

Préface

Licence logicielle et mises à jour du produit

1

Présentation de SIMATIC Automation Tool

2

Prérequis et configuration de la communication

3

Opérations de l'outil logiciel

4

Enregistrer vos informations de la table Appareil

5

Menu, barre d'outils, raccourcis clavier et informations de référence

6

API SIMATIC Automation Tool pour architecture .NET

7

Prise en charge des appareils SIMATIC Automation Tool

8

Mentions légales

Signalétique d'avertissement

Ce manuel donne des consignes que vous devez respecter pour votre propre sécurité et pour éviter des dommages matériels. Les avertissements servant à votre sécurité personnelle sont accompagnés d'un triangle de danger, les avertissements concernant uniquement des dommages matériels sont dépourvus de ce triangle. Les avertissements sont représentés ci-après par ordre décroissant de niveau de risque.

DANGER

signifie que la non-application des mesures de sécurité appropriées **entraîne** la mort ou des blessures graves.

ATTENTION

signifie que la non-application des mesures de sécurité appropriées **peut entraîner** la mort ou des blessures graves.

PRUDENCE

signifie que la non-application des mesures de sécurité appropriées **peut entraîner** des blessures légères.

IMPORTANT

signifie que la non-application des mesures de sécurité appropriées peut entraîner un dommage matériel.

En présence de plusieurs niveaux de risque, c'est toujours l'avertissement correspondant au niveau le plus élevé qui est reproduit. Si un avertissement avec triangle de danger prévient des risques de dommages corporels, le même avertissement peut aussi contenir un avis de mise en garde contre des dommages matériels.

Personnes qualifiées

L'appareil/le système décrit dans cette documentation ne doit être manipulé que par du **personnel qualifié** pour chaque tâche spécifique. La documentation relative à cette tâche doit être observée, en particulier les consignes de sécurité et avertissements. Les personnes qualifiées sont, en raison de leur formation et de leur expérience, en mesure de reconnaître les risques liés au maniement de ce produit / système et de les éviter.

Utilisation des produits Siemens conforme à leur destination

Tenez compte des points suivants:

ATTENTION

Les produits Siemens ne doivent être utilisés que pour les cas d'application prévus dans le catalogue et dans la documentation technique correspondante. S'ils sont utilisés en liaison avec des produits et composants d'autres marques, ceux-ci doivent être recommandés ou agréés par Siemens. Le fonctionnement correct et sûr des produits suppose un transport, un entreposage, une mise en place, un montage, une mise en service, une utilisation et une maintenance dans les règles de l'art. Il faut respecter les conditions d'environnement admissibles ainsi que les indications dans les documentations afférentes.

Marques de fabrique

Toutes les désignations repérées par ® sont des marques déposées de Siemens AG. Les autres désignations dans ce document peuvent être des marques dont l'utilisation par des tiers à leurs propres fins peut enfreindre les droits de leurs propriétaires respectifs.

Exclusion de responsabilité

Nous avons vérifié la conformité du contenu du présent document avec le matériel et le logiciel qui y sont décrits. Ne pouvant toutefois exclure toute divergence, nous ne pouvons pas nous porter garants de la conformité intégrale. Si l'usage de ce manuel devait révéler des erreurs, nous en tiendrons compte et apporterons les corrections nécessaires dès la prochaine édition.

Préface

Notes relatives à la sécurité

Siemens commercialise des produits et solutions comprenant des fonctions de sécurité industrielle qui contribuent à une exploitation sûre des installations, solutions, machines, équipements et réseaux.

Pour garantir la sécurité des installations, systèmes, machines et réseaux contre les cybermenaces, il est nécessaire d'implémenter (et de préserver) un concept de sécurité industrielle global et moderne. Les produits et solutions de Siemens ne constituent qu'une partie d'un tel concept.

Il incombe au client d'empêcher tout accès non autorisé à ses installations, systèmes, machines et réseaux. Les systèmes, machines et composants doivent uniquement être connectés au réseau d'entreprise ou à Internet si et dans la mesure où c'est nécessaire et si des mesures de protection correspondantes (p. ex. utilisation de pare-feux et segmentation du réseau) ont été prises.

En outre, vous devez tenir compte des recommandations de Siemens concernant les mesures de protection correspondantes. Pour plus d'informations sur la sécurité industrielle, rendez-vous sur (<https://www.siemens.com/industrialsecurity>).

Les produits et solutions Siemens font l'objet de développements continus pour être encore plus sûrs. Siemens vous recommande donc vivement d'effectuer des actualisations dès que les mises à jour correspondantes sont disponibles et de ne toujours utiliser que les versions de produit actuelles. L'utilisation de versions obsolètes ou qui ne sont plus prises en charge peut augmenter le risque de cybermenaces.

Afin d'être informé des mises à jour produit dès qu'elles surviennent, abonnez-vous au flux RSS Siemens Industrial Security sous (<https://www.siemens.com/industrialsecurity>).

Service et assistance

En complément de ce guide utilisateur, Siemens propose son expertise technique sur Internet et via le site Web Siemens Automation (<https://www.siemens.com/automation>) et le site Web du support en ligne Siemens Industry (<https://support.industry.siemens.com/cs/ww/fr>) Adressez-vous à votre agence Siemens si certaines de vos questions techniques restent sans réponse, si vous voulez connaître les offres de formation ou si vous désirez commander des produits S7. Comme ce personnel est techniquement formé et a des connaissances très pointues sur vos activités, vos processus et vos industries, ainsi que sur les différents produits Siemens que vous utilisez, il peut apporter les réponses les plus rapides et les plus efficaces possibles à tout problème que vous pourriez rencontrer.

Langue source du document

La version anglaise du *Guide utilisateur SIMATIC Automation Tool* est la langue officielle (originale) pour les informations SIMATIC Automation Tool. Siemens identifie le manuel le manuel anglais comme étant la source officielle originale en cas de divergences entre les manuels traduits.

Sommaire

Préface	3
1 Licence logicielle et mises à jour du produit.....	11
1.1 Licence logicielle.....	11
1.2 Mise à jour logicielle de SIMATIC Automation Tool	13
2 Présentation de SIMATIC Automation Tool	15
2.1 Gestion des réseaux	15
2.2 Accès au réseau	19
2.3 Options de réseau.....	22
2.4 Fichier .NET API (interface d'application) .dll	24
3 Prérequis et configuration de la communication.....	25
3.1 Prise en charge du système d'exploitation du PG/PC, du logiciel VM et du logiciel de protection	25
3.2 Installation de SIMATIC Automation Tool.....	26
3.3 Démarrage de SIMATIC Automation Tool.....	27
3.4 Configuration requise.....	27
3.5 Configuration de la communication	30
4 Opérations de l'outil logiciel.....	37
4.1 Mots de passe de la CPU	37
4.2 Travailler avec la table Appareil et le journal des événements (Event Log).....	42
4.3 Options de traitement multifil (multi-thread).....	46
4.4 Scanner un réseau.....	49
4.5 Insérer un appareil	55
4.6 Mettre à jour les adresses IP, de sous-réseau et de passerelle	56
4.7 Mettre à jour les noms d'appareils PROFINET.....	58
4.8 Placement des CPU en mode RUN ou STOP (Marche/arrêt).....	60
4.9 Identifier les appareils	61
4.10 Mettre à jour les programmes des appareils depuis SIMATIC Automation Tool.....	63
4.11 Charger, ajouter, remplacer et supprimer des recettes dans les CPU	73
4.12 Charger et supprimer des journaux de données (Data logs) dans les CPU	76
4.13 Installation de nouveaux firmware dans les appareils	78
4.14 Sauvegarde et restauration de la CPU ou des données IHM.....	83

4.15	Restaurer la mémoire de la CPU	88
4.16	Restaurer les paramètres d'usine des CPU et des modules	89
4.17	Formater la carte mémoire.....	91
4.18	Récupérer les Service Data depuis des CPU.....	93
4.19	Réglage de l'heure sur les CPU.....	95
4.20	Afficher le tampon de diagnostic de la CPU	96
4.21	Ordre d'exécution des opérations	98
5	Enregistrer vos informations de la table Appareil.....	99
5.1	Enregistrer/Enregistrer sous - Table Appareil stockée au format sécurisé .sat	99
5.2	Importation/Exportation - Table Appareil chargée de/stockées dans un format .csv ouvert	100
6	Menu, barre d'outils, raccourcis clavier et informations de référence	103
6.1	Menu principal	103
6.1.1	Menu Fichier	103
6.1.2	Menu Edition	104
6.1.3	Menu Opérations.....	105
6.1.4	Menu Options.....	107
6.1.4.1	Menu Options.....	107
6.1.4.2	Noms de chemin de SIMATIC Automation Tool	107
6.1.4.3	Paramètres généraux	109
6.1.4.4	Paramètres de communication	110
6.1.4.5	Paramètres des projets	112
6.1.4.6	Paramètres de mise à jour du firmware	113
6.1.4.7	Paramètres de mise à jour des programmes	114
6.1.4.8	Paramètres des Service Data	115
6.1.4.9	Paramètres de sauvegarde/restauration	116
6.1.4.10	Paramètres des recettes	117
6.1.4.11	Paramètres des journaux de données (Data logs)	118
6.1.4.12	Paramètres de l'Event Log	119
6.1.4.13	Paramètres d'importation/exportation	120
6.1.5	Menu Outils	120
6.1.6	Menu Aide	121
6.2	Icônes de la barre d'outils	122
6.3	Raccourcis clavier	124
6.4	Définition du programme de sécurité	124
7	API SIMATIC Automation Tool pour architecture .NET	125
7.1	Introduction à l'API	125
7.2	Licence logicielle API et compatibilité des versions.....	125
7.3	Conception d'une application d'interface utilisateur pour des appareils de sécurité et des opérations relatives à la sécurité.....	126
7.3.1	Prise en charge d'opérations relatives à la sécurité et d'appareils de sécurité par l'API	126
7.3.2	Directives de programmation de l'interface utilisateur pour les opérations relatives à la sécurité.....	127

7.3.3	Utilisez un code couleurs pour les champs de sécurité dans votre interface utilisateur	131
7.3.3.1	Appliquer une couleur à une icône d'appareil CPU	132
7.3.3.2	Appliquer une couleur aux données d'un appareil.....	133
7.3.3.3	Appliquer une couleur au mot de passe de la CPU.....	134
7.3.3.4	Appliquer une couleur à un dossier de programme.....	135
7.3.3.5	Appliquer une couleur à un mot de passe de programme.....	136
7.3.4	Codes de Hamming	137
7.4	Présentation de l'architecture	137
7.5	Référencement de l'API dans une application d'interface utilisateur.....	140
7.6	Classes de base	141
7.6.1	Classe EncryptedString	141
7.6.2	Classe Result.....	143
7.6.3	Classe de diagnostic.....	145
7.6.4	Classe DataChangedEventArgs	146
7.6.5	Classe ProgressChangedEventArgs	146
7.7	Interfaces de support communes	147
7.7.1	Interface IRemoteFile.....	147
7.7.2	Interface IRemoteFolder	147
7.7.3	Interface IRemoteInterface	148
7.7.4	Interface IHardware	148
7.7.5	Interface IModule	149
7.7.6	Interface IBaseDevice.....	150
7.7.7	Interface IHardwareCollection	150
7.7.8	Interface IModuleCollection	150
7.8	Classe Network.....	151
7.8.1	Constructeur réseau	151
7.8.2	Méthode QueryNetworkInterfaceCards	151
7.8.3	Méthode SetCurrentNetworkInterface	152
7.8.4	Propriété CurrentNetworkInterface	153
7.8.5	Méthode ScanNetworkDevices.....	153
7.8.6	Méthode SetCommunicationsTimeout.....	154
7.8.7	Méthode GetCommunicationsTimeout	155
7.8.8	Méthode CheckValidLicense	156
7.8.9	Méthode GetEmptyCollection	156
7.9	Classe IProfinetDeviceCollection.....	157
7.9.1	Itération des éléments dans la collection.....	157
7.9.1.1	Itération des éléments dans la collection.....	157
7.9.1.2	Méthode GetEnumerator	158
7.9.1.3	Propriété décompte.....	158
7.9.1.4	Propriété []	158
7.9.2	Filtrage des éléments dans la collection.....	159
7.9.2.1	Éléments de la collection	159
7.9.2.2	Méthode FilterByDeviceFamily	159
7.9.2.3	Méthode FilterOnlyCPUs	160
7.9.3	Rechercher un appareil spécifique dans la collection	161
7.9.3.1	Méthode FinidDeviceByIP	161
7.9.3.2	Méthode FinidDeviceByMAC	162
7.9.4	Sérialisation	163
7.9.4.1	Transfert d'une collection de/vers un fichier de données externe	163

7.9.4.2	Méthode WriteToStream.....	163
7.9.4.3	Méthode ReadFromStream	164
7.9.5	Ajouter manuellement des éléments à la collection.....	165
7.9.5.1	Méthode InsertDeviceByIP	165
7.9.5.2	Méthode InsertDeviceByMAC	166
7.9.6	Copie de données depuis une collection	167
7.9.6.1	Méthode CopyUserData	167
7.9.7	Supprimer des appareils de la collection	168
7.9.7.1	Méthode Clear.....	168
7.9.7.2	Méthode Remove.....	168
7.10	Interface IProfinetDevice.....	169
7.10.1	Propriétés IProfinetDevice	169
7.10.2	Méthode IProfinetDevice.....	173
7.10.2.1	Méthode RefreshStatus	173
7.10.2.2	Méthode FirmwareUpdate	174
7.10.2.3	Méthode Identify.....	176
7.10.2.4	Méthode Reset.....	177
7.10.2.5	Méthode SetIP	178
7.10.2.6	Méthode SetProfinetName.....	179
7.10.2.7	Méthode ValidateIPAddressSubnet.....	180
7.10.2.8	Méthode ValidatePROFINETName	181
7.10.3	Événements IProfinetDevice.....	182
7.10.3.1	Événement DataChanged.....	182
7.10.3.2	Événement ProgressChanged.....	183
7.11	Classe IModuleCollection et propriétés des modules.....	185
7.11.1	Propriété de modules et classe IModuleCollection.....	185
7.11.2	Interface IModule	186
7.12	Interface ICPU.....	187
7.12.1	Identification de l'appareil CPU dans un IProfinetDeviceCollection.....	187
7.12.2	Propriétés ICPU	188
7.12.3	Drapeaux ICPU	189
7.12.3.1	Drapeaux relatifs aux mises à jour de programmes	189
7.12.3.2	Restaurer les drapeaux.....	191
7.12.3.3	Drapeaux de fonctionnalités	192
7.12.4	Méthodes ICPU.....	193
7.12.4.1	ICPU protégées et mots de passe	193
7.12.4.2	Méthode SetPassword	193
7.12.4.3	Méthode SetProgramFolder.....	194
7.12.4.4	Méthode SetProgramPassword	196
7.12.4.5	Méthode ProgramUpdate.....	197
7.12.4.6	Méthode SetBackupFile	199
7.12.4.7	Méthode SetBackupFilePassword.....	202
7.12.4.8	Méthode Restore (interface ICPU).....	203
7.12.4.9	Méthode Backup (interface ICPU)	204
7.12.4.10	Méthode DownloadRecipe	205
7.12.4.11	Méthode DeleteDataLog	206
7.12.4.12	Méthode DeleteRecipe	208
7.12.4.13	Méthode GetCurrentDateTime.....	210
7.12.4.14	Méthode GetDiagnosticsBuffer	211
7.12.4.15	Méthode MemoryReset.....	212
7.12.4.16	Méthode ResetToFactoryDefaults	213

7.12.4.17	Méthode SetOperatingState	215
7.12.4.18	Méthode SetCurrentDateTime	216
7.12.4.19	Méthode UploadDataLog	217
7.12.4.20	Méthode UploadRecipe	219
7.12.4.21	Méthode UploadServiceData	221
7.12.4.22	Méthode FormatMemoryCard	222
7.12.4.23	DetermineConfirmationMessage	224
7.12.5	Propriétés RemoteInterfaces	226
7.12.5.1	Modules d'E/S décentralisés	226
7.12.5.2	Propriétés IRemoteInterfaces	226
7.13	Interface IHMI	230
7.13.1	Interface IHMI	230
7.13.2	Propriétés et drapeaux IHMI	231
7.13.2.1	Propriétés IHMI	231
7.13.2.2	Drapeaux relatifs aux mises à jour de programmes	231
7.13.2.3	Restaurer les drapeaux	232
7.13.2.4	Drapeaux de fonctionnalités	232
7.13.3	Méthode Backup (interface IHMI)	233
7.13.4	Méthode ProgramUpdate (interface IHMI)	234
7.13.5	Méthode Restore (interface IHMI)	236
7.13.6	Méthode SetProgramFolder	237
7.13.7	Méthode SetBackupFile	238
7.14	Prise en charge des classes	240
7.14.1	Diagnosticsitem	240
7.15	Exceptions	241
7.15.1	CriticalInternalErrorException	241
7.16	Enumérations API	242
7.16.1	DataChangedType	242
7.16.2	DeviceFamily	242
7.16.3	ConfirmationType	242
7.16.4	ErrorCode	243
7.16.5	Language	246
7.16.6	OperatingState	247
7.16.7	OperatingStateREQ	247
7.16.8	ProgressAction	248
7.16.9	RemoteInterfaceType	248
7.16.10	FeatureSupport	249
7.16.11	ProtectionLevel	250
7.16.12	ConfirmationType	250
7.16.13	FailsafeOperation	250
7.16.14	RemoteFolderType	250
7.17	Exemple de réseau	251

8	Prise en charge des appareils SIMATIC Automation Tool.....	255
8.1	Versions de firmware et appareils non reconnus.....	255
8.2	ET 200.....	256
8.2.1	ET 200AL	256
8.2.1.1	Prise en charge ET 200AL IM.....	256
8.2.1.2	Prise en charge ET 200AL SM et IO-Link.....	256
8.2.2	Prise en charge de ET 200eco	257
8.2.3	Prise en charge ET 200M IM	258
8.2.4	Prise en charge ET 200MP IM.....	259
8.2.5	ET 200S	260
8.2.6	ET 200pro	261
8.2.6.1	Prise en charge de la CPU ET 200pro (basée sur S7-1516)	261
8.2.6.2	Prise en charge ET 200pro IM	262
8.2.6.3	ET 200pro IO-Link, prise en charge RFID	262
8.2.7	ET 200SP	263
8.2.7.1	Prise en charge de la CPU ET 200SP (basée sur S7-151x)	263
8.2.7.2	Prise en charge ET 200SP IM et module serveur	264
8.2.7.3	Prise en charge ET 200SP SM, AS-i, CM, CP, TM, IO-Link, départs-moteurs	265
8.3	S7-1200.....	267
8.3.1	Prise en charge de la CPU S7-1200.....	267
8.3.2	Prise en charge de S7-1200 I/O et autres modules.....	268
8.4	S7-1500.....	272
8.4.1	Prise en charge de la CPU S7-1500.....	272
8.4.2	Prise en charge de S7-1500 I/O et autres modules.....	273
8.5	SIMATIC HMI (interface homme-machine).....	274
8.5.1	Prise en charge des pupitres HMI Basic.....	274
8.5.2	Prise en charge des pupitres HMI Comfort.....	275
8.5.3	Prise en charge des pupitres HMI Mobile	276
8.6	SITOP (Alimentations)	277
8.6.1	Prise en charge de SITOP (alimentation)	277
8.7	RFID et MOBY (modules de communication).....	278
8.7.1	RFID (identification par radiofréquence)	278
8.7.2	MOBY (interface DeviceNet).....	278
Index	279

Licence logicielle et mises à jour du produit

1.1 Licence logicielle

Exigences en matière de licence

Une licence logicielle est requise pour bénéficier de l'intégralité des fonctions de SIMATIC Automation Tool.

Fonctionnement sans licence

En l'absence de licence, SIMATIC Automation Tool applique les restrictions suivantes :

- Vous ne pouvez effectuer qu'une seule opération sur un seul appareil à la fois.
- Vous ne pouvez pas utiliser l'API (interface de programmation d'application) pour la programmation d'application personnalisée.
- Vous ne pouvez actualiser qu'un appareil à la fois.

Comment obtenir une licence

Vous pouvez acheter une licence en ligne via un compte de messagerie sur Siemens Mall et l'installer avec le logiciel Automation License Manager (ALM).

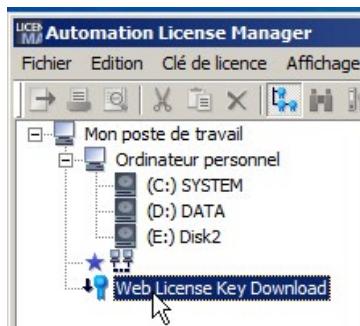
Comment installer une licence

S'il ne l'est pas déjà, le gestionnaire ALM est installé/mis à jour pendant l'installation de SIMATIC Automation Tool.

1. Commandez le produit sur Siemens Mall.
2. Vous recevez par courriel un "numéro d'avis de livraison" (Delivery Note No.) et un mot de passe temporaire.
3. Démarrez l'application ALM (Automation License Manager) sur votre PG/PC :



4. Double-cliquez sur "Web License Key Download" dans le volet de navigation. Patientez jusqu'à ce que la page SIEMENS Online Software Delivery s'affiche dans la fenêtre ALM.



1.1 Licence logicielle

5. Sur la page SIEMENS Online Software Delivery, saisissez les données suivantes :
 - Numéro d'avis de livraison (Delivery Note No.)
 - Mot de passe
 - Code d'accès (voir l'image)

Connexion avec la clé de livraison 

no de bon de livraison:

mot de passe:

[> Mot de passe oublié/expiré ?](#)

Entrez vos codes d'accès.: 

[> Demander un nouveau code d'accès](#)

Connexion

6. Téléchargez le logiciel à l'aide du bouton Disponible.
7. Faites glisser le rectangle de la licence vers le disque dur. Déposez le rectangle de la licence à un emplacement sur l'ordinateur indiqué à droite sur l'arborescence du gestionnaire de licences ALM.

1.2 Mise à jour logicielle de SIMATIC Automation Tool

Obtenir automatiquement les mises à jour logicielles

Le logiciel de mise à jour TIA Software Updater est installé au cours de l'installation de SIMATIC Automation Tool. Si votre PG/PC est connecté à Internet, vous pouvez télécharger les mises à jour logicielles de SIMATIC Automation Tool directement depuis le site Internet de SIEMENS.

SIMATIC Automation Tool vous permet de rechercher automatiquement les mises à jour disponibles pour les produits logiciels SIMATIC installés. Si des mises à jour sont disponibles, vous pouvez les installer sur votre PG/PC.

Présentation de SIMATIC Automation Tool

2.1 Gestion des réseaux

Gestion de réseaux d'appareils SIMATIC

Une fois qu'un programme de commande est créé, vérifié et téléchargé avec le Siemens TIA Portal, SIMATIC Automation Tool peut être utilisé pour configurer, piloter, effectuer la maintenance et documenter les réseaux d'automatisation.

Remarque

Vous ne pouvez pas utiliser SIMATIC Automation Tool pour lire des appareils dans votre réseau d'automatisation si S7-PLCSIM est en cours d'exécution sur votre PG/PC (station de programmation/ordinateur personnel) en même temps que SIMATIC Automation Tool. Vous devez fermer S7-PLCIM pour que SIMATIC Automation Tool puisse lire un réseau d'appareil.

Opérations de configuration, de pilotage et de maintenance réalisables à l'aide d'Automation Tool

SIMATIC Automation Tool vous permet de réaliser de nombreuses opérations utiles sur un réseau d'appareils :

Opérations réseau

- Scruter le réseau et créer une table qui indique les appareils accessibles sur le réseau. (Page 49) La table Appareil inclut les CPU, les modules, les IHM et les autres appareils Siemens. Vous pouvez enregistrer la table Appareil (Page 99) dans un fichier projet de sécurité *.sat ou dans un fichier open text .csv.
- Identifier les appareils avec des LED / écrans IHM clignotants (Page 61)
- Mettre à jour les adresses des appareils (IP, sous-réseau, passerelle) (Page 56)
- Mettre à jour le nom PROFINET (nom de la station) d'un appareil (Page 58)

Configuration système

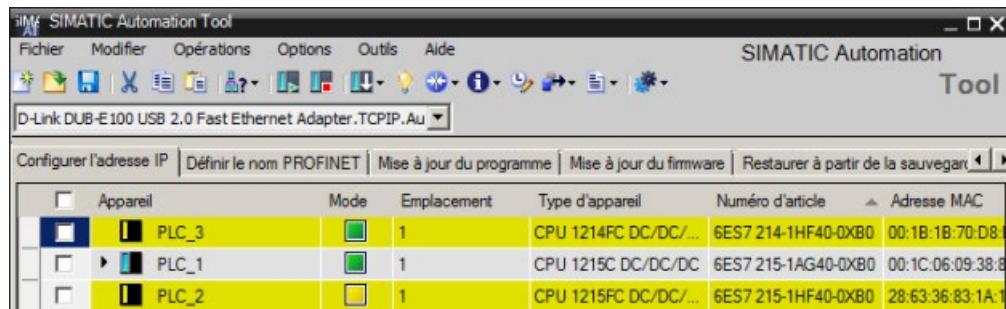
- Régler la date et l'heure d'une CPU sur celles de votre PG/PC (programmateur/ordinateur personnel) (Page 95)
- Mettre à jour un programme CPU ou le système d'exploitation et le logiciel exécutif d'un appareil IHM (Page 63)
- Charger, ajouter, remplacer ou supprimer des données de recette d'une CPU (Page 73)
- Charger ou supprimer les données de Data log d'une CPU (Page 76)
- Mettre à jour le firmware des modules (Page 78)

Opération

- Placer une CPU en mode RUN ou STOP (Page 60)

Diagnostic système et maintenance

- Sauvegarder/restaurer des données vers/depuis un fichier de sauvegarde pour une CPU ou un appareil HMI (Page 83)
- Afficher le tampon de diagnostic d'une CPU (Page 96)
- Réinitialiser la mémoire de la CPU (Page 88)
- Récupérer les Service Data depuis une CPU (Page 93)
- Formater la carte mémoire SIMATIC installée dans une CPU (Page 91)
- Restaurer les paramètres d'usine des appareils (Page 89)
- Documenter et sauvegarder les informations réseau dans un fichier texte standard .cvs ou un dans un fichier .sat protégé par mot de passe. (Page 256)



The screenshot shows the SIMATIC Automation Tool software window. The title bar reads "SIMATIC Automation Tool". The menu bar includes "Fichier", "Modifier", "Opérations", "Options", "Outils", and "Aide". The toolbar contains various icons for file operations and tools. The main area shows a table titled "Table Appareil SIMATIC Automation Tool". The table has columns: "Appareil", "Mode", "Emplacement", "Type d'appareil", "Numéro d'article", and "Adresse MAC". There are three entries in the table:

Appareil	Mode	Emplacement	Type d'appareil	Numéro d'article	Adresse MAC
PLC_3	1	CPU 1214FC DC/DC/...	6ES7 214-1HF40-0XB0	00:1B:1B:70:D8:00	
PLC_1	1	CPU 1215C DC/DC/DC	6ES7 215-1AG40-0XB0	00:1C:06:09:38:80	
PLC_2	1	CPU 1215FC DC/DC/...	6ES7 215-1HF40-0XB0	28:63:36:83:1A:10	

Figure 2-1 Table Appareil SIMATIC Automation Tool

Prise en charge d'appareils en réseau

SIMATIC Automation Tool prend en charge les types d'appareils Siemens suivants :

- CPU standard
- CPU de sécurité
- IHM (HMI)
- Appareils PROFINET

Les opérations DCP (Discovery and Configuration Protocol) peuvent fonctionner avec les appareils Siemens PROFINET pris en charge, les appareils Siemens PROFINET non pris en charge et les appareils PROFINET ou Ethernet de tiers non pris en charge.

Liste des opérations DCP basées sur l'adresse MAC :

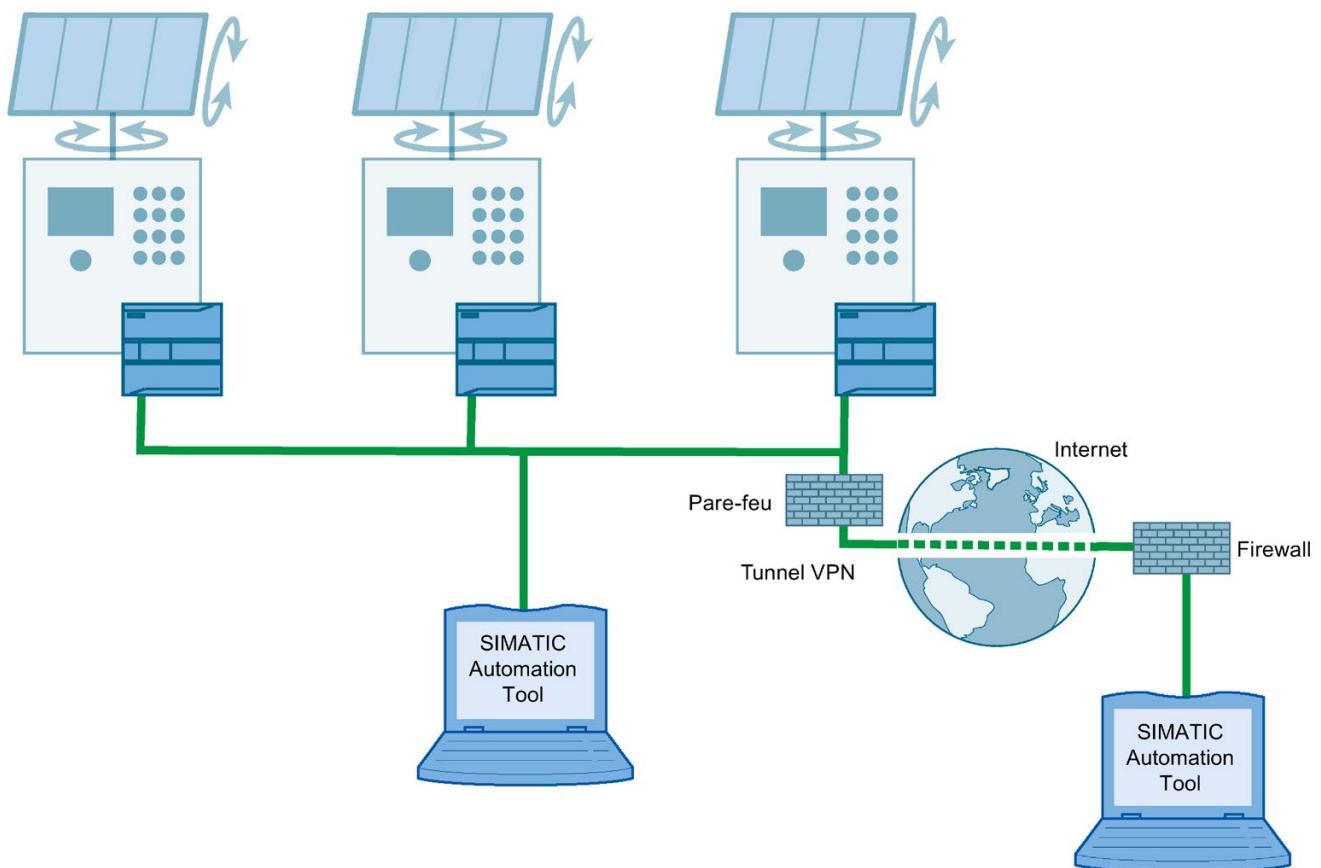
- Scanner un réseau
- Identifier les appareils (LED / écrans IHM clignotants)
- Mettre à jour l'adresse IP, le masque de sous-réseau et l'adresse de la passerelle sur les appareils
- Mettre à jour le nom d'appareil PROFINET

Pour plus d'informations sur les appareils Siemens pris en charge et les opérations, voir Prise en charge des appareils SIMATIC Automation Tool (Page 255)

Exemple d'installation d'un réseau S7-1200

Fabrication /// Panel shop et chargement initial du programme

- Les CPU utilisent toutes la même configuration matérielle et le même programme de commande.
- La configuration de la CPU PROFINET utilise les options "Set IP address on the device" (configurer l'adresse IP sur l'appareil) et "Set PROFINET device name on the device option" (définir le nom d'appareil PROFINET sur l'option de l'appareil).
- Les panels sont tous identiques.
- Vous pouvez utiliser SIMATIC Automation Tool, une carte mémoire SD SIMATIC ou TIA Portal pour charger les appareils CPU / IHM avec les données de projet.



Si vous gérez un réseau contenant de nombreux appareils, SIMATIC Automation Tool peut simplifier les opérations et réduire le temps nécessaire. Vous pouvez effectuer une opération sur plusieurs appareils en même temps. SIMATIC Automation Tool utilise le traitement multifil (multi-threading), ce qui signifie que de nombreuses opérations peuvent être exécutées simultanément. Lorsque que le PG / PC effectue une opération sur un seul appareil, un autre fil (thread) peut commencer l'opération sur un autre appareil. Le traitement multi-thread est paramétré par défaut dans le Paramètres de communication (Page 110). Vous pouvez désactiver le traitement multi-thread si vous le souhaitez.

Voir aussi

- Prise en charge ET 200AL SM et IO-Link (Page 256)
- Prise en charge de ET 200eco (Page 257)
- Prise en charge ET 200M IM (Page 258)
- Prise en charge ET 200MP IM (Page 259)
- ET 200S (Page 260)
- Prise en charge ET 200pro IM (Page 262)
- ET 200pro IO-Link, prise en charge RFID (Page 262)
- Prise en charge de la CPU ET 200SP (basée sur S7-151x) (Page 263)
- Prise en charge ET 200SP IM et module serveur (Page 264)
- Prise en charge ET 200SP SM, AS-i, CM, CP, TM, IO-Link, départs-moteurs (Page 265)
- Prise en charge de la CPU S7-1200 (Page 267)
- Prise en charge de S7-1200 I/O et autres modules (Page 268)
- Prise en charge de la CPU S7-1500 (Page 272)
- Prise en charge de S7-1500 I/O et autres modules (Page 273)
- Prise en charge des pupitres HMI Basic (Page 274)
- Prise en charge des pupitres HMI Comfort (Page 275)
- Prise en charge des pupitres HMI Mobile (Page 276)
- Prise en charge de SITOP (alimentation) (Page 277)
- RFID (identification par radiofréquence) (Page 278)
- MOBY (interface DeviceNet) (Page 278)

2.2 Accès au réseau

Communiquer avec les appareils réseau

Le scan réseau SIMATIC Automation Tool détecte les appareils raccordés directement à un réseau à l'aide de l'adresse MAC (Media Access Control). Chaque appareil possède une adresse MAC unique gravée qui ne peut pas être modifiée. Les appareils connectés sont détectés, qu'ils soient configurés avec une adresse IP, de sous-réseau ou de passerelle ou qu'ils ne soient pas configurés (les adresses sont égales à 0.0.0.0).

Les opérations basées sur l'adresse MAC utilisent le DCP (Discovery and Configuration Protocol). Le DCP est un standard Ethernet. SIMATIC Automation Tool peut utiliser les opérations DCP basées sur l'adresse MAC "Scanner le réseau complet", "Identifier", "Mettre à jour l'adresse IP" et "Mettre à jour le nom d'appareil PROFINET" pour tous les appareils réseau directement connectés (appareils PROFINET pris en charge, appareils PROFINET non pris en charge, et appareils Ethernet de tiers non pris en charge).

Les opérations basées sur l'adresse IP ne fonctionnent que pour les appareils Siemens pris en charge.

Réseau simple avec une CPU et périphériques locaux :

- Possibilité d'effectuer des opérations basées sur l'adresse MAC
- La CPU doit présenter une adresse IP et une configuration d'appareil TIA Portal valides afin de permettre au scan réseau d'afficher les périphériques locaux de la CPU et d'activer les opérations basées sur l'adresse IP.

CPU multiples avec E/S locales connectées via un commutateur Ethernet :

- Possibilité d'effectuer des opérations basées sur l'adresse MAC pour tous les appareils
- La CPU doit présenter une adresse IP valide et une configuration de l'appareil de TIA Portal afin de permettre au scan réseau d'afficher les périphériques locaux de la CPU et d'activer les opérations basées sur l'adresse IP.

CPU en réseaux complexes avec d'E/S décentralisées et routeurs d'adresses IP :

- Les opérations basées sur l'adresse MAC ne sont possibles que pour les appareils directement connectés au même sous-réseau que le PG/PC qui exécute SIMATIC Automation Tool.
- Les CPU doivent présenter une adresse IP et une configuration d'appareil TIA Portal valides afin de permettre au scan réseau d'afficher les appareils situés derrière les CPU et d'activer les opérations basées sur l'adresse IP.

Par exemple, une CPU S7-1500 possède une connexion réseau PROFINET à SIMATIC Automation Tool et utilise un module CP local pour se raccorder à un autre réseau PROFINET sur lequel des périphériques I/O décentralisés sont connectés. Vous devez affecter une adresse IP valide à la CPU S7-1500 et compiler/télécharger la configuration d'appareil TIA Portal de votre projet pour que les modules d'E/S décentralisés soient visibles sur la table Appareil SIMATIC Automation Tool.

Notez que, pour les appareils raccordés indirectement derrière une CPU, un module CM ou CP, seule la mise à jour du firmware est possible.

Le type d'accès réseau dépend de la commande exécutée, comme indiqué sur la table suivante.

Commande SIMATIC Automation Tool	Adresse d'appareil utilisée	Mot de passe obligatoire pour une CPU protégée	Connectivité du PG/PC et de l'appareil
Scanner (détecter des CPU, des IHM, des E/S et d'autres appareils)	MAC	Non	
Identifier les appareils (LED / écrans IHM clignotants)	MAC	Non	
Configurer l'adresse IP, le masque de sous-réseau et l'adresse de la passerelle sur les appareils	MAC	Non	
Définir le nom PROFINET sur les appareils	MAC	Non	
Restaurer les paramètres d'usine des appareils (périphériques I/O PROFINET uniquement)	MAC	Sans objet	
Mettre les CPU en mode RUN ou STOP (Marche/arrêt)	IP	Oui ¹	<ul style="list-style-type: none"> Réseau local : vous pouvez accéder aux appareils du réseau à l'aide des commutateurs Ethernet, mais vous ne pouvez pas accéder aux appareils d'un autre réseau via un routeur d'adresse IP.
Régler l'heure de la CPU sur l'heure PG/PC	IP	Oui ¹	<ul style="list-style-type: none"> Réseau local : vous pouvez accéder aux appareils du réseau à l'aide des commutateurs Ethernet.
Mettre à jour le programme : Programmes de la CPU	IP	Oui ³	<ul style="list-style-type: none"> Réseau distant : vous pouvez accéder aux appareils d'un autre réseau via un routeur d'adresse IP.
Mettre à jour le programme : Mise à jour du système d'exploitation et du logiciel exécutif d'un appareil IHM	IP	Non	<ul style="list-style-type: none"> Connexion VPN au réseau local
Charger des données Recette d'une CPU	IP	Oui ²	
Ajouter, remplacer ou supprimer des données Recette d'une CPU	IP	Oui ¹	
Charger les données de Data log d'une CPU	IP	Oui ²	
Supprimer les données de Data log d'une CPU	IP	Oui ¹	
Réaliser une copie de sauvegarde de la CPU dans un fichier	IP	Oui ²	
Réaliser une copie de sauvegarde de l'IHM dans un fichier	IP	Non	
Restaurer la CPU à partir d'un fichier de sauvegarde	IP	Oui ³	
Restaurer l'IHM à partir d'un fichier de sauvegarde	IP	Non	
Charger les Service data depuis des CPU	IP	Oui ²	
Lire le tampon de diagnostic de la CPU	IP	Oui ²	
Restaurer la mémoire de la CPU	IP	Oui ³	

Commande SIMATIC Automation Tool	Adresse d'appareil utilisée	Mot de passe obligatoire pour une CPU protégée	Connectivité du PG/PC et de l'appareil
Restaurer les paramètres d'usine des appareils	IP	Oui ³	
Mettre à jour le firmware dans des appareils	IP	Oui ³	
Formater la carte mémoire	IP	Oui ³	

¹ nécessite le niveau d'accès "Accès complet (pas de protection)" pour toutes les CPU.

² nécessite le niveau d'accès "Accès en lecture" pour toutes les CPU.

³ Appareils IHM : aucun mot de passe n'est nécessaire.

CPU standard : nécessite le niveau d'accès "Accès complet (pas de protection)".

CPU de sécurité : Nécessite le niveau d'accès "Accès complet (pas de protection)" ou l'"Accès complet, y compris fail-safe (pas de protection)" selon la version de firmware de la CPU. Vous devez également resélectionner et confirmer les appareils pour l'opération.

Remarque

Protocoles pour sous-réseaux IP et interface réseau

Le PG/PC qui exécute SIMATIC Automation Tool et les appareils raccordés au réseau local doivent utiliser des affectations de sous-réseau IP correctes.

Selon le type de protocole d'interface réseau que vous sélectionnez ("TCP/IP" ou "TCP/IP.Auto"), SIMATIC Automation Tool sera ou non capable de détecter des appareils durant le scan du réseau.

Voir l'exemple sous l'rubrique Réglage de la communication (Page 30).

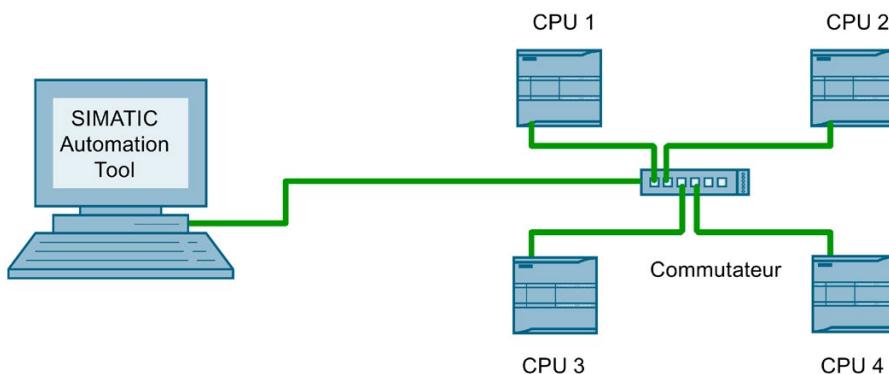
2.3 Options de réseau

Réseaux locaux et réseaux distants

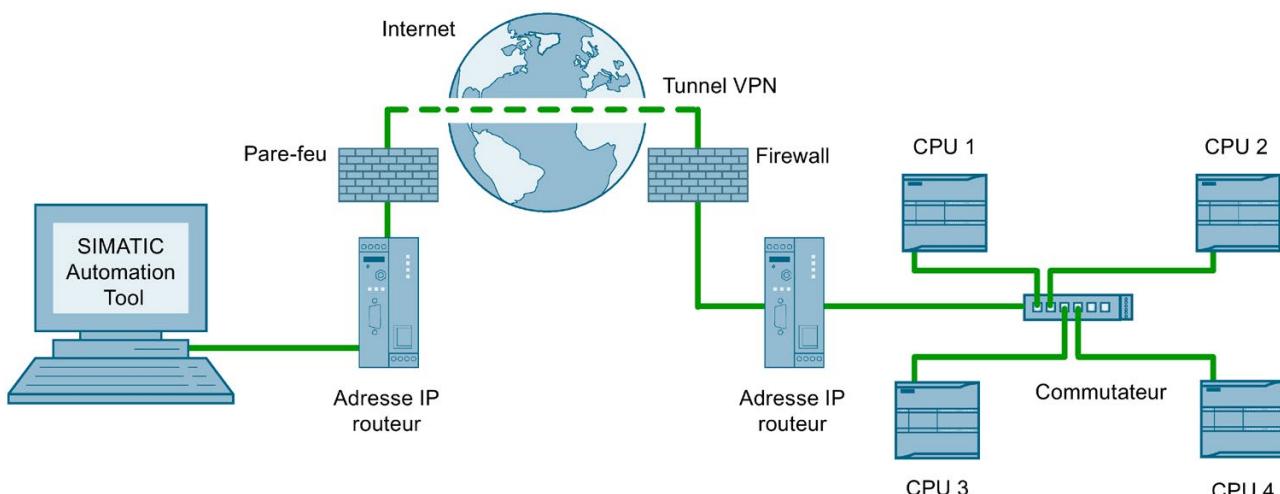
Les exemples suivants montrent des réseaux locaux des réseaux distants utilisables par SIMATIC Automation Tool. Ces diagrammes simplifiés montrent la connectivité de base et n'affichent pas les appareils IHM, les périphériques d'E/S locaux et décentralisés (PROFINET et PROFIBUS) et d'autres appareils qui sont également accessibles. D'autres environnements réseau sont aussi possibles.

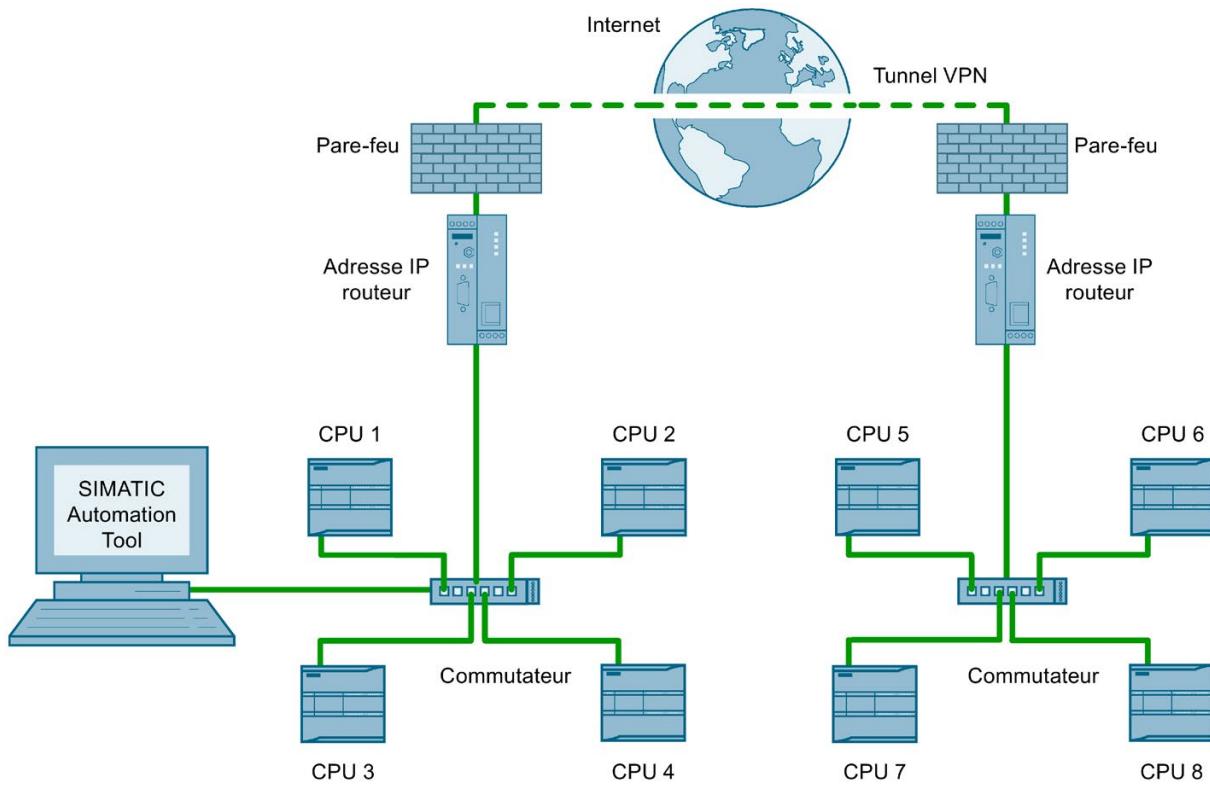
Vous pouvez remplir la table Appareil SIMATIC Automation Tool en scannant un réseau (Page 49). Les données appareil dans une table Appareil existante peuvent être exportées au format .csv. Vous pouvez modifier les données appareil existantes dans le texte .csv et utiliser la commande d'importation pour intégrer les nouvelles données dans la table Appareil SIMATIC Automation Tool.

Exemple 1 : Réseau local S7-1200



Exemple 2 : Réseau distant S7-1200



Exemple 3 : Réseaux combinés S7-1200**Voir aussi**

Importation/Exportation - Table Appareil chargée de/stockées dans un format .csv ouvert
(Page 100)

2.4 Fichier .NET API (interface d'application) .dll

Vous pouvez créer votre propre outil logiciel qui utilise l'API Microsoft .NET de SIMATIC Automation Tool (Page 125) pour effectuer les mêmes opérations que SIMATIC Automation Tool.

SIMATIC Automation Tool doit être installé sur tous les PG/PC qui utilisent cette API. SIMATIC Automation Tool et votre outil logiciel utilisent le fichier API .dll et des fichiers de communication S7 supplémentaires.

Les fichiers dont vous avez besoin sont installés avec SIMATIC Automation Tool.

Les fichiers API se trouvent dans le dossier dans lequel est installé SIMATIC Automation Tool.

Licence logicielle requise pour V3.0 et versions ultérieures

L'API est désactivée en mode sans licence sous V3.0 ou une version ultérieure.

