. Der Benutzer hat auch hier keine Rechte für die unterschiedlichen Funktionen

Auf jede Funktion kommt eines dieser Rechte. Die Funktionen können festgelegt werden. Beim obigen Beispiel gibt es zwei Gruppen mit demselben Level. Diese unterscheiden sich aber durch ihre Rechte auf unterschiedliche Funktionen:

Funktion	Maschinenbediener	Wartungstechniker
Automatische Prozesssteuerung	Start-/Stopzugriff	Keine Rechte
Rezeptverwaltung	Keine Rechte	Keine Rechte
Öffnen der Sicherheitstüren	Keine Rechte	Voller Zugriff
Manuelle Prozesssteuerung	Start-/Stopzugriff	Voller Zugriff

Die Funktionen sind Teil der Maschinensoftware und werden nicht über mapp definiert. In der Konfiguration kann beschrieben werden, welches Recht für welche Maschinenfunktion verwendet wird:

User group	Name	User level	Administrator	Access rights
User Group: Operator	Operator	100	TRUE	Access rights
Access right 1	Activate			Right to use the "Automatic process control" function of the machine
Access right 2	None			Right to use the "Recipe management" function of the machine
Access right 3	None			Right to open the safety doors of the machine
Access right 4	Activate			Right to use the "Manual process control" function of the machine
Access right 5	Unassigned			
User Group: Maintenance	Maintenance	10	FALSE	Access rights
Access right 1	None			Right to use the "Automatic process control" function of the machine
Access right 2	None			Right to use the "Recipe management" function of the machine
Access right 3	Full			Right to open the safety doors of the machine
Access right 4	Full			Right to use the "Manual process control" function of the machine
Access right 5	Unassigned			

2.3 MpUserXSign Konfiguration

Wie die Konfiguration zur Laufzeit angepasst werden kann, wird hier im Abschnitt "Konfiguration zur Laufzeit ändern" erklärt.

In der Konfiguration können Einstellungen für die Signatur festgelegt werden.

"User Level" bestimmt welches Benutzerlevel mindestens verwendet werden muss, um die Signatur zu starten.

Im Abschnitt "Actions" können die einzelnen Aktionen festgelegt werden. Pro Aktion muss eine eindeutige ID über "ActionID" angegeben werden. "UserLevel" bestimmt welches Benutzerlevel mindestens für diese Aktion verwendet werden muss. Für mehr Information siehe [Elektronische Signatur](#).

2.4 MpUserXMapping Konfiguration

Config_3.mpuserxmapping [mappServices Configuration]

MpLink: gUserXMapping

Local user role	Server group
1 Everyone	Guests
2 Operator	Machine Operator
3 Administrator	Admin
4	

Die MpUserXMapping Konfiguration ist eine Tabelle über welche die Rollen in Automation Studio mit denen des Directory Servers verbunden werden.

Über "Local user role" werden die in Automation Studio erstellten Rollen angegeben. "Server group" definiert die Rollen des Active Directory Servers.

Für mehr Information zum mappen der Rollen siehe Abschnitt [Mappen der Rollen](#).

3 Anwendungsfälle

In diesem Abschnitt werden mögliche Anwendungsfälle der mapp UserX Komponenten beschrieben.

3.1 Anwendungsfall 1: Application Session mit Visualisierung verbinden

Anforderung

Eine Visualisierung einer Maschine verwendet OPC UA um mit den Prozessvariablen der Steuerung zu kommunizieren. In der Applikation sind 2 Benutzer und 2 Rollen im Benutzerrollensystem von Automation Studio angelegt. Die Zugriffsrechte auf Prozessvariablen sind in der OPC UA Konfiguration eingestellt.

Lösung

Komponentenliste

- **MpUserXLogin** (Eigener MpLink): Anmelden / Abmelden eines Benutzers
- **MpUserXLoginUI** (MpLink von MpUserXLogin): Anmelden / Abmelden eines Benutzers über die Visualisierung

Verbindungsschema



Konfiguration

Um den Benutzern deren Rollen zu vergeben wird das Benutzer-Rollen-System von Automation Studio verwendet:

Das Bild zeigt die Konfiguration des Benutzer-Rollen-Systems in Automation Studio. Links ist die Projektstruktur zu sehen, rechts die Konfigurationstabelle.

Name	Value
Administrator	
Role ID	1
Everyone	
Role ID	2

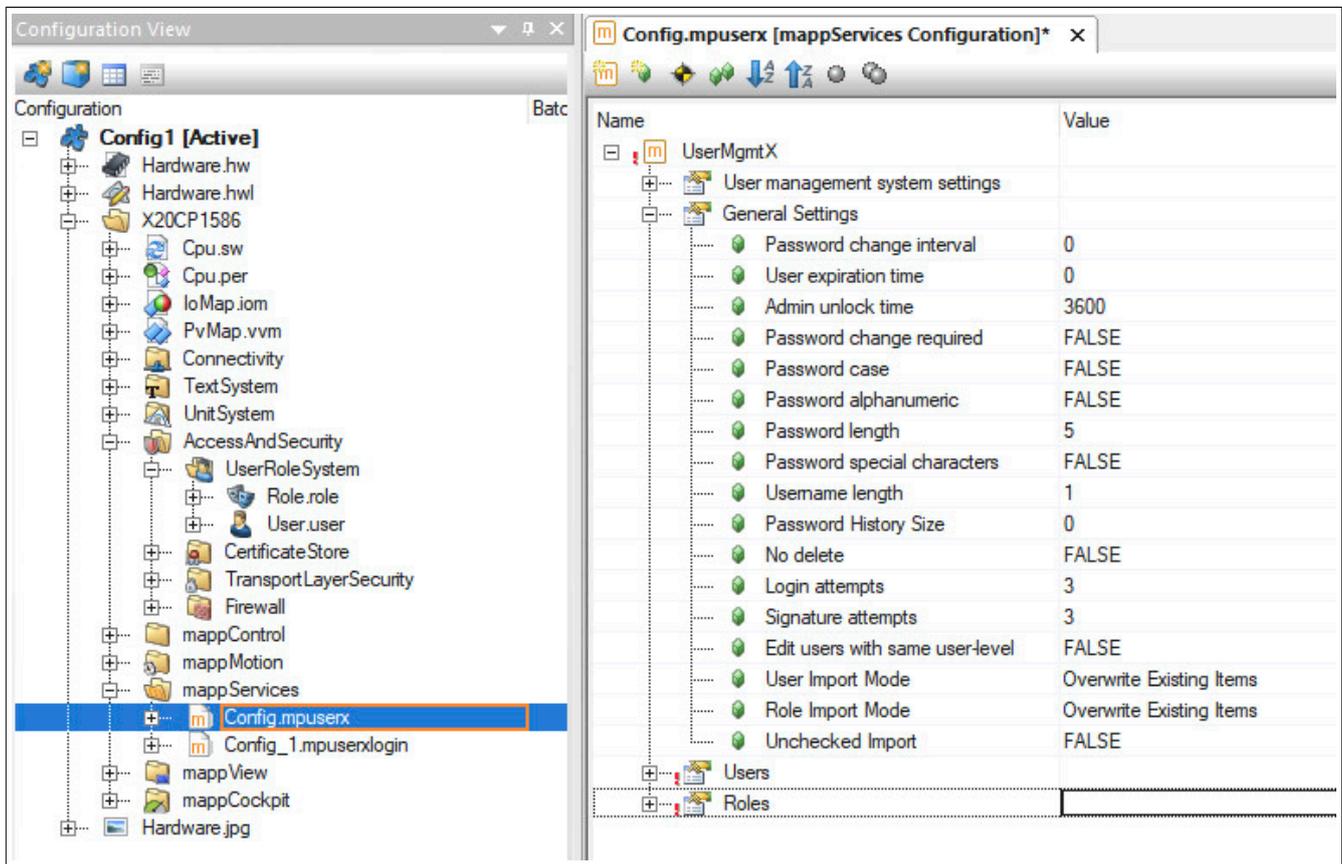
Um Benutzer anzulegen und Passwörter zu vergeben wird die Benutzerkonfiguration von Automation Studio verwendet. Die Benutzer-ID wird nur intern verwendet und sollte nicht verändert werden.

Das Bild zeigt die Benutzerkonfiguration in Automation Studio. Links ist die Projektstruktur zu sehen, rechts die Konfigurationstabelle.

Name	Value	Ur
Anonymous		
User ID	1	
Password	*****	
Roles		
Assigned Role [1]	Everyone	
Assigned Role [2]		
Default		
User ID	2	
Password	*****	
Roles		
Assigned Role [1]	Administrators	
Assigned Role [2]	Everyone	
Assigned Role [3]		

Es muss die Konfiguration MpUserX und MpUserXLogin eingefügt werden.

In der mapp Konfiguration MpUserX können, falls benötigt, weitere Eigenschaften für Benutzergruppen eingestellt werden. In unserem Beispiel wurden keine Änderungen vorgenommen.



Verwendung der mapp Komponenten

Die Standard-Benutzerkonfiguration kann direkt im AS-Projekt (Konfigurationssystem AS 4.2) bearbeitet werden.

Die Komponente **MpUserXLogin** erzeugt einen Anmeldepunkt, an dem sich ein Benutzer mit seinem Benutzernamen und seinem Passwort anmeldet.

Der **MpUserXLoginUI** liefert die entsprechenden Datenpunkte für eine VC4 Visualisierung (z.B. Anmelden, Passwortänderung, usw.)

3.2 Anwendungsfall 2: Editieren von Benutzern mithilfe des User managements

Anforderung

In einer Applikation sind 2 Benutzer und 2 Rollen über das Benutzer-Rollen-System von Automation Studio angelegt. Jeder Benutzer soll andere Rechte besitzen. Das heißt für manche Benutzer sind Funktionen gesperrt, wie zum Beispiel das ändern von Rezeptdaten und für andere Benutzer sind diese Funktionen freigeschaltet.

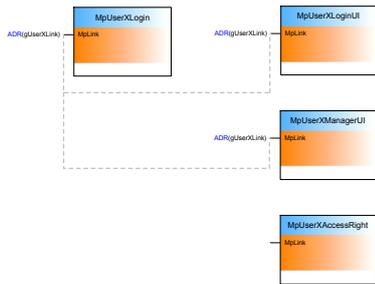
Es soll möglich sein während der Laufzeit neue Benutzer hinzuzufügen und deren Rechte zu vergeben. Das hinzufügen von neuen Benutzern und deren Rechtevergabe soll nur für den Administrator möglich sein.

Lösung

Komponentenliste

- **MpUserXLogin** (Eigener MpLink): Anmelden / Abmelden eines Benutzers
- **MpUserXLoginUI** (MpLink von MpUserLogin): Anmelden / Abmelden eines Benutzers über die Visualisierung
- **MpUserXManagerUI** (MpLink von MpUserLogin): Verwaltet die Benutzer, sowie deren Rollen und Rechten
- **MpUserXAccessRight**: Prüfung von Benutzerrechte

Verbindungsschema



Konfiguration

Benutzer und deren Rollen können im Benutzer-Rollen-System von Automation Studio eingestellt werden. Die Rollen-ID wird nur intern benutzt und sollte nicht verändert werden.

Name	Value	Unit	Description
Administrator			
Role ID	1		
Description	Role for..		
Operator			
Role ID	2		
Description	Role for..		

Um Passwörter und Rollen den Benutzern zuzuweisen wird die Benutzerkonfiguration von Automation Studio verwendet. Die Benutzer-ID wird nur intern verwendet und sollte nicht verändert werden.

Name	Value	Unit
Admin		
User ID	1	
Password	*****	
Roles		
Assigned Role [1]	Administrators	
Assigned Role [2]		
Default		
User ID	2	
Password	*****	
Roles		
Assigned Role [1]	Operator	
Assigned Role [2]		

Es werden die Konfigurationen MpUserX und MpUserXLogin eingefügt.

In der MpUserX Konfiguration können, falls benötigt, zusätzliche Parameter unter "Roles" eingestellt werden.

Über die zusätzliche Funktion, welche das mapp Benutzermanagement bietet, können nun für einzelne Rollen Rechte definiert werden. Die Rechte sind an bestimmte Funktionen der Maschine verknüpft. Die Rechteverwaltung kann folgendermaßen aussehen:

Funktion	Operator	Administrator
Bewegung der Standard Achse	1 None	Full (Bewegung durchführen + ändern von Achsparametern)
Quittieren kritischer Alarme	2 None	Full (Ansicht der Alarmliste + Quittieren von Alarmen)
Verändern der Rezeptdaten	3 View (Ansicht der Rezeptliste)	Full (Ansicht der Rezeptliste + Laden von Rezepten + Speichern von Rezepten)

User groups	
User Group: Administrator	
Name	Administrator
Level	100
Administrator	TRUE
Rights	
Access right 1	Full
Access right 2	Full
Access right 3	Full
Access right 4	Undefined
User Group: Operator	
Name	Operator
Level	10
Administrator	FALSE
Rights	
Access right 1	None
Access right 2	None
Access right 3	View
Access right 4	Undefined

Verwendung der mapp Komponenten

Es werden die Komponenten **MpUserXLogin**, **MpUserXLoginUI**, **MpUserXManagerUI** und **MpUserXAccessRight** eingefügt. Danach werden diese, wie im Punkt Verbindungsschema, miteinander verbunden.

Anschließend werden die Komponenten parametrisiert.

Der **MpUserXLogin** erzeugt einen Anmeldepunkt, an dem sich ein Benutzer mit seinem Benutzernamen und seinem Passwort anmeldet.

Loggt sich nun ein Benutzer ein, kontrolliert das mapp Benutzermanagement welche Rechte dieser besitzt und setzt dementsprechend bestimmte Funktionen frei. Handelt es sich um einen Administrator ist dieser in der Lage Rechte zu editieren. Mithilfe von **MpUserXAccessRight** kann geprüft werden, welches Recht der aktuell eingeloggte Benutzer für eine bestimmte Funktion besitzt.

Es ist also möglich sich mit dem im Benutzerrollensystem definierten Benutzer anzumelden. Während der Laufzeit kann über die Komponente **MpUserXManagerUI** weitere Benutzer erstellt werden.

4 Bibliotheken

4.1 Funktionsbausteine

In diesem Abschnitt werden alle Funktionen und Funktionsbausteine aufgelistet, welche über das **User management** oder innerhalb einer **Application Session** oder **Visualization Session** verwendet werden können.

Name	Beschreibung	User management	Application Session	Visualization Session
MpUserXLogin	Dieser Funktionsblock ermöglicht das Ein- und Ausloggen von Benutzern (über die Applikation) und er ist für jedes Benutzermanagement notwendig		✓	
MpUserXManagerUI	Dieser Funktionsblock ermöglicht Zugang zum kompletten Benutzermanagement über die Visualisierung	✓		
MpUserXLoginUI	Dieser Funktionsblock ermöglicht das Ein- und Ausloggen von Benutzern (über die Visualisierung)		✓	
MpUserXHasRole	Diese Funktion prüft ob der aktuelle eingeloggte Benutzer eine bestimmte Rolle besitzt		✓	
MpUserXLevel	Diese Funktion gibt das Level des aktuell eingeloggten Benutzers zurück		✓	

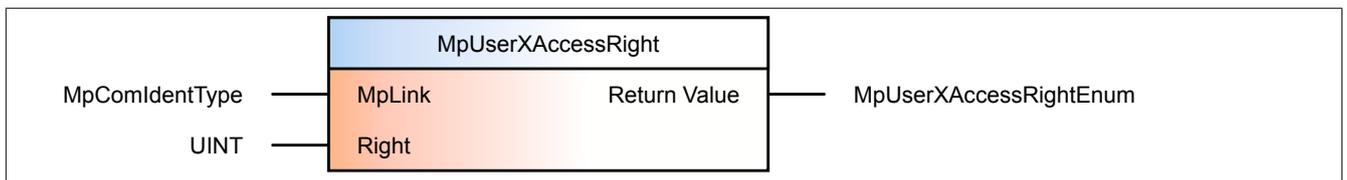
Name	Beschreibung	User management	Application Session	Visualization Session
MpUserXAccessRight	Diese Funktion gibt die Rechte des aktuell eingeloggten Benutzers zurück		✓	
MpUserXConfig	Dieser Funktionsblock speichert/lädt die Konfiguration des User managements	✓		
MpUserXLoginConfig	Dieser Funktionsblock speichert/lädt die Konfiguration des Login- und Logout-verhaltens		✓	
MpUserXSignature	Dieser Funktionsblock prüft, ob das Zeichnen anhand einer elektronischen Signatur erfolgreich war		✓	✓
MpUserXSignatureUI	Dieser Funktionsblock ermöglicht eine elektronische Signatur über die Visualisierung		✓	✓
MpUserXGetData	Mithilfe dieser Funktion können von einem Benutzer zusätzliche Informationen abgefragt werden.		✓	✓
MpUserXMappingConfig	Mithilfe dieses Funktionsbausteins kann das Mapping der Rollen zur Laufzeit geändert und angepasst werden.	✓		
MpUserXServerConfig	Mithilfe dieses Funktionsbausteins können die Verbindungs-Parameter zum Active Directory Server zur Laufzeit angepasst werden.	✓		

4.1.1 MpUserXAccessRight

Diese Funktion gibt die Rechte des aktuell eingeloggten Benutzers zurück.

Diese Funktion kann nur innerhalb einer Application Session verwendet werden.

Funktion



Schnittstelle

I/O	Parameter	Datentyp	Beschreibung
IN	MpLink	MpComIdentType	Verbindung zur mapp Konfiguration (MpLink einer MpUserXLogin Konfiguration).
IN	Right	UINT	Funktion, für welche das Recht überprüft werden soll.
OUT	ReturnValue	MpUserXAccess-RightEnum	Benutzerrecht für die hinterfragte Funktion.

mapp Konzept

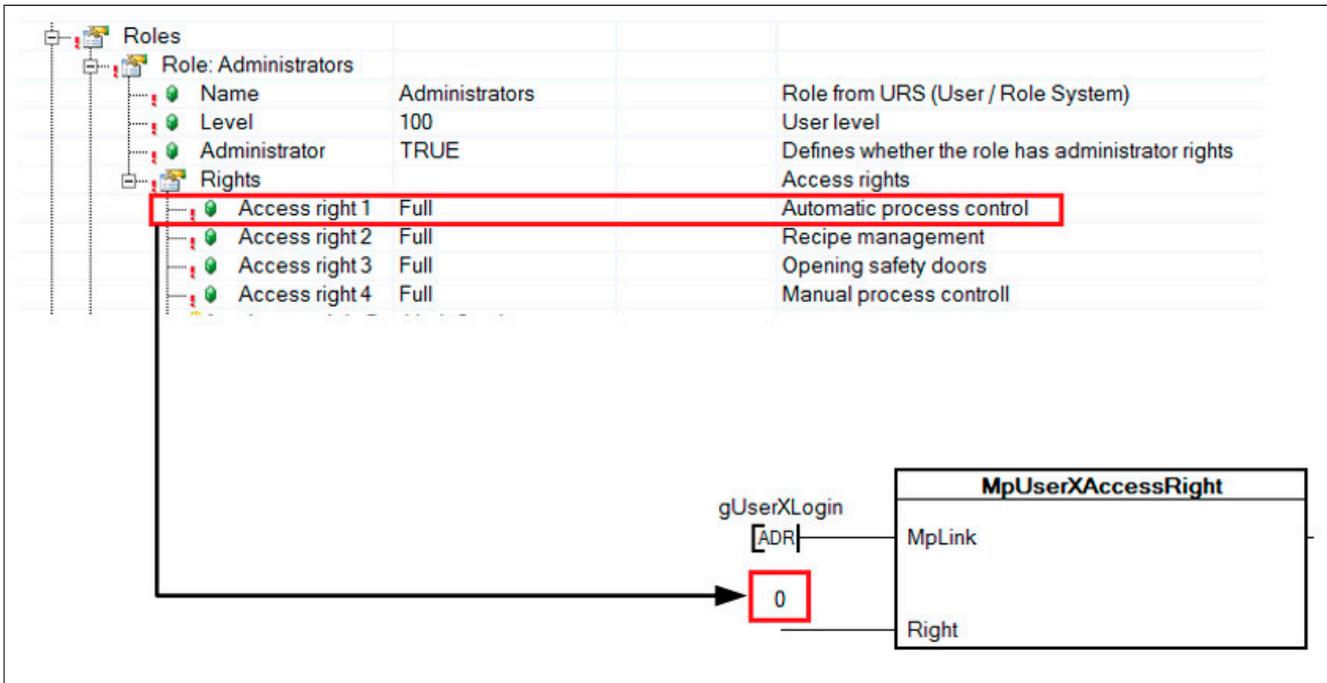
Im Abschnitt mapp Komponenten wird erklärt wie die mapp Komponenten aufgebaut sind. Außerdem wird auf die richtige Verwendung von mapp Komponenten hingewiesen (z.B. beim Download).

4.1.1.1 Beschreibung

Für diese Komponente wird derselbe MpLink verwendet, wie für [MpUserXLogin](#).

Über den Parameter "Right" wird die Funktion (siehe Rechte, Levels und Funktionen) angegeben, für welche das Recht überprüft werden soll.

Es ist zu beachten, dass der Eingang "Right" mit dem Index 0 beginnt, aber in der [MpUserX Konfiguration](#) die Rechtevergabe ab Index 1 beginnt. Soll also das Recht "Automatic process control" geprüft werden, welches in der Konfiguration Recht 1 ist, so muss am Parameter "Right" zur Überprüfung der Index 0 angegeben werden.



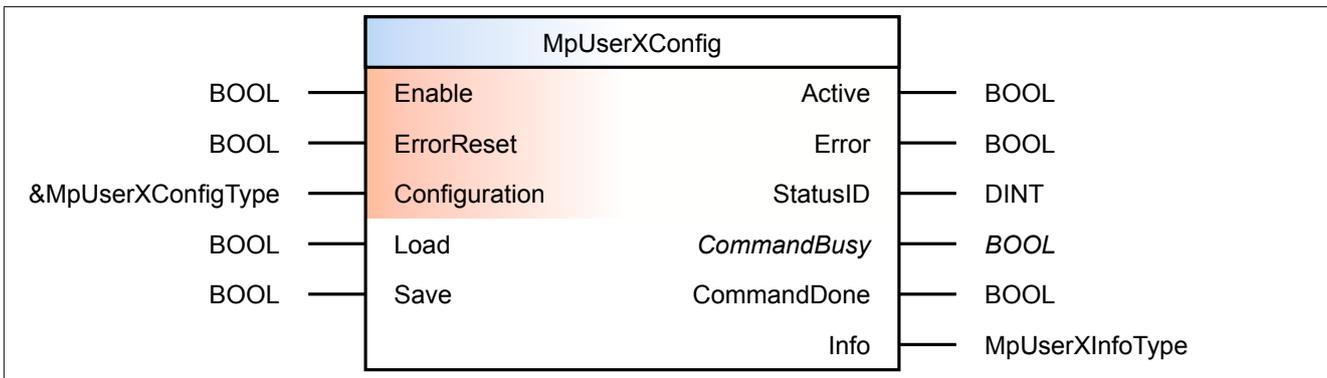
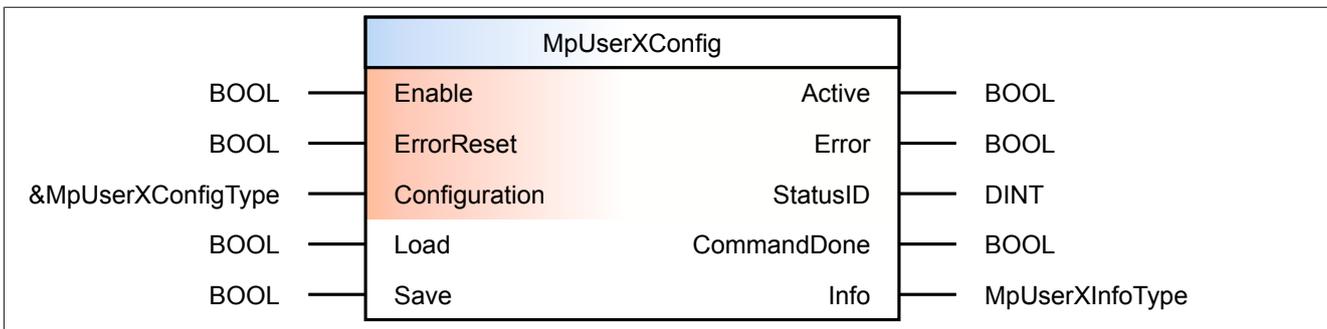
4.1.2 MpUserXConfig

Dieser Funktionsblock speichert/lädt die Konfiguration des Benutzermanagements.

Dieser Funktionsbaustein kann nur innerhalb des User management verwendet werden.

Funktionsblock

Optionale Parameter



Schnittstelle

I/O	Parameter	Datentyp	Beschreibung
IN	Enable	BOOL	FB ist aktiv solange der Eingang gesetzt ist.
IN	ErrorReset	BOOL	Dient zum Zurücksetzen von Funktionsbaustein Fehlern.
IN	Configuration	Zeiger auf MpUserXConfigType	Struktur, in der die Konfiguration festgelegt wird.
IN	Load	BOOL	Lädt die Konfiguration der Komponente.
IN	Save	BOOL	Speichert die Konfiguration der Komponente.
OUT	Active	BOOL	FB ist aktiv.

I/O	Parameter	Datentyp	Beschreibung
OUT	Error	BOOL	Fehler bei Abarbeitung aufgetreten.
OUT	StatusID	DINT	Statusinformation.
OUT	CommandBusy	BOOL	Funktionsblock führt aktuell einen Befehl aus.
OUT	CommandDone	BOOL	Abarbeitung erfolgreich. FB ist fertig.
OUT	Info	MpUserXInfoType	Zusätzliche Komponenteninformationen.

mapp Konzept

Im Abschnitt mapp Komponenten wird erklärt wie die mapp Komponenten aufgebaut sind. Außerdem wird auf die richtige Verwendung von mapp Komponenten hingewiesen (z.B. beim Download).

Weiters wird im Abschnitt Konfigurieren statt Programmieren erklärt, wie man mit der Konfiguration arbeitet.

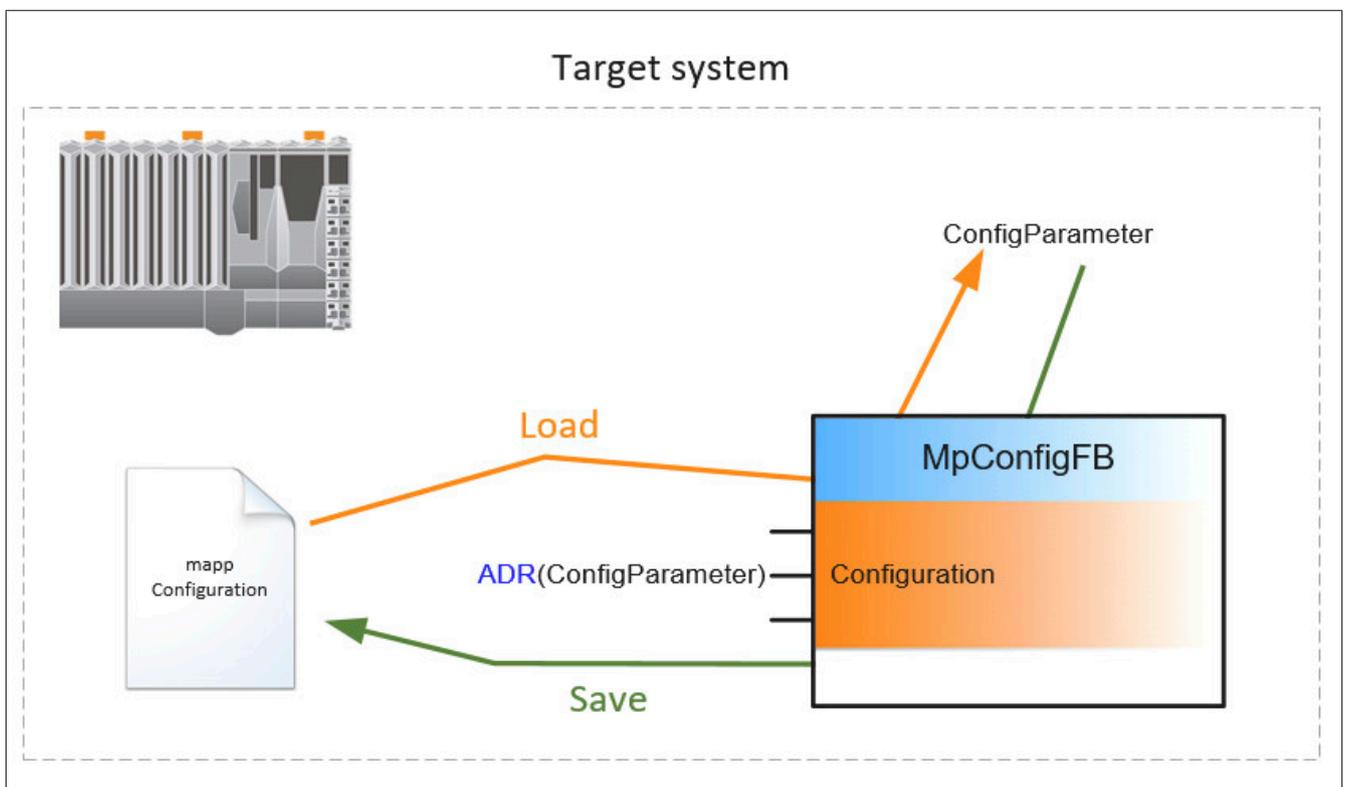
4.1.2.1 Beschreibung

Mithilfe des Funktionsbausteins sind Änderungen der Konfiguration zur Laufzeit möglich.

Der Parameter "Configuration" ermöglicht es die Konfiguration, bzw. einen Teil der Konfiguration zu bearbeiten.

Der Befehl "Load" wird verwendet, um die aktuell konfigurierten Parameter auszulesen. Dabei wird die aktuell aktive Konfiguration der Steuerung in die PV am Eingang "Configuration" geschrieben. Die Daten in der Struktur werden dabei überschrieben. Danach können die Werte beliebig editiert werden.

Über den Befehl "Save" werden die Werte aus der "Configuration" Struktur auf die aktive Konfiguration übernommen und, wenn möglich, direkt verwendet.



Um die Konfigurations-Parameter in der Applikation korrekt anzupassen, wird folgendermaßen vorgegangen:

1. Befehl "Load = TRUE"

Die aktuelle Konfiguration wird vom Zielsystem in die Konfigurations-Struktur geladen. Der Vorgang ist mit "CommandDone = TRUE" abgeschlossen.

2. Befehl "Load = FALSE"

Rücksetzen des "Load" Befehls. Der Vorgang ist mit "CommandDone = TRUE" abgeschlossen.

3. Parametrieren der Konfigurationsstruktur

Die Parameter der Konfigurationsstruktur können nun wie gewünscht angepasst werden.

4. Befehl "Save = TRUE" setzen

Die geänderte Konfigurationsstruktur wird auf das Zielsystem übertragen. Der Vorgang ist mit "CommandDone = TRUE" abgeschlossen.

5. Befehl "Save = FALSE" setzen

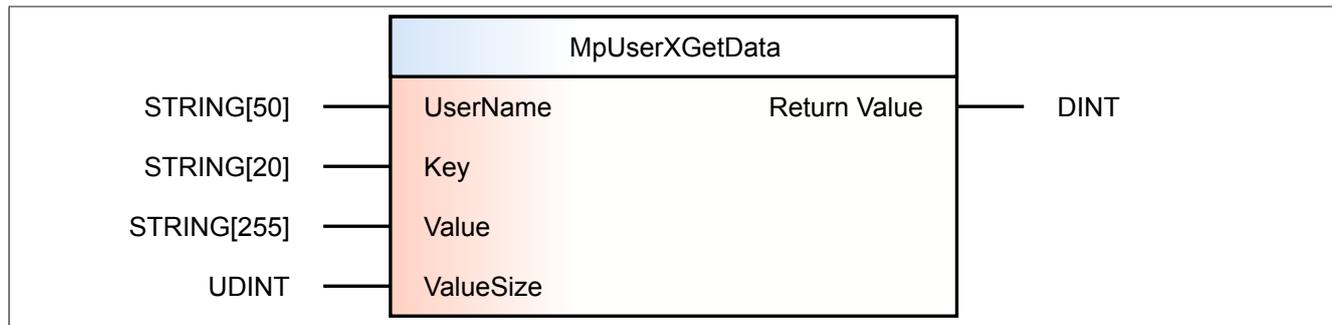
Rücksetzen des "Save" Befehls. Der Vorgang ist mit "CommandDone = FALSE" abgeschlossen. Um zu kontrollieren ob die Parameter erfolgreich geändert worden sind, können die Schritte 1 und 2 erneut ausgeführt werden.

4.1.3 MpUserXGetData

Mithilfe dieser Funktion können von einem Benutzer zusätzliche Informationen abgefragt werden.

