



---

**1 Contact**

---

[www.siebert-group.com](http://www.siebert-group.com)

**ALLEMAGNE**

Siebert Industrieelektronik GmbH  
Siebertstrasse, D-66571 Eppelborn  
Postfach 11 30, D-66565 Eppelborn  
Tél. +49 (0)6806 980-0, Fax +49 (0)6806 980-999  
E-Mail: [info.de@siebert-group.com](mailto:info.de@siebert-group.com)

**AUTRICHE**

Siebert Österreich GmbH  
Simmeringer Landstrasse 24., A-1110 Wien  
Tél. +43 (0)1 890 63 86-0, Fax +43 (0)14 890 63 86-99  
E-Mail: [info.at@siebert-group.com](mailto:info.at@siebert-group.com)

**FRANCE**

Siebert France Sarl  
4 rue de l'Abbé Louis Verdet, F-57200 Sarreguemines  
BP 90 334, F-57203 Sarreguemines Cédex  
Tél. +33 (0)3 87 98 63 68, Fax +33 (0)3 87 98 63 94  
E-Mail: [info.fr@siebert-group.com](mailto:info.fr@siebert-group.com)

**ITALIE**

Siebert Italia Srl  
Via Galileo Galilei 2A, I-39100 Bolzano (BZ)  
Tél. +39 (0)471 053753, Fax +39 (0)471 053754  
E-Mail: [info.it@siebert-group.com](mailto:info.it@siebert-group.com)

**PAYS-BAS**

Siebert Nederland B.V.  
Jadedreef 26, NL-7828 BH Emmen  
Tél. +31 (0)591-633444, Fax +31 (0)591-633125  
E-Mail: [info.nl@siebert-group.com](mailto:info.nl@siebert-group.com)

**SUISSE**

Siebert AG  
Bützbergstrasse 2, CH-4912 Aarwangen  
Tél. +41 (0)62 922 18 70, Fax +41 (0)62 922 33 37  
E-Mail: [info.ch@siebert-group.com](mailto:info.ch@siebert-group.com)

---

## **2 Mentions légales**

---

© Siebert Industrieelektronik GmbH

Ce manuel d'utilisation a été élaboré avec le plus grand soin. Cependant, nous déclinons toute responsabilité en cas d'erreur affectant les informations. N'hésitez pas à nous faire part de vos corrections, suggestions d'amélioration, critiques et idées. Ecrivez-nous à l'adresse suivante: [redaktion@siebert-group.com](mailto:redaktion@siebert-group.com)

Siebert<sup>®</sup>, LRD<sup>®</sup> et XC-Board<sup>®</sup> sont des marques déposées de la société Siebert Industrieelektronik GmbH. Tous les autres noms de produits ou de sociétés mentionnés dans cette documentation peuvent être des marques ou des désignations commerciales de leurs propriétaires respectifs.

Sous réserve de modifications techniques et des possibilités de livraison. – Tous droits réservés, y compris ceux de la traduction. Il est interdit de reproduire, de traiter, de polycopier ou de diffuser ce document intégralement ou partiellement à l'aide de systèmes électroniques, sous quelle forme que ce soit (impression, photocopie, microfilm ou autre procédé) sans notre autorisation écrite préalable.

---

## Sommaire

---

<b>1 Contact</b>	<b>2</b>
<b>2 Mentions légales</b>	<b>3</b>
<b>3 Introduction</b>	<b>6</b>
Champ d'application et connaissances nécessaires.....	6
SPS soutenu.....	6
Outils d'ingénierie soutenus .....	6
Protocole de transmission .....	6
Intervalle d'interrogation .....	6
<b>4 Cas d'application typiques</b>	<b>7</b>
Utiliser des noms de variables prédéfinis dans l'API .....	7
Utiliser ses propres noms de variables dans l'API .....	7
Plusieurs afficheurs avec une mise en page identique .....	7
Plusieurs afficheurs avec une mise en page différente et les mêmes noms de variables.....	7
Plusieurs afficheurs avec une mise en page et des variables différentes .....	7
Commande supplémentaire via le site web .....	7
<b>5 Démarrage rapide et déroulement général</b>	<b>8</b>
Démarrage rapide .....	8
Procédure générale.....	8
<b>6 Démarrage rapide Partie 1 : Configuration de l'afficheur et création des fichiers pour le contrôleur</b>	<b>9</b>
Téléchargement des fichiers PLC-Exemple sur le site de Siebert.....	9
Téléchargement de l'exemple de mise en page sur l'afficheur .....	9
Sélection de la mise en page sur l'afficheur.....	9
Saisie des données de connexion sur la page web de l'afficheur .....	10
Sélection des éléments pour l'exportation de la configuration de l'API.....	10
Contenu du fichier XML.....	11
Contenu du fichier de modules de données.....	12
<b>7 Quick Start partie 2a : Configuration de l'automate et du serveur web dans le TIA Portal</b>	<b>13</b>
Mise en marche du serveur web et création d'un utilisateur .....	13
Définir les droits des utilisateurs.....	14
Télécharger la base de données et le fichier XML dans l'automate .....	14
Générer un module de serveur web.....	15
Appel cyclique de la page web.....	15
Fournir des données pour l'afficheur.....	16
Contrôle de fonctionnement .....	16

<b>8 Démarrage rapide Partie 2b : Configuration de l'automate et du serveur web avec STEP 7</b>	<b>17</b>
Vérifier le serveur web de l'unité centrale .....	17
Utiliser les fichiers exportés par l'afficheur .....	17
Créer un nouveau projet STEP 7 .....	18
Créer un bloc de données .....	19
Créer le SFC99.....	19
Démarrer Web2PLC .....	20
Créer un nouveau projet Web2PLC .....	20
Exportation des données des blocs de données du projet S7 .....	21
Ajuster les paramètres de projection .....	22
Traduction de la source des blocs de données et chargement du projet web sur le CPU .....	23
Télécharger le projet complet dans l'unité centrale.....	23
Contrôle de la page web de l'afficheur .....	24
<b>9 Démarrage rapide Partie 3 : Conseils de dépannage</b>	<b>25</b>
Remarques générales .....	25
Messages de code HTTP .....	25
Messages du protocole .....	26
Contrôle du fichier XML.....	26
Caractères spéciaux dans le fichier XML .....	26
Date et heure.....	27
Caractères .....	27
<b>10 Exploiter plusieurs afficheurs sur un API</b>	<b>28</b>
Toutes les afficheurs ont la même mise en page et doivent afficher les mêmes données.....	28
Les afficheurs doivent indiquer différentes données.....	28
<b>11 Données techniques</b>	<b>28</b>

---

### **3 Introduction**

---

Avec l'extension Connecteur PLC, les displays de la série XC50/XC55 vous offrent un moyen simple de récupérer et d'afficher automatiquement les données d'application d'un PLC Siemens. Pour ce faire, l'API doit disposer d'un serveur web qui fournit les données souhaitées sous la forme d'un fichier XML. Ce fichier XML est récupéré de manière cyclique par le display XC50/XC55 et son contenu est affiché sur le display.

#### **Champ d'application et connaissances nécessaires**

Champ d'application et connaissances nécessaires

Ce manuel s'applique aux afficheurs Siebert de la série XC50/XC55 à partir de la version 3.1 du logiciel. Pour les versions plus récentes du logiciel XC50/XC55, les captures d'écran peuvent différer de ce manuel, mais les procédures décrites sont valables par analogie.

Les caractéristiques générales des afficheurs XC50 et XC55, en particulier le modèle de données de base et les instructions de sécurité, sont documentées en détail dans le manuel d'utilisation. Leur connaissance est supposée.

Il faut également avoir des connaissances en programmation PLC.

#### **SPS soutenu**

En principe, votre PLC doit disposer d'un serveur web et doit être en mesure de fournir les données requises par le display.

Par exemple les contrôleurs suivants sont appropriés :

- Siemens S7 1x Famille
- Siemens S7 3 xx Famille avec serveur web à partir de la version V3.2

#### **Outils d'ingénierie soutenus**

L'outil d'ingénierie dépend du PLC utilisé.

- Portail TIA 15 pour les PLC de la famille S7 1x00
- STEP 7 pour les PLC de la famille S7 3xx

Dans ce manuel, la procédure est décrite à l'aide d'exemples pour les deux possibilités.

Dans ces cas-là, Siebert GmbH ne peut toutefois pas apporter de soutien spécifique.

#### **Protocole de transmission**

Le transport des données s'effectue toujours via le protocole http ou https. Vous devez saisir les données d'accès au PLC sur la page web du display.

#### **Intervalle d'interrogation**

Intervalle d'interrogation

L'intervalle d'interrogation est défini par les afficheurs des séries XC50/XC55 et peut être réglé dans une fourchette allant d'une seconde à 30 secondes. Vous pouvez définir cet intervalle à l'aide d'un curseur sur la page web „Data – PLC settings“.

---

## 4 Cas d'application typiques

---

Dans chaque cas d'application, vous définissez une mise en page sur l'afficheur. On choisit d'abord les propriétés qui doivent être commandées plus tard par l'API et on exporte les données nécessaires. Dans le fichier modèle XML exporté, le lien entre les éléments de mise en page de l'afficheur et les noms de variables de l'API est défini. De plus, un bloc de données correspondant est exporté et peut être utilisé dans le TIA Portal.

### Utiliser des noms de variables prédéfinis dans l'API

Si une CPU S7 1x00 est utilisée, vous pouvez importer le bloc de données exporté par l'afficheur dans l'outil d'ingénierie. Dans ce cas, les noms des variables dans l'API sont définis par l'afficheur.

Si une CPU S7 3xx est utilisée, le bloc de données doit être créé manuellement dans le logiciel STEP 7. Dans ce cas, les noms des variables doivent correspondre aux spécifications du fichier XML Template.

Dans l'API, utilisez les noms de variables exportés par l'afficheur.

### Utiliser ses propres noms de variables dans l'API

Vous pouvez sans problème utiliser vos propres noms de variables dans l'API. Dans ce cas, vous devez ouvrir le fichier XML Template exporté par l'afficheur dans un éditeur de texte et adapter manuellement les caractères de remplacement correspondants avant d'importer le fichier XML dans votre automate.

### Plusieurs afficheurs avec une mise en page identique

Créez votre application pour un afficheur. Si cette application fonctionne, la mise en page existante peut être téléchargée et activée sur d'autres afficheurs. Les données d'accès à votre API sont ensuite saisies sur ces afficheurs.

### Plusieurs afficheurs avec une mise en page différente et les mêmes noms de variables

L'un des principaux avantages du modèle de données orienté objet de l'afficheur réside dans la séparation de la mise en page et des données. Si différentes mises en page sont utilisées sur différents afficheurs et que les noms des éléments de mise en page restent identiques sur tous les afficheurs, aucune modification n'est nécessaire sur l'API.

Tous les afficheurs demandent les mêmes données de l'API et les affichent selon leur mise en page respective.