Si une licence valide est détectée pour SIMATIC Automation Tool, l'API est activée sous la version V3.0 ou une version ultérieure.

Prérequis et configuration de la communication

3.1 Prise en charge du système d'exploitation du PG/PC, du logiciel VM et du logiciel de protection

Prise en charge des systèmes d'exploitation Microsoft Windows 64 bits

SIMATIC Automation Tool fonctionne avec les systèmes d'exploitation 64 bits suivants :

- Windows 7 Home Premium SP1
- Windows 7 Professionnel SP1
- Windows 7 Enterprise SP1
- Windows 7 Ultimate SP1
- Windows 10 Home Version 1607 (OS Build 14393)
- Windows 10 Pro Version 1607 (OS Build 14393)
- Windows 10 Enterprise Version 1607 (OS Build 14393)
- Windows 10 Enterprise 2016 LTSB (OS Build 14393)
- Windows 10 IoT Enterprise 2015 LTSB (OS Build 10240)

Vous pouvez installer SIMATIC Automation Tool et utiliser la version sans licence pour tester le bon fonctionnement d'autres systèmes d'exploitation Windows 64 bits. SIMATIC Automation Tool peut être installé et fonctionner correctement avec ces systèmes d'exploitation. Siemens ne garantit pas le fonctionnement de SIMATIC Automation Tool avec les autres systèmes d'exploitation Windows 64 bits, pour lesquels aucune assistance technique ne sera fournie.

Prise en charge des logiciels de machine virtuelle

SIMATIC Automation Tool fonctionne avec les logiciels de machine virtuelle (MV) suivants :

- VMware Workstation 12.5
- VMware Player 12.5

Prise en charge des logiciels de protection et antivirus

SIMATIC Automation Tool fonctionne avec les logiciels de protection et antivirus suivants :

- Symantec Endpoint Protection 14
- McAfee VirusScan Enterprise 8.8
- Trend Micro Office Scan Corporate Edition 12.0
- Kaspersky Anti-Virus 2017
- Windows Defender (en tant qu'élément constitutif du système d'exploitation Windows)
- Qihoo "360 Total Security Essential" 8.8 (pour le marché chinois)
- McAfee Application Control 7.0.1
- Windows Bitlocker (en tant qu'élément constitutif du système d'exploitation Windows)

SIMATIC Automation Tool peut être installé et fonctionner correctement avec d'autres logiciels de protection et antivirus. Siemens ne garantit pas le fonctionnement de SIMATIC Automation Tool avec les autres logiciels de protection et antivirus, pour lesquels aucune assistance technique ne sera fournie.

3.2 Installation de SIMATIC Automation Tool

Enregistrez le travail en cours et fermez toutes les applications PG/PC avant d'installer SIMATIC Automation Tool.

Règles d'installation

Une seule version de SIMATIC Automation Tool peut être installée sur un PG/PC. Si vous avez installé une version antérieure, vous devez tout d'abord la désinstaller. Le fichier exécutable d'installation vérifie qu'aucune version antérieure n'est installée et répond comme suit :

- Si aucune version de SIMATIC Automation Tool n'a été identifiée, le programme d'installation se poursuit.
- Si une version antérieure à V3.0 de SIMATIC Automation Tool est identifiée, le programme d'installation vous informe que vous devez désinstaller l'ancienne version. Le programme d'installation vous guide tout au long de la désinstallation de l'ancienne version. Pour poursuivre l'installation, vous devez fermer et redémarrer l'installation.
- Si la version V3.0 est identifiée, le processus d'installation désinstalle la version V3.0 et installe V3.1.
- Si une version actuelle V3.1 de SIMATIC Automation Tool a été identifiée, le programme d'installation propose les options Modifier/mettre à niveau, Réparer ou Désinstaller l'installation précédente.

Remarque

Vous pouvez installer SIMATIC Automation Tool sur n'importe quel appareil qui dispose de l'espace requis comme indiqué par le programme d'installation. Vous devez cependant avoir au moins 1,4 Go libres sur le lecteur C: \\ pour les fichiers système.

3.3 Démarrage de SIMATIC Automation Tool

Utilisez l'une des méthodes suivantes pour démarrer SIMATIC Automation Tool :

- Double-cliquer sur l'icône SIMATIC Automation Tool du Bureau.
- Utiliser le bouton Windows Démarrer :
 - Cliquer sur le bouton Windows Démarrer puis sur "Tous les programmes".
 - Cliquer sur le dossier "Siemens Automation", puis sur le dossier "SIMATIC Automation Tool", puis sur "SIMATIC Automation Tool".

3.4 Configuration requise

Pour permettre la définition de l'adresse IP ou du nom PROFINET d'un appareil avec SIMATIC Automation Tool, le projet TIA Portal de l'appareil doit activer ces actions dans la configuration de l'appareil du projet. Compilez le projet et téléchargez-le sur l'appareil cible avant d'essayer de modifier l'adresse IP ou le nom PROFINET d'un appareil.

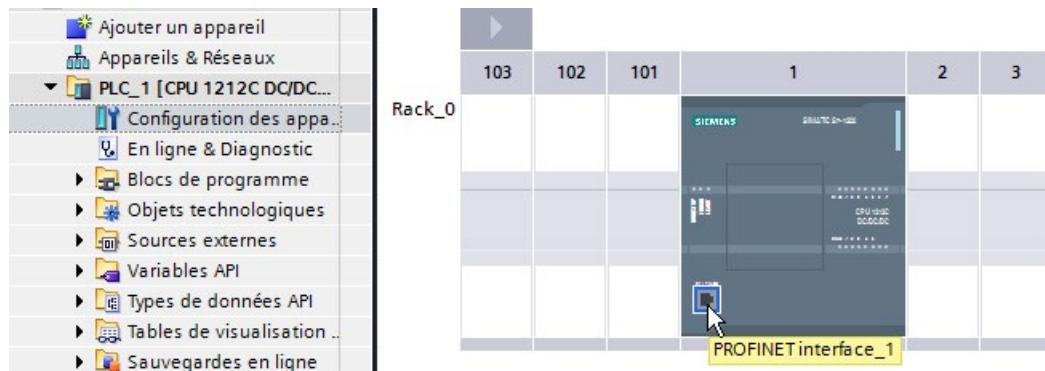
Mettre à jour l'adresse IP et mettre à jour le nom PROFINET sont des opérations DCP qui utilisent une adresse MAC pour accéder à l'appareil cible. Les appareils connectés indirectement doivent utiliser une adresse IP pour l'accès et ne peuvent pas utiliser les opérations DCP. Vous devez établir temporairement une connexion directe entre l'appareil cible et SIMATIC Automation Tool pour modifier l'adresse IP ou le nom PROFINET.

Opérations Modifier l'adresse IP et le nom PROFINET

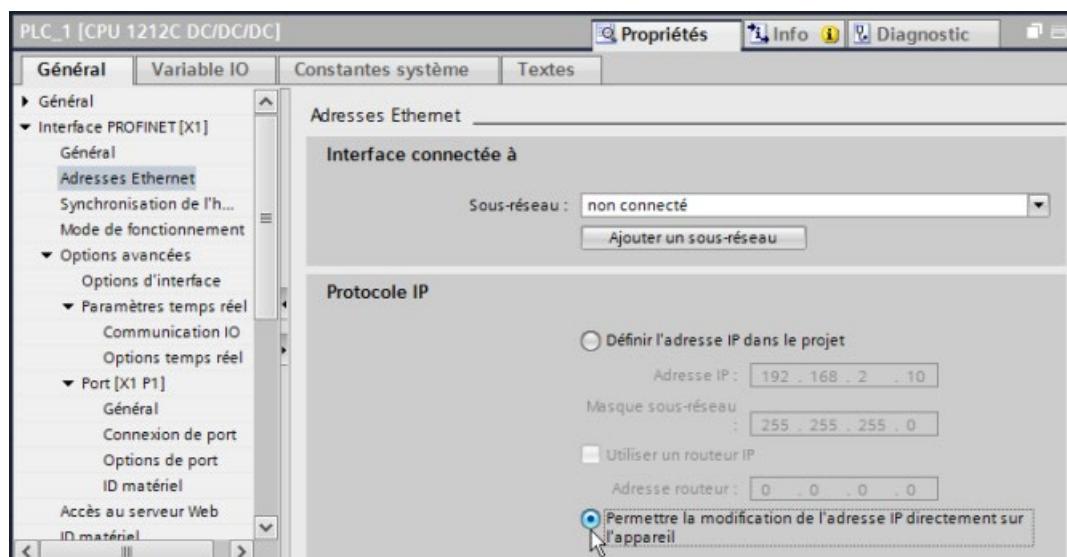
- **Possibles** pour les appareils PROFINET (CPU, IHM, E/S décentralisées et autres appareils) connectés directement au sous-réseau du réseau connecté à SIMATIC Automation Tool (y compris via un commutateur Ethernet).
- **Impossible**s pour les appareils PROFINET connectés indirectement derrière une CPU, un module CP/CM ou un module d'interface connecté directement, ou un port Ethernet secondaire d'une CPU.
- **Impossible**s pour les appareils PROFINET d'un autre réseau avec une connexion à SIMATIC Automation Tool qui passe par un routeur d'adresses IP.

Exemple de configuration de S7-1200 avec le logiciel TIA Portal

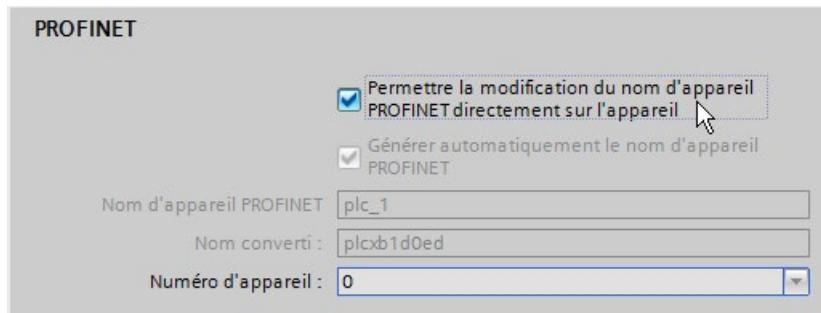
1. Cliquez sur le port PROFINET sur l'image de la configuration d'appareil de la CPU pour afficher les paramètres du port.



2. Sur l'onglet **Propriétés**, cliquez sur l'onglet **Général** pour afficher les options **Adresses Ethernet**. Cliquez sur l'option **Permettre la modification de l'adresse IP directement sur l'appareil**. Cette option peut aussi porter le nom "Set IP address on the device" (configurer l'adresse IP sur l'appareil) ou "Set IP address using a different method" (définir l'adresse IP avec une autre méthode) selon la version de TIA Portal utilisée. Pour les appareils multiports comme la CPU S7-1500, vous pouvez configurer tous les ports de manière similaire pour permettre la modification de l'adresse IP (si connecté à SIMATIC Automation Tool) ou vous pouvez configurer uniquement le port que vous souhaitez modifier.



3. Pour les options **Adresses Ethernet**, cliquez sur **Nom PROFINET défini directement sur l'appareil**. Cette option peut aussi porter le nom "Set PROFINET device name on the device option" (définir le nom d'appareil PROFINET sur l'option de l'appareil) selon la version de TIA Portal utilisée. La sélection permet à SIMATIC Automation Tool d'affecter un nom de station PROFINET. Pour les appareils multi-ports comme la CPU S7-1500, vous pouvez configurer tous les ports de manière similaire pour permettre la modification du nom PROFINET (si connecté à SIMATIC Automation Tool) ou vous pouvez configurer uniquement le port que vous souhaitez modifier.



4. Enregistrez le projet et téléchargez la nouvelle configuration sur la CPU.

Remarque

Réglages par défaut des paramètres IP PROFINET

Lorsque vous créez un nouveau projet TIA Portal, les options par défaut des paramètres PROFINET sont "**Set IP address in the project**" (définir l'adresse IP dans le projet) et "**Generate PROFINET device name automatically**" (Générer automatiquement de nom d'appareil PROFINET). Les options par défaut ne permettent pas de définir des adresses IP ou des noms d'appareil PROFINET avec SIMATIC Automation Tool. Cependant, vous pouvez utiliser d'autres opérations CPU comme la commande MARCHE/ARRET, les mises à jour du programme/du firmware, le réglage de l'heure et l'analyse des données de maintenance/ de diagnostic.

3.5 Configuration de la communication

Identification de la carte d'interface réseau raccordée au réseau de l'appareil.

Une fois que le PG/PC est connecté à un réseau, vous pouvez utiliser le panneau de commande Windows pour afficher le nom de la carte d'interface réseau.

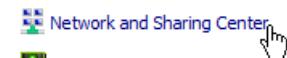
Dans l'exemple suivant, le réseau d'appareils SIMATIC est connecté via un convertisseur Ethernet à USB à un PC exécutant Windows 7. Les noms de réseau que vous voyez effectivement sur le PG/PC dépendent du matériel mis en réseau.

Utiliser le panneau de commande Windows pour identifier le nom de l'appareil.

1. Ouvrir le panneau de commande



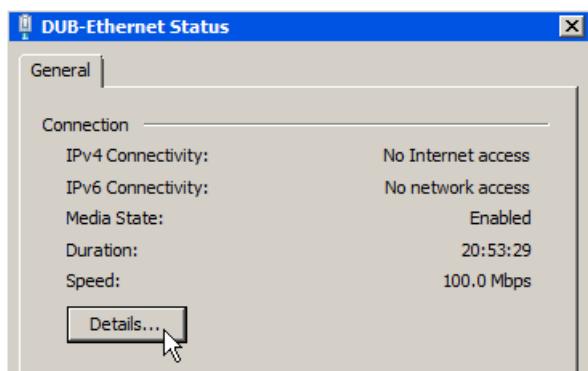
2. Cliquez sur "Centre Réseau et partage"



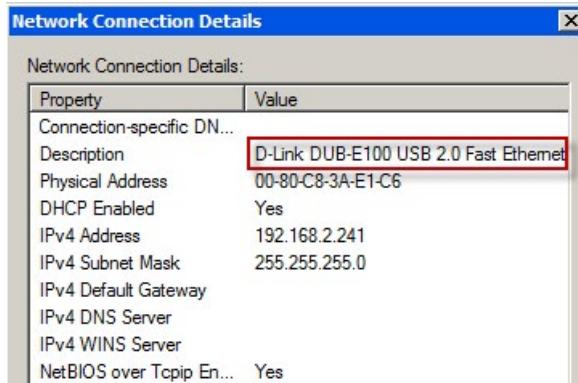
3. Affichez les réseaux actifs et cliquez sur celui qui est connecté aux CPU S7-1200.



4. Cliquer sur le bouton Détails dans la vue État de la connexion.



5. Afficher la description de l'interface réseau.

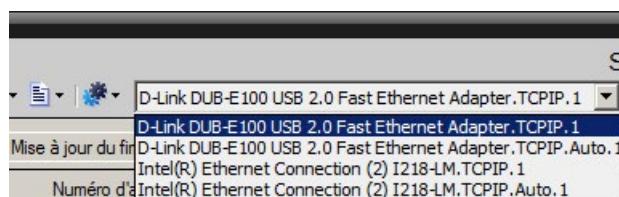


Affectation de l'interface réseau dans SIMATIC Automation Tool

Vous devez affecter l'interface réseau à un nouveau projet pour que la communication puisse commencer. Pour configurer l'interface réseau, procédez comme suit :

1. Lancez SIMATIC Automation Tool
2. Cliquez sur la liste déroulante de la carte d'interface réseau :
3. Choisissez l'interface réseau connectée à votre réseau d'appareils Siemens.

Il se peut que les interfaces réseau qui s'affichent soient différentes de celles présentées sur l'image suivante car la liste montre les interfaces réseau effectivement disponibles sur votre PG/PC.



Si vous avez sélectionné une carte d'interface réseau, mais que les appareils ne possèdent pas d'adresse IP valide, vous ne pouvez pas utiliser les opérations basées sur l'adresse IP. Vous pouvez utiliser les opérations basées sur l'adresse MAC et paramétrer des adresses IP valides pour les appareils de votre réseau.

Opérations avec adresse MAC

- Rechercher les appareils du réseau (Page 49)
- Identifier les appareils (Page 61)
- Définir les adresses IP Ethernet (Page 56)
- Set PROFINET names (Page 58)
- Restaurer les paramètres d'usine (Page 89) L'opération Restaurer les paramètres d'usine basée sur l'adresse MAC ne fonctionne que pour les périphériques PROFINET I/O ; les autres appareils utilisent une adresse IP.

Sélection de l'interface réseau

Comme nous l'avons vu précédemment, il peut y avoir deux entrées pour chaque carte réseau et la différence se trouve dans l'ajout des caractères ".Auto".

Lorsque vous sélectionnez l'interface Ethernet, vous avez le choix entre les types de protocole réseau suivants :

- TCP/IP
- TCP/IP.Auto

Il est recommandé de choisir **TCPIP** sans "Auto" car les adresses IP "virtuelles" ne sont pas créées automatiquement dans l'adaptateur Windows Ethernet. Vous devez affecter une adresse IP valide dans la configuration Windows pour l'adaptateur Ethernet du PG/PC.

Sinon, vous pouvez aussi choisir **TCPIP.Auto**. Une fois que vous avez scruté le réseau, vous devez vérifier que les adresses IP virtuelles créées automatiquement ne sont pas en conflit avec les adresses IP des autres appareils sur le réseau.

Le protocole **TCPIP.Auto** présente les avantages suivants :

- Le protocole **TCPIP.Auto** peut détecter des appareils accessibles que le protocole **TCPIP** ne trouve pas.
- Vous pouvez modifier les adresses IP des appareils réseau accessibles pour utiliser un sous-réseau compatible avec le protocole **TCPIP**.
- Après un scan du réseau, l'adaptateur réseau du PG/PC dispose toujours d'adresses IP virtuelles valides pour tous les appareils Siemens. Vous n'avez pas besoin d'affecter de nouvelles adresses IP explicitement dans Windows.

Cependant, le protocole **TCPIP.Auto** peut créer des problèmes de communication dans le réseau :

- Vous ne pouvez pas affecter des adresses IP virtuelles. Le système d'exploitation Windows affecte automatiquement les adresses IP virtuelles.
- Les adresses virtuelles sont perdues après un redémarrage ou une RAZ du PG/PC. De nouvelles adresses IP virtuelles sont créées lors du scan réseau suivant qui utilise le protocole **TCPIP.Auto**.
- Il est possible qu'une adresse IP virtuelle créée automatiquement soit déjà utilisée sur un autre nœud (p.ex. autre PG/PC non visible par un scan réseau SIMATIC Automation Tool). Un conflit d'adresses peut générer des erreurs de communication difficiles à diagnostiquer dans certains secteurs du réseau.

Exemple d'utilisation des protocoles TCPIP et TCPIP.Auto

Vous pouvez vérifier les adresses IP de l'adaptateur réseau Windows en entrant "ipconfig /all" sur la ligne de commande.

La commande "ipconfig /all" a été utilisée pour obtenir les adresses IP affichées dans l'exemple suivant.

- Après une réinitialisation du PG/PC (reboot) et avant de lancer un scan réseau SIMATIC Automation Tool, exécutez "ipconfig /all" sur la ligne de commande. Vous voyez ci-dessous le résultat pour la carte de l'adaptateur Ethernet connecté au réseau d'appareils Siemens. L'adresse IP configurée de l'adaptateur Windows Ethernet est 192.168.2.200.

```
Ethernet adapter Local Area Connection 3:
  Connection-specific DNS Suffix  . :
  Description . . . . . : D-Link USB2.0 Ethernet Adapter
  Physical Address . . . . . : BC-F6-85-D7-70-A2
  DHCP Enabled . . . . . : No
  Autoconfiguration Enabled . . . . . : Yes
  IPv4 Address . . . . . : 192.168.2.200<Preferred>
  Subnet Mask . . . . . : 255.255.255.0
  Default Gateway . . . . . :
  DHCPv4 Class ID . . . . . : wu004-id
  NetBIOS over Tcpip. . . . . : Disabled
```

- Raccordez un automate programmable Siemens S7-1200 configuré avec l'adresse IP 192.168.3.1. Le masque de sous-réseau est 255.255.255.0, l'appareil S7-1200 est donc effectivement configuré pour un autre sous-réseau. Le troisième octet est "3" et doit être "2" afin que la communication avec l'adresse de sous-réseau 192.168.2 de l'adaptateur Ethernet soit possible.
- Démarrez SIMATIC Automation Tool, configurez l'interface réseau sur le protocole **TCIP** et lancez un scan réseau. Dans ce cas, l'API S7-1200 **n'est pas** détecté car il est configuré avec une adresse de sous-réseau incorrecte.
- Modifiez l'interface réseau SIMATIC Automation Tool sur le protocole **TCPIP.Auto** et lancez un scan réseau.
- Le scan réseau utilise le protocole **TCPIP.Auto** et détecte l'appareil S7-1200. Les informations sur le nouvel appareil S7-1200 sont ajoutées à la table Appareil de SIMATIC Automation Tool.
- Exécutez "ipconfig /all" sur la ligne de commande.

Comme indiqué précédemment, une adresse IP virtuelle de substitution pour l'adaptateur Ethernet 192.168.3.241 a été créée automatiquement. L'adresse IP virtuelle de substitution permet d'accéder au sous-réseau 192.168.3.

```
Ethernet adapter Local Area Connection 3:
  Connection-specific DNS Suffix  . :
  Description . . . . . : D-Link USB2.0 Ethernet Adapter
  Physical Address . . . . . : BC-F6-85-D7-70-A2
  DHCP Enabled . . . . . : No
  Autoconfiguration Enabled . . . . . : Yes
  IPv4 Address . . . . . : 192.168.2.200<Preferred>
  Subnet Mask . . . . . : 255.255.255.0
  IPv4 Address . . . . . : 192.168.3.241<Preferred>
  Subnet Mask . . . . . : 255.255.255.0
  Default Gateway . . . . . :
  DHCPv4 Class ID . . . . . : wu004-id
  NetBIOS over Tcpip. . . . . : Disabled
```

Des adresses IP multiples sont créées lorsque le protocole **TCPIP.Auto** détecte plusieurs sous-réseaux dans un réseau complexe.

Les adresses IP virtuelles sont temporaires et sont supprimées si le PG/PC Windows est réinitialisé.

7. SIMATIC Automation Tool peut maintenant se connecter à l'appareil S7-1200 avec l'adresse IP virtuelle 192.168.3.241 et changer l'adresse IP de l'appareil S7-1200. Utilisez SIMATIC Automation Tool pour mettre à jour l'adresse IP. Changez l'adresse IP de 192.168.3.1 en 192.168.2.1.
8. Réinitialisez le PG/PC et redémarrez Windows. Toutes les adresses IP virtuelles sont supprimées au redémarrage.
9. Démarrer SIMATIC Automation Tool, configurez l'interface réseau sur le protocole **TCIP** et lancez un scan réseau.
L'API S7-1200 (192.168.2.1) **est détecté** et peut communiquer avec l'adaptateur Ethernet du PG/PC (192.168.2.200).
Aucune adresse IP virtuelle n'est créée. Seule l'adresse IP configurée dans les propriétés de l'adaptateur réseau Windows est utilisée.

Si la carte d'interface réseau est sélectionnée et que les adresses IP sont valides, vous pouvez utiliser les opérations SIMATIC Automation Tool qui s'appuient sur une adresse IP.

Opérations utilisant les adresses IP

- Mettre les CPU en mode RUN ou STOP (Marche/arrêt) (Page 60)
- Régler l'heure de la CPU sur l'heure du PG/PC (Page 95)
- Mettre à jour le programme pour une CPU et des appareils IHM (Page 63)
- Charger, ajouter, remplacer ou supprimer des données de recette d'une CPU (Page 73)
- Charger ou supprimer les données de Data log d'une CPU (Page 76)
- Sauvegarder/restaurer les données de la CPU et de l'IHM (Page 83)
- Récupérer les Service Data depuis des CPU (Page 93)
- Afficher le tampon de diagnostic de la CPU (Page 96)
- Réinitialiser la mémoire de la CPU (Page 88)
- Formater la carte mémoire dans une CPU (Page 91)
- Restaurer les paramètres d'usine des appareils (Page 89)
- Mettre à jour le firmware dans des appareils (Page 78)

Remarque

Problèmes de communication avec SIMATIC Automation Tool

Par exemple, vous envoyez une commande d'opération à plusieurs appareils, mais un des appareils ne termine pas l'opération. Le journal des événements (Event log) affiche une erreur de communication. Cependant, les autres appareils communiquent et exécutent l'opération comme prévu. Si vous rencontrez ce problème, procédez comme suit :

1. Réduisez le nombre d'opérations simultanées que vous autorisez sous Paramètres de communication (Page 110).
2. Fermez et redémarrez SIMATIC Automation Tool.
3. Essayez à nouveau l'opération groupée.

Si vous envoyez une commande d'opération à un appareil et le débit de transfert de données est très bas, une erreur timeout de communication peut être renvoyée. Si vous rencontrez ce problème, augmentez la durée du délai d'attente affecté aux opérations de communication sous Paramètres de communication (Page 110).

Opérations de l'outil logiciel

4.1 Mots de passe de la CPU

Si une protection par mot de passe est configurée sur une CPU, vous devez entrer ce mot de passe pour le niveau d'accès qui autorise SIMATIC Automation Tool à exécuter l'opération que vous voulez. Vous renseignez le mot de passe dans la colonne de la table Appareil intitulée "Mot de passe dans la CPU".

Le mot de passe saisi sous la colonne "Mot de passe dans la CPU" renvoie à la protection par mot de passe existant sur la CPU cible.

Par exemple, une nouvelle CPU Siemens dans son emballage n'a pas de programme, pas de configuration matérielle et pas de protection par mot de passe. Une fois qu'un nouveau projet est chargé sur la CPU, vous devez utiliser les mots de passe qui sont configurés dans ce projet.

SIMATIC Automation Tool vous indique si une CPU est ou non protégée par mot de passe. La cellule de mot de passe d'une CPU est gris foncé et n'est pas modifiable si la CPU n'a pas de mot de passe.

Utiliser des mots de passe

- Si une CPU est protégée par mot de passe, vous devez entrer un mot de passe dans la cellule "Mot de passe dans la CPU" pour que la Mise à jour du programme ou l'opération Restaurer à partir de la sauvegarde puisse être exécutée avec succès.
- L'onglet "Mise à jour du programme" comporte deux colonnes pour la saisie des mots de passe : "Mot de passe dans la CPU" et "Mot de passe dans le fichier de programme".

Un fichier programme peut avoir un mot de passe, qui peut être différent du mot de passe actuel de la CPU. Lorsqu'un fichier programme comporte un mot de passe, vous devez entrer le mot de passe du fichier programme dans la cellule "Mot de passe dans le fichier de programme" pour effectuer une mise à jour du programme. Une fois que l'opération Mise à jour du programme est terminée avec succès, SIMATIC Automation Tool copie le mot de passe du fichier programme vers la cellule de mot de passe de la CPU. À l'aide de ce mot de passe, SIMATIC Automation Tool essaie ensuite de se connecter à la CPU

- L'onglet "Restaurer à partir de la sauvegarde" comporte deux colonnes pour la saisie des mots de passe : "Mot de passe dans la CPU" et "Mot de passe dans le fichier de sauvegarde".

Un fichier de sauvegarde peut avoir un mot de passe, qui peut être différent du mot de passe actuel de la CPU. Lorsqu'un fichier de sauvegarde comporte un mot de passe, vous devez entrer le mot de passe du fichier de sauvegarde dans la cellule "Mot de passe dans le fichier de sauvegarde". Le mot de passe du fichier de sauvegarde devient le mot de passe de la CPU après l'achèvement de l'opération Restaurer à partir de la sauvegarde. SIMATIC Automation Tool copie le mot de passe du fichier de sauvegarde dans la cellule de mot de passe de la CPU après l'achèvement de l'opération Restaurer à partir de la sauvegarde. À l'aide de ce mot de passe, SIMATIC Automation Tool essaie ensuite de se connecter à la CPU.

- En cas de saisie de mot de passe de CPU valide, vous pouvez déplacer le curseur de votre souris sur le champ mot de passe pour afficher une infobulle qui indique le niveau d'accès.

Niveaux d'accès associés aux mots de passe dans la CPU

Une CPU standard possède quatre niveaux d'accès et une CPU de sécurité en possède cinq.

Les opérations SIMATIC Automation Tool qui nécessitent un accès en lecture ou en écriture ne peuvent pas fonctionner avec une CPU qui possède un niveau de protection de niveau "Accès IHM" ou "Pas d'accès". Vous devez configurer un mot de passe "Accès en lecture" ou "Accès complet", puis saisir ce mot de passe dans la ligne de la table Appareil correspondant à la CPU cible.

Vous pouvez voir les niveaux d'accès de la CPU dans la section Protection & Sécurité de la Configuration de l'appareil de TIA Portal.

Protection

Sélectionnez le niveau d'accès pour l'API.

Niveau d'accès	Accès				Permission ..
	IHM	Lire	Ecrire	De sécu...	
<input type="radio"/> Accès complet, y compris failsafe (pas d..	✓	✓	✓	✓	*****
<input type="radio"/> Accès complet (pas de protection)	✓	✓	✓		*****
<input type="radio"/> Accès en lecture	✓	✓			*****
<input checked="" type="radio"/> Accès IHM	✓				
<input type="radio"/> Aucun accès (protection complète)					

Accès IHM :
Les utilisateurs de TIA Portal n'obtiennent pas d'accès aux fonctions standard et failsafe.
Les applications IHM peuvent accéder à toutes les fonctions (failsafe et standard).

Mot de passe obligatoire :
Pour un accès supplémentaire en lecture/écriture et un accès aux fonctions failsafe, l'utilisateur de TIA Portal doit saisir le mot de passe pour "Accès complet, y compris failsafe".

Mot de passe optionnel :
Pour un accès en lecture/écriture supplémentaire à des fonctions standard sans accès aux fonctions Failsafe, un mot de passe peut être défini pour "Accès en lecture/écriture" ou "Accès en lecture".

Pour plus d'informations sur les niveaux d'accès et les mots de passe, reportez-vous au *Système d'information STEP 7* (aide en ligne TIA Portal).

CPU de sécurité et mots de passe

SIMATIC Automation Tool peut fonctionner avec des CPU de sécurité. Vous pouvez vous connecter à une CPU de sécurité et effectuer certaines opérations à l'aide d'un mot de passe "Accès en lecture", ou d'un mot de passe "Accès complet (pas de protection)", appelé dans ce guide "mot de passe de CPU F de sécurité". Toutefois, les opérations relatives à la sécurité requièrent le mot de passe de CPU F de sécurité.

- The SIMATIC Automation Tool affiche la cellule de mot de passe d'un appareil en jaune lorsque vous entrez un mot de passe de CPU F de sécurité.
- Si un API de sécurité est protégé par mot de passe, toutes les opérations relatives à la sécurité nécessitent que vous entrez le mot de passe de CPU F de sécurité dans la colonne "Mot de passe dans la CPU".

Si une CPU de sécurité n'utilise pas de protection par mot de passe, les opérations relatives à la sécurité ne nécessitent pas le mot de passe de CPU F de sécurité pour lancer l'opération.

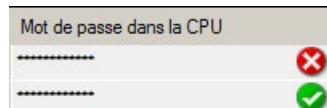
Les opérations relatives à la sécurité sont les suivantes :

- Mise à jour du programme
- Restaurer l'appareil depuis un fichier
- Restaurer les paramètres d'usine
- Formater la carte mémoire
- Lorsqu'un fichier programme comporte un programme de sécurité, vous devez entrer le mot de passe de CPU F de sécurité dans la colonne "Mot de passe dans le fichier de programme" pour effectuer l'opération "Mise à jour du programme".
- Lorsqu'un fichier de sauvegarde comporte un programme de sécurité, vous devez entrer le mot de passe de CPU F de sécurité dans la colonne "Mot de passe dans le fichier de sauvegarde" pour effectuer l'opération "Restaurer à partir de la sauvegarde".

Icônes relatives aux mots de passe

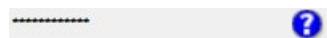
SIMATIC Automation Tool génère trois icônes pour indiquer le statut du mot de passe saisi.

Une coche verte signifie que le mot de passe est un mot de passe valide dans la CPU ou le fichier programme. Une croix rouge signifie que le mot de passe n'est pas un mot de passe valide dans la CPU ni le fichier programme.



Lorsque vous entrez un mot de passe, SIMATIC Automation Tool ne sait pas quelles opérations vous allez lancer. Par conséquent, le mot de passe n'est pas validé à la saisie pour une opération spécifique. Par exemple, la mise à jour d'un programme de sécurité requiert le mot de passe de CPU F de sécurité. Pour de nombreuses autres opérations, le mot de passe standard Accès complet (lecture / écriture) suffit. Une coche verte ne signifie pas que le mot de passe est validé pour toutes les opérations. La validation du niveau d'accès s'effectue lorsque l'opération est lancée. Si le mot de passe n'est pas suffisant pour tel niveau d'accès, cela génère une erreur dans le journal Event log pour chaque appareil concerné.

La cellule de mot de passe du fichier de sauvegarde est la seule à pouvoir afficher un point d'interrogation :



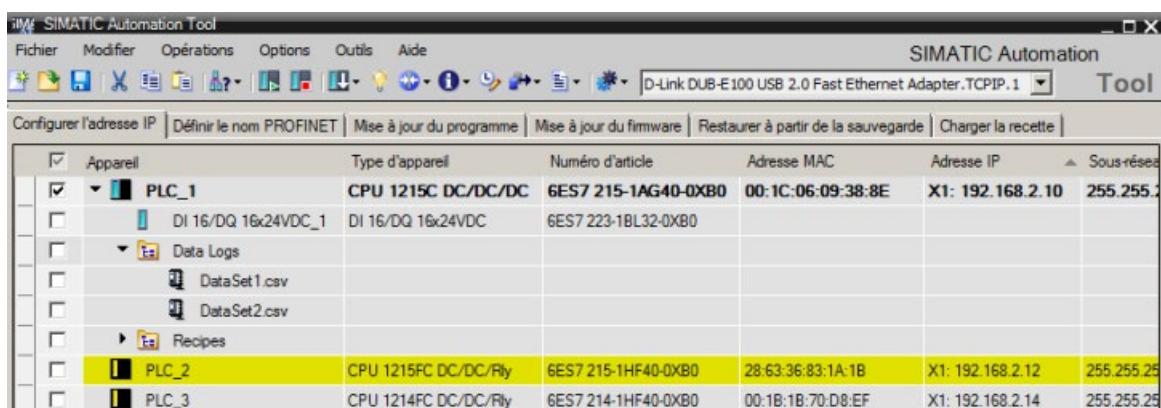
La cellule de mot de passe du fichier de sauvegarde ne comporte aucune icône lorsque le champ est désactivé ou vide. Lorsque vous sélectionnez un fichier de sauvegarde dans la liste ou entrez un mot de passe de CPU dans la colonne de mot de passe du fichier de sauvegarde, SIMATIC Automation Tool affiche l'icône de point d'interrogation. SIMATIC Automation Tool ne peut pas valider le mot de passe au moment où vous sélectionnez un fichier ou saisissez un mot de passe.

4.2 Travailler avec la table Appareil et le journal des événements (Event Log)

SIMATIC Automation Tool est constitué de deux parties :

- Table Appareil : Dans un premier temps, la table Appareil est vide, mais, après un scan du réseau, les appareils connectés y sont affichés.
- Journal des événements (Event Log) : L'Event log affiche les résultats des opérations.

Pour chaque appareil, SIMATIC Automation Tool affiche des colonnes contenant des données sur l'appareil. Les onglets dans la table Appareil prennent en charge différentes opérations et fournissent des champs permettant de saisir les données requises.



The screenshot shows the SIMATIC Automation Tool interface with the title bar 'SIMATIC Automation Tool'. The menu bar includes 'Fichier', 'Modifier', 'Opérations', 'Options', 'Outils', and 'Aide'. The toolbar contains various icons for file operations. The main window shows a table titled 'Appareil' with the following data:

	Type d'appareil	Numéro d'article	Adresse MAC	Adresse IP	Sous-réseau
<input checked="" type="checkbox"/>	PLC_1	CPU 1215C DC/DC/DC	6ES7 215-1AG40-0XB0	00:1C:06:09:38:8E	X1: 192.168.2.10 255.255.255.255
<input type="checkbox"/>	DI 16/DQ 16x24VDC_1	DI 16/DQ 16x24VDC	6ES7 223-1BL32-0XB0		
<input type="checkbox"/>	Data Logs				
<input type="checkbox"/>	DataSet1.csv				
<input type="checkbox"/>	DataSet2.csv				
<input type="checkbox"/>	Recipes				
<input checked="" type="checkbox"/>	PLC_2	CPU 1215FC DC/DC/Ry	6ES7 215-1HF40-0XB0	28:63:36:83:1A:1B	X1: 192.168.2.12 255.255.255.255
<input type="checkbox"/>	PLC_3	CPU 1214FC DC/DC/Ry	6ES7 214-1HF40-0XB0	00:1B:1B:70:D8:EF	X1: 192.168.2.14 255.255.255.255

Travailler avec la table Appareil

La table Appareil est similaire à Microsoft Excel et permet d'effectuer des opérations de copier et coller vers ou à partir d'autres applications.

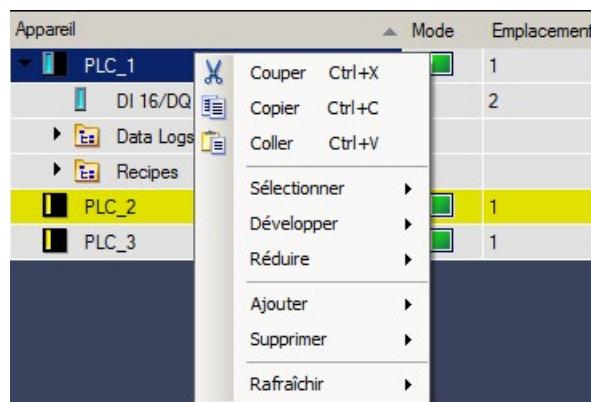
Les conseils suivants peuvent vous aider à utiliser la table Appareil :

- Cliquer sur l'en-tête d'une colonne pour trier les lignes selon les données de colonne par ordre croissant ou décroissant.
- Faire un clic droit sur l'en-tête d'une colonne pour l'afficher/la masquer.
- Cochez la case Appareil située en haut de la table Appareil pour sélectionner / désélectionner tous les appareils. Vous pouvez également utiliser la commande de menu "Modifier > Sélectionner" pour choisir soit "Sélectionner les lignes" soit "Désélectionner les lignes". Vous pouvez également faire un clic droit sur une ligne de l'appareil pour accéder au menu Modifier.
- Cliquez sur le coin supérieur gauche de la table Appareil pour mettre en surbrillance toutes les lignes.
- Sélectionnez plusieurs lignes consécutives : Vous pouvez cliquer sur la cellule à gauche de la case à cocher d'une ligne et faire glisser le curseur vers le haut / bas pour mettre en surbrillance plusieurs lignes.
- Lorsque vous faites un clic droit sur une ligne ou une sélection de lignes, le menu contextuel ci-dessous vous permet d'utiliser les commandes Sélectionner, Développer, Réduire, Insérer, Supprimer et Actualiser avec un groupe de lignes.

- Vous pouvez créer des filtres de lignes pour les colonnes Appareil, Type d'appareil et Numéro d'article.
- Sélectionnez un ou plusieurs appareils pour effectuer des opérations sur plusieurs appareils. SIMATIC Automation Tool affiche en gras les lignes sélectionnées.
- Vous pouvez également Exporter (Page 100) une table Appareil vers un fichier .csv ou Importer (Page 100) un fichier .csv dans la table Appareil.

Vous pouvez Scanner le réseau (Page 49) pour remplir la table Appareil de votre réseau. Vous pouvez également Insérer des appareils (Page 55) directement.

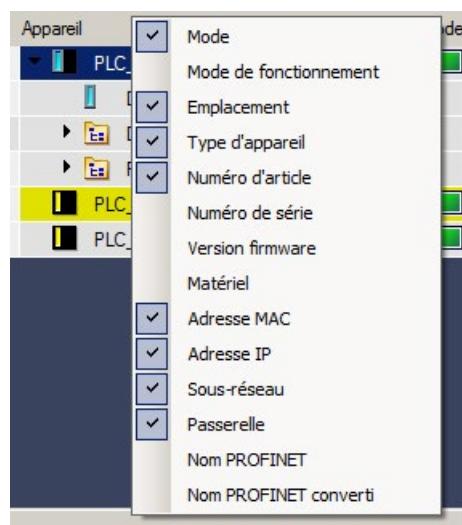
Menu contextuel (clic droit) pour les cellules de la table



Lorsque vous faites un clic droit sur une ligne de la table Appareil, le menu contextuel est identique au menu Modifier (Page 104).

Menu contextuel (clic droit) pour les en-têtes des colonnes

Pour chaque onglet de la table Appareil, SIMATIC Automation Tool affiche un ensemble de colonnes par défaut. Vous pouvez configurer les colonnes que vous souhaitez afficher et les colonnes que vous souhaitez masquer.



Pour configurer les colonnes à afficher ou à masquer, procédez comme suit :

1. Faites un clic droit sur la ligne d'en-tête de l'appareil pour afficher le menu contextuel pour les colonnes.
2. Sélectionnez les cases à cocher pour afficher ou masquer les colonnes pour cette vue d'onglet.

Filtrage des lignes affichées

Vous pouvez filtrer les colonnes Appareil, Type d'appareil et Numéro d'article. Cliquez sur l'un de ces en-têtes de colonne pour afficher le bouton . Cliquez sur ce bouton pour ouvrir une fenêtre de filtre.

Par exemple, vous pouvez sélectionner les numéros d'article 6ES7 214-1HF40-0XB0 et 6ES7 215-1HF40-0XB0. Lorsque vous cliquez sur le bouton OK, la table Appareil affiche uniquement les lignes contenant cette valeur.

Appareil	Mode	Emplacem...	Type d'appareil	Numéro d'article	Adresse MAC
PLC_3		1	CPU 1214FC DC/DC/Rly		0:D8:EF
PLC_1		1	CPU 1215C DC/DC/DC		9:38:8E
DI 16/DQ 16x24VDC_1		2	DI 16/DQ 16x24VDC		
▶ Data Logs					
▶ Recipes					
PLC_2		1	CPU 1215FC DC/DC/Rly		3:1A:1B

Filtrage des appareils non pris en charge

Vous pouvez utiliser Paramètres généraux (Page 109) pour activer/désactiver l'affichage des appareils non pris en charge.

SIMATIC Automation Tool affiche en gris les appareils non pris en charge. Vous pouvez effectuer uniquement les opérations suivantes sur les appareils non pris en charge :

- Configurer l'adresse IP
- Définir le nom PROFINET
- Identifier l'appareil
- Éditer le commentaire pour l'appareil
- Utiliser des fonctions d'édition comme par ex. copier - coller

Travailler avec le journal des événements (Event Log)

SIMATIC Automation Tool affiche l'Event Log dans la zone de la fenêtre qui se trouve sous la table Appareil. Lorsque vous sélectionnez des appareils et exécutez des opérations, les messages figurant dans l'Event Log affichent des informations d'état sur les résultats opérationnels.

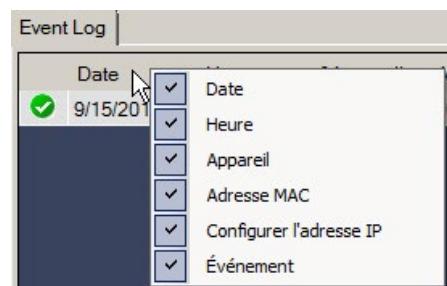
Par défaut, toute opération nouvelle sur un appareil efface l'Event Log au début de l'opération. Vous pouvez choisir de supprimer ou non l'Event Log au début d'une opération dans Paramètres de l'Event Log (Page 119).

Le SIMATIC Automation Tool peut également journaliser automatiquement des états d'opérations dans un fichier.

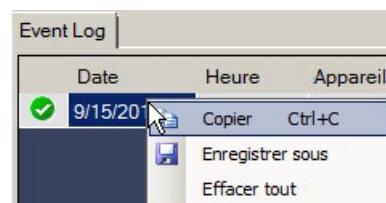
Ligne de l'Event Log affichant une opération réussie :

Date	Heure	&Appareil	Adresse MAC	Adresse IP	Événement	Résultat
9/15/2017	2:21 PM	PLC_1	00:1C:06:09:38:8E	X1: 192.168.2.10	Transition vers RUN	Opération exécutée.

Faites un clic droit sur l'en-tête d'une colonne de l'Event Log pour afficher/masquer la colonne :



Cliquez avec le bouton droit sur une ligne d'événement pour ouvrir le menu ci-dessous :



Les icônes de l'Event Log ont les significations suivantes :

	Opération terminée avec succès
	Échec de l'opération La colonne Résultat indique la cause de l'échec. Si vous sauvegardez l'Event Log, ces entrées commencent par "ERREUR :" .
	Opération terminée avec succès, mais ayant généré un avertissement. La colonne Résultat décrit l'avertissement. Si vous sauvegardez l'Event Log, ces entrées commencent par "AVERTISSEMENT :" .

Remarque

Modifier la langue de l'Event Log et de l'interface utilisateur

Lorsque vous modifiez la langue de l'interface utilisateur de SIMATIC Automation Tool, l'Event Log est effacé. Les informations concernant les événements antérieurs sont supprimées.

4.3 Options de traitement multifil (multi-thread)

Traitement Multi-thread

Si vous gérez un réseau contenant de nombreux appareils, SIMATIC Automation Tool peut simplifier et réduire le temps de pilotage en traitant automatiquement un groupe d'appareils avec des fils (threads) multiples. Pendant qu'un thread de traitement de communication du PG/PC attend le message indiquant que la tâche de l'appareil SIMATIC est terminée, d'autres threads peuvent utiliser ce temps pour communiquer avec d'autres appareils du groupe.

Pour les versions 3.0 ou ultérieures, vous devez avoir une licence logicielle SIMATIC Automation Tool installée avant de pouvoir utiliser plus d'un thread de traitement.

Appareils dans un réseau à topologie en étoile

Si votre réseau possède une topologie en étoile dans laquelle chaque appareil est connecté directement au PG/PC via un commutateur Ethernet, vous pouvez activer l'option de traitement multi-thread en toute sécurité.

Appareils dans un réseau à topologie en chaîne

Si votre réseau possède une topologie en chaîne, vous devez désactiver l'option multi-thread, afin d'éviter qu'un appareil n'interrompe la communication vers les autres appareils, comme décrit à la rubrique Paramètres de communication (Page 110).

Options de traitement multifil (multi-thread)

Dans la boîte de dialogue Options> Paramètres> Communications, vous pouvez modifier les paramètres suivants :

- Activer ou désactiver le traitement multifil au moment d'effectuer ces opérations : Mettre à jour le firmware, Restaurer les paramètres d'usine, RAZ mémoire, Restaurer des données à partir du fichier de sauvegarde, Réinitialiser la mémoire de la CPU, Formater la carte mémoire SIMATIC, et Restaurer les paramètres d'usine des appareils.
- Définissez le nombre maximal de threads autorisé (de 1 à 5 threads).
- Définissez le délai d'attente, qui est le temps maximal pendant lequel un thread de communication attendra une réponse (de 180 à 999 secondes).

Tableau des restrictions applicables au traitement multi-thread

- ✓ Traitement multi-thread permanent
- Traitement multi-thread possible, si activé dans Paramètres de communication (Page 110).
- ✗ Opération de sécurité sur un appareil de sécurité : Traitement à thread unique uniquement.

Opération SIMATIC Automation Tool	Traitement multi-thread pour appareil standard	Traitement multi-thread pour appareil de sécurité
Scanner le réseau	✓	✓
Identifier les appareils	✓	✓
Mettre à jour les adresses des appareils	✓	✓
Mettre à jour le nom PROFINET d'un appareil	✓	✓
Régler la date et l'heure d'une CPU sur celles de votre PG/PC	✓	✓
Mettre à jour le programme pour une CPU et des appareils IHM	✓	✗
Charger, ajouter, remplacer ou supprimer des données de recette d'une CPU	✓	✓
Charger ou supprimer les données de Data log d'une CPU	✓	✓
Mettre à jour le firmware dans un appareil	○	○
Placer une CPU en mode RUN ou STOP	✓	✓
Sauvegarder des données vers un fichier de sauvegarde pour une CPU ou un appareil IHM	✓	✓
Restaurer des données depuis un fichier de sauvegarde pour une CPU ou un appareil IHM	○	✗
Afficher le tampon de diagnostic d'une CPU	✓	✓
Réinitialiser la mémoire de la CPU	○	○
Récupérer les Service Data depuis une CPU	✓	✓
Formater la carte mémoire dans une CPU (carte mémoire SIMATIC insérée dans une CPU)	○	✗
Restaurer les paramètres d'usine	○	✗

Files d'attente de traitement

Vous pouvez sélectionner un groupe d'appareils standard et de sécurité dans les lignes de la table Appareil, puis lancer le traitement groupé afin que SIMATIC Automation Tool effectue la même opération sur tous les appareils du groupe. SIMATIC Automation Tool utilise deux files d'attente de traitement, une file d'attente prioritaire pour les appareils de sécurité, dédiée au traitement des opérations relatives à la sécurité (traitement à thread unique uniquement) et une file d'attente secondaire qui utilise de 1 à 5 threads pour les appareils standard et les appareils de sécurité permettant le traitement multi-thread.