Diese Funktion kann innerhalb einer **Application Session** sowie **Visualization Session** verwendet werden.

Funktion



Schnittstelle

I/O	Parameter	Datentyp	Beschreibung
IN	UserName	STRING[50]	Definiert den Benutzernamen für welchen man zusätzliche Information abfragen möchte.
IN	Key	SRING[20]	Durch "Key" wird festgelegt, welche Information man vom Benutzer abfragen möchte
IN	Value	SRING[255]	Hier wird die abgefragte Information angezeigt.
IN	ValueSize	UDINT	Definiert die maximale Buffergröße für den Schreibvorgang an "Value".
OUT	ReturnValue	DINT	Rückgabewert der Funktion.

mapp Konzept

Im Abschnitt mapp Komponenten wird erklärt wie die mapp Komponenten aufgebaut sind. Außerdem wird auf die richtige Verwendung von mapp Komponenten hingewiesen (z.B. beim Download).

4.1.3.1 Beschreibung

Über den Parameter "UserName" wird der Name eines Benutzers angegeben, für welchen man zusätzliche Information abfragen möchte.

"Key" bestimmt welche Information abgefragt werden soll. Folgende Schlüsselwörter sind möglich:

- **Language:** Liefert die bevorzugte Sprache des Benutzers.
- **DisplayUnit:** Liefert das bevorzugte Einheitensystem des Benutzers.
- **eMail:** Wird "eMail" verwendet, wird die E-Mail-Adresse des Benutzers angezeigt.
- **PhoneNumber:** Durch "telNr" wird die Telefonnummer des Benutzers angezeigt.
- **any:** Durch "any" kann eine, in der **MpUserX Konfiguration** festgelegte, zusätzliche Information angezeigt werden, wie zum Beispiel "Team".

Am Parameter "Value" wird die abgefragte Information angezeigt.

Wichtig hierbei ist, dass über "ValueSize" die Buffergröße für den Schreibvorgang an "Value" festgelegt wird.

"Return value" liefert den Rückgabewert der Funktion. Dabei können folgende Rückgabewerte auftreten:

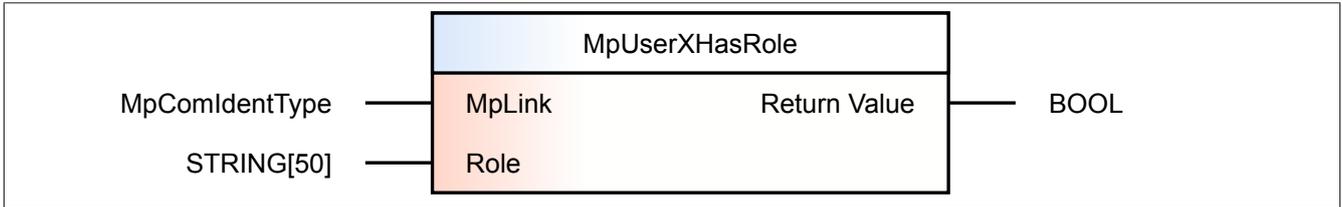
- mpUSERX_ERR_NO_USERNAME
- mpUSERX_ERR_EMPTY_KEY
- mpUSERX_ERR_INFO_NOT_FOUND
- mpUSERX_ERR_BUFFER_TOO_SMALL
- mpUSERX_WRN_KEY_DUPLICATE

4.1.4 MpUserXHasRole

Diese Funktion prüft ob der aktuelle eingeloggte Benutzer eine bestimmte Rolle besitzt.

Diese Funktion kann nur innerhalb einer Application Session verwendet werden.

Funktion



Schnittstelle

I/O	Parameter	Datentyp	Beschreibung
IN	MpLink	MpComIdentType	Verbindung zur mapp Konfiguration (MpLink einer MpUserXLogin Konfiguration).
IN	Role	STRING[50]	Rolle, welche abgefragt werden soll.
IN	ReturnValue	BOOL	Zeigt an, ob der eingeloggte Benutzer die abgefragte Rolle besitzt.

mapp Konzept

Im Abschnitt mapp Komponenten wird erklärt wie die mapp Komponenten aufgebaut sind. Außerdem wird auf die richtige Verwendung von mapp Komponenten hingewiesen (z.B. beim Download).

4.1.4.1 Beschreibung

Für diese Funktion wird derselbe MpLink verwendet wie für [MpUserXLogin](#).

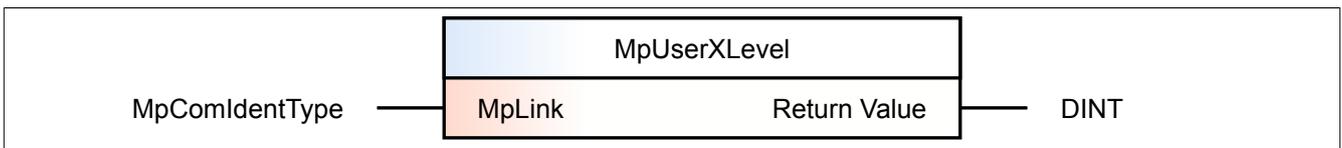
Mithilfe dieser Funktion wird überprüft, ob der aktuell eingeloggte Benutzer eine bestimmte Rolle besitzt. Der Parameter "Role" definiert auf welche Rolle geprüft wird, der Rückgabewert zeigt an ob der eingeloggte Benutzer die Rolle hat.

4.1.5 MpUserXLevel

Diese Funktion gibt das Level des aktuell eingeloggten Benutzers zurück.

Diese Funktion kann nur innerhalb einer Application Session verwendet werden.

Funktion



Schnittstelle

I/O	Parameter	Datentyp	Beschreibung
IN	MpLink	MpComIdentType	Verbindung zur mapp Konfiguration (MpLink einer MpUserXLogin Konfiguration).
OUT	ReturnValue	DINT	Gibt das Level des eingeloggten Benutzers zurück.

mapp Konzept

Im Abschnitt mapp Komponenten wird erklärt wie die mapp Komponenten aufgebaut sind. Außerdem wird auf die richtige Verwendung von mapp Komponenten hingewiesen (z.B. beim Download).

4.1.5.1 Beschreibung

Für diese Komponente wird derselbe MpLink verwendet, wie für [MpUserXLogin](#).

4.1.6 MpUserXLogin

Dieser Funktionsblock ermöglicht das Ein- und Ausloggen von Benutzern (über die Applikation) und er ist für jedes Benutzermanagement notwendig.

Dieser Funktionsbaustein kann nur innerhalb einer **Application Session** verwendet werden.



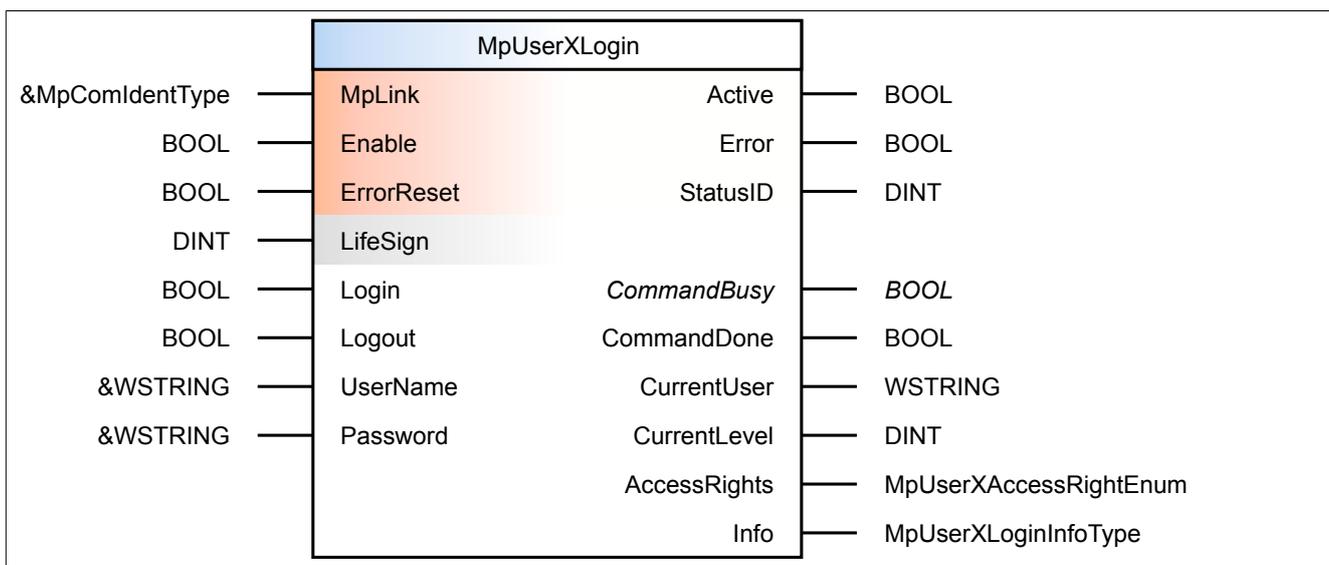
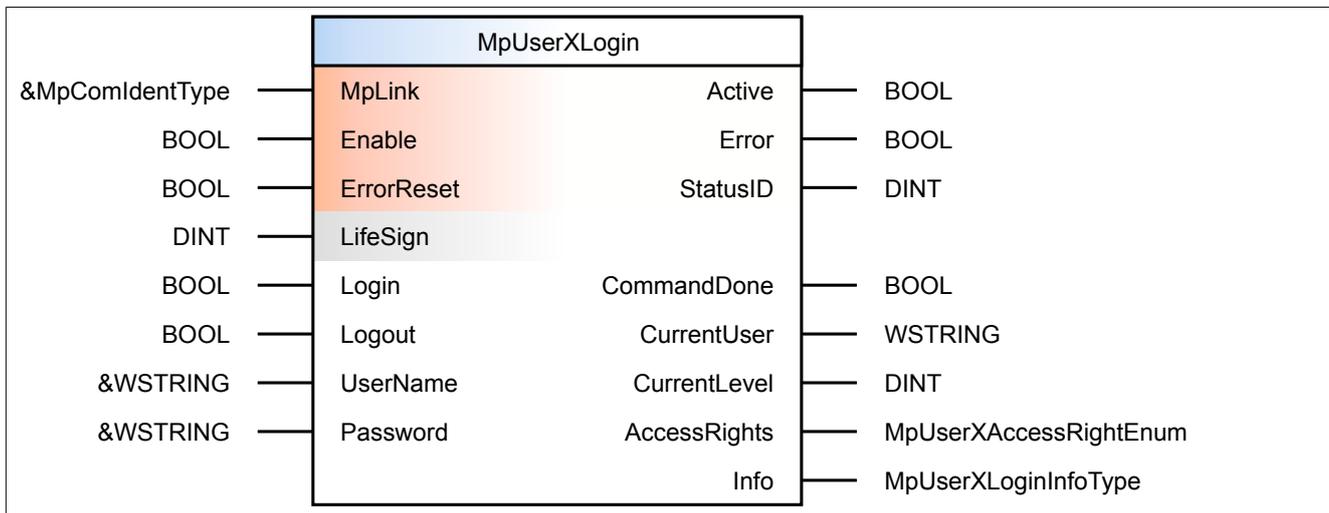
Auf unserem [B&R Tutorial Portal](#) sind Tutorials zum Thema mapp UserX ([GER/ENG](#)) zu finden. (Zugangsdaten)



Auf unserem [Youtube Kanal](#) sind viele Tutorials zum Thema zu finden.

Funktionsblock

Optionale Parameter



Schnittstelle

I/O	Parameter	Datentyp	Beschreibung
IN	MpLink	Zeiger auf Mp-ComIdentType	Verbindung zur mapp Konfiguration (MpLink einer MpUserXLogin Konfiguration).
IN	Enable	BOOL	FB ist aktiv solange der Eingang gesetzt ist.
IN	ErrorReset	BOOL	Dient zum Zurücksetzen von Funktionsbaustein Fehlern.
IN	LifeSign	DINT	Damit wird festgestellt, ob der eingeloggte Benutzer seit längerer Zeit inaktiv ist. Inaktivzeit wird bei Veränderung dieses Eingangs zurückgesetzt.
IN	Login	BOOL	Befehl, um sich einzuloggen.
IN	Logout	BOOL	Befehl, um sich auszuloggen.
IN	UserName	Zeiger auf WSTRING[50]	Benutzername, mit dem sich eingeloggt werden soll.
IN	Password	Zeiger auf WSTRING[50]	Passwort, mit dem sich eingeloggt werden soll.
OUT	Active	BOOL	FB ist aktiv.
OUT	Error	BOOL	Fehler bei Abarbeitung aufgetreten.
OUT	StatusID	DINT	Statusinformation.
OUT	CommandBusy	BOOL	Funktionsblock führt aktuell einen Befehl aus.
OUT	CommandDone	BOOL	Abarbeitung erfolgreich. FB ist fertig.

I/O	Parameter	Datentyp	Beschreibung
OUT	CurrentUser	STRING[50]	Benutzername des eingeloggten Benutzers.
OUT	CurrentLevel	DINT	Level des eingeloggten Benutzers.
OUT	AccessRights	ARRAY[0..19] von MpUserXAccess-RightEnum	Zugriffsrechte des eingeloggten Benutzers.
OUT	Info	MpUserXLoginInfoType	Zusätzliche Komponenteninformationen.

mapp Konzept

Im Abschnitt mapp Komponenten wird erklärt wie die mapp Komponenten aufgebaut sind. Außerdem wird auf die richtige Verwendung von mapp Komponenten hingewiesen (z.B. beim Download).

4.1.6.1 Beschreibung

Für diesen Funktionsbaustein wird der MpLink einer [MpUserLoginX Konfiguration](#) verwendet.

MpUserXLogin wird für die Verwaltung des Benutzermanagement verwendet. Mithilfe des Funktionsbausteins können Benutzer sich an-, und abmelden.

Mithilfe von "UserName" und "Password" kann der Benutzername und das Passwort angegeben werden. Bei der Verwendung des [Zentrales User management](#), kann der Benutzername auch mit der Angabe der Domain angegeben werden, wie beispielsweise "userPrincipalName@domain.local" oder "DOMAIN\SAMAccountName".

Durch "Login = TRUE" wird der Login-Vorgang gestartet. Es wird überprüft ob der Benutzer existiert und ob das korrekte Passwort dabei verwendet wurde. Existiert der Benutzer beispielsweise nicht, oder wurde das falsche Passwort verwendet, so wird ein [Fehler oder eine Warnung](#) am Funktionsbaustein angezeigt.



Für mehr Informationen zu einem Login-Vorgang siehe Anwendungsfall "[Einfache Benutzerverwaltung mit Benutzerrollensystem](#)".

Bei einem erfolgreichen Login-Vorgang wird der Name des eingeloggte Benutzer an "CurrentUser" angezeigt sowie dessen Benutzerlevel und Zugriffsrechte an "CurrentLevel" und "AccessRights". Es kann immer nur ein Benutzer pro MpUserXLogin-Instanz angemeldet sein.

Mithilfe des Eingangsparameters "LifeSign" wird überprüft ob der Benutzer noch aktiv ist.

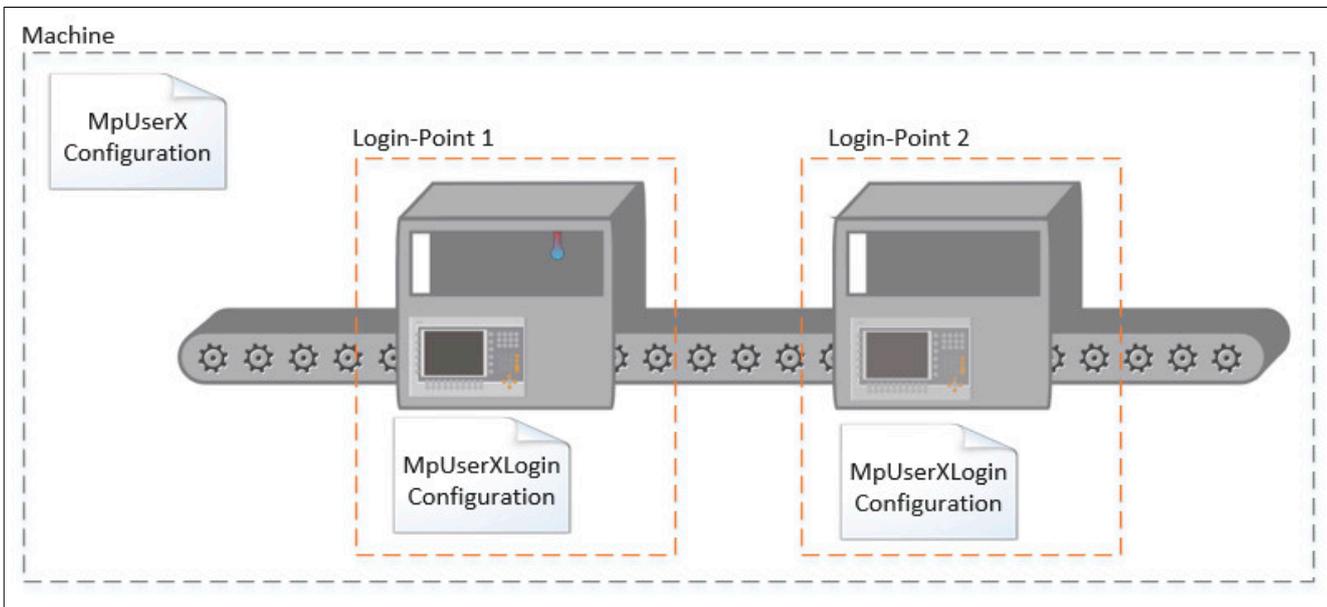
Damit sich der Benutzer ausloggen kann, wird der Eingang "Logout" verwendet.

Bei "CommandDone = TRUE" wurde das Kommando fertig ausgeführt. Um ein neues Kommando zu starten, muss der vorherige Befehl auf FALSE gesetzt werden. Der Funktionsbaustein ist nun bereit einen neuen Befehl auszuführen.

Über die [Info-Stuktur](#) des Funktionsbausteins sind weitere zusätzliche Diagnose-Informationen zu finden. Der Parameter "DaysUntilPasswordExpiration" gibt dabei an, wie viele Tage noch bis zum Passwort-Ablauf vorhanden sind. Bei "-1" wird die Passwort-Ablauf-Funktion nicht verwendet. Bei "0" und "Password change required = TRUE" in der [MpUserX Konfiguration](#) ist das Passwort bereits abgelaufen. Bei einem Wert ≥ 0 wird angezeigt, dass das Passwort ablaufen wird. Über den Ausgangsparameter StatusID an [MpUserXLogin](#) wird angezeigt, ob das Passwort bereits abgelaufen ist oder nicht.

Verwenden von mehreren MpUserXLogin-Instanzen

Pro Login-Punkt am Zielsystem kann eine MpUserXLogin-Instanz verwendet werden. Werden mehrere Login-Punkte benötigt, so kann mithilfe der MpComGroup Konfiguration eine Hierarchie erstellt werden. Für mehr Information wie eine Hierarchie erstellt wird siehe Anwendungsfall "[Alarmverwaltung für Druckermaschinenlinie](#)". Pro Login-Punkt gelten die gleichen Benutzerdaten, welche in der [MpUserX Konfiguration](#) festgelegt wurden. Ein Benutzer kann sich an mehreren Login-Punkten gleichzeitig anmelden. Wird ein Benutzer an einen der Loginpunkte gesperrt, da er beispielsweise für einen [Signatur-Vorgang](#) das Passwort zu oft falsch eingegeben hat, so wird der Benutzer automatisch an allen Loginpunkten abgemeldet und gesperrt.



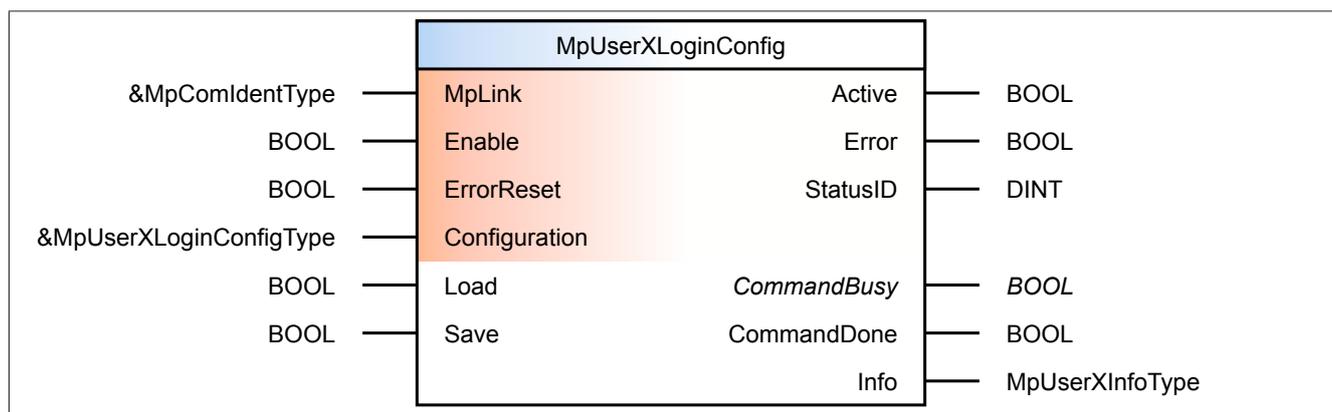
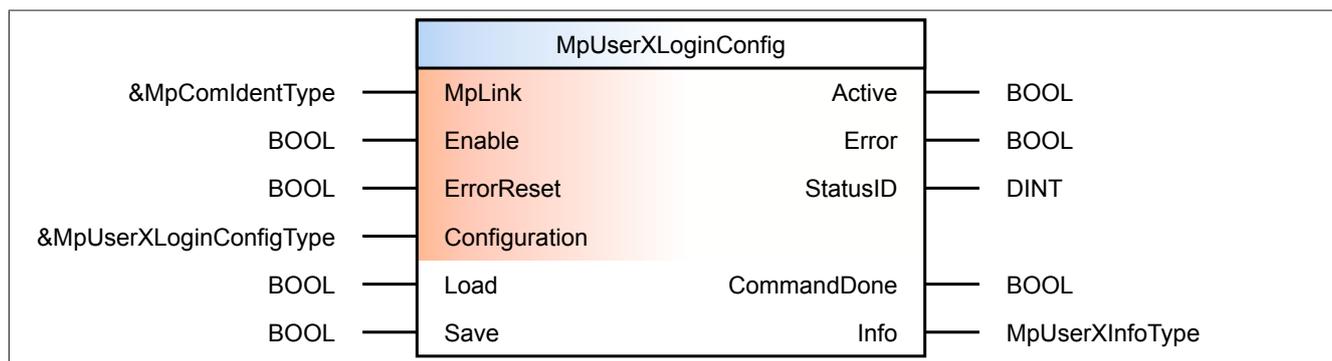
4.1.7 MpUserXLoginConfig

Dieser Funktionsblock speichert/lädt die Konfiguration des Login- und Logoutverhaltens.

Dieser Funktionsbaustein kann nur innerhalb einer Application Session verwendet werden.

Funktionsblock

Optionale Parameter



Schnittstelle

I/O	Parameter	Datentyp	Beschreibung
IN	MpLink	Zeiger auf MpComIdentType	Verbindung zur mapp Konfiguration (MpLink einer MpUserXLogin Konfiguration).
IN	Enable	BOOL	FB ist aktiv solange der Eingang gesetzt ist.
IN	ErrorReset	BOOL	Dient zum Zurücksetzen von Funktionsbaustein Fehlern.
IN	Configuration	Zeiger auf MpUserXLoginConfigType	Struktur, in der die Konfiguration festgelegt wird.
IN	Load	BOOL	Lädt die Konfiguration der Komponente.

I/O	Parameter	Datentyp	Beschreibung
IN	Save	BOOL	Speichert die Konfiguration der Komponente.
OUT	Active	BOOL	FB ist aktiv.
OUT	Error	BOOL	Fehler bei Abarbeitung aufgetreten.
OUT	StatusID	DINT	Statusinformation.
OUT	CommandBusy	BOOL	Funktionsblock führt aktuell einen Befehl aus.
OUT	CommandDone	BOOL	Abarbeitung erfolgreich. FB ist fertig.
OUT	Info	MpUserXInfoType	Zusätzliche Komponenteninformationen.

mapp Konzept

Im Abschnitt mapp Komponenten wird erklärt wie die mapp Komponenten aufgebaut sind. Außerdem wird auf die richtige Verwendung von mapp Komponenten hingewiesen (z.B. beim Download).

Weiters wird im Abschnitt Konfigurieren statt Programmieren erklärt, wie man mit der Konfiguration arbeitet.

4.1.7.1 Beschreibung

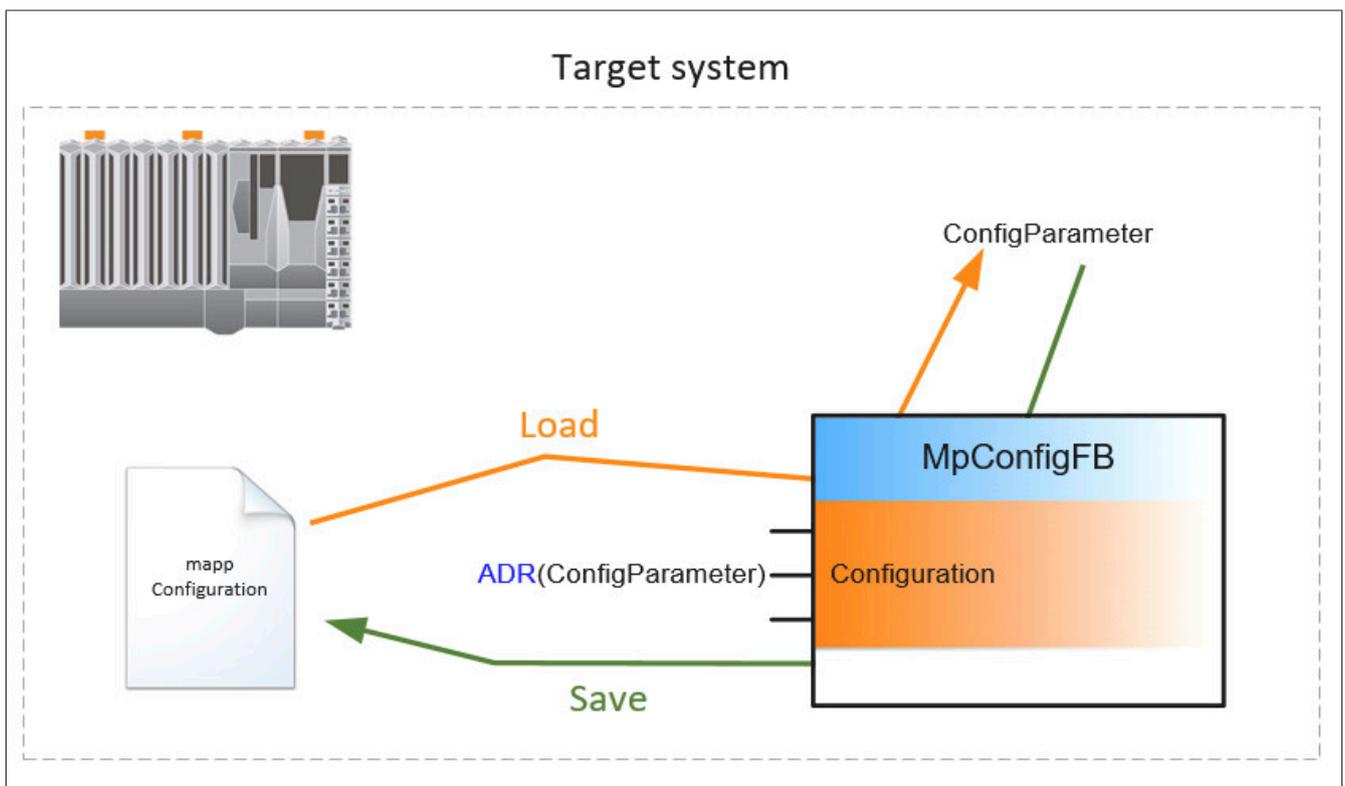
Für diesen Funktionsbaustein wird derselbe MpLink verwendet, wie für [MpUserXLogin](#). Für eine große Maschine an der zum Beispiel 3 Personen gleichzeitig an unterschiedlichen Stationen arbeiten können, kann über diesen Funktionsbaustein eine der Stationen konfiguriert werden.

Mithilfe des Funktionsbausteins sind Änderungen der Konfiguration zur Laufzeit möglich.

Der Parameter "Configuration" ermöglicht es die Konfiguration, bzw. einen Teil der Konfiguration zu bearbeiten.

Der Befehl "Load" wird verwendet, um die aktuell konfigurierten Parameter auszulesen. Dabei wird die aktuell aktive Konfiguration der Steuerung in die PV am Eingang "Configuration" geschrieben. Die Daten in der Struktur werden dabei überschrieben. Danach können die Werte beliebig editiert werden.

Über den Befehl "Save" werden die Werte aus der "Configuration" Struktur auf die aktive Konfiguration übernommen und, wenn möglich, direkt verwendet.



Um die Konfigurations-Parameter in der Applikation korrekt anzupassen, wird folgendermaßen vorgegangen:

1. Befehl "Load = TRUE"

Die aktuelle Konfiguration wird vom Zielsystem in die Konfigurations-Struktur geladen. Der Vorgang ist mit "CommandDone = TRUE" abgeschlossen.

2. Befehl "Load = FALSE"

Rücksetzen des "Load" Befehls. Der Vorgang ist mit "CommandDone = TRUE" abgeschlossen.

3. Parametrieren der Konfigurationsstruktur

Die Parameter der Konfigurationsstruktur können nun wie gewünscht angepasst werden.

4. Befehl "Save = TRUE" setzen

Die geänderte Konfigurationsstruktur wird auf das Zielsystem übertragen. Der Vorgang ist mit "CommandDone = TRUE" abgeschlossen.

5. Befehl "Save = FALSE" setzen

Rücksetzen des "Save" Befehls. Der Vorgang ist mit "CommandDone = FALSE" abgeschlossen. Um zu kontrollieren ob die Parameter erfolgreich geändert worden sind, können die Schritte 1 und 2 erneut ausgeführt werden.

4.1.8 MpUserXLoginUI

Dieser Funktionsblock ermöglicht das Ein- und Ausloggen von Benutzern (über die Visualisierung).

Dieser Funktionsbaustein kann nur innerhalb einer Application Session verwendet werden.

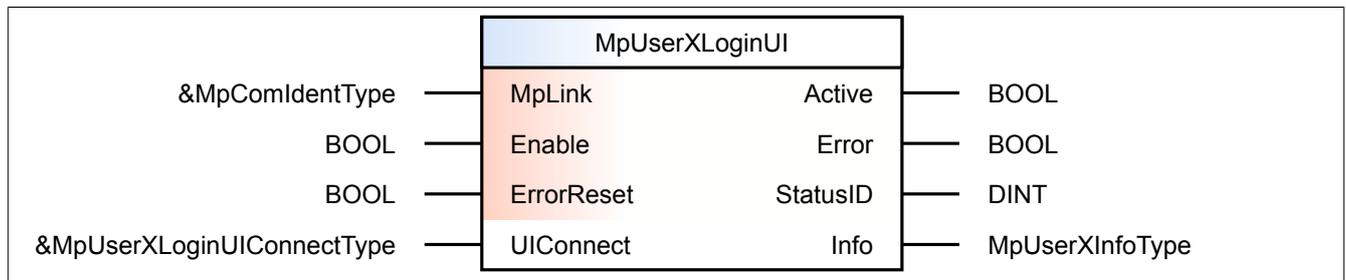


Auf unserem [B&R Tutorial Portal](#) sind Tutorials zum Thema mapp UserX ([GER](#)/[ENG](#)) zu finden. (Zugangsdaten)



Auf unserem [Youtube Kanal](#) sind viele Tutorials zum Thema zu finden.