### Plusieurs afficheurs avec une mise en page et des variables différentes

Pour ce faire, vous pouvez exporter les données souhaitées sur chaque afficheur dans les PLC settings et les charger dans l'automate en tant que bloc de données propre. Ainsi, il y a un bloc de données pour chaque afficheur dans la commande.

### Commande supplémentaire via le site web

Il est possible de commander des éléments de l'afficheur par l'API et de régler manuellement les mêmes éléments ou d'autres éléments par le biais de la page web.

Si un élément est commandé à la fois via l'API et manuellement via le site web, c'est toujours le dernier accès qui est valable.

---

## 5 Démarrage rapide et déroulement général

---

### Démarrage rapide

Le démarrage rapide suivant se divise en quatre sections :

Démarrage rapide partie 1 :

Configuration de l'afficheur et création des fichiers pour l'automate. Cette partie est indépendante de l'API spécifique ou de l'outil d'ingénierie.

Démarrage rapide partie 2a :

Configuration de la commande et du serveur web dans le TIA Portal. Vous trouverez ici toutes les informations nécessaires pour les API 1x00 et le TIA Portal.

Démarrage rapide partie 2b :

Configuration de la commande et du serveur web avec STEP 7. Vous trouverez ici toutes les informations nécessaires pour les API 3xx et STEP 7.

Démarrage rapide partie 3 :

Conseils pour la recherche d'erreurs.

### Procédure générale

Tout d'abord, une mise en page est définie sur l'afficheur. Chaque élément de ce layout a un nom et un type uniques. Le nom permet de s'adresser ultérieurement à chaque élément individuellement. Le type permet de définir comment les données envoyées doivent être interprétées. Les éléments possibles et leurs propriétés sont décrits dans le mode d'emploi des afficheurs XC50/XC55.

Dans ce manuel, le layout 'PLC-Example' est utilisé comme exemple. Ce layout peut être téléchargé sur le site [www.siebert-group.com](http://www.siebert-group.com). Toutes les captures d'écran suivantes se réfèrent à cet exemple.

Les paramètres du PLC Connector se trouvent sur la page web 'Data - PLC settings' de l'afficheur. Vérifier La première étape consiste à vérifier et à enregistrer les données de connexion à l'API. Ces données sont ensuite enregistrées de manière permanente sur l'afficheur.

Dans la deuxième phase, on définit les propriétés des éléments qui seront utilisés par l'API. Pour sélectionner les éléments, on fait appel à l'"outil d'exportation". Une nouvelle page web s'ouvre alors sur laquelle tous les éléments du layout sont disposés dans une arborescence. Sélectionnez les éléments souhaités dans cette arborescence et enregistrez votre sélection.

L'afficheur crée alors un fichier zip contenant un module de données API et un fichier XML.

Importez le bloc de données et le fichier XML dans votre commande. Dans le bloc de données, les variables correspondantes sont définies pour toutes les valeurs qui doivent être affichées sur l'afficheur. Le fichier XML définit le modèle pour le fichier XML généré par l'API avec les valeurs à afficher.

Activez le serveur web propre à l'API de la commande.

Créez un nouvel utilisateur sur l'API pour la communication avec l'afficheur. L'accès de l'afficheur à l'API s'effectuera ultérieurement via cet utilisateur. Le nom d'utilisateur et le mot de passe doivent correspondre aux indications que vous avez données auparavant sur l'afficheur sous 'Display-IP-Addr - Data - PLC settings'.

Ensuite, le serveur web de l'API reprend de manière cyclique les données souhaitées de l'automate et les met à disposition sur son site web.

Les afficheurs XC50/XC55 s'enregistrent automatiquement en tant qu'utilisateur sur le serveur web de l'API. Une fois l'authentification réussie, les afficheurs demandent de manière cyclique les données souhaitées au serveur web de l'API et les affichent sur le display.



## 6 Démarrage rapide Partie 1 : Configuration de l'afficheur et création des fichiers pour le contrôleur

### Téléchargement des fichiers PLC-Exemple sur le site de Siebert

Tous les fichiers nécessaires à ce démarrage rapide sont disponibles sur le site Siebert [www.siebert-group.com](http://www.siebert-group.com) dans la zone de téléchargement des afficheurs XC50/55.

Le fichier 'PLC-Connector-Exemple.Zip' contient les fichiers :

- Le fichier zip PLC-Exemple.Zip avec l'exemple de mise en page. Ce fichier doit être chargé sur l'afficheur.
- Ce mode d'emploi
- Un exemple de projet pour le TIA Portal, voir Quick Start partie 2a.
- Un exemple de projet pour STEP 7, voir Quick Start partie 2b.

Sauvegardez le fichier 'PLC-Connector-Exemple.Zip' à un endroit approprié sur votre PC et décompressez le contenu dans le répertoire PLC-Connector-Exemple.

La mise en page d'exemple a une taille de 32x16 pixels. Vous pouvez l'utiliser sur tous les afficheurs Siebert de la série XC50/XC55.

### Téléchargement de l'exemple de mise en page sur l'afficheur

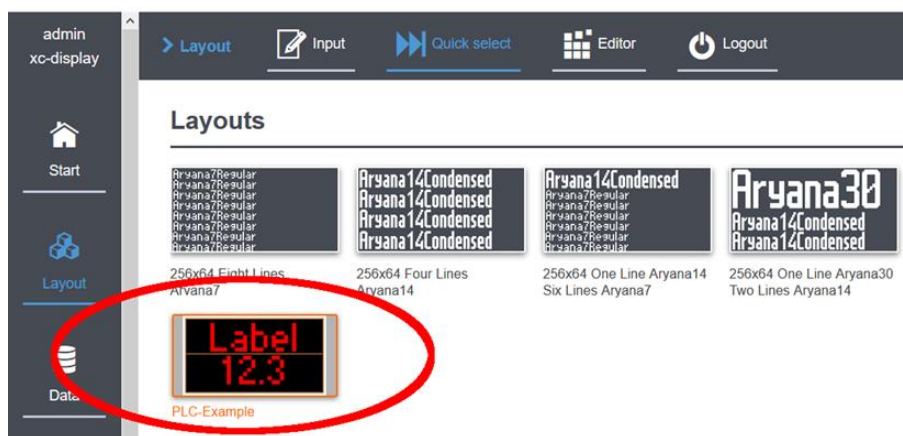
Accédez à la page web de l'afficheur.

Allez dans l'option 'Utilities - File manager' et cliquez sur le bouton Upload pour télécharger le fichier zip 'PLC-Exemple.Zip' sur l'afficheur.

Ce fichier contient toutes les données nécessaires à ce démarrage rapide.

### Sélection de la mise en page sur l'afficheur

Après avoir téléchargé le fichier zip, vous trouverez sur la page web 'Layout - Quick Select' le layout 'PLC- Exemple'. Activez ce layout en cliquant dessus et confirmez la demande de sécurité. Le layout actif est marqué en couleur :



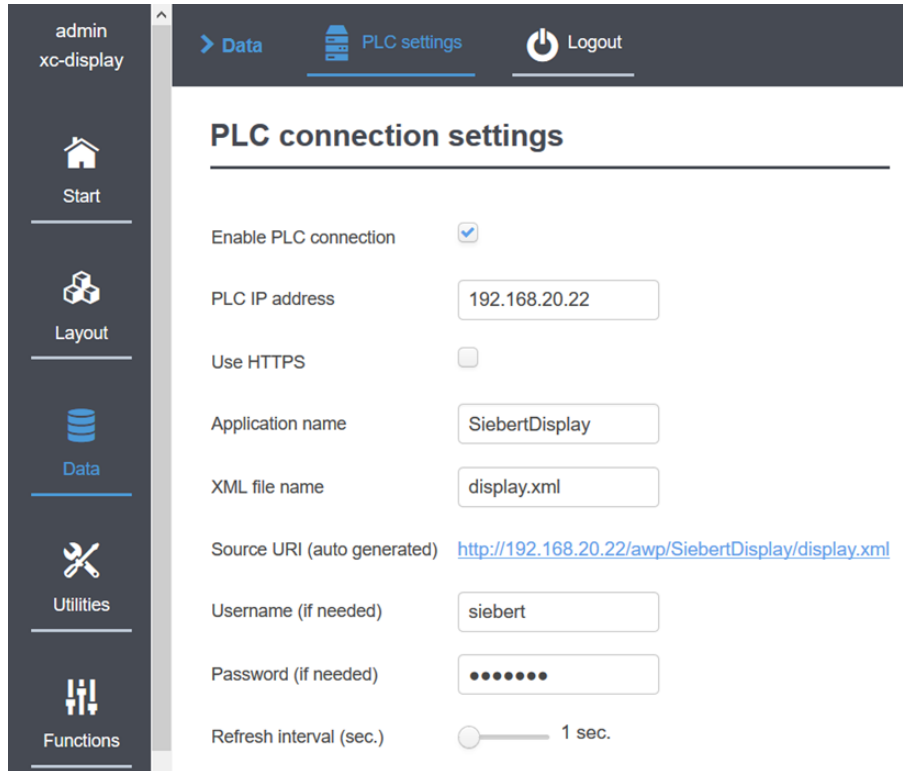
Cet exemple de layout se compose d'un champ de texte nommé 'txt1' et d'un champ numérique nommé 'num1'. Le contenu du texte et la couleur du champ de texte doivent être commandés à partir de l'API. Pour le champ numérique, la valeur numérique et la couleur doivent également être commandées.

La correspondance entre les éléments de l'afficheur et les variables de l'automate s'effectue via un DB et un fichier XML. Dans les deux cas, il s'agit de fichiers ASCII lisibles par l'homme, qui peuvent être ouverts et modifiés dans n'importe quel éditeur de texte.

Les deux fichiers doivent correspondre à la mise en page active sur l'afficheur.

## Saisie des données de connexion sur la page web de l'afficheur

Sur la page web 'Data - PLC settings', saisissez l'adresse IP de la commande, les données d'accès et les modes de transmission vers le serveur web. Les mêmes données d'accès doivent être saisies sur le serveur web de la commande.