En premier lieu, dans SIMATIC Automation Tool, la file d'attente des opérations de sécurité portant sur les appareils de sécurité fait l'objet d'un traitement à thread unique. Chaque opération est effectuée sur un appareil de sécurité à la fois.

Une fois que SIMATIC Automation Tool a terminé la file d'attente des opérations de sécurité portant sur les appareils de sécurité, il applique à la seconde file d'attente un traitement multi-thread. Certaines opérations utilisent systématiquement le traitement multi-thread. D'autres l'utilisent à condition qu'il soit activé dans les Paramètres de communication (Page 110).

État du traitement des threads

Lorsque vous démarrez une opération groupée, SIMATIC Automation Tool affiche un message indiquant, pour chaque file d'attente, le nombre d'appareils sur lesquels l'opération est terminée et la progression (%) du traitement des threads actifs.

4.4 Scanner un réseau

La table Appareil d'un nouveau projet SIMATIC Automation Tool est vide. Pour commencer à travailler avec SIMATIC Automation Tool, vous scannez le réseau de communication pour remplir la table Appareil. Vous pouvez également insérer un appareil (Page 55) manuellement.

Scanner le réseau

Pour scanner le réseau, sélectionnez la commande de menu "Opérations>Scanner le réseau>Scanner le réseau complet". Vous pouvez également cliquer sur le bouton Scan dans la barre d'outils  et sélectionner "Scanner le réseau complet" dans le menu déroulant du bouton.

Remarque

SIMATIC Automation Tool fonctionne en mode non connecté.

Notez que SIMATIC Automation Tool n'effectue pas de mise à jour continue des données des appareils. SIMATIC Automation Tool affiche les informations sur les appareils au moment où vous avez scanné ou actualisé votre réseau de communication, ou au moment où vous avez inséré des appareils. TIA Portal ou le serveur Internet, par exemple, peuvent avoir modifié les données de vos appareils depuis votre dernier scan. Avant d'effectuer des opérations sur les appareils, scannez le réseau ou actualisez les appareils sur lesquels vous souhaitez effectuer des opérations.

Conventions applicables à la table Appareil

Des icônes de ligne vous aident à identifier les lignes de la table Appareil :

 Appareil non reconnu ou non pris en charge entièrement. Le texte d'adresse de la ligne est grisé.

 Appareil PROFINET

 Appareil PROFINET de sécurité

 Appareil PROFINET IHM

 Appareil PROFINET IHM de sécurité

 Dossier contenant les appareils maître PROFINET

 Dossier contenant les appareils maître PROFIBUS

 Dossier contenant les appareils maître AS-i PROFINET

 Dossier contenant le Data Log ou les données de recette

 Données du Data log

 Données de recette

 192.168.2.22 Les adresses IP et les noms de station PROFINET en double sont affichés en rouge.

 Problème d'identification de l'appareil standard

 Problème d'identification de l'appareil de sécurité

Remarque

Connexions en ligne vers les appareils depuis TIA Portal

Si un appareil possède une connexion en ligne dans TIA Portal, SIMATIC Automation Tool ne peut pas lire les informations de l'appareil. La table Appareil affiche l'icône indiquant que l'appareil n'est pas reconnu et écrit l'un des messages suivants dans l'Event Log :

- SIMATIC Automation Tool ne prend pas en charge cet appareil.
- Impossible d'établir une connexion à l'appareil.

Pour pouvoir lire l'appareil, passez en mode non connecté dans TIA Portal.

Cliquez sur la case située à côté de l'appareil que vous voulez sélectionner. SIMATIC Automation Tool affiche le texte sur l'appareil en noir pour les appareils que vous n'avez pas sélectionnés et en **noir gras** lorsque vous les avez sélectionnés.

Vous pouvez saisir du texte dans les cellules à fond gris clair. Vous ne pouvez pas saisir de texte dans les cellules à fond gris foncé. Une cellule à fond gris foncé indique que SIMATIC Automation Tool ne prend pas en charge l'opération pour ce type d'appareil / cette version de firmware.

Comprendre la table Appareil

Remarque

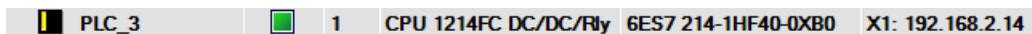
Effectuez un nouveau scan du réseau pour résoudre les problèmes d'identification des appareils.

Un problème d'identification de l'appareil peut se produire, p. ex. si TIA Portal a modifié le programme ou la configuration de l'appareil depuis le dernier scan du réseau par SIMATIC Automation Tool. Aucune opération, ni commande d'actualisation, n'est possible sur un appareil qui présente un problème d'identification. Vous devez scanner votre réseau pour résoudre le problème d'identification.

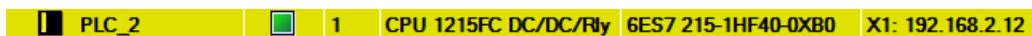
Appareils de sécurité

Lorsque vous modifiez l'identité d'un appareil ou l'état de son programme de sécurité à partir de SIMATIC Automation Tool, l'outil effectue les modifications sans réitérer le scan du réseau. Par exemple, si vous téléchargez une nouvelle version de firmware à partir de SIMATIC Automation Tool, SIMATIC Automation Tool met à jour les variables de l'appareil pour tenir compte des nouvelles valeurs. SIMATIC Automation Tool est un outil approuvé pour fonctionner sur des appareils de sécurité et est capable de gérer les changements d'état de sécurité.

Si vous avez une CPU de sécurité dans votre réseau, mais que vous n'avez pas téléchargé de programme de sécurité (Page 124) sur cette CPU, les cellules de la ligne correspondant à l'appareil sont affichées en gris.



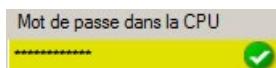
Si vous avez téléchargé un programme de sécurité pour votre CPU de sécurité, les champs de la ligne correspondant à l'appareil sont affichés en jaune.



Dans le cas des CPU de sécurité pour lesquelles il existe un programme de sécurité, les champs de saisie utilisateur suivants sont affichés initialement en gris. SIMATIC Automation Tool affiche ces champs de saisie utilisateur en jaune une fois que vous avez saisi des valeurs valides.

- Mot de passe dans la CPU
- Dossier de mise à jour du programme
- Mot de passe dans le fichier de programme
- Fichier de sauvegarde

Exemple :



Appareils connectés via des CPU et des routeurs d'adresses IP

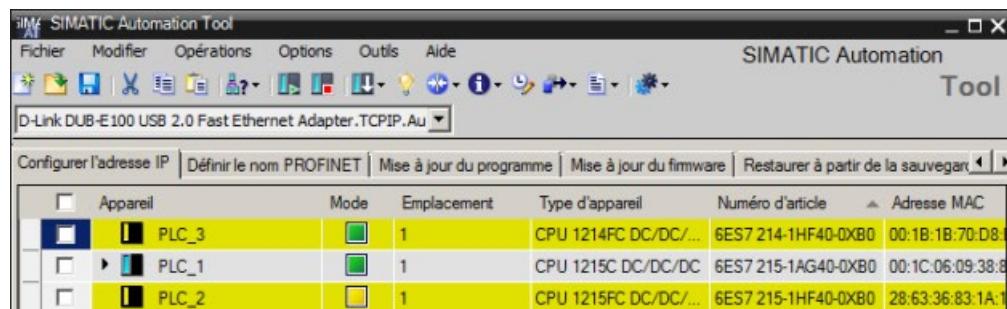
Une fois que des adresses IP valides ont été saisies dans l'onglet Adresse IP de la table Appareil, vous pouvez utiliser la commande de menu "Opérations>Mettre à jour>Adresse IP" pour transférer les affectations d'adresses vers les appareils directement connectés que vous avez sélectionnés. Vous pouvez également cliquer sur le bouton Mettre à jour  et choisir la commande "Adresse IP" dans le menu déroulant du bouton.

Si les appareils du réseau pris en charge ont des adresses IP valides, un scan réseau affiche les appareils situés derrière les CPU et les routeurs d'adresse IP.

Appareils raccordés directement (notamment avec une connexion via un commutateur Ethernet)

Un appareil raccordé directement peut utiliser toutes les opérations avec adresse MAC (avec adresse IP configurée ou non configurée) et toutes les opérations avec adresses IP (avec adresse IP configurée).

Exemple de scan initial :



Règles de scan pour les entrées de table existantes

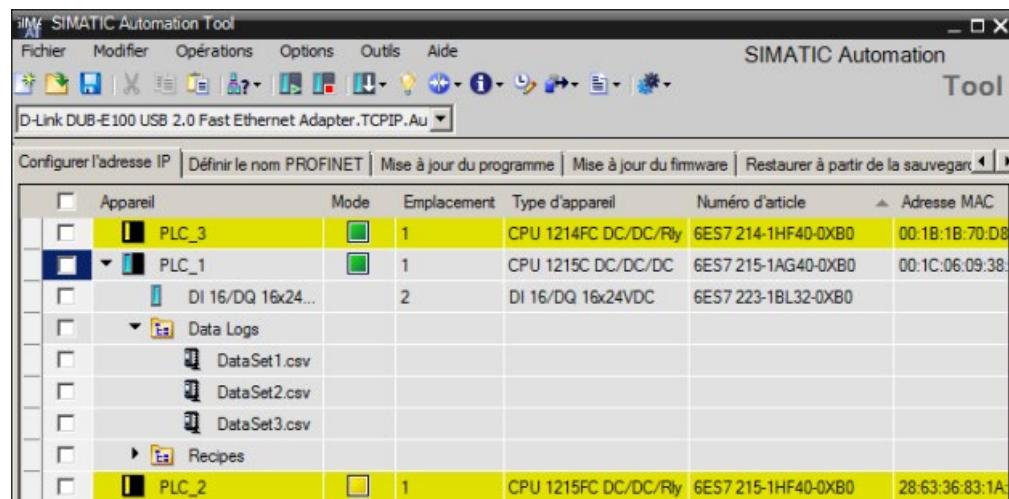
- Si une adresse MAC existe déjà dans la table, un scan du réseau permet de mettre à jour les champs Adresse IP, Sous-réseau et Passerelle de la ligne correspondant à cet appareil. Les données de tous les autres champs sont conservées.
- Si l'adresse MAC est nouvelle, SIMATIC Automation Tool crée une nouvelle ligne avec les champs Adresse MAC, Adresse IP, Sous-réseau et Passerelle. Tous les autres champs sont vides.

PROFINET I/O

Les périphériques PROFINET peuvent apparaître deux fois dans la table Appareil. L'appareil est affiché une fois sur une ligne de niveau supérieur où la connexion directe à l'outil permet toutes les opérations SIMATIC Automation Tool prises en charge. Il apparaît également dans une ligne de niveau inférieur derrière une CPU (avec une adresse IP et une configuration matérielle valides), où une connexion indirecte à l'outil restreint la ligne d'appareil à la seule mise à jour du firmware. Les deux lignes de la table Appareil résultent des deux différents chemins de connexion possibles sur le réseau Ethernet.

Développez les lignes d'appareil et affichez les modules locaux, les périphériques décentralisés, les HMI panels et les fichiers de CPU (recette et Data logs).

Cliquez sur l'icône ▶ pour développer une ligne d'appareil. Utilisez le menu contextuel (clic droit) ou le menu Modifier pour développer/réduire les niveaux.



Pour les appareils raccordés indirectement, seule la mise à jour du firmware est possible.

Les appareils sur des niveaux inférieurs représentent les appareils et les fichiers données de la CPU qui sont connectés indirectement à SIMATIC Automation Tool via une CPU raccordée directement. Une CPU doit disposer d'une adresse IP et d'une configuration matérielle valides pour que les appareils qui sont connectés par son intermédiaire deviennent visibles dans la table Appareil.

Les appareils des niveaux 3 et 4 peuvent représenter des périphériques I/O décentralisés (périphériques PROFINET et PROFIBUS). Un contrôleur IO décentralisé de niveau 2 doit disposer d'une configuration IP pour que les périphériques I/O décentralisés (modules de tête et modules E/S, par ex.) deviennent visibles dans la table Appareil.

Identification par mot de passe

Si l'appareil est protégé par mot de passe (Page 37), SIMATIC Automation Tool active le champ mot de passe quel que soit le niveau de protection.

Actualiser les données de la table Appareil

SIMATIC Automation Tool actualise les lignes de données de la table Appareil dans les cas suivants :

- Lors d'un scan du réseau
- Après avoir terminé une opération
- Lorsque vous actualisez un ou plusieurs appareils

Pour actualiser les données de la table Appareil, choisissez l'une des méthodes suivantes :

- Sélectionnez "Scanner le réseau complet" soit dans le menu déroulant du bouton  de la barre d'outils, soit dans le menu "Opérations > Scanner le réseau" pour actualiser tous les appareils.
- Sélectionnez les appareils et sélectionnez "Rafraîchir l'état de tous les appareils sélectionnés" soit dans le menu déroulant du bouton  de la barre d'outils, soit dans le menu "Opérations > Scanner le réseau".
- Sélectionner la commande de menu Modifier>Actualiser ou faites un clic droit sur la ligne associée à un appareil et sélectionnez Actualiser dans le menu contextuel. Ensuite, choisissez l'une des options suivantes à partir de la commande de menu Actualiser :
 - Appareil
 - Tous les appareils sélectionnés - F5
 - Tous les appareils
- Appuyez sur la touche F5 pour actualiser "Tous les appareils sélectionnés"

SIMATIC Automation Tool actualise les données qu'il lit sur les appareils et conserve tous les champs de données saisis par l'utilisateur.

Si vous actualisez des appareils qui ne sont plus présents sur le réseau, SIMATIC Automation Tool affiche les données de la ligne correspondante en italique.

Voir aussi

Paramètres de l'Event Log (Page 119)

Paramètres généraux (Page 109)

4.5 Insérer un appareil

Vous pouvez également insérer un appareil dans la table Appareil. Vous pouvez uniquement ajouter un appareil doté d'une adresse MAC et d'une adresse IP uniques par rapport aux autres appareils répertoriés dans la table Appareil.

Pour insérer un appareil, procédez comme suit :

1. Sélectionnez la commande de menu "Insérer > Appareil" soit dans le menu Modifier, soit dans le menu contextuel (clic droit) de la table Appareil.
2. Dans la boîte de dialogue "Ajouter l'appareil", entrez soit une adresse IP soit une adresse MAC pour l'appareil. L'adresse que vous entrez ne doit pas correspondre à l'adresse d'un appareil déjà répertorié.

Dans le cas contraire, SIMATIC Automation Tool rejette la tentative d'insertion de l'appareil et génère un message dans l'event log.

Lorsque vous entrez une adresse IP ou une adresse MAC unique, SIMATIC Automation Tool essaie de communiquer avec l'adresse que vous avez indiquée. Si la communication est établie avec succès, SIMATIC Automation Tool insère l'appareil dans la table Appareil. Si la tentative de communication échoue, SIMATIC Automation Tool vous informe que l'appareil n'existe pas dans le réseau.

Si l'appareil se trouve derrière un routeur, SIMATIC Automation Tool affiche le nom de l'appareil en bleu.

4.6 Mettre à jour les adresses IP, de sous-réseau et de passerelle

Modification des adresses IP

Pour mettre à jour l'adresse IP d'un appareil, procédez comme suit :

1. Cliquez sur l'onglet "Configurer l'adresse IP".
2. Sélectionnez un ou plusieurs appareils à inclure dans l'opération. Vous pouvez utiliser la case Appareils située en haut de la table Appareil pour sélectionner / désélectionner tous les appareils. Vous pouvez également utiliser le menu contextuel (clic droit) ou la commande de menu "Modifier > Sélectionner" pour accéder aux commandes "Sélectionner tout" et "Désélectionner tout".
3. Saisissez les adresses modifiées dans les colonnes "Nouvelle adresse IP", "Nouveau sous-réseau" et "Nouvelle passerelle". Notez que la table Appareil affiche l'interface de communication, par ex. "X1". Toutefois, vous n'entrez pas l'interface de communication lorsque vous entrez la nouvelle adresse IP. Si vous entrez une syntaxe invalide, SIMATIC Automation Tool affiche le champ en rouge.

Adresse IP	Sous-réseau	Passerelle	Mot de passe d...	Nouvelle adresse IP	Nouveau sous-réseau	Nouvelle passerelle
X1: 192.168.2.10	255.255.255.0	0.0.0.0				
X1: 192.168.2.14	255.255.255.0	0.0.0.0				

4. Sélectionnez Mettre à jour dans le menu Opérations ou cliquez sur le bouton Mettre à jour dans la barre d'outils  et sélectionnez "Configurer l'adresse IP" dans le menu déroulant du bouton.

L'opération de mise à jour définit les adresses IP, de sous-réseau et de passerelle dans les appareils sélectionnés.

Le journal sous la table Appareil montre le résultat de cette opération.

Duplicer les adresses IP

Lorsque plusieurs appareils ont la même adresse IP, les adresses sont indiquées en rouge, comme le montre l'image suivante. Vous pouvez sélectionner les appareils qui présentent des adresses IP identiques, mettre à jour les adresses IP et corriger le problème réseau. Seules les opérations suivantes sont possibles sur les appareils qui présentent des adresses IP identiques :

- Supprimer
- Configurer l'adresse IP
- Définir le nom PROFINET
- Identifier les appareils

Aucune autre opération n'est possible sur les appareils qui présentent des adresses IP identiques.

Configurer l'adresse IP Définir le nom PROFINET Mise à jour du programme Mise à jour du firmware Restaurer à partir de la sauvegarde Charger la recette				
<input type="checkbox"/>	Appareil	Type d'appareil	Numéro d'article	Adresse MAC
<input type="checkbox"/>	 S7-1200			00:1C:06:09:38:8E
<input type="checkbox"/>	 S7-1200			28:63:36:83:1A:1B
<input type="checkbox"/>	 PLC_3	CPU 1214FC DC/DC/Rly	6ES7 214-1HF40-0XB0	00:1B:1B:70:D8:EF
				X1: 192.168.2.14

Configurer l'adresse IP sur des appareils non pris en charge

Les opérations basées sur l'adresse MAC utilisent le DCP (Discovery and Configuration Protocol). Le DCP est un standard Ethernet. SIMATIC Automation Tool peut utiliser les opérations DCP basées sur l'adresse MAC "Scanner le réseau complet", "Identifier", "Mettre à jour l'adresse IP" et "Mettre à jour le nom d'appareil PROFINET" pour tous les appareils réseau directement connectés (CPU, IHM, périphériques I/O décentralisés, et autres appareils).

Sélectionnez la ligne de l'appareil non pris en charge, entrez les nouvelles données dans la colonne appropriée et mettez à jour l'adresse IP de l'appareil non pris en charge de la même manière que vous mettez à jour les appareils pris en charge.

Les appareils non pris en charge risquent de ne pas accepter une modification fondée sur la configuration matérielle de l'appareil.

Par défaut, SIMATIC Automation Tool affiche les appareils non pris en charge. Vous pouvez désactiver cette option dans Paramètres généraux (Page 109)

Remarque

Certaines informations affichées par SIMATIC Automation Tool ne peuvent pas être acquises depuis des appareils connectés derrière un routeur IP.

Par exemple, l'adresse de passerelle par défaut est acquise depuis les appareils à l'aide de DCP.

DCP n'est pas un protocole routable, et les informations ne peuvent donc pas être lues dans des appareils connectés derrière un routeur. Dans ce cas, les champs de la table Appareil correspondants sont vides.

4.7 Mettre à jour les noms d'appareils PROFINET

Règles de noms PROFINET

Les noms valides suivent les conventions classiques de dénomination du DNS (Domain Name System).

Le nombre maximal de caractères pour le nom de l'appareil est de 63. Les caractères valides sont les minuscules de "a" à "z", les chiffres 0 à 9, le tiret (signe moins) et la virgule.

Noms invalides

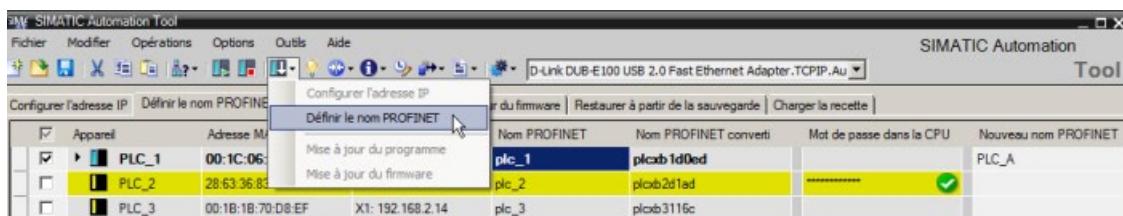
- Le nom ne doit pas avoir le format n.n.n.n, avec une valeur allant de 0 à 999.
- Le nom ne peut pas commencer par la chaîne port-nnn ou la chaîne port-nnnnnnnn, n étant un chiffre compris entre 0 et 9. Par ex., "port-123" et "port-123-45678" sont des noms invalides.
- Un nom ne peut commencer ou se terminer par un tiret "-" ou un point ".".

Modification du nom PROFINET

Cliquez sur l'onglet "Définir le nom PROFINET".

Selectionnez un ou plusieurs appareils à inclure dans l'opération. Vous pouvez utiliser la case Appareils située en haut de la table Appareil pour sélectionner / désélectionner tous les appareils. Vous pouvez également utiliser le menu contextuel (clic droit) ou la commande de menu "Modifier > Sélectionner" pour accéder aux commandes "Sélectionner tout" et "Désélectionner tout".

1. Saisissez un nouveau nom PROFINET dans la colonne "Nouveau nom PROFINET".
2. Sélectionnez Mettre à jour dans le menu Opérations ou cliquez sur le bouton Mettre à jour dans la barre d'outils  et sélectionnez "Nom PROFINET" dans le menu déroulant du bouton.



L'opération de mise à jour définit les nouveaux noms PROFINET sur les appareils sélectionnés. Si vous entrez un nom PROFINET invalide selon les règles de dénomination PROFINET, SIMATIC Automation Tool corrige le nom en nom valide. La colonne "Nom PROFINET converti" affiche le nom converti.

Le journal sous la table Appareil montre le résultat de cette opération.

Noms PROFINET identiques

SIMATIC Automation Tool indique en rouge les appareils qui présentent des noms PROFINET identiques. SIMATIC Automation Tool prend en charge la pleine fonctionnalité de ces appareils et affiche toutes les autres informations.

Configurer l'adresse IP Définir le nom PROFINET Mise à jour du programme Mise à jour du firmware Restaurer à partir de la sauvegarde Charger la recette					
<input type="checkbox"/>	Appareil	Adresse MAC	Adresse IP	Nom PROFINET	Nom PROFINET converti
<input type="checkbox"/>	▶ PLC_1	00:1C:06:09:38:8E	X1: 192.168.2.10	plc_1	plcxb1d0ed
<input type="checkbox"/>	PLC_1	28:63:36:83:1A:1B	X1: 192.168.2.12	plc_1	plcxb1d0ed

Définir le nom PROFINET sur des appareils non pris en charge

Les opérations basées sur l'adresse MAC utilisent le DCP (Discovery and Configuration Protocol). Le DCP est un standard Ethernet. SIMATIC Automation Tool peut utiliser les opérations DCP basées sur l'adresse MAC "Scanner le réseau complet", "Identifier", "Mettre à jour l'adresse IP" et "Mettre à jour le nom d'appareil PROFINET" pour tous les appareils réseau directement connectés (CPU, IHM, périphériques I/O décentralisés, et autres appareils).

Sélectionnez la ligne de l'appareil non pris en charge, entrez les nouvelles données dans la colonne appropriée et mettez à jour le nom PROFINET de l'appareil non pris en charge, de la même manière que vous mettez à jour les appareils pris en charge.

Les appareils non pris en charge risquent de ne pas accepter une modification fondée sur la configuration matérielle de l'appareil.

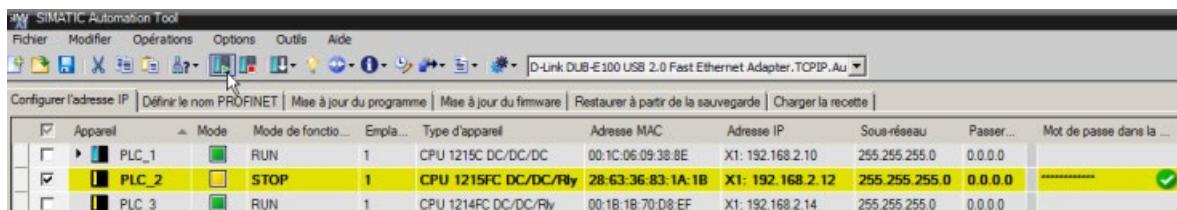
Par défaut, SIMATIC Automation Tool affiche les appareils non pris en charge. Vous pouvez désactiver cette option dans Paramètres généraux (Page 109)

4.8 Placement des CPU en mode RUN ou STOP (Marche/arrêt)

Bascule des CPU en mode RUN ou STOP

Pour modifier l'état de fonctionnement d'un appareil, procédez comme suit :

1. Sélectionnez un ou plusieurs appareils à inclure dans l'opération. Vous pouvez utiliser la case Appareils située en haut de la table Appareil pour sélectionner / désélectionner tous les appareils. Vous pouvez également utiliser le menu contextuel (clic droit) ou la commande de menu "Modifier > Sélectionner" pour accéder aux commandes "Sélectionner tout" et "Désélectionner tout".
2. Pour chaque CPU sélectionnée, entrez un mot de passe, le cas échéant, dans la colonne "Mot de passe dans la CPU" de l'onglet ouvert.
3. Définissez le mode de fonctionnement, en choisissant entre mode RUN (MARCHE) et mode STOP (ARRÊT) :
 - Dans le menu Opérations, sélectionnez RUN ou cliquez sur le bouton RUN  de la barre d'outils. Pour que la CPU passe en mode RUN, il faut qu'un programme valide existe.
 - Dans le menu Opérations, sélectionnez STOP ou cliquez sur le bouton STOP  de la barre d'outils.



Appareil	Mode	Mode de fonctionnement	Empla...	Type d'appareil	Adresse MAC	Adresse IP	Sous-réseau	Passer...	Mot de passe dans la ...
PLC_1		RUN	1	CPU 1215C DC/DC/DC	00:1C:06:09:38:8E	X1: 192.168.2.10	255.255.255.0	0.0.0.0	-----
PLC_2		STOP	1	CPU 1215FC DC/DC/Rly	28-63-36-83-1A-1B	X1: 192.168.2.12	255.255.255.0	0.0.0.0	-----
PLC_3		RUN	1	CPU 1214FC DC/DC/Rly	00:1B:1B:70:D8:EF	X1: 192.168.2.14	255.255.255.0	0.0.0.0	-----

SIMATIC Automation Tool configure les CPU sélectionnées en mode RUN ou STOP.

Les colonnes Mode et État de fonctionnement sur la table Appareil indiquent l'état actuel de la CPU. Jaune représente le mode STOP. La couleur verte représente le mode RUN. La couleur rouge signifie que la CPU est défaillante.

Le journal des événements (Event log) sous la table Appareil montre le résultat de cette opération.

4.9 Identifier les appareils

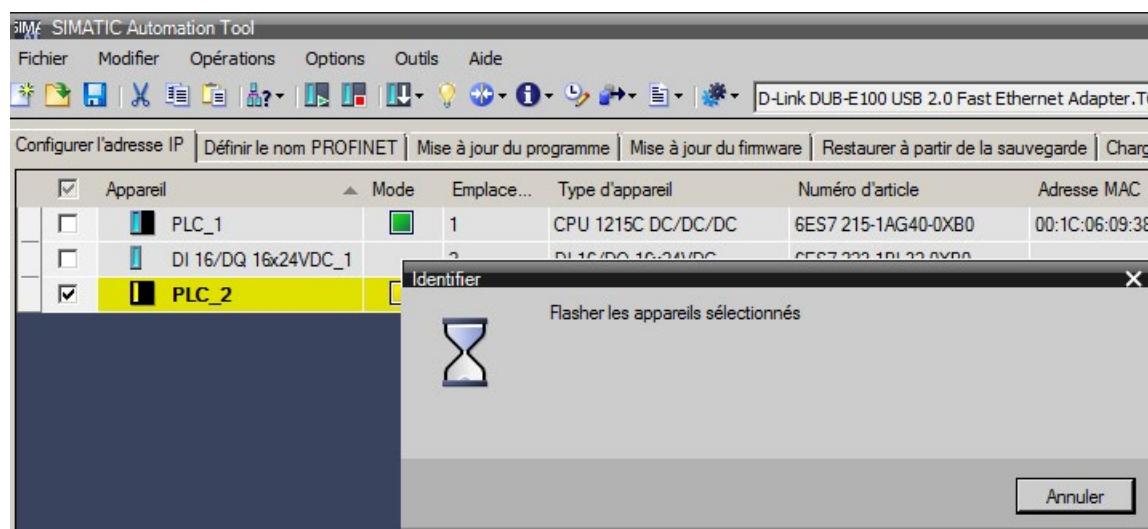
Localiser un appareil en faisant clignoter les LED ou les écrans IHM.

L'opération Identifier vous aide à localiser physiquement les appareils dans la table Appareil. Vous pouvez utiliser l'opération Identifier en mode RUN et en mode STOP. Procédez comme suit pour identifier les appareils :

1. Sélectionnez un ou plusieurs appareils à inclure dans l'opération. Vous pouvez utiliser la case Appareils située en haut de la table Appareil pour sélectionner / désélectionner tous les appareils. Vous pouvez également utiliser le menu contextuel (clic droit) ou la commande de menu "Modifier > Sélectionner" pour accéder aux commandes "Sélectionner tout" et "Désélectionner tout".
2. Sélectionnez la commande de menu "Opérations > Identifier" ou cliquez sur le bouton  de la barre d'outils pour identifier les appareils sélectionnés.

Les LED des appareils CPU sélectionnés clignotent pour faciliter leur localisation. Les appareils IHM sélectionnés effectuent un flash d'écran.

Le clignotement continue jusqu'à ce que vous appuyiez sur le bouton Annuler.



Identifier des appareils non pris en charge (LED / écrans IHM clignotants)

Les opérations basées sur l'adresse MAC utilisent le DCP (Discovery and Configuration Protocol). Le DCP est un standard Ethernet. SIMATIC Automation Tool peut utiliser les opérations DCP basées sur l'adresse MAC "Scanner le réseau complet", "Identifier", "Mettre à jour l'adresse IP" et "Mettre à jour le nom d'appareil PROFINET" pour tous les appareils réseau directement connectés (appareils PROFINET pris en charge, appareils PROFINET non pris en charge, et appareils Ethernet de tiers non pris en charge).

Sélectionnez la ligne de l'appareil non pris en charge et identifiez un appareil non pris en charge, de la même manière que vous identifiez les appareils pris en charge.

Les appareils non pris en charge risquent de ne pas activer leurs LED en fonction de la configuration matérielle de l'appareil.

Par défaut, SIMATIC Automation Tool affiche les appareils non pris en charge. Vous pouvez désactiver cette option dans Paramètres généraux (Page 109)

4.10 Mettre à jour les programmes des appareils depuis SIMATIC Automation Tool

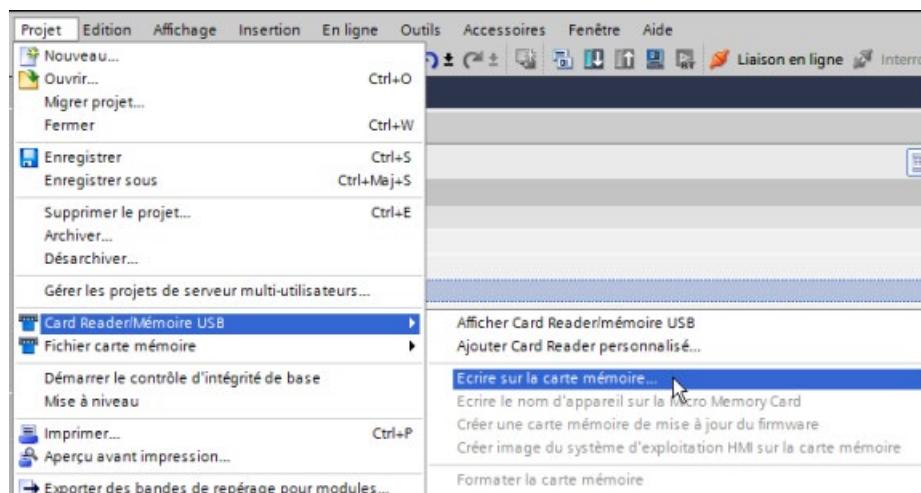
Avant de pouvoir transférer un programme sur une CPU à l'aide de SIMATIC Automation Tool, vous devez avoir accès au programme sur l'un des supports suivants :

- une carte mémoire SIMATIC
- un lecteur flash USB
- un disque dur de votre PG / PC

Préparation d'un programme CPU pour une utilisation avec SIMATIC Automation Tool

Pour transférer un projet TIA Portal sur CPU vers une carte mémoire SIMATIC, procédez comme suit :

1. Insérez une carte mémoire SIMATIC dans le lecteur de carte de votre PG/PC.
2. À partir de STEP 7, sélectionnez la CPU dans le navigateur du projet.
3. Sélectionner la commande de menu "Projet > Card Reader/Mémoire USB > Écrire sur la carte mémoire".



4. Sélectionnez votre carte mémoire dans la boîte de dialogue.



STEP 7 enregistre un dossier SIMATIC.S7S sur votre carte mémoire SIMATIC, qui contient votre projet de CPU. Vous pouvez également copier le projet STEP 7 vers la carte mémoire en faisant glisser le projet vers la carte mémoire dans l'arborescence de projet.

Pour plus d'informations, reportez-vous au Système d'information STEP 7 (aide en ligne).

Une fois que TIA Portal a transféré les données de programme sur un support de stockage, vous pouvez utiliser l'Explorateur de fichiers Windows pour transférer le programme dans le dossier utilisé par SIMATIC Automation Tool.

Copie du dossier "SIMATIC.S7S" pour chaque programme de CPU

Suivez les étapes suivantes pour que SIMATIC Automation Tool puisse accéder à un programme de CPU.

1. Créer des sous-dossiers sous Dossier de mise à jour du programme (Page 114). Créez un dossier par programme et créez un nom de dossier qui identifie le programme. Les noms des dossiers que vous créez sont affichés dans la liste déroulante de SIMATIC Automation Tool.
2. Utilisez l'Explorateur Windows pour copier le dossier "SIMATIC.S7S" (avec tous les sous-dossiers et fichiers) dans chaque sous-dossier pour chaque programme. Vous pouvez placer un programme TIA Portal (un dossier "SIMATIC.S7S") dans un fichier d'archive compressé et l'extraire vers l'emplacement de votre sous-dossier. Notez qu'il s'agit d'une opération distincte de celle consistant à mettre à jour les données de recette (Page 73).

Voir plus loin la partie "Exemple de mise à jour du programme de CPU".

Remarque

Données du programme TIA Portal

Les données du programme sont protégées. Il n'est pas possible de détecter certaines informations telles que le nom du projet ou la CPU cible d'un programme TIA Portal à partir des données enregistrées dans un dossier SIMATIC.S7S. Vous ne pouvez pas distinguer un dossier SIMATIC.S7S d'un autre dossier SIMATIC.S7S.

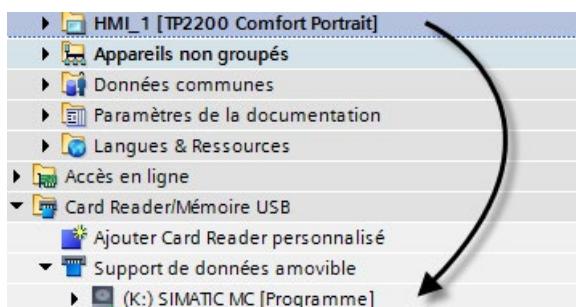
Vous devez créer et nommer les sous-dossiers sous le dossier SIMATIC Automation Tool de mise à jour du programme (Page 114) qui identifie la fonction du programme ou la CPU cible. Copiez un dossier SIMATIC.S7S du programme dans le sous-dossier que vous avez nommé. Les noms des sous-dossiers que vous créez sont affichés sur la liste déroulante "Programme" de SIMATIC Automation Tool et indiquent le chemin d'accès vers le dossier SIMATIC.S7S correct.

Préparer le système d'exploitation et le logiciel exécutif d'un appareil IHM en vue de leur utilisation avec SIMATIC Automation Tool

Les appareils IHM des versions 14 et supérieures prennent en charge la sauvegarde du système d'exploitation et du logiciel d'exécution depuis STEP 7.

Pour copier le système d'exploitation et les fichiers d'exécution d'un appareil IHM sur une carte mémoire SIMATIC, procédez comme suit :

1. Insérez une carte mémoire SIMATIC dans le lecteur de carte de votre PG/PC.
2. Développez "Card Reader/Mémoire USB" dans l'arborescence du projet pour afficher le lecteur correspondant à votre lecteur de carte.
3. Sélectionnez votre IHM dans l'arborescence du projet et faites-le glisser vers la lettre de lecteur de votre lecteur de carte.



STEP 7 enregistre un dossier SIMATIC.HMI sur votre carte mémoire SIMATIC, qui contient votre logiciel d'exécution IHM et votre système d'exploitation IHM. Les mises à jour IHM comprennent le système d'exploitation et les données de runtime. Vous ne pouvez pas choisir une mise à jour partielle.

Une fois que TIA Portal a transféré le dossier SIMATIC.HMI sur un support de stockage, utilisez l'Explorateur Windows pour rendre le dossier SIMATIC.HMI accessible à SIMATIC Automation Tool :

- Créez un sous-dossier pour le dossier IHM dans le dossier Mise à jour du programme (Page 114).
- Copiez le dossier SIMATIC.HMI dans le sous-dossier.

Mettre à jour le programme de CPU ou le système d'exploitation et le logiciel runtime d'un appareil IHM

Si votre réseau présente une topologie en chaîne et que les Paramètres de communication (Page 110) activent le traitement multi-thread (multifil), prenez garde au risque d'interruption de communication lors de cette opération.

Remarque

Appareils de sécurité

Si la CPU de sécurité est protégée, vous devez entrer le mot de passe de CPU F de sécurité dans la colonne "Mot de passe dans la CPU" pour mettre à jour le programme dans un appareil de sécurité.

Vous devez confirmer une invite supplémentaire pour les mises à jour de programme pour les CPU F et sélectionner de nouveau votre appareil dans les conditions suivantes :

- Vous mettez à jour un programme de sécurité (Page 124) avec un autre programme de sécurité
- Vous mettez à jour un programme de sécurité avec un programme standard
- Vous chargez un programme de sécurité pour la première fois
- Vous chargez un programme standard qui nécessite le mot de passe de la CPU pour le niveau d'accès "Accès complet, y compris failsafe (pas de protection)".

SIMATIC Automation Tool place les demandes "Mettre à jour le programme" pour les appareils de sécurité dans la file d'attente d'opérations sur CPU F relatives à la sécurité. SIMATIC Automation Tool utilise uniquement le traitement séquentiel à thread unique pour la file d'attente d'opérations de sécurité.

L'appareil de destination pour un programme de sécurité doit être une CPU de sécurité.

ATTENTION

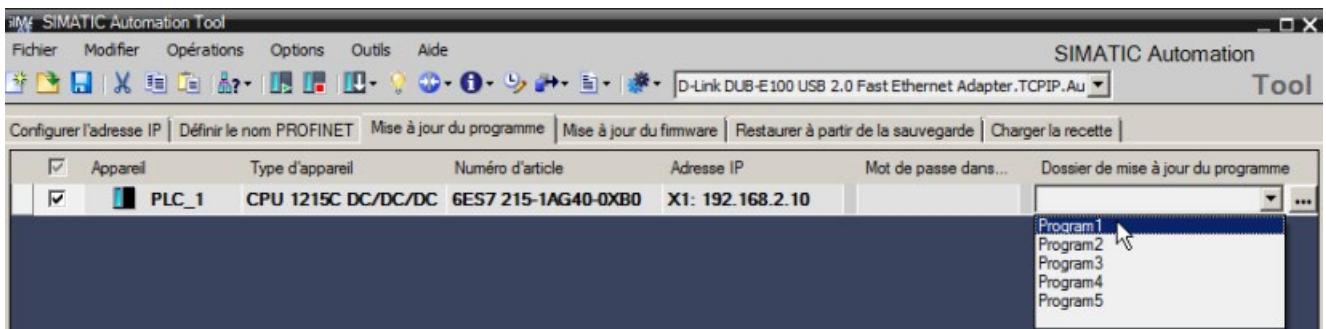
Vérifiez qu'aucun appareil ne soit en train d'exécuter un processus avant de mettre à jour le programme.

L'installation d'un nouveau programme fait passer les CPU à l'arrêt, ce qui peut affecter l'exécution d'un processus ou d'une machine en ligne, dont la mise en route inattendue peut entraîner la mort, des blessures graves et des dommages matériels importants.

4.10 Mettre à jour les programmes des appareils depuis SIMATIC Automation Tool

Une fois que vous avez stocké les données du programme dans le dossier de mise à jour, vous pouvez utiliser SIMATIC Automation Tool pour charger de nouveaux programmes sur un ou plusieurs appareils. Procédez comme suit pour mettre à jour le programme :

1. Cliquez sur l'onglet "Mettre à jour le programme".
2. Sélectionnez un ou plusieurs appareils à inclure dans l'opération. Vous pouvez utiliser la case Appareils située en haut de la table Appareil pour sélectionner / désélectionner tous les appareils. Vous pouvez également utiliser le menu contextuel (clic droit) ou la commande de menu "Modifier > Sélectionner" pour accéder aux commandes "Sélectionner tout" et "Désélectionner tout".
3. Pour chaque appareil sélectionné, utilisez la colonne "Dossier de mise à jour du programme" de la liste déroulante pour choisir un nom de dossier. La liste déroulante affiche les dossiers que vous avez créés sous le chemin d'accès vers la mise à jour des programmes.

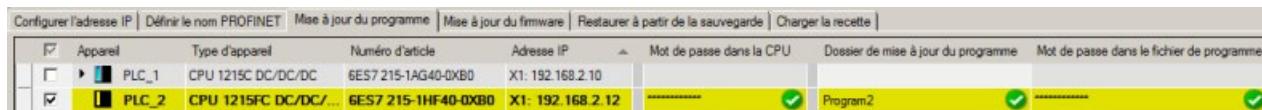


Vous pouvez aussi utiliser le bouton de navigation et aller au dossier dans lequel vous avez stocké un programme sur votre PG/PC. Lorsque vous sélectionnez un programme, SIMATIC Automation Tool l'ajoute à la liste déroulante. Si le fichier sélectionné porte le même nom qu'un fichier de la liste, SIMATIC Automation Tool ajoute un chiffre au nouveau fichier afin que le nom reste unique. Pour vous aider à identifier les fichiers, lorsque vous faites passer votre curseur sur le nom du programme dans le champ Mettre à jour le programme, une info-bulle affiche les informations suivantes concernant l'appareil et le programme sélectionné :

Interface : X1
IP actuelle : 192.168.2.12
IP du programme : 192.168.2.12
Sous-réseau du programme : 255.255.255.0
Passerelle du programme : 0.0.0.0

Si le fichier de programme ne contient pas d'adresse IP, l'info-bulle affiche "Défini directement sur l'appareil" pour tous les champs d'adresse IP.

4. Le cas échéant, entrez les mots de passe dans les colonnes "Mot de passe dans la CPU" et "Mot de passe dans le fichier de programme". L'opération de mise à jour du programme est liée à la sécurité. S'il s'agit d'un appareil de sécurité, vous devez entrer le mot de passe de CPU F de sécurité.



Appareil	Type d'appareil	Numéro d'article	Adresse IP	Mot de passe dans la CPU	Dossier de mise à jour du programme	Mot de passe dans le fichier de programme
<input type="checkbox"/> PLC_1	CPU 1215C DC/DC/DC	6ES7 215-1AG40-0XB0	X1: 192.168.2.10	-----	<input checked="" type="checkbox"/> Program2	<input checked="" type="checkbox"/> -----
<input checked="" type="checkbox"/> PLC_2	CPU 1215FC DC/DC/...	6ES7 215-1HF40-0XB0	X1: 192.168.2.12	-----	<input checked="" type="checkbox"/> Program2	<input checked="" type="checkbox"/> -----

5. Sélectionnez la commande de menu "Opérations > Mettre à jour > Mettre à jour le programme" pour démarrer l'opération. Vous pouvez également sélectionner le bouton de barre d'outils Mettre à jour  et sélectionner "Mettre à jour le programme" dans le menu déroulant du bouton.

L'Event Log sous la table Appareil montre le résultat de cette opération.

Validation du programme

SIMATIC Automation Tool vérifie les données du programme, avant de mettre à jour le programme dans une CPU.

S'il y a une erreur dans les données du programme, une icône "X" rouge s'affiche dans la cellule "Dossier de mise à jour du programme". D'autres messages d'erreur sont disponibles dans une info-bulle lorsque vous passez votre curseur sur la cellule.

Gestion des mots de passe après l'opération de mise à jour du programme

Un fichier programme peut avoir un mot de passe, qui peut être différent du mot de passe actuel de la CPU. Lorsqu'un fichier programme comporte un mot de passe, vous devez entrer le mot de passe du programme dans la cellule "Mot de passe dans le fichier de programme" pour effectuer une mise à jour du programme. Le mot de passe du programme devient le mot de passe de la CPU une fois que l'opération Mise à jour du programme est terminée.

Une fois que l'opération Mise à jour du programme est terminée avec succès, SIMATIC Automation Tool copie automatiquement le Mot de passe dans le fichier de programme vers le champ de mot de passe de la CPU et effectue une tentative de connexion à l'aide du nouveau mot de passe. SIMATIC Automation Tool efface alors le champ Mot de passe dans le fichier de programme et le champ Dossier de mise à jour du programme.

Si le mot de passe que vous entrez dans la colonne "Mot de passe dans le fichier de programme" n'est pas le mot de passe configuré pour le projet dans TIA Portal, l'Event Log affiche un avertissement une fois l'opération terminée. Dans ce cas, le mot de passe de la CPU affiche une icône 'X' rouge qui indique un mot de passe non-valide.

Validation de la signature de sécurité

Un projet TIA Portal qui contient un programme de sécurité comporte une signature de sécurité, qui sert à vérifier les données dans un programme copié et renforce le niveau de sécurité pour des programmes de sécurité. Après une opération de mise à jour du programme, la signature de sécurité dans le projet est comparée à la signature de sécurité qui vient d'être chargée sur l'appareil CPU.

Lorsqu'elle est réussie, la comparaison est indiquée comme suit dans le journal des événements : "Les résultats de la comparaison CRC en ligne et des signatures de sécurité collectives hors ligne correspondent."

Lorsqu'elle échoue, la comparaison est indiquée comme suit dans l'Event Log par SIMATIC Automation Tool : "Les résultats de la comparaison CRC en ligne et des signatures de sécurité collectives hors ligne ne correspondent pas.". Dans ce cas, restaurez les paramètres d'usine de l'appareil (Page 89) et réessayer de mettre à jour le programme.

ATTENTION

Assurez-vous de charger le bon programme de sécurité.

Le fait de faire fonctionner le mauvais programme sur une CPU F peut affecter le fonctionnement d'un processus ou d'une machine. Le démarrage inattendu d'un processus ou d'une machine peut entraîner la mort ou des dommages corporels et/ou matériels.

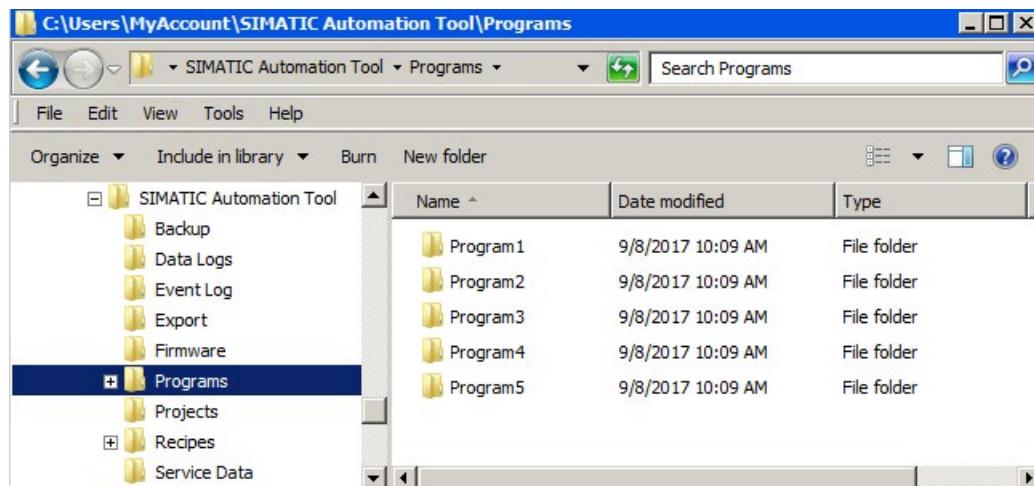
N'essayez pas de passer en mode RUN si vous n'êtes pas certain(e) d'avoir chargé le bon programme.