Funktionsblock



Schnittstelle

I/O	Parameter	Datentyp	Beschreibung
IN	MpLink	Zeiger auf MpComIdentType	Verbindung zur mapp Konfiguration (MpLink einer MpUserXLogin Konfiguration).
IN	Enable	BOOL	FB ist aktiv solange der Eingang gesetzt ist.
IN	ErrorReset	BOOL	Dient zum Zurücksetzen von Funktionsbaustein Fehlern.
IN	UIConnect	Zeiger auf MpUserXLoginUIConnectType	Diese Struktur beinhaltet Parameter, die für die Verbindung zur Visualisierung gedacht sind.
OUT	Active	BOOL	FB ist aktiv.
OUT	Error	BOOL	Fehler bei Abarbeitung aufgetreten.
OUT	StatusID	DINT	Statusinformation.
OUT	Info	MpUserXInfoType	Zusätzliche Komponenteninformationen.

mapp Konzept

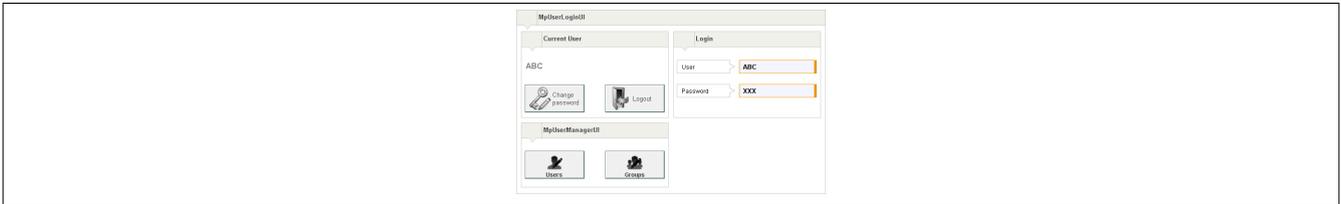
Im Abschnitt mapp Komponenten wird erklärt wie die mapp Komponenten aufgebaut sind. Außerdem wird auf die richtige Verwendung von mapp Komponenten hingewiesen (z.B. beim Download).

4.1.8.1 Beschreibung

Für diese Komponente wird derselbe MpLink verwendet, wie für [MpUserXLogin](#).

Diese Komponente stellt eine Verbindung zwischen Benutzerlogin und einer Visual Components 4 Visualisierung dar. Der Austausch der Daten zwischen [MpUserXLoginUI](#) und der Visualisierung erfolgt über die Struktur [MpUserXLoginUIConnectType](#).

Pro [MpUserXLogin](#) kann nur ein MpUserXLoginUI verwendet werden. Es ist nicht möglich mehrere MpUserXLoginUI-Instanzen zu verwenden.



MpUserLoginUIConnectType

Die UIConnect Struktur unterteilt sich in folgende Themen:

- **Status:** Der "Status" Parameter gibt Auskunft über die aktuelle Aktivität der Benutzerverwaltung
- **CurrentUser, DisplayUnit, Language:** Auskunft über aktuell eingeloggten Benutzer und seine bevorzugten Einstellungen in Bezug auf Sprache und Einheitensystem
- **LoggedIn:** Zeigt an ob aktuell ein Benutzer eubgeloggt ist
- **Login:** Über die Struktur [MpUserXLoginUILoginType](#) sich ein Benutzer ein- und ausloggen kann
- **ChangePassword:** Über die Struktur [MpUserXLoginUIPwdType](#) kann man das Passwort ändern
- **MessageBox:** In der Struktur [MpUserXUIMessageBoxType](#) können die unterschiedlichen Dialogfenster für MpUserX konfiguriert werden
- **DefaultLayerStatus:** Solange eines der Dialogfenster aktiv ist, kann der dahinterliegende Layer über diesen Datenpunkt gesperrt werden

Fehlernummern

Tritt bei einem Login-Versuch ein Fehler auf, da beispielsweise das Passwort falsch angegeben wurde, so wird der Fehler in der MessageBox-Struktur [MpUserXUIMessageBoxType](#) unter "ErrorNumber" angezeigt. Folgende Fehler können auftreten:

- 28672: Passwort ungültig
- 28673: Benutzer existiert nicht
- 28674: Benutzer existiert bereits
- 28675: Login Token ungültig
- 28678: Rolle existiert nicht
- 28679: Passwort nicht sicher genug
- 28680: Benutzer gesperrt
- 28681: Passwortänderung erforderlich
- 28682: Zu viele Login Versuche, Login nicht mehr möglich
- 28685: Kein Benutzer eingeloggt
- 28687: Passwörter identisch
- 28688: Benutzername existiert bereits
- 28693: Benutzer bereits eingeloggt
- 28694: Fehlender Wert an UIConnect
- 28698: Passwort zum Bestätigen falsch
- 28712: Passwort wurde bereits verwendet
- 28718: LDAP Fehler
- 28719: Keine Server-Verbindung
- 28720: Ungültige Eingabe
- 28721: Aktion nicht erlaubt
- 28722: LDAP Warnung
- 28723: Passwort kann nicht geändert werden

4.1.9 MpUserXManagerUI

Dieser Funktionsblock ermöglicht Zugang zum kompletten Benutzermanagement über die Visualisierung.

Es ist nötig die Bibliothek MpFile auf die Steuerung zu übertragen, um bei der "Import" Funktion dieses Funktionsblocks eine Liste mit möglichen Dateien anzuzeigen.

Die "Export" Funktion ist ab der Automation Runtime Version 4.31 verfügbar

Dieser Funktionsbaustein kann nur innerhalb des **User management** verwendet werden.

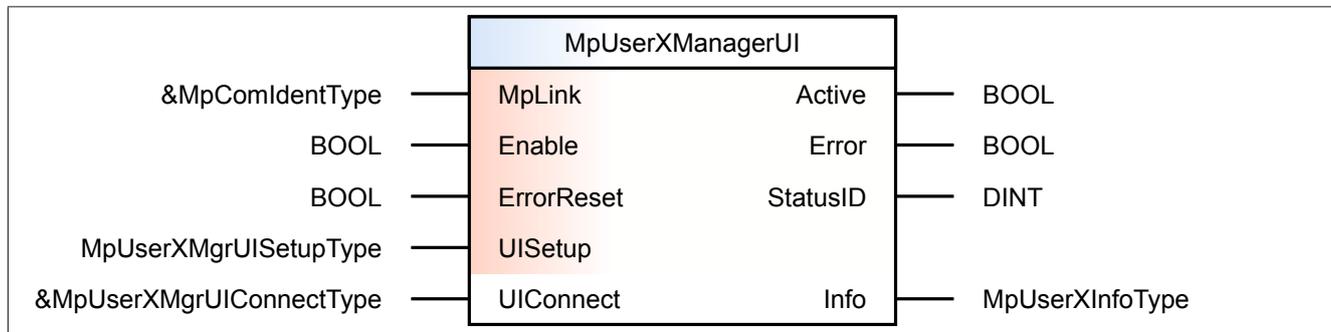


Auf unserem [B&R Tutorial Portal](#) sind Tutorials zum Thema mapp UserX ([GER/ENG](#)) zu finden. (Zugangsdaten)



Auf unserem [Youtube Kanal](#) sind viele Tutorials zum Thema zu finden.

Funktionsblock



Schnittstelle

I/O	Parameter	Datentyp	Beschreibung
IN	MpLink	Zeiger auf MpComIdentType	Verbindung zur mapp Konfiguration (MpLink einer MpUserXLogin Konfiguration)
IN	Enable	BOOL	FB ist aktiv solange der Eingang gesetzt ist.
IN	ErrorReset	BOOL	Dient zum Zurücksetzen von Funktionsbaustein Fehlern.
IN	UISetup	MpUserXMgrUISetupType	Dient zur Konfiguration der Elemente, welche zur Visualisierung verbunden werden.
IN	UIConnect	Zeiger auf MpUserXMgrUIConnectType	Diese Struktur beinhaltet Parameter, die für die Verbindung zur Visualisierung gedacht sind.
OUT	Active	BOOL	FB ist aktiv.
OUT	Error	BOOL	Fehler bei Abarbeitung aufgetreten.
OUT	StatusID	DINT	Statusinformation.
OUT	Info	MpUserXInfoType	Zusätzliche Komponenteninformationen.

mapp Konzept

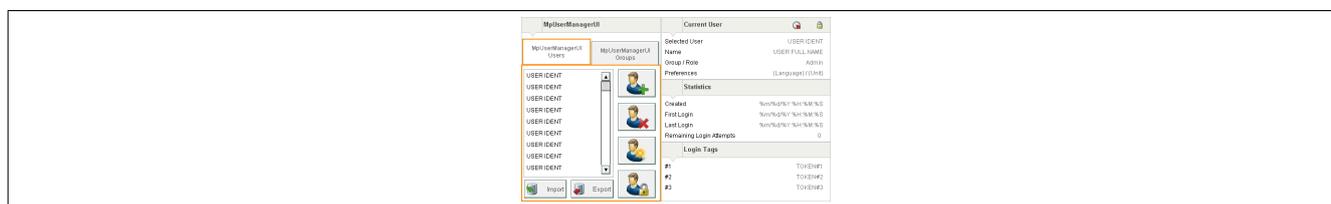
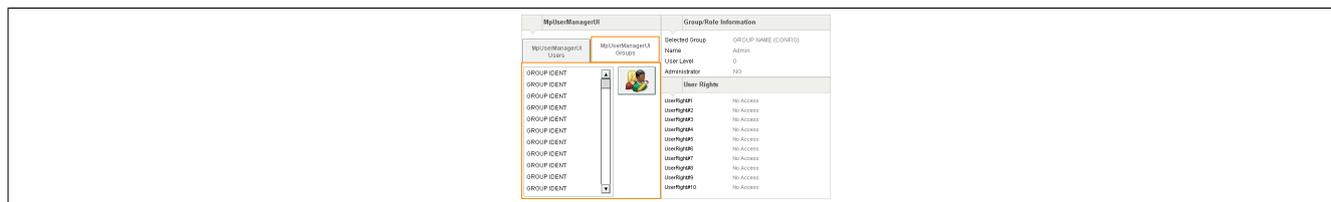
Im Abschnitt mapp Komponenten wird erklärt wie die mapp Komponenten aufgebaut sind. Außerdem wird auf die richtige Verwendung von mapp Komponenten hingewiesen (z.B. beim Download).

4.1.9.1 Beschreibung

Für diese Komponente wird derselbe MpLink verwendet, wie für **MpUserXLogin**.

Diese Komponente stellt eine Verbindung zwischen Benutzermanagement und einer Visual Components 4 Visualisierung dar. Der Austausch der Daten zwischen **MpUserXManagerUI** und der Visualisierung erfolgt über die Struktur **MpUserXMgrUIConnectType**.

Pro **MpUserXLogin** kann nur ein **MpUserXManagerUI** verwendet werden. Es ist nicht möglich mehrere **MpUserXManagerUI**-Instanzen zu verwenden.



Beim Arbeiten auf Linux-Systemen kann auf keine Systemordner mit Zugangsbeschränkung zugegriffen werden. Dies kann zum Beispiel der "lost+found" Ordner sein. Betroffen sind alle Linux Geräte mit EXT2/3/4 Dateisystemen zb. MP71xx oder T30.

```
C:\Users\knatzern>dir \\10.43.37.3\usershare
Volume in drive \\10.43.37.3\usershare is usershare
Volume Serial Number is 9325-608F

Directory of \\10.43.37.3\usershare

12/12/2017  07:42 PM    <DIR>      .
01/19/2018  02:01 PM    <DIR>      ..
12/12/2017  07:27 PM    <DIR>      TestDir
01/01/1970  02:00 AM    <DIR>      lost+found
              0 File(s)              0 bytes
              4 Dir(s)        473,661,440 bytes free
```

MpUserXMgrUIConnectType

Die UIConnect Struktur unterteilt sich in folgende Themen:

- **Status:** Der "Status" Parameter gibt Auskunft über die aktuelle Aktivität der Benutzerverwaltung
- **User:** In der Struktur [MpUserXMgrUIUserType](#) sind die Benutzerdaten aufgelegt. In einer Liste können genaue Informationen über einzelne Benutzer abgerufen werden. Es können auch neue Benutzer hinzugefügt werden, bzw. bereits bestehende editiert werden. Beim Anlegen eines neuen Benutzers wird dem Benutzer automatisch die Rolle "Everyone" zugewiesen
- **Role:** Über die Struktur [MpUserXMgrUIRoleType](#) wird eine Liste mit allen Rollen und deren Berechtigungen angezeigt. Die Rollen können darin auch editiert werden
- **MessageBox:** In der Struktur [MpUserXUIMessageBoxType](#) können die unterschiedlichen Dialogfenster für MpUser konfiguriert werden
- **DefaultLayerStatus:** Solange eines der Dialogfenster aktiv ist, kann der dahinterliegende Layer über diesen Datenpunkt gesperrt werden
- **Export:** Ein Export wird mithilfe der Struktur [MpUserXMgrUIExportType](#) durchgeführt. Über einen Dialog wird bestimmt, welche Datei exportiert werden soll. Exportiert werden alle Benutzer und Benutzergruppen.
- **Import:** Ein Import wird mithilfe der Struktur [MpUserXMgrUIImportType](#) durchgeführt. Über einen Dialog wird bestimmt, welche Datei importiert werden soll. Es können in der [MpUserX Konfiguration](#) verschiedene Import Modi unter "User Import Mode" und "Role Import Mode" ausgewählt werden.

Die zusätzlichen Informationen eines Benutzers, wie zum Beispiel Einheit oder Sprache, werden in der [Konfiguration](#) und nicht im Benutzer-Rollen-System gespeichert.

Aus diesem Grund werden, bei einer Neuübertragung des Projekts auf das Zielsystem, die Benutzerdaten auf dem Zielsystem, mit denen aus der Konfiguration überschrieben. Um die zusätzlichen Informationen, welche zur Laufzeit vielleicht bearbeitet wurden, nicht zu verlieren, empfiehlt es sich die Benutzerdaten vorher zu exportieren.

Nachdem das Projekt auf das Zielsystem übertragen wurde, kann die Benutzerinformation wieder importiert werden. Somit ist sichergestellt, dass keine Information verloren geht.

Folgende Parameter sind davon betroffen:

- Full name
- Language
- Measurement system
- Additional data

Es ist nötig die Bibliothek MpFile auf die Steuerung zu übertragen, um bei der "Import" Funktion dieses Funktionsblocks eine Liste mit möglichen Dateien anzuzeigen.

MpUserXMgrUISetupType

In dieser Struktur wird über "UserListSize" definiert, wieviele Einträge der Benutzerliste auf einer Seite der Visualisierung angezeigt werden. Über "ScrollWindow" wird entschieden wieviele Einträge der Liste beim auf- oder abscrollen im Vorhinein angezeigt werden. Über "FileDevice" wird das Speichermedium ausgewählt, auf welchem die Benutzerdaten zum Import liegen, bzw. wohin sie exportiert werden sollen. Über "FileExtension" wird die Endung für Exportdateien festgelegt, bzw. werden in der Importliste nur Dateien mit dieser Endung angezeigt.

Über "ConfirmOverwriteUser" und "ConfirmOverwriteRole" kann festgelegt werden ob das Überschreiben von Benutzern/Rollen über einen Dialog bestätigt werden soll.

Fehlernummern

Tritt bei einem Login-Versuch ein Fehler auf, da beispielsweise das Passwort falsch angegeben wurde, so wird der Fehler in der MessageBox-Struktur `MpUserXUIMessageBoxType` unter "ErrorNumber" angezeigt. Folgende Fehler können auftreten:

- 28673: Benutzer existiert nicht
- 28674: Benutzer existiert bereits
- 28675: Login Token ungültig
- 28676: Keine Administratorrechte
- 28677: Ungenügende Benutzerrechte
- 28678: Rolle existiert nicht
- 28679: Passwort nicht sicher genug
- 28680: Benutzer gesperrt
- 28681: Passwortänderung erforderlich
- 28682: Zu viele Login Versuche, Login nicht mehr möglich
- 28683: Benutzer hat keine Adminrechte
- 28685: Kein Benutzer eingeloggt
- 28687: Passwörter identisch
- 28688: Benutzername existiert bereits
- 28693: Benutzer bereits eingeloggt
- 28694: Fehlender Wert an UIConnect
- 28695: Ungültiger Benutzername
- 28697: MpUserXLogin fehlt
- 28698: Passwort zum Bestätigen falsch
- 28699: Angegebenes Systemrecht existiert nicht
- 28700: Import fehlgeschlagen
- 28701: Fehler beim Laden der Importdatei
- 28702: Fehler beim Speichern der Exportdatei
- 28707: Name des Benutzers ist zu kurz
- 28708: Rolle existiert bereits
- 28712: Passwort wurde bereits verwendet
- 28713: Ungültige Import Datei
- 28721: Aktion nicht erlaubt

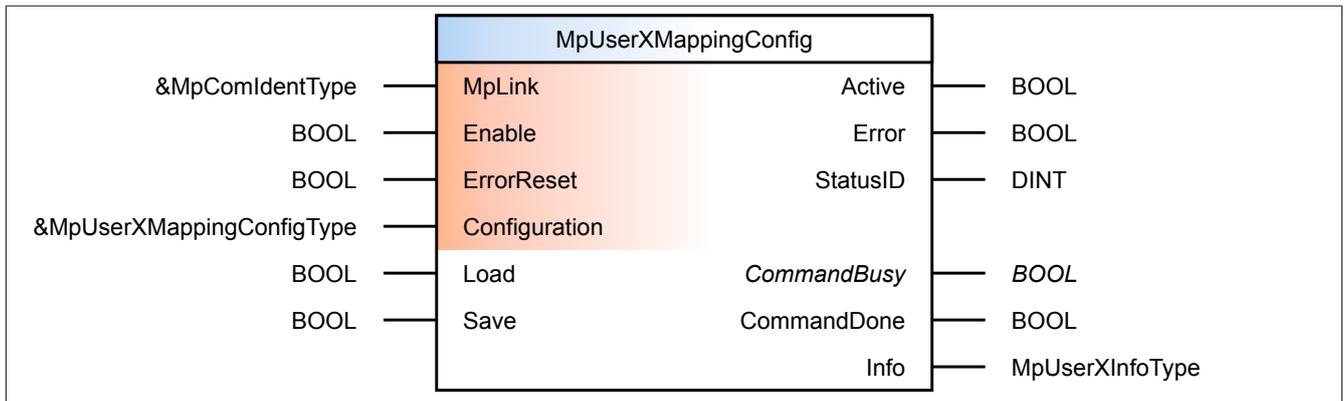
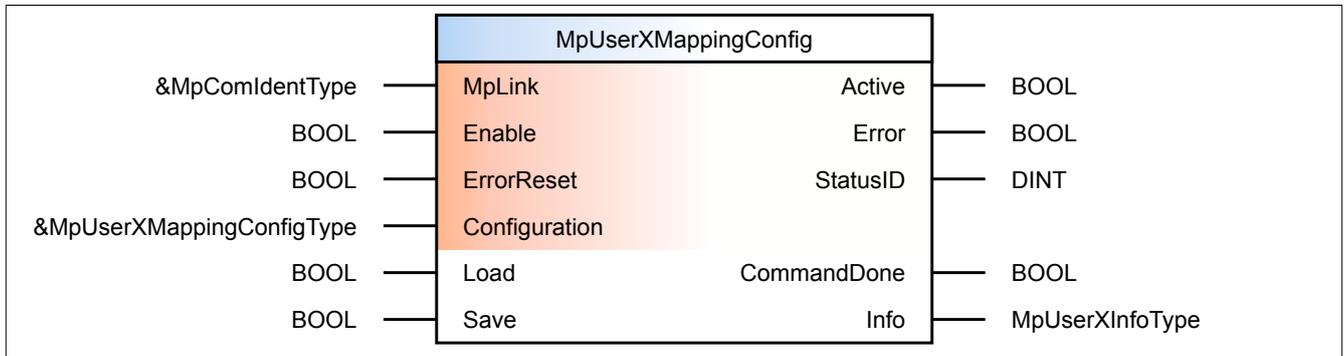
4.1.10 MpUserXMappingConfig

Mithilfe dieses Funktionsbausteins kann das Mapping der Rollen zur Laufzeit geändert und angepasst werden.

Dieser Funktionsbaustein kann nur innerhalb des `User management` verwendet werden.

Funktionsbaustein

Optionale Parameter



Schnittstelle

I/O	Parameter	Datentyp	Beschreibung
IN	MpLink	Zeiger auf MpComIdentType	Verbindung zur mapp Konfiguration (MpLink einer MpUserXMapping Konfiguration)
IN	Enable	BOOL	FB ist aktiv solange der Eingang gesetzt ist.
IN	ErrorReset	BOOL	Dient zum Zurücksetzen von Funktionsbaustein Fehlern.
IN	Configuration	Zeiger auf MpUserXMappingConfigType	Struktur, in der die Konfiguration festgelegt wird.
IN	Load	BOOL	Lädt die Konfiguration der Komponente.
IN	Save	BOOL	Speichert die Konfiguration der Komponente.
OUT	Active	BOOL	FB ist aktiv.
OUT	Error	BOOL	Fehler bei Abarbeitung aufgetreten.
OUT	StatusID	DINT	Statusinformation.
OUT	CommandBusy	BOOL	Funktionsblock führt aktuell einen Befehl aus.
OUT	CommandDone	BOOL	Abarbeitung erfolgreich. FB ist fertig.
OUT	Info	MpUserXInfoType	Zusätzliche Komponenteninformationen.

mapp Konzept

Im Abschnitt mapp Komponenten wird erklärt wie die mapp Komponenten aufgebaut sind. Außerdem wird auf die richtige Verwendung von mapp Komponenten hingewiesen (z.B. beim Download).

4.1.10.1 Beschreibung

Dieser Funktionsbaustein verwendet den MpLink einer [MpUserXMapping Konfiguration](#).

Konfigurieren der [MpUserXMappingConfigType](#) Struktur

Über diese Konfiguration kann das Rollen-Mapping zwischen den Rollen des Active Directory Servers und den Rollen innerhalb Automation Studio angepasst werden.

Für mehr Information zum Rollen-Mapping siehe [hier](#).

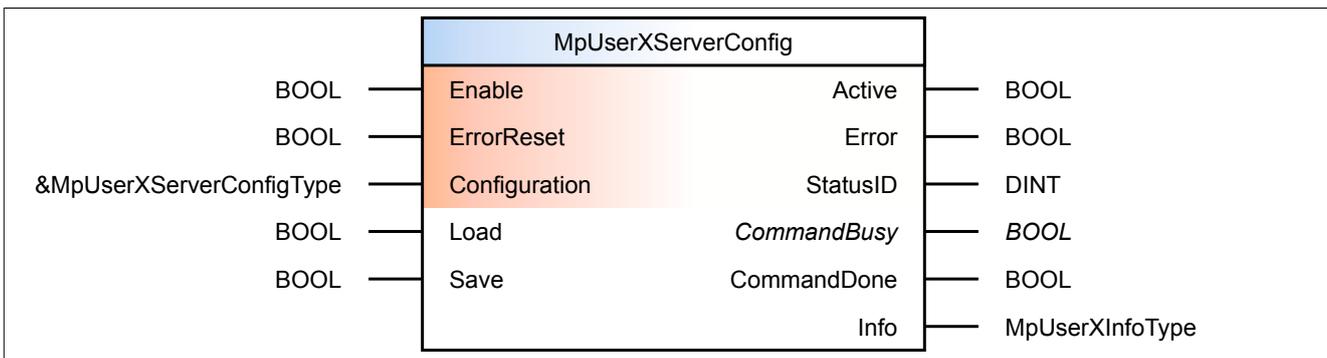
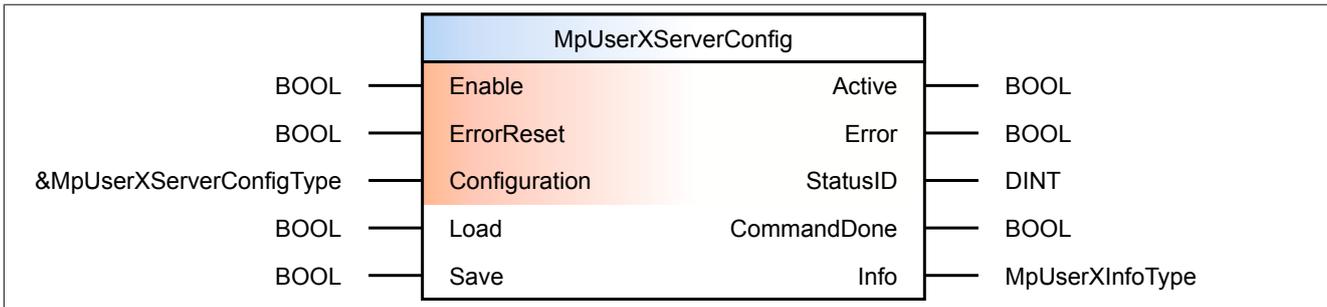
4.1.11 MpUserXServerConfig

Mithilfe dieses Funktionsbausteins können die Verbindungs-Parameter zum Active Directory Server zur Laufzeit angepasst werden.

Dieser Funktionsbaustein kann nur innerhalb des [User management](#) verwendet werden.

Funktionsbaustein

Optionale Parameter



Schnittstelle

I/O	Parameter	Datentyp	Beschreibung
IN	MpLink	Zeiger auf Mp-ComIdentType	Verbindung zur mapp Konfiguration (MpLink einer MpUserXServer Konfiguration)
IN	Enable	BOOL	FB ist aktiv solange der Eingang gesetzt ist.
IN	ErrorReset	BOOL	Dient zum Zurücksetzen von Funktionsbaustein Fehlern.
IN	Configuration	Zeiger auf MpUserXServerConfigType	Struktur, in der die Konfiguration festgelegt wird.
IN	Load	BOOL	Lädt die Konfiguration der Komponente.
IN	Save	BOOL	Speichert die Konfiguration der Komponente.
OUT	Active	BOOL	FB ist aktiv.
OUT	Error	BOOL	Fehler bei Abarbeitung aufgetreten.
OUT	StatusID	DINT	Statusinformation.
OUT	CommandBusy	BOOL	Funktionsblock führt aktuell einen Befehl aus.
OUT	CommandDone	BOOL	Abarbeitung erfolgreich. FB ist fertig.
OUT	Info	MpUserXInfoType	Zusätzliche Komponenteninformationen.

mapp Konzept

Im Abschnitt mapp Komponenten wird erklärt wie die mapp Komponenten aufgebaut sind. Außerdem wird auf die richtige Verwendung von mapp Komponenten hingewiesen (z.B. beim Download).

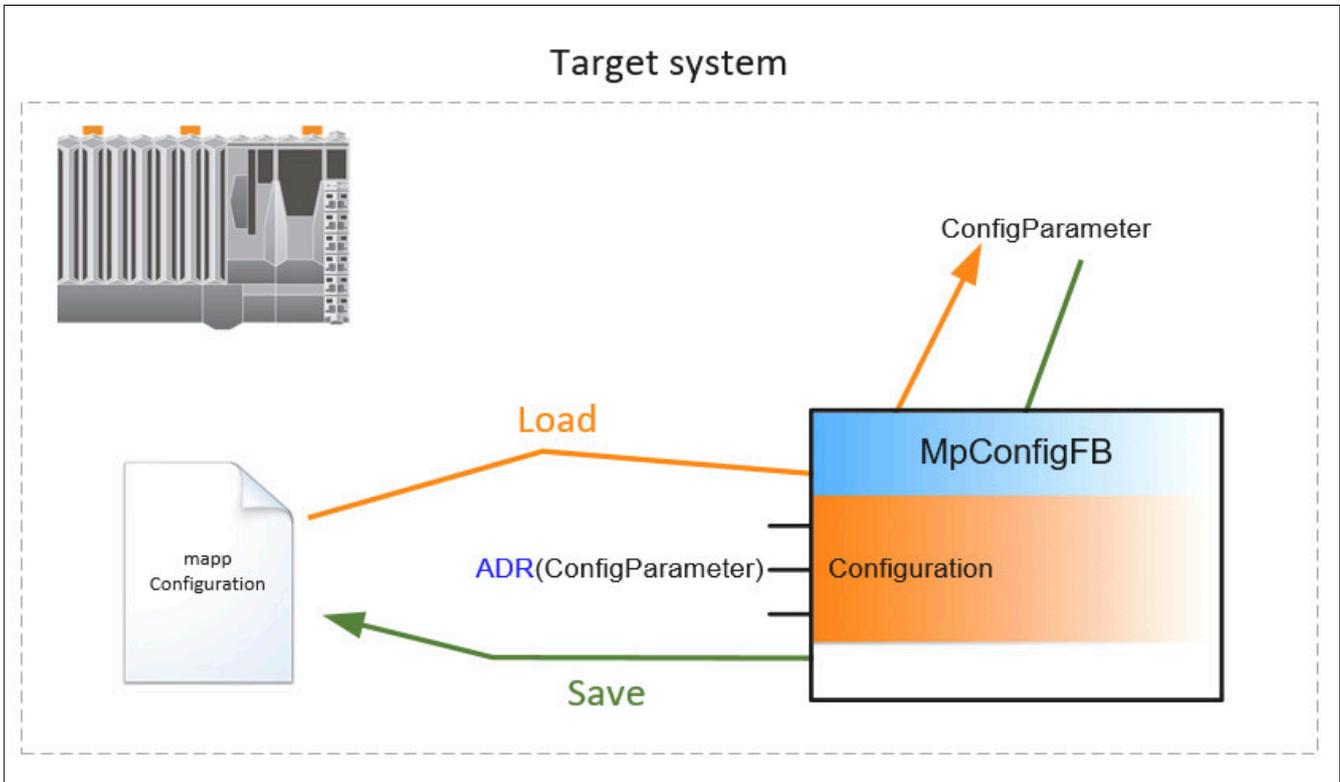
4.1.11.1 Beschreibung

Mithilfe des Funktionsbausteins sind Änderungen der Konfiguration zur Laufzeit möglich.

Der Parameter "Configuration" ermöglicht es die Konfiguration, bzw. einen Teil der Konfiguration zu bearbeiten.

Der Befehl "Load" wird verwendet, um die aktuell konfigurierten Parameter auszulesen. Dabei wird die aktuell aktive Konfiguration der Steuerung in die PV am Eingang "Configuration" geschrieben. Die Daten in der Struktur werden dabei überschrieben. Danach können die Werte beliebig editiert werden.

Über den Befehl "Save" werden die Werte aus der "Configuration" Struktur auf die aktive Konfiguration übernommen und, wenn möglich, direkt verwendet.



Um die Konfigurations-Parameter in der Applikation korrekt anzupassen, wird folgendermaßen vorgegangen:

1. Befehl "Load = TRUE"

Die aktuelle Konfiguration wird vom Zielsystem in die Konfigurations-Struktur geladen. Der Vorgang ist mit "CommandDone = TRUE" abgeschlossen.

2. Befehl "Load = FALSE"

Rücksetzen des "Load" Befehls. Der Vorgang ist mit "CommandDone = TRUE" abgeschlossen.

3. Parametrieren der Konfigurationsstruktur

Die Parameter der Konfigurationsstruktur können nun wie gewünscht angepasst werden.

4. Befehl "Save = TRUE" setzen

Die geänderte Konfigurationsstruktur wird auf das Zielsystem übertragen. Der Vorgang ist mit "CommandDone = TRUE" abgeschlossen.

5. Befehl "Save = FALSE" setzen

Rücksetzen des "Save" Befehls. Der Vorgang ist mit "CommandDone = FALSE" abgeschlossen. Um zu kontrollieren ob die Parameter erfolgreich geändert worden sind, können die Schritte 1 und 2 erneut ausgeführt werden.

Konfigurieren der **MpUserXServerConfigType** Struktur

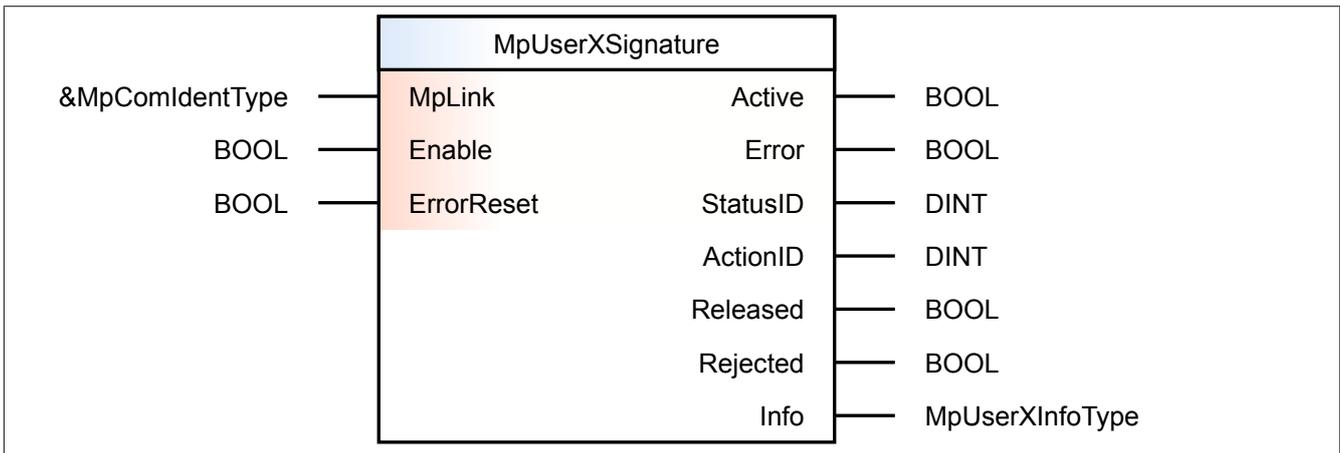
Über diese Konfiguration können die Server-Einstellungen der **MpUserX Konfiguration** angepasst werden.