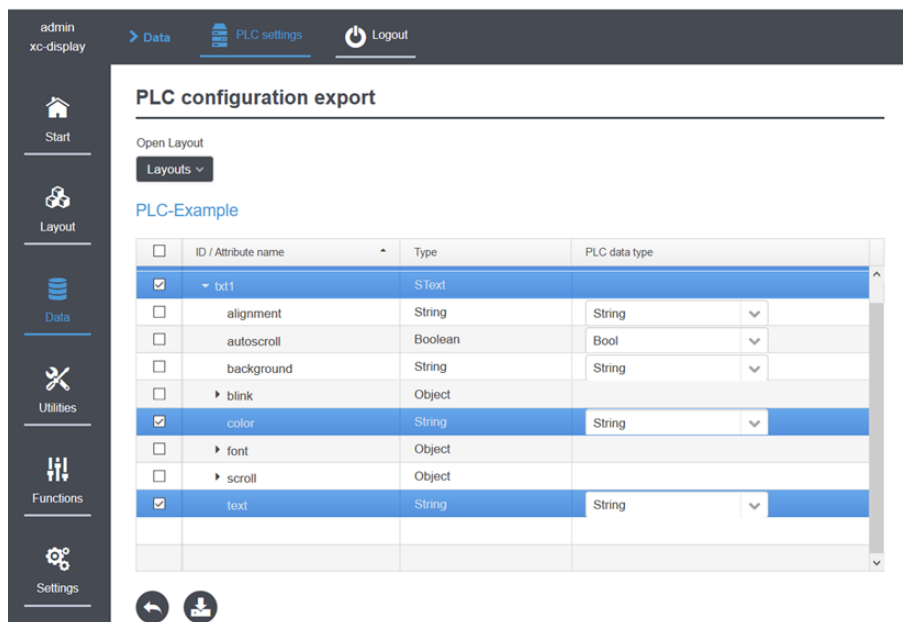
Remarque : Le nom que vous donnez dans le champ <XML file name> sera utilisé plus tard dans l'API comme nom du bloc de données par lequel les données seront mises à disposition pour l'afficheur.

## Sélection des éléments pour l'exportation de la configuration de l'API

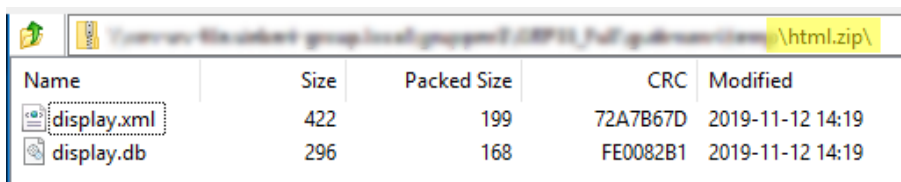
Toujours sur la page web 'Data - PLC settings', appelez l'"Export tool". Sur la page suivante, vous pouvez sélectionner les éléments qui doivent être remplis par l'automate.

Dans cet exemple, il s'agit du contenu et de la couleur du champ de texte 'txt1' et de la valeur numérique et de la couleur du champ numérique 'num1'.

Recommandation : ne sélectionnez ici que les éléments et propriétés qui doivent vraiment être pilotés par l'API. Vous éviterez ainsi des fichiers inutilement volumineux et des variables inutiles sur l'API.



En cliquant sur le bouton d'exportation, les fichiers nécessaires sont créés et enregistrés sur le PC sous le nom 'html.zip'. Les deux fichiers lisibles par l'homme, display.db et display.xml, sont enregistrés dans ce fichier zip. Le nom du fichier est repris du champ 'XML file name' de la page web 'Data - PLC settings'.



Name	Size	Packed Size	CRC	Modified
display.xml	422	199	72A7B67D	2019-11-12 14:19
display.db	296	168	FE0082B1	2019-11-12 14:19

### Contenu du fichier XML

Le fichier XML est requis par le serveur web de l'automate comme modèle dans lequel les contenus des variables sont transférés en cours de fonctionnement. Les mots-clés de ce fichier sont définis par l'afficheur et sont décrits dans le mode d'emploi XC50/XC55.

Dans cet exemple, le fichier XML a la structure suivante :

```
<?xml version="1.0" encoding="UTF-8"?><siebert-display>
  <meta>
    <request>setData</request>
    <version>1</version>
  </meta>
  <data>
    <SText id="txt1">
      <text>:= "display".txt1.text:</text>
      <color>:= "display".txt1.color:</color>
    </SText>
    <SNumeric id="num1">
      <value>:= "display".num1.value:</value>
      <color>:= "display".num1.color:</color>
    </SNumeric>
  </data>
</siebert-display>
```

Pour les parties entre les deux points, par ex : := "display".txt1.text : Il s'agit de caractères de remplacement pour les variables qui seront remplies ultérieurement par l'API avec les valeurs actuelles..

### Contenu du fichier de modules de données

Le bloc de données permet à l'automate de transmettre le contenu de certaines variables au serveur web. Celui-ci crée, avec le fichier XML qui sert de modèle, un nouveau fichier XML avec des données pour l'afficheur. Dès que les données de l'automate sont modifiées, le fichier XML contenant les données pour l'afficheur est automatiquement mis à jour.

Dans l'exemple, le BD a la structure suivante et définit les variables dans l'automate:

```
DATA_BLOCK "display"
{ S7_Optimized_Access := 'TRUE' }
VERSION : 0.1
NON_RETAIN
VAR
    "txt1" : Struct
        "text" : String;
        "color" : String;
    END_STRUCT;
    "num1" : Struct
        "value" : Int;
        "color" : String;
    END_STRUCT;
END_VAR
```

## 7 Quick Start partie 2a : Configuration de l'automate et du serveur web dans le TIA Portal

Vous pouvez télécharger le projet TIA Portal utilisé pour les captures d'écran suivantes sur notre site web.

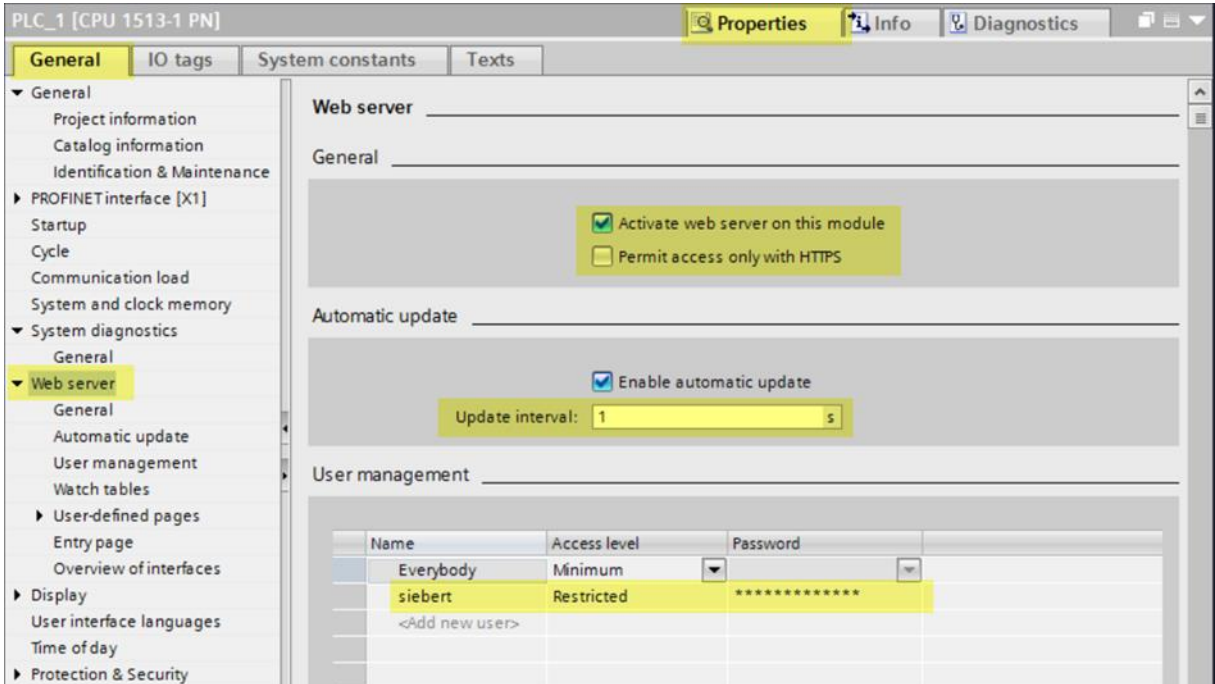
Pour l'API, les étapes suivantes doivent être effectuées :

### Mise en marche du serveur web et création d'un utilisateur

Activez le serveur web de l'API.

Saisissez ensuite le nom d'utilisateur, le mot de passe et les modes de transmission dans le serveur web de l'API.

Ces informations doivent correspondre aux paramètres saisis sur la page web de l'afficheur.



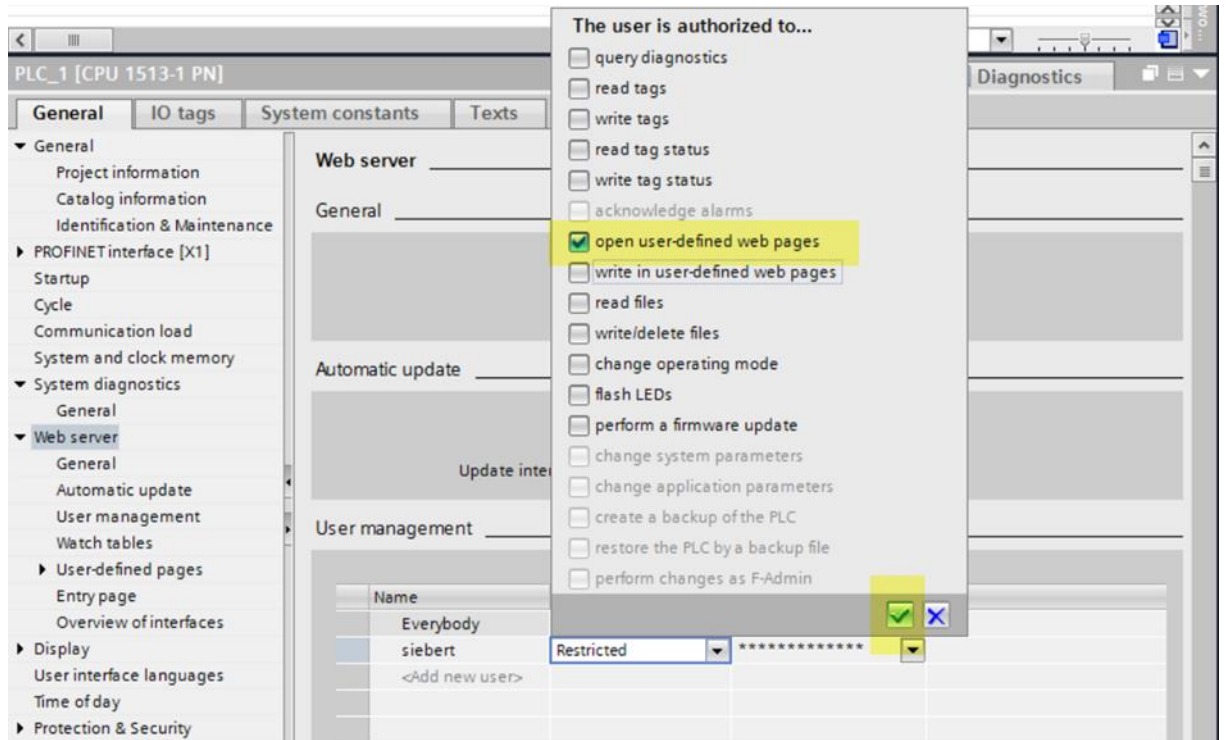
The screenshot shows the 'Web server' configuration page in the TIA Portal. The left-hand navigation pane is expanded to 'Web server'. The main area shows the following settings:

- Web server**
  - Activate web server on this module
  - Permit access only with HTTPS
- Automatic update**
  - Enable automatic update
  - Update interval: 1 s
- User management**

Name	Access level	Password
Everybody	Minimum	
siebert	Restricted	*****
<Add new user>		

## Définir les droits des utilisateurs

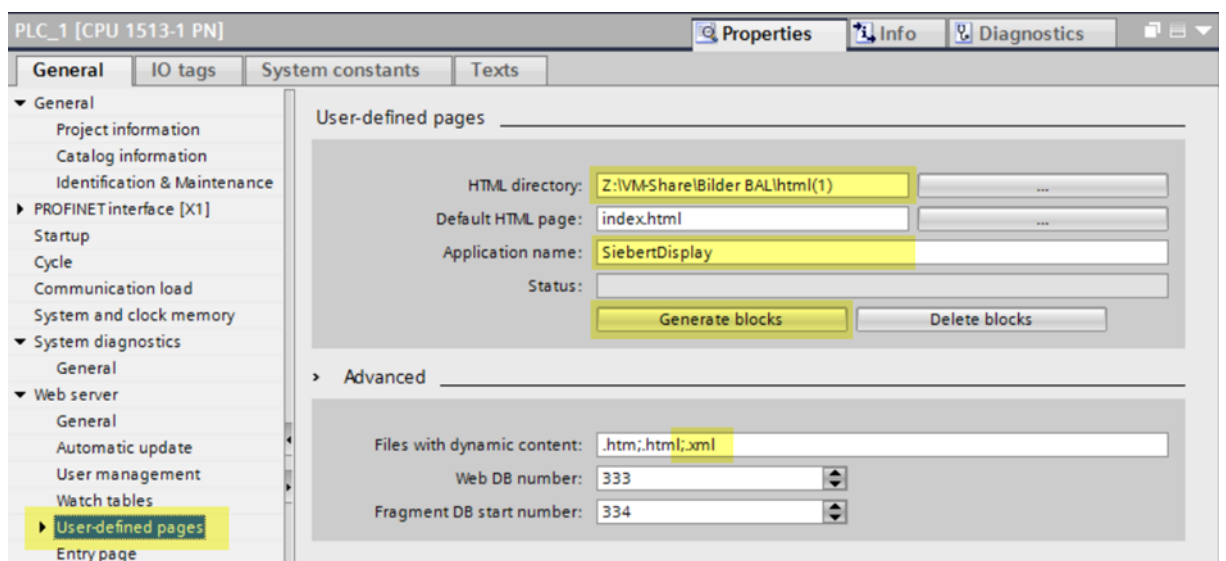
Activez la case à cocher sélectionnée pour permettre à l'utilisateur, dans ce cas donc à l'afficheur XC50/XC55, d'interroger les données du serveur web de l'automate.



## Télécharger la base de données et le fichier XML dans l'automate

Téléchargez le DB et le fichier XML générés par l'afficheur dans l'automate.

Important : dans le champ <Advanced / Field with dynamic content>, inscrivez impérativement l'entrée .xml.



Copiez le contenu du fichier html.zip que vous avez exporté de l'afficheur à la première étape dans le répertoire indiqué dans le champ <User-defined pages / HTML directory> de la boîte de dialogue ci-dessus.

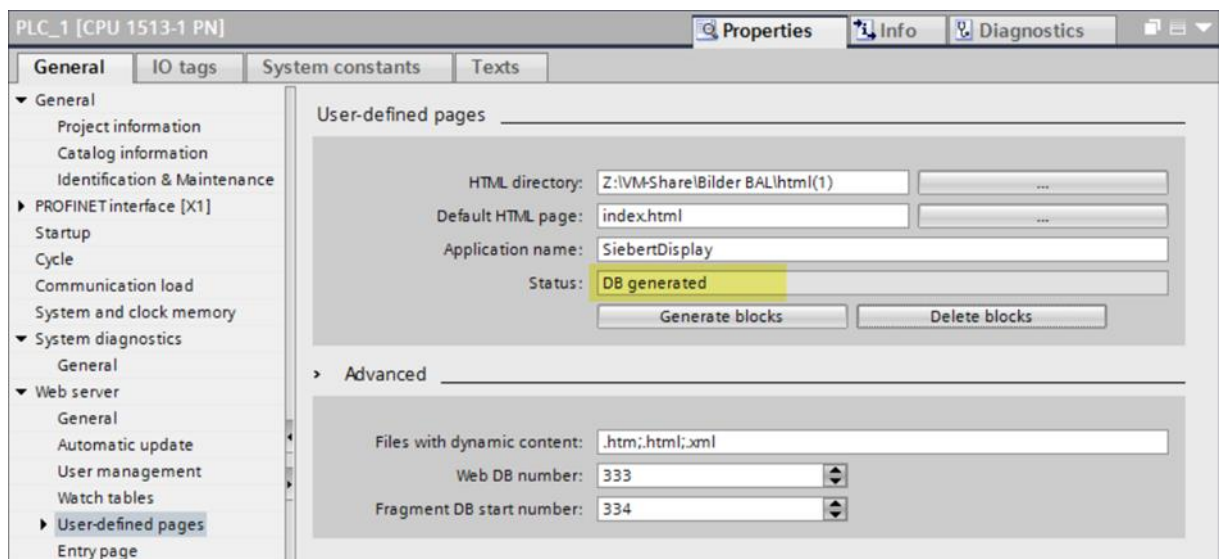
Si vous avez suivi l'exemple de démarrage rapide, les deux fichiers display.xml et display.db doivent se trouver dans ce répertoire.

Importez maintenant le fichier display.db dans l'outil d'ingénierie et créez le bloc de données.

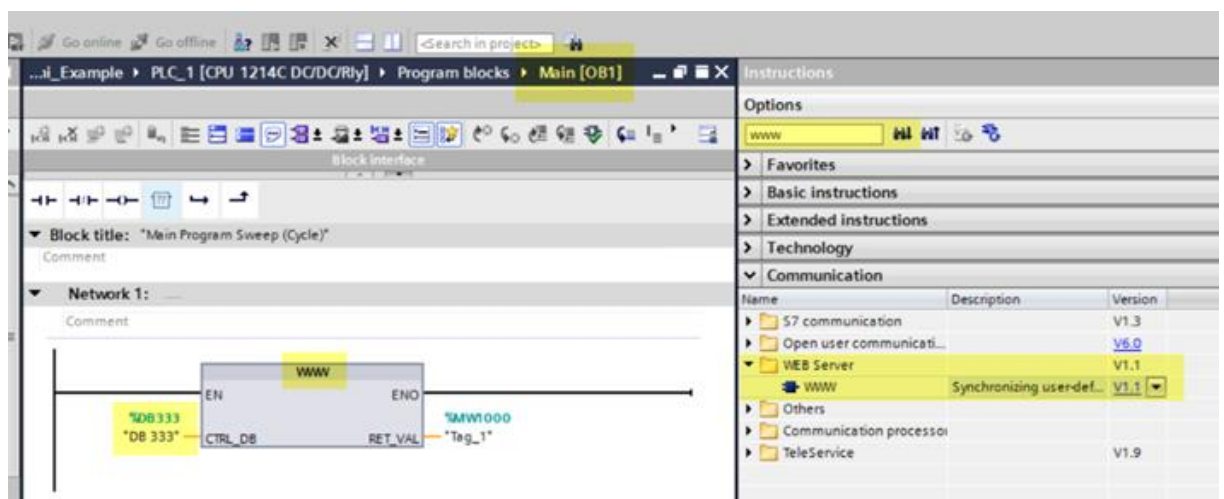
Ceci s'applique par analogie si vous avez donné un autre nom au fichier XML sur la page web de l'afficheur dans les paramètres PLC.

### Générer un module de serveur web

En cliquant sur le bouton 'Générer des blocs', les fichiers sont analysés par la commande. En cas de succès, le statut DB generated s'affiche.



### Appel cyclique de la page web

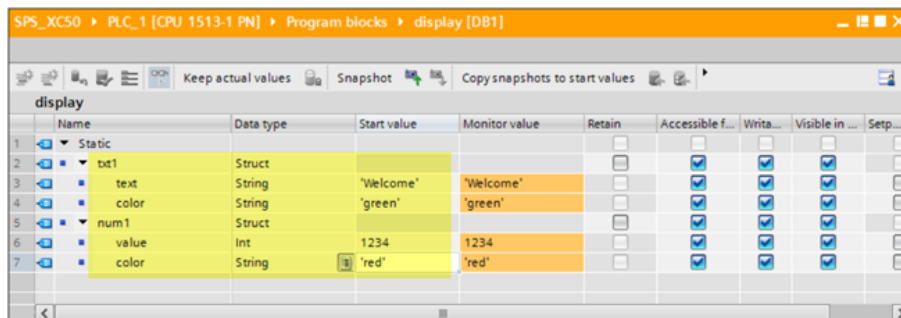


## Fournir des données pour l'afficheur

Ouvrez le bloc de données généré.

Toutes les variables définies y sont affichées et peuvent être envoyées à l'automate à partir du bloc de données. Le serveur web de l'automate actualise alors son fichier de sortie. Lors de la prochaine interrogation cyclique des données de commande par l'afficheur, les nouvelles données seront lues et affichées sur l'afficheur.

Important : l'afficheur vérifie les données reçues. La représentation sur l'afficheur n'est actualisée que si les données sont correctes. Veillez en particulier à ce que les valeurs saisies pour les couleurs soient valables : 'red', 'green', 'blue', 'orange', 'yellow', 'black', 'white', 'cyan'.

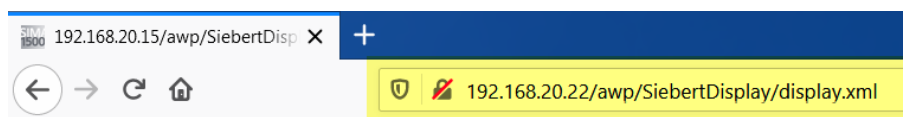


	Name	Data type	Start value	Monitor value	Retain	Accessible f...	Writa...	Visible in ...	Setp...
1	Static								
2	txt1	Struct							
3	text	String	'Welcome'	'Welcome'		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4	color	String	'green'	'green'		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	num1	Struct							
6	value	Int	1234	1234		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
7	color	String	'red'	'red'		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

## Contrôle de fonctionnement

Comme les données mises à disposition par le serveur web de l'automate sont disponibles sous forme de fichier XML, un contrôle simple est possible. Pour ce faire, connectez-vous directement au serveur web de l'API à l'aide d'un navigateur.