Exemple : Mise à jour du programme

Si vous voulez disposer de cinq programmes de CPU différents pour la Mise à jour du programme, vous devez créer et nommer cinq dossiers dans le Dossier de mise à jour du programme (Page 112). Copiez tous les dossiers "SIMATIC.S7S" dans les cinq dossiers correspondants.

Dans l'exemple, les noms de dossier "Program1", "Program2", "Program3", "Program4" et "Program5" identifient les programmes disponibles. Vous pouvez choisir un nom de dossier quelconque. Le nom peut renvoyer à une fonction du programme ou à un emplacement sur la CPU.

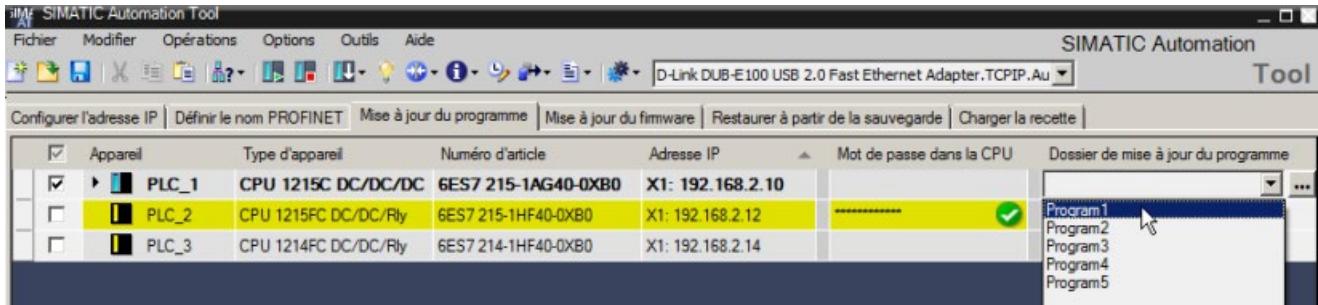
L'image suivante montre la vue de l'Explorateur Windows avec les quatre sous-dossiers du dossier Programmes. Vous copiez les dossiers programme SIMATIC.S7S correspondants dans ces dossiers.



L'image suivante montre l'onglet Mettre à jour le programme de SIMATIC Automation Tool avec les exemples de noms de dossiers dans la liste déroulante de la colonne "Mettre à jour le nouveau programme". Vous devez utiliser la liste déroulante dans la colonne "Mettre à jour le nouveau programme" pour définir le programme à utiliser. Si plusieurs lignes de CPU sont sélectionnées, vous devez répéter la procédure et affecter le bon programme à chaque CPU sélectionnée.

4.10 Mettre à jour les programmes des appareils depuis SIMATIC Automation Tool

Sélectionnez la commande de menu "Opérations > Mettre à jour > Mettre à jour le programme" pour lancer la mise à jour du programme. Vous pouvez également cliquer sur le bouton de barre d'outils Mettre à jour et sélectionner "Mettre à jour le programme" dans le menu déroulant du bouton.



Le processus est similaire pour les données IHM. Le nom du dossier dans un dossier de projet est "SIMATIC.HMI" au lieu de "SIMATIC.S7S". La procédure est la même.

Règles de mise à jour du programme de CPU

SIMATIC Automation Tool prend en charge l'opération de mise à jour du programme pour les CPU standard et les CPU de sécurité.

Règles de mise à jour du programme :

- La version de firmware du matériel de la CPU doit être supérieure ou égale à celle du projet à télécharger.
Vous pouvez contourner cette restriction en mettant à jour le firmware dans la CPU si cela est possible.
- Pour les CPU S7-1200, S7-1500 et ET 200SP (S7-1500), SIMATIC Automation Tool prend en charge l'opération de mise à jour du programme si la version du firmware de la CPU affectée au projet est prise en charge comme indiqué dans les tableaux suivants.

Table de prise en charge de la mise à jour du programme

- La mise à jour du programme est possible si ✓ est affiché.
- La mise à jour du programme n'est pas possible si une cellule vide est affichée.

CPU S7-1500 ET 200SP		Version de firmware de CPU configurée dans le projet pour mise à jour de la CPU						
		1.0	1.1	1,5	1.6	1.7	1.8	2.0
Version de firmware de la CPU cible	1.0	✓						
	1.1	✓	✓					
	1.5	✓	✓	✓				
	1.6	✓	✓	✓	✓			
	1.7	✓	✓	✓	✓	✓		
	1.8	✓	✓	✓	✓	✓	✓	
	2.0	✓	✓	✓	✓	✓	✓	✓
	2.1	✓	✓	✓	✓	✓	✓	✓

Utilisez TIA Portal pour changer la version de la CPU du projet S7-1200 en une version prise en charge, afin d'assurer le succès de la mise à jour du programme.

S7-1200		Version de firmware de CPU configurée dans le projet pour mise à jour de la CPU						
		2.0	2.1	2.2	3.0	4.0	4.1	4.2
Version de firmware de la CPU cible	2.0	✓						
	2.1	✓	✓					
	2.2	✓	✓	✓				
	3.0	✓	✓	✓	✓			
	4.0					✓		
	4.1					✓	✓	
	4.2					✓	✓	✓

4.11 Charger, ajouter, remplacer et supprimer des recettes dans les CPU

Les opérations de recette fonctionnent pour les CPU qui possèdent des recettes dans une mémoire de chargement externe (carte mémoire SIMATIC). La CPU peut être à l'état RUN ou à l'état STOP.

Les données de recette sont des fichiers au format .CSV (valeurs séparées par des virgules).

Vous pouvez ajouter ou remplacer plusieurs recettes en une seule opération si elles se trouvent toutes sur différentes CPU. Vous ne pouvez pas ajouter ni remplacer plus d'une recette dans une CPU en une seule opération.

Vous pouvez sélectionner plusieurs recettes à charger ou supprimer dans une CPU en une seule opération.

Lorsque vous sélectionnez un fichier de recette ou un dossier de recette, SIMATIC Automation Tool affiche en **gras** la ligne de l'appareil dans la table Appareil.

Pour le téléchargement, cliquez sur l'onglet Recette et sélectionnez un dossier Recette.

SIMATIC Automation Tool crée un nom de dossier unique pour chaque CPU dans lequel sont stockés les fichiers de recette chargés sur le PG/PC. Le nom de fichier comprend le nom de la CPU et l'adresse MAC. Si vous sélectionnez et chargez le même fichier de recette une deuxième fois, SIMATIC Automation Tool ajoute un chiffre au nom pour conserver l'univocité des noms de fichiers.

SIMATIC Automation Tool doit avoir un accès en lecture pour charger les fichiers de recette. SIMATIC Automation Tool doit avoir un accès complet (lecture et écriture) pour supprimer, ajouter, ou remplacer les fichiers recette. Vous aurez peut-être besoin de saisir un mot de passe pour pouvoir effectuer une opération de suppression ou d'ajout / de remplacement. Si vous n'entrez pas de mot de passe ou si le mot de passe ne permet pas l'accès en écriture sur la CPU, l'opération va échouer pour cette CPU et un message d'erreur sera entré dans le journal des événements (Event Log).

Actions de la barre d'outils

Dans le menu déroulant du bouton de barre d'outils "Opérations sur fichier" , vous avez les choix suivants :

Charger les recettes : copie les fichiers recette sélectionnés d'une CPU vers le dossier recette (Page 117) de votre PG/PC.

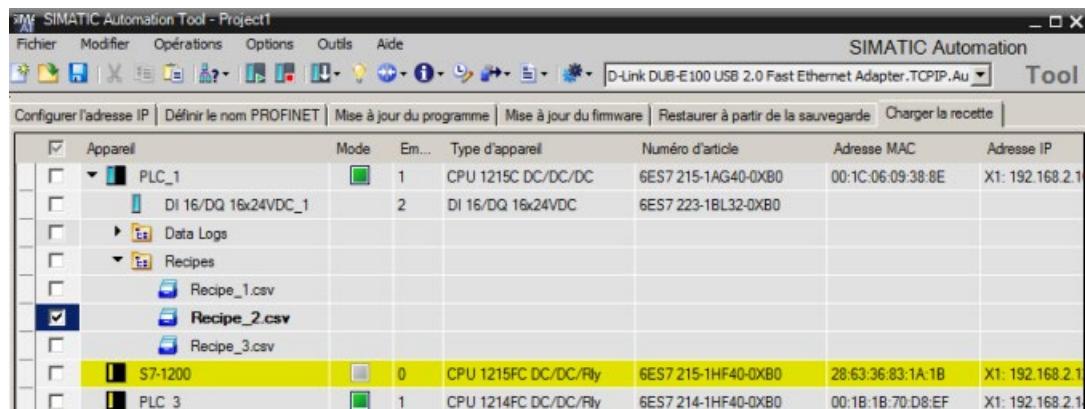
Ajouter/remplacer une recette (choix possible uniquement lorsque l'onglet Charger les recettes est affiché) : copie les fichiers recette sélectionnés de votre PG/PC vers une CPU.

Supprimer les recettes : supprime les fichiers recette sélectionnés d'une CPU.

Charger ou supprimer les fichiers recette

Pour charger ou supprimer les fichiers recette d'une CPU, procédez comme suit :

1. Cliquez sur l'onglet "Charger la recette" sur la table Appareil.
2. Développez une ligne d'appareil CPU et affichez un dossier recette .
3. Développez un dossier recette et sélectionnez les fichiers recette  à inclure dans l'opération.
4. Pour chaque CPU sélectionnée, entrez un mot de passe, le cas échéant, dans la colonne "Mot de passe dans la CPU".
5. Sélectionnez la commande de menu "Opérations > Opérations sur fichier" ou cliquez sur le bouton  de barre d'outils "Opérations sur fichier" :
6. Sélectionnez soit "Charger les recettes" soit "Supprimer les recettes" dans le menu "Opérations sur fichier".

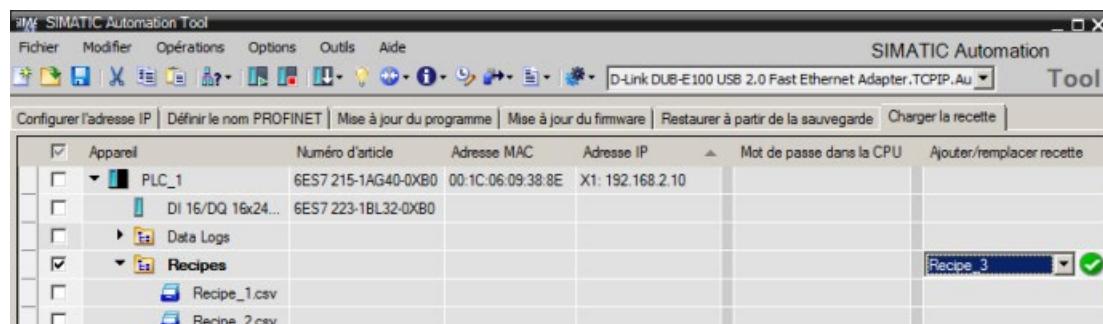


Le journal des événements (Event Log) sous la table Appareil montre le résultat de cette opération.

Ajouter ou remplacer (télécharger) des fichiers recette

Pour ajouter ou remplacer des fichiers de recette CPU parmi les fichiers présents sur votre PG/PC, procédez comme suit :

1. Copiez les fichiers de données de recette .csv que vous souhaitez ajouter ou remplacer dans une CPU vers le dossier Recettes. Dossier recette. (Page 117)
2. Cliquez sur l'onglet "Charger les recettes" dans la table Appareil.
3. Développez une ligne d'appareil CPU et affichez un dossier recette .
4. Sélectionnez le dossier recette que vous souhaitez charger.
5. Pour chaque dossier recette sélectionnée, cliquez sur la liste déroulante dans la colonne "Ajouter/remplacer une recette" et sélectionnez un nom de fichier de recette. La liste déroulante affiche les noms des fichiers .csv qui existent dans le chemin de répertoire défini dans la boîte de dialogue "Options > Paramètres".
Vous pouvez également utiliser le bouton de navigation  et accéder au dossier contenant les fichiers recette stockés sur votre PG/PC. SIMATIC Automation Tool ajoute le fichier sélectionné à la liste déroulante. Si le fichier sélectionné porte le même nom qu'un fichier de la liste, SIMATIC Automation Tool ajoute un chiffre au nouveau fichier afin que le nom reste univoque. Pour vous aider à identifier les fichiers, une info-bulle affiche le chemin d'accès complet et le nom du fichier.



6. Sélectionnez la commande de menu "Opérations > Opérations sur fichier" ou cliquez sur le bouton  de barre d'outils "Opérations sur fichier" :
7. Sélectionnez soit "Ajouter/remplacer une recette" dans le menu Opérations sur fichier. S'il s'agit d'une recette existante, SIMATIC Automation Tool la remplace. S'il ne s'agit pas d'une recette existante, SIMATIC Automation Tool l'ajoute.

L'Event Log sous la table Appareil montre le résultat de cette opération.

Une fois que l'opération de téléchargement de recette est terminée avec succès, le chemin du fichier recette est supprimé.

 **ATTENTION**

Note de sécurité

Le fait de faire fonctionner un processus ou une machine avec des données corrompues peut affecter l'exécution d'un processus ou le fonctionnement d'une machine en ligne. Le démarrage inattendu d'un processus ou d'une machine peut entraîner la mort ou des dommages corporels et/ou matériels.

Assurez-vous de protéger les fichiers de recette .csv contre une éventuelle compromission par l'utilisation de différentes méthodes, notamment en limitant l'accès au réseau ou en utilisant des pare-feu.

4.12 Charger et supprimer des journaux de données (Data logs) dans les CPU

L'opération de chargement du Data Log fonctionne pour les CPU qui possèdent des Data Logs dans une mémoire de chargement externe (carte mémoire SIMATIC). La CPU peut être à l'état RUN ou à l'état STOP.

Pour les CPU, l'opération de suppression d'un journal de données est possible uniquement en mode STOP. Si vous sélectionnez un ou plusieurs Data logs à supprimer sur une ou plusieurs CPU et si l'une d'entre elles est en mode RUN, un message vous signale que toutes les CPU doivent être en mode STOP avant de lancer l'opération. Si vous décidez de ne pas basculer en mode STOP, la procédure complète de suppression est annulée.

Les Data logs sont chargés sous forme de fichiers texte au format .CSV (valeurs séparées par des virgules).

Vous pouvez sélectionner plusieurs fichiers de données sur une ou plusieurs CPU et traiter tous les fichiers sélectionnés en une seule fois.

SIMATIC Automation Tool crée un nom de dossier unique pour chaque CPU, dans lequel sont stockés les fichiers de Data log chargés sur votre PG/PC. Le nom de fichier est composé à partir du nom de la CPU et de l'adresse MAC. Si vous sélectionnez et chargez le même fichier Data log une deuxième fois, un chiffre est ajouté au nom pour conserver l'unicité.

SIMATIC Automation Tool doit avoir un accès en lecture pour charger les fichiers Data log et un accès complet (lecture et écriture) pour supprimer les fichiers Data log dans une CPU. Vous aurez sans doute besoin de saisir un mot de passe pour la suppression. Si vous n'entrez pas de mot de passe ou si le mot de passe ne permet pas l'accès en écriture sur la CPU, la procédure échoue et un message d'erreur est entré dans le journal des opérations.

Actions relatives au Data Log

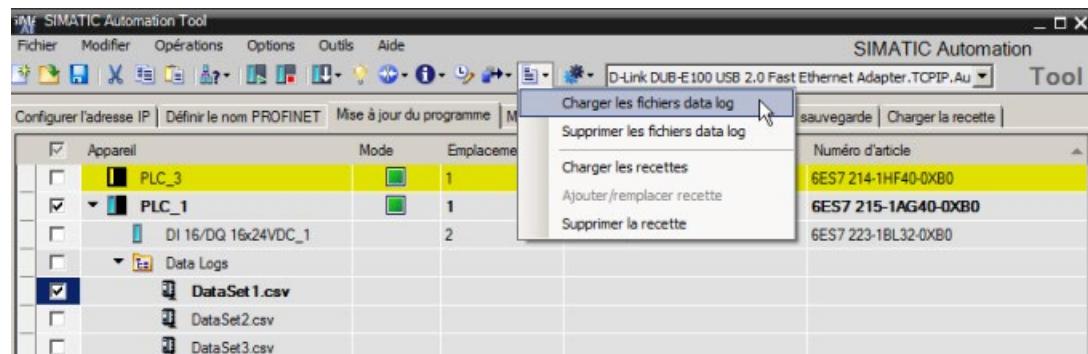
Le bouton de barre d'outils "Opérations sur fichier"  et le menu Opérations sur les fichiers proposent les commandes de menu suivantes :

- **Charger les Data logs** : charge une copie des fichiers Data log sélectionnés de la CPU vers le PG/PC. SIMATIC Automation Tool copie les fichiers sous le répertoire défini dans Paramètres des journaux de données (Data logs) (Page 118).
- **Supprimer les Data logs** : supprime les fichiers Data log sélectionnés enregistrés dans une CPU.

Charger ou supprimer des fichiers Data log

Pour charger ou supprimer des fichiers Data log, procédez comme suit :

1. Développez une ligne de CPU et affichez tout dossier Data log .
2. Développez un dossier Data log et sélectionnez les fichiers Data log : .
3. Pour chaque CPU, entrez un mot de passe, le cas échéant, dans la colonne "Mot de passe dans la CPU" de l'onglet ouvert.
4. Sélectionnez la commande de menu "Opérations > Opérations sur fichier > Charger les Data logs" ou la commande de menu "Opérations > Opérations sur fichier > Supprimer les Data logs". Vous pouvez également cliquer sur le bouton de barre d'outils "Opérations sur les fichiers"  et sélectionner la commande "Charger les Data logs" ou "Supprimer les Data logs" dans le menu déroulant du bouton.



L'Event log sous la table Appareil montre le résultat de cette opération.

ATTENTION

Note de sécurité

Le fait de faire fonctionner un processus ou une machine avec des données corrompues peut affecter l'exécution d'un processus ou le fonctionnement d'une machine en ligne. Le démarrage inattendu d'un processus ou d'une machine peut entraîner la mort ou des dommages corporels et/ou matériels.

Assurez-vous de protéger les fichiers Data log .csv contre une éventuelle compromission par l'utilisation de différentes méthodes, notamment en limitant l'accès au réseau ou en utilisant des pare-feu.

4.13 Installation de nouveaux firmware dans les appareils

Mise à jour du firmware TIA Portal

Vous pouvez utiliser une carte mémoire SIMATIC pour installer des mises à jour de firmware dans des appareils. Vous pouvez également mettre le firmware à jour en utilisant la page d'informations sur les modules d'un serveur Web intégré à la CPU ou des fonctions en ligne et de diagnostic de TIA Portal.

Remarque

Mises à jour du système d'exploitation et du logiciel exécutif d'appareils IHM

Vous devez utiliser l'opération Mise à jour du programme (Page 63) de SIMATIC Automation Tool pour mettre à jour le système d'exploitation et le logiciel exécutif d'appareils IHM. Vous ne pouvez pas choisir une mise à jour partielle. Si nécessaire, l'opération Mise à jour du programme met à jour tous les composants de données afin d'assurer un téléchargement cohérent.

Mise à jour du firmware dans SIMATIC Automation Tool

SIMATIC Automation Tool peut mettre à jour le firmware d'un groupe d'appareils. Vous pouvez utiliser le nouveau fichier unique au format .upd et l'ancien format (classique) qui utilise trois ou plus fichiers .upd séparés.

Si votre réseau présente une topologie de communication en chaîne et que les Paramètres de communication (Page 110) activent le traitement multi-thread (multifil), prenez garde au risque d'interruption de communication lors de cette opération.

Remarque

Les modules de communication S7-1200 doivent être configurés avant toute mise à jour du firmware.

Vous pouvez utiliser SIMATIC Automation Tool pour mettre à jour le firmware dans ces modules SM et CM configurés et non configurés, à l'exception des appareils S7-1200 CM gauches. Pour un module CM S7-1200, vous devez configurer le module CM dans TIA Portal et télécharger la configuration dans le module avant de pouvoir utiliser SIMATIC Automation Tool pour mettre à jour le firmware du module CM.

Remarque

Mise à niveau inférieur du firmware de la CPU

Vous pouvez utiliser SIMATIC Automation Tool pour rétrograder le firmware de la CPU (chargement d'une version antérieure), mais l'adresse IP et le programme risquent d'être écrasés. Dans ce cas, l'adresse IP est restaurée à 0.0.0.0 et il faut lancer un nouveau scan du réseau pour communiquer avec cet appareil. Vous devez définir l'adresse IP pour rétablir l'adresse réseau précédente.

Vous ne pouvez pas rétrograder le firmware pour certains appareils. Consultez la documentation de votre appareil.

Notez que les programmes prévus pour une version de firmware de CPU risquent de ne pas fonctionner avec une autre version de firmware. La CPU ne peut pas passer en mode Run si le programme est incompatible avec la version de firmware.

Préparation des fichiers de mise à jour du firmware pour utilisation avec SIMATIC Automation Tool

- Les mises à jour du firmware sont disponibles sur le site Web d'assistance client (<https://www.siemens.com/automation/>).
- Une autre option consiste à sélectionner la ligne d'un appareil, puis à sélectionner "Rechercher des mises à jour du firmware", soit depuis le menu Outils ou depuis l'icône Outils de la barre d'outils. SIMATIC Automation Tool lance la page Web de support client de l'appareil. Le site Internet d'assistance de Siemens affiche la page correspondant au numéro d'article de la ligne. Par ex., une commande "Rechercher des mises à jour" sur le numéro d'article 6ES7 215-1HG31-0XB0 renvoie vers la page web d'assistance du CPU 1215C (<https://support.industry.siemens.com/cs/ww/fr/ps/6ES7215-1HG31-0XB0>)

Pour une CPU, le fichier de mise à jour du firmware nommé **6ES7 211-1AE40-0XB0_V04.00.02.exe** est, par exemple, destiné uniquement au modèle **CPU 1211C DC/DC/DC**. Si vous utilisez le fichier .upd contenu dans ce package pour un autre modèle de CPU S7-1200, la mise à jour échouera.

Lorsque vous exécutez le fichier de mise à jour et que les fichiers sont extraits, les dossiers et fichiers suivants sont affichés.

- fichier : S7-JOB.SYS
- dossier : FWUPDATE.SYS contient le fichier .upd.
 - fichier : **6ES7 211-1AE40-0XB0_V04.00.02.upd** (fichier .upd utilisé par SIMATIC Automation Tool)

Pour un module E/S, le fichier de mise à jour du firmware nommé **232-4HD32-0XB0_V203.exe** est destiné uniquement au module **SM1232 ANALOG OUTPUT 4AO**. Le fichier .exe auto-extractible contient le fichier **6ES7 232-4HD32-0XB0_V02.00.03_00.00.00.upd** qui est utilisé par SIMATIC Automation Tool.

Remarque

Nouveau format des fichiers de mise à jour du firmware

- Le nom du fichier .exe de mise à jour auto-extractible doit renvoyer au numéro d'article de l'appareil que vous voulez mettre à jour.
 - Le nom du fichier .upd extrait doit correspondre au numéro d'article de l'appareil et à la version de firmware que vous voulez charger.
-

Remarque

Ancien format des fichiers de mise à jour du firmware

- Le nom du fichier .exe de mise à jour auto-extractible doit renvoyer au numéro d'article de l'appareil que vous voulez mettre à jour.
 - Contient trois fichiers ou plus en fonction de la taille du firmware.
 - Créez un dossier avec un nom au choix dans le Dossier de mise à jour du firmware (Page 113). Vous pouvez choisir un nom reprenant le numéro d'article et le numéro de version, afin de simplifier l'identification, mais vous pouvez choisir un nom quelconque. SIMATIC Automation Tool examine tous les fichiers de firmware au démarrage pour confirmer les numéros de version du firmware.
-

Copie des fichiers .upd dans le dossier de mise à jour du firmware

Les noms des fichiers .upd au nouveau format reprennent le modèle du module cible et les numéros de version. Vous pouvez copier plusieurs fichiers .upd dans un même dossier firmware et identifier le module cible) l'aide du nom du fichier .upd. Copiez tous les fichiers .upd requis dans le dossier de mise à jour du firmware (Page 113).

ATTENTION

Vérifiez qu'un processus ne soit pas en cours sur la CPU avant d'installer les mises à jour du firmware.

L'installation d'une mise à jour du firmware sur une CPU ou un module fait passer la CPU en mode STOP, ce qui peut affecter l'exécution d'un processus ou d'une machine en ligne, dont la mise en route inattendue peut entraîner la mort, des blessures graves et des dommages matériels importants.

Effectuer la mise à jour du firmware dans les CPU et les modules

Une fois que les fichiers .upd sont copiés dans le dossier de mise à jour du firmware, vous pouvez utiliser SIMATIC Automation Tool pour mettre à jour le firmware dans les appareils. Procédez comme suit pour effectuer la mise à jour :

1. Cliquez sur l'onglet "Mise à jour du firmware".
2. Sélectionnez un ou plusieurs appareils à inclure dans l'opération. Vous pouvez utiliser la case Appareils située en haut de la table Appareil pour sélectionner / désélectionner tous les appareils. Vous pouvez également utiliser le menu contextuel (clic droit) ou la commande de menu "Modifier > Sélectionner" pour accéder aux commandes "Sélectionner tout" et "Désélectionner tout".
3. Pour chaque ligne d'appareil sélectionnée, utilisez la liste déroulante de la colonne "Nouvelle version firmware" et sélectionnez une version de firmware pour une CPU ou un module. La liste déroulante affiche le nom des fichiers .upd que vous avez copiés vers le dossier de mise à jour du firmware (Page 113). Si de nouvelles versions du firmware (fichiers .upd) sont disponibles dans le dossier de mise à jour du firmware, alors ces fichiers sont disponibles à partir de la liste déroulante "Nouvelle version firmware".

Vous pouvez également utiliser le bouton de navigation  et accéder au dossier contenant les fichiers de mise à jour du firmware sur votre PG/PC. Sélectionnez un fichier pour l'ajouter à la liste déroulante. Si le fichier sélectionné porte le même nom qu'un fichier de la liste, SIMATIC Automation Tool ajoute un chiffre au nouveau fichier afin que le nom reste univoque. Pour vous aider à identifier les fichiers, une info-bulle affiche le chemin d'accès complet et le nom du fichier.

4. Pour chaque appareil sélectionné, entrez un mot de passe, le cas échéant, dans la colonne "Mot de passe dans la CPU".
5. Sélectionner la commande de menu "Opérations > Mettre à jour > Mise à jour du firmware" pour démarrer l'opération. Vous pouvez également cliquer sur le bouton de barre d'outils Mettre à jour  et la commande "Mise à jour du firmware" dans le menu déroulant du bouton.

Le journal des événements (Event Log) sous la table Appareil montre le résultat de cette opération. Une fois que l'opération de mise à jour du firmware est terminée avec succès, SIMATIC Automation Tool efface le champ "Nouvelle version firmware".

Mots de passe des CPU de sécurité requis pour la mise à jour du firmware

Pour les versions de firmware antérieures à S7-1200 V4.2 et S7-1500 V2.0, une CPU de sécurité protégée par mot de passe nécessite le mot de passe de CPU F de sécurité (niveau max. 5) pour l'opération de mise à jour du firmware. Étant donné qu'une mise à jour du firmware n'est pas une opération liée à la sécurité, le niveau de sécurité du mot de passe a été abaissé. Les versions ultérieures du firmware de CPU F nécessitent uniquement un accès en lecture-écriture (accès complet de niveau 4).

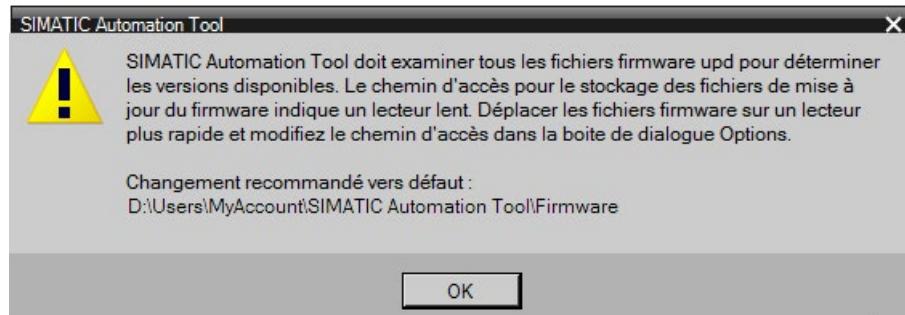
SIMATIC Automation Tool ne vérifie pas le niveau de légitimation pour différentes versions de firmware. L'opération est lancée et l'appareil rejette un mot de passe selon la configuration du firmware, ce qui génère l'erreur suivante dans l'Event Log :

"Le mot de passe saisi est insuffisant pour l'exécution de l'opération".

La version S7-1200 V4.2 ou les versions ultérieures et la version S7-1500 V2.0 ou les versions ultérieures, ne nécessitent qu'un "Accès complet (aucune protection)". Si le mot de passe de CPU F de sécurité est utilisé, cela vous permet également d'obtenir un accès en lecture-écriture, de sorte que l'opération réussira toujours si le mot de passe de sécurité est entré.

Message d'erreur de dépassement de temps suite à une communication trop lente avec le support de stockage du fichier .upd.

Si le message d'erreur suivant s'affiche, plus de dix secondes se sont écoulées et SIMATIC Automation Tool n'a pas fini de traiter tous les fichiers .upd dans le dossier de stockage du firmware. La durée nécessaire pour ouvrir et scanner tous les fichiers .upd est fonction du temps d'accès aux données et du nombre de fichiers .upd qui se trouvent dans le dossier.



Ce dépassement de temps peut survenir lorsque la communication avec un support de stockage distant est trop lente.

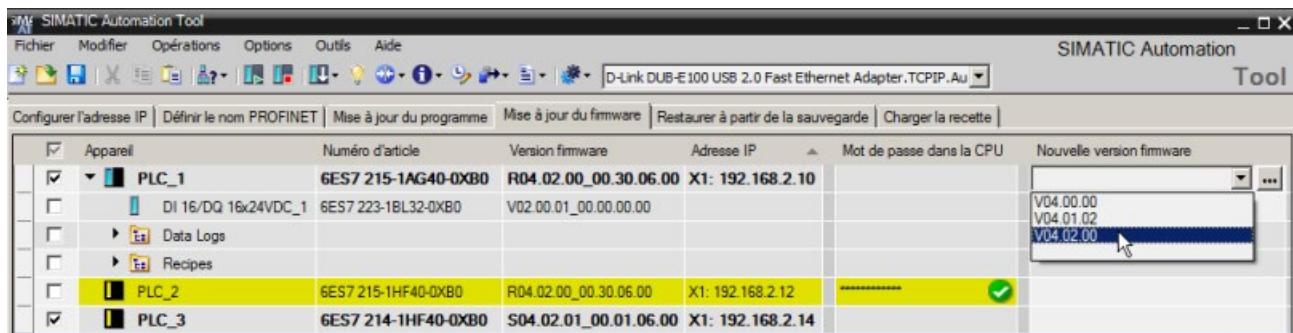
Pour éviter ce problème, définissez un chemin de stockage plus rapide pour les données de firmware dans la partie Mise à jour du firmware de la boîte de dialogue de la commande de menu "Options > Paramètres". Copiez les fichiers .upd requis sur un support local plus rapide et renouvez l'opération.

Exemple de mise à jour du firmware

Cet exemple montre comment mettre à jour le firmware pour une seule CPU.

Procédez comme suit pour effectuer la mise à jour du firmware :

1. Ouvrez la liste déroulante des versions disponibles à partir de la colonne "Nouvelle version firmware". La liste déroulante affiche tous les fichiers de mise à jour du firmware dans le dossier de mise à jour du firmware (Page 113).
2. Sélectionnez la version de mise à jour du firmware à utiliser (si vous avez sélectionné plusieurs appareils, vous devez choisir un fichier de mise à jour pour chaque appareil sélectionné).
3. Sélectionnez la commande de menu "Opérations > Mettre à jour > Mise à jour du firmware" pour démarrer l'opération. Vous pouvez également cliquer sur le bouton de barre d'outils Mettre à jour et la commande de menu "Mise à jour du firmware" dans le menu déroulant du bouton.



Remarque

Le firmware de certains modules S7-1200 ne peut pas être mis à jour avec SIMATIC Automation Tool

Si vous voyez s'afficher le message d'erreur "Pour cet appareil, la mise à jour du firmware doit être prise en charge à la fois par la CPU et par le module. Cet appareil ne peut être mis à jour qu'avec une carte SD", vous ne pouvez pas mettre à jour le firmware du module avec SIMATIC Automation Tool.

4.14 Sauvegarde et restauration de la CPU ou des données IHM

Sauvegarder les données d'un appareil

La commande "Sauvegarder l'appareil dans un fichier" crée de nouveaux fichiers de sauvegarde de données et copie les fichiers dans le dossier de sauvegarde et de restauration (Page 116).

Vous pouvez utiliser ces fichiers avec l'opération Restaurer l'appareil dans SIMATIC Automation Tool.

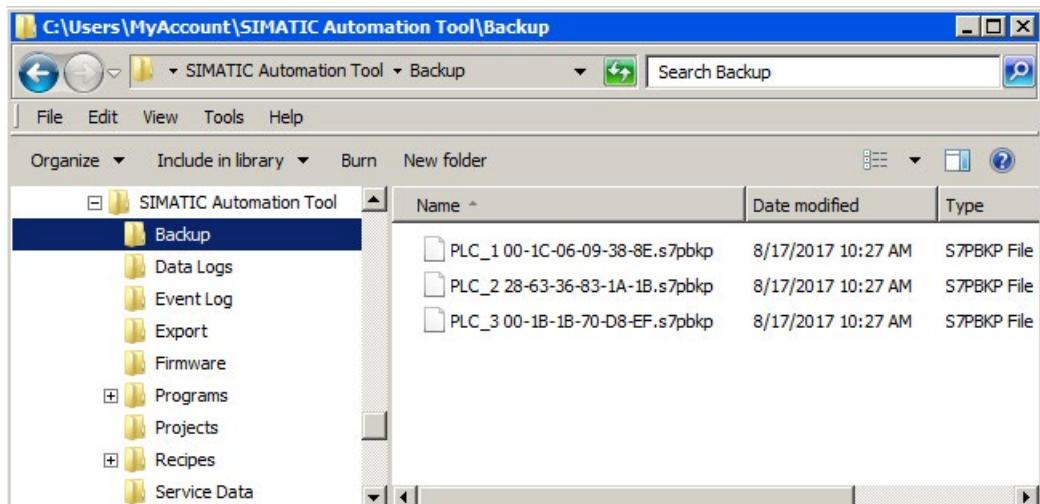
Vous pouvez lancer la sauvegarde à partir de n'importe quel onglet.

Pour créer un fichier de sauvegarde, procédez comme suit :

1. Sélectionnez un ou plusieurs appareils à inclure dans l'opération. Vous pouvez utiliser la case "Appareils" située en haut de la table Appareil pour sélectionner / désélectionner tous les appareils. Vous pouvez également utiliser le menu contextuel (clic droit) ou la commande de menu "Modifier > Sélectionner" pour accéder aux commandes "Sélectionner tout" et "Désélectionner tout".
2. Sélectionner la commande de menu "Sauvegarder/Restaurer > Sauvegarder l'appareil dans un fichier" dans le menu Opérations. Vous pouvez également cliquer sur le bouton Sauvegarder/restaurer  de la barre d'outils et sélectionner "Sauvegarder l'appareil dans un fichier" dans le menu déroulant du bouton.

Le journal des événements (Event Log) sous la table Appareil montre le résultat de cette opération.

Si l'opération est terminée avec succès, SIMATIC Automation Tool crée un nom de fichier de sauvegarde pour les appareils S7 et IHM. Le nom de fichier combine le nom du projet, l'adresse MAC et .s7pbkp. SIMATIC Automation Tool copie les fichiers dans le dossier de sauvegarde et de restauration (Page 116).



Restaurer un appareil à partir d'un fichier de sauvegarde

Vous utilisez la commande "Restaurer l'appareil depuis un fichier" pour restaurer les fichiers de sauvegarde sur les appareils correspondants. Les fichiers de sauvegarde S7 et IHM que vous avez créés avec la commande "Sauvegarder l'appareil dans un fichier" portent le nom d'extension "s7pbkp". Vous pouvez restaurer les fichiers à partir du dossier de sauvegarde et de restauration (Page 116) ou naviguer vers un autre emplacement.

Si votre réseau présente une topologie en chaîne et que les Paramètres de communication (Page 110) activent le traitement multi-thread (multifil), prenez garde au risque d'interruption de communication lors de cette opération.

Remarque

Appareils de sécurité

Si la CPU de sécurité est protégée, vous devez entrer le mot de passe de CPU F de sécurité dans la colonne "Mot de passe dans la CPU" pour restaurer l'appareil de sécurité à partir d'un fichier de sauvegarde.

Vous devez confirmer une invite supplémentaire et sélectionner de nouveau votre appareil si le programme installé dans une CPU F est un programme de sécurité (Page 124).

SIMATIC Automation Tool place les demandes "Restaurer à partir de la sauvegarde" pour les appareils de sécurité dans la file d'attente d'opérations sur CPU F relatives à la sécurité. SIMATIC Automation Tool utilise uniquement le traitement séquentiel à thread unique pour la file d'attente d'opérations de sécurité.

L'appareil de destination pour un programme de sécurité doit être une CPU de sécurité.

 **ATTENTION**

Vérifiez qu'aucun appareil ne soit en train d'exécuter un processus avant de restaurer un appareil à partir d'un fichier de sauvegarde.

La restauration d'un appareil fait passer la CPU en mode STOP, ce qui peut affecter l'exécution d'un processus ou d'une machine en ligne. Le démarrage inattendu d'un processus ou d'une machine peut entraîner la mort ou des dommages corporels et/ou matériels.

Pour restaurer les appareils sélectionnés à partir d'un fichier de sauvegarde, procédez comme suit :

1. Cliquez sur l'onglet "Restaurer à partir de la sauvegarde" dans la table Appareil.
2. Sélectionnez un ou plusieurs appareils à inclure dans l'opération. Vous pouvez utiliser la case Appareils située en haut de la table Appareil pour sélectionner / désélectionner tous les appareils. Vous pouvez également utiliser le menu contextuel (clic droit) ou la commande de menu "Modifier > Sélectionner" pour accéder aux commandes "Sélectionner tout" et "Désélectionner tout".
3. Pour chaque appareil, sélectionnez un nom de fichier de sauvegarde dans la liste déroulante "Fichier de sauvegarde". La liste déroulante affiche le nom des fichiers .s7pbkp qui se trouvent dans le dossier de sauvegarde et de restauration (Page 116).

Vous pouvez également utiliser le bouton de navigation  et accéder au dossier contenant les fichiers de sauvegarde sur votre PG/PC. Sélectionnez un fichier pour l'ajouter à la liste déroulante. Si le fichier que vous avez sélectionné porte le même nom qu'un fichier existant, SIMATIC Automation Tool ajoute un chiffre au nouveau fichier afin que le nom reste unique.

Pour les fichiers valides, une info-bulle affiche le chemin d'accès et le nom du fichier. Vous voyez également une coche verte à côté du nom du fichier.

Pour les fichiers non-valides, l'info-bulle affiche l'erreur de fichier, également affichée dans l'Event Log. Vous voyez une icône "X" rouge à côté du nom du fichier.

4. Le cas échéant, entrez les mots de passe dans les colonnes "Mot de passe dans la CPU" et "Mot de passe dans le fichier de sauvegarde". L'opération Restaurer à partir de la sauvegarde est liée à la sécurité ; le mot de passe de CPU F de sécurité est donc requis pour un appareil de sécurité protégé par mot de passe.
5. Sélectionnez la commande de menu "Opérations > Sauvegarder/Restaurer > Restaurer l'appareil depuis un fichier" pour démarrer l'opération. Vous pouvez également cliquer sur le bouton Sauvegarder/Restaurer  de la barre d'outils et sélectionner "Restaurer l'appareil depuis un fichier" dans le menu déroulant du bouton.

Validation du fichier de sauvegarde

Avant de lancer l'opération de restauration, SIMATIC Automation Tool effectue des contrôles limités sur les données du fichier de sauvegarde.

- Le nom de l'extension de fichier et les données d'en-tête sont validés.
- Vous ne pouvez pas restaurer à partir d'un fichier de sauvegarde qui contient un programme de sécurité lorsque l'appareil cible n'est pas une CPU de sécurité.

Si un fichier de sauvegarde n'est pas valide, SIMATIC Automation Tool affiche une icône "X" rouge dans le champ "Fichier de sauvegarde". De plus amples informations sur les erreurs sont disponibles dans une info-bulle lorsque vous passez votre curseur sur la cellule.

Gestion des mots de passe après l'opération de restauration

Si une CPU est protégée par mot de passe, vous devez entrer un mot de passe pour que l'opération Restaurer à partir de la sauvegarde puisse être terminée avec succès.

Après qu'un fichier de sauvegarde a été restauré sur une CPU, le nouveau fichier peut avoir un mot de passe. Le mot de passe que vous avez restauré peut être différent du mot de passe précédent si la CPU avait un mot de passe. Vous devez donc entrer un second mot de passe dans la colonne "Mot de passe dans le fichier de sauvegarde". Le second mot de passe devient le mot de passe de la CPU une fois que l'opération de restauration est terminée.

Une fois que l'opération de restauration est terminée avec succès, SIMATIC Automation Tool copie automatiquement le second mot de passe (le mot de passe que vous avez entré sous "Mot de passe dans le fichier de sauvegarde") vers le champ mot de passe de la CPU et effectue une tentative de connexion à l'aide du nouveau mot de passe. SIMATIC Automation Tool supprime ensuite le second mot de passe et le chemin d'accès vers le fichier de sauvegarde.

Avant la restauration d'un fichier de sauvegarde sur une CPU :

Mot de passe dans la CPU	Fichier de sauvegarde	Mot de passe dans le fichier de sauvegarde
*****	PLC_2 28-63-36-83-1A-1B	?

Après la restauration d'un fichier de sauvegarde sur une CPU :

Mot de passe dans la CPU	Fichier de sauvegarde	Mot de passe dans le fichier de sauvegarde

Si le mot de passe que vous avez entré dans la colonne "Mot de passe dans le fichier de sauvegarde" est incorrect et n'est pas le mot de passe configuré dans les données restaurées de la CPU, l'Event Log affiche un avertissement une fois l'opération terminée. Dans ce cas, le mot de passe de la CPU affiche une icône 'X' rouge qui indique un mot de passe non valide.

Validation de la signature de sécurité

Un projet TIA Portal qui contient un programme de sécurité comporte une signature de sécurité. SIMATIC Automation Tool utilise la signature de sécurité pour vérifier les données dans un fichier programme, ce qui renforce le niveau de sécurité des programmes de sécurité. Après une opération Restaurer à partir de la sauvegarde, SIMATIC Automation Tool compare la signature de sécurité dans le fichier de projet à la signature de sécurité qui se trouve maintenant dans le programme de la CPU.

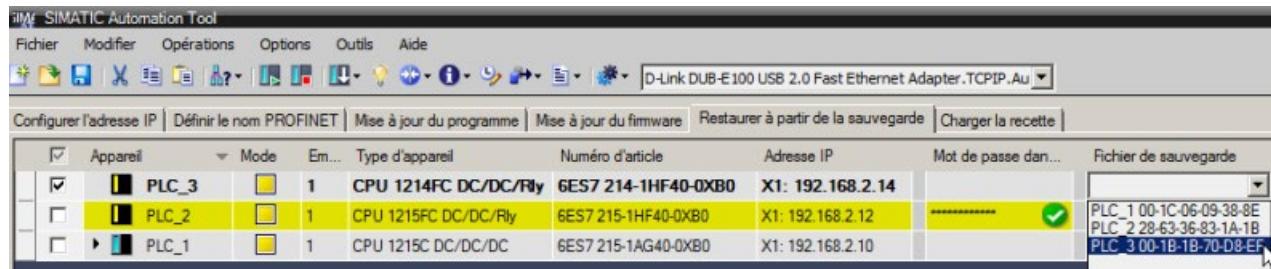
Lorsqu'elle est réussie, la comparaison est indiquée comme suit dans l'Event Log : "Les résultats de la comparaison CRC en ligne et des signatures de sécurité collectives hors ligne correspondent."

Lorsqu'elle échoue, la comparaison est indiquée comme suit dans l'Event Log par SIMATIC Automation Tool : "Les résultats de la comparaison CRC en ligne et des signatures de sécurité collectives hors ligne ne correspondent pas.". Dans ce cas, restaurez les paramètres d'usine de l'appareil (Page 89) et réessayer de mettre à jour le programme. N'essayez pas de passer en mode RUN si vous n'êtes pas certain(e) d'avoir chargé le bon programme.

Exemple de restauration à partir d'un fichier de sauvegarde

Cet exemple montre un appareil sélectionné et la sélection d'un fichier de sauvegarde pour le champ "Fichier de sauvegarde". Si votre installation comporte plusieurs appareils, vous sélectionnez un fichier de sauvegarde à restaurer pour chaque appareil.

L'Event Log sous la table Appareil montre le résultat de cette opération.



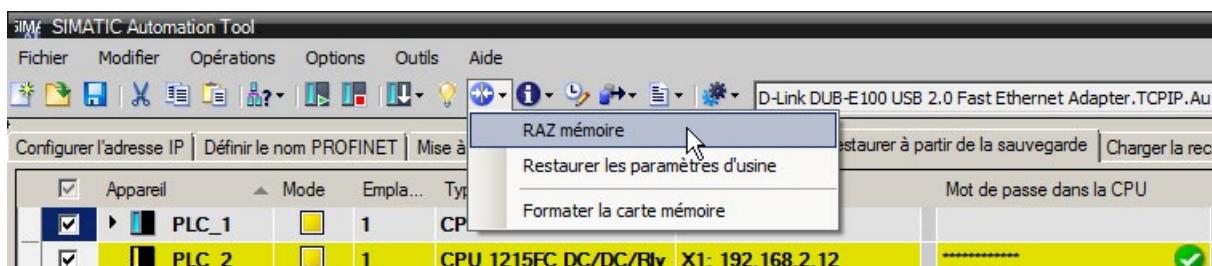
4.15 Restaurer la mémoire de la CPU

Restaurer la mémoire des CPU sélectionnées

Pour restaurer la mémoire de la CPU sur les appareils sélectionnés, procédez comme suit :

1. Sélectionnez un ou plusieurs appareils à inclure dans l'opération. Vous pouvez utiliser la case Appareils située en haut de la table Appareil pour sélectionner / désélectionner tous les appareils. Vous pouvez également utiliser le menu contextuel (clic droit) ou la commande de menu "Modifier > Sélectionner" pour accéder aux commandes "Sélectionner tout" et "Désélectionner tout".
2. Pour chaque CPU sélectionnée, entrez un mot de passe, le cas échéant, dans la colonne "Mot de passe dans la CPU".
3. Sélectionnez la commande "Opérations > Réinitialiser > RAZ mémoire" pour démarrer l'opération. Vous pouvez également cliquer sur le bouton "Réinitialiser" de la barre d'outils et sélectionner "RAZ mémoire" dans le menu déroulant du bouton.
4. Cliquez sur le bouton "Continuer" dans la boîte de dialogue "RAZ mémoire".

SIMATIC Automation Tool effectue une réinitialisation de la mémoire sur les appareils sélectionnés.



Le journal sous la table Appareil montre le résultat de cette opération.

4.16 Restaurer les paramètres d'usine des CPU et des modules

Restaurer les paramètres d'usine des appareils sélectionnés

Vous pouvez restaurer les paramètres d'usine des appareils sélectionnés, sauf l'adresse IP. L'appareil conserve l'adresse IP existante afin de préserver les affectations IP du réseau.

Si votre réseau présente une topologie en chaîne et que les Paramètres de communication (Page 110) activent le traitement multi-thread (multifil), prenez garde au risque d'interruption de communication lors de cette opération.

Remarque

Appareils de sécurité

Si la CPU de sécurité est protégée, vous devez entrer le mot de passe de CPU F de sécurité dans la colonne "Mot de passe dans la CPU" pour restaurer les paramètres d'usine d'un appareil de sécurité.

Vous devez confirmer une invite supplémentaire et sélectionner de nouveau votre appareil si le programme installé dans la CPU F est un programme de sécurité (Page 124).

Les demandes de restauration des paramètres d'usine pour les appareils de sécurité sont placées dans la file d'attente des opérations relatives à la sécurité et seul le traitement séquentiel à thread unique est autorisé.

ATTENTION

Vérifiez que l'appareil ne soit pas en train d'exécuter un processus avant de lancer l'opération Restaurer les paramètres d'usine.