4.1.12 MpUserXSignature

Dieser Funktionsblock prüft, ob das Zeichnen anhand einer elektronischen Signatur erfolgreich war.

Dieser Funktionsbaustein kann innerhalb einer **Application Session sowie **Visualization Session** verwendet werden.**

Funktionsblock



Schnittstelle

I/O	Parameter	Datentyp	Beschreibung
IN	MpLink	Zeiger auf Mp-ComIdentType	Verbindung zur mapp Konfiguration (MpLink einer MpUserXSign Konfiguration).
IN	Enable	BOOL	FB ist aktiv solange der Eingang gesetzt ist.
IN	ErrorReset	BOOL	Dient zum Zurücksetzen von Funktionsbaustein Fehlern.
OUT	Active	BOOL	FB ist aktiv.
OUT	Error	BOOL	Fehler bei Abarbeitung aufgetreten.
OUT	StatusID	DINT	Statusinformation.
OUT	ActionID	DINT	Angabe der Aktion, welche signiert werden soll.
OUT	Released	BOOL	Wird für einen Zyklus auf TRUE gesetzt, wenn die elektronische Signatur erfolgreich durchgeführt wurde.
OUT	Rejected	BOOL	Wird für einen Zyklus auf TRUE gesetzt, wenn die elektronische Signatur nicht erfolgreich durchgeführt wurde.
OUT	Info	MpUserXInfoType	Zusätzliche Komponenteninformationen.

mapp Konzept

Im Abschnitt mapp Komponenten wird erklärt wie die mapp Komponenten aufgebaut sind. Außerdem wird auf die richtige Verwendung von mapp Komponenten hingewiesen (z.B. beim Download).

4.1.12.1 Beschreibung

Für diesen Funktionsblock wird der MpLink einer MpUserXSign Konfiguration verwendet.

Dieser Funktionsblock wird für das elektronische Signieren einer Aktion benötigt. Das Prinzip einer elektronischen Signatur ist im Abschnitt [Elektronische Signatur](#) erklärt.

Wird über die Visualisierung eine elektronische Signatur durchgeführt (Visualisierung ist hierbei mit [MpUserXSignatureUI](#) verbunden), so wird an diesem Funktionsblock angezeigt, ob der Prozess erfolgreich war oder nicht. Die Ausgängen "Released", bzw. "Rejected" werden jeweils für einen Zyklus gesetzt.

- **Released:** Die elektronische Signatur wurde erfolgreich durchgeführt, es wurde der richtige Benutzername und das korrekte Passwort verwendet. Das Benutzerlevel des Signierenden ist ausreichend für die geforderte Aktion
- **Rejected:** Die elektronische Signatur wurde nicht erfolgreich durchgeführt, der Benutzer hat den Prozess abgebrochen oder die Anzahl der fehlgeschlagenen Signatur - Versuche hat den maximal erlaubten Wert überschritten. Tritt dieser Fall ein kann eine neue Signatur gestartet werden. Dabei ist die Anzahl der Signatur-Versuche wieder genau gleich. Die Anzahl der Signatur-Versuche wird in der [MpUserX Konfiguration](#) festgelegt. Wird das Passwort des Benutzers bei der Eingabe der Signatur zu oft falsch eingegeben, wird der Benutzer autoamtisch gesperrt. Wie oft das Passwort falsch eingegeben werden kann, wird in der [MpUserX Konfiguration](#) definiert.

Abhängig von dem gesetzten Ausgang, kann in der Applikation auf die elektronische Signatur reagiert werden, beispielsweise der auf der Visualisierung eingegeben neue Prozesswert von einer temporären Variable auf die eigentliche Prozessvariable übertragen werden.

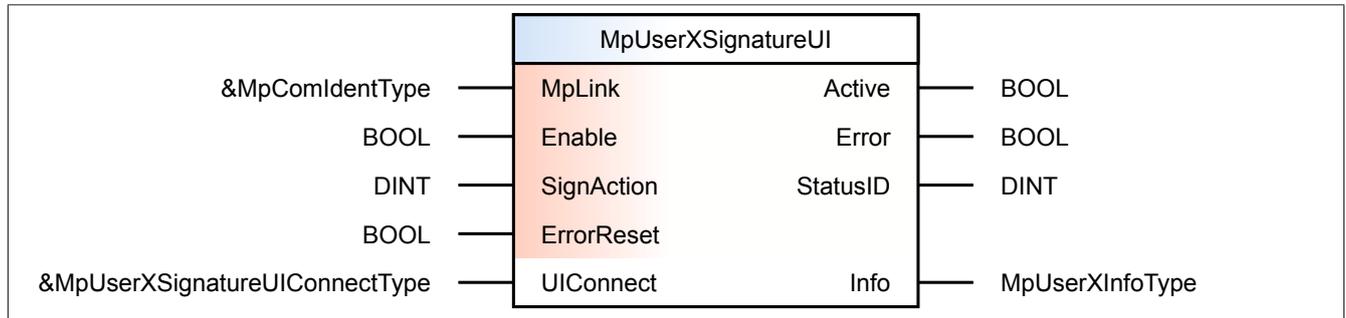
Der Ausgang "ActionID" gibt an welche Aktion signiert wurde. Die unterschiedlichen Aktionen können in der [Konfiguration](#) definiert werden.

4.1.13 MpUserXSignatureUI

Dieser Funktionsblock ermöglicht eine elektronische Signatur über die Visualisierung.

Dieser Funktionsbaustein kann innerhalb einer **Application Session** sowie **Visualization Session** verwendet werden.

Funktionsblock



Schnittstelle

I/O	Parameter	Datentyp	Beschreibung
IN	MpLink	Zeiger auf MpComIdentType	Verbindung zur mapp Konfiguration (MpLink einer MpUserXSign Konfiguration).
IN	Enable	BOOL	FB ist aktiv solange der Eingang gesetzt ist.
IN	SignAction	DINT	Löst den Prozess rund um die elektronische Signatur über die Applikation aus.
IN	ErrorReset	BOOL	Dient zum Zurücksetzen von Funktionsbaustein Fehlern.
IN	UIConnect	Zeiger auf MpUserXSignatureUIConnectType	Diese Struktur beinhaltet Parameter, die für die Verbindung zur Visualisierung gedacht sind.
OUT	Active	BOOL	FB ist aktiv.
OUT	Error	BOOL	Fehler bei Abarbeitung aufgetreten.
OUT	StatusID	DINT	Statusinformation.
OUT	Info	MpUserXInfoType	Zusätzliche Komponenteninformationen.

mapp Konzept

Im Abschnitt mapp Komponenten wird erklärt wie die mapp Komponenten aufgebaut sind. Außerdem wird auf die richtige Verwendung von mapp Komponenten hingewiesen (z.B. beim Download).

4.1.13.1 Beschreibung

Für diese Komponente wird derselbe MpLink verwendet, wie für [MpUserXSignature](#)

Diese Komponente verwaltet das Dialogfenster zur elektronischen Signatur. Der Austausch der Daten zwischen [MpUserXSignatureUI](#) und der Visualisierung erfolgt über die Struktur [MpUserXSignatureUIConnectType](#).

Pro [MpUserXSignature](#) kann nur ein MpUserXSignatureUI verwendet werden. Es ist nicht möglich mehrere MpUserXSignatureUI-Instanzen zu verwenden.

MpUserXSignatureUIConnectType

Die UIConnect Struktur unterteilt sich in folgende Themen:

- **Status:** Der "Status" Parameter gibt Auskunft über die aktuelle Aktivität der Benutzerverwaltung
- **SignAction:** Dieser Parameter gibt an welche Aktion signiert werden soll. Dieser Parameter muss mit jedem *Completion Datenpunkt eines Visualisierungs-Elements* verbunden werden, welches eine elektronische Signatur fordert. Der Wert, auf welchen der *Completion Datenpunkt* gesetzt wird, muss der "ActionID" der jeweiligen Aktion aus der Konfiguration entsprechen
- **Dialog:** Dialogfenster, welches sich öffnet, falls eine elektronische Signatur gefordert wird. Es müssen Benutzername und Passwort eingegeben werden, sowie optional ein Kommentar
- **MessageBox:** Unterschiedliche Meldungen an den Benutzer (z.B. fehlerhafter Benutzername oder Passwort wurde angegeben)
- **DefaultLayerStatus:** Solange eines der Dialogfenster aktiv ist, kann der dahinterliegende Layer über diesen Datenpunkt gesperrt werden

Fehlernummern

Tritt bei einem Signatur-Versuch ein Fehler auf, da beispielsweise das Passwort falsch angegeben wurde, so wird der Fehler in der MessageBox-Struktur `MpUserXUIMessageBoxType` unter "ErrorNumber" angezeigt. Folgende Fehler können auftreten:

- 28672: Passwort ungültig
- 28673: Benutzer existiert nicht
- 28677 : Ungenügende Benutzerrechte
- 28680: Benutzer gesperrt
- 28681: Passwortänderung erforderlich
- 28694: Fehlender Wert an UIConnect
- 28695 : Ungültiger Benutzername
- 28696 : Passwort fehlt
- 28710 : Signieren fehlgeschlagen
- 28711 : Letzte Signatur wurde noch nicht geprüft
- 28718: LDAP Fehler
- 28719: Keine Server-Verbindung
- 28720: Ungültige Eingabe
- 28722 : LDAP Warnung

4.2 Datentypen und Enumeratoren

4.2.1 Datentypen

4.2.1.1 MpUserXConfigType

Über diesen Datentyp wird das Benutzermanagement konfiguriert.

Wie das Konfigurationsupdate der einzelnen Parameter zur Laufzeit durchgeführt wird, ist im Abschnitt [Konfiguration während der Laufzeit](#) erklärt.

Alle Passwort Parameter gelten nur für Benutzer die neu über die Visualisierung mithilfe von `MpUserX-ManagerUI` angelegt werden. Für Benutzer die über das Konfigurationssystem in Automation Studio, oder über `ArUser` erstellt werden, gelten diese Regeln nicht.

Parameter	Datentyp	Wert	Beschreibung
NoDelete	BOOL	TRUE	Definiert das Lösch-Verhalten innerhalb des Benutzermanagements. Wird der Parameter auf TRUE gesetzt, so kann der Benutzername, welcher ein gelöschte Benutzer verwendet hat, nicht mehr vergeben werden. Durch diese Funktion wird sichergestellt, dass Benutzernamen nur einmalig im System verwendet werden können. Wurde also beispielsweise der Benutzer "Dave" gelöscht, so kann kein Benutzer mehr mit diesen Namen erstellt werden. Dadurch wird sichergestellt, dass beispielsweise in Ereignislisten keine Verwechslung im Benutzernamen stattfindet.
PasswordChangeReq	BOOL	FALSE	Passwortänderung beim ersten Login notwendig
PasswordCase	BOOL	FALSE	Passwort muss Klein- und Großbuchstaben enthalten
PasswordAlpha	BOOL	TRUE	Passwort muss alphanumerisch sein
PasswordLength	UINT	5	Minimale Passwortlänge. Es muss mindestens ein Zeichen angegeben werden. Es ist nicht erlaubt eine Passwort-Länge von 0 zu haben.

Parameter	Datentyp	Wert	Beschreibung
LoginAttempts	UINT	3	Maximale Anzahl an möglichen Login-Fehlversuchen, bis der Benutzer gesperrt wird. Bei der Eingabe von 0 wird der Benutzer niemals gesperrt. Bei Benutzergruppen mit Adminrechten werden diese maximal eine Stunde gesperrt, bzw. sind Benutzergruppen mit Adminrechten nach einem Warmstart der Maschine nicht mehr gesperrt.
PasswordChangeInterval	DINT	0	Intervall, in dem das Passwort geändert werden muss
UserExpirationTime	DINT	0	Gibt die Verfallszeit für einen Benutzer an. Nach Ablauf der Zeit ist der Benutzer gesperrt. Der Benutzer kann nur von einem Admin entsperrt werden. Bei der Angabe von "0" wird die Funktion nicht verwendet.
UserNameLength	UINT	0	Minimale Länge bei der Eingabe eines neuen Benutzernamens
SignAttempts	UINT	3	Maximale Anzahl an möglichen Signatur-Fehlversuchen, bis der Signatur-Vorgang abgebrochen wird. Die Login-Versuche ("LoginAttempts") und die Signatur-Versuche hängen miteinander zusammen. Bei jeder falschen Benutzereingabe während eines Signatur-Versuchs wird auch ein Login-Versuch verwendet. Gibt es 2 Signatur-Versuche und 1 Login-Versuch, so wird der Benutzer bei einem fehlgeschlagenen Signatur-Versuch (Eingabe eines falschen Passworts/Benutzernamens) bereits gesperrt. Wird die Signatur abgebrochen, so bleibt der Login-Versuch erhalten. Für mehr Information siehe hier .
PasswordHistory	UINT	0	Gibt an wie viele vorherigen Passwörter nicht wiederverwendet werden dürfen Es kann ein Wert zwischen 0 und 5 eingegeben werden Bei der Angabe von 0 wird die Funktion nicht verwendet Bei der Angabe von zum Beispiel 2, darf das neu eingegebene Passwort nicht mit den zwei vorherigen Passwörtern übereinstimmen
PasswordSpecial	BOOL	FALSE	Bestimmt, ob ein Passwort Sonderzeichen enthalten muss. Folgende Sonderzeichen können verwendet werden: !#\$%&'()*+,-./:;<=>@[\\]^_`{ }~
EditSameLevel	BOOL	FALSE	Bestimmt, ob ein Admin Benutzer mit dem gleichem Benutzerlevel editieren darf
ImportUser	MpUserXImportModeEnum	mpUSERX_IMPORT_OVERWRITE	Modus für Benutzerimport
ImportRole	MpUserXImportModeEnum	mpUSERX_IMPORT_OVERWRITE	Modus für Rollenimport

Konfigurationsupdate

Parameter	Datentyp	Wert	Beschreibung
ImportUnchecked	BOOL	FALSE	Ignorieren der Check-Summe in der Importdatei
AdminUnlockTime	DINT	3600	Nach Ablauf der angegebenen Zeit wird der gesperrte Administrator entsperrt. (Einheit: [sec]) Wird 0 angegeben, wird der Administrator nicht automatisch entsperrt. Die Entsperr-Funktion ist bei 0 nicht verfügbar
AutoLogoutTime	DINT	3600	Durch "Auto logout time" wird der Benutzer bei Inaktivität automatisch nach der angegebenen Zeit ausgeloggt. In einer Application Session wird der Parameter "Auto logout time" automatisch auf den angegebenen Wert zurückgesetzt sobald sich der Eingang "LifeSign" von MpUserXLogin erhöht. Verändert sich der Wert an "LifeSign" während der angegebenen Ausloggzeit nicht, wird der Benutzer automatisch ausgeloggt. In einer Visualization Session wird der Parameter "Auto logout time" automatisch auf den angegebenen Wert zurückgesetzt sobald eine Benutzer-Aktion an der Visualisierung durchgeführt wurde. Wird als automatische Ausloggzeit 0 Sekunden angegeben, so wird die Funktion deaktiviert.
PasswordExpirationNotification	DINT	0	Legt fest, wann der Benutzer über den Ablauf des Passworts benachrichtigt werden soll. Die Angabe erfolgt hier in Sekunden. Wird beispielsweise 432000 eingegeben, so wird der Benutzer 5 Tage vor Ablauf des Passworts informiert. Bei der Eingabe von 0 wird die Benachrichtigungs-Funktion deaktiviert.

4.2.1.2 MpUserXDiagType

Dieser Datentyp wird in der Struktur für zusätzliche Informationen als Unterstruktur für die Diagnose verwendet und liefert weitere Infos über die StatusID.

Parameter	Datentyp	Beschreibung
StatusID	MpUserXStatusIDType	StatusID Diagnose Struktur

4.2.1.3 MpUserXGroupToRoleMappingType

Mithilfe dieses Datentypen kann die Mapping-Tabelle definiert werden. Für mehr Information dazu siehe Abschnitt [Mappen der Rollen](#).

Parameter	Datentyp	Beschreibung
TableName	STRING [50]	Angabe der Mapping-Tabelle. Es wird der mapp Link einer MpUserXMapping Konfiguration angegeben.

4.2.1.4 MpUserXHostType

Über diesen Datentypen werden die Server-Einstellungen festgelegt. Mehr Informationen zu den einzelnen Parametern kann in der [MpUserX Konfiguration](#) gefunden werden.

Parameter	Datentyp	Beschreibung	Konfiguration- update
Host	STRING [255]	Angabe des Hosts.	.
Port	UINT	Angabe der Port-Nummer	.
Certificate	STRING [255]	Name des Zertifikats	.
BaseDN	STRING [255]	Dieser Parameter kann optional bei der Verwendung eines Active Directory Servers verwendet werden. Es wird die Base-DN angegeben.	.
UserLocationDN	STRING [255]	Dieser Parameter muss bei der Verwendung eines 389 Directory Servers angegeben werden. Es wird der vollqualifizierte Namen der Organisationseinheit, in der sich der Nutzer am Server befindet, angegeben. Am 389 Directory Server wird die "UserLocationDN" als "entrydn" bezeichnet.	.

4.2.1.5 MpUserXInfoType

Mit diesem Datentyp wird für die [MpUserX](#) Komponenten zusätzliche Information bereitgestellt.

Parameter	Datentyp	Beschreibung
Diag	MpUserXDiagType	Eine Diagnosestruktur für die StatusID

4.2.1.6 MpUserXLoginConfigType

In diesem Datentyp sind Parameter gesammelt, die für jeden Loginpunkt (Zugriffspunkt auf die Maschine) unterschiedlich sein können. Der Parameter "AutoLogout" wird automatisch auf den angegebenen Wert zurückgesetzt sobald sich der Eingang "LifeSign" von [MpUserXLogin](#) erhöht. Verändert sich der Wert an "LifeSign" während der angegebenen Ausloggzeit nicht, wird der Benutzer automatisch ausgeloggt.

Wie das Konfigurationsupdate der einzelnen Parameter zur Laufzeit durchgeführt wird, ist im Abschnitt [Konfiguration während der Laufzeit](#) erklärt.

Parameter	Datentyp	Wert	Beschreibung	Konfiguration- update
AutoLogout	DINT	600	Automatische Ausloggzeit (bei inaktivem Benutzer) [s]	.

4.2.1.7 MpUserXLoginInfoType

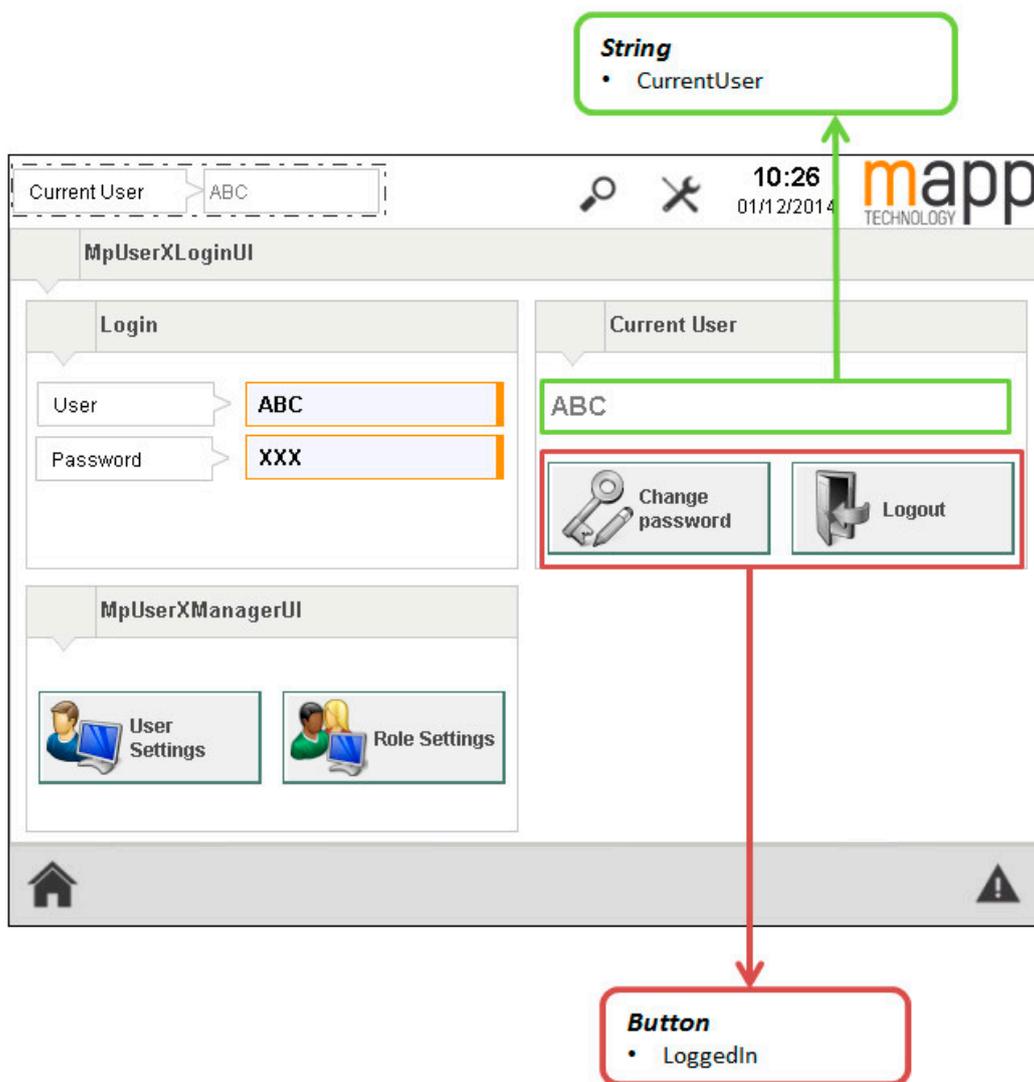
Mit diesem Datentyp wird für die [MpUserXLogin](#) Komponente zusätzliche Information bereitgestellt.

Parameter	Datentyp	Beschreibung
AutoLogoutRemain	TIME	Verbleibende Zeit bis zum automatischen Ausloggen [s]
Diag	MpUserXDiagType	Eine Diagnosestruktur für die StatusID
DaysUntilPasswordExpiration	UDINT	Verbleibende Zeit bis zum Ablauf des Passworts [Tage] Bei "-1" wird die Passwort-Ablauf-Funktion nicht verwendet. Bei "0" und "Password change required = TRUE" in der MpUserX Konfiguration ist das Passwort bereits abgelaufen. Bei einem Wert >= 0 wird angezeigt, dass das Passwort ablaufen wird. Über den Ausgangsparameter StatusID an MpUserXLogin wird angezeigt, ob das Passwort bereits abgelaufen ist oder nicht.

4.2.1.8 MpUserXLoginUIConnectType

Mithilfe dieses Datentypes wird eine Verbindung zwischen der [MpUserXLoginUI](#) Komponente und der Visual Components 4 Visualisierung erstellt. Über diese Struktur kann ein- und ausgeloggt werden.

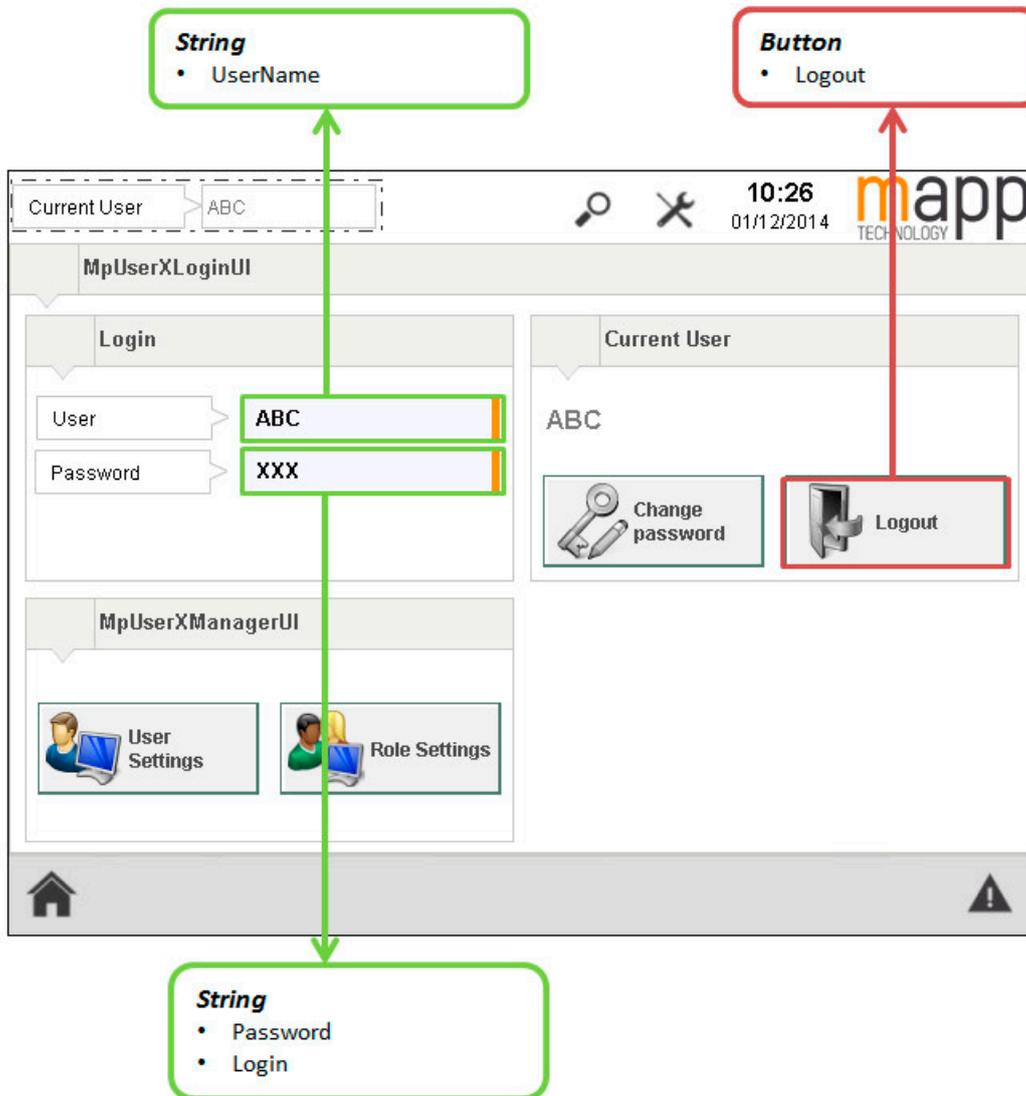
Parameter	Datentyp	Beschreibung
Status	MpUserXUIStatusEnum	Aktuelle Operation
CurrentUser	WSTRING[50]	Aktuell eingeloggtter Benutzer VC4 Anbindung: Datapoint von <i>String</i>
Language	STRING[20]	Bevorzugte Sprache des aktuellen (oder letzten) aktiven Benutzers
DisplayUnit	STRING[20]	Bevorzugte Einheit des aktuellen (oder letzten) aktiven Benutzers
LoggedIn	BOOL	Benutzer ist eingeloggt VC4 Anbindung: LockingDatapoint von <i>Button</i> / Locking: Datapoint < Level / Level: 1
Login	MpUserXLoginUILoginType	Grundlegende Login Informationen
ChangePassword	MpUserXLoginUIPwdType	Dient zum Ändern des Passwortes
MessageBox	MpUserXUIMessageBoxType	Bedienung der Dialogfenster
DefaultLayerStatus	UINT	Statusdatenpunkt für den Default Layer der Visualisierungsseite, von der sich ein- und ausgeloggt werden kann VC4 Anbindung: StatusDatapoint von <i>Layer</i>
UserLevel	DINT	Level des aktuell eingeloggtten Benutzers



4.2.1.9 MpUserXLoginUILoginType

Dient zum Ein- und Ausloggen über die Visual Componets 4 Visualisierung.

Parameter	Datentyp	Beschreibung
Login	BOOL	Befehl zum Einloggen VC4 Anbindung: CompletionData-point von <i>String</i>
Logout	BOOL	Befehl zum Ausloggen VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1
UserName	WSTRING [50]	Benutzername VC4 Anbindung: Datapoint von <i>String</i>
Password	WSTRING [50]	Passwort VC4 Anbindung: Datapoint von <i>String</i>

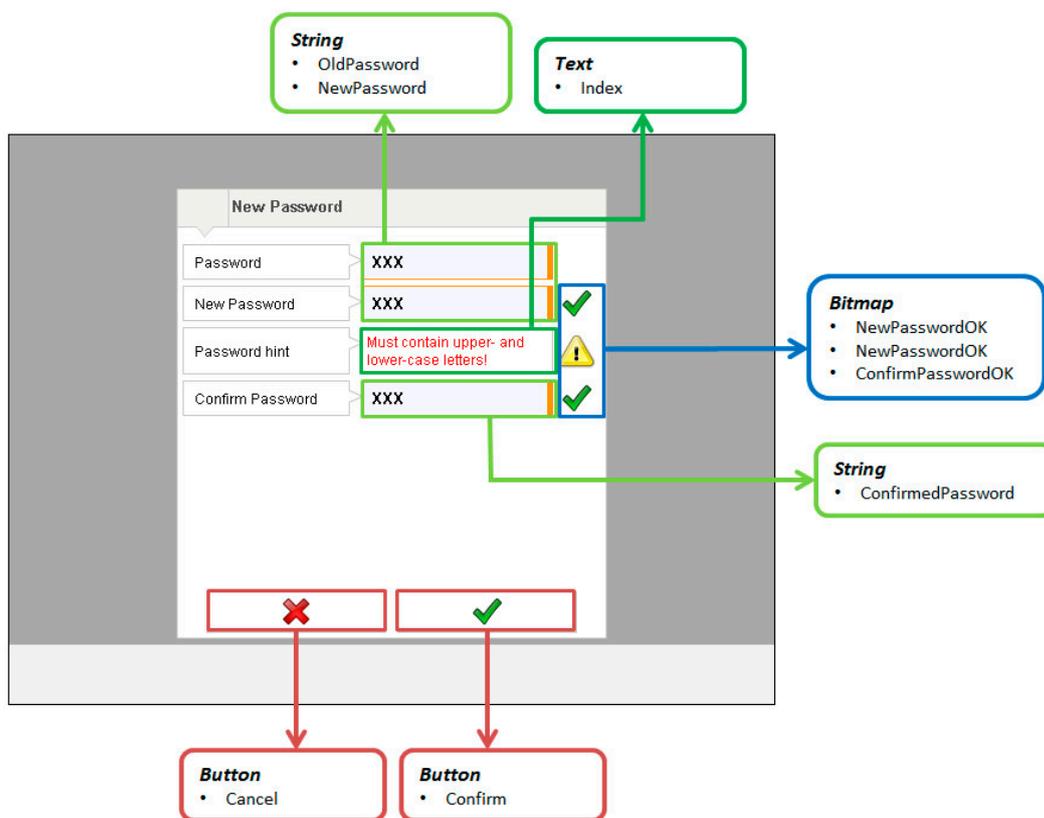


4.2.1.10 MpUserXLoginUIPwdDlgType

Datentyp, der verwendet wird, um die Anzeige eines Dialogfensters zu steuern. Über dieses Dialogfenster wird das Passwort eines Benutzers geändert.