Avec l'URL <IP Adresse>/awp/SiebertDisplay/display.xml vous pouvez visualiser le contenu du fichier XML. Ce fichier devrait avoir exactement la même structure que le fichier modèle XML. Toutefois, au lieu des références aux variables dans la commande, les valeurs actuelles sont inscrites dans ce fichier :



```

<-siebert-display>
  <-meta>
    <request>setData</request>
    <version>1</version>
  </meta>
  <-data>
    <-SText id="txt1">
      <text>'Welcome'</text>
      <color>'green'</color>
    </SText>
    <-SNumeric id="num1">
      <value>1234</value>
      <color>'red'</color>
    </SNumeric>
  </data>
</siebert-display>

```



## 8 Démarrage rapide Partie 2b : Configuration de l'automate et du serveur web avec STEP 7

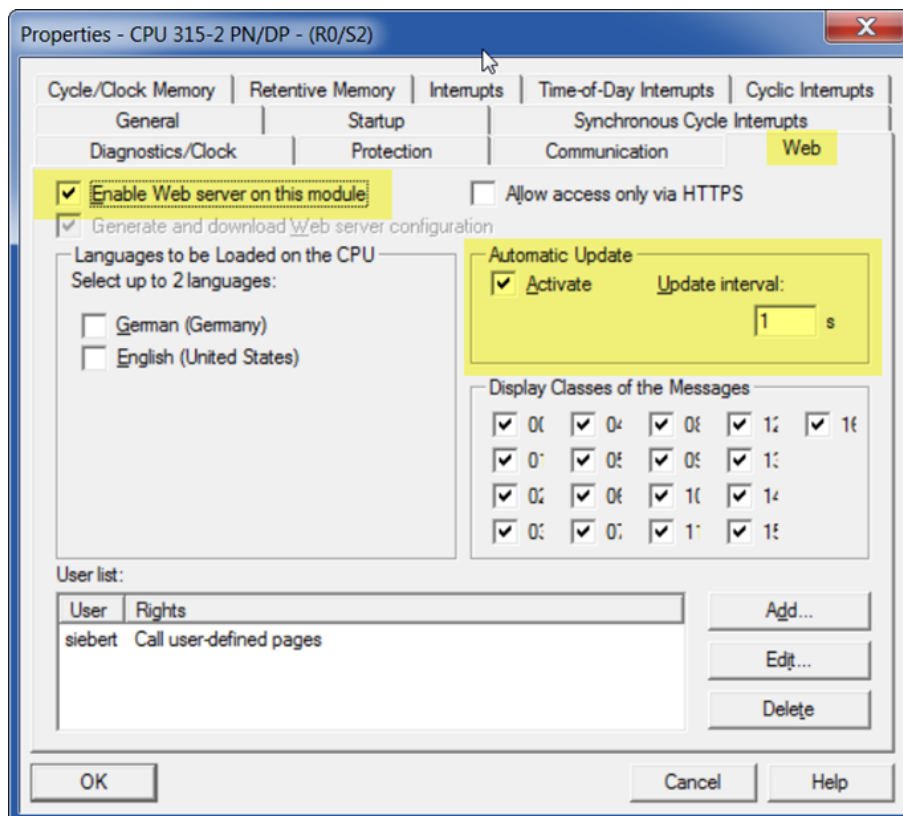
Si vous utilisez un API S3xx, la programmation s'effectue avec le logiciel STEP 7. Pour cela, suivez les étapes suivantes :

La création de la mise en page est décrite dans la partie 1 du démarrage rapide et est supposée être effectuée ici.

### Vérifier le serveur web de l'unité centrale

Pour pouvoir utiliser le connecteur API avec un S7-3xx ou un S7-4xx, il faut s'assurer que les contrôleurs disposent d'un serveur web et qu'ils peuvent traiter des pages HTML propres à l'utilisateur.

Vérifiez d'abord cette condition dans la configuration matérielle de l'API. Pour ce faire, double-cliquez sur l'icône de l'automate dans la configuration matérielle. Dans la boîte de dialogue 'Propriétés', un onglet 'Web' doit apparaître et la case à cocher 'Activer' doit pouvoir être sélectionnée dans la zone 'Actualisation automatique'.



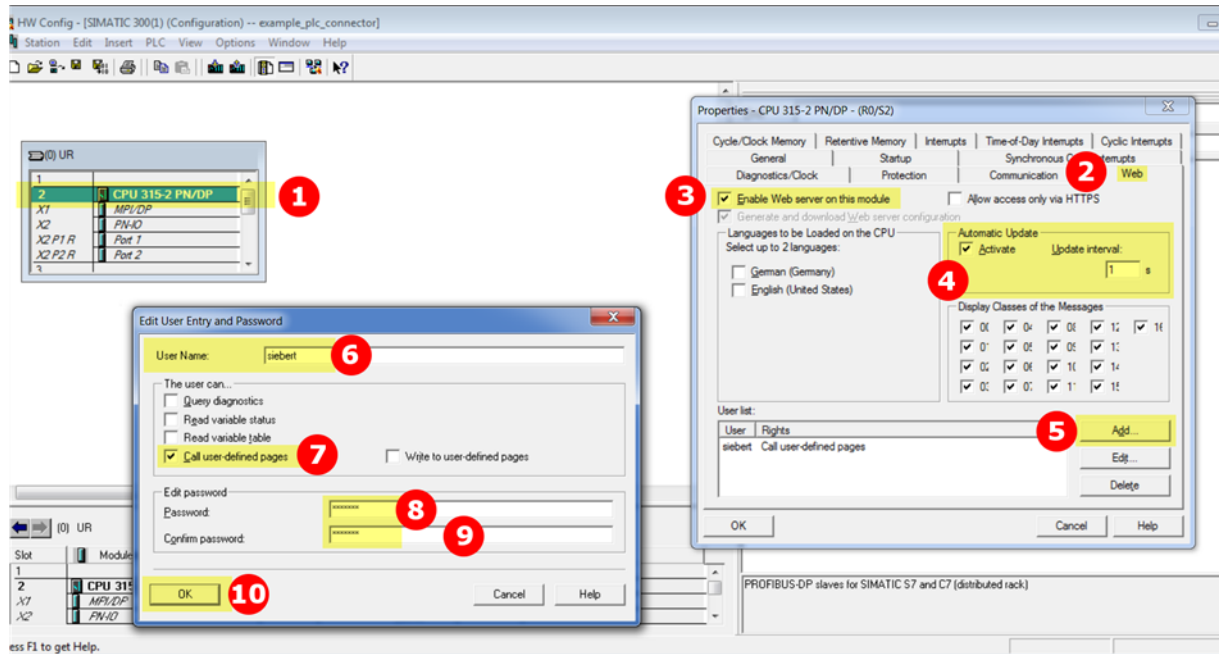
### Utiliser les fichiers exportés par l'afficheur

Copier le fichier XML exporté par l'afficheur dans un répertoire nommé 'html'. Le fichier '.db' également exporté n'est pas nécessaire avec STEP 7.

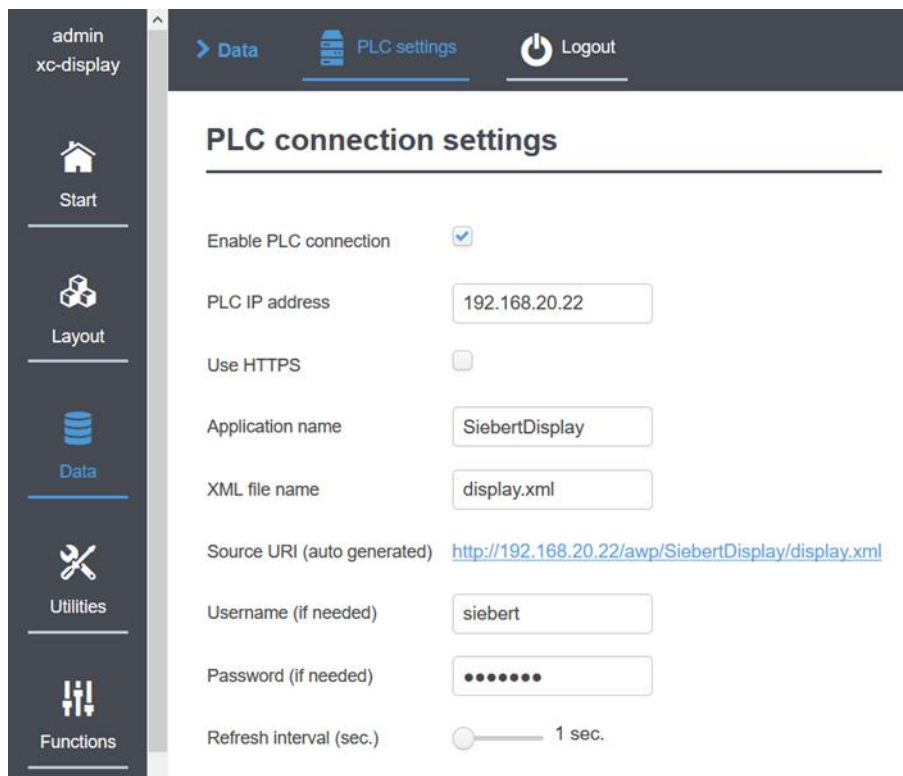


## Créer un nouveau projet STEP 7

Démarrez le STEP7 Manager V5.6, créez un nouveau projet et configurez le contrôleur dans la configuration matérielle selon l'illustration suivante :

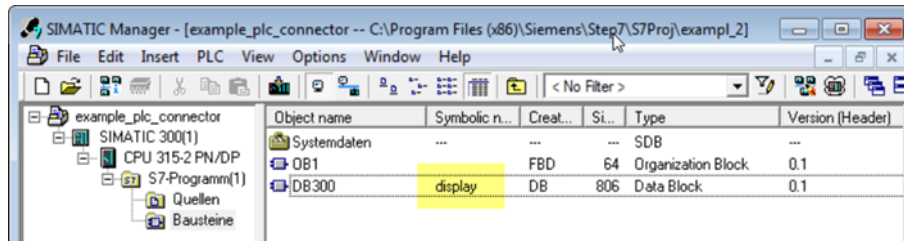


Saisissez le nom d'utilisateur et le mot de passe défini sur la page web 'Data - PLC settings' de l'afficheur et enregistrez les données en cliquant sur le bouton Enregistrer.

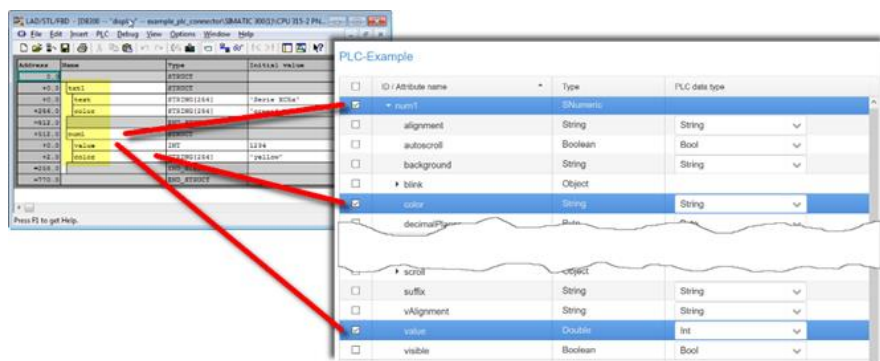


## Créer un bloc de données

Créez un bloc de données. Le nom du module de données doit être identique au nom du fichier XML.