La restauration des paramètres d'usine fait passer la CPU en mode STOP, ce qui peut affecter l'exécution d'un processus ou le fonctionnement d'une machine en ligne. Le démarrage inattendu d'un processus ou d'une machine peut entraîner la mort ou des dommages corporels et/ou matériels.

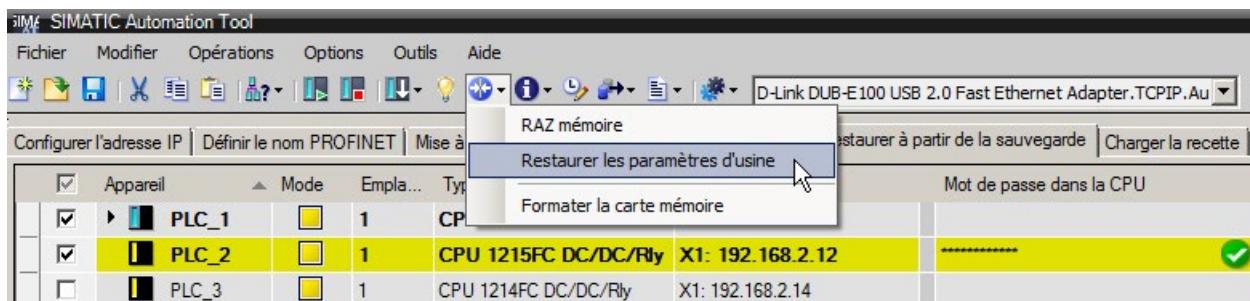
Pour restaurer les paramètres d'usine des appareils sélectionnés, procédez comme suit :

1. Sélectionnez un ou plusieurs appareils à inclure dans l'opération. Vous pouvez utiliser la case Appareils située en haut de la table Appareil pour sélectionner / désélectionner tous les appareils. Vous pouvez également utiliser le menu contextuel (clic droit) ou la commande de menu "Modifier > Sélectionner" pour accéder aux commandes "Sélectionner tout" et "Désélectionner tout".

Pour chaque CPU sélectionnée, entrez un mot de passe, le cas échéant, dans la colonne "Mot de passe dans la CPU" de l'onglet ouvert. L'opération Restaurer les paramètres d'usine est liée à la sécurité ; le mot de passe de CPU F de sécurité est donc requis pour un appareil de sécurité protégé par mot de passe.

2. Sélectionnez la commande de menu "Opérations > Restaurer > Restaurer les paramètres d'usine" pour démarrer l'opération. Vous pouvez également cliquer sur le bouton "Restaurer"  de la barre d'outils et sélectionner "Restaurer les paramètres d'usine" dans le menu déroulant du bouton.

Pour les CPU de sécurité, SIMATIC Automation Tool affiche la boîte de dialogue "Mise à jour du programme" pour une confirmation supplémentaire. Sélectionnez l'appareil, les appareils ou tous les appareils dont vous souhaitez restaurer les paramètres d'usine.



3. Cliquez sur le bouton "Continuer" dans la boîte de dialogue "Restaurer les paramètres d'usine".

SIMATIC Automation Tool restaure les paramètres d'usine des appareils sélectionnés.

4. Patientez pendant la réinitialisation. Attendez que les voyants de l'appareil cessent de clignoter avant de lancer toute autre opération.

Remarque

La restauration des paramètres d'usine ne supprime pas les données de la carte mémoire SIMATIC.

Si vous disposez d'une carte mémoire SIMATIC dans une CPU, l'opération "Restaurer les paramètres d'usine" ne supprime pas son contenu. Si vous ne disposez pas d'une carte mémoire SIMATIC dans une CPU, l'opération "Restaurer les paramètres d'usine" supprime le programme dans la mémoire de chargement interne de la CPU.

Le journal des événements (Event log) sous la table Appareil montre le résultat de cette opération.

4.17 Formater la carte mémoire

Les cartes mémoire SIMATIC s'insèrent dans les appareils SIMATIC et prennent en charge diverses applications. Selon le type ou la gamme d'appareils, vous pouvez utiliser des cartes mémoire aux fins suivantes :

- Mémoire de chargement d'une CPU
- Support de stockage pour les projets
- Sauvegarde et mise à jour du firmware
- Support de stockage pour le nom d'appareil PROFINET
- Transfert de projet d'un appareil à un autre
- Autres fichiers

Remarque

Le formatage des cartes mémoire SIMATIC s'effectue uniquement à l'aide de logiciels Siemens.

Si vous utilisez une carte mémoire SIMATIC à des fins autres que SIMATIC ou si vous ne la formatez pas correctement, la structure interne de la carte SIMATIC est écrasée. La structure n'est pas récupérable et la carte mémoire SIMATIC devient inutilisable pour les appareils SIMATIC.

N'utilisez pas de cartes mémoire SIMATIC dans des contextes différents de SIMATIC et ne formatez pas les cartes mémoire SIMATIC avec des appareils tiers ni des outils Windows.

Si votre réseau présente une topologie en chaîne et que les Paramètres de communication (Page 110) activent le traitement multi-thread (multifil), prenez garde au risque d'interruption de communication lors de cette opération.

 ATTENTION
<p>Vérifiez que l'appareil ne soit pas en train d'exécuter un processus avant de formater une carte mémoire.</p> <p>Le formatage d'une carte mémoire fait passer la CPU en mode STOP, ce qui peut affecter le fonctionnement d'une machine ou d'un processus en ligne. Le démarrage inattendu d'un processus ou d'une machine peut entraîner la mort ou des dommages corporels et/ou matériels.</p>

Pour formater les cartes mémoire SIMATIC sur les appareils sélectionnés, procédez comme suit :

1. Sélectionnez un ou plusieurs appareils à inclure dans l'opération. Vous pouvez utiliser la case Appareils située en haut de la table Appareil pour sélectionner / désélectionner tous les appareils. Vous pouvez également utiliser le menu contextuel (clic droit) ou la commande de menu "Modifier > Sélectionner" pour accéder aux commandes "Sélectionner tout" et "Désélectionner tout".
2. Pour chaque CPU sélectionnée, entrez un mot de passe, le cas échéant, dans la colonne "Mot de passe dans la CPU" de l'onglet ouvert.
3. Sélectionner la commande "Opérations > Restaurer > Formater la carte mémoire" pour démarrer l'opération. Vous pouvez également cliquer sur le bouton "Restaurer" de la barre d'outils et sélectionner "Formater la carte mémoire" dans le menu déroulant du bouton.
4. Cliquez sur le bouton "Continuer" dans la boîte de dialogue "Formater la carte mémoire".

SIMATIC Automation Tool formate la carte mémoire des appareils sélectionnés.

Remarque

Appareils de sécurité

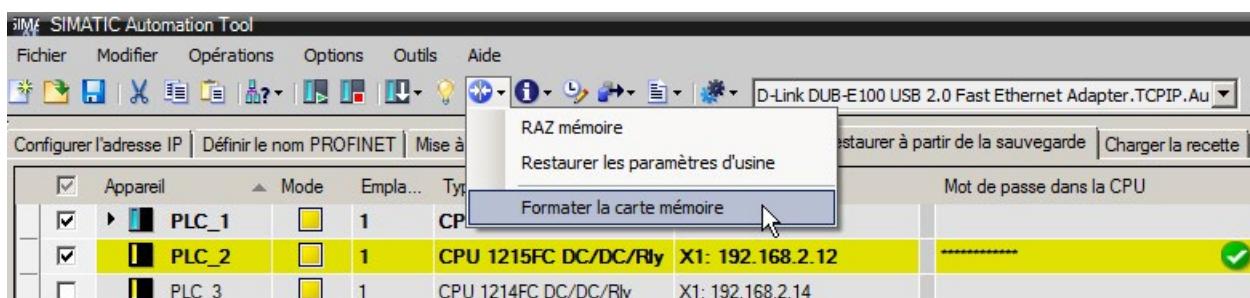
Si une CPU de sécurité est protégée, vous devez entrer le mot de passe de CPU F de sécurité dans la colonne "Mot de passe dans la CPU" pour formater une carte mémoire SIMATIC dans un appareil de sécurité.

Vous devez confirmer une invite supplémentaire et sélectionner de nouveau votre appareil si le programme installé dans la CPU F est un programme de sécurité (Page 124).

Les demandes de formatage de cartes mémoire pour les appareils de sécurité sont placées dans la file d'attente des opérations relatives à la sécurité et seul le traitement séquentiel à thread unique est autorisé.

Dans cet exemple, SIMATIC Automation Tool formate les cartes mémoire des appareils sélectionnés lorsque vous cliquez sur le bouton "Continuer".

L'Event Log sous la table Appareil montre le résultat de cette opération.



4.18 Récupérer les Service Data depuis des CPU

Lorsqu'une CPU passe à un état de défaut, elle enregistre des informations de défaillance qui peuvent être chargées sur le PG/PC. Vous pouvez envoyer ces Service Data à l'assistance client de Siemens afin de l'aider à trouver la cause de la défaillance.

Vous pouvez récupérer les Service Data lorsque la CPU est en mode STOP ou RUN. Les service data contiennent plusieurs fichiers qui sont compressés dans un seul fichier .zip dont le nom reprend le nom de l'API, la date et l'heure. Un chiffre est ajouté au nom du fichier pour éviter tout doublon.

Vous configurez ou acceptez le chemin des données de maintenance (Service Data) par défaut à partir de Paramètres des Service Data (Page 115).

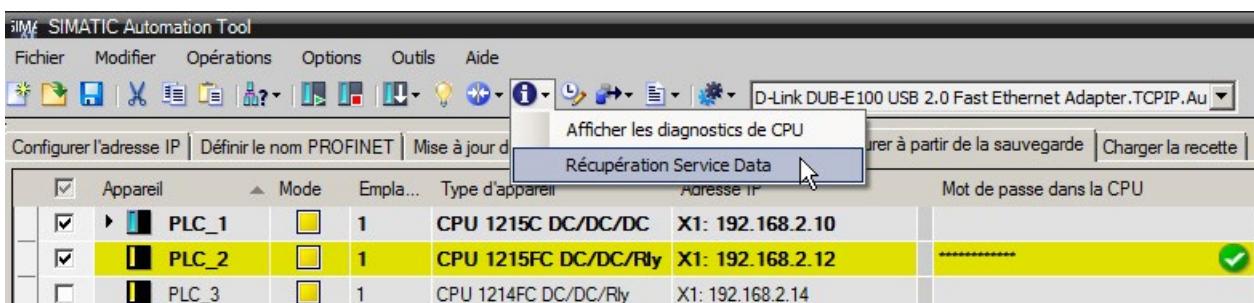
Pour récupérer les Service Data depuis des CPU sélectionnées, procédez comme suit :

1. Sélectionnez un ou plusieurs appareils à inclure dans l'opération. Vous pouvez utiliser la case Appareils située en haut de la table Appareil pour sélectionner / désélectionner tous les appareils. Vous pouvez également utiliser le menu contextuel (clic droit) ou la commande de menu "Modifier > Sélectionner" pour accéder aux commandes "Sélectionner tout" et "Désélectionner tout".
2. Pour chaque CPU sélectionnée, entrez un mot de passe, le cas échéant, dans la colonne "Mot de passe dans la CPU" de l'onglet ouvert.
3. Sélectionner la commande "Opérations > Diagnostic > Récupérer les Service Data" pour démarrer l'opération. Vous pouvez également cliquer sur le bouton "Afficher le diagnostic de l'appareil"  de la barre d'outils et sélectionner "Récupérer les Service Data" dans le menu déroulant du bouton.
4. Cliquez sur le bouton "Continuer" dans la boîte de dialogue "Chargement Service Data".

L'outil d'automatisation SIMATIC récupère les Service Data des CPU sélectionnées et stocke les fichiers dans le dossier de Service Data (Page 115).

Exemple :

Une fois que vous avez cliqué sur le bouton "Continuer", l'outil d'automatisation SIMATIC récupère les Service Data des appareils sélectionnés.



Le journal des événements (Event log) sous la table Appareil montre le résultat de cette opération.

Fichiers de Service Data

Remarque

Récupérer les fichiers de Service Data dans des CPU protégées par mot de passe

Si une CPU est protégée par un mot de passe, vous devez fournir un mot de passe avec accès en lecture ou accès complet pour récupérer les fichiers de Service Data. Saisissez les mots de passe de la CPU dans la colonne "Mot de passe dans la CPU" de SIMATIC Automation Tool avant d'exécuter la commande "Récupérer les Service Data".

Exemple de fichier de données de maintenance pour S7-1200 : PLC_1 00-1C-06-13-58-10.zip

Contenu du fichier .zip :

ResourceStats.txt

RAM.img

PLCInformation.txt

NAND.img

General.txt

Fault.bin

DNN.txt

CommBuffers.txt

ASLog.txt

Alarms.txt

ATTENTION

Les Service Data sont affichées en texte clair.

Un utilisateur mal intentionné pourrait lire les fichiers de Service Data pour se procurer les états et les détails de la configuration du système de conduite. Les fichiers de Service Data sont enregistrés en texte clair non crypté (codage binaire) dans la CPU. Un mot de passe CPU peut contrôler l'accès à ces informations.

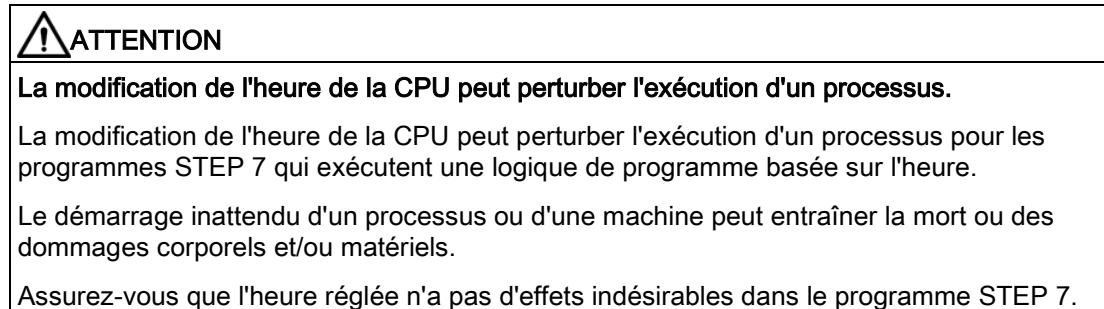
Le fait de faire fonctionner un processus ou une machine avec des données corrompues peut affecter l'exécution d'un processus ou le fonctionnement d'une machine en ligne. Le démarrage inattendu d'un processus ou d'une machine peut entraîner la mort ou des dommages corporels et/ou matériels.

Utilisez la configuration de l'appareil dans TIA Portal pour définir une protection CPU à l'aide d'un mot de passe fort. Les mots de passe forts contiennent au moins dix caractères, lettres, chiffres et caractères spéciaux mélangés, ne sont pas des mots du dictionnaire et ne sont pas des noms ou identifiants pouvant être déduits de vos données personnelles. Ne divulguez pas le mot de passe et changez-en fréquemment.

4.19 Réglage de l'heure sur les CPU

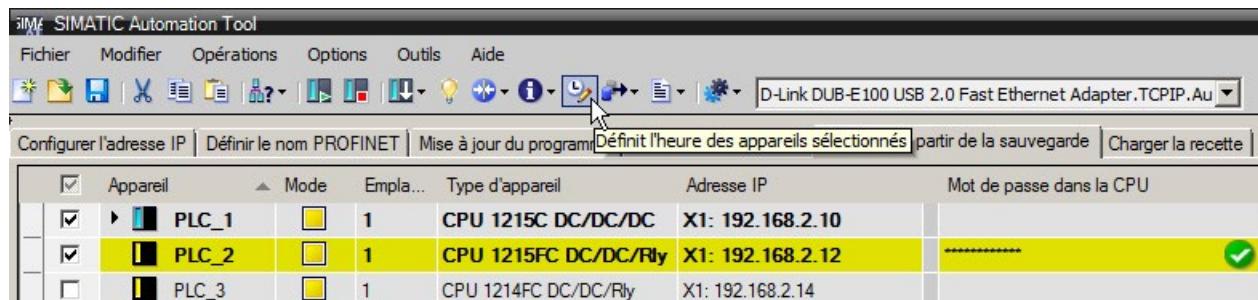
Réglage de l'heure des CPU sur celle du PG/PC

Le bouton Heure règle l'heure des CPU sélectionnées sur celle de votre PG/PC actif. Les informations relatives au fuseau horaire et à l'heure d'été ne sont pas modifiées et doivent être changées dans le projet TIA Portal.



Pour régler la date et l'heure de la CPU sur celles de votre PG/PC, procédez comme suit :

1. Sélectionnez un ou plusieurs appareils à inclure dans l'opération. Vous pouvez utiliser la case Appareils située en haut de la table Appareil pour sélectionner / désélectionner tous les appareils. Vous pouvez également utiliser le menu contextuel (clic droit) ou la commande de menu "Modifier > Sélectionner" pour accéder aux commandes "Sélectionner tout" et "Désélectionner tout".
2. Pour chaque CPU sélectionnée, entrez un mot de passe, le cas échéant, dans la colonne "Mot de passe dans la CPU" de l'onglet ouvert.
3. Sélectionnez la commande de menu "Opérations > Régler l'heure" ou cliquez sur le bouton "Régler l'heure des appareils sélectionnés"  de la barre d'outils.



4. Cliquez sur le bouton "Continuer" dans la boîte de dialogue "Régler l'heure".

SIMATIC Automation Tool règle l'heure système des appareils sélectionnés sur celle de votre PG/PC actif.

Le journal sous la table Appareil montre le résultat de cette opération.

4.20 Afficher le tampon de diagnostic de la CPU

Tampon de diagnostic CPU

Un tampon de diagnostic CPU contient une entrée pour chaque événement de diagnostic. Chaque entrée inclut la date et l'heure auxquelles l'événement s'est produit, une catégorie d'événement, ainsi qu'une description de l'événement. Le tampon de diagnostic affiche les entrées par ordre chronologique, l'événement le plus récent se trouvant en haut. Lorsque le journal est plein, l'événement le plus ancien dans le journal est remplacé par un nouvel événement. En cas de coupure de courant, les événements sont sauvegardés.

Pour afficher le diagnostic de l'appareil, procédez comme suit :

1. Sélectionnez une ou plusieurs CPU dans la table Appareil. Si vous souhaitez d'abord désélectionner tous les appareils, vous pouvez utiliser la case de désélection des appareils située en haut de la table Appareil. Vous pouvez également utiliser le menu contextuel (clic droit) ou le menu "Modifier > Sélectionner" pour accéder à la commande "Désélectionner tout".
2. Pour chaque CPU sélectionnée, entrez un mot de passe, le cas échéant, dans la colonne "Mot de passe dans la CPU" de l'onglet ouvert.
3. Sélectionner la commande de menu "Opérations > Diagnostic > Afficher les diagnostics de CPU". Vous pouvez également cliquer sur le bouton "Afficher le diagnostic de l'appareil"  de la barre d'outils et sélectionner "Afficher les diagnostics de CPU" dans le menu déroulant du bouton.

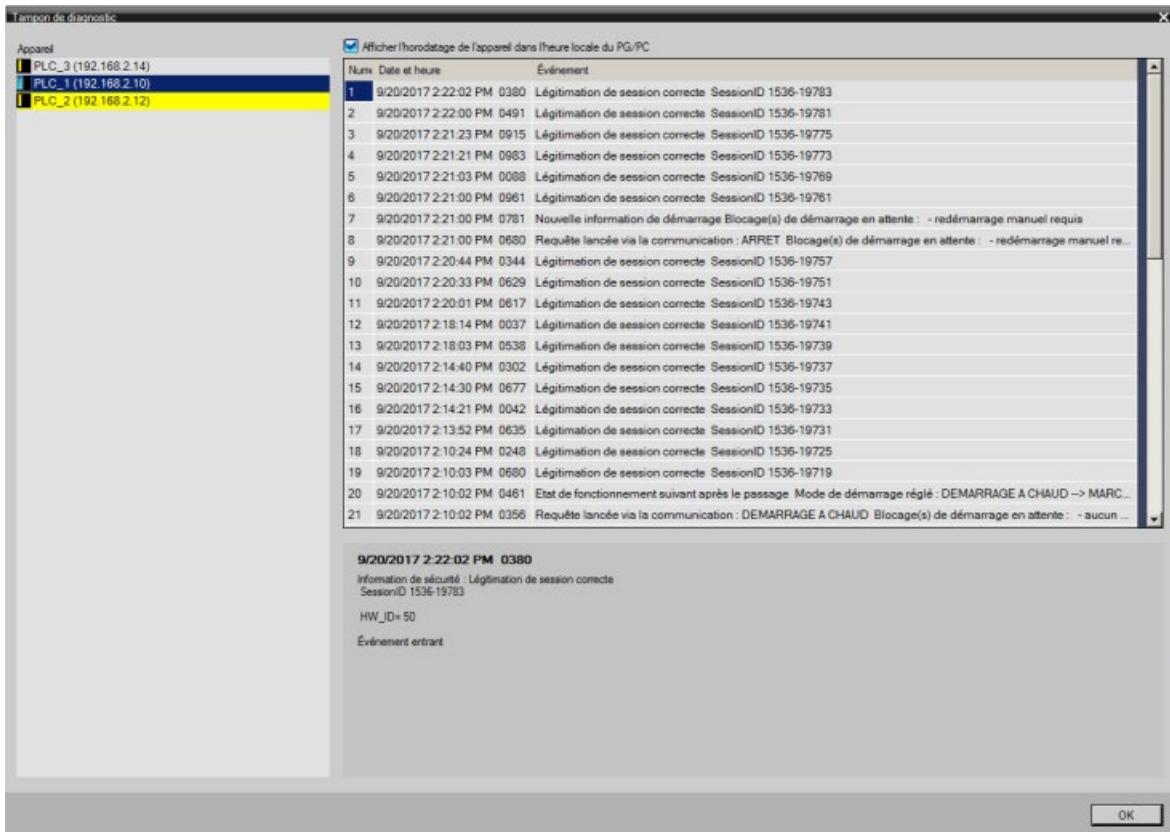
SIMATIC Automation Tool affiche alors une boîte de dialogue contenant les tampons de diagnostic des CPU sélectionnées. Vous pouvez sélectionner une CPU à partir de la liste des appareils à gauche pour voir le tampon de diagnostic de cette CPU.

Exemple de journal de diagnostic

Le tampon de diagnostic contient les types d'entrées suivants :

- Événement de diagnostic système (toutes les erreurs de CPU et de module)
- Changement d'état de la CPU (chaque mise sous tension, chaque passage en mode STOP, chaque passage en mode RUN)

Vous pouvez utiliser la case à cocher "Afficher l'horodatage de la CPU dans l'heure locale du PG/PC" pour afficher les horodatages dans l'heure locale ou dans l'heure UTC (Coordinated Universal Time).



4.21 Ordre d'exécution des opérations

Les entrées du tampon de diagnostic correspondent aux opérations des boutons de la barre d'outils ou aux commandes de menu. A chaque pression sur un bouton, SIMATIC Automation Tool ajoute une unique opération à la file d'attente pour chaque ligne d'appareil sélectionnée. Par exemple, si vous sélectionnez 20 CPU différentes et que vous cliquez sur le bouton RUN, SIMATIC Automation Tool ajoute 20 opérations RUN à la file d'attente.

Afin d'améliorer la performance, il est possible de lancer des threads séparés pour lancer et exécuter les opérations de la file d'attente. Vous assignez le nombre d'opérations simultanées sous Paramètres de communication (Page 110). Il n'est pas permis d'utiliser plusieurs fils (threads) pour lancer des tâches simultanément sur une CPU afin d'éviter toute concurrence entre les tâches. Par exemple, une tâche ne peut pas mettre la CPU en mode STOP alors qu'une autre tente de placer la même CPU en mode RUN.

Exemples d'exécution

Exemple 1 :

Si la file d'attente contient 10 tâches Passer en RUN pour différentes CPU, des threads multiples travaillent en parallèle pour mettre toutes les CPU en mode RUN. Comme les exécutions sont en parallèle, l'ordre dans lequel les CPU opèrent leur transition en RUN n'est pas garanti. Les vitesses de communication peuvent être différentes, et pour chaque CPU, la tâche peut se terminer plus ou moins rapidement.

Exemple 2 :

Vous pouvez mettre en file d'attente autant de tâches que vous le souhaitez. Par exemple, vous pouvez mettre 100 CPU en mode STOP en les sélectionnant toutes et en cliquant sur le bouton STOP. Une boîte de dialogue contenant une barre de progression s'affiche cependant, jusqu'à ce que les 100 tâches soient terminées. La boîte de dialogue bloque le démarrage d'une autre opération jusqu'à ce que ces opérations STOP soient terminées.

Enregistrer vos informations de la table Appareil

5.1 Enregistrer/Enregistrer sous - Table Appareil stockée au format sécurisé .sat

Utilisez la commande de menu "Fichier > Enregistrer/Enregistrer sous" ou cliquez sur le bouton Enregistrer  pour stocker les informations de la table Appareil dans un fichier chiffré .sat. Une fois que vous avez enregistré le projet SIMATIC Automation Tool, vous pouvez utiliser la commande de menu "Fichier > Ouvrir" ou le bouton Ouvrir  pour restaurer les informations de la table Appareil du projet. Le fichier de projet n'enregistre pas le mode de fonctionnement de l'appareil, l'état de sélection ou la confirmation des données d'état de sélection. Vous pouvez utiliser la commande Actualiser pour lire les états du mode de fonctionnement.

- Vous affectez le dossier à utiliser pour enregistrer les projets dans Paramètres des projets (Page 112).
- Vous devez fournir un mot de passe valide pour enregistrer un fichier projet .sat SIMATIC Automation Tool.
- Vous devez saisir le mot de passe correct pour rouvrir un fichier projet .sat SIMATIC Automation Tool existant.

Protection du fichier .sat SIMATIC Automation Tool

Protégez votre projet SIMATIC Automation Tool à l'aide d'un mot de passe fort. Les mots de passe fiables respectent les règles suivantes :

- Au moins dix caractères
- Mélange de lettres, de chiffres et de caractères spéciaux
- Ne correspondent pas à des entrées de dictionnaires.
- Ce ne sont pas des noms ni des identifiants pouvant être déduits de vos données personnelles.

Ne divulguez pas le mot de passe et changez-en fréquemment.

Exigences minimales relatives pour les mots de passe SIMATIC Automation Tool

Pour les mots de passe, SIMATIC Automation Tool applique les exigences minimales suivantes :

- Au moins dix caractères
- Mélange de lettres, de chiffres et de caractères spéciaux

Vos fichiers .sat sont protégés

Un mot de passe valide est nécessaire pour déchiffrer et ré-ouvrir un fichier .sat.

Compatibilité des fichiers de projet avec les versions précédentes

SIMATIC Automation Tool V3.1 prend en charge les opérations relatives à la sécurité qui étaient interdites dans les versions antérieures et les anciens fichiers de projet .sat ne contiennent pas les données de sécurité nécessaires.

L'ouverture des fichiers de projet V1.x n'est pas possible.

Lorsque vous ouvrez un fichier de projet V2.x - V3.0, vous êtes informé qu'un scan réseau doit être effectué avant d'ouvrir le fichier de projet. Une fois le scan réseau terminé, le fichier est ouvert et les données du fichier ouvert seront appliquées aux appareils identifiés lors du scan réseau.

5.2

Importation/Exportation - Table Appareil chargée de/stockées dans un format .csv ouvert

SIMATIC Automation Tool inclut les commandes de menu suivantes pour exporter et importer les tables Appareil :

- La commande de menu **Fichier>Exportation** enregistre la table Appareil au format texte .csv (valeurs séparées par des virgules).
- La commande de menu **Fichier>Importation** lit un fichier texte .csv et place les données dans la table Appareil de SIMATIC Automation Tool.

La première ligne de texte est une description de l'en-tête, elle est suivie d'une ou plusieurs lignes de données. Le texte de données doit correspondre au format attendu, avec 15 virgules (",") sur chaque ligne de texte. Ces 15 virgules séparent les 16 colonnes de données que vous voyez sur l'exemple d'exportation.

La table Appareil dans SIMATIC Automation Tool configure la communication avec un groupe d'appareils. Si vous placez des informations incorrectes dans les cellules d'une table Appareil ou dans un fichier .cvs importé, l'opération affectée peut échouer. Corrigez les données de l'appareil et renouvez l'opération.

Pour des raisons de sécurité, les mots de passe de la CPU ne sont pas exportés.

Remarque

Copier toutes les lignes et colonnes affichées dans la table Appareil

Si vous souhaitez créer un document qui affiche l'intégralité de votre réseau tel qu'indiqué dans la table Appareil, vous pouvez utiliser la table Appareil et MS Excel.

1. Développez toutes les lignes dans la table Appareil avec la commande **Edition>Développer tous les appareils**.
 2. Pressez Ctrl-A pour mettre en surbrillance toutes les lignes et colonnes.
 3. Pressez Ctrl-C pour copier les données de la table Appareil dans le presse-papiers Windows.
 4. Pressez Ctrl-V et collez les données du presse-papiers dans une feuille de calcul MS Excel.
-

Remarque

Importer un fichier .csv à partir de versions antérieures de SIMATIC Automation Tool

Le nombre de colonnes de la table Appareil et le nom des colonnes est modifié dans SIMATIC Automation Tool V3.1 par rapport aux versions précédentes. Par conséquent, vous ne pouvez pas importer de fichiers .csv créés ou formatés pour les versions précédentes.

Vous trouverez sous Paramètres d'importation/exportation (Page 120) le chemin des fichiers pour les opérations d'importation et d'exportation.

Exemple d'exportation

L'image suivante montre le format texte d'un fichier .csv exporté de SIMATIC Automation Tool.

Notez que la commande Exporter ne fournit que la liste des appareils directement connectés au même sous-réseau que le PG/PC qui exécute SIMATIC Automation Tool ; elle n'affiche pas les fichiers ou appareils de la CPU connectés derrière des CPU ou des modules d'interface.

L'image suivante montre le même fichier texte ouvert dans Microsoft Excel.

Lorsque vous utilisez SIMATIC Automation Tool et que vous importez un fichier .csv exporté, procédez comme suit pour afficher tous les fichiers et appareils de la CPU :

1. Importez le fichier avec la commande Fichier>Importer.
 2. Saisissez de nouveau tous les mots de passe des CPU protégées.
 3. Sélectionnez tous les appareils dans la table Appareil.
 4. Actualisez la table Appareil à l'aide de la commande Opérations> Scanner le réseau> Rafraîchir l'état des appareils sélectionnés.

Menu, barre d'outils, raccourcis clavier et informations de référence

6.1

Menu principal

SIMATIC Automation Tool inclut les menus suivants pour effectuer les opérations sur les appareils.

- Fichier (Page 103)
- Modifier (Page 104)
- Opérations (Page 105)
- Options (Page 107)
- Outils (Page 120)
- Aide (Page 121)

Si vous appuyez sur la touche Alt, la lettre soulignée indique la touche Alt que vous pouvez utiliser pour activer une commande de menu ou de sous-menu.

De plus, vous pouvez activer certaines des commandes de menu avec des combinaisons de touches de raccourci (Page 124).

6.1.1

Menu Fichier

Icône outil	Commandes de menu	Description
	Nouveau	Créer un nouveau projet SIMATIC Automation Tool
	Ouvrir	Affiche une boîte de dialogue "Ouvrir" permettant de naviguer vers un dossier, de sélectionner un fichier de projet .sat et de saisir un mot de passe pour ouvrir un fichier projet protégé. La boîte de dialogue "Ouvrir" affiche le dossier du projet (Page 112), mais vous pouvez naviguer vers n'importe quel emplacement pour un projet.
	Enregistrer	Enregistre (Page 99) les données de la table Appareil dans un fichier .sat. En l'absence de nom de fichier, l'opération utilise la commande "Enregistrer sous". Le dossier du projet (Page 112) est le dossier par défaut pour l'enregistrement des projets.
	Enregistrer sous	Enregistre les données de la table Appareil dans un fichier .sat. Vous pouvez parcourir un dossier, donner un nom de fichier .sat et définir un mot de passe pour protéger le fichier projet.
	Importer	Importe (Page 100) les données provenant d'un fichier au format .csv vers la table Appareil.
	Exporter	Exporte (Page 100) les données de la table Appareil dans un fichier au format .cvs.
	Quitter	Ferme l'application. L'opération "Enregistrer" est exécutée si le projet a été modifié depuis le dernier enregistrement.

6.1.2 Menu Edition

Icône outil	Commandes de menu	Description
	Couper	Couper les données sélectionnées et les copier dans le presse-papier. Les données du presse-papier sont compatibles avec Excel, ce qui permet un échange entre les deux applications. Les cellules en lecture seule ne sont pas supprimées.
	Copier	Coller les données sélectionnées à partir du presse-papier dans un format compatible Excel.
	Coller	Coller les données du presse-papier dans les champs sélectionnés de SIMATIC Automation Tool. Les cellules en lecture seule ne sont pas copiées.
	Sélectionner	
	• Sélectionner les lignes	Sélectionner les lignes de la table Appareil souhaitées.
	• Désélectionner les lignes	Désélectionner les lignes de la table Appareil souhaitées.
	Développer	
	• Appareil	Développer l'appareil actif
	• Tous les appareils sélectionnés	Développer tous les appareils sélectionnés
	• Tous les appareils	Développer toutes les lignes des appareils et des modules.
	Réduire	
	• Appareil	Réduire l'appareil actif
	• Tous les appareils sélectionnés	Réduire tous les appareils sélectionnés
	• Tous les appareils	Réduire toutes les lignes des appareils et des modules.
	Ajouter	
	• Appareil	Ajouter un nouvel appareil sur la ligne sélectionnée et décaler les lignes d'appareil suivantes vers le bas. Vous pouvez utiliser cette commande pour ajouter rapidement l'appareil à la table Appareil. Si vous utilisez cette commande pour ajouter un appareil situé derrière un routeur, le nom de cet appareil s'affiche en bleu. Le bleu signifie que les opérations basées sur l'adresse MAC (identifier l'appareil, définir l'adresse IP, définir le nom PROFINET) ne sont pas possibles et les cellules correspondantes de la table Appareil sont désactivées.
	Supprimer	
	• Texte de la cellule	Supprimer le contenu de la cellule active
	• Appareil	Supprimer le contenu de la ligne appareil active
	• Tous les appareils sélectionnés	Supprime le contenu de tous les appareils sélectionnés
	• Tous les appareils	Supprime le contenu de toutes les lignes appareil actives

Icône outil	Commandes de menu	Description
	Rafraîchir	
	• Appareil	Actualiser l'appareil actif
	• Tous les appareils sélectionnés	Actualiser tous les appareils sélectionnés
	• Tous les appareils	Actualiser tous les appareils

6.1.3 Menu Opérations

Icône outil	Commandes de menu	Description
	Scanner le réseau	
	• Scanner le réseau complet	Scanner le réseau de l'appareil (Page 49)
	• Actualiser l'état des appareils sélectionnés	Actualiser les appareils sélectionnés dans la table Appareil (Page 49)
	MARCHE	Mettre les CPU sélectionnées en mode RUN. (Page 60)
	ARRÊT	Mettre les CPU sélectionnées en mode STOP. (Page 60)
	Mettre à jour	
	• Configurer l'adresse IP	Mettre à jour la CPU avec les informations d'adresse IP du ou des appareil(s) sélectionné(s) (Page 56)
	• Définir le nom PROFINET	Mettre à jour la CPU avec le nom PROFINET du ou des appareil(s) sélectionné(s) (Page 58)
	• Mise à jour du programme	Mettre à jour le programme CPU ou le système d'exploitation et le logiciel exécutif de l'appareil IHM pour les appareils sélectionnés (Page 63)
	• Mise à jour du firmware	Mettre à jour le firmware de la CPU avec le(s) fichier(s) de mise à jour du programme pour le ou les appareil(s) sélectionné(s) (Page 78)
	Identifier	Faire clignoter les LED des appareils CPU ou les écrans IHM. (Page 61) Permet d'identifier l'emplacement physique d'un appareil.
	Réinitialiser	
	• RAZ mémoire	Réinitialiser la mémoire des appareils sélectionnés. (Page 88)

6.1 Menu principal

Icône outil	Commandes de menu	Description
	<ul style="list-style-type: none"> • Restaurer les paramètres d'usine 	Restaurer les paramètres d'usine des appareils sélectionnés. (Page 89)
	<ul style="list-style-type: none"> • Formater la carte mémoire 	Formater la carte mémoire dans les appareils sélectionnés (Page 91)
	Diagnostic	
	<ul style="list-style-type: none"> • Afficher les diagnostics de CPU 	Afficher le tampon de diagnostic pour une CPU sélectionnée. (Page 96)
	<ul style="list-style-type: none"> • Récupérer les Service Data 	Récupérer les Service Data des appareils sélectionnés (Page 93)
	Régler l'heure	Régler l'heure des CPU sélectionnées sur celle du PG/PC. (Page 95)
	Sauvegarder/Restaurer	
	<ul style="list-style-type: none"> • Sauvegarder l'appareil dans un fichier 	Effectuer une sauvegarde de tous les appareils sélectionnés. (Page 83) The SIMATIC Automation Tool enregistre un fichier de sauvegarde pour chaque appareil sélectionné.
	<ul style="list-style-type: none"> • Restaurer l'appareil depuis un fichier 	Restaurer les données à partir de fichiers de sauvegarde vers le/les appareil(s) correspondant(s). (Page 83)
	Opérations sur fichier	Remarque : les opérations sur les fichiers ne s'appliquent qu'aux fichiers de CPU.
	<ul style="list-style-type: none"> • Charger les Data logs 	Charger les fichiers Data log sélectionnés sur votre PG/PC. (Page 76)
	<ul style="list-style-type: none"> • Supprimer des Data logs 	Supprimer les fichiers Data log sélectionnés. (Page 76)
	<ul style="list-style-type: none"> • Charger les recettes 	Charger les fichiers de recette depuis les CPU sélectionnées vers votre PG/PC. (Page 73)
	<ul style="list-style-type: none"> • Ajouter/remplacer une recette 	Ajouter/remplacer les fichiers de recette depuis le PG/PC vers les CPU sélectionnées. (Page 73) Cette opération permet d'ajouter des recettes qui ne sont pas encore présentes dans la CPU ; dans le cas contraire, les anciennes recettes sont remplacées.
	<ul style="list-style-type: none"> • Supprimer les recettes 	Supprimer les recettes sélectionnées des CPU correspondantes. (Page 73)

6.1.4 Menu Options

6.1.4.1 Menu Options

Le menu Options contient les commandes de menu suivantes :

Icône d'outil	Commande de menu	Description
	Paramètres	Ouvre la boîte de dialogue Paramètres, où vous pouvez définir les paramètres par défaut pour les catégories suivantes : <ul style="list-style-type: none"> • Général (Page 109) • Communications (Page 110) • Projets (Page 112) • Mise à jour du firmware (Page 113) • Mise à jour du programme (Page 114) • Service Data (Page 115) • Sauvegarde / Restauration (Page 116) • Recettes (Page 117) • Data Logs (Page 118) • Event Log (Page 119) • Importer / Exporter (Page 120)
	Démarrer Automation License Manager	Lance l'Automation License Manager avec lequel vous pouvez obtenir une licence pour SIMATIC Automation Tool.

6.1.4.2 Noms de chemin de SIMATIC Automation Tool

Les exemples de chemin d'accès pour la boîte de dialogue "Options > Paramètres" montrent les noms de chemin des dossiers dans C:\Users\MyAccount\SIMATIC Automation Tool\, où "MyAccount" représente votre identifiant utilisateur.

Si vous utilisez la fonction de navigation dans la table Appareil pour localiser un fichier tel qu'un fichier de mise à jour du firmware ou un fichier de mise à jour du programme, vous ne voyez pas votre identifiant utilisateur apparaître sous forme de dossier sous "Utilisateurs". Au lieu de cela, vous voyez le dossier "Mes documents". Lorsque vous utilisez la fonction de navigation de SIMATIC Automation Tool, naviguer vers le dossier "Mes documents" équivaut à naviguer vers le dossier avec votre identifiant d'utilisateur ("MyAccount").

Depuis l'Explorateur Windows, le dossier Documents sous le dossier Bibliothèques équivaut également au dossier "Mes documents" et au dossier "MyAccount".

6.1 Menu principal

Les parties suivantes de la boîte de dialogue de la commande de menu "Options > Paramètres" fournissent un chemin d'accès par défaut que vous pouvez modifier :

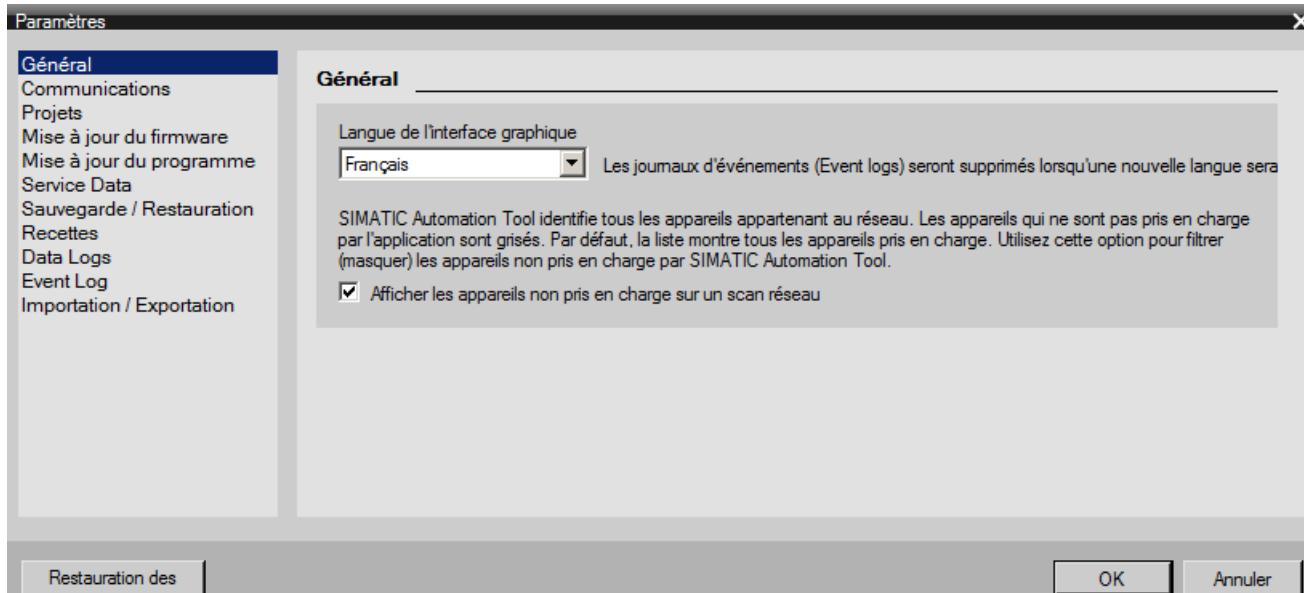
- Projets (Page 112)
- Mise à jour du firmware (Page 113)
- Mise à jour du programme (Page 114)
- Service Data (Page 115)
- Sauvegarder/Restaurer (Page 116)
- Recettes (Page 117)
- Data Logs (Page 118)
- Event Log (Page 119)

6.1.4.3 Paramètres généraux

Vous pouvez sélectionner la langue de l'interface utilisateur : anglais, allemand, français, espagnol ou italien

Sélectionnez la case à cocher pour afficher les appareils non pris en charge dans un scan réseau (Page 49). SIMATIC Automation Tool affiche (en gris dans la table Appareil) les appareils non pris en charge comme étant désactivés.

Si vous désélectionnez la case à cocher, SIMATIC Automation Tool filtre les appareils non pris en charge dans la table Appareil.



Remarque

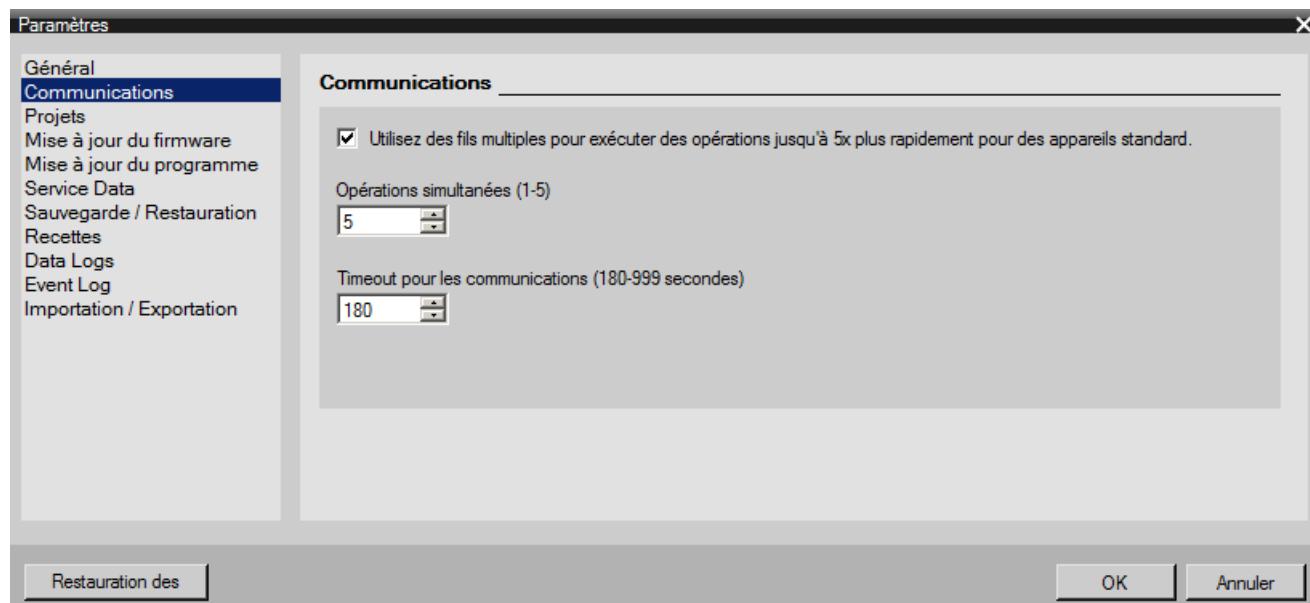
La modification de la langue de l'interface utilisateur efface l'Event Log.

Lorsque vous modifiez la langue de l'interface utilisateur, SIMATIC Automation Tool efface l'Event Log.

6.1 Menu principal

6.1.4.4 Paramètres de communication

Vous utilisez les options de communication pour définir les options liées au traitement multi-thread.



Utilisation de plusieurs threads pour les opérations

Si le réseau possède une topologie en étoile dans laquelle chaque CPU est raccordée directement au PG/PC via un commutateur Ethernet, vous pouvez utiliser l'option de traitement multifil en toute sécurité.

Si le réseau possède une topologie en chaîne, vous devez désactiver cette option, afin d'éviter qu'une CPU entraîne une coupure de communication vers les autres appareils. Une topologie en chaîne, par exemple, consisterait en des connexions en chaîne du PG/PC à la CPU 1, à la CPU 2, à la CPU 3, etc.

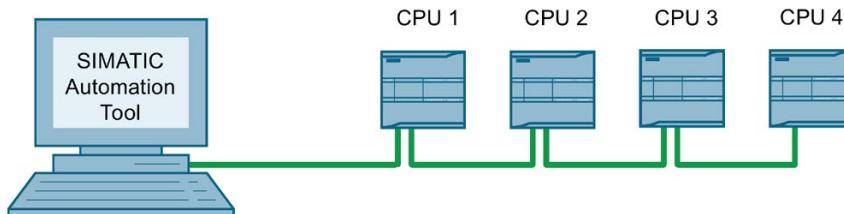


Figure 6-1 Exemple : Topologie en chaîne

Avec le traitement multi-thread, un thread qui fait redémarrer la CPU 1 perturbe une opération en cours sur la CPU 2 ou toute autre CPU de la chaîne. Notez qu'une topologie en chaîne peut aussi être mise en œuvre avec des modules CM ou CP.

Opérations simultanées

Il est possible d'améliorer les performances de SIMATIC Automation Tool en permettant l'exécution simultanée des opérations sur plusieurs appareils sur des threads différents.

Remarque

Problèmes de communication avec SIMATIC Automation Tool

Par exemple, vous envoyez une commande d'opération à plusieurs appareils, mais un des appareils ne termine pas l'opération et renvoie une erreur de communication pour cette CPU. Cependant, les autres appareils communiquent et exécutent l'opération comme prévu. Si vous rencontrez ce problème, réduisez le nombre de fils/connexions simultanés. Fermez et redémarrez SIMATIC Automation Tool, puis essayez à nouveau l'opération groupée.