Parameter	Datentyp	Beschreibung
OldPassword	WSTRING [50]	Altes Passwort VC4 Anbindung: Datapoint von <i>String</i>
NewPassword	WSTRING [50]	Neues Passwort VC4 Anbindung: Datapoint von <i>String</i>
NewPasswordOk	UINT	Das neue Passwort entspricht den Kriterien für ein Passwort VC4 Anbindung: Indexeddatapoint von <i>Bitmap</i>

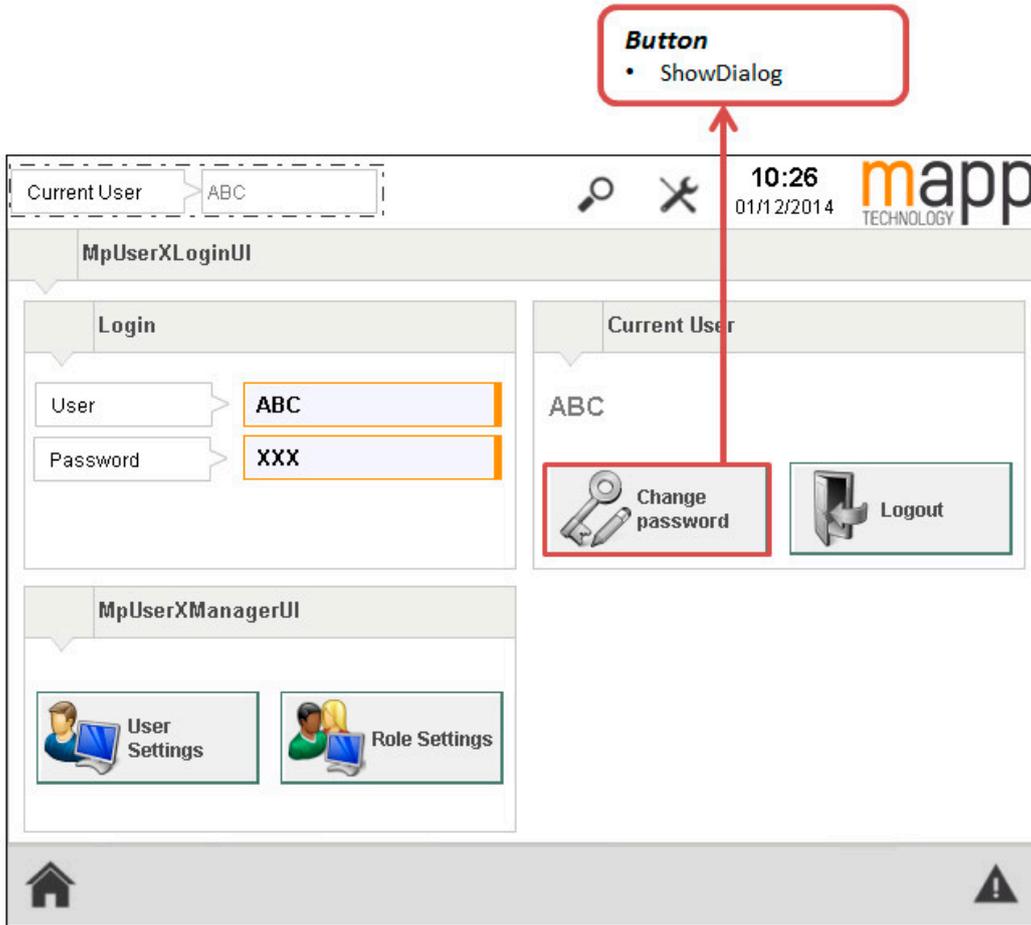
Parameter	Datentyp	Beschreibung
CriteriaNotMet	MpUserXUIPasswordCriteria-Enum	Falls das eingegebene Passwort nicht den Kriterien für ein neues Passwort entspricht, wird über diesen Parameter angezeigt was nicht passt. Bei mehreren nicht erfüllten Kriterien, wird immer nur eines über diese Parameter angezeigt VC4 Anbindung: Indexdatapoint von Text / Textgruppe: MpUserXUIError-Text
ConfirmPassword	WSTRING[50]	Bestätigung des neuen Passwortes VC4 Anbindung: Datapoint von <i>String</i>
ConfirmPasswordOk	UINT	Das Passwort zum Bestätigen der richtigen Eingabe stimmt VC4 Anbindung: Indexdatapoint von <i>Bitmap</i>
LayerStatus	UINT	Sichtbarkeit des Dialogfensters (Statusdatenpunkt für den Layer des Dialogfensters) VC4 Anbindung: StatusDatapoint von <i>Layer</i>
Confirm	BOOL	Bestätigen der Operation VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1
Cancel	BOOL	Abbrechen der Operation VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1



4.2.1.11 MpUserXLoginUIPwdType

Vereint Anzeige und Daten für ein Dialogfenster zum Passwort Ändern.

Parameter	Datentyp	Beschreibung
ShowDialog	BOOL	Befehl, der das Dialogfenster öffnet VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1
Dialog	MpUserXLoginUIPwdDlgType	Dialog zum Passwort Ändern



4.2.1.12 MpUserXMappingConfigType

Über diesen Datentypen kann das Rollen-Mapping angepasst werden.

Wie das Konfigurationsupdate der einzelnen Parameter zur Laufzeit durchgeführt wird, ist im Abschnitt [Konfiguration während der Laufzeit](#) erklärt.

Parameter	Datentyp	Beschreibung
Mapping	ARRAY[0..99] von <code>MpUserXMappingMappingType</code>	Über diesen Datentypen kann das Rollen-Mapping für den Active Directory Server angepasst werden.

4.2.1.13 MpUserXMappingMappingType

Über diesen Datentypen kann festgelegt werden, welche Active Directory Rolle auf welche lokal in Automation Studio angelegte Rolle, gemappt werden soll. Damit die Änderung übernommen wird, muss sich der Benutzer ab- und wieder anmelden.

Parameter	Datentyp	Beschreibung	Konfigurationsupdate
LocalRole	WSTRING[50]	Lokale in Automation Studio angelegte Rolle.	.
ServerGroup	WSTRING[50]	Rolle auf Active Directory Server	.

4.2.1.14 MpUserXMgrUIAdditionalDataType

Mithilfe dieses Datentypen kann man den Benutzer zusätzliche Information hinzufügen.

Parameter	Datentyp	Beschreibung
Key	WSTRING[20]	Kennzeichnung der zusätzlichen Information
Value	WSTRING[255]	Zusätzliche Information des Benutzers

4.2.1.15 MpUserXMgrUIConnectType

Dieser Datentyp enthält alle notwendigen Informationen, um eine Verbindung zur Visualisierung herzustellen.

Parameter	Datentyp	Beschreibung
Status	MpUserXUIStatusEnum	Aktuelle Operation
User	MpUserXMgrUIUserType	Benutzerinformationen
Role	MpUserXMgrUIRoleType	Rolleninformationen
MessageBox	MpUserXUIMessageBoxType	Bedienung der Dialogfenster
DefaultLayerStatus	UINT	Statusdatenpunkt für den Default Layer der Visualisierungsseite, auf der das Benutzermanagement angezeigt wird VC4 Anbindung: StatusDatapoint von <i>Layer</i>
Export	MpUserXMgrUIExportType	Exportieren der Benutzer und Rolleneinstellungen
Import	MpUserXMgrUIImportType	Importieren der Benutzer und Rolleneinstellungen

4.2.1.16 MpUserXMgrUICreateDlgType

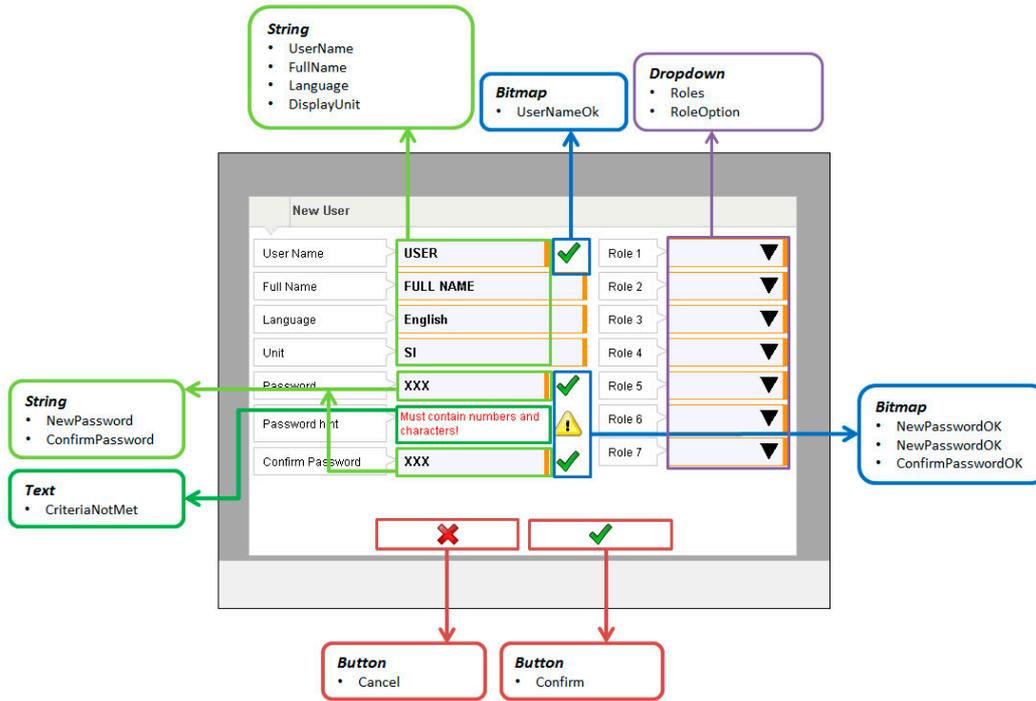
Datentyp, der verwendet wird, um die Anzeige eines Dialogfensters zu steuern. Über dieses Dialogfenster wird ein neuer Benutzer angelegt bzw. editiert.

Nur ein Benutzer mit Administrator Rechten ist in der Lage Benutzer zu erstellen, welche ebenfalls Administrator Rechte besitzen.

Beim Anlegen eines neuen Benutzers wird dem Benutzer automatisch die Rolle "Everyone" zugewiesen.

Parameter	Datentyp	Beschreibung
UserName	WSTRING[50]	Name des Benutzers. Wird ein Benutzer editiert so kann der Name nicht verändert werden. Wird ein neuer Benutzer erstellt, so kann der Benutzername hier angegeben werden. VC4 Anbindung: Datapoint von <i>String</i>
UserNameOk	UINT	Der Benutzername entspricht den Kriterien für einen Benutzernamen VC4 Anbindung: Indexdatapoint von <i>Bitmap</i>
FullName	WSTRING[100]	Voller Name des neuen Benutzers VC4 Anbindung: Datapoint von <i>String</i>
Roles	ARRAY[0..9] von UINT	Rollen, welche dem neuen Benutzer zugeordnet werden sollen. Wird mit einer Dropdown Liste verbunden, welche eine Textgruppe mit den Rollen anzeigt VC4 Anbindung: IndexDatapoint von <i>Dropdown</i> (Textgruppe für Benutzergruppen)
RoleOption	ARRAY[0..9] von USINT	Sortiert Rollen in der Dropdown Liste aus, welche nicht ausgewählt werden können VC4 Anbindung: OptionDatapoint von <i>Dropdown</i> (Textgruppe für Benutzergruppen)
NewPassword	WSTRING[50]	Neues Passwort VC4 Anbindung: Datapoint von <i>String</i>
NewPasswordOk	UINT	Das neue Passwort entspricht den Kriterien für ein Passwort VC4 Anbindung: Indexdatapoint von <i>Bitmap</i>
CriteriaNotMet	MpUserXUIPasswordCriteriaEnum	Falls das eingegeben Passwort nicht den Kriterien für ein neues Passwort entspricht, wird über diesen Parameter angezeigt was nicht passt. Bei mehreren nicht erfüllten Kriterien, wird immer nur eines über diese Parameter angezeigt VC4 Anbindung: Indexdatapoint von <i>Text</i> / Textgruppe: MpUserXUIError-Text
ConfirmPassword	WSTRING[50]	Bestätigung des Passwortes VC4 Anbindung: Datapoint von <i>String</i>
ConfirmPasswordOk	UINT	Das Passwort zum Bestätigen der richtigen Eingabe stimmt VC4 Anbindung: Indexdatapoint von <i>Bitmap</i>
Language	STRING[20]	Bevorzugte Sprache des neuen Benutzers VC4 Anbindung: Datapoint von <i>String</i>

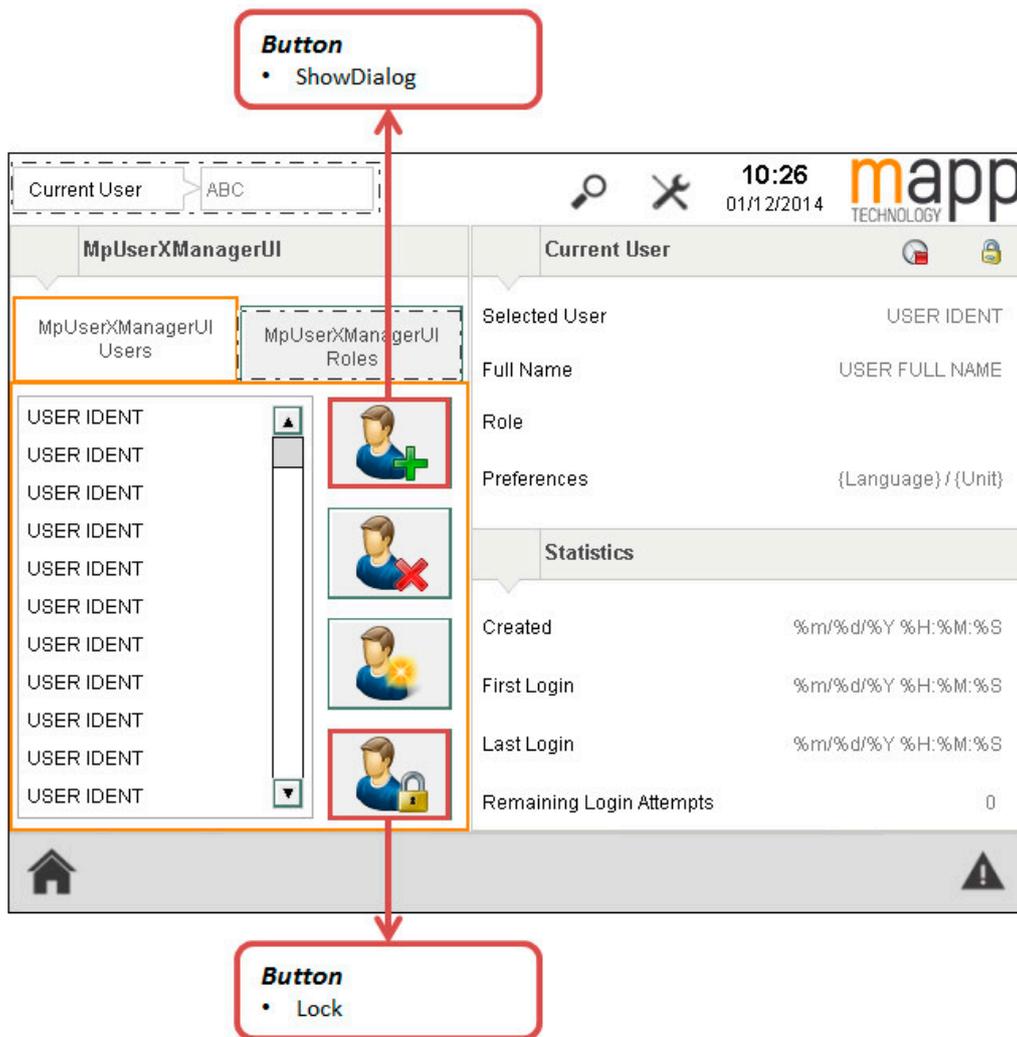
Parameter	Datentyp	Beschreibung
DisplayUnit	STRING[20]	Bevorzugte Einheit des neuen Benutzers VC4 Anbindung: Datapoint von <i>String</i>
LayerStatus	UINT	Sichtbarkeit des Dialogfensters (Statusdatenpunkt für den Layer des Dialogfensters) VC4 Anbindung: StatusDatapoint von <i>Layer</i>
Confirm	BOOL	Bestätigen der Operation VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1
Cancel	BOOL	Abbrechen der Operation VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1
AdditionalData	ARRAY[0..9] von <i>MpUserXMgrUI-AdditionalDataType</i>	Zusätzliche Information des Benutzers



4.2.1.17 MpUserXMgrUICreateType

Vereint Anzeige und Daten für ein Dialogfenster um einen Benutzer zu erstellen und zu editieren.

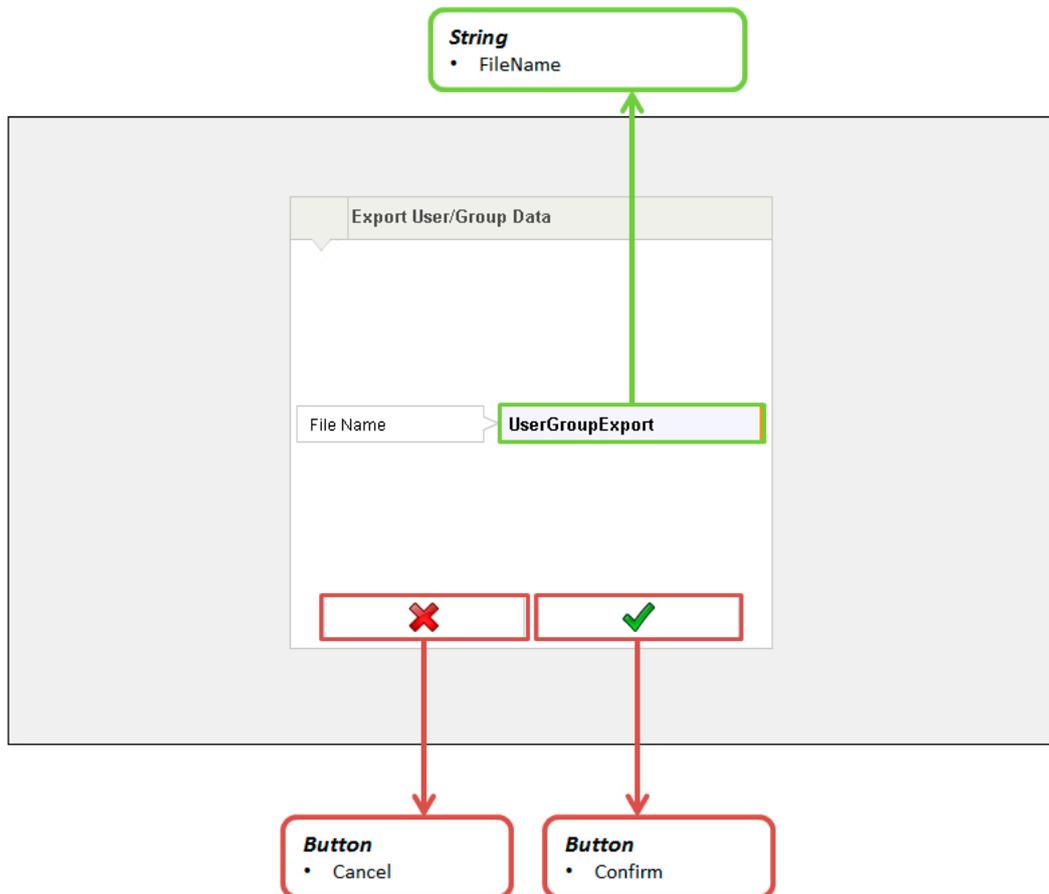
Parameter	Datentyp	Beschreibung
ShowDialog	BOOL	Befehl, der das Dialogfenster öffnet VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1
Lock	BOOL	Sperrt die Funktion Benutzer zu Editieren / zu Erstellen VC4 Anbindung: LockingDatapoint von <i>Button</i> / Locking: Datapoint >= Level / Level: 1
Dialog	<i>MpUserXMgrUICreateDlgType</i>	Dialog zum Benutzer Erstellen



4.2.1.18 MpUserXMgrUIExportDlgType

Datentyp, der verwendet wird, um die Anzeige eines Dialogfensters zu steuern. Über dieses Dialogfenster wird eine Datei exportiert, welche alle Benutzerdaten und Benutzergruppeneinstellungen enthält.

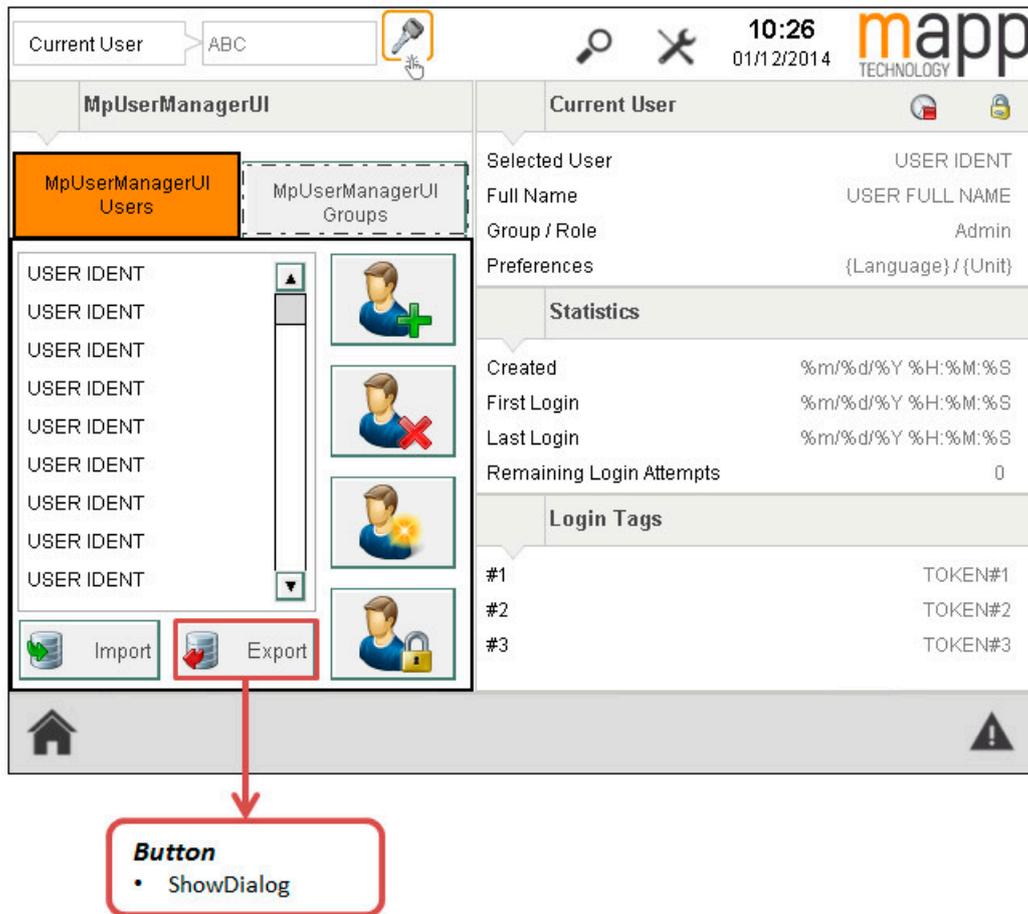
Parameter	Datentyp	Beschreibung
FileName	STRING [255]	Name der Datei, welche exportiert werden soll. Die Datei kann auch in einem Ordner auf dem angegebenen Speichermedium abgelegt werden. Dabei muss der Ordnername mitangegeben werden, wie zum Beispiel "Ordnername/FileName" VC4 Anbindung: Datapoint von <i>String</i>
LayerStatus	UINT	Sichtbarkeit des Dialogfensters (Statusdatenpunkt für den Layer des Dialogfensters) VC4 Anbindung: StatusDatapoint von <i>Layer</i>
Confirm	BOOL	Bestätigen der Operation VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1
Cancel	BOOL	Abbrechen der Operation VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1



4.2.1.19 MpUserXMgrUIExportType

Vereint Anzeige und Daten für ein Dialogfenster zum Exportieren der Benutzerdaten.

Parameter	Datentyp	Beschreibung
ShowDialog	BOOL	Befehl, der das Dialogfenster öffnet VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1
Dialog	MpUserXMgrUIExportDlgType	Dialog zum Exportieren der Benutzerdaten



4.2.1.20 MpUserXMgrUIImportConfirmType

Datentyp, der verwendet wird, um die Anzeige eines Dialogfensters zu steuern.

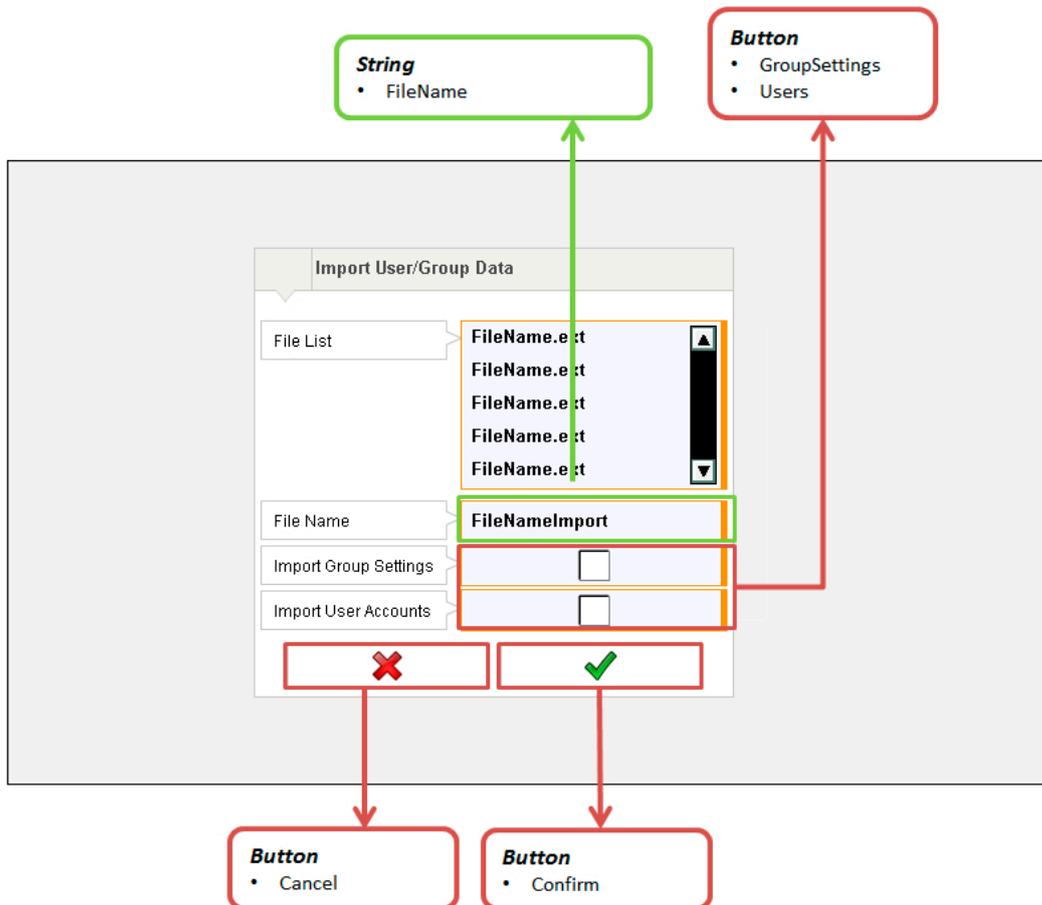
Parameter	Datentyp	Beschreibung
LayerStatus	UINT	Sichtbarkeit des Dialogfensters (Statusdatenpunkt für den Layer des Dialogfensters) VC4 Anbindung: StatusDatapoint von Layer
Type	MpUserXMgrUIImportConfirmEnum	Gibt an ob Benutzer oder Rollen überschrieben werden sollen
Name	WSTRING[50]	Name des Benutzers oder der Rolle
NumberOfConflicts	UINT	Anzahl an vorhandenen Konflikten
ApplyForAll	BOOL	Auswahl für alle vorhandenen Konflikte übernehmen
Confirm	BOOL	Bestätigen der Operation VC4 Anbindung: Datapoint von Button / Type: SetDatapoint / SetValue: 1
Cancel	BOOL	Abbrechen der Operation VC4 Anbindung: Datapoint von Button / Type: SetDatapoint / SetValue: 1

4.2.1.21 MpUserXMgrUIImportDlgType

Datentyp, der verwendet wird, um die Anzeige eines Dialogfensters zu steuern. Über dieses Dialogfenster werden Benutzerdaten und Benutzerrolleneinstellungen importiert. Bei der Auswahl der zu importierenden Datei wird der Name in "FileName" angegeben oder die gewünschte Datei wird aus der Liste ausgewählt.

Parameter	Datentyp	Beschreibung
List	MpUserXMgrUIImportListType	Liste aller zum Import verfügbaren Dateien, sowie die Navigation für die Liste
FileName	STRING[255]	Name der Datei, welche importiert werden soll VC4 Anbindung: Datapoint von String
Users	BOOL	Option, die angibt, ob Benutzerdaten übernommen werden sollen VC4 Anbindung: Datapoint von Button / Type: ToggleDatapoint / SetValue: 1 / ResetValue: 0

Parameter	Datentyp	Beschreibung
Roles	BOOL	Option, die angibt, ob Rollen übernommen werden sollen VC4 Anbindung: Datapoint von <i>Button</i> / Type: ToggleDatapoint / SetValue: 1 / ResetValue: 0
LayerStatus	UINT	Sichtbarkeit des Dialogfensters (Statusdatenpunkt für den Layer des Dialogfensters) VC4 Anbindung: StatusDatapoint von <i>Layer</i>
Confirm	BOOL	Bestätigen der Operation VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1
Cancel	BOOL	Abbrechen der Operation VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1



4.2.1.22 MpUserXMgrUIImportListType

Dieser Datentyp zeigt eine Liste mit Dateien an, welche importiert werden können.

Es ist nötig die Bibliothek MpFile auf die Steuerung zu übertragen, um bei der "Import" Funktion von MpUserXManagerUI eine Liste mit möglichen Dateien anzuzeigen.

Parameter	Datentyp	Beschreibung
FileNames	ARRAY[0..4] von STRING[255]	Liste an Dateien, welche importiert werden können
SelectedIndex	UINT	Index der aktuell ausgewählten Datei
MaxSelection	UINT	Index des letzten Eintrages in der Liste

4.2.1.23 MpUserXMgrUIImportType

Vereint Anzeige und Daten für ein Dialogfenster zum Importieren der Benutzerdaten.

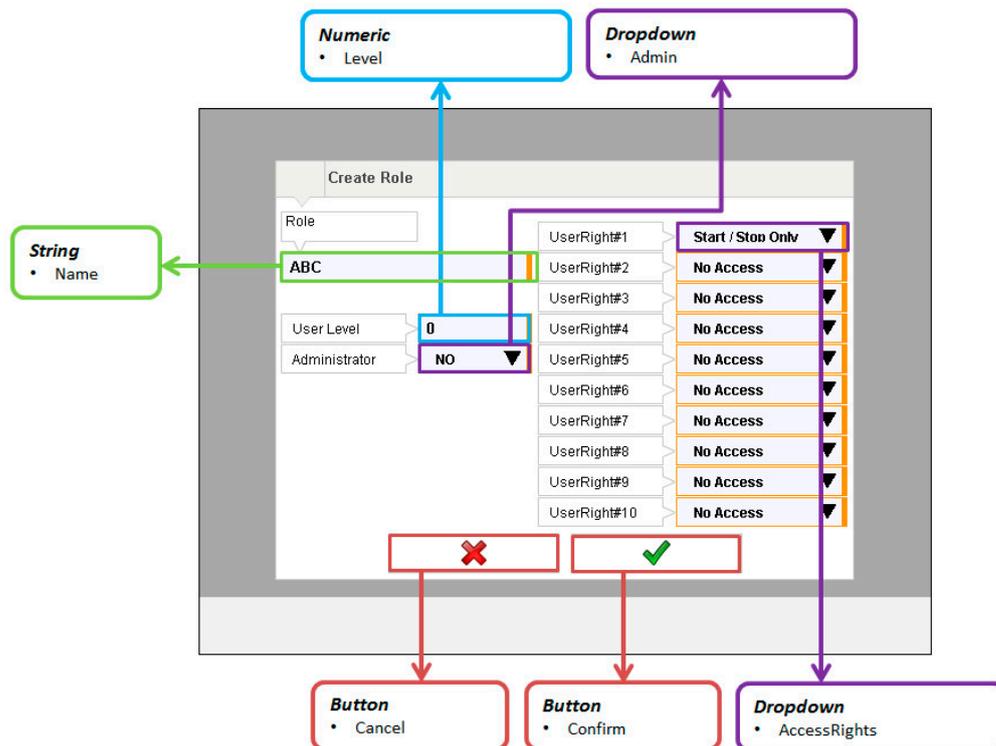
Die "Import" Funktion ist ab der Automation Runtime Version 4.3x (LTS Version) verfügbar

Parameter	Datentyp	Beschreibung
ShowDialog	BOOL	Befehl der das Dialogfenster öffnet VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1
Dialog	MpUserXMgrUIImportDlgType	Dialog zum Importieren der Benutzerdaten
ConfirmDialog	MpUserXMgrUIImportConfirmType	Dialog zum Bestätigen des Imports

4.2.1.24 MpUserXMgrUIRoleCreateDlgType

Datentyp, der verwendet wird, um die Anzeige eines Dialogfensters zu steuern. Über dieses Dialogfenster wird eine neue Rolle angelegt.