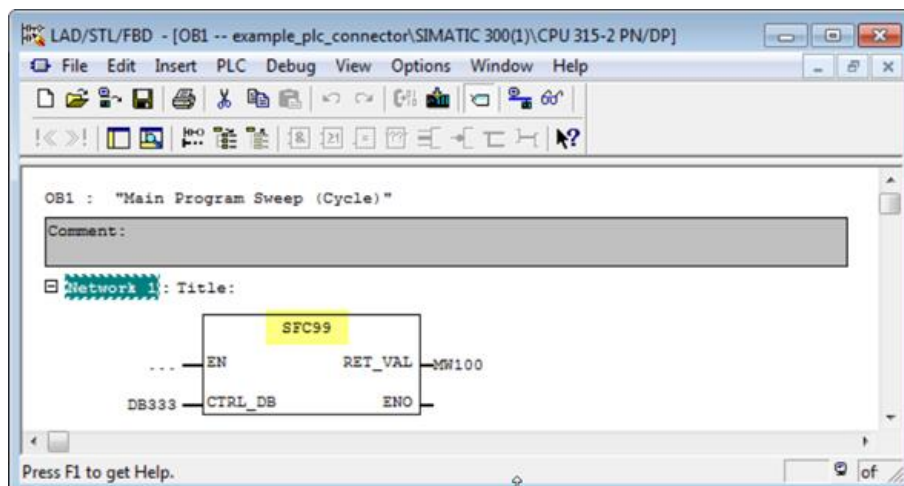


Saisissez maintenant les noms des variables dans le bloc de données. Les noms et les types doivent correspondre exactement aux caractères de remplacement dans le fichier XML :

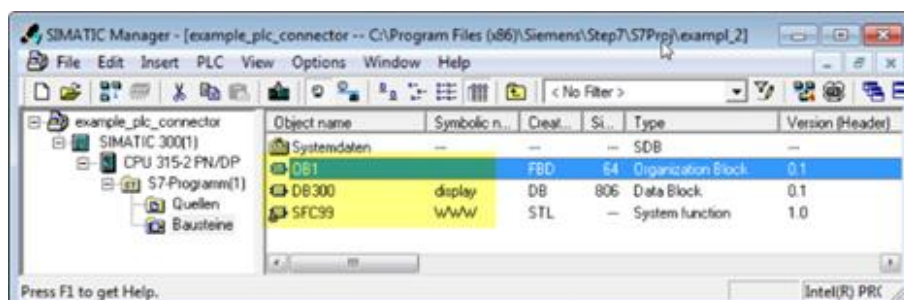


## Créer le SFC99

Activez le serveur web via le SFC99.

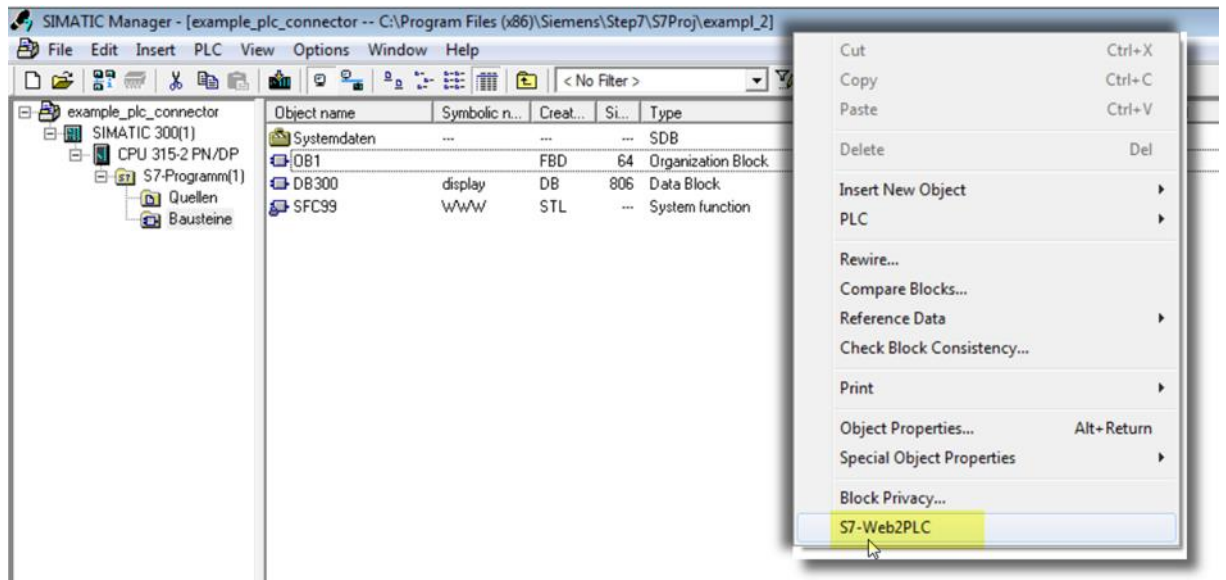


Ensuite, les éléments suivants devraient être présents dans votre projet:

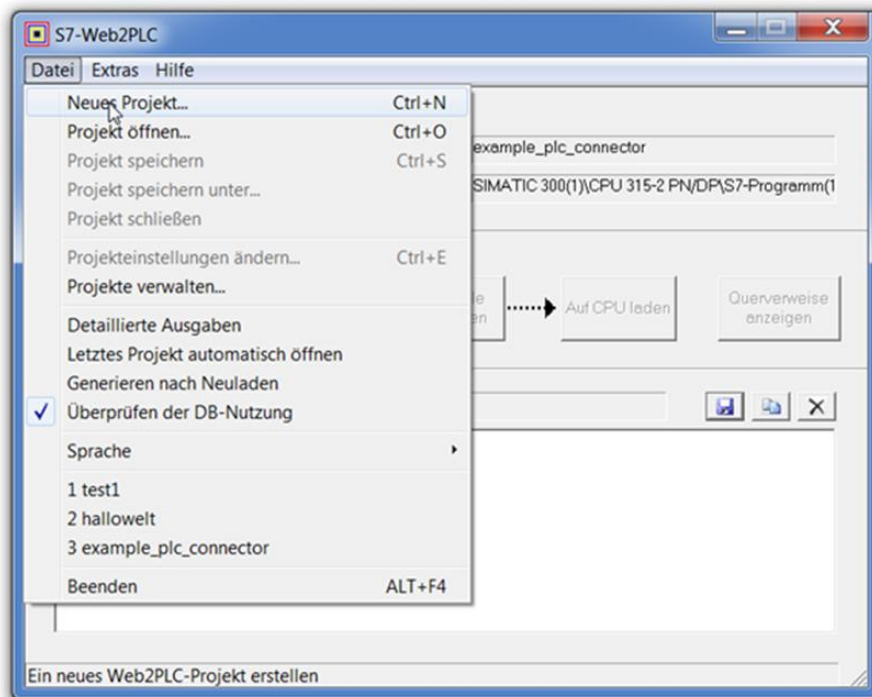


## Démarrer Web2PLC

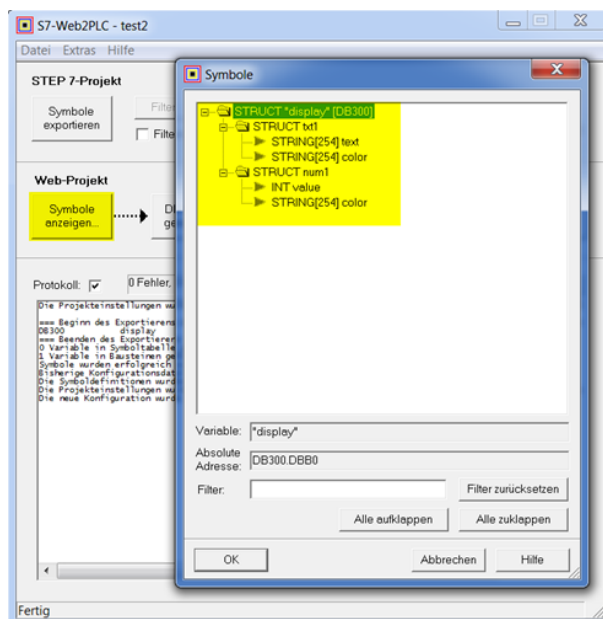
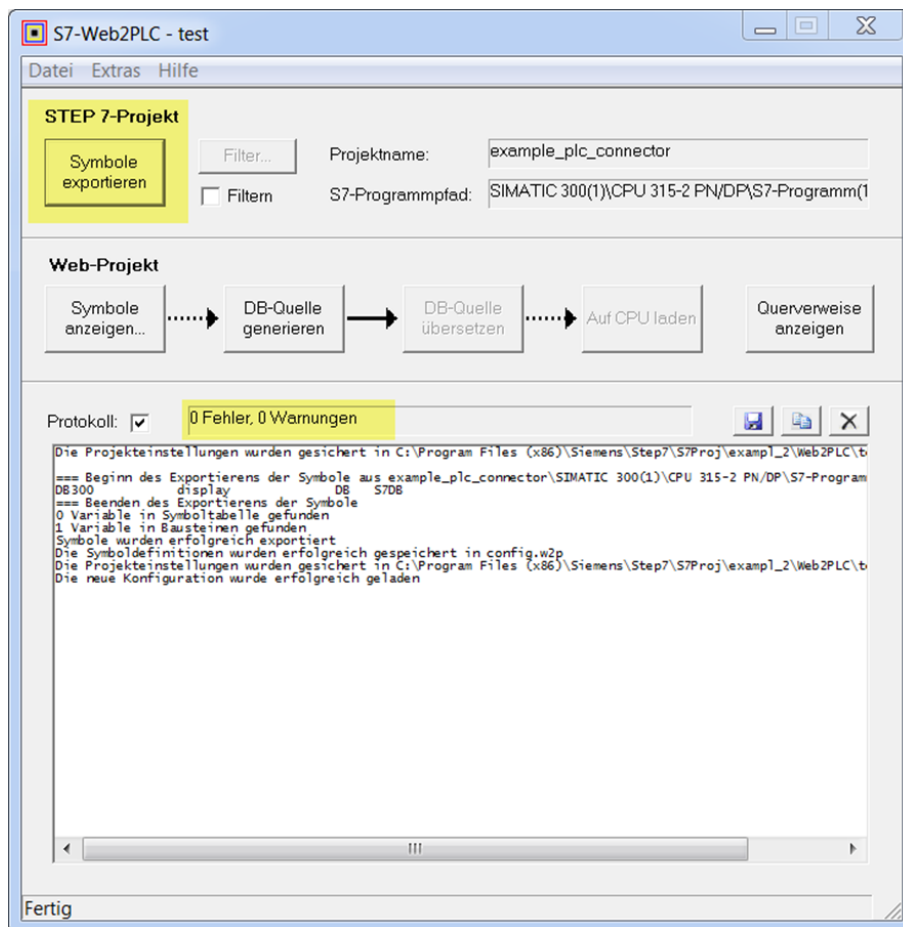
Appeler le menu contextuel en cliquant avec le bouton droit de la souris dans le champ de l'objet et démarrez le programme S7-Web2PLC.