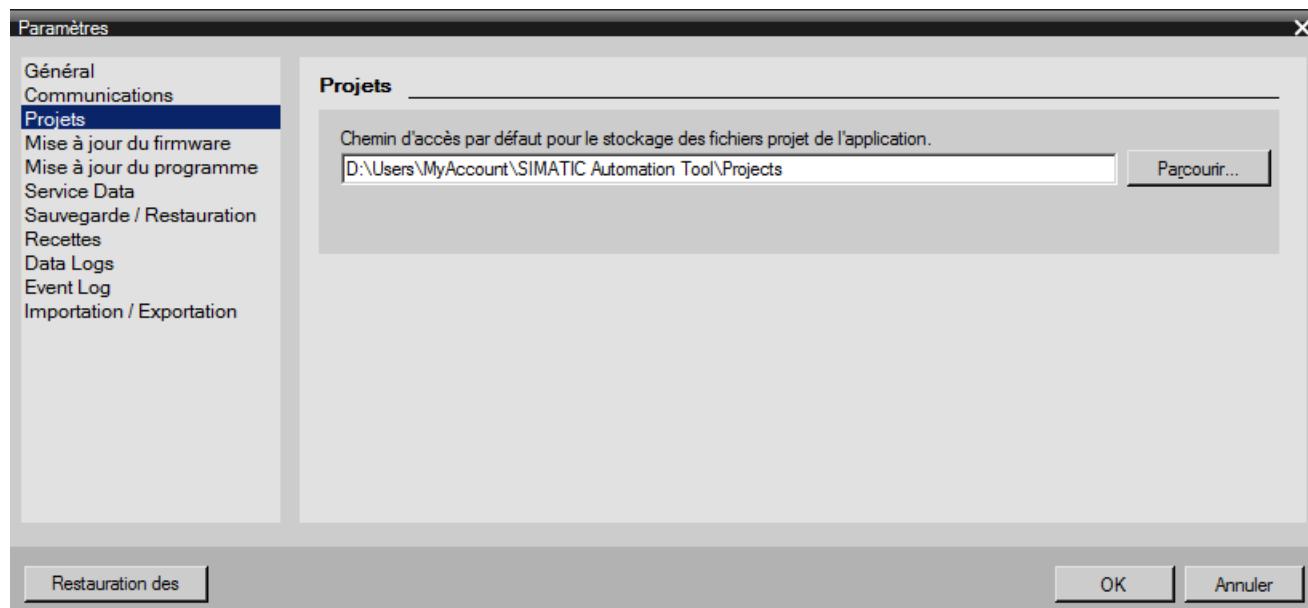
Timeout pour les communications

Si vous envoyez une commande d'opération à un appareil et que le débit de transfert de données est très bas, une erreur de dépassement de temps de la communication peut être renvoyée. Si vous rencontrez ce problème, augmentez la durée d'expiration affectée aux opérations de communication.

6.1.4.5 Paramètres des projets

Vous pouvez accepter le chemin d'accès par défaut pour l'enregistrement des données de projet (Page 99) SIMATIC Automation Tool ou en définir un nouveau.

Votre chemin peut avoir une lettre de lecteur différente et "MyAccount" représente l'identifiant de l'utilisateur actuel (Page 107).

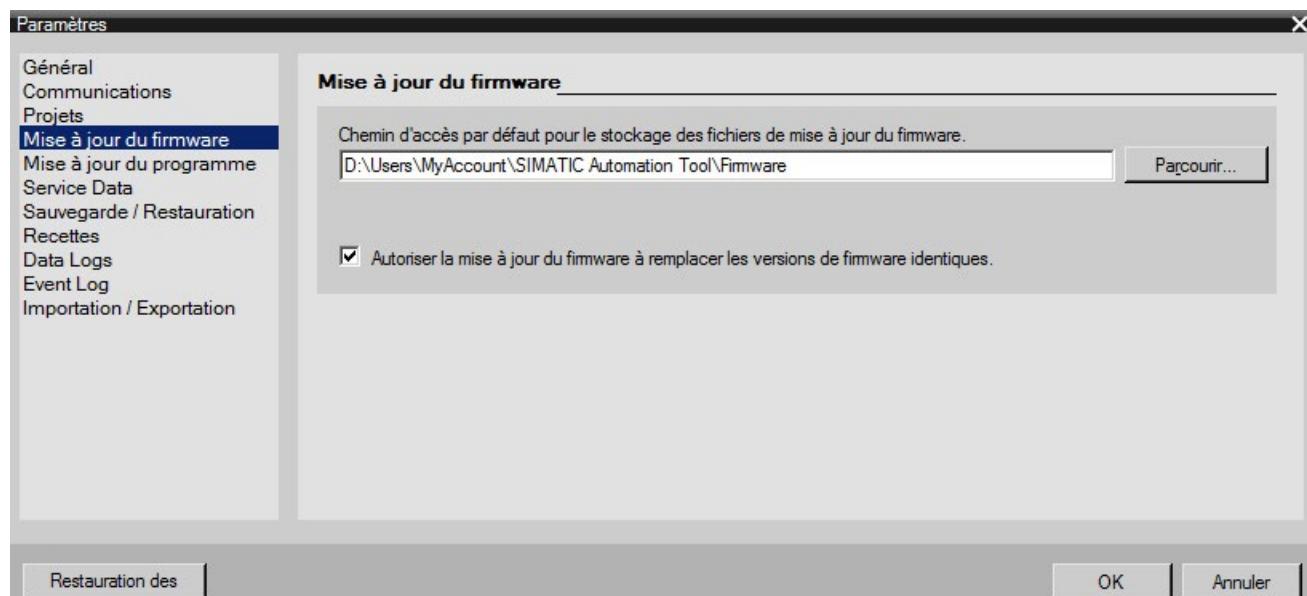


6.1.4.6 Paramètres de mise à jour du firmware

Vous pouvez accepter le chemin d'accès par défaut vers les fichiers de mise à jour du firmware (Page 78) ou en définir un nouveau.

Votre chemin peut avoir une lettre de lecteur différente et "MyAccount" représente l'identifiant de l'utilisateur actuel (Page 107).

Activez cette case à cocher pour autoriser ou interdire la mise à jour du firmware avec une version identique. Interdire le remplacement d'une version de firmware identique permet d'économiser le temps de traitement en évitant les opérations inutiles.

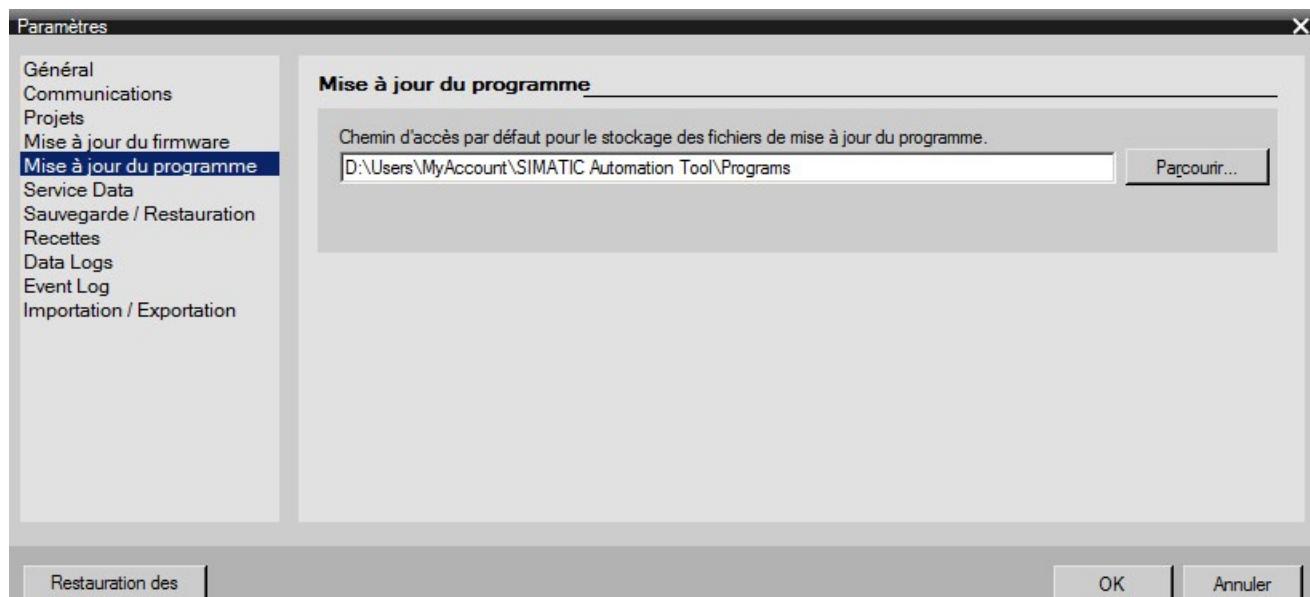


6.1 Menu principal

6.1.4.7 Paramètres de mise à jour des programmes

Vous pouvez accepter le chemin d'accès par défaut vers les fichiers programme (Page 63) ou en définir un nouveau.

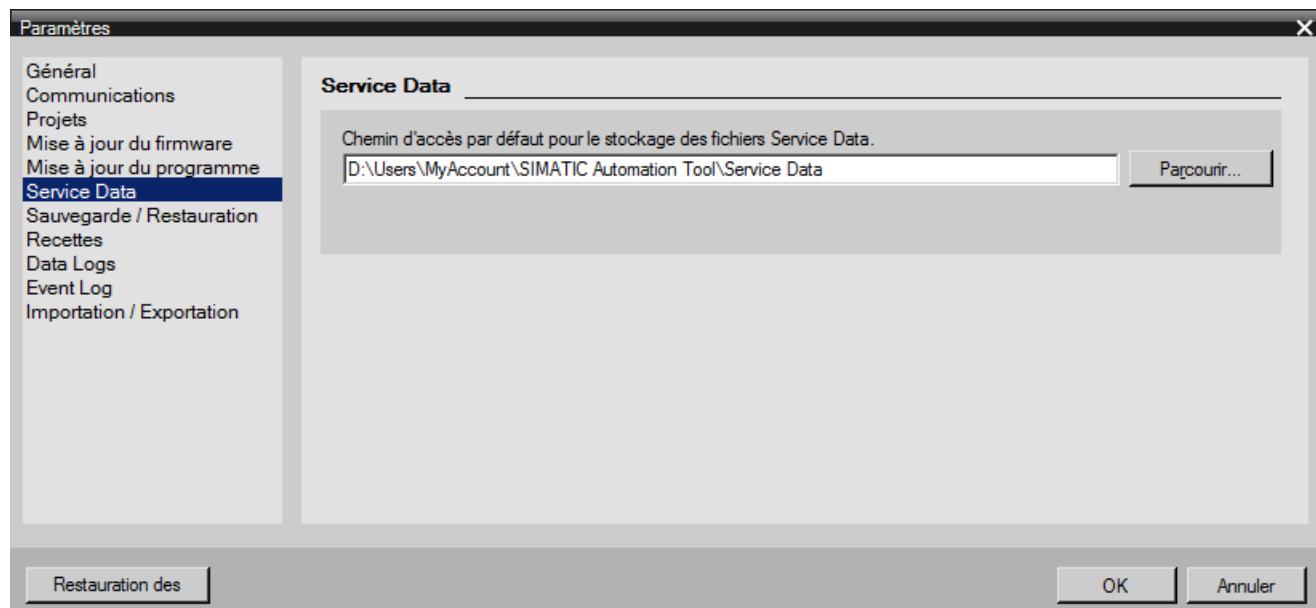
Votre chemin peut avoir une lettre de lecteur différente et "MyAccount" représente l'identifiant de l'utilisateur actuel (Page 107).



6.1.4.8 Paramètres des Service Data

Vous pouvez accepter le chemin d'accès par défaut vers les fichiers de Service Data (Page 93) ou en définir un nouveau.

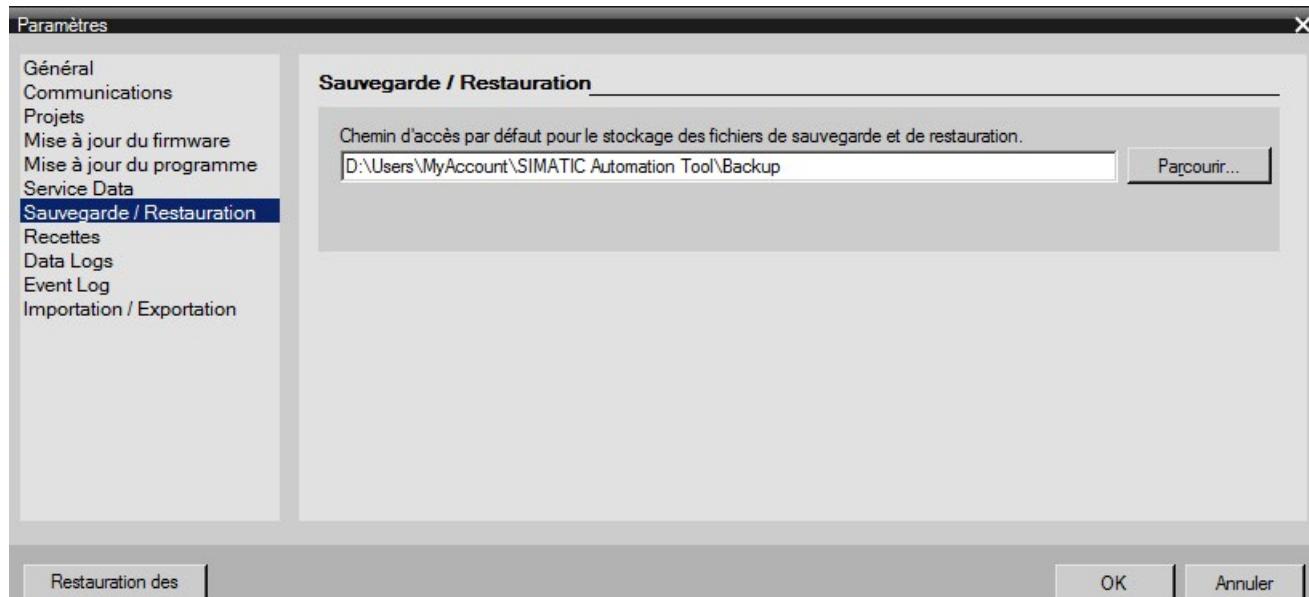
Votre chemin peut avoir une lettre de lecteur différente et "MyAccount" représente l'identifiant de l'utilisateur actuel (Page 107).



6.1.4.9 Paramètres de sauvegarde/restauration

Vous pouvez accepter le chemin d'accès par défaut vers les fichiers de sauvegarde et de restauration (Page 83) ou en définir un nouveau.

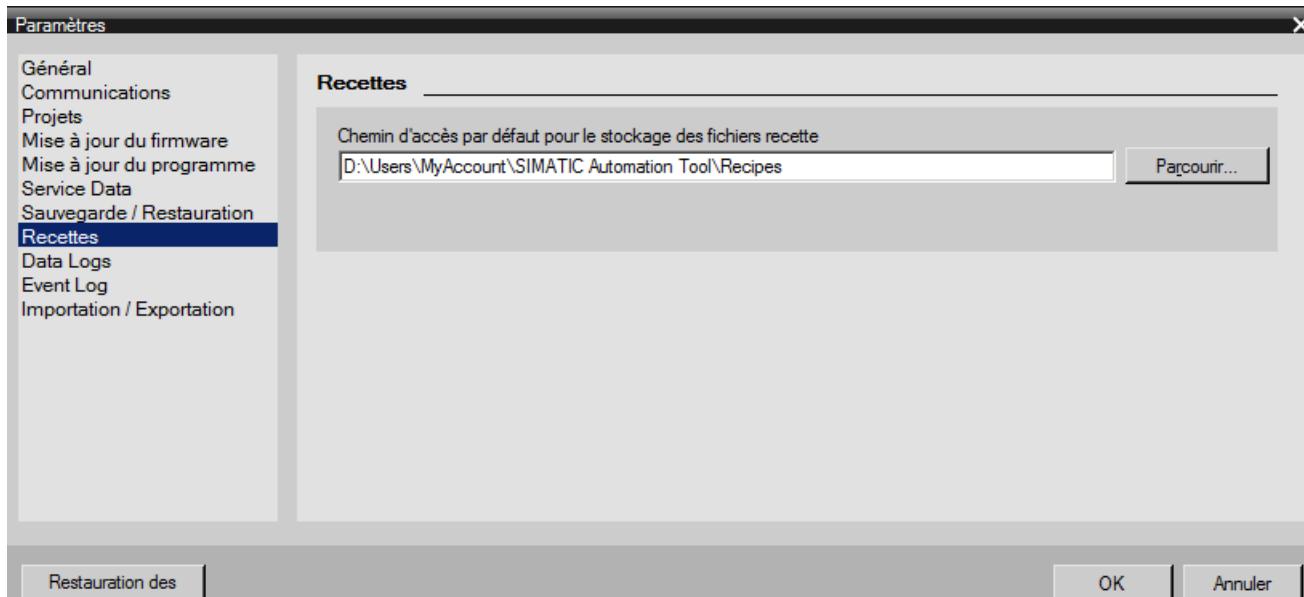
Votre chemin peut avoir une lettre de lecteur différente et "MyAccount" représente l'identifiant de l'utilisateur actuel (Page 107).



6.1.4.10 Paramètres des recettes

Vous pouvez accepter le chemin d'accès par défaut vers les fichiers recette (Page 73) ou en définir un nouveau.

Votre chemin peut avoir une lettre de lecteur différente et "MyAccount" représente l'identifiant de l'utilisateur actuel (Page 107).



6.1 Menu principal

6.1.4.11 Paramètres des journaux de données (Data logs)

Vous pouvez accepter le chemin d'accès par défaut vers les fichiers Data Log (Page 76) ou en définir un nouveau.

Votre chemin peut avoir une lettre de lecteur différente et "MyAccount" représente l'identifiant de l'utilisateur actuel (Page 107).



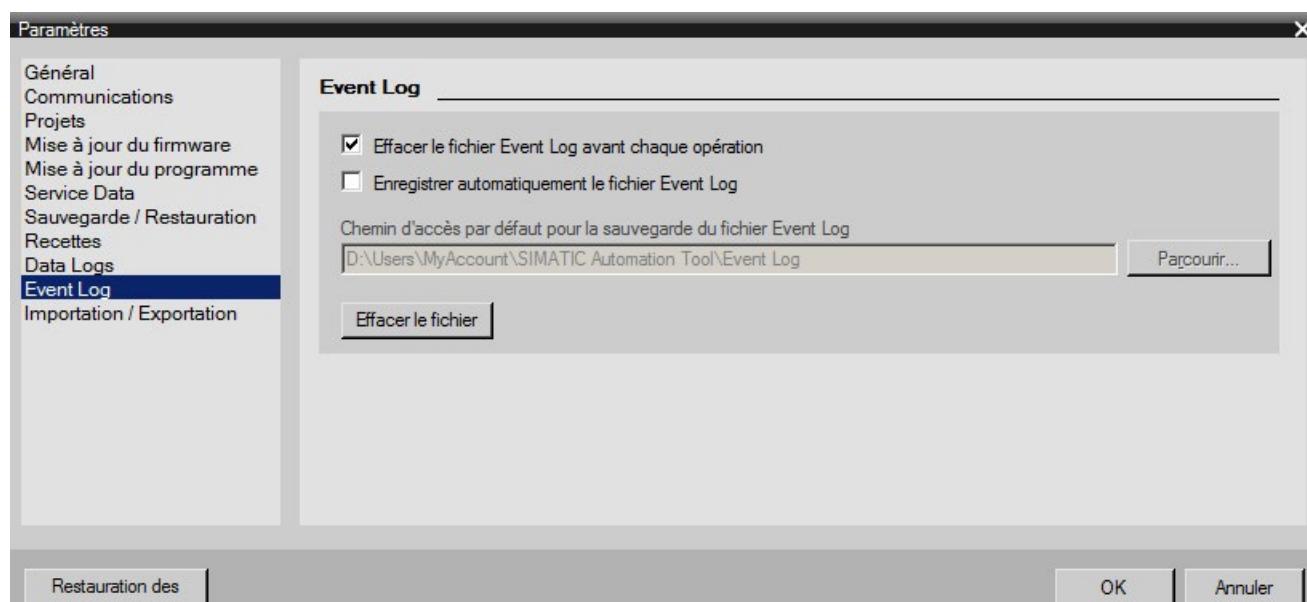
6.1.4.12 Paramètres de l'Event Log

Par défaut, SIMATIC Automation Tool efface l'Event Log (Page 42) au démarrage de toute opération d'appareil. Vous pouvez désélectionner "Effacer le fichier Event Log avant chaque opération" si vous souhaitez désactiver le paramètre par défaut.

Lorsque vous sélectionnez "Enregistrer automatiquement le fichier Event Log", vous pouvez accepter le chemin par défaut ou définir un autre chemin.

Votre chemin peut avoir une lettre de lecteur différente et "MyAccount" représente l'identifiant de l'utilisateur actuel (Page 107). SIMATIC Automation Tool enregistre ensuite chaque message dans la fenêtre Event Log dans le fichier "EventLogFile.csv". Lorsque vous fermez et ouvrez à nouveau SIMATIC Automation Tool, la connexion au fichier Event Log est rétablie automatiquement.

Vous pouvez effacer le contenu du fichier Event Log en cliquant sur le bouton "Effacer le fichier journal". Ceci efface le contenu, mais ne supprime pas le fichier.

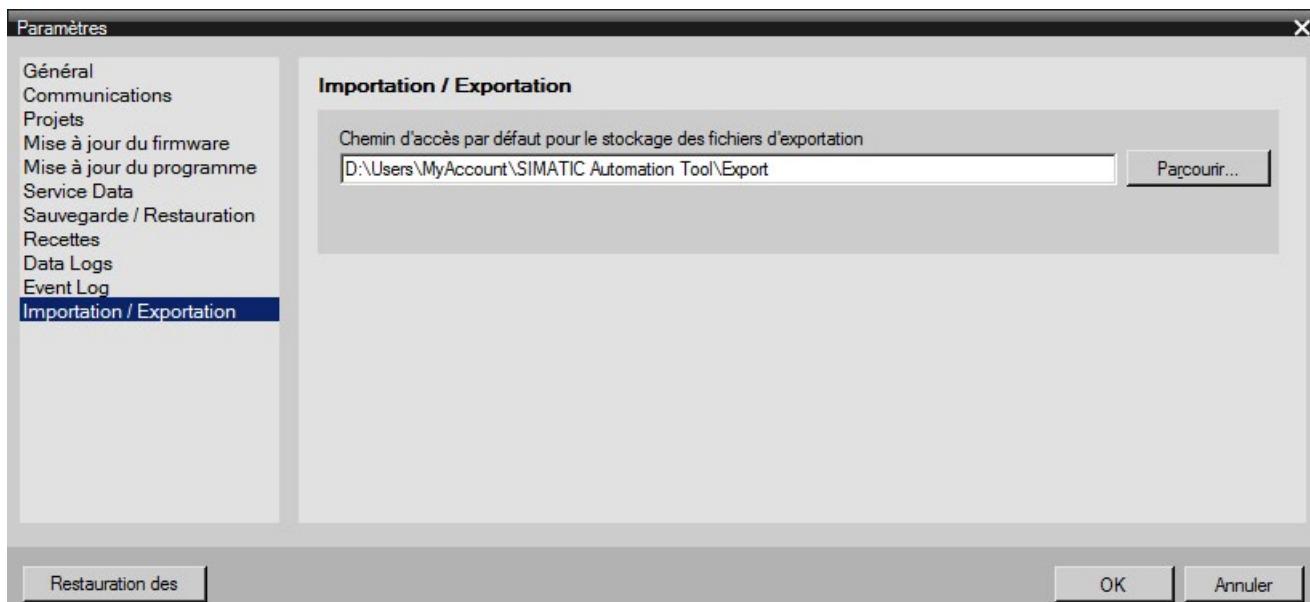


6.1 Menu principal

6.1.4.13 Paramètres d'importation/exportation

Vous pouvez accepter le chemin d'accès par défaut des fichiers d'importation / exportation (Page 100) ou en définir un nouveau.

Votre chemin peut avoir une lettre de lecteur différente et "MyAccount" représente l'identifiant de l'utilisateur actuel (Page 107).



6.1.5 Menu Outils

Menu Outils

Le menu Outils est également accessible à partir de l'icône  de la barre d'outils.

Icône outil	Commandes de menu	Description
	Rechercher des mises à jour du firmware	Rechercher les mises à jour du firmware lorsque vous n'avez sélectionné qu'un appareil. SIMATIC Automation Tool affiche la page Web avec les fichiers de mise à jour du firmware pour l'appareil sélectionné.
	Précharger les fichiers de mise à jour du firmware	Rechercher les fichiers de mise à jour du firmware pour les appareils sélectionnés. SIMATIC Automation Tool précharge les fichiers de mise à jour du firmware dans la liste déroulante "Nouvelle version firmware" pour les appareils sélectionnés.

6.1.6 Menu Aide

Icône d'outil	Commande de menu	Description
	Voir guide utilisateur	Ouvre le guide de l'utilisateur de SIMATIC Automation Tool.
	À propos de SIMATIC Automation Tool	Affiche la boîte de dialogue À propos de qui contient les informations suivantes : <ul style="list-style-type: none">• Nom du produit : Le texte "(pas de licence)" à côté du nom du produit indique que vous ne disposez pas d'une licence, ce qui limite les fonctionnalités disponibles.• Version• Le bouton Rechercher des mises à jour vous permet de trouver les mises à jour de SIMATIC Automation Tool.

6.2 Icônes de la barre d'outils

Icône outil	Description
	Nouveau: Créer un nouveau projet SIMATIC Automation Tool avec l'extension ".sat".
	Ouvrir : Afficher une boîte de dialogue "Ouvrir" permettant de naviguer vers un dossier, de sélectionner un fichier de projet et de saisir un mot de passe pour ouvrir le fichier projet crypté.
	Enregistrer les données du projet ouvert dans un fichier. En l'absence de nom de fichier et de mot de passe, la boîte de dialogue "Enregistrer sous" s'affiche.
	Couper les données sélectionnées et les copier dans le presse-papier. Les données du presse-papier sont compatibles avec Excel, ce qui permet un échange entre les deux applications.
	Couper les données sélectionnées et les copier dans le presse-papier.
	Coller les données du presse-papier dans les champs sélectionnés.
	Scanner le réseau sélectionné, avec les options suivantes : <ul style="list-style-type: none"> Scanner l'interface réseau sélectionnée pour rechercher des CPU et des modules accessibles. Actualiser l'état de tous les appareils dans la table Appareil
	RUN : Met les CPU sélectionnées en mode RUN.
	STOP : Met les CPU sélectionnées en mode STOP.
	Mettre à jour les appareils avec les données de SIMATIC Automation Tool à partir de l'une des options suivantes : <ul style="list-style-type: none"> Configurer l'adresse IP Définir le nom PROFINET Mise à jour du programme Mise à jour du firmware Vous devez sélectionner les onglets correspondants de la table Appareil pour entrer les données à mettre à jour.
	Identifier les appareils avec des LED / écrans IHM clignotants sur les appareils sélectionnés. Permet d'identifier l'emplacement physique d'un appareil.
	Fonctions de réinitialisation des appareils sélectionnés : <ul style="list-style-type: none"> RAZ mémoire Restaurer les paramètres d'usine Formater la carte mémoire
	Accéder aux informations de diagnostic : <ul style="list-style-type: none"> Lire le tampon de diagnostic d'une CPU Récupérer les Service Data depuis les appareils sélectionnés
	Régler l'heure : Régler l'heure système des CPU sélectionnées sur celle du PG/PC.

Icône outil	Description
	<p>Fonctions Sauvegarde et Restauration :</p> <ul style="list-style-type: none"> Créer des fichiers de sauvegarde pour les CPU et les appareils IHM sélectionnés. Restaurer les appareils sélectionnés à partir de fichiers de sauvegarde
	<p>Opérations sur fichier :</p> <ul style="list-style-type: none"> Charger ou supprimer les Data logs Charger, ajouter, remplacer ou supprimer des recettes <p>Les opérations de suppression nécessitent que la CPU soit en mode STOP.</p>
	Rechercher les mises à jour du firmware ou précharger les fichiers de mise à jour du firmware
D-Link DUB-E100 USB 2.0 Fast Ethernet Adapter.TCPIP.1	Liste déroulante de l'interface réseau : Sélectionner l'interface réseau Ethernet qui est connectée au réseau de commande industrielle.

6.3 Raccourcis clavier

SIMATIC Automation Tool prend en charge les touches de raccourcis suivantes pour naviguer dans la table Appareil et utiliser certaines commandes de menu (Page 103).

CTRL+Haut	Bascule entre les onglets, de gauche à droite
CTRL+Bas	Bascule entre les onglets, de droite à gauche
Ctrl+A	Sélectionne toute la table
Ctrl+C	Copie la sélection dans le presse-papiers
Ctrl+O	Affiche la boîte de dialogue ouvrir pour ouvrir un nouveau fichier projet
Ctrl+S	Affiche la boîte de dialogue "Enregistrer sous"
Ctrl+V	Colle le contenu du presse-papier au point d'insertion et écrase la sélection.
Ctrl+X	Coupe les cellules sélectionnées
CTRL+Z	Annuler la dernière modification ou supprimer l'action
FLÈCHE DIRECTIONNELLES	Déplacer les cellules vers le haut, le bas, la gauche ou la droite.
MAJ + FLÈCHE DIRECTIONNELLES	Élargit la sélection
Suppr	Supprime le contenu de la cellule active
Entrée	Termine l'édition de la cellule et valide les données
Échap	Annule l'édition de la cellule et rétablit la valeur d'origine
Origine	Aller au début de la cellule
Ctrl+Début	Aller au début de la table
Fin	Aller à la fin de la cellule
Ctrl+Fin	Aller à la fin de la table
Page suivante	Descendre un écran sur la table
Page haut	Remonter un écran sur la table
BARRE D'ESPACEMENT	Activer ou désactiver la case à cocher des lignes ou des lignes multiples, si sélectionnée
TABULATION	Se déplacer d'une cellule vers la droite
Alt+F4	Quitter

6.4

Définition du programme de sécurité

Un programme de sécurité est un programme que vous créez dans STEP 7 et qui comprend des blocs de sécurité. Si vous avez installé STEP 7 Safety Advanced, le dossier Blocs programme inclut automatiquement les blocs de sécurité chaque fois que vous ajoutez une CPU de sécurité à votre projet STEP 7. Lorsque vous téléchargez ce programme dans une CPU de sécurité, il s'agit d'un programme de sécurité.

7.1

Introduction à l'API

L'API SIMATIC Automation Tool permet de créer des applications personnalisées sur la base des fonctionnalités disponibles dans l'application SIMATIC Automation Tool. Vous pouvez optimiser une application personnalisée pour combiner les opérations et prendre en charge les workflows utilisateur spécifiques à votre réseau d'automatisation industrielle.

7.2

Licence logicielle API et compatibilité des versions

Licence logicielle requise pour V3.0 et versions ultérieures

Vous devez disposer d'une licence SIMATIC Automation Tool valide pour utiliser l'API (Application Programming Interface / Interface de programmation d'application).

Une fois que vous avez installé SIMATIC Automation Tool, fourni une licence et accepté l'intégralité des termes de la licence, vous avez accès à l'API pour votre propre programmation. Vous ne pouvez ni copier ni mettre à disposition d'un tiers tout ou partie du logiciel SIMATIC Automation Tool, y compris les fichiers .exe ou .dll de l'API.

Si vous souhaitez mettre votre application d'interface utilisateur personnalisée à disposition d'un tiers, celui-ci doit également disposer d'une licence valide pour SIMATIC Automation Tool V3.1.

Compatibilité avec les versions précédentes

L'API V3.1 n'est pas compatible avec les versions précédentes. Vous devez réécrire les programmes écrits avec les versions de l'API antérieures à la version V3.1.

Voir aussi

Exemple de réseau (Page 251)

7.3 Conception d'une application d'interface utilisateur pour des appareils de sécurité et des opérations relatives à la sécurité

7.3.1 Prise en charge d'opérations relatives à la sécurité et d'appareils de sécurité par l'API

L'API SIMATIC Automation Tool V3.1 prend en charge les opérations relatives à la sécurité qui n'étaient pas possibles avec les versions antérieures de SIMATIC Automation Tool :

- Mise à jour du programme
- Restaurer l'appareil à partir d'un fichier de sauvegarde
- Restaurer les paramètres d'usine
- Formater la carte mémoire

Remarque

Le manuel "*SIMATIC Safety - Configuring and Programming*" contient un avertissement identifié comme "S078". Cet avertissement indique ce qui suit : "L'utilisation d'outils pour l'automatisation / le fonctionnement d'interfaces utilisateur (par ex. TIA Portal, serveur Web) qui permettent de contourner la protection d'accès pour la CPU F (par ex. enregistrement ou saisie automatique d'un mot de passe de CPU F de sécurité ou du mot de passe du serveur Web) n'est pas autorisée." (en anglais)

Cet avertissement ne s'applique pas à SIMATIC Automation Tool. SIMATIC Automation Tool et l'API SIMATIC Automation Tool fonctionnent avec des CPU F et peuvent stocker des mots de passe dans la CPU pour les CPU F.

Caractéristiques de sécurité de l'API

TÜV SÜD a certifié SIMATIC Automation Tool V3.1 et l'API associé.

L'API SIMATIC Automation Tool utilise un traitement diversifié et redondant. L'API contribue ainsi à empêcher le code programme de l'application utilisateur d'effectuer des opérations non sûres. L'API offre les caractéristiques suivantes :

- Processus de connexion et de légitimation indépendant pour chaque opération de sécurité
- Contrôles d'identité pour opérations relatives à la sécurité d'appareils de sécurité
- Identification des programmes de sécurité
- Utilisation requise du mot de passe de CPU F de sécurité pour toute opération de sécurité sur une CPU F protégée par mot de passe
- Utilisation des sommes de contrôle CRC 32 bits pour comparer les représentations des appareils de sécurité en mode non connecté et en mode connecté
- Utilisation de codes de Hamming (Page 137) pour indiquer les états TRUE et FALSE
- Comparaison de signatures F après les opérations Mise à jour du programme et Restaurer à partir de la sauvegarde pour vérifier que les opérations se sont déroulées avec succès

7.3.2 Directives de programmation de l'interface utilisateur pour les opérations relatives à la sécurité

ATTENTION

Protéger autant que possible les opérations relatives à la sécurité

Les CPU de sécurité, ainsi que les E/S et les programmes de sécurité, permettent de mettre en place un niveau élevé de sécurité opérationnelle.

Lorsque vous utilisez l'API SIMATIC Automation Tool, assurez-vous que les opérations relatives à la sécurité sont aussi sécurisées que possible. Siemens n'assume aucune responsabilité concernant les applications d'interface utilisateur développées avec l'API SIMATIC Automation Tool. L'entièvre responsabilité incombe au développeur du logiciel.

Le non-respect des pratiques de programmation adéquates peut entraîner la mort ou des dommages corporels lorsque l'utilisateur fait fonctionner votre application d'interface utilisateur.

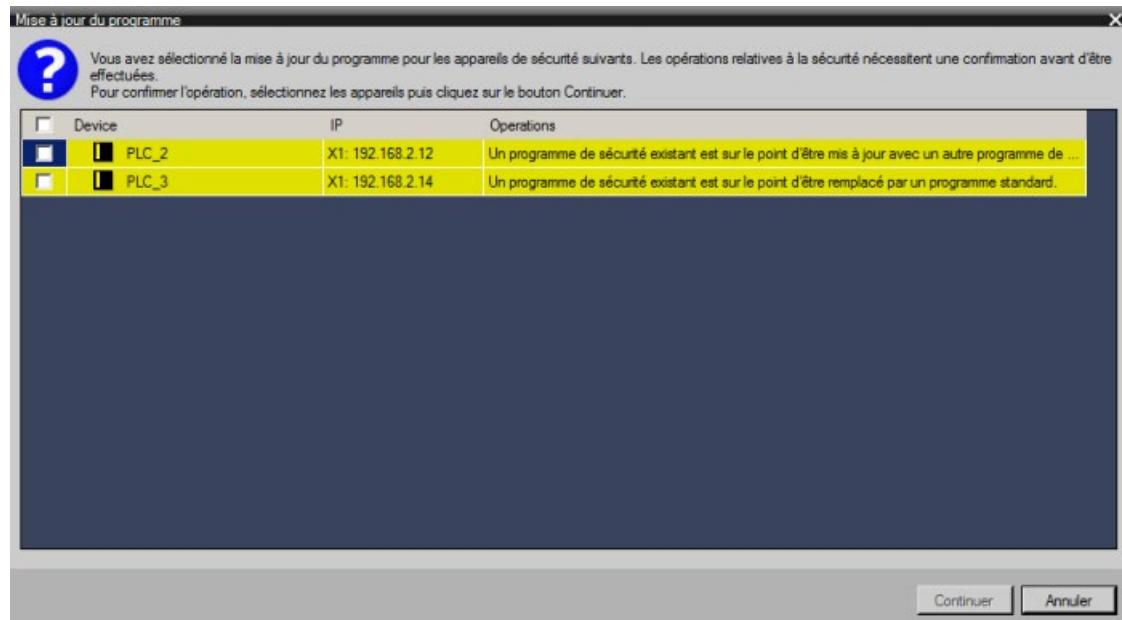
Identification et protection des opérations relatives à la sécurité

SIMATIC Automation Tool V3.1 affiche une boîte de dialogue permettant de confirmer les opérations relatives à la sécurité. L'utilisateur doit sélectionner manuellement chaque appareil de sécurité pour l'opération à effectuer, puis cliquer sur "Continuer" avant que le traitement ne commence.

Lorsque vous développez votre application d'interface utilisateur à l'aide de l'API, déterminez si l'opération à effectuer est l'une des opérations relatives à la sécurité suivantes :

- Mise à jour du programme
- Restaurer l'appareil à partir d'un fichier de sauvegarde
- Restaurer les paramètres d'usine
- Formater la carte mémoire

Pour les opérations relatives à la sécurité, prévoyez une boîte de dialogue de confirmation pour vos utilisateurs. Utilisez la méthode API DetermineConfirmationMessage (Page 224) pour déterminer le type de boîte de dialogue de confirmation à afficher. Prévoir une boîte de dialogue de confirmation supplémentaire protège les utilisateurs contre l'exécution accidentelle d'une opération relative à la sécurité. La boîte de dialogue suivante est un exemple de boîte de dialogue de confirmation de SIMATIC Automation Tool pour une opération de mise à jour de programme :



Pratiques de programmation recommandées

Appliquez les pratiques de programmation suivantes pour vous assurer que vous protégez les opérations relatives à la sécurité et minimisez les risques d'action à risque de la part de l'utilisateur :

- Prévoyez que toutes les opérations relatives à la sécurité s'effectuent par traitement à thread unique.
- Demandez la saisie du mot de passe de CPU F de sécurité pour les opérations relatives à la sécurité. Comparez le mot de passe saisi avec le mot de passe de la CPU. Utilisez des astérisques pour masquer les mots de passe au moment de leur saisie par l'utilisateur.
- Vérifiez les codes de retour de toutes les méthodes. Assurez-vous que la logique de votre programme ne s'appuie que sur des retours de méthodes réussis.
- Prévoyez une gestion appropriée des exceptions dans votre implémentation. L'API génère des exceptions pour les défaillances internes critiques qu'il détecte. Assurez-vous que votre logiciel gère les exceptions de manière appropriée.
- Pour toutes les opérations relatives à la sécurité, déterminez si l'opération a pu ou non être terminée avec succès. Affichez un message à l'attention de l'utilisateur après toute mise à jour réussie du programme. Affichez un message d'erreur en cas d'échec de la mise à jour d'un programme. Suivez les conventions de l'Event Log de SIMATIC Automation Tool pour informer l'utilisateur des résultats des opérations relatives à la sécurité.
- Utilisez des codes de Hamming (Page 137) dans votre application pour implémenter des états booléens.
- Utilisez un code couleurs (Page 131) dans l'application pour identifier des appareils de sécurité, les programmes de sécurité, les mots de passe de CPU F de sécurité et autres données saisies par l'utilisateur.
- Demandez une confirmation pour tous les changements de mode de fonctionnement (RUN/STOP).
- Actualisez l'interface utilisateur après chaque opération afin que l'application affiche les dernières données à jour de l'appareil.

Exigences relatives à la mise à jour des programmes

Pour les mises à jour de programmes sur les CPU F sélectionnées, fournissez à l'utilisateur une boîte de dialogue supplémentaire pour sélectionner à nouveau les appareils de sécurité et confirmer les opérations suivantes :

- Mise à jour d'un programme de sécurité existant avec un autre programme de sécurité
- Mise à jour d'un programme de sécurité existant avec un programme standard
- Mise à jour d'un programme standard existant avec un programme de sécurité
- Mise à jour d'une CPU sans programme avec un programme de sécurité
- Suppression d'un programme de sécurité existant

Suite à la mise à jour d'un programme de sécurité, vérifiez que la signature de sécurité du programme mis à jour dans la CPU correspond à la signature de sécurité du fichier de mise à jour du programme.

Exigences relatives à la restauration à partir d'un fichier de sauvegarde

Avant de restaurer un fichier de sauvegarde, déterminez si le fichier est un programme de sécurité et demandez la confirmation de l'utilisateur selon les mêmes exigences pour les mises à jour de programmes.

Certification

Remarque

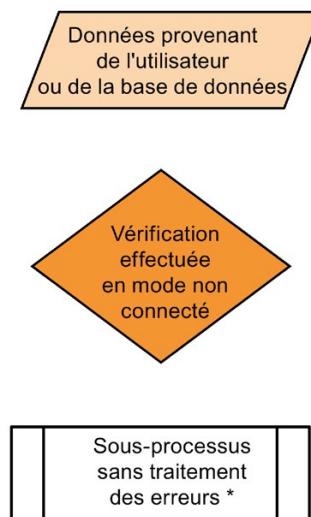
Obtenez une certification pour votre application d'interface utilisateur

Siemens recommande fortement d'utiliser un organisme notifié tel que TÜV SÜD pour certifier la sécurité de votre conception et de votre implémentation.

7.3.3 Utilisez un code couleurs pour les champs de sécurité dans votre interface utilisateur

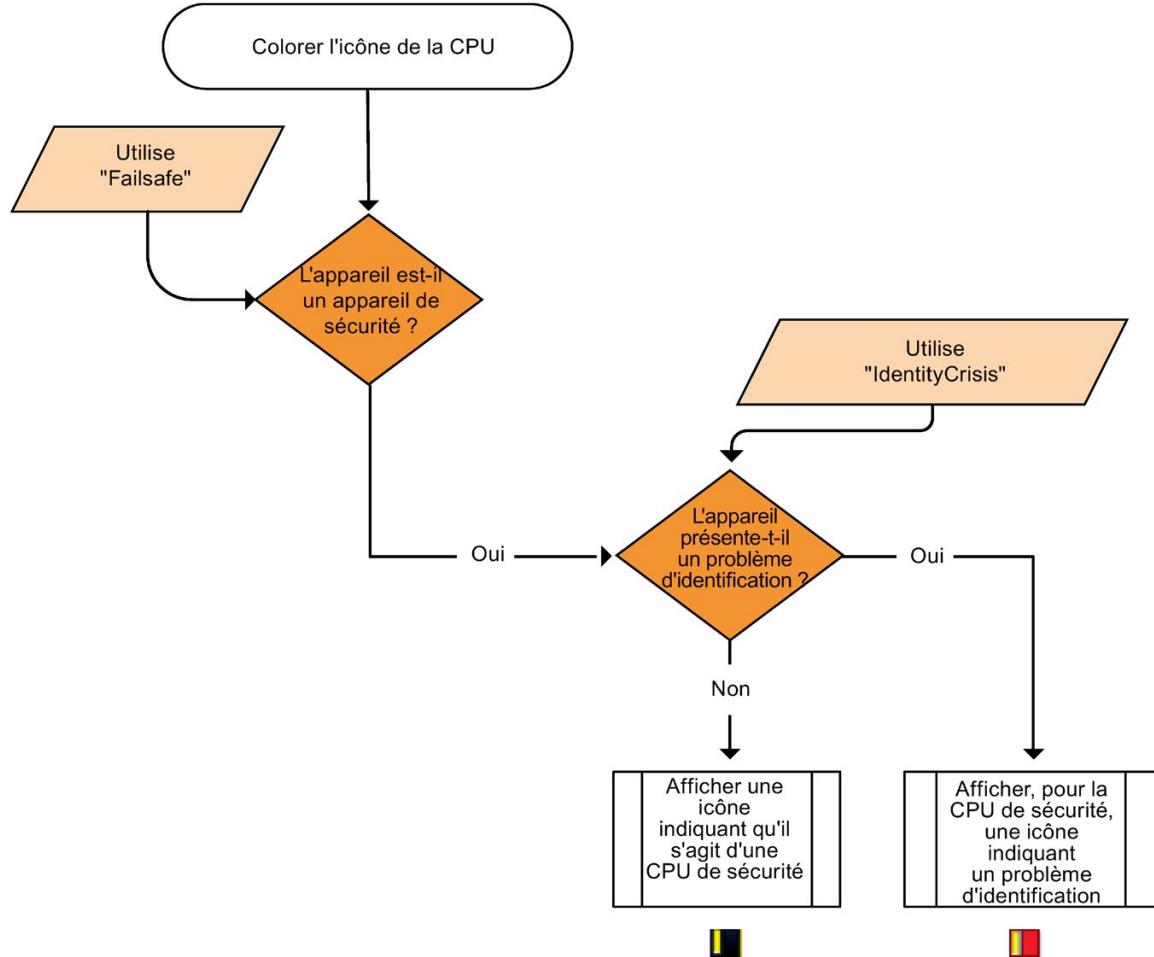
Siemens recommande fortement d'utiliser un code couleurs pour donner à l'utilisateur une indication visuelle de tout ce qui concerne les CPU et les programmes de sécurité. Les schémas décisionnels indiquent la logique du code couleurs utilisé par Siemens pour les différents champs de sécurité dans l'interface utilisateur SIMATIC Automation Tool. Envisagez d'adopter une approche identique ou similaire lorsque vous concevez votre application.

Conventions des schémas décisionnels

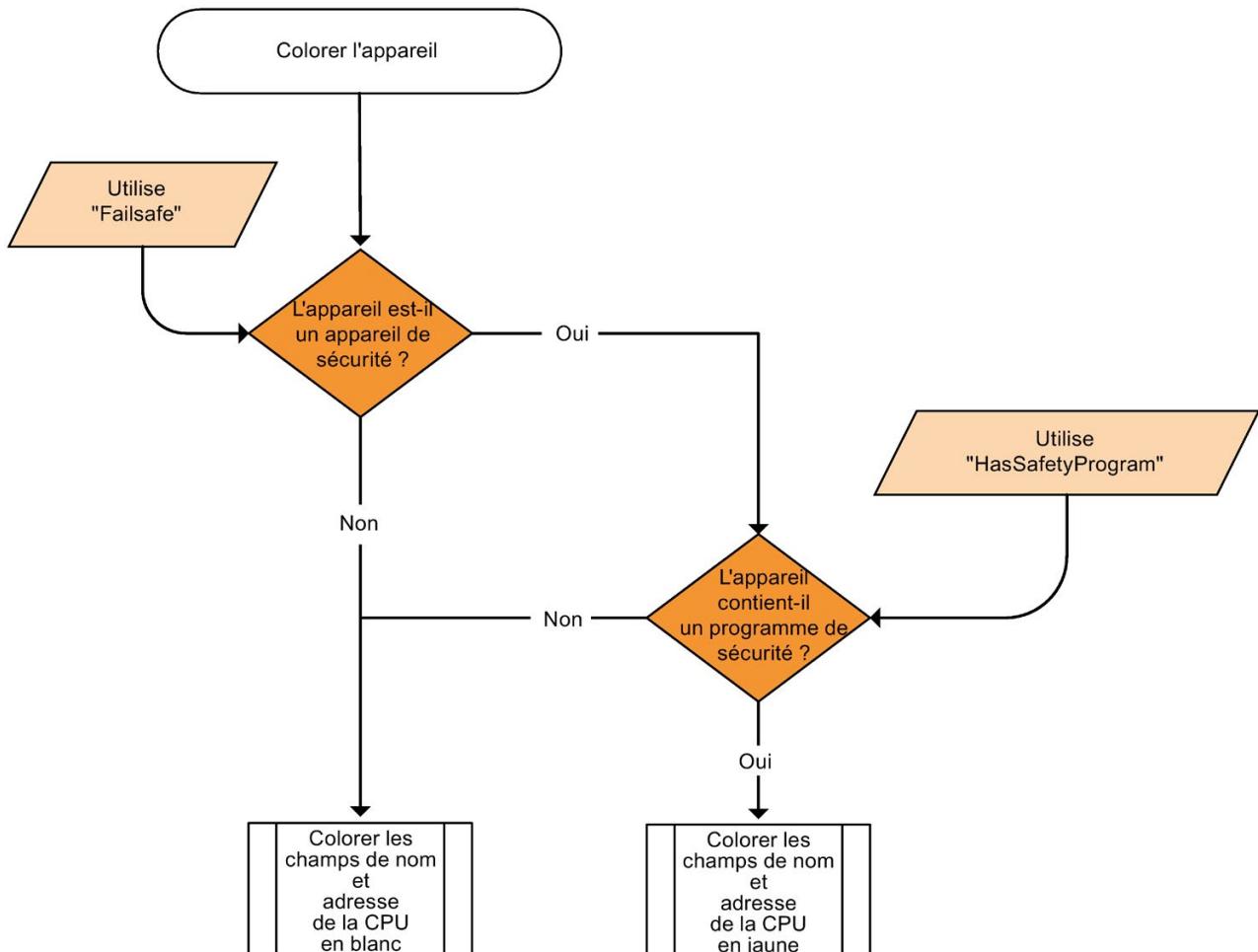


* Sous-processus qui ne peuvent pas échouer, n'incluent pas de traitement des erreurs ou représentent une situation dans laquelle l'application ignore intentionnellement des erreurs.