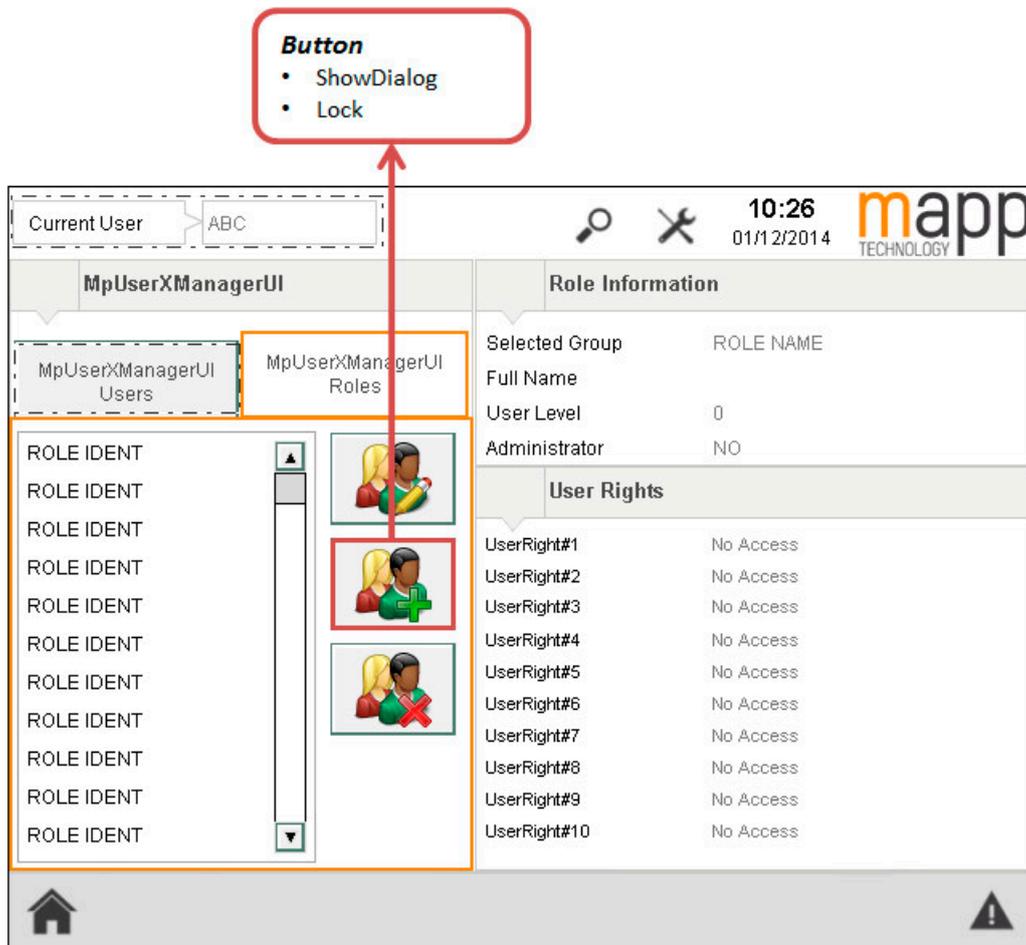
Parameter	Datentyp	Beschreibung
Name	WSTRING[50]	Rollenname VC4 Anbindung: Datapoint von <i>String</i>
Level	DINT	Level dieser Rolle VC4 Anbindung: IndexDatapoint von <i>Dropdown</i> (Textgruppe für Benutzergruppen)
LevelLimit	DINT	Maximal erlaubtes Level, welches für diese Rolle eingegeben werden kann (hängt vom aktuell eingeloggten Benutzer ab)
Admin	BOOL	Definiert, ob die neue Rolle Administrator Rechte besitzt VC4 Anbindung: Indexdatapoint von <i>Dropdown</i>
AccessRights	ARRAY[0..19] von MpUserXAccessRightEnum	Bestimmt, welche Rechte die neue Rolle besitzt VC4 Anbindung: OptionDatapoint von <i>Dropdown</i> (Textgruppe für Benutzergruppen)
AccessRightsLimit	ARRAY[0..19] von MpUserXAccessRightEnum	Mögliche Rechte, welche für diese Rolle eingegeben werden kann (hängt vom aktuell eingeloggten Benutzer ab)
LayerStatus	UINT	Sichtbarkeit des Dialogfensters (Statusdatenpunkt für den Layer des Dialogfensters) VC4 Anbindung: StatusDatapoint von <i>Layer</i>
Confirm	BOOL	Bestätigen der Operation VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1
Cancel	BOOL	Abbrechen der Operation VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1



4.2.1.25 MpUserXMgrUIRoleCreateType

Vereint Anzeige und Daten für ein Dialogfenster zum Rollen Erstellen.

Parameter	Datentyp	Beschreibung
ShowDialog	BOOL	Befehl, der das Dialogfenster öffnet VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1
Lock	BOOL	Sperrt die Funktion neue Rollen zu Erstellen VC4 Anbindung: LockingDatapoint von <i>Button</i> / Locking: Datapoint >= Level / Level: 1
Dialog	MpUserXMgrUIRoleCreateDlgType	Dialog zum Rollen Erstellen

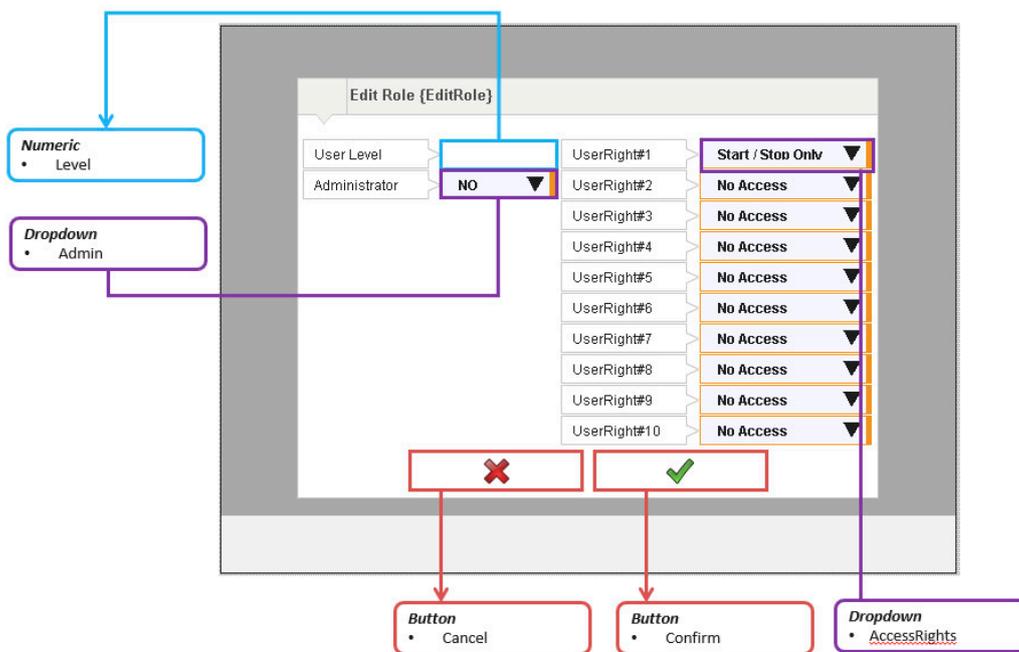


4.2.1.26 MpUserXMgrUIRoleDlgType

Datentyp, der verwendet wird, um die Anzeige eines Dialogfensters zu steuern. Über dieses Dialogfenster wird eine Rolle editiert.

Parameter	Datentyp	Beschreibung
Name	WSTRING[50]	Name der Rolle
Index	UINT	Index der Rolle VC4 Anbindung: Datapoint von <i>Numeric</i>
Level	DINT	Level der Rolle VC4 Anbindung: Datapoint von <i>Numeric</i>
LevelLimit	DINT	Maximal erlaubtes Level, das für diese Rolle gesetzt werden darf
Admin	BOOL	Bestimmt, ob diese Rolle Administratorrechte besitzt VC4 Anbindung: IndexDatapoint von <i>Dropdown</i> (Textgruppe für Adminrecht)
AccessRights	ARRAY[0..19] von MpUserXAccessRightEnum	Liste mit den Benutzerrechten für einzelne Aktionen VC4 Anbindung: IndexDatapoint von <i>Dropdown</i> (Textgruppe für Zugriffsrechte)

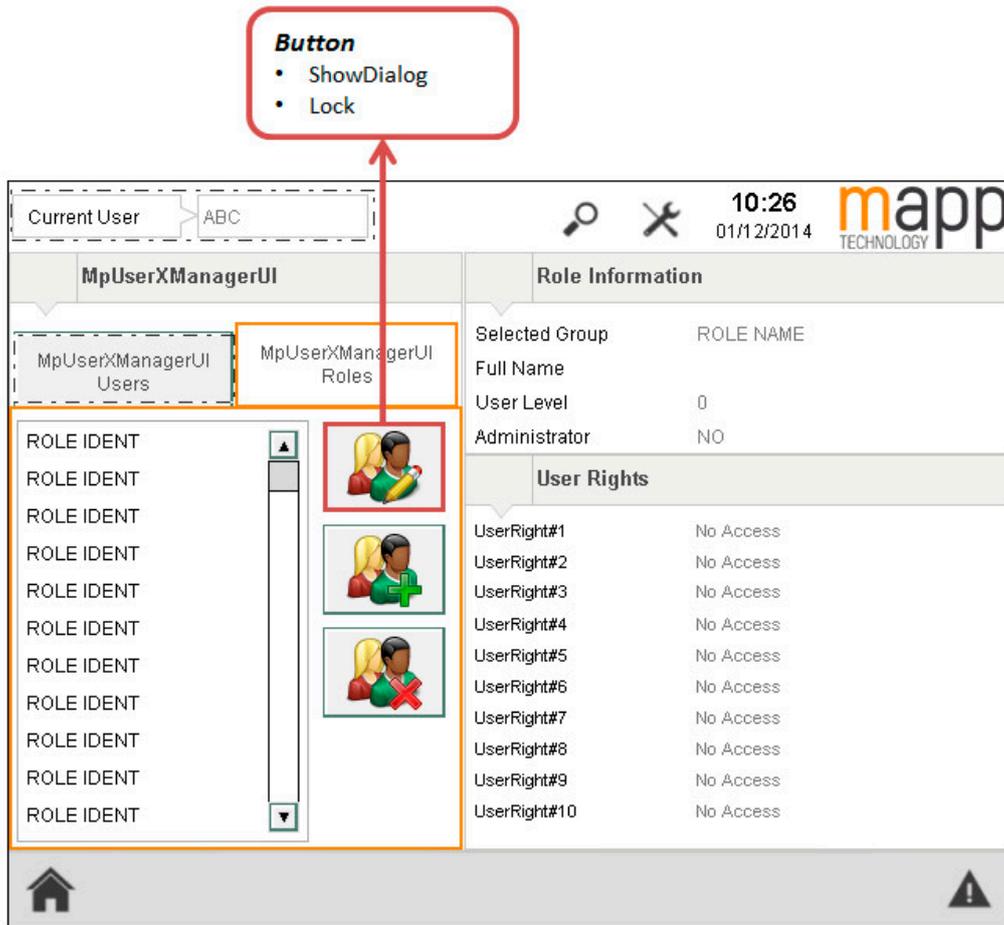
Parameter	Datentyp	Beschreibung
AccessRightsLimit	ARRAY[0..19] von MpUserXAccessRightEnum	Limits für die Eingabe der Benutzerrechte
LayerStatus	UINT	Sichtbarkeit des Dialogfensters (Statusdatenpunkt für den Layer des Dialogfensters) VC4 Anbindung: StatusDatapoint von Layer
Confirm	BOOL	Bestätigen der Operation VC4 Anbindung: Datapoint von Button / Type: SetDatapoint / SetValue: 1
Cancel	BOOL	Abbrechen der Operation VC4 Anbindung: Datapoint von Button / Type: SetDatapoint / SetValue: 1



4.2.1.27 MpUserXMgrUIRoleEditType

Vereint Anzeige und Daten für ein Dialogfenster zum Editieren einer Rolle.

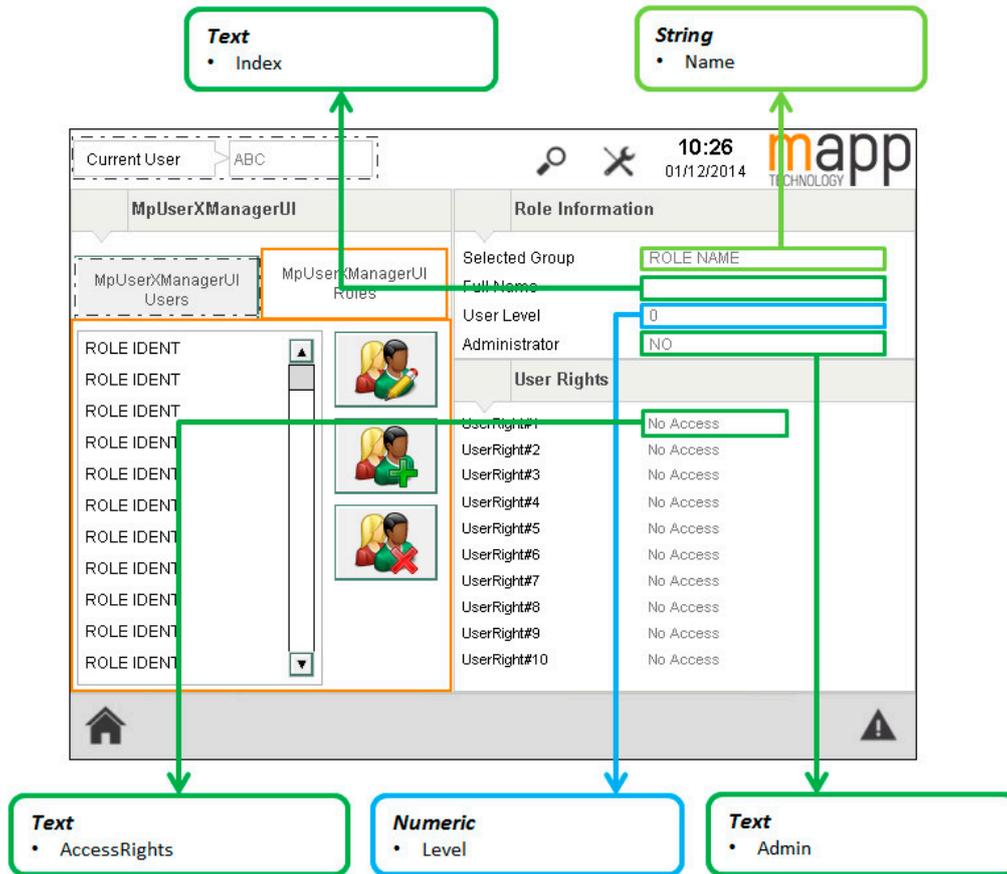
Parameter	Datentyp	Beschreibung
ShowDialog	BOOL	Befehl der das Dialogfenster öffnet VC4 Anbindung: Datapoint von Button / Type: SetDatapoint / SetValue: 1
Lock	BOOL	Sperrt die Funktion Rollen zu Editieren VC4 Anbindung: LockingDatapoint von Button / Locking: Datapoint >= Level / Level: 1
Dialog	MpUserXMgrUIRoleDlgType	Dialog zum Editieren einer Rolle



4.2.1.28 MpUserXMgrUIRoleInfoType

Gibt Auskunft über eine Rolle.

Parameter	Datentyp	Beschreibung
Name	WSTRING[50]	Name der Rolle VC4 Anbindung: Datapoint von <i>String</i>
Index	UINT	Index der Rolle VC4 Anbindung: IndexDatapoint von <i>Text</i> (Textgruppe für Benutzergruppen)
Level	DINT	Level der Rolle VC4 Anbindung: Datapoint von <i>Numeric</i>
Admin	BOOL	Zeigt an, ob diese Rolle Administratorrechte besitzt VC4 Anbindung: IndexDatapoint von <i>Text</i> (Textgruppe für Adminrecht)
AccessRights	ARRAY[0..19] von <i>MpUserXAccessRightEnum</i>	Liste mit den Benutzerrechten für einzelne Aktionen VC4 Anbindung: IndexDatapoint von <i>Text</i> (Textgruppe für Zugriffsrechte)



4.2.1.29 MpUserXMgrUIRoleListType

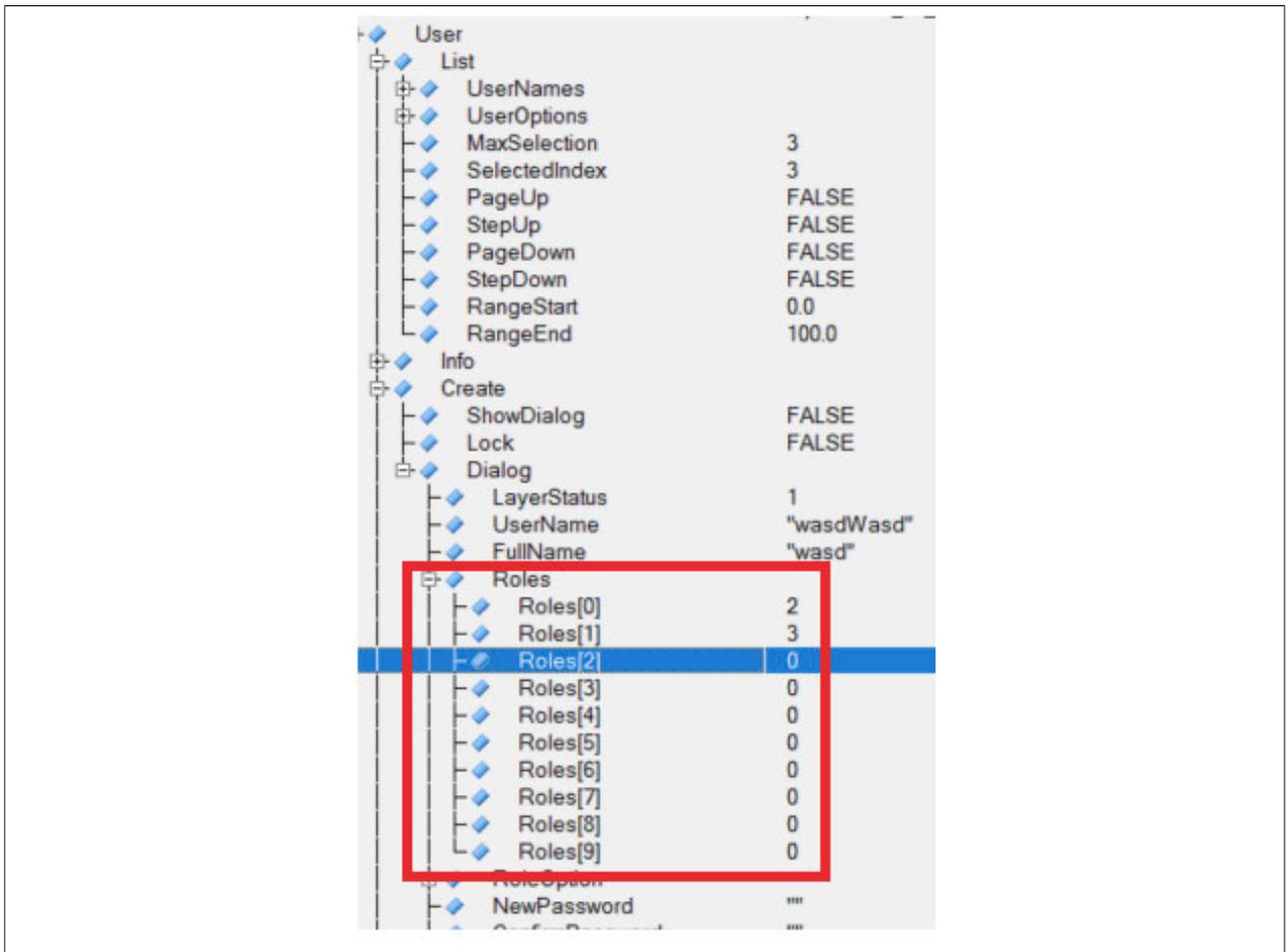
Liste mit möglichen Rollen und die dazugehörige Navigation.

Parameter	Datentyp	Beschreibung
Names	ARRAY[0..9] von WSTRING[50]	Liste aller verfügbaren Rollen VC4 Anbindung: TextDatapoint von <i>Listbox</i>
SelectedIndex	UINT	Index des aktuell ausgewählten Eintrages in der Liste. Startet automatisch mit Index 1 VC4 Anbindung: IndexDatapoint von <i>Listbox</i>
MaxSelection	UINT	Index des letzten Eintrages in der Liste VC4 Anbindung: MaxDatapoint von <i>Listbox</i>
PageUp	BOOL	Springt an den Anfang der aktuellen Seite und danach immer eine Seite weiter nach oben. VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1 / ResetValue: 0
StepUp	BOOL	Wählt vorhergehenden Eintrag in der Liste aus VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1 / ResetValue: 0

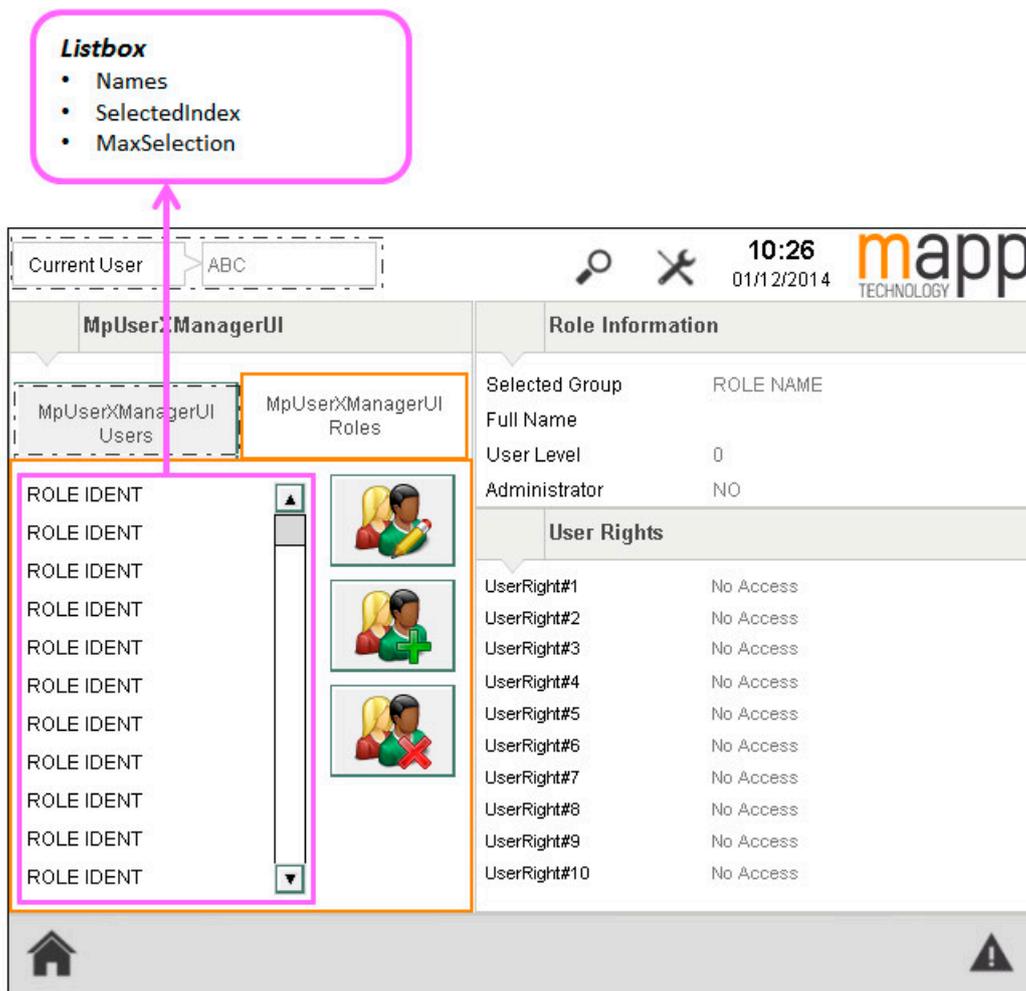
Parameter	Datentyp	Beschreibung
PageDown	BOOL	Springt ans Ende der aktuellen Seite und danach immer eine Seite weiter nach unten. VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1 / ResetValue: 0
StepDown	BOOL	Wählt nächsten Eintrag in der Liste aus VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1 / ResetValue: 0
RangeStart	REAL	Zeigt in Balkenform an, welcher Teil der Liste gerade angezeigt wird. Wird für den Startwert der "Range" eines Scale Elements der Visualisierung verwendet. Diese Scale Element sollte mit der (möglichen) Größe der Liste korrespondieren. Die "Range" sollte mit der Anzahl an Einträgen, die auf einer Seite angezeigt werden, korrespondieren VC4 Anbindung: StartDatapoint von <i>Scale</i>
RangeEnd	REAL	Zeigt in Balkenform an, welcher Teil der Liste gerade angezeigt wird. Wird für den Endwert der "Range" eines Scale Elements der Visualisierung verwendet. Diese Scale Element sollte mit der (möglichen) Größe der Liste korrespondieren. Die "Range" sollte mit der Anzahl an Einträgen, die auf einer Seite angezeigt werden, korrespondieren VC4 Anbindung: EndDatapoint von <i>Scale</i>

Bei der Liste an verfügbaren Rollen ("Names") muss folgendes beachtet werden:

Die Rolle an Index 0 bleibt immer leer. Grund dafür ist, dass wenn neue Benutzer angelegt wird und keine Rolle definiert wird, die Rolle "Keine Rolle"(= Index 0) verwendet wird:



Aus diesem Grund ist auch der Parameter "SelectedIndex" automatisch 1, da sich keine Rolle an Names[0] befindet.



4.2.1.30 MpUserXMgrUIRoleSelectType

Über diesen Datentyp werden die Namen der Rollen und die maximale Anzahl an Rollen angezeigt

Parameter	Datentyp	Beschreibung
Names	ARRAY[0..19] von WSTRING[50]	Liste mit vorhandenen Rollen
MaxSelection	UINT	Maximale Anzahl an Rollen

4.2.1.31 MpUserXMgrUIRoleType

Mit diesem Datentyp wird eine Liste existierender Rollen angezeigt.

Parameter	Datentyp	Beschreibung
List	MpUserXMgrUIRoleListType	Liste aller Rollen, sowie die Navigation für die Liste
Info	MpUserXMgrUIRoleInfoType	Enthält detaillierte Rolleninformationen
Edit	MpUserXMgrUIRoleEditType	Beinhaltet die Daten, welche zum Editieren einer Rolle notwendig sind
Create	MpUserXMgrUIRoleCreateType	Beinhaltet die Daten, welche zum Erstellen einer neuen Rolle notwendig sind
Remove	BOOL	Löschen der ausgewählten Rolle
SelectList	MpUserXMgrUIRoleSelectType	Liste an vorhandenen Rollen

4.2.1.32 MpUserXMgrUISetupConfirmType

Über diesen Datentyp kann definiert werden, wann Bestätigungs Dialoge angezeigt werden sollen.

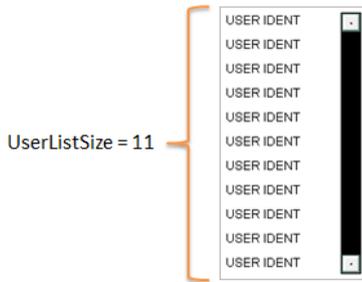
Parameter	Datentyp	Wert	Beschreibung
ConfirmOverwriteUser	BOOL	FALSE	Das überschreiben von Benutzern soll über einen Dialog bestätigt werden
ConfirmOverwriteRole	BOOL	FALSE	Das überschreiben von Rollen soll über einen Dialog bestätigt werden

4.2.1.33 MpUserXMgrUISetupType

Zusätzliche Konfigurationsmöglichkeiten, um die Visualisierung zu beschreiben.

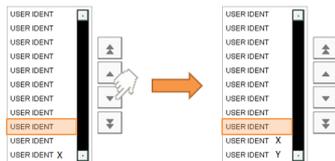
Parameter	Datentyp	Wert	Beschreibung
UserListSize	UINT	20	Zahl an Benutzern, die auf einer Seite der Liste auf der Visualisierung dargestellt werden können Es können maximal 20 Benutzer angezeigt werden
RoleListSize	UINT	9	Zahl an Rollen, die auf einer Seite der Liste auf der Visualisierung dargestellt werden können Es können maximal 20 Rollen angezeigt werden
ScrollWindow	USINT	1	Zeigt an, wieviele Einträge der Liste beim auf- oder abscrollen im Vorhinein angezeigt werden
FileDevice	STRING[20]	'HD'	File Device (Speichermedium) zum Exportieren bzw. Importieren der Benutzerdaten
FileExtension	STRING[20]	'usr'	Dateiendung für die Export- bzw. Importdateien
Confirmation	MpUserXMgrUISetupConfirmType		Anzeige der Bestätigungs Dialoge

UserListSize



UserListSize gibt an, wie viele User auf einer Seite einer Listbox im VC4 angezeigt werden. Die Listbox in der Visualisierung zeigt je nach Größe unterschiedlich viele Elemente an. Wie viele Elemente die Listbox anzeigt, kann durch definieren eines Simulationwertes (SimulationValue) in den Listbox Eigenschaften ermittelt werden (siehe Bild oben: "USER IDENT"). Die Anzahl dieser Elemente muss mit dem Wert von UserListSize übereinstimmen.

ScrollWindow



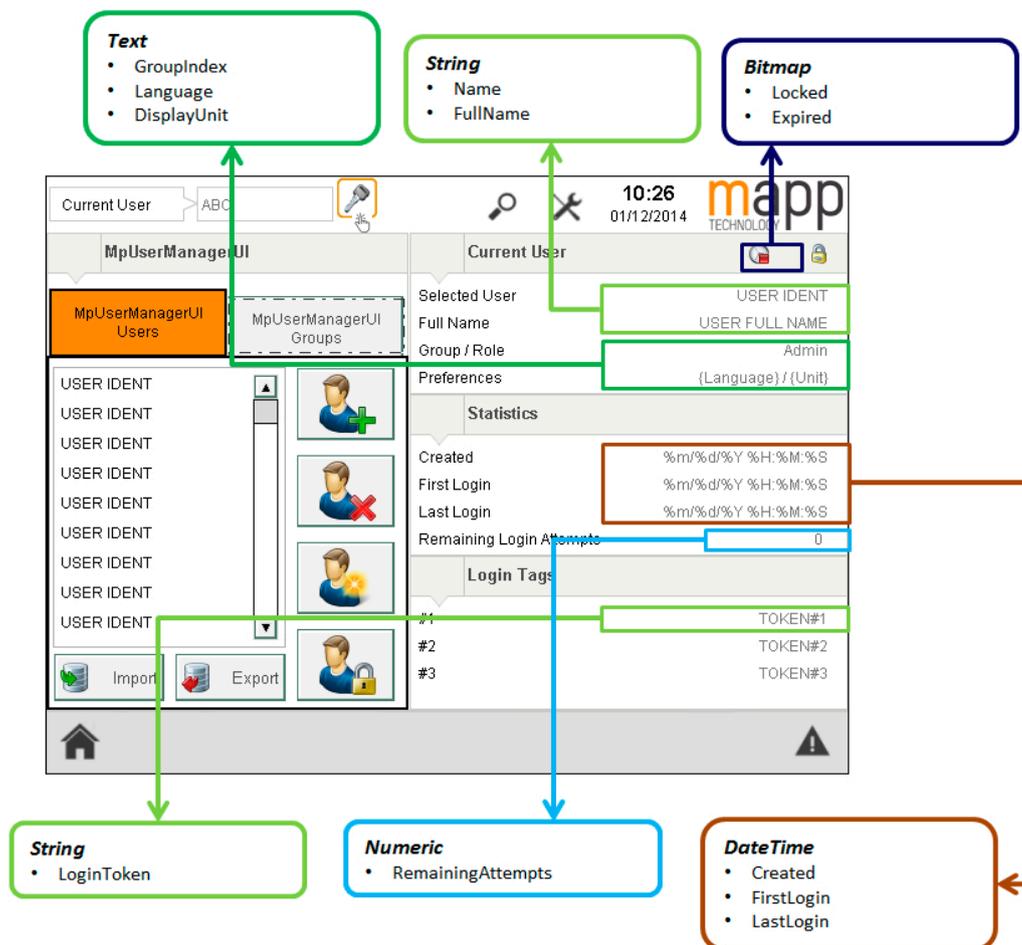
ScrollWindow gibt an, wie viele Einträge beim Scrollen "vorausschauend" angezeigt werden sollen. In diesem Beispiel wurde ScrollWindow = 2 definiert. Befindet sich nun die aktuelle Auswahl an 3. letzter Stelle in der Liste, so wird bei erneutem Drücken des StepDown Buttons der nächste Eintrag ausgewählt und vorausschauend die nächsten 2 Einträge angezeigt.

4.2.1.34 MpUserXMgrUIUserInfoType

Gibt Auskunft über einen Benutzer.