## Créer un nouveau projet Web2PLC



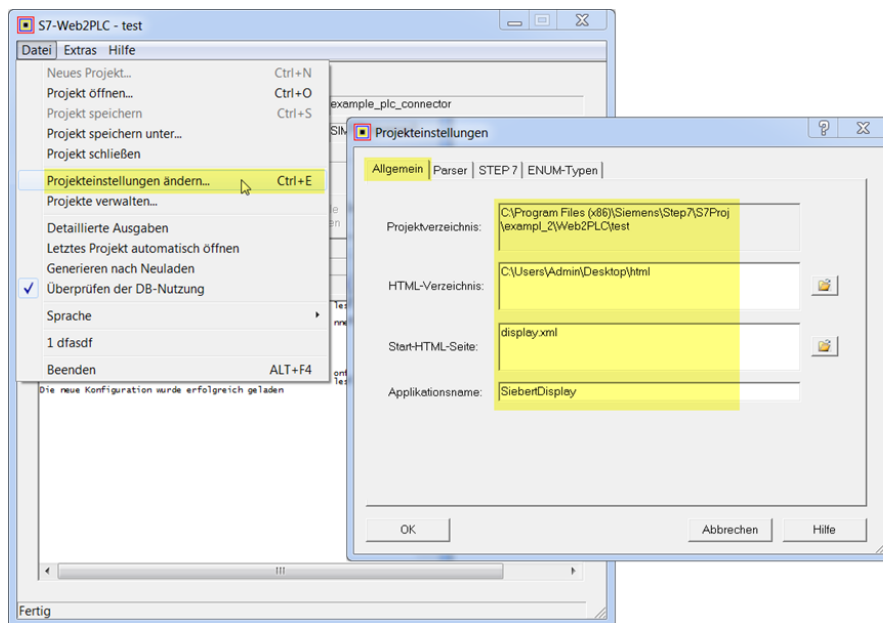
## Exportation des données des blocs de données du projet S7



## Ajuster les paramètres de projection

Dans la boîte de dialogue suivante "Paramètres du projet", saisissez les données correspondant à la mise en page :

- Le lieu du répertoire HTML.
- Le fichier qui doit être affiché sur la page HTML de départ : Sélectionnez ici le fichier XML que vous avez exporté de l'afficheur. Remarque : Modifiez le filtre de données sur "\*.xml".
- Dans le champ nom de l'application, saisissez le même nom que celui que vous avez défini sur la page web de l'afficheur.

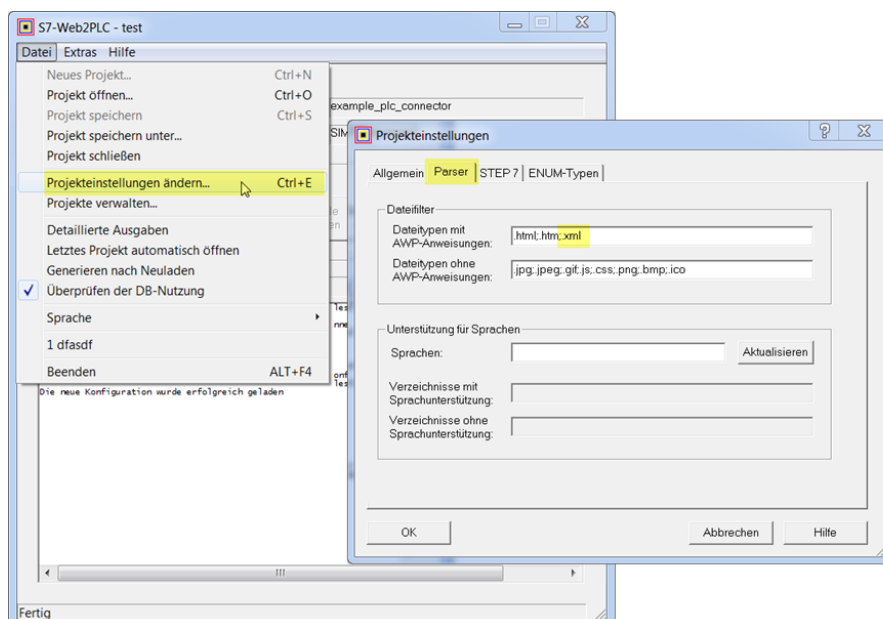


Allez maintenant dans l'onglet 'Parser'.

Important :

Ajoutez le type ".xml" à la ligne "Types de données avec instructions AWP :".

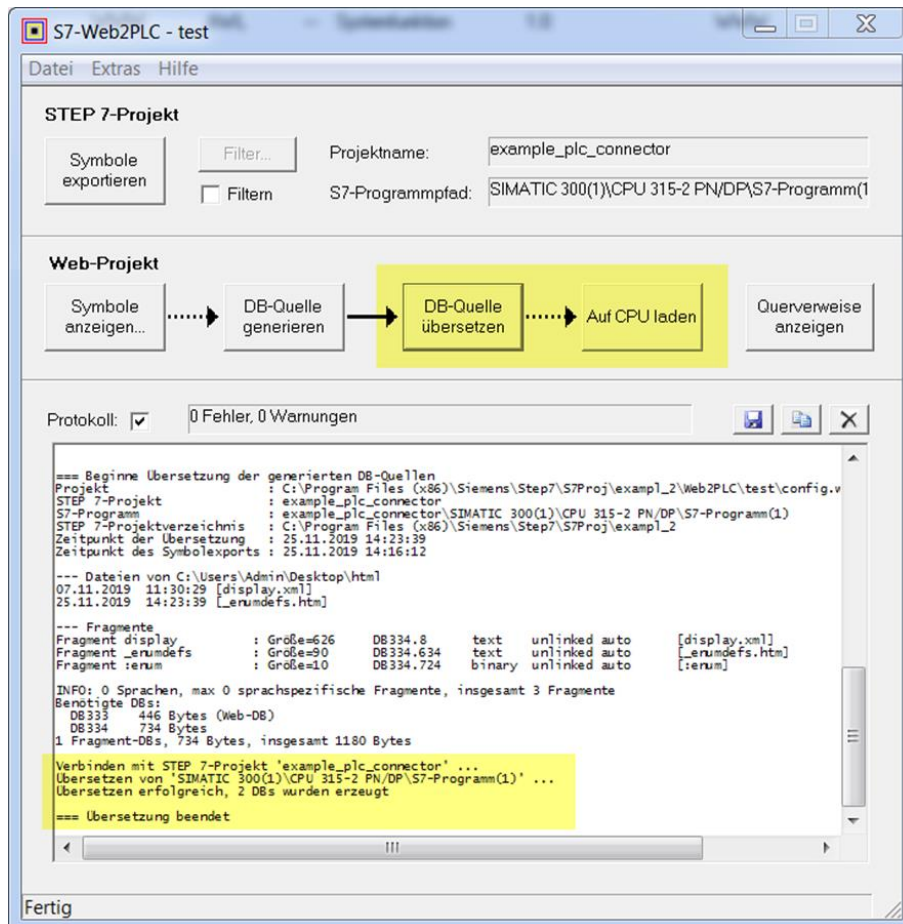
Supprimez le type ".xml" dans la ligne "Types de fichiers sans instructions AWP :".



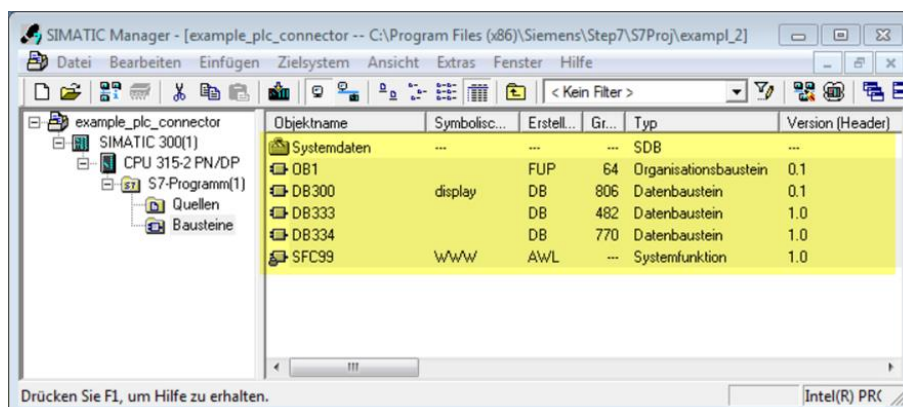


## Traduction de la source des blocs de données et chargement du projet web sur le CPU

Si aucune erreur n'a été trouvée lors de la traduction, vous pouvez maintenant charger le projet web sur l'unité centrale.