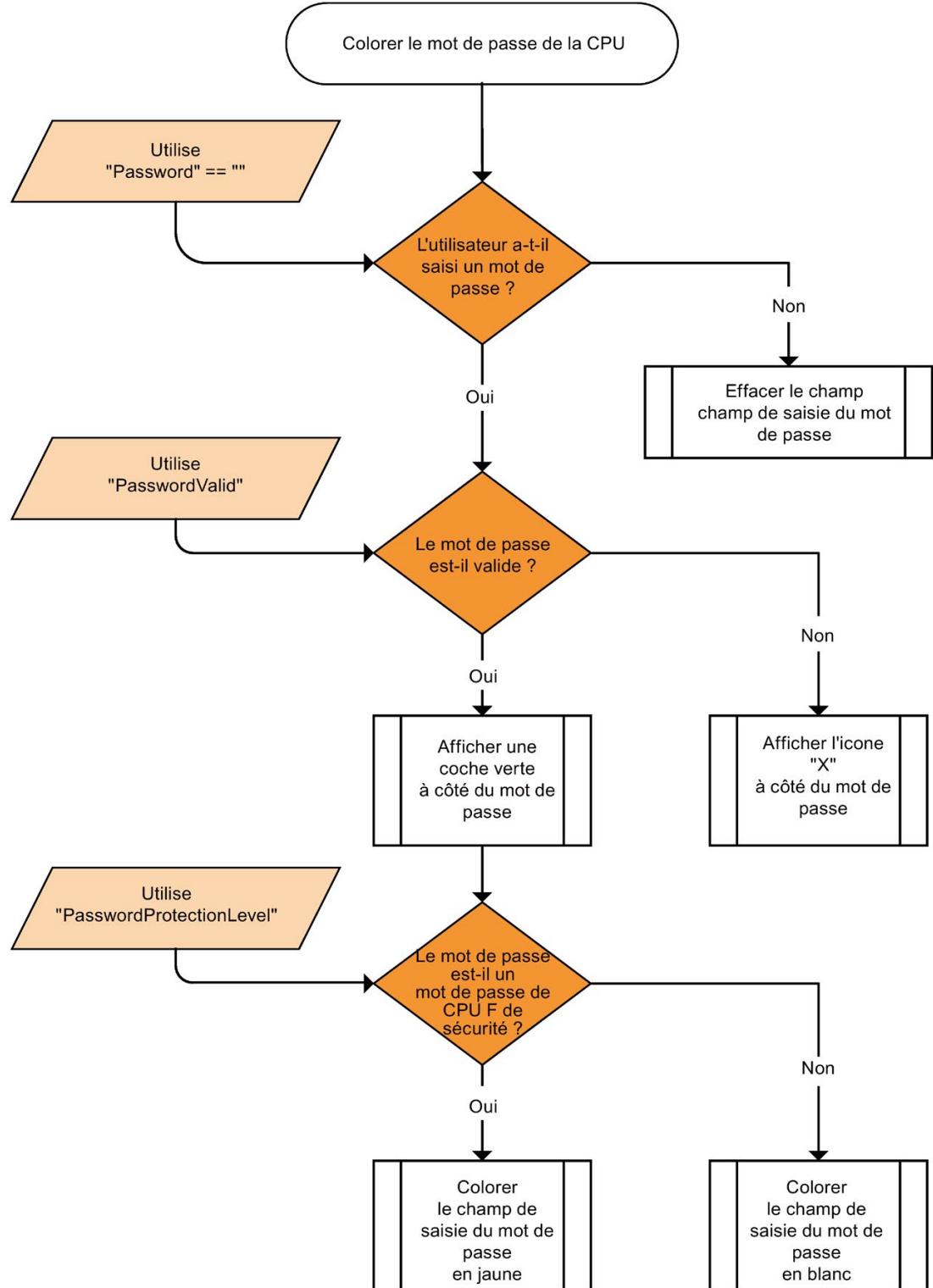
7.3.3.1 Appliquer une couleur à une icône d'appareil CPU



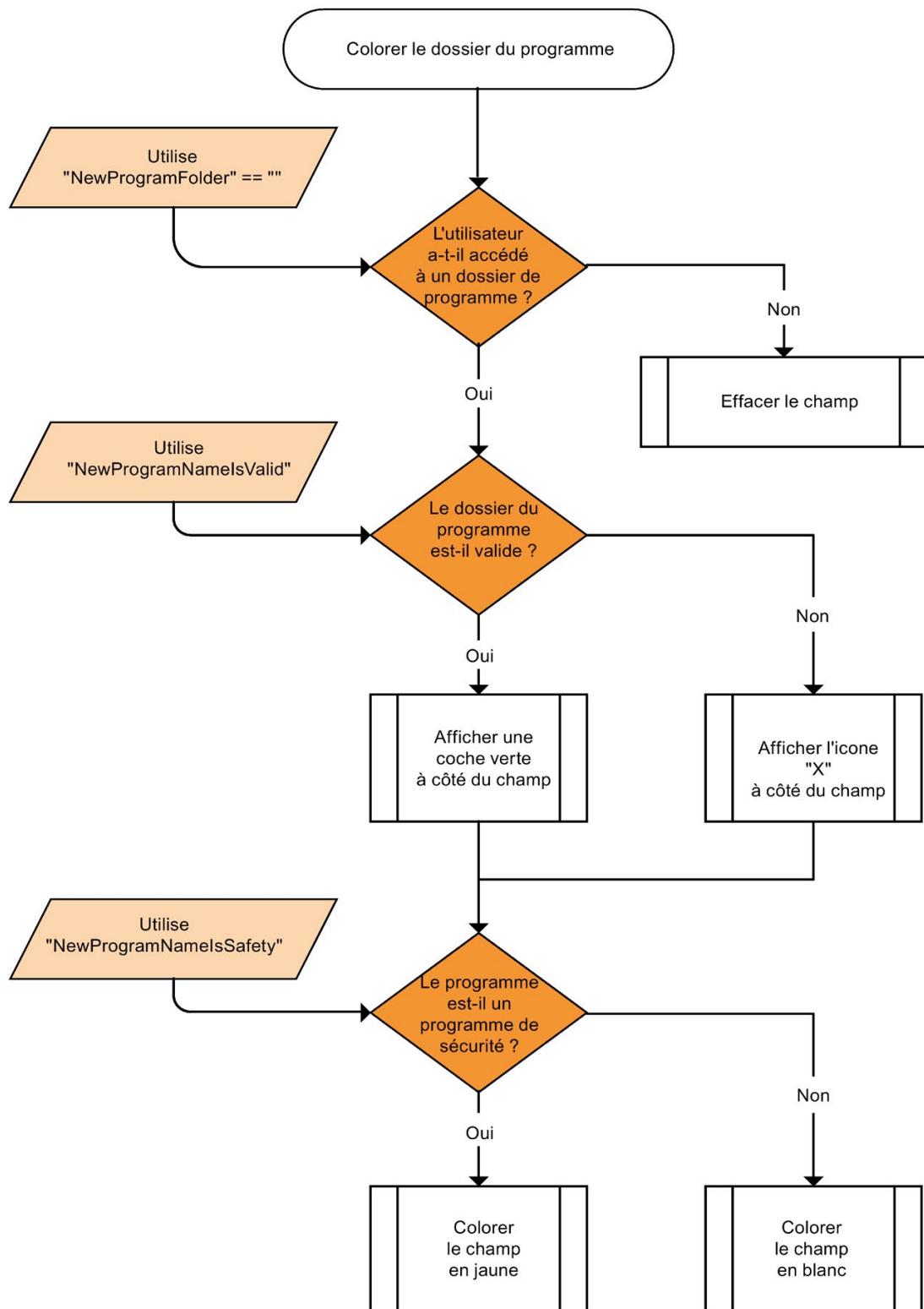
7.3.3.2 Appliquer une couleur aux données d'un appareil



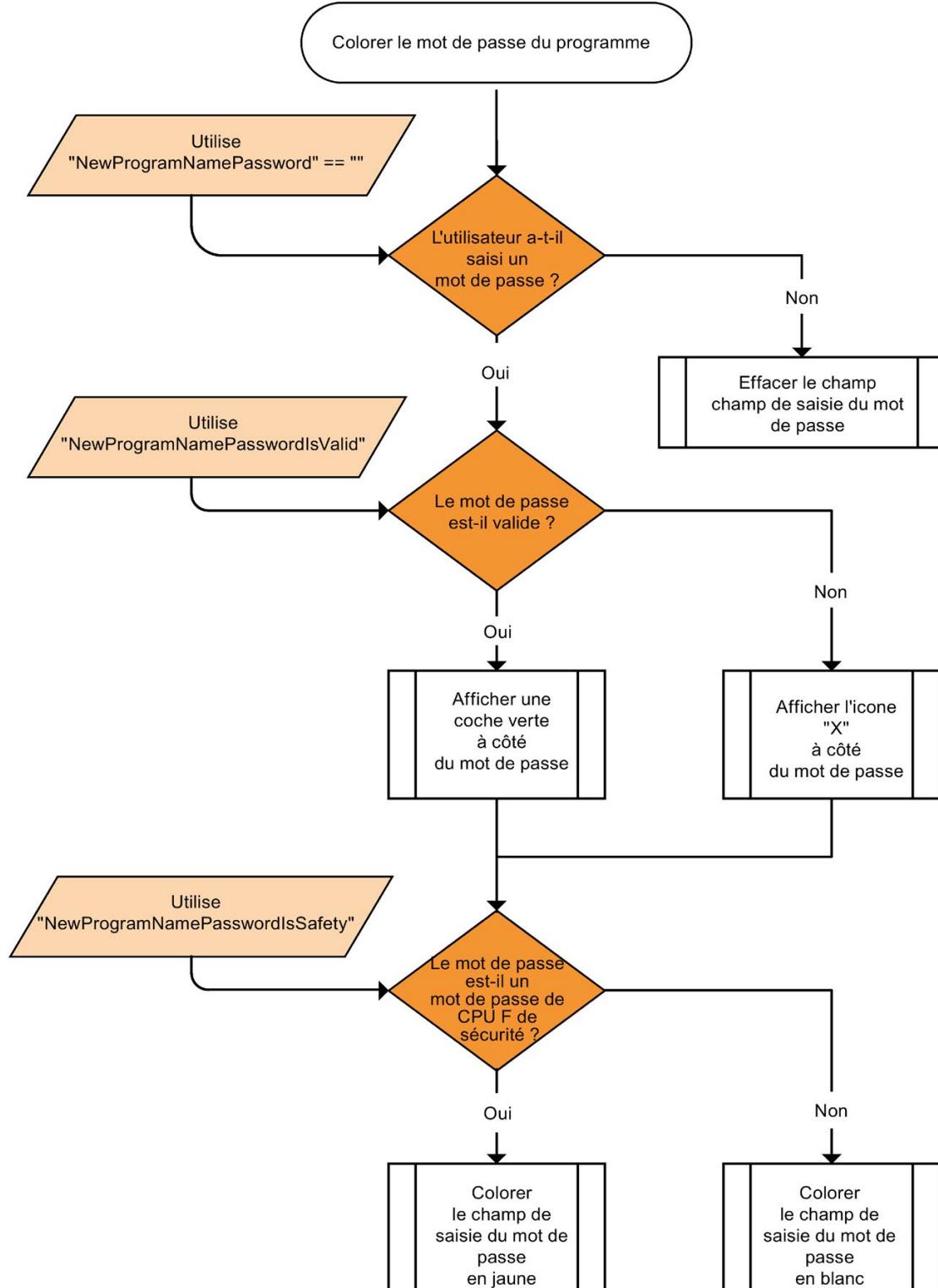
7.3.3.3 Appliquer une couleur au mot de passe de la CPU



7.3.3.4 Appliquer une couleur à un dossier de programme



7.3.3.5 Appliquer une couleur à un mot de passe de programme



7.3.4

Codes de Hamming

Les codes de Hamming sont des codes binaires. Ils peuvent détecter des erreurs de bits accessoires. SIMATIC Automation Tool utilise des codes de Hamming 32 bits avec une distance de Hamming de 8. L'API utilise des codes de Hamming pour représenter toutes les valeurs booléennes liées aux opérations relatives à la sécurité. Vous pouvez programmer votre application d'interface utilisateur de manière à utiliser les états de valeurs booléennes fournis pour les opérations relatives à la sécurité. Étant donné que l'API implémente ces états avec des codes de Hamming, vous pouvez avoir une grande confiance dans l'intégrité des données des états booléens relatifs à la sécurité.

7.4

Présentation de l'architecture

Réseaux

La classe `.NET Network` représente le réseau industriel dans son ensemble. Cette classe sert à exécuter des fonctions à l'aide d'une carte d'interface réseau (NIC) installée sur un PG/PC. La classe `Réseau` sert à rechercher les cartes d'interface disponibles et à choisir la carte à utiliser.

- Constructeur réseau (Page 151)
- Méthode `QueryNetworkInterfaceCards` (Page 151)
- Méthode `SetCurrentNetworkInterface` (Page 152)
- Propriété `CurrentNetworkInterface` (Page 153)
- Méthode `ScanNetworkDevices` (Page 153)
- Méthode `SetCommunicationsTimeout` (Page 154)
- Méthode `GetCommunicationsTimeout` (Page 155)
- Méthode `CheckValidLicense` (Page 156)
- Méthode `GetEmptyCollection` (Page 156)

Appareils

Les appareils du réseau sont représentés par des interfaces. Chaque classe d'interface fournit des propriétés et des méthodes appropriées pour l'appareil du réseau représenté. Les appareils matériels du réseau sont représentés au mieux par les interfaces suivantes :

`IProfinetDevice` – tout appareil accessible directement sur le réseau industriel peut être représenté par cette interface.

`ICPU` – représente les CPU S7 connectées directement au réseau. Une fonctionnalité spécifique est prise en charge pour les CPU.

`IHMI` – représente les IHM SIMATIC connectées directement au réseau. Une fonctionnalité spécifique est prise en charge pour les IHM.

`IBaseDevice` – cette interface sert à représenter les appareils qui ne sont pas connectés directement au réseau Ethernet, mais sont accessibles via un autre appareil. Par exemple, une station esclave PROFIBUS raccordée à une CPU du réseau est représentée en tant que `IBaseDevice`.

IModule – cette interface représente des modules I/O qui sont enfichés sur une CPU, un périphérique PROFINET ou une station PROFIBUS.

IHardware – il s'agit de la classe de base pour toutes les autres interfaces. Cette interface fournit un accès aux propriétés qui sont communes à tous les matériaux directement détectés sur le réseau.

Les interfaces sont rassemblées dans des collections qui constituent des groupes d'appareils. Les collections sont fournies pour prendre en charge les itérations, les filtres et la recherche.

IProfinetDeviceCollection – une collection de tous les appareils directement accessibles sur le réseau.

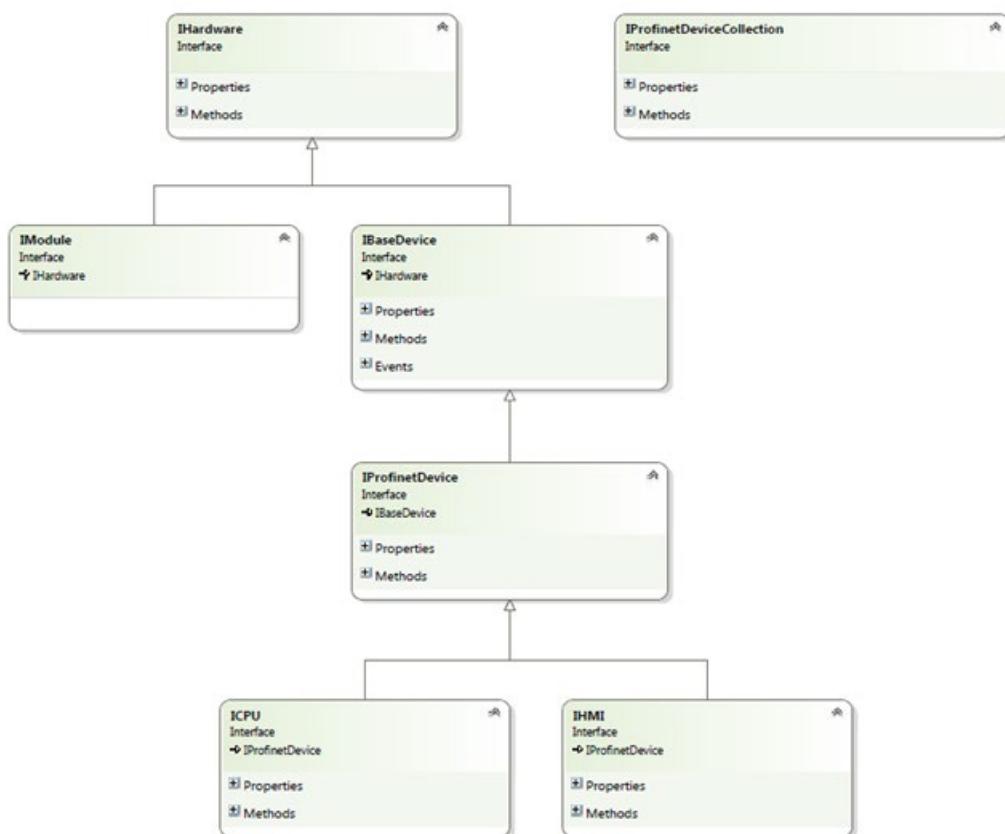
IModuleCollection – une collection pouvant représenter tous les modules enfichés sur une CPU ou un module d'interface.

IHardwareCollection – cette collection représente une CPU et tous ses modules.

Classes d'appareils, interfaces et méthodes :

- Classe **IProfinetDeviceCollection** (Page 157)
- Interface **IProfinetDevice** (Page 169)
- Interface **ICPU** (Page 187)
- Interface **IHMI** (Page 230)

Le diagramme de classes suivant montre la relation de filiation entre ces classes d'interface :



Remarque

Voir l'exemple (Page 251) de réseau industriel et les classes d'API SIMATIC Automation Tool utilisées pour représenter chaque composant du réseau.

Voir aussi

- Méthode FirmwareUpdate (Page 174)
- Méthode Backup (interface ICPU) (Page 204)
- Méthode SetOperatingState (Page 215)
- Propriété de modules et classe IModuleCollection (Page 185)
- Méthode Restore (interface ICPU) (Page 203)
- Méthode SetIP (Page 178)
- Méthode Identify (Page 176)
- Méthode RefreshStatus (Page 173)
- Méthode Reset (Page 177)
- Méthode SetProfinetName (Page 179)
- Méthode GetCurrentDateTime (Page 210)
- Méthode GetDiagnosticsBuffer (Page 211)
- Méthode MemoryReset (Page 212)
- Méthode ProgramUpdate (Page 197)
- Méthode ResetToFactoryDefaults (Page 213)
- Méthode SetCurrentDateTime (Page 216)
- Méthode UploadServiceData (Page 221)
- Interface IModule (Page 186)
- Méthode DeleteDataLog (Page 206)
- Méthode DeleteRecipe (Page 208)
- Méthode UploadDataLog (Page 217)
- Méthode UploadRecipe (Page 219)
- Méthode Backup (interface IHMI) (Page 233)
- Méthode ProgramUpdate (interface IHMI) (Page 234)
- Méthode Restore (interface IHMI) (Page 236)
- Méthode DownloadRecipe (Page 205)
- Propriétés RemoteInterfaces (Page 226)
- Événements IProfinetDevice (Page 182)

7.5 Référencement de l'API dans une application d'interface utilisateur

Siemens fournit l'API avec plusieurs DLL, exécutables et fichiers source :

- AutomationToolAPI.dll
- DeviceManagerClient.dll (IHM)
- hmitr.dm.client.proxy.dll (IHM)
- hmitr.dm.client.stub.exe (IHM)
- hmitr.ipc.dll (IHM)
- Dossier et sous-dossiers AsModels (modèles d'objets en mode non connecté)

L'API a été créée avec Microsoft Visual Studio 2015 SP2 Update 3 à l'aide de .NET Framework 4.6.2. Vous pouvez utiliser cette API avec les applications que vous créez avec cette version ou des versions ultérieures de Visual Studio. Tous les exemples de code et les captures d'écran du présent document ont été créés en langage de programmation C# sous Visual Studio 2015 SP2 Update 3.

Pour inclure l'API dans votre application, vous devez ajouter `AutomationTool.dll` comme "référence" dans la solution Visual Studio.

Dans tous les fichiers source qui réfèrent les classes de l'API, vous devez ajouter l'instruction suivante qui référence l'espace-nom de l'API :

```
using Siemens.Automation.AutomationTool.API;
```

Pour compiler les échantillons de code dans ce document, l'instruction `using` correcte doit être présente dans le même fichier source (*.cs) comme l'exemple de code. Pour des raisons de simplicité, les exemples de code du présent document ne contiennent pas l'instruction `using`.

Pour utiliser l'API en runtime, les versions correctes des communications S7 doivent être installées sur le PG/PC. Pour être sûr d'utiliser les fichiers corrects, il vaut mieux installer SIMATIC Automation Tool sur cette machine. Après installation, vous pouvez placer le fichier dll de l'API (AutomationToolAPI.dll), avec les fichiers d'IHM énumérés ci-dessus dans un dossier quelconque sur le PG/PC et les utiliser avec succès.

7.6 Classes de base

7.6.1 Classe EncryptedString

Avant de décrire les opérations disponibles via l'API, il est important de comprendre certaines classes basiques qui sont utilisées dans la majorité des exemples de code.

La classe `EncryptedString`

De nombreuses opérations API requièrent une connexion légitimée à une CPU S7 protégée. Pour ces opérations, un mot de passe est requis parmi les paramètres de la méthode. La CPU S7 accepte le mot de passe dans un format chiffré. Pour cela, l'API fournit la classe `EncryptedString`.

Constructeur	Description
<code>EncryptedString()</code>	Une chaîne de caractères chiffrée vide
<code>EncryptedString(string strText)</code>	Une chaîne chiffrée

Nom de la propriété	Type de retour	Description
<code>IsEmpty</code>	<code>bool</code>	Vrai, s'il n'y a pas de mot de passe.
<code>IsEncrypted</code>	<code>byte</code>	Vrai, s'il existe un mot de passe chiffré

Nom de la méthode	Type de retour	Description
<code>ToString()</code>	<code>string</code>	Représentation du mot de passe chiffré sous forme de chaîne hexadécimale
<code>Clear()</code>	<code>void</code>	Supprime le mot de passe chiffré
<code>GetHash()</code>	<code>byte[]</code>	Représentation sous forme de tableau de hachage chiffré par mot de passe
<code>WriteToStream(Stream stream)</code>	<code>void</code>	Sérialiser un mot de passe à partir d'un flux
<code>ReadFromStream(Stream stream)</code>	<code>void</code>	Désérialiser un mot de passe à partir d'un flux

Cette classe fournit un moyen de chiffrer un mot de passe en texte clair pour que vous puissiez ensuite légitimer une connexion CPU. De nombreux exemples de code montrent un usage type de cette classe.

Si vous voulez chiffrer un mot de passe à utiliser plusieurs fois dans le code, vous pouvez instancier la `EncryptedString` et la transmettre ensuite comme paramètre pour des appels multiples, comme ci-dessous :

```
EncryptedString pwd = new EncryptedString("password");

devAsCpu.Selected = true;
devAsCpu.SetPassword(pwd);
```

Remarque

Si une CPU n'est pas protégée par mot de passe, envoyez simplement une chaîne vide au constructeur `EncryptedString`. Par exemple, le code suivant est efficace pour une CPU configurée sans protection :

```
devAsCpu.SetPassword(new EncryptedString("Password"));
devAsCpu.Selected = true;
Result retVal = devAsCpu.RefreshStatus();
```

L'objet `EncryptedString` ne stocke pas le mot de passe défini par l'utilisateur en texte clair. Toutefois, si votre application code les mots de passe en tant que chaînes littérales, vous créez un risque de sécurité.

Par exemple : `new EncryptedString("myPassword")`

Le texte en clair "myPassword" est compilé dans l'application utilisateur et peut être visible pour quiconque utilise des outils de réflexion .NET.

7.6.2 Classe Result

La classe `Result` encapsule la logique qui détermine si une action API donnée a réussi. La plupart des actions implique un certain degré de communications réseau. Nombre d'entre elles impliquent l'ouverture d'une connexion à un appareil du réseau. Le succès n'est jamais garanti pour ces actions. L'objet `Result` renvoyé par une action API doit toujours être inspecté pour savoir si l'action a été terminée avec succès ou a échoué.

Constructeur	Description
<code>Result()</code>	Crée un résultat réussi sans avertissement
<code>Result(ErrorCode nCode)</code>	Crée une erreur spécifique sans avertissement

Nom de la propriété	Type de retour	Description
Avertissements	<code>ErrorCode[] {get;}</code>	Renvoie tous les avertissements dans un tableau de codes d'erreurs
Erreur	<code>ErrorCode {get;}</code>	Renvoie un code d'erreur
HasWarnings	<code>Bool {get;}</code>	Vrai, lorsqu'il existe des avertissements
Échec	<code>Bool {get;}</code>	Vrai, lorsque le résultat a échoué
Terminé avec succès	<code>Bool {get;}</code>	Vrai, lorsque le résultat est un succès Succès = contraire d'échec

Souvent, il suffit de savoir si une action donnée a réussi. Dans ce cas, une simple vérification de la propriété `Succeeded` suffit :

```
dev.Selected = true;
Result retVal = dev.RefreshStatus();
if (retVal.Succeeded)
{
    //-----
    // Poursuivre les opérations....
    //-----
}
```

Dans les autres cas, il peut être utile d'obtenir plus d'informations sur l'échec. Pour inspecter une erreur donnée, utilisez la propriété de code comme indiqué ci-dessous :

```
dev.Selected = true;
Result retVal = dev.RefreshStatus();
if (retVal.Succeeded)
{
    //-----
    // Continuer les opérations....
    //-----
}
else
{
    //-----
    // Que s'est-il passé ?
    //-----
    switch (retVal.Error)
    {
        case ErrorCode.AccessDenied:
            break;
        case ErrorCode.TooManySessions:
            break;
    }
}
```

Voir le sujet `ErrorCode` (Page 243) pour une liste de valeurs.

La classe `Result` fournit également une description de l'erreur dans la langue spécifiée. La méthode `GetDescription` utilise une valeur `Language` comme paramètre.

Par exemple, le code suivant renvoie la description de l'erreur en allemand :

```
String strError = result.GetErrorDescription(Language.German);
```

Voir la rubrique `énumération de la Langue` (Page 246) pour une liste de valeurs.

SIMATIC Automation Tool V3.1 dispose d'une nouvelle fonction d'avertissement dans les cas où vous devez être informé(e) des problèmes survenus. Par exemple, l'actualisation effectuée sur l'appareil à la fin de la mise à jour d'un programme peut créer des avertissements qui ne sont pas directement liés à la fonction d'appel principale. Vous pouvez accéder à ces avertissements dans la classe `Result`, comme suit.

```
if (RetVal.HasWarnings)
{
    foreach (ErrorCode warning in RetVal.Warnings)
    {
        //-----
        // Continuer les opérations....
        //-----
    }
}
```

7.6.3 Classe de diagnostic

Un élément de diagnostic contient des informations de diagnostic pour un seul événement. Il est possible de lire le tampon de diagnostic d'une CPU. Voir le chapitre Interface ICPU (Page 187) pour plus de détails.

Constructeur	Description
<code>DiagnosticsItem()</code>	Crée un élément de diagnostic par défaut

Nom de la propriété	Type de retour	Description
<code>TimeStamp</code>	<code>DateTime {get;}</code>	Horodatage de l'événement de diagnostic
<code>State</code>	<code>Byte {get;}</code>	0 = événement sortant ; 1 = événement entrant
<code>Description1</code>	<code>String {get;}</code>	Description sommaire
<code>Description2</code>	<code>String {get;}</code>	Description détaillée

7.6.4 Classe DataChangedEventArgs

Un événement lié à une modification de données contient des informations sur les données qui ont changé au sein de l'API. Voir le chapitre Interface IProfinet (Page 169) pour plus de détails.

Constructeur	Description
DataChangedEventArgs (DataChangedType type)	Crée un événement de type spécifique
DataChangedType type	Type de données ayant changé

Nom de la propriété	Type de retour	Description
Type	DataChangedType	Type d'événement

Utilisé avec le gestionnaire d'événements suivant :

```
public delegate void DataChangedEventHandler(object sender, DataChangedEventArgs e);
```

7.6.5 Classe ProgressChangedEventArgs

Un événement lié à une modification de progression contient des informations sur les données qui ont changé dans l'API. Voir le chapitre Interface IProfinet (Page 169) pour plus de détails.

Constructeur	Description
ProgressChangedEventArgs (ProgressAction action, int index, int count, uint hardwareID)	Utilisé pour créer et assigner par défaut une classe ProgressChangedEventArgs
ProgressAction action	Type de progression observé.
int index	Index de l'élément actuellement en cours de traitement
int count	Total des éléments à traiter
uint hardwareID	Identité de l'élément en cours de traitement

Nom de la propriété	Type de retour	Description
ID	uint {get;}	Identité de l'élément
Cancel	bool {get;}	Choisir "Vrai" pour terminer l'opération en cours
Count	int {get;}	Valeur maximale
Index	int {get;}	Valeur actuelle
Action	ProgressAction {get;}	Type d'action de cet événement

Utilisé avec le gestionnaire d'événements suivant :

```
public delegate void ProgressChangedEventHandler(object sender,
ProgressChangedEventArgs e);
```

7.7 Interfaces de support communes

7.7.1 Interface IRemoteFile

IRemoteFile est une interface utilisée pour représenter les fichiers utilisés dans les Data Logs et les recettes.

Nom de la propriété	Type de retour	Description
Sélectionné(e)	bool {get; set;}	État sélectionné
FileSize	ulong {get;}	Taille du fichier sur la CPU
Nom	string {get; set;}	Nom et extension du fichier sur l'appareil.

7.7.2 Interface IRemoteFolder

IRemoteFolder est utilisée pour représenter les dossiers utilisés dans les datalogs et les recettes.

Nom de la méthode	Type de retour	Description
SetRemoteFile(string strFile)	Result	
string strFile		Nom et chemin d'accès complets du fichier distant

Nom de la propriété	Type de retour	Description
NewFileNameErrorCode	Result {get;}	Code d'erreur enregistré après appel de la méthode SetRemoteFile
NewFileNameIsValid	bool {get;}	Vrai, si le nom du fichier est valide.
FileUpdateAllowed	bool {get;}	Vrai, si le dossier peut ajouter ou remplacer un fichier dans la liste.
SelectedCount	int {get;}	Nombre de fichiers sélectionnés
Files	List<IRemoteFile> {get;}	Tableau des fichiers présents dans ce dossier
FolderType	RemoteFolderType {get;}	Type de dossier (datalog ou recette)
Exists	bool {get;}	Vrai, si ce dossier existe sur l'appareil.
Selected	bool {get; set;}	Vrai, si le dossier est sélectionné.
NewFile	string {get;}	Chemin d'accès complet du fichier à ajouter ou à remplacer
NewFileName	string {get;}	Nom du fichier à ajouter ou à remplacer
Name	string {get;}	Nom du dossier

7.7.3 Interface IRemoteInterface

`IRemoteInterface` est une interface utilisée pour représenter la périphérie décentralisée dans un réseau.

Nom de la propriété	Type de retour	Description
Devices	<code>List<IBaseDevice> {get;}</code>	Tableau des interfaces distantes, utilisé pour représenter la périphérie décentralisée.
InterfaceType	<code>RemoteInterfaceType {get;}</code>	Type d'interface distante comme PROFINET ou PROFIBUS
Name	<code>string {get;}</code>	Nom et extension du fichier sur l'appareil.

7.7.4 Interface IHardware

`IHardware` est une interface utilisée pour représenter l'interface matérielle commune de base pour les appareils et les modules. `IModule` élargit l'interface `IHardware`.

Nom de la méthode	Type de retour	Description
<code>SetFirmwareFile(string strFile)</code>	<code>Result</code>	Définit le fichier de firmware à mettre à jour sur cet appareil ou ce module.

Nom de la propriété	Type de retour	Description
Comment	<code>string {get; set;}</code>	Commentaire pour chaque appareil et module
Selected	<code>bool {get; set;}</code>	Utilisé pour le stockage externe de l'état sélectionné
NewFirmwareNameErrorCode	<code>Result {get;}</code>	Dernière erreur reçue par la méthode <code>SetFirmwareFile</code>
FirmwareUpdateAllowed	<code>bool {get;}</code>	Vrai, si cet appareil ou ce module prend en charge la mise à jour du firmware.
NewFirmwareNameIsValid	<code>bool {get;}</code>	Vrai, si le fichier de firmware est valide pour cet appareil ou ce module.
Failsafe	<code>bool {get;}</code>	Vrai, s'il s'agit d'un appareil ou module de sécurité.
Supported	<code>bool {get;}</code>	Vrai, si cet appareil ou module est pris en charge.
NewFirmwareFile	<code>string {get;}</code>	Chemin d'accès complet du fichier de firmware
NewFirmwareVersion	<code>string {get;}</code>	Version du fichier de firmware qui s'affiche dans la liste déroulante
Configured	<code>bool {get;}</code>	Vrai, si cet appareil ou ce module est configuré.
HardwareNumber	<code>short {get;}</code>	Numéro de révision matérielle de l'appareil ou du module.
SlotName	<code>string {get;}</code>	Nom de l'emplacement de l'appareil ou du module.

Nom de la propriété	Type de retour	Description
SubSlot	uint {get;}	Numéro de sous-emplacement de l'appareil ou du module.
Slot	uint {get;}	Numéro d'emplacement de l'appareil ou du module.
StationNumber	uint {get;}	Numéro de station de l'appareil ou du module.
SerialNumber	string {get;}	Numéro de série de l'appareil ou du module.
FirmwareVersion	string {get;}	Version de firmware de l'appareil ou du module.
ArticleNumber	string {get;}	Numéro d'article de l'appareil ou du module.
Description	string {get;}	Description du numéro d'article de l'appareil ou du module.
Name	string {get;}	Nom de l'appareil ou du module.
ID	uint {get;}	Identifiant de l'appareil ou du module.

7.7.5 Interface IModule

`IModule` est une interface utilisée pour représenter un module. `IModule` élargit l'interface `IHardware`.

Nom de la méthode	Type de retour	Description
Aucune		

Nom de la propriété	Type de retour	Description
Aucune		

7.7.6 Interface **IBaseDevice**

IBaseDevice est une interface utilisée pour élargir l'interface **IHardware**, qui représente le type d'appareil le plus basique.

Nom de la méthode	Type de retour	Description
GetHardwareFromID(uint hardwareID)	IHardware	Trouve un appareil ou un module à l'aide d'un identifiant

Nom de la propriété	Type de retour	Description
HardwareInDisplayOrder	IHardwareCollection	Tableau d'éléments matériels présentés dans l'ordre d'affichage
HardwareInFirmwareOrder	IHardwareCollection	Tableau d'éléments matériels présentés par ordre de mise à jour du firmware
Modules	IModuleCollection	Tableau de modules
ThreadNumber	int	Nombre actuel de threads de l'opération
Family	DeviceFamily	Énumération des types de familles

Événements	Type de retour	Description
ProgressChanged	ProgressChangedEventArgs	Appelé pour suivre la progression
DataChanged	DataChangedEventArgs	Appelé lorsque les données changent dans l'API

7.7.7 Interface **IHardwareCollection**

IHardwareCollection est une interface utilisée pour représenter un tableau d'interfaces **IHardware**. Cette interface élargit une classe de liste .NET.

Nom de la propriété	Type	Description
Aucune		

7.7.8 Interface **IModuleCollection**

IModuleCollection est une interface utilisée pour représenter un tableau d'interface **IModule**. Cette interface élargit une classe de liste .NET. Cette interface est utilisée pour représenter les modules locaux et distants dans le châssis matériel.

Nom de la propriété	Type	Description
Aucune		

7.8 Classe Network

7.8.1 Constructeur réseau

La classe `.NET Network` sert à exécuter des fonctions à l'aide d'une carte d'interface réseau (NIC) installée sur le PG/PC. La classe `Network` sert à rechercher des cartes d'interface disponibles et à choisir la carte qui communique avec le réseau industriel.

Pour interagir avec le réseau industriel, votre programme déclare une variable de type `Network` comme suit :

```
Network myNetwork = new Network();
```

Vous pouvez utiliser cet objet pour rechercher les interfaces réseau disponibles et sélectionner celle que vous voulez utiliser.

7.8.2 Méthode QueryNetworkInterfaceCards

Type de retour	Nom de la méthode
<code>Result</code>	<code>QueryNetworkInterfaceCards</code>

Parameters			
Nom	Type de données	Type de paramètre	Description
<code>aInterfaces</code>	<code>List<string></code>	<code>Out</code>	Une collection de toutes les cartes d'interface réseau sur le PG/PC, classées par nom.

Pour identifier les cartes d'interface réseau disponibles, utilisez la méthode `QueryNetworkInterfaceCards` comme indiqué dans l'exemple suivant :

```
Network myNetwork = new Network();

List<String> interfaces = new List<String>();
Result retVal = myNetwork.QueryNetworkInterfaceCards(out interfaces);
if (retVal.Succeeded)
{
    //-----
    // La méthode renvoie une liste de chaînes.
    // Chaque chaîne de la liste représente une carte d'interface réseau disponible.
    // La liste peut être itérée à l'aide d'une notation array.
    //-----
    for (Int32 index = 0; index < interfaces.Count; index++)
    {
        String strInterfaceName = interfaces[index];
    }
}
```

Comme le montre l'exemple, la méthode sort une liste de chaînes. Chaque élément de la liste représente une carte d'interface réseau disponible, identifiée par son nom.

La méthode `QueryNetworkInterfaceCards` renvoie un objet `Result`. Celui-ci représente l'état de l'opération. A un niveau supérieur, cet objet indique si l'opération a réussi (la propriété `Succeeded` est vraie) ou échoué (la propriété `Succeeded` est fausse). De nombreuses causes peuvent être à l'origine de l'échec d'une opération.

Pour une description complète de la classe `Result`, voir aussi Classe `Result` (Page 143)

7.8.3 Méthode SetCurrentNetworkInterface

Type de retour	Nom de la méthode
<code>Result</code>	<code>SetCurrentNetworkInterface</code>

Paramètres			
Nom	Type de données	Type de paramètre	Description
<code>strInterface</code>	<code>string</code>	<code>In</code>	Le nom de l'interface réseau à utiliser. Normalement, il s'agit de l'un des noms renvoyés par la Méthode <code>QueryNetworkInterfaceCards</code> .

Pour utiliser une des cartes d'interface réseau pour accéder au réseau industriel, il est nécessaire de la définir ("set"). Le code suivant montre comme définir une des interfaces réseau identifiée pour les opérations API. Dans cet exemple, le code choisit d'utiliser la première carte d'interface réseau identifiée dans l'exemple présent.

```
Network myNetwork = new Network();

List<String> interfaces = new List<String>();
Result retVal = myNetwork.QueryNetworkInterfaceCards(out interfaces);
if (retVal.Succeeded)
{
    retVal = myNetwork.SetCurrentNetworkInterface(interfaces[0]);
    if (retVal.Succeeded)
    {
        //-----
        // L'action a été terminée avec succès. Poursuivre les opérations.
        //-----
    }
}
```

7.8.4 Propriété CurrentNetworkInterface

Cette propriété est fournie pour rechercher les interfaces réseau actuellement sélectionnée. Cette propriété est en lecture seule. L'exemple suivant montre comment utiliser cette propriété.

```
Network myNetwork = new Network();
string currentInterface = myNetwork.CurrentNetworkInterface;
```

Remarque

Cette propriété renvoie une chaîne vide si aucune interface réseau n'a été sélectionnée par un appel précédent de la méthode `SetCurrentNetworkInterface`.

7.8.5 Méthode ScanNetworkDevices

Type de retour	Nom de la méthode
Result	ScanNetworkDevices

Paramètres			
Nom	Type de données	Type de paramètre	Description
strFile	IProfinetDeviceCollection	Out	Une collection contenant un élément <code>IProfinetDevice</code> pour chaque appareil accessible sur le réseau industriel.

Une fois que l'interface réseau est sélectionnée, il est possible de rechercher les appareils sur le réseau industriel. La méthode `ScanNetworkDevices` renvoie une collection d'éléments qui représentent chacun un appareil raccordé directement au réseau Industrial Ethernet. Ces appareils peuvent être des CPU, des modules locaux, des stations E/S décentralisées, des IHM et d'autres appareils.

L'exemple suivant crée une collection de tous les appareils accessibles sur l'interface réseau sélectionnée.

```
Network myNetwork = new Network();

IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (retVal.Succeeded)
{
    //-----
    // L'action a été terminée avec succès. Poursuivre les opérations.
    //-----
}
```

Cette méthode génère une `IProfinetDeviceCollection`. Cette classe est abordée au chapitre suivant.

Remarque

Licence logicielle SIMATIC Automation Tool requise pour la méthode `ScanNetworkDevices`.

Si aucune licence logicielle pour SIMATIC Automation Tool n'est trouvée en runtime, la méthode `ScanNetworkDevices` renvoie une collection vide. Aucune information sur l'appareil n'est rapportée à l'application appelante.

7.8.6 Méthode `SetCommunicationsTimeout`

Type de retour	Nom de la méthode
Result	<code>SetCommunicationsTimeout</code>

Paramètres			
Nom	Type de données	Type de paramètre	Description
<code>nTimeout</code>	<code>uint</code>	Out	Une collection contenant un élément <code>IProfinetDevice</code> pour chaque appareil accessible sur le réseau industriel.

Vous pouvez définir un délai pour les opérations appelées. `SetCommunicationsTimeout` vous permet de fixer un délai allant de 180 à 999 secondes. Toute valeur en dehors de cette plage entraîne un échec de l'opération.

L'exemple suivant montre comment utiliser cette méthode pour fixer un délai dans le cas des opérations ScanNetworkDevices.

```
Network myNetwork = new Network();

IProfinetDeviceCollection scannedDevices;

Result retVal = Network.SetCommunicationsTimeout(180); //Expiration de la
temporisation dans 3 minutes
retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (retVal.Succeeded)
{
    //-----
    // L'action a été terminée avec succès. Poursuivre les opérations.
    //-----
}
```

7.8.7 Méthode GetCommunicationsTimeout

Type de retour	Nom de la méthode
uint	GetCommunicationsTimeout

Une fois qu'une valeur de temporisation a été définie pour la communication, vous pouvez récupérer la valeur de temporisation avec l'appel `GetCommunicationsTimeout`. Cette méthode renvoie la valeur de temporisation actuelle.

L'exemple suivant montre comment récupérer la valeur de temporisation actuelle et définir une nouvelle valeur de temporisation de 180 secondes si la valeur de temporisation actuelle est supérieure à 180 secondes.

```
uint timeout = Network.GetCommunicationsTimeout();
if (timeout > 180)
{
    Result retVal = Network.SetCommunicationsTimeout(180);
}
```

7.8.8 Méthode CheckValidLicense

Type de retour	Nom de la méthode
Result	CheckValidLicense

Une licence valide est nécessaire pour utiliser l'API. La méthode `CheckValidLicense` peut être appelée pour déterminer s'il existe une licence valide.

L'exemple suivant montre l'utilisation de la méthode `CheckValidLicense`.

```
Result result = Network.CheckValidLicense();
if (result.Succeeded)
{
    //-----
    // Licence valide
    //-----
}
Else
{
    //-----
    // Pas de licence valide
    //-----
}
```

7.8.9 Méthode GetEmptyCollection

Type de retour	Nom de la méthode
IProfinetDeviceCollection	GetEmptyCollection

`IProfinetDeviceCollection` est une interface et ne peut être instanciée. Vous pouvez ouvrir un projet ou effectuer une importation sans effectuer de scan. Appelez cette méthode pour renvoyer une collection vide.

L'exemple suivant montre comment utiliser la méthode `GetEmptyCollection`.

```
IProfinetDeviceCollection collection = Network.GetEmptyCollection();
MemoryStream stream = new MemoryStream();

Result result = collection.WriteToStream(stream);
if (result.Succeeded)
{
    //-----
    // La collection a été sérialisée avec succès
    //-----
}
```

7.9 Classe IProfinetDeviceCollection

7.9.1 Itération des éléments dans la collection

7.9.1.1 Itération des éléments dans la collection

La méthode `ScanNetworkDevices` renvoie un objet de type `IProfinetDeviceCollection`. Cette classe offre la capacité à itérer les éléments dans la collection de différentes manières. Elle fournit également des méthodes pour filtrer les éléments de la collection en fonction de certains critères. Les sections suivantes décrivent la fonctionnalité disponible pour la collection.

```
Network myNetwork = new Network();
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (retVal.Succeeded)
{
    for (int deviceIdx = 0; deviceIdx < scannedDevices.Count; deviceIdx++)
    {
        //-----
        // Chaque élément de la collection est un appareil IProfinet.
        // Cette interface est décrite en détail dans la partie suivante
        //-----
        IProfinetDevice dev = scannedDevices[deviceIdx];
    }
}
```

La collection prend également en charge l'itération à l'aide de la syntaxe `foreach`. L'exemple suivant montre la même itération dans cette syntaxe :

```
Network myNetwork = new Network();
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (retVal.Succeeded)
{
    foreach (IProfinetDevice dev in scannedDevices)
    {
        //-----
        // La variable "dev" représente maintenant l'élément suivant dans la
        // collection
        //-----
    }
}
```

7.9.1.2 Méthode GetEnumerator

Type de retour	Nom de la méthode
IEnumera- tor<IProfinetDevice >	GetEnumerator

Paramètres			
Nom	Type de données	Type de paramètre	Description
Aucune			

Cette méthode permet d'énumérer tous les `IProfinetDevices` dans la `IProfinetDeviceCollection`.

7.9.1.3 Propriété décompte

Type de retour	Nom de la propriété
int	Count

Cette propriété renvoie le décompte du nombre de `IProfinetDevices` dans la `IProfinetDeviceCollection`.

7.9.1.4 Propriété []

Type de retour	Nom de la propriété
<code>IProfinetDevice</code>	<code>this[int index]</code>

Cette propriété renvoie le `IProfinetDevice` à un index spécifique. Voir l'exemple ci-dessous.

```
IProfinetDeviceCollection collection = Network.GetEmptyCollection();
MemoryStream stream = new MemoryStream();

Result result = collection.WriteToStream(stream);
if (RetVal.Succeeded)
{
    //-----
    // La collection a été sérialisée avec succès
    //-----
    IProfinetDevice device = collection[0];
}
```

7.9.2 Filtrage des éléments dans la collection

7.9.2.1 Éléments de la collection

La collection contiendra un élément pour chaque appareil du réseau industriel Ethernet. La collection peut contenir des appareils provenant de gammes de produits différents (p. ex. S7-1200, S7-1500, ET200S).

La collection peut aussi contenir différentes "catégories" d'appareils (p. ex. CPU ou stations E/S). Pour différentes catégories d'appareils, des opérations spécifiques sont disponibles. Il peut donc être utile de filtrer la collection pour inclure uniquement certains appareils.

7.9.2.2 Méthode FilterByDeviceFamily

Type de retour	Nom de la méthode
List<IProfinetDevice>	FilterByDeviceFamily

Paramètres			
Nom	Type de données	Type de paramètre	Description
Familiestoinclude	List<DeviceFamily>	Int	Gammes à inclure

Cette méthode renvoie une collection qui comprend uniquement les appareils des gammes de produit spécifiées. Le filtre est d'abord construit sous forme de liste d'une ou plusieurs gammes d'appareils. Par exemple, cette déclaration crée un filtre uniquement pour les appareils S7-1200 et S7-1500.

Transmettez ce filtre à la méthode `FilterByDeviceFamily`. Le résultat est une `IProfinetDeviceCollection` qui contient uniquement les appareils des gammes de produits spécifiées.

```
Network myNetwork = new Network();
List<DeviceFamily> fams = new List<DeviceFamily> { DeviceFamily.CPU1200,
DeviceFamily.CPU1500 };

IProfinetDeviceCollection scannedDevices;
Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);

List<IProfinetDevice> onlyPlus = scannedDevices.FilterByDeviceFamily(fams);
```

La collection qui en résulte peut être itérée pour exécuter des actions uniquement sur les appareils inclus.

Remarque

La transmission d'un `List<DeviceFamily>` vide entraîne le renvoi d'une collection vide.