Parameter	Datentyp	Beschreibung
UserName	WSTRING[50]	Benutzername fürs Einloggen VC4 Anbindung: Datapoint von <i>String</i>
FullName	WSTRING[100]	Vollständiger Benutzername VC4 Anbindung: Datapoint von <i>String</i>
Roles	ARRAY[0..9] of UINT	Rollen, welche der Benutzer besitzt VC4 Anbindung: IndexDatapoint von <i>Text</i> (Textgruppe für Rollen)
Locked	BOOL	Benutzer ist gesperrt VC4 Anbindung: IndexDatapoint von <i>Bitmap</i> (BitmapGroup für Gesperrt)
Expired	BOOL	Benutzerpasswort ist abgelaufen VC4 Anbindung: IndexDatapoint von <i>Bitmap</i> (BitmapGroup für Abgelaufen)

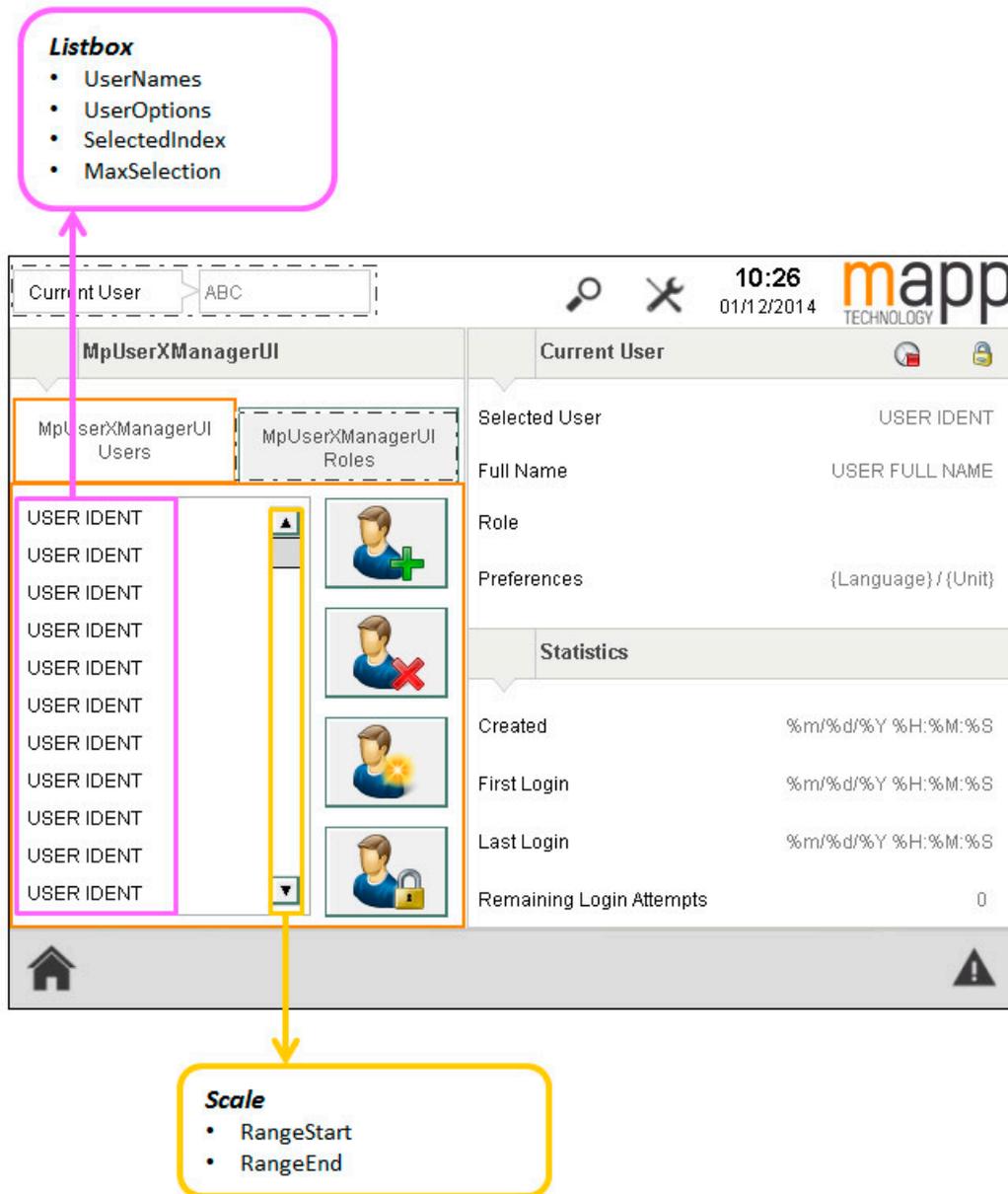
Parameter	Datentyp	Beschreibung
RemainingAttempts	USINT	Anzahl an übrigen Loginversuchen, bevor der Benutzer gesperrt wird VC4 Anbindung: Datapoint von <i>Numeric</i>
Language	STRING[20]	Bevorzugte Sprache VC4 Anbindung: IndexDatapoint von <i>Text</i> (Textgruppe für Sprachen)
DisplayUnit	STRING[20]	Bevorzugte Einheit VC4 Anbindung: IndexDatapoint von <i>Text</i> (Textgruppe für Einheiten)
Creation	DATE_AND_TIME	Datum und Uhrzeit, wann dieser Benutzer erstellt wurde VC4 Anbindung: Datapoint von <i>DateTime</i>
FirstLogin	DATE_AND_TIME	Datum und Uhrzeit des ersten Logins VC4 Anbindung: Datapoint von <i>DateTime</i>
LastLogin	DATE_AND_TIME	Datum und Uhrzeit des letzten Logins VC4 Anbindung: Datapoint von <i>DateTime</i>
PasswordExpired	DATE_AND_TIME	Gibt Datum und Uhrzeit an wann das Passwort abgelaufen ist VC4 Anbindung: Datapoint von <i>DateTime</i>
AdditionalData	MpUserXMgrUIAdditionalDataType	Zusätzliche Information des Benutzers.



4.2.1.35 MpUserXMgrUIUserListType

Liste mit möglichen Benutzern und die dazugehörige Navigation.

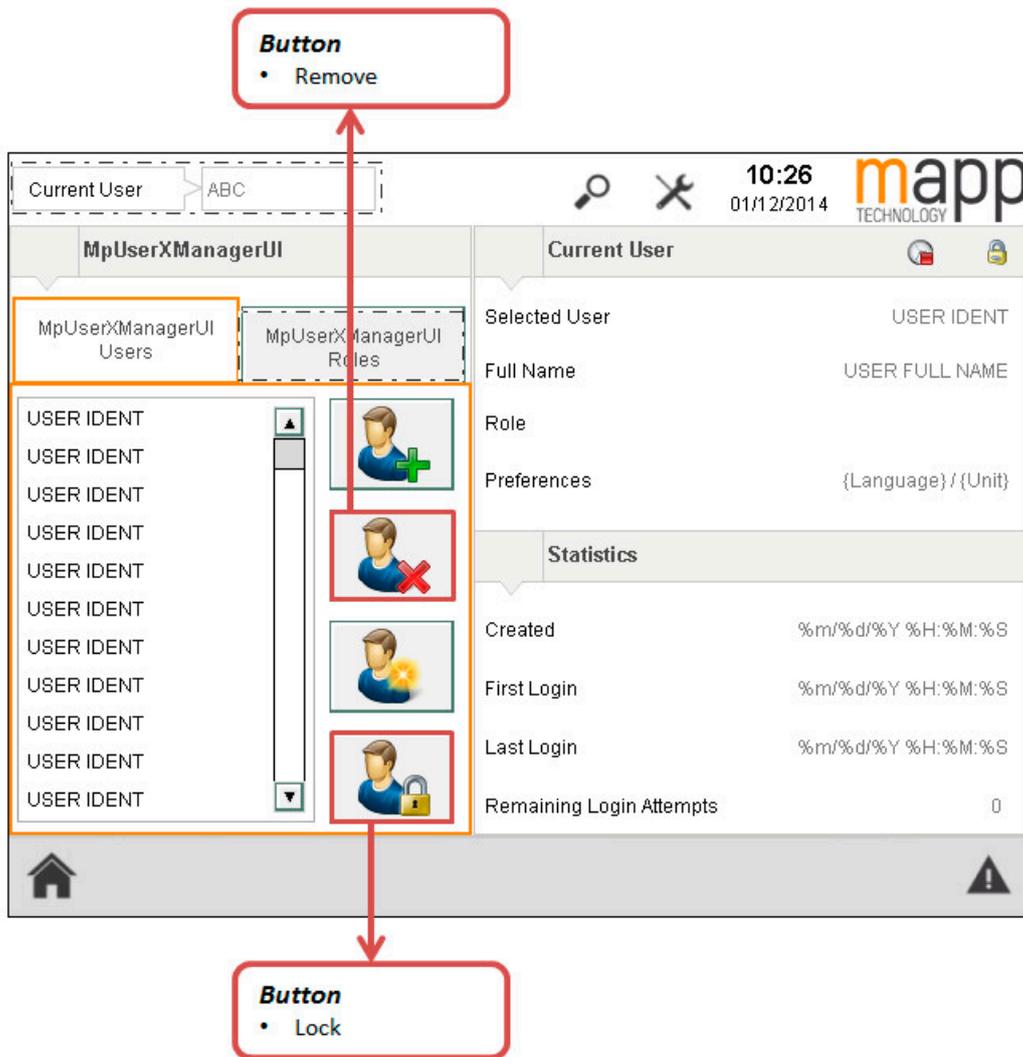
Parameter	Datentyp	Beschreibung
UserNames	ARRAY[0..19] von WSTRING[50]	Liste aller verfügbaren Benutzer VC4 Anbindung: TextDatapoint von <i>Listbox</i>
UserOptions	ARRAY[0..19] von USINT	Steuert die Anzeige der Elemente in der Liste VC4 Anbindung: OptionDatapoint von <i>Listbox</i>
SelectedIndex	UINT	Index des aktuell ausgewählten Eintrages in der Liste VC4 Anbindung: IndexDatapoint von <i>Listbox</i>
MaxSelection	UINT	Index des letzten Eintrages in der Liste VC4 Anbindung: MaxDatapoint von <i>Listbox</i>
PageUp	BOOL	Springt an den Anfang der aktuellen Seite und danach immer eine Seite weiter nach oben. Die Größe der Seite wird über den Parameter "UserListSize" der Struktur <i>MpUserXMgrUISetupType</i> definiert VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1 / ResetValue: 0
PageDown	BOOL	Springt ans Ende der aktuellen Seite und danach immer eine Seite weiter nach unten. Die Größe der Seite wird über den Parameter "UserListSize" der Struktur <i>MpUserXMgrUISetupType</i> definiert VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1 / ResetValue: 0
StepUp	BOOL	Wählt vorhergehenden Eintrag in der Liste aus VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1 / ResetValue: 0
StepDown	BOOL	Wählt nächsten Eintrag in der Liste aus VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1 / ResetValue: 0
RangeStart	REAL	Zeigt in Balkenform an, welcher Teil der Liste gerade angezeigt wird. Wird für den Startwert der "Range" eines Scale Elements der Visualisierung verwendet. Diese Scale Element sollte mit der (möglichen) Größe der Liste korrespondieren. Die "Range" sollte mit der Anzahl an Einträgen, die auf einer Seite angezeigt werden, korrespondieren VC4 Anbindung: StartDatapoint von <i>Scale</i>
RangeEnd	REAL	Zeigt in Balkenform an, welcher Teil der Liste gerade angezeigt wird. Wird für den Endwert der "Range" eines Scale Elements der Visualisierung verwendet. Diese Scale Element sollte mit der (möglichen) Größe der Liste korrespondieren. Die "Range" sollte mit der Anzahl an Einträgen, die auf einer Seite angezeigt werden, korrespondieren VC4 Anbindung: EndDatapoint von <i>Scale</i>



4.2.1.36 MpUserXMgrUIUserType

Dient zum Verwalten der Benutzer.

Parameter	Datentyp	Beschreibung
List	MpUserXMgrUIUserListType	Liste aller Benutzer, sowie die Navigation für die Liste
Info	MpUserXMgrUIUserInfoType	Enthält detaillierte Benutzerdaten
Create	MpUserXMgrUICreateType	Beinhaltet die Daten, welche zum Erstellen eines Benutzers notwendig sind
Edit	MpUserXMgrUICreateType	Beinhaltet die Daten, welche zum Editieren eines Benutzers notwendig sind
Remove	BOOL	Löschen des ausgewählten Benutzers. Es können nur Benutzer gelöscht werden, welche auch zur Laufzeit erstellt wurden. Benutzer welche in der <i>MpUserX Konfiguration</i> erstellt wurden werden nicht gelöscht sondern deaktiviert. VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1
Lock	BOOL	Sperren des ausgewählten Benutzers VC4 Anbindung: LockingDatapoint von <i>Button</i> / Locking: Datapoint >= Level / Level: 1



4.2.1.37 MpUserXServerType

Über diesen Datentypen können die Server-Einstellungen der [MpUserX Konfiguration](#) angepasst werden. Mehr Information zu den einzelnen Parametern kann [hier](#) gefunden werden.

Parameter	Datentyp	Beschreibung	Konfigurationsupdate
Type	MpUserXServerEnum	Angabe des Server-Typs.	.
Hosts	ARRAY[0..9] von MpUserXHostType	Zu verwendender Host. Die Funktion mehrere Hosts zu definieren ist für die spätere Nutzung reserviert. Aktuell kann nur ein konfigurierter Host (Host[0]) verwendet werden.	.
Mapping	MpUserXGroupToRoleMappingType	Angabe der Rollen-Mapping Tabelle.	.
AdditionalUserData	ARRAY[0..9] von STRING[100]	Angabe von zusätzlichen Benutzerdaten.	.

4.2.1.38 MpUserXServerConfigType

Über diesen Datentypen können die Einstellungen des verwendeten Servers angepasst werden.

Wie das Konfigurationsupdate der einzelnen Parameter zur Laufzeit durchgeführt wird, ist im Abschnitt [Konfiguration während der Laufzeit](#) erklärt.

Parameter	Datentyp	Beschreibung	Konfigurationsupdate
UserMgmtSystemType	MpUserXUserMgmtEnum	Modus des Benutzermanagements	.
Server	MpUserXServerType	Über diesen Parameter können die Server-Einstellungen angepasst werden.	.

4.2.1.39 MpUserXSignatureUIConnectType

Mithilfe dieses Datentypes wird eine Verbindung zwischen der [MpUserXSignatureUI](#) Komponente und der Visual Components 4 Visualisierung erstellt. Über diese Struktur kann eine elektronische Signatur über die Visualisierung durchgeführt werden.

Parameter	Datentyp	Beschreibung
Status	MpUserXUIStatusEnum	Aktuelle Operation
SignAction	DINT	Löst den Prozess rund um eine elektronische Signatur aus. Kann mit <i>Completion Datapoint</i> verbunden werden. So wird beispielsweise bei einer Eingabe für einen bestimmten Parameter der Dialog für eine elektronische Signatur geöffnet. Es wird die ActionID angegeben, für die Aktion, welche durchgeführt werden soll
Dialog	MpUserXSignatureUIDlgType	Dialog, der zur Eingabe von Benutzernamen und Passwort auffordert
MessageBox	MpUserXUIMessageBoxType	Bedienung der Dialogfenster
DefaultLayerStatus	UINT	Statusdatenpunkt für den Default Layer der Visualisierungsseite, von der sich ein- und ausgeloggt werden kann VC4 Anbindung: StatusDatapoint von <i>Layer</i>

4.2.1.40 MpUserXSignatureUIDlgType

Datentyp, der verwendet wird, um die Anzeige eines Dialogfensters zu steuern. Über dieses Dialogfenster wird eine elektronische Signatur durchgeführt.

Parameter	Datentyp	Beschreibung
UserName	WSTRING[50]	Benutzername VC4 Anbindung: Datapoint von <i>String</i>
Password	WSTRING[50]	Passwort VC4 Anbindung: Datapoint von <i>String</i>
Comment	WSTRING[100]	Kommentar, welcher dem erzeugten Ereignis im Audit Trail hinzugefügt wird VC4 Anbindung: Datapoint von <i>String</i>
LayerStatus	UINT	Sichtbarkeit des Dialogfensters (Statusdatenpunkt für den Layer des Dialogfensters) VC4 Anbindung: StatusDatapoint von <i>Layer</i>
Confirm	BOOL	Bestätigen der Operation VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1
Cancel	BOOL	Abbrechen der Operation VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1

4.2.1.41 MpUserXStatusIDType

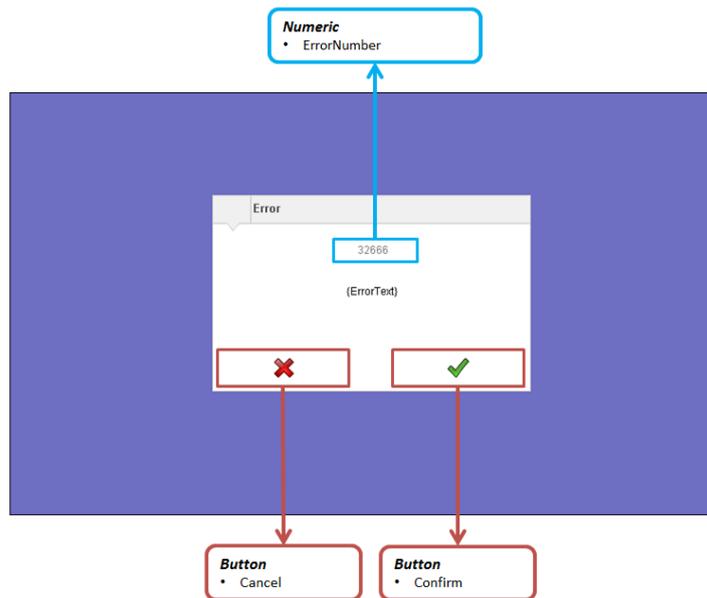
Dieser Datentyp wird in der Struktur für zusätzliche Informationen als Unterstruktur für die Diagnose verwendet und liefert weitere Infos über die StatusID.

Parameter	Datentyp	Beschreibung
ID	MpUserXErrorEnum	Fehlercode der mapp Komponente
Severity	MpComSeveritiesEnum	Beschreibt welche Art Info die StatusID liefert (Erfolg, Information, Warnung, Fehler)
Code	UINT	Kode der Status ID. Diese Fehlernummer kann in der Hilfe für weitere Informationen gesucht werden

4.2.1.42 MpUserXUIMessageBoxType

Datentyp, der verwendet wird, um die Anzeige eines Dialogfensters zu steuern.

Parameter	Datentyp	Beschreibung
LayerStatus	UINT	Sichtbarkeit des Dialogfensters (Statusdatenpunkt für den Layer des Dialogfensters) VC4 Anbindung: StatusDatapoint von <i>Layer</i>
Type	MpUserXUIMessageEnum	Art des Dialogfensters
ErrorNumber	UINT	Aktuelle Fehlernummer. Mehr Information zu den Fehlernummern kann in den Beschreibungen der einzelnen UI-Funktionsbausteine gefunden werden: <ul style="list-style-type: none"> • MpUserXLoginUI • MpUserXManagerUI • MpUserXSignatureUI VC4 Anbindung: Datapoint von <i>Numeric</i>
Confirm	BOOL	Bestätigen der Operation VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1
Cancel	BOOL	Abbrechen der Operation VC4 Anbindung: Datapoint von <i>Button</i> / Type: SetDatapoint / SetValue: 1



4.2.2 Enumeratoren

4.2.2.1 MpUserXImportModeEnum

Dieser Aufzählungsdantentyp zeigt die verschiedenen Import Modi an

Aufzählung	Beschreibung
mpUSERX_IMPORT_SKIP	Es werden keine Daten importiert
mpUSERX_IMPORT_IGNORE_EXISTING	Es werden alle neuen Daten (Benutzer/Rollen) importiert. Bereits bestehende Daten werden nicht überschrieben
mpUSERX_IMPORT_OVERWRITE	Es werden alle Daten importiert. Bereits bestehende Daten werden überschrieben. Default Benutzer, welche in der Konfiguration MpUserX angelegt wurden, werden dabei auch überschrieben und sind somit nicht mehr in der Konfiguration als Default Benutzer vorhanden
mpUSERX_IMPORT_OVERWRITE_ONLY	Es werden alle bereits existierenden Daten überschrieben. Neue Daten werden nicht importiert. Default Benutzer, welche in der Konfiguration MpUserX angelegt wurden, werden dabei auch überschrieben und sind somit nicht mehr in der Konfiguration als Default Benutzer vorhanden
mpUSERX_IMPORT_REMOVE_EXISTING	Es werden alle Daten importiert. Bereits vorhandene Daten, welche aber nicht in der importierten Datei vorkommen, werden gelöscht

4.2.2.2 MpUserXAccessRightEnum

Definiert die Zugriffsrechte der einzelnen Benutzer.

Aufzählung	Beschreibung
mpUSERX_ACCESS_UNDEFINED	Zugriffsrecht nicht definiert
mpUSERX_ACCESS_NONE	Keine Zugriffsrechte
mpUSERX_ACCESS_VIEW	Lesezugriff
mpUSERX_ACCESS_ACTUATE	Prozesse können gestartet / gestoppt werden, aber nicht editiert.
mpUSERX_ACCESS_FULL	Volle Zugriffsrechte (Lese-/Schreibzugriff)

4.2.2.3 MpUserXMgrUIImportConfirmEnum

Dieser Aufzählungsdatentyp zeigt an, ob alle vorhandenen Benutzer oder Rollen überschrieben werden sollen.

Aufzählung	Beschreibung
mpUSERX_CONFIRM_USER	Überschreiben aller vorhandenen Benutzer
mpUSERX_CONFIRM_ROLE	Überschreiben aller vorhandenen Rollen

4.2.2.4 MpUserXServerEnum

Dieser Enumerator gibt an welcher Server-Typ verwendet werden soll.

Aufzählung	Beschreibung
mpUSERX_SERVER_ACTIVE_DIRECTORY	Verwendung eines Active Directory Servers.

4.2.2.5 MpUserXUIMessageEnum

Dieser Aufzählungsdatentyp zeigt an, wofür das Dialogfenster verwendet werden soll.

Aufzählung	Beschreibung
mpUSERX_MSG_NONE	Kein Dialogfenster
mpUSERX_MSG_ERROR	Dialog: Fehler
mpUSERX_MSG_CONFIRM_DELETE	Dialog: Bestätige das Löschen eines Benutzers
mpUSERX_MSG_CONFIRM_LOCK	Dialog: Bestätige das Sperren eines Benutzers
mpUSERX_MSG_CONFIRM_UNLOCK	Dialog: Bestätige das Freigeben eines Benutzers
mpUSERX_MSG_CONFIRM_ROLE_DELETE	Dialog: Bestätige das Löschen einer Rolle

4.2.2.6 MpUserXUIPasswordCriteriaEnum

Dieser Enumerator zeigt an welchen Kriterien ein neues Passwort nicht entspricht.

Aufzählung	Beschreibung
mpUSERX_PASSWORD_CRIT_NONE	Das Passwort stimmt mit allen Kriterien für ein neues Passwort überein
mpUSERX_PASSWORD_CRIT_LENGTH	Das Passwort ist zu kurz
mpUSERX_PASSWORD_CRIT_ALPHA	Das Passwort muss alphanumerische Zeichen enthalten
mpUSERX_PASSWORD_CRIT_CASE	Das Passwort muss Groß- und Kleinbuchstaben enthalten
mpUSERX_PASSWORD_CRIT_SPECIAL	Das Passwort muss Sonderzeichen enthalten

4.2.2.7 MpUserXUIStatusEnum

Dieser Aufzählungsdatentyp gibt Auskunft über die aktuelle Aktivität der Komponente.

Aufzählung	Beschreibung
mpUSERX_UI_STATE_IDLE	Aktuell ist kein Prozess aktiv
mpUSERX_UI_STATE_WAIT_DLG	Warten auf Bestätigung oder Abbruch
mpUSERX_UI_STATE_EXECUTE	Ausführen eines Befehls
mpUSERX_UI_STATE_ERROR	Die letzte Operation hat einen Fehler generiert

4.2.2.8 MpUserXUserMgmtEnum

Definiert die möglichen Modi des Benutzermanagements. Für mehr Information siehe Abschnitt [Modus des User managements](#).

Aufzählung	Beschreibung
mpUSERX_USERMGMT_LOCAL	Verwendung des lokalen Benutzermanagements
mpUSERX_USERMGMT_CENTRAL	Verwendung des zentralen Benutzermanagements
mpUSERX_USERMGMT_LOCALANDCENTRAL	Verwendung des lokalen und zentralen Benutzermanagements

4.3 Statusnummern

4.3.1 1083338786: MpUserXLogin ist noch nicht aktiv

Beschreibung:

Diese Komponente kann erst aktiv werden, wenn [MpUserXLogin](#) aktiv ist.

Reaktion:

Der Funktionsblock wird diese Information an dem Ausgang "StatusID" anzeigen.

Ursache / Abhilfe:

- [MpUserXLogin](#) aktivieren

Konstante:

mpUSERX_INF_WAIT_LOGIN_FB

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLoginUI](#)
- [MpUserXManagerUI](#)

4.3.2 -2137980917: Warnung beim Laden der Konfiguration

Beschreibung:

Warnung beim Laden der Konfigurationsdatei (in die PV). Weitere Details siehe Logger.

Reaktion:

Der Funktionsbaustein wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Konfigurationsdatei beschädigt
- Unterschied zwischen dem Inhalt der Konfigurationsdatei und der PV (z.B. mehr Daten in der Konfigurationsdatei als in der PV)

Diese Funktionsbausteine / Funktionen können diesen Fehler melden:

- [MpUserXConfig](#)
- [MpUserXLoginConfig](#)

Konstante:

mpUSERX_WRN_CONFIG_LOAD

4.3.3 -2137886702: Passwortänderung erforderlich

Beschreibung:

Der Benutzer muss das Passwort ändern. Einloggen ist weiterhin möglich, weitere Funktionen sind jedoch gesperrt, bis das Passwort geändert wurde.

Reaktion:

Der Funktionsblock wird am Ausgang "StatusID" anzeigen, dass eine Warnung aktiv ist.

Ursache / Abhilfe:

- Passwort ändern
- Bei der Verwendung des [zentralen Benutzermanagements](#) gilt es zu beachten, dass am Active Directory Server die Einstellung "User must change password at next logon" nicht aktiv ist. Für mehr Information siehe Abschnitt FAQ.

Konstante:

mpUSERX_WRN_PASSWORD_CHANGE_REQ

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)

4.3.4 -2137886670: LDAP Warnung

Beschreibung:

LDAP Warnung: {2:ErrorText} ({1:ErrorNumber})

Reaktion:

Der Funktionsblock wird am Ausgang "StatusID" anzeigen, dass eine Warnung aktiv ist.

Ursache / Abhilfe:

- Kontrollieren des LDAP-Fehlercodes. Mehr Information dazu kann [hier](#) gefunden werden

Zusätzliche Information

- {1:ErrorNumber}: LDAP Warnung
- {2:ErrorText}: Fehlertext

Konstante:

mpUSERX_WRN_LDAP_WARNING

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)
- [MpUserXLoginUI](#)
- [MpUserXSignatureUI](#)

4.3.5 -2137886678 : Key wird bereits verwendet

Beschreibung:

Der Key {2:Key} wird bereits verwendet. Der Wert {3:Value} wird nicht übernommen. {1:ErrorNumber}

Reaktion:

Der Funktionsblock wird am Ausgang "StatusID" anzeigen, dass eine Warnung aktiv ist.

Zusätzliche Information

- {1:ErrorNumber}: Fehlernummer.
- {2:Key}: Key, welcher angegeben wurde.
- {3:Value}: Wert der übergeben wurde.

Ursache / Abhilfe:

- Anderen Key angeben

Konstante:

mpUSERX_WRN_KEY_DUPLICATE

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXGetData](#)

4.3.6 -2137886667: Passwort wird ablaufen**Beschreibung:**

Das angegebene Passwort wird in {1:RemainingDays} Tagen ablaufen.

Reaktion:

Der Funktionsblock wird am Ausgang "StatusID" anzeigen, dass eine Warnung aktiv ist.

Ursache / Abhilfe:

- Ändern des Passworts. Für mehr Information siehe [Application Session](#) bzw. [Visualization Session](#).

Zusätzliche Information

- {1:RemainingDays}: Tage bevor das Passwort ablaufen wird

Konstante:

mpUSERX_WRN_PASSWORD_WILL_EXPIRE

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)
- [MpUserXLoginUI](#)

4.3.7 -1064297640: Fehler beim Lesen der Konfiguration**Beschreibung:**

Fehler beim Lesen der Konfiguration für den Benutzer {2:UserName}. Fehlerursache: {1:ErrorNumber}

Zusätzliche Informationen

- {1:ErrorNumber}: Grund des Fehlers.
- {2:UserName}: Name des Benutzers.

Ursache / Abhilfe:

- Überprüfen der Konfiguration auf Vollständigkeit

Konstante:

mpUSERX_ERR_USER_CONFIG_READ

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)

4.3.8 -1064297639: Fehler beim Speichern der Konfiguration

Beschreibung:

Fehler beim Speichern der Benutzer-Konfiguration für den Benutzer {2:UserName}.

Zusätzliche Informationen

- {1:ErrorNumber}: Grund des Fehlers.
- {2:UserName}: Name des Benutzers.

Ursache / Abhilfe:

- Erstellen Sie mit Hilfe von MpComDump eine Datei, welche alle Informationen über verwendete Komponenten und Dienste beinhaltet und Speichern Sie zusätzlich die Loggerinformationen ab. Kontaktieren Sie mit diesen Informationen den B&R Support.

Konstante:

mpUSERX_ERR_USER_CONFIG_SAVE

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)

4.3.9 -1064297638: Fehler beim Löschen der Konfiguration

Beschreibung:

Beim Löschen der Benutzer-Konfiguration für den Benutzer {2:UserName} ist ein Fehler aufgetreten. Fehlerursache: {1:ErrorNumber}

Zusätzliche Informationen

- {1:ErrorNumber}: Grund des Fehlers.
- {2:UserName}: Name des Benutzers.

Ursache / Abhilfe:

- Erstellen Sie mit Hilfe von MpComDump eine Datei, welche alle Informationen über verwendete Komponenten und Dienste beinhaltet und Speichern Sie zusätzlich die Loggerinformationen ab. Kontaktieren Sie mit diesen Informationen den B&R Support.

Konstante:

mpUSERX_ERR_USER_CONFIG_REMOVE

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)

4.3.10 -1064297637: Fehler beim Lesen der Konfiguration

Beschreibung:

Beim Lesen der Konfiguration für die Rolle {2:RoleName} ist ein Fehler aufgetreten. Fehlerursache {1:ErrorNumber}

Zusätzliche Informationen

- {1:ErrorNumber}: Grund des Fehlers.
- {2:RoleName}: Name der Rolle.

Ursache / Abhilfe:

- Erstellen Sie mit Hilfe von MpComDump eine Datei, welche alle Informationen über verwendete Komponenten und Dienste beinhaltet und Speichern Sie zusätzlich die Loggerinformationen ab. Kontaktieren Sie mit diesen Informationen den B&R Support.

Konstante:

mpUSERX_ERR_ROLE_CONFIG_READ

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)

4.3.11 -1064297636: Fehler beim Speichern der Konfiguration**Beschreibung:**

Beim Speichern der Konfiguration für die Rolle {2:RoleName} ist ein Fehler aufgetreten. Fehlerursache: {1:ErrorNumber}

Zusätzliche Informationen

- {1:ErrorNumber}: Grund des Fehlers.
- {2:RoleName}: Name der Rolle.

Ursache / Abhilfe:

- Erstellen Sie mit Hilfe von MpComDump eine Datei, welche alle Informationen über verwendete Komponenten und Dienste beinhaltet und Speichern Sie zusätzlich die Loggerinformationen ab. Kontaktieren Sie mit diesen Informationen den B&R Support.

Konstante:

mpUSERX_ERR_ROLE_CONFIG_SAVE

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)

4.3.12 -1064297635: Fehler beim Löschen der Konfiguration**Beschreibung:**

Beim Löschen der Konfiguration für die Rolle {2:RoleName} ist ein Fehler aufgetreten. Fehlerursache: {1:ErrorNumber}

Zusätzliche Informationen

- {1:ErrorNumber}: Grund des Fehlers.
- {2:RoleName}: Name der Rolle.

Ursache / Abhilfe:

- Erstellen Sie mit Hilfe von MpComDump eine Datei, welche alle Informationen über verwendete Komponenten und Dienste beinhaltet und Speichern Sie zusätzlich die Loggerinformationen ab. Kontaktieren Sie mit diesen Informationen den B&R Support.

Konstante:

mpUSERX_ERR_ROLE_CONFIG_REMOVE

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)

4.3.13 -1064297634: Fehler beim Wiederherstellen permanenter Daten**Beschreibung:**

Beim Wiederherstellen permanenter Daten aus der Datei {2:FileName} ist ein Fehler aufgetreten. Fehlerursache {1:ErrorNumber}

Zusätzliche Informationen

- {1:ErrorNumber}: Grund des Fehlers.
- {2:FileName}: Name der Datei.

Ursache / Abhilfe:

- Überprüfen, ob Datei beschädigt ist

Konstante:

mpUSERX_ERR_PERMANENT_DATA

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)

4.3.14 -1064297633: Fehler bei der Initialisierung der Benutzer-Konfiguration

Beschreibung:

Bei der Initialisierung der Benutzer-Konfiguration ist ein Fehler aufgetreten. Fehlerursache: {1:ErrorNumber}

Zusätzliche Informationen

- {1:ErrorNumber}: Grund des Fehlers.