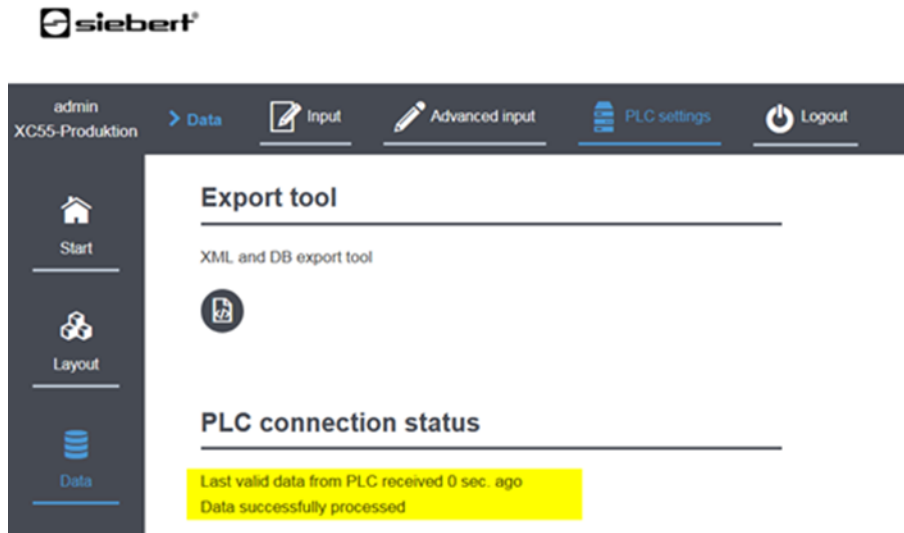


## Télécharger le projet complet dans l'unité centrale

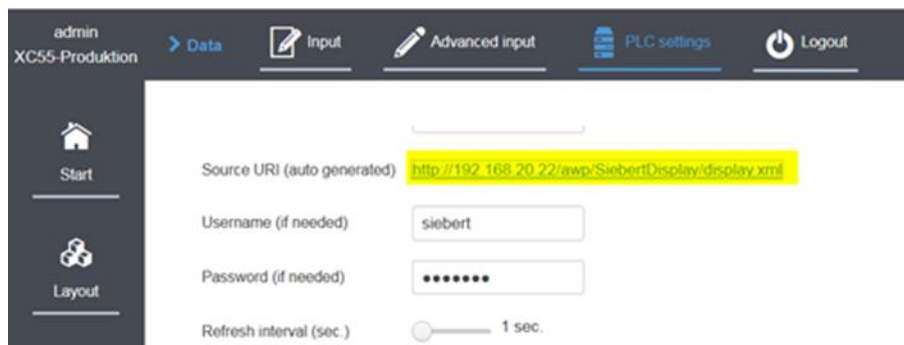


## Contrôle de la page web de l'afficheur

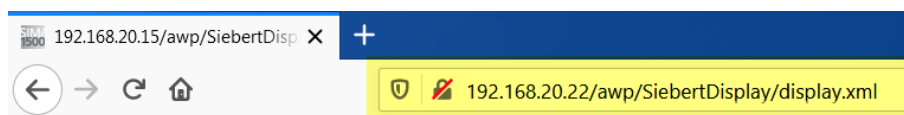
Sur la page web "Data - PLC Settings", vérifiez les messages de l'afficheur dans la section 'PLC connection status'.



En cliquant sur le lien dans la ligne "Source URI (auto generated)", le fichier XML mis à disposition par l'API s'ouvre



Sur la page web qui s'affiche maintenant, on peut voir le contenu des variables de l'API. Les ID contiennent les noms des éléments de mise en page de l'afficheur:



```

<!--siebert-display-->
<!--meta-->
  <request>setData</request>
  <version>1</version>
</meta-->
<!--data-->
  <!--SText id="txt1"-->
    <text>'Welcome'</text>
    <color>'green'</color>
  </SText-->
  <!--SNumeric id="num1"-->
    <value>1234</value>
    <color>'red'</color>
  </SNumeric-->
</data-->
</siebert-display-->

```



## 9 Démarrage rapide Partie 3 : Conseils de dépannage

Les afficheurs XC50/XC55 indiquent sur la page web 'Data - PLC Settings', dans la partie inférieure, le résultat du dernier accès aux données de l'API. Les messages d'erreur peuvent être regroupés en plusieurs catégories.

- Messages HTTP de statut du serveur web.  
Ces messages sont standardisés par HTTP. Leur signification se trouve sur Internet.
- Messages lors de l'évaluation des données envoyées  
Ces messages sont générés par l'afficheur XC50/XC55. Dans ces cas, la connexion à l'API a été établie avec succès et les messages se rapportent à l'évaluation des données reçues.

### Remarques générales

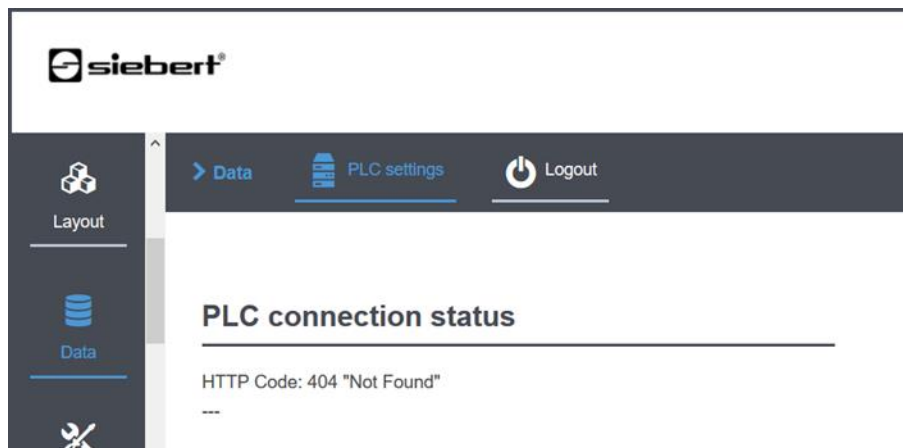
À chaque attribut que vous exportez, vous devez également attribuer une variable dans l'API.

Lors de l'exportation de la configuration PLC, ne sélectionnez donc que les attributs que vous souhaitez réellement commander à partir de l'API. Vous réduisez ainsi votre travail de programmation au strict nécessaire.

Lors de l'importation des données dans l'API, il n'est pas possible d'importer des valeurs par défaut. Attribuez donc à chaque variable utilisée des valeurs pertinentes.

### Messages de code HTTP

En cas de messages d'erreur imputables au serveur web, le code HTTP signalé est affiché. Un code d'erreur typique en cas d'erreur de frappe de l'adresse IP est par exemple :



## Messages du protocole

<pre>Last valid data from PLC received x sec. ago  Data successfully processed</pre>	<p>La connexion a été établie avec succès, le XML a été lu et analysé sans erreur.</p>
<pre>timeout('time out',) ---</pre>	<p>La connexion au serveur web de l'API a échoué.</p> <p>Causes possibles des erreurs :</p> <p>L'API n'est pas activée Les adresses IP ne sont pas compatibles Le nom d'utilisateur ou le mot de passe sont incorrects</p>
<pre>BadStatusLine(' ',) ---</pre>	<p>Le fichier XML a été lu. Une erreur est survenue lors de l'analyse du fichier.</p> <p>PAR EX : Noms inconnus si les noms des éléments de mise en page de l'afficheur ne correspondent pas aux noms des variables de l'API.</p>
<pre>Could not set shortcut value.: Neither color name nor hex value #rrggbb given</pre>	<p>Une valeur incorrecte a été transmise pour un élément avec la propriété color, par exemple une chaîne vide car aucune couleur n'a encore été définie.</p>

## Contrôle du fichier XML

Vérifier si le fichier xml est lu par l'API en cliquant sur le lien de la page d'affichage <Data / PLC settings>.

Vérifiez également que le fichier xml a la bonne forme. En particulier, les caractères de remplacement <:="name"....> doivent être remplacés par des valeurs pertinentes. Si ce n'est pas le cas, il est possible que l'entrée .xml ait été oubliée dans les paramètres du serveur web.

## Caractères spéciaux dans le fichier XML

Le codage des caractères spéciaux dans le fichier XML suit les directives du W3C.

Si vous souhaitez afficher les caractères suivants sur l'afficheur, ceux-ci doivent être codés dans le fichier XML doivent être codés comme suit :

Caractère	Description	Symbolique	Hexadécimal	Nombre décimal
"	Guillemets en haut	&quot;	&#x22;	&#34;
&	igne d'ampère, commercial et	&amp;	&#x26;	&#38;
<	parenthèse pointue ouvrante	&lt;	&#x3C;	&#60;
>	parenthèse fermante pointue	&gt;	&#x3E;	&#62;
'	apostrophe, guillemet simple en haut	&apos;	&#x27;	&#39;

Dans de tels cas, utilisez le type de données 'WString' dans l'API.

Veuillez noter que l'affichage du fichier XML dépend du navigateur web utilisé. En cas de doute, vous pouvez enregistrer le fichier XML localement sur votre système à partir du navigateur web et examiner le contenu avec un éditeur de texte.

Par exemple, si vous souhaitez afficher le texte '<Hello>' en bleu sur le champ 'txt1' de l'exemple de mise en page, l'entrée correspondante dans le fichier XML doit se présenter comme suit :

```

<SText id="txt1">
  <text>&#x27;&#x3c;Hello&#x3e;&#x27;</text>
  <color>&#x27;blue&#x27;</color>
</SText>

```

Display preview

<Hello>

### Date et heure

Transmettez la date ou l'heure sous forme de chaînes simples, car les types de données spéciaux pour la date et l'heure ne fonctionnent pas toujours dans l'API.

### Caractères

Lorsque vous commandez des jeux de caractères, utilisez uniquement les noms de polices définis sur l'afficheur. Vous trouverez ces noms sur la page web 'Data - Advanced Input - Properties' :

Font	Aryana7Regular <span style="float: right;">▼</span>
Alignment	<ul style="list-style-type: none"> <li>Aryana14Bold</li> <li>Aryana14Condensed</li> <li style="background-color: #4a7ebb; color: white; padding: 2px;">Aryana14Regular</li> <li>Aryana30Regular</li> <li>Aryana7Bold</li> </ul>
Blink	Aryana7Regular
Blinkspeed	Metric14Regular
Scrollable	Metric7Regular

Les propriétés 'font.size' et 'font.spacingFactor', sélectionnables lors de l'exportation de la configuration PLC, ne sont pas encore fonctionnelles.

---

## 10 Exploiter plusieurs afficheurs sur un API

---

L'utilisation de plusieurs afficheurs sur un API est possible sans problème.

### **Toutes les afficheurs ont la même mise en page et doivent afficher les mêmes données.**

Dans ce cas, il suffit de suivre les étapes décrites dans le Quick Start pour un afficheur.

Si la communication avec une annonce fonctionne, exportez la mise en page de cet afficheur, importez-la sur les autres afficheurs et sélectionnez-la comme mise en page actuelle.

La dernière étape consiste à saisir les données de communication avec l'API sur les autres afficheurs. Comme tous les afficheurs interrogent le même API, les données du premier afficheur peuvent être reprises.

### **Les afficheurs doivent indiquer différentes données**

In diesem Fall führen Sie die Schritte aus dem Quick Start nacheinander für jede Anzeige aus.

Important : le fichier XML doit avoir un nom différent sur chaque afficheur. Ce n'est qu'alors que vous pourrez attribuer à chaque afficheur son propre bloc de données sur l'API.

Comme les données exportées sont toujours regroupées dans le fichier html.zip, il est utile de renommer ce fichier après l'exportation.

Toutes les données ainsi exportées doivent être copiées dans le répertoire que vous avez indiqué dans l'API lors de la configuration du serveur web.

Après l'importation des différents blocs de données dans l'API et la compilation du projet, l'API peut créer pour chaque afficheur un fichier XML séparé avec les données que vous avez définies et les afficheurs peuvent les lire et les afficher.

---

## 11 Données techniques

---

Vous trouverez toutes les données techniques, telles que les références, les dimensions, la puissance absorbée, les schémas du boîtier, la plage de température, etc. dans le document PSP XC50\_XC55.pdf intitulé "Série XC50/XC55 LED Matrix Displays" dans la zone de téléchargement du site Siebert : [www.siebert-group.com](http://www.siebert-group.com).