Ursache / Abhilfe:

- Erstellen Sie mit Hilfe von MpComDump eine Datei, welche alle Informationen über verwendete Komponenten und Dienste beinhaltet und speichern Sie zusätzlich die Loggerinformationen ab. Kontaktieren Sie mit diesen Informationen den B&R Support.

Konstante:

mpUSERX_ERR_INIT_CONFIG

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)

4.3.15 -1064297632: Fehler beim Importieren der Rolle

Beschreibung:

Beim Importieren der Rolle {2:RoleName} ist ein Fehler aufgetreten. Fehlerursache: {1:ErrorNumber}

Zusätzliche Informationen

- {1:ErrorNumber}: Grund des Fehlers.
- {2:RoleName}: Name der Rolle.

Ursache / Abhilfe:

- Besteht kein Unterschied zu bereits vorhandenen Rollen am Zielsystem und den zu importierenden Rollen, kann Rolle nicht importiert werden

Konstante:

mpUSERX_ERR_IMPORT_ROLE

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)

4.3.16 -1064297631: Fehler beim Importieren des Benutzers**Beschreibung:**

Beim Importieren des Benutzers {2:UserName} ist ein Fehler aufgetreten. Fehlerursache: {1:ErrorNumber}

Zusätzliche Informationen

- {1:ErrorNumber}: Grund des Fehlers.
- {2:UserName}: Name des Benutzers.

Ursache / Abhilfe:

- Besteht kein Unterschied zu bereits vorhandenen Benutzern am Zielsystem und den zu importierenden Benutzer, kann Benutzer nicht importiert werden

Konstante:

mpUSERX_ERR_IMPORT_USER

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)

4.3.17 -1064297630: Fehler beim Löschen der URS-Rolle**Beschreibung:**

Beim Löschen der URS-Rolle {2:RoleName} ist ein Fehler aufgetreten. Fehlerursache: {1:ErrorNumber}

Zusätzliche Informationen

- {1:ErrorNumber}: Grund des Fehlers.
- {2:RoleName}: Name der Rolle.

Ursache / Abhilfe:

- Erstellen Sie mit Hilfe von MpComDump eine Datei, welche alle Informationen über verwendete Komponenten und Dienste beinhaltet und Speichern Sie zusätzlich die Loggerinformationen ab. Kontaktieren Sie mit diesen Informationen den B&R Support.

Konstante:

mpUSERX_ERR_DELETE_URS_ROLE

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)

4.3.18 -1064297629: Fehler beim Löschen des URS-Benutzers**Beschreibung:**

Beim Löschen des URS-Benutzers {2:UserName} ist ein Fehler aufgetreten. Fehlerursache: {1:ErrorNumber}

Zusätzliche Informationen

- {1:ErrorNumber}: Grund des Fehlers.
- {2:UserName}: Name des Benutzers.

Ursache / Abhilfe:

- Erstellen Sie mit Hilfe von MpComDump eine Datei, welche alle Informationen über verwendete Komponenten und Dienste beinhaltet und Speichern Sie zusätzlich die Loggerinformationen ab. Kontaktieren Sie mit diesen Informationen den B&R Support.

Konstante:

mpUSERX_ERR_DELETE_URS_USER

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)

4.3.19 -1064297628: Fehler beim Erstellen eines URS-Benutzers

Beschreibung:

Beim Erstellen des URS-Benutzers {2:UserName} ist ein Fehler aufgetreten. Fehlerursache: {1:ErrorNumber}

Zusätzliche Informationen

- {1:ErrorNumber}: Grund des Fehlers.
- {2:UserName}: Name des Benutzers.

Ursache / Abhilfe:

- Erstellen Sie mit Hilfe von MpComDump eine Datei, welche alle Informationen über verwendete Komponenten und Dienste beinhaltet und Speichern Sie zusätzlich die Loggerinformationen ab. Kontaktieren Sie mit diesen Informationen den B&R Support.

Konstante:

mpUSERX_ERR_CREATE_URS_USER

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)

4.3.20 -1064297627: Fehler beim Ändern des Passworts

Beschreibung:

Beim Ändern des Passworts für den URS-Benutzer {2:UserName} ist ein Fehler aufgetreten. Fehlerursache: {1:ErrorNumber}

Zusätzliche Informationen

- {1:ErrorNumber}: Grund des Fehlers.
- {2:UserName}: Name des Benutzers.

Ursache / Abhilfe:

- Kontrolle ob Passwort Kriterien erfüllt wurden

Konstante:

mpUSERX_ERR_CHANGE_URS_PASSWORD

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)

4.3.21 -1064297626: Fehler beim Erstellen von Rollen

Beschreibung:

Beim Erstellen von Rollen für den Benutzer {2:UserName} ist ein Fehler aufgetreten. Fehlerursache: {1:ErrorNumber}

Zusätzliche Informationen

- {1:ErrorNumber}: Grund des Fehlers.
- {2:UserName}: Name des Benutzers.

Ursache / Abhilfe:

- Erstellen Sie mit Hilfe von MpComDump eine Datei, welche alle Informationen über verwendete Komponenten und Dienste beinhaltet und Speichern Sie zusätzlich die Loggerinformationen ab. Kontaktieren Sie mit diesen Informationen den B&R Support.

Konstante:

mpUSERX_ERR_CREATE_URS_ROLE

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)

4.3.22 -1064297623: Manager konnte nicht installiert werden.**Beschreibung:**

Das User-Management konnte nicht registriert werden. Der Fehler kann auftreten wenn die Komponente MpUser auf dem Zielsystem vorhanden ist. Fehlerursache:{1:ErrorNumber}

Ursache / Abhilfe:

- Auf dem Zielsystem darf MpUser nicht vorhanden sein

Zusätzliche Information

- {1:ErrorNumber}: Fehlerursache

Konstante:

mpUSERX_INT_ERR_INSTALL_MANAGER

4.3.23 -1064239103: Komponente konnte nicht erstellt werden**Beschreibung:**

Die mapp Komponente konnte nicht erstellt werden und wird nicht aktiviert. Weitere Details siehe Logger.

Reaktion:

Der Funktionsbaustein wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Service zur Konfiguration des Funktionsbausteins nicht verfügbar - Problem mit MpCom
- Registry nicht lesbar - Problem mit MpCom
- Details zur Fehlerursache im Logger

Diese Funktionsbausteine / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)
- [MpUserXLoginUI](#)
- [MpUserXManagerUI](#)
- [MpUserXLoginConfig](#)
- [MpUserXSignature](#)
- [MpUserXSignatureUI](#)

Konstante:

mpUSERX_ERR_ACTIVATION

4.3.24 -1064239102: MpLink ist ein Nullzeiger**Beschreibung:**

Der "MpLink" Eingang ist nicht verbunden - NULL Zeiger.

Reaktion:

Der Funktionsbaustein wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- "MpLink" Eingang am Funktionsbaustein überprüfen

Diese Funktionsbausteine / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)
- [MpUserXLoginUI](#)
- [MpUserXManagerUI](#)
- [MpUserXLoginConfig](#)
- [MpUserXSignature](#)
- [MpUserXSignatureUI](#)

Konstante:

`mpUSERX_ERR_MPLINK_NULL`

4.3.25 -1064239101: MpLink Verbindung nicht erlaubt**Beschreibung:**

Der verwendete Wert am "MpLink" Eingang ist nicht erlaubt.

Reaktion:

Der Funktionsbaustein wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Komponente ist direkt mit `mpCOM_MAIN` oder `mpCOM_STANDALONE` verbunden, das wird nicht unterstützt

Diese Funktionsbausteine / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)
- [MpUserXLoginUI](#)
- [MpUserXManagerUI](#)
- [MpUserXLoginConfig](#)
- [MpUserXSignature](#)
- [MpUserXSignatureUI](#)

Konstante:

`mpUSERX_ERR_MPLINK_INVALID`

4.3.26 -1064239100: MpLink wurde verändert**Beschreibung:**

Der Wert am "MpLink" Eingang hat sich zur Laufzeit der Komponenten ("Enable" = `TRUE`) verändert.

Reaktion:

Der Funktionsbaustein wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Den Wert am "MpLink" Eingang nur verändern während die Komponente inaktiv ist ("Enable" = FALSE)

Diese Funktionsbausteine / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)
- [MpUserXLoginUI](#)
- [MpUserXManagerUI](#)
- [MpUserXLoginConfig](#)
- [MpUserXSignature](#)
- [MpUserXSignatureUI](#)

Konstante:

mpUSERX_ERR_MPLINK_CHANGED

4.3.27 -1064239099: MpLink Inhalt ist ungültig**Beschreibung:**

Der Wert in der Variable "MpLink" am Eingang des Funktionsbaustein ist ungültig.

Reaktion:

Der Funktionsbaustein wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Die Werte in der Struktur MpComIdentType dürfen nicht beschrieben werden
- Es muss eine entsprechende Konfiguration für diese mapp Komponenten vorhanden sein (siehe erster Absatz der Beschreibung von den unten angeführten Funktionen / Funktionsbausteinen)

Diese Funktionsbausteine / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)
- [MpUserXLoginUI](#)
- [MpUserXManagerUI](#)
- [MpUserXLoginConfig](#)
- [MpUserXSignature](#)
- [MpUserXSignatureUI](#)

Konstante:

mpUSERX_ERR_MPLINK_CORRUPT

4.3.28 -1064239098: MpLink wird bereits benützt**Beschreibung:**

Dieser MpLink wird bereits benützt.

Reaktion:

Der Funktionsbaustein wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Ist der ausgewählte Download Modus "Overload", sollte im Exit Programm die Mapp Komponente mit "Enable = FALSE" deaktiviert werden. Als Abhilfe kann auch der Modus "Copy" oder "One cycle" verwendet werden. Hierbei ist es nicht nötig die mapp Komponente zu deaktivieren.
- Überprüfen welche Komponente diesen MpLink bereits benützt
- Neuen MpLink erstellen

Diese Funktionsbausteine / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)
- [MpUserXLoginUI](#)
- [MpUserXManagerUI](#)
- [MpUserXLoginConfig](#)

Konstante:

mpUSER_ERR_MPLINK_IN_USE

4.3.29 -1064239096: Konfigurations Struktur ist ein Nullzeiger**Beschreibung:**

Der "Configuration" Eingang ist nicht verbunden - NULL Zeiger.

Reaktion:

Der Funktionsbaustein wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- "Configuration" Eingang am Funktionsbaustein überprüfen

Diese Funktionsbausteine / Funktionen können diesen Fehler melden:

- [MpUserXConfig](#)
- [MpUserXLoginConfig](#)

Konstante:

mpUSERX_ERR_CONFIG_NULL

4.3.30 -1064239095: Konfigurations Zeiger ist keine PV**Beschreibung:**

Der angegebene Zeiger ist kein Zeiger auf eine gültige PV. Weitere Details siehe Logger.

Reaktion:

Der Funktionsbaustein wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Den Wert am "Configuration" Eingang prüfen

Diese Funktionsbausteine / Funktionen können diesen Fehler melden:

- [MpUserXConfig](#)
- [MpUserXLoginConfig](#)

Konstante:

mpUSERX_ERR_CONFIG_NO_PV

4.3.31 -1064239094: Fehler beim Laden der Konfiguration**Beschreibung:**

Fehler beim Laden der Konfigurationsdatei (in die PV). Weitere Details siehe Logger.

Reaktion:

Der Funktionsbaustein wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Konfigurationsdatei beschädigt
- Unterschied zwischen dem Inhalt der Konfigurationsdatei und der PV (z.B. unterschiedliche Datentypen, Daten außerhalb der gültigen Limits, ...)

Diese Funktionsbausteine / Funktionen können diesen Fehler melden:

- [MpUserXConfig](#)
- [MpUserXLoginConfig](#)

Konstante:

mpUSERX_ERR_CONFIG_LOAD

4.3.32 -1064239092: Fehler beim Speichern der Konfiguration**Beschreibung:**

Fehler beim Speichern in die Konfigurationsdatei (aus der PV). Weitere Details siehe Logger.

Reaktion:

Der Funktionsbaustein wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Konfigurationsdatei beschädigt
- Unterschied zwischen dem Inhalt der Konfigurationsdatei und der PV (z.B. unterschiedliche Datentypen, Daten außerhalb der gültigen Limits, ...)

Diese Funktionsbausteine / Funktionen können diesen Fehler melden:

- [MpUserXConfig](#)
- [MpUserXLoginConfig](#)

Konstante:

mpUSERX_ERR_CONFIG_SAVE

4.3.33 -1064239091: Ungültige Konfiguration**Beschreibung:**

Konfiguration konnte während dem Erstellen der Komponente nicht gelesen werden. Weitere Details siehe Logger.

Reaktion:

Der Funktionsbaustein wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Die Konfiguration ist beschädigt
- Die Funktion ist in der Konfiguration nicht aktiviert

Diese Funktionsbausteine / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)
- [MpUserXMappingConfig](#)

Konstante:

mpUSERX_ERR_CONFIG_INVALID

4.3.34 -1064144874: Fehlender Wert an UIConnect

Beschreibung:

An "UIConnect" wurde NULL angehängt.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Eingang "UIConnect" vergessen
- Am Eingang wurde 0 angehängt

Konstante:

mpUSERX_ERR_MISSING_UICONNECT

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLoginUI](#)
- [MpUserXManagerUI](#)
- [MpUserXSignatureUI](#)

4.3.35 -1064144896: Passwort ungültig

Beschreibung:

Das eingegebene Passwort ist ungültig.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Passwort überprüfen

Konstante:

mpUSERX_ERR_PASSWORD_INCORRECT

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)
- [MpUserXLoginUI](#)
- [MpUserXSignatureUI](#)

4.3.36 -1064144895: Benutzer existiert nicht

Beschreibung:

Der eingegebene Benutzer existiert nicht

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Benutzernamen überprüfen

Konstante:

mpUSERX_ERR_USER_NOT_EXISTING

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)
- [MpUserXLoginUI](#)
- [MpUserXManagerUI](#)
- [MpUserXSignatureUI](#)

4.3.37 -1064144894: Benutzer existiert bereits**Beschreibung:**

Der eingegebene Benutzername existiert bereits

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Benutzername überprüfen und / oder ändern

Konstante:

mpUSERX_ERR_USER_EXISTS

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)
- [MpUserXLoginUI](#)
- [MpUserXManagerUI](#)

4.3.38 -1064144893: Login Token ungültig**Beschreibung:**

Der verwendete Login Token ist ungültig für den angegebenen Benutzer.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Login Token prüfen
- Benutzernamen prüfen

Konstante:

mpUSERX_ERR_LOGIN_ID_INVALID

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)
- [MpUserXLoginUI](#)
- [MpUserXManagerUI](#)

4.3.39 -1064144892: Keine Administratorrechte

Beschreibung:

Benutzer hat keine Administratorrechte.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Benutzer ist kein Administrator
- Unzureichende Benutzerrechte

Konstante:

mpUSERX_ERR_ADMIN_INVALID

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXManagerUI](#)

4.3.40 -1064144891: Ungenügende Benutzerrechte

Beschreibung:

Benutzerrechte sind für die gewünschte Operation unzureichend.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Benutzer hat keine Freigabe für die gewünschte Operation
- Benutzer mit zu niedrigem Benutzerlevel hat versucht eine Signatur durchzuführen
- Bei der Verwendung des [Zentrales User management](#) gilt es zu beachten, dass am Active Directory Server die Einstellung "Password never expires" deaktiviert ist. Ansonsten kann sich der Benutzer nicht anmelden. Für mehr Information siehe Abschnitt FAQ.

Konstante:

mpUSERX_ERR_INSUFFICIENT_RIGHTS

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXManagerUI](#)
- [MpUserXSignatureUI](#)

4.3.41 -1064144890: Rolle existiert nicht

Beschreibung:

Die angegebene Rolle existiert nicht.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Rolle überprüfen

Konstante:

mpUSERX_ERR_ROLE_NOT_PRESENT

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)
- [MpUserXLoginUI](#)
- [MpUserXManagerUI](#)

4.3.42 -1064144889: Passwort nicht sicher genug**Beschreibung:**

Gewähltes Passwort ist nicht sicher genug.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Passwort ändern
- Benötigter Sicherheitsstandard für das Passwort kann über den Funktionsblock [MpUserXConfig](#) festgelegt werden.

Konstante:

mpUSERX_ERR_PASSWORD_WEAK

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLoginUI](#)
- [MpUserXManagerUI](#)

4.3.43 -1064144888: Benutzer gesperrt**Beschreibung:**

Dieser Benutzer ist gesperrt.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Admin muss diesen Benutzer wieder freischalten

Konstante:

mpUSERX_ERR_USER_IS_LOCKED

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)
- [MpUserXLoginUI](#)
- [MpUserXManagerUI](#)

4.3.44 -1064144887: Passwortänderung erforderlich

Beschreibung:

Der Benutzer muss das Passwort ändern, davor sind weitere Funktionen nicht möglich.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Passwort ändern

Konstante:

mpUSERX_ERR_PASSWORD_CHANGE_REQ

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLoginUI](#)
- [MpUserXManagerUI](#)
- [MpUserXSignatureUI](#)

4.3.45 -1064144886: Zuvielen Login Versuche, Login nicht mehr möglich

Beschreibung:

Der Benutzer hat zu oft das falsche Passwort verwendet und wurde gesperrt.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Admin muss Benutzer wieder freischalten

Konstante:

mpUSERX_ERR_TOO_MANY_LOGINS

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)
- [MpUserXLoginUI](#)
- [MpUserXManagerUI](#)

4.3.46 -1064144885: Benutzer hat keine Adminrechte

Beschreibung:

Der eingeloggte Benutzer hat keine Adminrechte.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Admin kontaktieren

Konstante:

mpUSERX_ERR_NO_ADMIN_RIGHTS

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXManagerUI](#)

4.3.47 -1064144884: Benutzermanagement existiert bereits**Beschreibung:**

In dieser Gruppe existiert bereits ein Benutzermanagement. Eine weitere Instanz dieses Funktionsblocks ist nicht möglich.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- In eine andere Gruppe wechseln (Hierarchie Konzept)
- Die zusätzliche [MpUserXLogin](#) Instanz entfernen

Konstante:

mpUSERX_ERR_LOGIN_INST_EXISTS

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)

4.3.48 -1064144883: Kein Benutzer eingeloggt**Beschreibung:**

Es ist kein Benutzer eingeloggt.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Mithilfe von Benutzernamen und Passwort einloggen

Konstante:

mpUSERX_ERR_NOT_LOGGED_IN

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLoginUI](#)
- [MpUserXManagerUI](#)

4.3.49 -1064144882: Funktionsindex übersteigt Maximum**Beschreibung:**

Der Index für die gewählte Funktion übersteigt den maximalen Index, die Rechte können nicht abgefragt werden (reserviert für spätere Verwendung).

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Den Eingang "RightIndex" überprüfen
- Index für gewählte Funktion zu hoch

Konstante:

mpUSERX_ERR_MAX_USERRIGHTS

4.3.50 -1064144881: Passwörter identisch**Beschreibung:**

Das neue und das alte Passwort sind identisch.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Passwort ändern

Konstante:

mpUSERX_ERR_PASSWORD_IDENTICAL

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLoginUI](#)
- [MpUserXManagerUI](#)

4.3.51 -1064144880: Benutzername existiert bereits**Beschreibung:**

Der gewählte Benutzername für diesen Token existiert bereits

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Anderen Benutzernamen wählen

Konstante:

mpUSERX_ERR_LOGIN_ID_EXISTS

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserRFID](#)

4.3.52 -1064144879: Benutzer konnte nicht gelöscht werden**Beschreibung:**

Der gewählte Benutzer konnte nicht gelöscht werden. Weitere Details siehe Logger.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Weitere Details siehe Logger

Konstante:

mpUSERX_ERR_DELETING_USER

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXManagerUI](#)

4.3.53 -1064144877: Ungültiger Rollen-Index**Beschreibung:**

Es wurde ein ungültiger Rollen-Index übergeben.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Überprüfen des Rollen-Index

Konstante:

ROLE_INDEX_NOT_OK

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXManagerUI](#)

4.3.54 -1064144876: Ungültiger Rollenname**Beschreibung:**

Es wurde ein ungültiger Rollenname (NULL) übergeben.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Rollenname ist NULL

Konstante:

mpUSERX_ERR_NO_ROLENAME

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXManagerUI](#)

4.3.55 -1064144875: Benutzer bereits eingeloggt**Beschreibung:**

Der angegebene Benutzer ist bereits eingeloggt.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Benutzernamen überprüfen

Konstante:

mpUSERX_ERR_LOGGED_IN

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)
- [MpUserXLoginUI](#)
- [MpUserXManagerUI](#)

4.3.56 -1064144873: Ungültiger Benutzername**Beschreibung:**

Es wurde ein ungültiger Benutzername (NULL) übergeben.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Benutzername ist NULL

Konstante:

mpUSERX_ERR_NO_USERNAME

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXManagerUI](#)
- [MpUserXSignatureUI](#)

4.3.57 -1064144872: Passwort fehlt**Beschreibung:**

Es wurde kein Passwort eingegeben.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Passwort eingeben

Konstante:

mpUSERX_ERR_NO_PASSWORD

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXManagerUI](#)
- [MpUserXSignatureUI](#)

4.3.58 -1064144871: MpUserXLogin fehlt**Beschreibung:**

Der Funktionsblock [MpUserXLogin](#) fehlt oder ist falsch konfiguriert.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- [MpUserXLogin](#) überprüfen

Konstante:`mpUSERX_ERR_MISSING_LOGIN`**Diese Funktionsblöcke / Funktionen können diesen Fehler melden:**

- [MpUserXLoginUI](#)
- [MpUserXManagerUI](#)

4.3.59 -1064144870: Passwort zum Bestätigen falsch**Beschreibung:**

Das zur Bestätigung eingegebene Passwort stimmt nicht mit dem neuen Passwort überein.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Passwörter überprüfen

Konstante:`mpUSERX_ERR_PASSWORD_CONFIRM`**Diese Funktionsblöcke / Funktionen können diesen Fehler melden:**

- [MpUserXLoginUI](#)
- [MpUserXManagerUI](#)

4.3.60 -1064144869: Angegebenes Systemrecht existiert nicht**Beschreibung:**

Das angegebene Systemrecht wurde nicht gefunden (reserviert für spätere Verwendung).

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Gewähltes Systemrecht überprüfen

Konstante:`mpUSERX_ERR_NO_SUCH_SYSRIGHT`**Diese Funktionsblöcke / Funktionen können diesen Fehler melden:**

- [MpUserXManagerUI](#)

4.3.61 -1064144868: Import fehlgeschlagen**Beschreibung:**

Der Import von Benutzergruppen und/oder Benutzerdaten ist fehlgeschlagen. Weitere Details siehe Logger.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Importdatei überprüfen
- Den Inhalt der angegebenen Datei auf Richtigkeit und Vollständigkeit prüfen

Konstante:

mpUSERX_ERR_IMPORT_DATA

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXManagerUI](#)

4.3.62 -1064144867: Fehler beim Laden der Importdatei

Beschreibung:

Beim Laden der Importdatei ist ein Fehler aufgetreten. Weitere Details siehe Logger.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Importdatei überprüfen
- Datei existiert nicht
- Speichermedium existiert nicht

Konstante:

mpUSERX_ERR_LOAD_IMPORT_FILE

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXManagerUI](#)

4.3.63 -1064144866: Fehler beim Speichern der Exportdatei

Beschreibung:

Beim Speichern der Exportdatei ist ein Fehler aufgetreten. Weitere Details siehe Logger.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Exportdatei überprüfen
- Datei existiert nicht
- Speichermedium existiert nicht

Konstante:

mpUSERX_ERR_SAVE_EXPORT_FILE

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXManagerUI](#)

4.3.64 -1064144865: Name der Rolle ist zu lange

Beschreibung:

Der Name der eingegebenen Rolle ist zu lange (> 50 Zeichen).

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Name der Rolle kürzen. Maximum sind 50 Zeichen

Konstante:

mpUSERX_ERR_ROLENAME_TOO_LONG

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXManagerUI](#)

4.3.65 -1064144864: Name des Benutzers ist zu lange**Beschreibung:**

Der Name des eingegebenen Benutzers ist zu lange (> 20 Zeichen).

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Name des Benutzers kürzen. Maximum sind 20 Zeichen

Konstante:

mpUSERX_ERR_USERNAME_TOO_LONG

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXManagerUI](#)

4.3.66 -1064144863: Name des Passwortes ist zu lange**Beschreibung:**

Der Name des eingegebenen Passwortes ist zu lange (> 20 Zeichen).

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Name des Passwortes kürzen. Maximum sind 20 Zeichen

Konstante:

mpUSERX_ERR_PASSWORD_TOO_LONG

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXManagerUI](#)

4.3.67 -1064144861: Name des Benutzers ist zu kurz**Beschreibung:**

Der eingegebene Benutzername ist zu kurz.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Längeren Benutzernamen auswählen

Konstante:

mpUSERX_ERR_USERNAME_TOO_SHORT

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXManagerUI](#)

4.3.68 -1064144860: Rolle existiert bereits**Beschreibung:**

Die eingegebene Rolle existiert bereits

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Rolle überprüfen und / oder ändern

Konstante:

mpUSERX_ERR_ROLE_EXISTS

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXManagerUI](#)

4.3.69 -1064144858: Signieren fehlgeschlagen**Beschreibung:**

Maximale Anzahl an Signatur-Versuchen überschritten. Der Signatur Vorgang wurde unterbrochen

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Angabe der richtigen Aktions-ID

Konstante:

mpUSERX_ERR_FAILED_SIGNATURES

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXSignatureUI](#)

4.3.70 -1064144857: Letzte Signatur wurde noch nicht geprüft**Beschreibung:**

Es wurde eine neue Signatur gestartet, bevor die letzte signierte Aktion ausgeführt wurde.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Letzte Signatur muss mithilfe von [MpUserXSignature](#) erfolgreich ausgeführt werden (Released = TRUE)

Konstante:

mpUSERX_ERR_SIGNATURE_BUSY

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXSignatureUI](#)

4.3.71 -1064144856: Passwort wurde bereits verwendet**Beschreibung:**

Das eingegebene Passwort wurde bereits verwendet und kann nicht nocheinmal angegeben werden

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Angabe eines anderen Passworts

Konstante:

mpUSERX_ERR_PASSWORD_REPEAT

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXManagerUI](#)
- [MpUserXLoginUI](#)

4.3.72 -1064144855: Ungültige Import Datei**Beschreibung:**

Die Datei welche Importiert werden soll, besitzt ein ungültiges Format.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Überprüfen der zu importierenden Datei

Konstante:

mpUSERX_ERR_IMPORT_FILE_FORMAT

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXManagerUI](#)

4.3.73 -1064144853: Information nicht gefunden**Beschreibung:**

Die gewünschte Information wurde nicht gefunden

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Überprüfen des angegebenen Benutzers
- Überprüfen der gewünschten Information
- Überprüfen des angegebenen Keys

Konstante:

mpUSERX_ERR_NO_ADD_DATA

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXGetData](#)

4.3.74 -1064144852 : Key ist leer

Beschreibung:

Der angegebene Key ist leer.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Angeben eines gültigen Keys

Konstante:

mpUSERX_ERR_EMPTY_KEY

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXGetData](#)

4.3.75 -1064144851: Zu kleiner Buffer

Beschreibung:

Es wurde ein zu kleiner Buffer festgelegt.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Festlegen eines größeren Buffers.

Konstante:

mpUSERX_ERR_BUFFER_TOO_SMALL

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXGetData](#)

4.3.76 -1064144850: LDAP Fehler

Beschreibung:

LDAP Fehler: {2:ErrorText} ({1:ErrorNumber}). Fehler in der Kommunikation zwischen mapp UserX und dem Directory Server.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Zusätzliche Information

- {1:ErrorNumber}: LDAP Fehlercode
- {2:ErrorText}: Fehlertext

Ursache / Abhilfe:

- Kontrollieren des LDAP-Fehlercodes. Mehr Information dazu kann [hier](#) gefunden werden
- Kontrollieren des verwendeten Passworts
- Kontrollieren ob benötigte Zertifikate vorhanden sind
- Kontrollieren ob angegebene Server-Einstellungen in der [MpUserX Konfiguration](#) korrekt sind

Konstante:

mpUSERX_ERR_LDAP_ERROR

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)
- [MpUserXLoginUI](#)
- [MpUserXSignature](#)
- [MpUserXSignatureUI](#)

4.3.77 -1064144849: Keine Server-Verbindung

Beschreibung:

Es konnte keine Verbindung mit dem Server hergestellt werden.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Überprüfen der Netzwerk-Verbindung
- Überprüfen ob Directory Server aktiv ist
- Überprüfen ob angegebene Server-Einstellungen in der [MpUserX Konfiguration](#) korrekt sind
- Überprüfen ob das [Zertifikat](#) gültig und auch nicht veraltet ist

Konstante:

mpUSERX_ERR_NO_SERVER_CONNECTION

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)
- [MpUserXLoginUI](#)
- [MpUserXSignature](#)
- [MpUserXSignatureUI](#)

4.3.78 -1064144848: Ungültige Eingabe**Beschreibung:**

Ungültige Eingabe.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Überprüfen des Benutzernamens und des Passworts

Konstante:

mpUSERX_ERR_INVALID_CREDENTIALS

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)
- [MpUserXLoginUI](#)
- [MpUserXSignature](#)
- [MpUserXSignatureUI](#)

4.3.79 -1064144847: Aktion nicht erlaubt**Beschreibung:**

Die aktuell ausgeführte Aktion ({2:Action}) ist bei der Verwendung des zentralen Benutzermanagements nicht erlaubt.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Zusätzliche Information

- {2:Action}: Aktuell ausgeführte Aktion

Ursache / Abhilfe:

Folgende Aktionen dürfen bei der Verwendung des zentralen Benutzermanagements nicht ausgeführt werden:

- Verändern der Benutzerinformation
- Bearbeiten eines Benutzers welcher nur über den Active Directory Server verfügbar ist
- Importieren/Exportieren von Benutzern und Rollen
- Erstellen von lokalen Benutzern
- Wird das zentrale bzw. lokal und zentrale Benutzermanagement verwendet und ein zentraler Benutzer meldet sich an, so kann der Benutzer das Passwort nicht ändern wenn dieser zuvor das Benutzermanagement auf lokal umstellt
- Wird das lokale bzw. lokal und zentrale Benutzermanagement verwendet und ein lokaler Benutzer meldet sich an, so kann der Benutzer das Passwort nicht ändern wenn dieser zuvor das Benutzermanagement auf zentral umstellt.

Konstante:

mpUSERX_ERR_ACTION_NOT_ALLOWED

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)
- [MpUserXLoginUI](#)
- [MpUserXManagerUI](#)

4.3.80 -1064144845: Passwort kann nicht geändert werden**Beschreibung:**

Das Passwort kann nicht geändert werden. Das neue Passwort stimmt nicht mit den Kennwortrichtlinien des Servers überein.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Bei Verwendung des [Zentrales User management](#) müssen die Kennwortrichtlinien des Directory Servers überprüft werden. Stimmt das neue Passwort mit den Server-Richtlinien nicht überein, so kann das Passwort nicht geändert werden.
- Verwenden eines anderen Passworts

Konstante:

mpUSERX_ERR_PASSWORD_RESTRICTION

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLoginUI](#)

4.3.81 -1064144844: Funktion nicht verfügbar**Beschreibung:**

Das [Zentrales User management](#) kann erst ab einer Automation Runtime Version \geq F4.44 verwendet werden.

Reaktion:

Der Funktionsblock wird an den Ausgängen "StatusID" und "Error" anzeigen, dass ein Fehler aktiv ist, weiters sind alle Funktionen solange nicht verfügbar.

Ursache / Abhilfe:

- Automation Runtime \geq F4.44 verwenden
- Verwenden des lokalen Benutzermanagements. Für mehr Information siehe [hier](#).

Konstante:

mpUSERX_ERR_FUNCTION_UNAVAILABLE

Diese Funktionsblöcke / Funktionen können diesen Fehler melden:

- [MpUserXLogin](#)
- [MpUserXSignature](#)

4.4 Alarme**4.4.1 mpUSERX_ALM_USER_LOCKED: Benutzer wurde aufgrund zu vieler fehlerhafter Loginversuche gesperrt****Beschreibung:**

Der Benutzer wurde nach zu vielen fehlerhaften Loginversuchen gesperrt.

Reaktion:

Wenn eine MpAlarmX Komponente aktiv ist, wird ein mapp Alarm ausgelöst.

Ursache / Abhilfe:

- Zum Wiederfreischalten des Benutzers muss der Administrator kontaktiert werden

Verhalten

Edge Alarm