7.9.2.3 Méthode FilterOnlyCPUs

Type de retour	Nom de la méthode
List<ICPU>	FilterOnlyCPUs

L'API SIMATIC Automation Tool prend en charge de nombreuses opérations qui sont autorisées uniquement pour les CPU. Il est donc utile de filtrer les collections pour inclure uniquement les CPU détectées sur le réseau.

```
Network myNetwork = new Network();
IProfinetDeviceCollection scannedDevices;
Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);

List<ICPU> cpus = scannedDevices.FilterOnlyCpus();
foreach (ICPU cpu in cpus)
{
    //-----
    // Effectuer une itération dans la liste n'incluant que des appareils CPU
    //-----
}
```

Cette méthode retourne une liste d'`ICPU`. De plus, les opérations API sont prises en charge pour les CPU. L'interface `ICPU` fournit ces opérations. L'interface `ICPU` est décrite en détails au chapitre Interface ICPU (Page 187).

7.9.3 Rechercher un appareil spécifique dans la collection

7.9.3.1 Méthode FindDeviceByIP

Vous pouvez rechercher un appareil spécifique dans la collection.

Type de retour	Nom de la méthode
IProfinetDevice	FindDeviceByIP

Paramètres			
Nom	Type de données	Type de paramètre	Description
ip	uint	In	L'adresse IP recherchée

L'exemple suivant montre la recherche d'un appareil se trouvant à une adresse IP donnée. Si l'appareil est introuvable dans la collection, une référence NULL (zéro) est renvoyée.

```
Network myNetwork = new Network();
uint targetIPAddress = 0xC0A80001; // 192.168.0.1
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (!retVal.Succeeded)
    return;

IProfinetDevice dev = scannedDevices.FindDeviceByIP(targetIPAddress);
if (dev != null)
{
    // Appareil trouvé
}
```

7.9.3.2 Méthode FindDeviceByMAC

La méthode `FindDeviceByMAC` peut rechercher un appareil avec une adresse MAC spécifique.

Type de retour	Nom de la méthode
<code>IProfinetDevice</code>	<code>FindDeviceByMAC</code>

Paramètres			
Nom	Type de données	Type de paramètre	Description
<code>mac</code>	<code>ulong</code>	<code>In</code>	L'adresse MAC recherchée

Dans l'exemple suivant, on recherche un appareil à une adresse MAC donnée. Si l'appareil est introuvable dans la collection, une référence NULL (zéro) est renvoyée.

```
ulong targetMAC = 0x112233445566; // équivalent à la chaîne de caractères
11:22:33:44:55:66
Network myNetwork = new Network();
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (!retVal.Succeeded)
    return;

IProfinetDevice dev = scannedDevices.FindDeviceByMAC(targetMAC);
if (dev != null)
{
    // Appareil trouvé
}
```

7.9.4 Sérialisation

7.9.4.1 Transfert d'une collection de/vers un fichier de données externe

Les méthodes suivantes sont fournies pour permettre de sérialiser et de transférer le contenu d'une collection de/vers un fichier de données externe. Ces méthodes sont utilisées par l'application SIMATIC Automation Tool pour prendre en charge les fichiers projet.

7.9.4.2 Méthode WriteToStream

Type de retour	Nom de la méthode
Result	WriteToStream

Paramètres			
Nom	Type de données	Type de paramètre	Description
stream	Stream	In	La destination de la sortie en série de la collection.

Cette méthode est utilisée pour stocker en externe les contenus de la collection. L'exemple suivant montre l'utilisation de cette méthode :

```
Network myNetwork = new Network();
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (!retVal.Succeeded)
    return;

FileStream f = File.Create("myDataFile.SAT");

retVal = scannedDevices.WriteToStream(f);

f.Close();
```

Cette méthode sérialise en interne les données sur les versions, afin d'assurer la compatibilité en amont des données enregistrées.

7.9.4.3 Méthode ReadFromStream

La méthode `ReadFromStream` est utilisée pour créer la collection à partir d'un fichier de sérialisation précédemment créé. L'exemple suivant montre comment utiliser cette méthode.

Type de retour	Nom de la méthode
Result	ReadFromStream

Paramètres			
Nom	Type de données	Type de paramètre	Description
stream	Stream	In	La source de désérialisation de la collection

Cette méthode est utilisée pour créer la collection à partir d'un fichier de sérialisation précédemment créé. L'exemple suivant montre l'utilisation de cette méthode :

```
IProfinetDeviceCollection devices = Network.GetEmptyCollection();

FileStream f = File.OpenRead("myDataFile.SAT");

Result retVal = devices.ReadFromStream(f);

f.Close();
```

7.9.5 Ajouter manuellement des éléments à la collection

Selon la topologie physique du réseau industriel, il se peut qu'il contienne des appareils qui ne peuvent pas répondre à une commande DCP (comme ceux utilisés dans la méthode `ScanNetworkDevices`), mais qui sont accessibles via leur adresse IP. Pour ce scénario, les méthodes sont fournies pour permettre d'ajouter manuellement un appareil à la collection sur la base de son adresse.

7.9.5.1 Méthode InsertDeviceByIP

Type de retour	Nom de la méthode
<code>Result</code>	<code>InsertDeviceByIp</code>

Paramètres			
Nom	Type de données	Type de paramètre	Description
<code>index</code>	<code>int</code>	<code>In</code>	Emplacement où insérer la valeur dans la collection
<code>ip</code>	<code>uint</code>	<code>In</code>	L'adresse IP de l'appareil à ajouter.

Le code suivant scanne le réseau et ajoute manuellement un appareil à une adresse IP donnée à l'index spécifié.

```
Network myNetwork = new Network();
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (!retVal.Succeeded)
    return;

UInt32 missingDeviceIPAddress = 0xC0A80001; // 192.168.0.1
Int32 index = 0;

retVal = scannedDevices.InsertDeviceByIP(index, missingDeviceIPAddress);
```

7.9.5.2 Méthode InsertDeviceByMAC

Type de retour	Nom de la méthode
Result	InsertDeviceByMAC

Paramètres			
Nom	Type de données	Type de paramètre	Description
index	int	In	Emplacement où insérer la valeur dans la collection
mac	ulong	In	L'adresse MAC de l'appareil à ajouter à la collection.

Le code suivant scanne le réseau et ajoute manuellement un appareil à une adresse MAC donnée à l'index spécifié.

```
Network myNetwork = new Network();
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (!retVal.Succeeded)
    return;

UInt64 targetMAC = 0x112233445566; // équivalent à la chaîne de caractères
11:22:33:44:55:66
Int32 index = 0;

retVal = scannedDevices.InsertDeviceByMAC(index, targetMAC);
```

7.9.6 Copie de données depuis une collection

7.9.6.1 Méthode CopyUserData

Dans certains cas, il peut être nécessaire de conserver les options définies dans la `IProfinetDeviceCollection` actuelle. Plutôt que de demander à l'utilisateur d'entrer de nouveau ces informations, l'API propose la méthode suivante.

Type de retour	Nom de la méthode
<code>Result</code>	<code>CopyUserData</code>

Paramètres			
Nom	Type de données	Type de paramètre	Description
<code>oldNetwork</code>	<code>IProfinetDeviceCollection</code>	<code>In</code>	Liste précédente utilisée dans l'application

Le code suivant copie des données saisies par l'utilisateur d'un scan de réseau à un autre.

```
Network myNetwork = new Network();
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (!retVal.Succeeded)
    return;

IProfinetDeviceCollection rescannedDevices;
retVal = myNetwork.ScanNetworkDevices(out rescannedDevices);
if (!retVal.Succeeded)
    return;

retVal = rescannedDevices.CopyUserData(scannedDevices);
if (!retVal.Succeeded)
    return;
```

7.9.7 Supprimer des appareils de la collection

7.9.7.1 Méthode Clear

Type de retour	Nom de la méthode
void	Clear

Paramètres			
Nom	Type de données	Type de paramètre	Description
Aucun			

Cette méthode est utilisée pour supprimer le contenu des appareils scannés.

7.9.7.2 Méthode Remove

Type de retour	Nom de la méthode
void	Remove

Paramètres			
Nom	Type de données	Type de paramètre	Description
device	IProfinetDevice	In	Appareil à supprimer de la liste

Cette méthode est utilisée pour supprimer un élément spécifique de la collection.

7.10 Interface IProfinetDevice

7.10.1 Propriétés IProfinetDevice

Chaque élément de la collection `IProfinetDeviceCollection` est représenté par l'interface `IProfinetDevice`. L'interface fournit un accès aux données et aux opérations qui sont communes à tous les appareils directement connectés sur le réseau industriel.

L'interface `IProfinetDevice` prend en charge les propriétés suivantes, qui fournissent des informations sur l'appareil du réseau. Ces propriétés sont en lecture seule. Pour être sûr d'obtenir les informations en retour, le code doit d'abord appeler la méthode `RefreshStatus` sur l'appareil.

Nom de la propriété	Type de retour	Description
ArticleNumber	<code>string {get;}</code>	Numéro de commande du module. Également appelé MLFB ou "numéro d'article".
Comment	<code>string {get; set;}</code>	Ceci permet à l'utilisateur de formuler un commentaire sur l'appareil utilisé dans l'interface utilisateur SIMATIC Automation Tool. Le commentaire ne concerne pas les opérations API.
Configured	<code>bool {get;}</code>	Vrai, quand l'appareil a une configuration valide.
DefaultGateway	<code>uint {get;}</code>	L'adresse passerelle par défaut de l'appareil, représentée sous la forme d'un entier non signé. L'adresse passerelle encodée utilise un octet pour représenter chaque décimale dans l'adresse. Par exemple, la valeur encodée 0xC0A80001 est équivalente à la chaîne de représentation plus courante 192.168.0.1
DefaultGatewayString	<code>string {get;}</code>	L'adresse passerelle par défaut de l'appareil, représentée sous la forme d'une chaîne de caractères de type "xx.xx.xx.xx" (ex. "192.168.0.1")
Description	<code>string {get;}</code>	Une description du matériel, sur la base du numéro d'article. Il s'agit de la même description que celle affichée dans TIA Portal. (par ex. "CPU-1215 DC/DC/DC")

Nom de la propriété	Type de retour	Description
DeviceFound	bool {get;}	L'appareil a-t-il été détecté lors du scan du réseau ?
DuplicateIP	bool {get;}	L'appareil présente-t-il une adresse IP qui existe déjà ?
DuplicateProfinetName	bool {get;}	L'appareil présente-t-il un nom PROFINET qui existe déjà ?
Failsafe	bool {get;}	D'après son ArticleNumber, l'appareil est-il un appareil de sécurité ?
Family	DeviceFamily {get;}	De quelle gamme est cet appareil ? Pour plus d'informations, reportez-vous à la description de l'énumération DeviceFamily.
FirmwareUpdateAllowed	bool {get;}	Cet appareil prend-il en charge la mise à jour du firmware ?
FirmwareVersion	string {get;}	La version de firmware actuelle de l'appareil
ID	uint {get;}	L'identifiant unique de chaque appareil et module dans la station. Il est utilisé comme identifiant unique pour l'exécution d'une FirmwareUpdate.
HardwareNumber	short {get;}	La version matérielle ou "version de firmware" de l'appareil. (état fonctionnel)
IP	uint {get;}	L'adresse IP de l'appareil, représentée sous la forme d'un entier non signé. L'adresse IP encodée utilise un octet pour représenter chaque décimale dans l'adresse. Par exemple, la valeur encodée 0xC0A80001 est équivalente à la chaîne de représentation plus courante "192.168.0.1" REMARQUE : SIMATIC Automation Tool V3.1 prend en charge uniquement les adresses IPv4. L'adressage IPv6 n'est pas pris en charge.
IPString	string {get;}	L'adresse IP de l'appareil, représentée sous la forme d'une chaîne de caractères de type "xx.xx.xx.xx" (par ex. 192.168.0.1)

Nom de la propriété	Type de retour	Description
MAC	ulong {get;}	L'adresse MAC unique assignée à l'appareil. L'adresse MAC encodée utilise un octet pour représenter chacun des 6 octets définis pour l'adresse. Par exemple, l'adresse MAC encodée 0x112233445566 est équivalente à la chaîne de représentation plus courante 11:22:33:44:55:66
MACString	string {get;}	L'adresse MAC unique assignée à l'appareil représentée sous la forme d'une chaîne de caractères de type "11:22:33:44:55:66".
Modules	IModuleCollection {get;}	Une collection de modules raccordés à la station. Cette propriété est décrite ici en détail.
Name	string {get;}	Le nom de l'appareil
NewFirmwareFile	string {get;}	Emplacement du fichier de firmware à utiliser dans la mise à jour du firmware.
NewFirmwareNameErrorCode	Result {get;}	ErrorCode associé au nom du nouveau firmware
NewFirmwareNameIsValid	bool {get;}	Le fichier de firmware défini est-il valide ?
NewFirmwareVersion	string {get;}	Cette propriété est utilisée dans l'interface utilisateur de SIMATIC Automation Tool. Elle ne concerne pas les opérations API.
NewDefaultGateway	String {get;}	Cette propriété est utilisée dans l'interface utilisateur de SIMATIC Automation Tool. Elle ne concerne pas les opérations API.
NewIP	String {get;set;}	Cette propriété est utilisée dans l'interface utilisateur de SIMATIC Automation Tool. Elle ne concerne pas les opérations API.
NewProfinetName	String {get;set;}	Cette propriété est utilisée dans l'interface utilisateur de SIMATIC Automation Tool. Elle ne concerne pas les opérations API.
NewProgramName	String {get;set;}	Cette propriété est utilisée dans l'interface utilisateur de SIMATIC Automation Tool. Elle ne concerne pas les opérations API.

Nom de la propriété	Type de retour	Description
NewRestoreName	String{get; set;}	Cette propriété est utilisée dans l'interface utilisateur de SIMATIC Automation Tool. Elle ne concerne pas les opérations API.
ProfinetName	String {get;}	Le nom PROFINET de l'appareil.
ResetToFactoryAllowed	bool {get;}	La restauration des paramètres d'usine (ResetToFactory) est-elle autorisée sur l'appareil ?
Selected	bool {get; set;}	Marque l'appareil comme étant sélectionné pour permettre l'exécution des opérations.
SerialNumber	string {get;}	Le numéro de série univoque de l'appareil.
Slot	uint {get;}	Le numéro d'emplacement du matériel.
SlotName	string {get;}	Cette propriété est utilisée dans l'interface utilisateur de SIMATIC Automation Tool. Elle ne concerne pas les opérations API.
StationNumber	uint {get;}	Le numéro de station de l'appareil.
SubSlot	uint {get;}	Le sous-emplacement de l'appareil. Ceci vaut pour les sous-modules enfichables tels que les modules SB S7-1200.
Supported	bool {get;}	Vrai, lorsque le numéro de référence est sorte dans la base de données et que l'appareil est pris en charge par l'API SIMATIC Automation actuelle.
SubnetMask	uint {get;}	L'adresse du masque de sous-réseau, représentée sous la forme d'un entier non signé. Le masque de sous-réseau encodé utilise un octet pour représenter chaque décimale dans l'adresse. Par exemple, la valeur encodée 0xFFFFFFF0 est équivalente à la chaîne de représentation plus courante 255.255.255.0.
SubnetMaskString	string {get;}	Le masque de sous-réseau de l'appareil représenté sous la forme d'une chaîne de caractères de type "xx.xx.xx.xx" (par ex. 192.168.0.1)

Voir aussi

[DeviceFamily \(Page 242\)](#)

[Propriété de modules et classe IModuleCollection \(Page 185\)](#)

7.10.2 Méthode IProfinetDevice

7.10.2.1 Méthode RefreshStatus

Type de retour	Nom de la méthode
Result	RefreshStatus

A la création de la collection `IProfinetDeviceCollection` par l'appel de la méthode `ScanNetworkDevices`, la quantité d'informations obtenue concernant chaque appareil est minimale. Pour obtenir toutes les informations disponibles concernant l'appareil, il est nécessaire d'appeler la méthode `RefreshStatus`. Cette méthode établit une connexion à l'appareil, recherche les différentes informations et se déconnecte de l'appareil.

Le code suivant appelle `RefreshStatus` pour chaque appareil du réseau.

```
Network myNetwork = new Network();

IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (retVal.Succeeded)
{
    foreach (IProfinetDevice dev in scannedDevices)
    {

        retVal = dev.RefreshStatus();
        if (retVal.Succeeded)
        {
            //-----
            // Opération réussie, les données sont fiables.
            //-----
        }
    }
}
```

La méthode `RefreshStatus` se connecte à l'appareil pour lire les informations. L'appareil peut être protégé par un mot de passe contre ce type d'accès. C'est pourquoi cette méthode (comme toutes celles qui se connectent en interne à l'appareil) requiert un paramètre Mot de passe.

L'exemple montre la classe `IProfinetDevice`. `ICPU` a besoin d'un mot de passe défini sur l'appareil avant d'appeler la méthode `RefreshStatus` en utilisant `SetPassword(EncryptedString)` pour légitimer la connexion.

Voir aussi

Classe EncryptedString (Page 141)

7.10.2.2 Méthode FirmwareUpdate

Type de retour	Nom de la méthode
Result	FirmwareUpdate

Paramètres			
Nom	Type de données	Type de paramètre	Description
hardwareID	uint	In	Identification matérielle du module
bUpdateSameVersion	Bool	In	Si vrai, la méthode procède à la mise à jour. La mise à jour est effectuée si le fichier de mise à jour indique que la version est identique à la version de firmware en cours sur le module.

Cette méthode met à jour la version de firmware du matériel spécifié (`hardwareID`) sur l'appareil. Le `hardwareID` peut spécifier l'appareil lui-même ou un module sur le même châssis.

Certains appareils ne prennent pas en charge la fonction de mise à jour du firmware. La propriété `FirmwareUpdateAllowed` peut être vérifiée pour s'assurer que l'appareil concerné prend en charge cette fonction.

Dans l'exemple suivant, on recherche un appareil à une adresse IP donnée et le firmware est mis à jour sur cet appareil.

```
Network myNetwork = new Network();
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
uint targetIPAddress = 0x0A80001; // 192.168.0.1
string updateFile = @"c:\myUpdates\6ES7 221 - 1BF32 - 0XB0 V02.00.00.upd";

if (!retVal.Succeeded)
    return;

IProfinetDevice dev = scannedDevices.FindDeviceByIP(targetIPAddress);
if (dev != null)
{
    dev.Selected = true;
    dev.SetFirmwareFile(updateFile);
    retVal = dev.FirmwareUpdate(dev.ID, true);
}
```

En utilisant la méthode `FirmwareUpdate`, il est également possible de mettre à jour le firmware d'un module sur une station centrale. Le code suivant montre comment rechercher une CPU à une adresse donnée et recherche ensuite un numéro d'article spécifique parmi les modules de la CPU. Le firmware est ensuite mis à jour sur les modules qui correspondent aux critères de recherche.

```
Network myNetwork = new Network();
IProfinetDeviceCollection scannedDevices;
Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
uint targetIPAddress = 0xC0A80001; // 192.168.0.1
string targetModule = @"6ES7 221 - 1BF32 - 0XB0";
string updateFile = @"c:\myUpdates\6ES7 221 - 1BF32 - 0XB0 V02.00.00.upd";

if (!retVal.Succeeded)
    return;

IProfinetDevice dev = scannedDevices.FindDeviceByIP(targetIPAddress);
if (dev != null)
{
    retVal = dev.RefreshStatus();
    if (!retVal.Succeeded)
        return;

    //-----
    // Rechercher les modules sur la CPU.
    //-----
    IModuleCollection mods = dev.Modules;
    foreach (IModule mod in mods)
    {
        if (mod.ArticleNumber == targetModule)
        {
            mod.Selected = true;
            mod.SetFirmwareFile(updateFile);

            //-----
            // Mettre à jour le firmware pour le(s) module(s) correspondant(s)
            //-----
            dev.FirmwareUpdate(mod.ID, true);
        }
    }
}
```

Remarquez que la méthode `FirmwareUpdate` est appelée sur la CPU. Le `hardwareID` transmis à la méthode indique le module à mettre à jour.

Remarque**Fichiers de mise à jour du firmware Classic et Plus**

Il existe deux types de fichiers de mise à jour du firmware.

- Les dossiers de mise à jour du firmware Classic qui constituent la mise à jour du firmware. Le fichier header.upd ou cpu_hd.upd dans ce dossier est le fichier transmis à la méthode `FirmwareUpdate`.
- Le fichier de mise à jour de firmware Plus est un fichier de mise à jour unique. Il s'agit du fichier transmis à la méthode `FirmwareUpdate`.

7.10.2.3 Méthode Identify

Type de retour	Nom de la méthode
Result	<code>Identify</code>

Cette méthode fait clignoter les LED ou l'écran IHM d'un appareil réseau donné. La lumière clignotante aide à identifier l'emplacement physique de l'appareil.

L'exemple suivant fait clignoter les LED ou l'écran de l'appareil utilisant l'adresse IP 192.168.0.1.

```
Network myNetwork = new Network();
uint targetIPAddress = 0xC0A80001; // 192.168.0.1
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (retVal.Succeeded)
{
    //-----
    // Rechercher l'appareil à cette adresse IP et faire clignoter les LED/l'écran
    IHM
    //-----
    IProfinetDevice dev = scannedDevices.FindDeviceByIP(targetIPAddress);
    if (dev != null)
    {
        retVal = dev.Identify();
    }
}
```

7.10.2.4 Méthode Reset

Type de retour	Nom de la méthode
Result	Reset

Cette méthode est utilisée pour restaurer les paramètres d'usine d'un appareil PROFINET.

L'exemple suivant appelle la méthode `Reset` pour un appareil se trouvant à une adresse IP donnée.

```
Network myNetwork = new Network();
uint targetIPAddress = 0xC0A80001; // 192.168.0.1
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (retVal.Succeeded)
{
    //-----
    // Rechercher l'appareil à cette adresse IP et restaurer les paramètres d'usine
    //-----
    IProfinetDevice dev = scannedDevices.FindDeviceByIP(targetIPAddress);
    if (dev != null)
    {
        retVal = dev.Reset();
    }
}
```

Remarque

Cette méthode ne peut pas servir à réinitialiser une CPU. L'interface `ICPU` prend en charge une méthode `ResetToFactory` spécifique pour les CPU.

7.10.2.5 Méthode SetIP

Type de retour	Nom de la méthode
Result	SetIP

Paramètres			
Nom	Type de données	Type de paramètre	Description
nIP	uint	In	Nouvelle adresse IP encodée
nSubnet	uint	In	Nouvelle adresse de sous-réseau encodée
nGateway	uint	In	Nouvelle adresse de passerelle encodée

Cette méthode est utilisée pour définir ou modifier l'adresse IP d'un appareil.

Pour que cette opération réussisse, la configuration du port doit être réglée sur l'option "IP address is set directly on the device" (L'adresse IP est définie directement sur l'appareil).

Dans l'exemple suivant, on recherche un appareil à une adresse MAC donnée et on définit l'adresse IP.

```
Network myNetwork = new Network();
ulong targetMACAddress = 0x112233445566; // 11:22:33:44:55:66
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (retVal.Succeeded)
{
    //-----
    // Rechercher l'appareil à cette adresse MAC et Définir IP
    //-----
    IProfinetDevice dev = scannedDevices.FindDeviceByMAC(targetMACAddress);
    if (dev != null)
    {
        retVal = dev.SetIP(0xC0A80001, 0xFFFFFFF00, 0x0);
    }
}
```

Remarque

La méthode `SetIP` suppose que les adresses ont un format encodé (comme indiqué ci-dessus). Les adresses peuvent être converties à partir d'un seul format dans un `uint` encodé à l'aide du code C# suivant :

```

string userEnteredAddress = @"192.168.0.1"; // Par exemple
//-----
// Convertir l'adresse string en uint
//-----
System.Net.IPEndPoint ip = IPEndPoint.Parse(userEnteredAddress);
byte[] bytes = ip.GetAddressBytes();
Array.Reverse(bytes);

uint encodedIp = BitConverter.ToUInt32(bytes, 0);
// adresse IP encodée disponible

```

7.10.2.6 Méthode SetProfinetName

Type de retour	Nom de la méthode
Result	SetProfinetName

Paramètres			
Nom	Type de données	Type de paramètre	Description
strName	String	In	Nouveau nom de la station PROFINET

Cette méthode est utilisée pour définir ou modifier la station PROFINET pour l'appareil. Pour que cette opération réussisse, le port doit être configuré sur l'option "PROFINET name is set directly on the device" (le nom d'appareil PROFINET est défini directement sur l'appareil).

```

Network myNetwork = new Network();
ulong targetMACAddress = 0x112233445566; // 11:22:33:44:55:66
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (retVal.Succeeded)
{
    //-----
    // Rechercher l'appareil à cette adresse MAC et Définir le nom PROFINET
    //-----
    IProfinetDevice dev = scannedDevices.FindDeviceByMAC(targetMACAddress);
    if (dev != null)
    {
        retVal = dev.SetProfinetName("new name");
    }
}

```

7.10.2.7 Méthode ValidateIPAddressSubnet

Type de retour	Nom de la méthode
Result	ValidateIPAddressSubnet

Paramètres			
Nom	Type de données	Type de paramètre	Description
nIP	uint	In	Adresse IP
nSubnetMask	uint	In	Masque de sous-réseau

Cette méthode permet de valider la combinaison adresse IP / masque de sous-réseau pour vérifier que la paire est compatible.

Dans l'exemple recherche un appareil à une adresse MAC donnée et valide la compatibilité de l'adresse IP et du masque de sous-réseau de l'appareil.

```
Network myNetwork = new Network();
ulong targetMACAddress = 0x112233445566; // 11:22:33:44:55:66
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (retVal.Succeeded)
{
    //-----
    // Rechercher l'appareil à cette adresse MAC et Définir IP
    //-----
    IProfinetDevice dev = scannedDevices.FindDeviceByMAC(targetMACAddress);

    if (dev != null)
    {
        retVal = dev.ValidateIPAddressSubnet(dev.IP, dev.SubnetMask);
    }
}
```

7.10.2.8 Méthode ValidatePROFINETName

Type de retour	Nom de la méthode
Result	ValidatePROFINETName

Paramètres			
Nom	Type de données	Type de paramètre	Description
strName	string	In	Nom PROFINET à valider

Cette méthode utilise un nom PROFINET que vous entrez. Elle teste la validité du nom PROFINET en tant que nom PROFINET d'un appareil.

Dans l'exemple suivant, on recherche un appareil à une adresse MAC donnée et on vérifie la validité du nom PROFINET avant de l'attribuer à l'appareil.

```
Network myNetwork = new Network();
ulong targetMACAddress = 0x112233445566; // 11:22:33:44:55:66
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (retVal.Succeeded)
{
    //-----
    // Rechercher l'appareil à cette adresse MAC et Définir IP
    //-----
    IProfinetDevice dev = scannedDevices.FindDeviceByMAC(targetMACAddress);

    if (dev != null)
    {
        string name = "ValidName";
        retVal = dev.ValidatePROFINETName(name);
        if (retVal.Succeeded)
        {
            retVal = dev.SetProfinetName(name);
        }
    }
}
```

7.10.3 Événements IProfinetDevice

7.10.3.1 Événement DataChanged

L'événement `DataChanged` est pris en charge sur l'interface `IProfinetDevice`.

Avec cet événement le programme peut voir si des modifications sont intervenues sur un appareil donné du réseau, du fait d'autres opérations sur l'API. Par exemple, si le programme garde une référence pour un `IProfinetDevice` donné, il est possible "d'écouter" certaines modifications apportées à l'appareil.

Dans l'exemple suivant, le code attache l'événement `DataChanged` pour chaque appareil du réseau.

```
private void AttachEvents(IProfinetDeviceCollection devices)
{
    foreach (IProfinetDevice dev in devices)
    {
        dev.DataChanged += new DataChangedEventHandler(Dev_DataChanged);
    }
}

private void Dev_DataChanged(object sender, DataChangedEventArgs e)
{
    if (e.Type == DataChangedType.OperatingState)
    {
        //-----
        // Le mode a changé pour cet appareil
        //-----
    }
}
```

Désormais, à chaque action de l'API entraînant le changement du mode de fonctionnement d'un appareil, la méthode `Dev_DataChanged` est appelée.

Remarque

L'événement `DataChanged` ne surveille pas directement le réseau réel, mais les propriétés de l'`IProfinetDevice`. L'état de cet objet doit changer pour que l'événement soit déclenché.

La classe DataChangedEventArgs

Le gestionnaire d'événements `DataChanged` reçoit un objet `DataChangedEventArgs`. Comme indiqué ci-dessus, cette classe possède une seule propriété (`type`) de type `DataChangedType`.

Voir aussi Énumération `DataChangedType` (Page 242)

7.10.3.2 Événement ProgressChanged

L'événement `ProgressChanged` est pris en charge sur l'interface `IProfinetDevice`.

Cet événement permet au programme de surveiller la progression de méthodes qui demandent du temps, comme par exemple `FirmwareUpdate`.

Un gestionnaire d'événements est attaché à l'événement pour que celui-ci soit utilisable. Le gestionnaire d'événements est appelé dès qu'il y a un changement dans la progression de l'opération.

L'exemple suivant montre comment vous pouvez surveiller la progression de l'exécution. Cet exemple montre une méthode qui met à jour le firmware d'un appareil du réseau. Cette opération peut prendre un temps notable. Afin de suivre la progression de l'action, un gestionnaire d'événements est défini et attaché à l'événement `ProgressChanged`. Une fois que la mise à jour du firmware est terminée, le gestionnaire est détaché de l'événement.

```
private void UpdateCpuAtAddress(IProfinetDeviceCollection devices, uint
targetIPAddress, string updateFile)
{
    IProfinetDevice dev = devices.FindDeviceByIP(targetIPAddress);
    if (dev != null)
    {
        dev.ProgressChanged += new
            ProgressChangedEventHandler(Dev_ProgressChanged);
        dev.SetFirmwareFile(updateFile);
        dev.FirmwareUpdate(dev.ID, true);

        dev.ProgressChanged -= new
            ProgressChangedEventHandler(Dev_ProgressChanged);
    }
}

private void Dev_ProgressChanged(object sender, ProgressChangedEventArgs e)
{
    IProfinetDevice device = sender as IProfinetDevice;
    double percent = 0;
    if (device != null)
    {
        if (e.Count != 0)
        {
            string sPercent = e.Index.ToString() + " %";
        }
    }
}
```

La classe ProgressChangedEventArgs

Le gestionnaire d'événements `ProgressChanged` reçoit un objet `ProgressChangedEventArgs`. L'objet présente les propriétés suivantes :

Nom de la propriété	Type de retour	Description
Action	<code>ProgressAction</code>	Description de l'action en cours. Voir aussi Énumération <code>ProgressAction</code> (Page 248)
Cancel	<code>bool</code>	L'action a-t-elle été annulée ?
Count	<code>int</code>	Quantité totale de données à transférer
ID	<code>uint</code>	ID du matériel
Index	<code>int</code>	Quantité totale de données transférées

7.11 Classe **IModuleCollection** et propriétés des modules

7.11.1 Propriété de modules et classe **IModuleCollection**

L'interface `IProfinetDevice` fournit des informations sur tous les modules (p. ex. modules d'entrées-sorties, signal boards, CM, CP) raccordés à la station. La propriété `Modules` renvoie une collection de ces modules.

Le code suivant montre comment accéder à ces informations pour un `IProfinetDevice` (créé dans l'exemple précédent).

```
Network myNetwork = new Network();
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (retVal.Succeeded)
{
    //-----
    // Pour s'assurer que les informations sont complètes et à jour,
    // appeler d'abord RefreshStatus()
    //-----
    retVal = scannedDevices[0].RefreshStatus();
    if (retVal.Succeeded)
    {
        //-----
        // La propriété Modules renvoie une collection de IModule
        //-----
        IModuleCollection modules = scannedDevices[0].Modules;
        foreach (IModule mod in modules)
        {
            //-----
            // Obtenir le numéro d'article de tous les modules de la station
            //-----
            string displayArticleNum = mod.ArticleNumber;
        }
    }
}
```

7.11.2 Interface `IModule`

Chaque module de la station est représenté sous la forme d'une interface `IModule`. Cette interface fournit un sous-ensemble de propriétés disponibles pour un appareil.

L'interface `IModule` ne fournit pas de méthode. Toutes les opérations sur un module doivent être initiées sur l'appareil.

L'interface `IModule` prend en charge les propriétés suivantes.

Nom de la propriété	Type de retour	Description
<code>ArticleNumber</code>	<code>string</code>	Numéro de commande du module. Également appelé MLFB ou "numéro d'article".
<code>Comment</code>	<code>string</code>	Ceci permet à l'utilisateur de formuler un commentaire sur l'appareil utilisé dans l'interface utilisateur SIMATIC Automation Tool. Le commentaire ne concerne pas les opérations API.
<code>Configured</code>	<code>bool</code>	Vrai, quand l'appareil a une configuration valide.
<code>Description</code>	<code>string</code>	Une description du matériel, sur la base du numéro d'article. Il s'agit de la même description que celle affichée dans TIA Portal. (par ex. "CPU-1215 DC/DC/DC")
<code>Failsafe</code>	<code>bool</code>	D'après son ArticleNumber, l'appareil est-il un appareil de sécurité ?
<code>FirmwareUpdateAllowed</code>	<code>bool</code>	Cet appareil prend-il en charge la mise à jour du firmware ?
<code>FirmwareVersion</code>	<code>string</code>	La version de firmware actuelle de l'appareil
<code>ID</code>	<code>uint</code>	Un identifiant unique pour chaque appareil dans la station. Il est utilisé comme identifiant unique pour l'exécution d'une <code>FirmwareUpdate</code> .
<code>Name</code>	<code>string</code>	Le nom de l'appareil.
<code>NewFirmwareNameErrorCode</code>	<code>Result</code>	<code>ErrorCode</code> associé au nom du nouveau firmware
<code>NewFirmwareNameIsValid</code>	<code>bool</code>	Vrai, si le fichier de firmware est valide pour cet appareil ou ce module.
<code>FirmwareVersion</code>	<code>string</code>	Cette propriété est utilisée dans l'interface graphique de SIMATIC Automation Tool. Elle ne concerne pas les opérations API.
<code>Selected</code>	<code>bool</code>	L'appareil est-il actuellement sélectionné ? Il s'agit de l'état indiqué par la case à cocher dans la GUI.
<code>SerialNumber</code>	<code>string</code>	Le numéro de série unique de l'appareil.
<code>Slot</code>	<code>uint</code>	Le numéro d'emplacement du matériel.
<code>SlotName</code>	<code>string</code>	Cette propriété est utilisée dans l'interface utilisateur de SIMATIC Automation Tool. Elle ne concerne pas les opérations API.
<code>StationNumber</code>	<code>uint</code>	Le numéro de station de l'appareil.
<code>SubSlot</code>	<code>uint</code>	Le sous-emplacement de l'appareil. Ceci vaut pour les sous-modules enfichables tels que les modules SB S7-1200.
<code>Supported</code>	<code>FeatureSupport</code>	L'appareil réseau détecté est-il pris en charge par les opérations actuelles de SIMATIC Automation Tool ?

7.12 Interface ICPU

7.12.1 Identification de l'appareil CPU dans un IProfinetDeviceCollection

Comme indiqué ci-dessus, la méthode `ScanNetworkDevices` est appelée pour générer une `IProfinetDeviceCollection`. Cette collection contient un élément pour chaque appareil accessible sur le réseau industriel. Ces appareils peuvent être des CPU et des stations d'E/S décentralisées.

L'interface `IProfinetDevice` fournit les propriétés et les méthodes qui s'appliquent à toutes les catégories d'appareils. Cependant, certaines propriétés et méthodes sont spécifiques à un appareil CPU. Ces propriétés et méthodes sont accessibles à l'aide de l'interface `ICPU`.

Pour déterminer si une interface `IProfinetDevice` donnée représente effectivement un appareil CPU, placez-la dans une `ICPU`. Si l'opération réussit, l'appareil est bien une CPU, et vous pouvez utiliser les méthodes/propriétés sur l'interface `ICPU`. L'exemple suivant illustre la procédure.

```
Network myNetwork = new Network();
IProfinetDeviceCollection scannedDevices;
Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (retVal.Succeeded)
{
    foreach (IProfinetDevice dev in scannedDevices)
    {
        ICPU devAsCpu = dev as ICPU;
        if (devAsCpu != null)
        {
            //-----
            // L'appareil est une CPU.
            // L'interface ICPU peut être utilisée pour interagir avec elle.
            //-----
        }
    }
}
```

Remarque

L'interface `ICPU` hérite de l'`IProfinetDevice`. De ce fait, les propriétés et méthodes prises en charge sur le `IProfinetDevice` le sont aussi sur l'interface `ICPU`.

Remarque

La nouveauté, pour l'interface `ICPU` de l'API V3.1, est l'obligation de définir les drapeaux `Selected` et `SelectedConfirmed` avant d'exécuter des opérations sur les appareils via l'API. Le drapeau `Selected` est exigé pour toutes les fonctions rattachées à l'interface `ICPU`, tandis que le drapeau `SelectedConfirmed` doit être défini si l'`ICPU` représente une opération de sécurité. Les opérations relatives à la sécurité sont la mise à jour d'un programme, le formatage de la carte mémoire, la restauration des paramètres d'usine et la restauration.

7.12.2 Propriétés ICPU

L'interface `ICPU` étend l'`IProfinetDevice` en ajoutant les propriétés suivantes. Ces propriétés sont en lecture seule. Pour être sûr d'obtenir les informations actuelles, le code doit d'abord appeler la méthode `RefreshStatus`.

Nom de la propriété	Type de retour	Description
<code>RemoteInterfaces</code>	<code>List<IRemoteInterface></code>	Une liste de toutes les interfaces E/S décentralisées pour la CPU. L'utilisation de cette propriété est décrite plus loin dans ce document.
<code>DataLogFolder</code>	<code>IRemoteFolder</code>	Informations concernant les Data Logs identifiés sur la carte mémoire SIMATIC de la CPU
<code>RecipeFolder</code>	<code>IRemoteFolder</code>	Informations sur les recettes identifiées sur la carte mémoire SIMATIC de la CPU
<code>OperatingMode</code>	<code>OperatingState</code>	Désigne le mode actuel de la CPU. Cette valeur est disponible uniquement en mode lecture.
<code>IdentityCrisis</code>	<code>bool</code>	Vrai, lorsque l'appareil ne peut pas être identifié.
<code>LastRefreshSuccessful</code>	<code>bool</code>	Vrai, lorsque le dernier appel à <code>RefreshStatus</code> a été effectué avec succès.
<code>SelectedConfirmed</code>	<code>bool</code>	Les méthodes qui exécutent des opérations relatives à la sécurité doivent régler le drapeau <code>SelectedConfirmed</code> sur TRUE, lorsque l'utilisateur resélectionne pour l'opération un ou plusieurs appareils à partir d'une boîte de dialogue de confirmation et confirme l'opération. <code>SelectedConfirmed</code> signifie que l'opération est sélectionnée et confirmée.
<code>Initialized</code>	<code>bool</code>	Vrai, lorsque l'appareil ou le module présente une configuration valide.
<code>InterfaceNumber</code>	<code>int</code>	L'interface via laquelle l'appareil est connecté
<code>Password</code>	<code>EncryptedString</code>	Mot de passe de la CPU utilisé dans les fonctions exécutées sur l'appareil.
<code>PasswordProtectionLevel</code>	<code>ProtectionLevel</code>	Le niveau de protection d'un mot de passe de la CPU légitime

Nom de la propriété	Type de retour	Description
Protected	bool	La CPU est-elle actuellement protégée par un mot de passe ? Cela signifie qu'un mot de passe est nécessaire pour accéder à tout ou partie des fonctionnalités, en fonction du niveau d'accès.
PasswordValid	bool	L'appel à <code>SetPassword()</code> est-il valide ?

Voir aussi Propriété RemoteInterfaces (Page 226)

7.12.3 Drapeaux ICPU

7.12.3.1 Drapeaux relatifs aux mises à jour de programmes

Pour exécuter avec succès des fonctions relatives à la sécurité sur un appareil, il faut obtenir des informations supplémentaires dans l'appareil. Les drapeaux suivants ont été ajoutés pour s'assurer que la fonction Mise à jour du programme peut être effectuée correctement et en toute sécurité sur un appareil de sécurité.

Nom de la propriété	Type de retour	Description
NewProgramNamePassword	EncryptedString	Mot de passe de la CPU utilisé pour une tentative de connexion une fois que la mise à jour du programme est achevée. La valeur est définie en utilisant <code>SetProgramPassword(EncryptedString)</code> .
HasSafetyProgram	bool	Valeur booléenne définie si l'appareil contient un programme de sécurité. Elle est déterminée lors de la connexion à une CPU.
NewProgramNameIsValid	bool	Vrai, lorsque la méthode <code>SetProgramFolder</code> est appelée avec un programme valide.
NewProgramNameIsSafety	bool	Vrai, lorsque la méthode <code>SetProgramFolder</code> est appelée avec un programme de sécurité valide.
NewProgramNameHasSafetyPassword	bool	Vrai, lorsque la méthode <code>SetProgramFolder</code> est appelée avec un programme de sécurité valide. Faux, si un programme standard est ouvert.
NewProgramNamePasswordIsValid	bool	Vrai, lorsque la méthode <code>SetProgramPassword</code> est appelée avec un mot de passe valide.
NewProgramNamePasswordIsSafety	bool	Vrai, lorsque la méthode <code>SetProgramPassword</code> est appelée avec un mot de passe valide et que le nouveau programme présente un mot de passe de CPU F de sécurité.
ProgramUpdateSucceeded	bool	Vrai, lorsque la méthode <code>ProgramUpdate</code> est exécutée avec succès. La mise à jour du programme peut toutefois renvoyer une erreur.

Nom de la propriété	Type de retour	Description
NewProgramNamePasswordPresent	bool	Vrai, lorsque la méthode <code>SetProgramFolder</code> est appelée et que le programme est protégé par un mot de passe.
NewProgramNamePasswordLevel	ProtectionLevel	Quel est le niveau de protection du mot de passe de la CPU pour le nouveau programme ?
NewProgramName	string	Quel est le nom du nouveau programme ?
NewProgramFolder	string	Quel est l'emplacement du dossier pour le nouveau programme ? La valeur est définie à l'aide de la méthode <code>SetProgramFolder</code> .
NewProgramNameFSignature	uint	Quelle est la FSignature du nouveau projet ? Utilisée dans la procédure de comparaison pour déterminer si la <code>ProgramUpdate</code> a été terminée avec succès.
NewProgramNameIP	uint	Adresse IP stockée dans le nouveau programme
NewProgramNameSubnetMask	uint	Masque de sous-réseau de l'appareil dans le nouveau programme
NewProgramNameGateway	uint	Passerelle de l'appareil dans le nouveau programme
NewProgramNameErrorCode	Result	Un moyen simple d'identifier les problèmes qui peuvent survenir lors de la validation du nouveau programme, par exemple si le programme est invalide pour l'appareil ou si l'adresse IP trouvée dans le programme existe déjà sur le réseau.
NewProgramNamePasswordErrorCode	Result	Stocke le code d'erreur du dernier appel dans la fonction <code>SetProgramPassword</code> .

7.12.3.2 Restaurer les drapeaux

Pour exécuter avec succès des fonctions relatives à la sécurité sur un appareil, il faut obtenir des informations supplémentaires dans l'appareil. Les drapeaux suivants sont ajoutés à l'API de la version 3.1, de sorte que la fonction Restaurer à partir de la sauvegarde peut être effectuée sur un appareil de sécurité correctement et en toute sécurité. En raison des différences entre le formatage du fichier programme et le fichier de restauration, toutes les informations présentes pour la mise à jour du programme ne peuvent pas être récupérées à partir des données du fichier de sauvegarde.

Nom de la propriété	Type de retour	Description
NewRestoreNamePassword	EncryptedString	Mot de passe de la CPU utilisé pour la tentative de connexion une fois que la restauration est achevée. La valeur est définie en utilisant SetBackupFilePassword(EncryptedString)
NewRestoreNameIsValid	bool	Vrai, lorsque la méthode SetBackupFile est appelée et que le fichier de restauration est valide
NewRestoreNameIsSafety	bool	Vrai, lorsque la méthode SetBackupFile est appelée et que le fichier de restauration est un fichier de restauration de sécurité
NewRestoreNamePasswordIsValid	bool	Vrai, lorsque l'appel SetBackupFilePassword contient un mot de passe valide.
NewRestoreNamePasswordIsSafety	bool	Vrai, lorsque l'appel SetBackupFilePassword contient un mot de passe de sécurité valide.
RestoreSucceeded	bool	L'opération Restore a-t-elle été effectuée avec succès ?
NewRestoreName	string	Quel est le nom du nouveau programme ?
NewRestoreFile	string	Quel est l'emplacement du fichier pour le nouveau programme ? La valeur est définie via la méthode SetBackupFile
NewRestoreNameFSignature	uint	Quelle est la FSignature du nouveau projet ? Utilisée dans la procédure de comparaison pour déterminer si l'opération Restore a été terminée avec succès
NewRestoreNameErrorCode	Result	Un moyen simple d'identifier les problèmes qui peuvent survenir lors de la validation du nouveau programme, par exemple si le programme est invalide ou incompatible avec l'appareil.

7.12.3.3 Drapeaux de fonctionnalités

Dans la version de l'API SIMATIC Automation Tool V3.0, les drapeaux indiquant si certaines fonctionnalités sont autorisées sur un appareil se trouvent au niveau `IProfinetDevice`.

L'API V3.1 a déplacé ces drapeaux vers les niveaux `ICPU` et `IHM`. Le type de retour de ces drapeaux est passé de `FeatureTypes` aux valeurs de type `bool`.

Nom de la propriété	Type de retour	Description
<code>ChangeModeAllowed</code>	<code>bool</code>	VRAI, si la CPU prend en charge une opération de changement de mode ? (RUN et STOP)
<code>BackupAllowed</code>	<code>bool</code>	VRAI, si la CPU prend en charge l'opération de sauvegarde.
<code>MemoryResetAllowed</code>	<code>bool</code>	VRAI, si la CPU prend en charge l'opération de restauration des paramètres d'usine.
<code>ProgramUpdateAllowed</code>	<code>bool</code>	VRAI, si la CPU prend en charge l'opération de mise à jour de programme.
<code>RestoreAllowed</code>	<code>bool</code>	VRAI, si la CPU prend en charge l'opération de restauration.
<code>FormatMCAllowed</code>	<code>bool</code>	VRAI, si la CPU prend en charge l'opération de formatage de la carte mémoire.
<code>PasswordAllowed</code>	<code>bool</code>	VRAI, si la CPU prend en charge un mot de passe.
<code>RemoteRecipesAllowed</code>	<code>bool</code>	VRAI, si la CPU prend en charge les opérations de recette.
<code>RemoteData-LogsAllowed</code>	<code>bool</code>	VRAI, si la CPU prend en charge les opérations relatives aux Data logs.
<code>ServiceDataAllowed</code>	<code>bool</code>	VRAI, si la CPU prend en charge l'opération de chargement des Service Data.
<code>SetTimeAllowed</code>	<code>bool</code>	VRAI, si la CPU prend en charge l'opération de réglage et de lecture de l'heure.
<code>DiagBufferAllowed</code>	<code>bool</code>	VRAI, si la CPU prend en charge l'opération de tampon de diagnostic.

7.12.4 Méthodes ICPU

7.12.4.1 ICPU protégées et mots de passe

Les méthodes suivantes sont fournies sur l'interface `ICPU`. La majorité des actions sur l'interface `ICPU` demande une connexion légitimée à la CPU. Un mot de passe peut être requis. Pour cette raison, la plupart des méthodes sur l'interface `ICPU` requièrent un paramètre mode de passe.

7.12.4.2 Méthode SetPassword

Type de retour	Nom de la méthode
<code>Result</code>	<code>SetPassword</code>

Paramètres			
Nom	Type de données	Type de paramètre	Description
<code>password</code>	<code>EncryptedString</code>	<code>In</code>	Définit le mot de passe de la CPU pour l'objet utilisé pour effectuer des opérations.

Auparavant, le paramètre de mot de passe de la CPU était traité en étant transmis à l'objet avec l'appel de méthode. SIMATIC Automation Tool V3.1 induit une nouveauté en vous permettant de définir le mot de passe de CPU pour l'objet au lieu d'envoyer le mot de passe de CPU à chaque fois avec les paramètres d'une méthode.

L'exemple suivant montre comment définir le mot de passe de CPU d'un appareil.

```
Network myNetwork = new Network();
ulong targetMACAddress = 0x112233445566; // 11:22:33:44:55:66
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (retVal.Succeeded)
{
    //-----
    // Rechercher l'appareil à cette adresse MAC et SetPassword
    //-----
    ICPU dev = scannedDevices.FindDeviceByMAC(targetMACAddress) as ICPU;
    if (dev != null)
    {
        if (dev.Protected)
        {
            retVal = dev.SetPassword(new EncryptedString("Password"));
        }
    }
}
```

7.12.4.3 Méthode SetProgramFolder

Type de retour	Nom de la méthode
Result	SetProgramFolder

Paramètres			
Nom	Type de données	Type de paramètre	Description
strFolder	string	In	Définir l'emplacement du dossier pour le programme téléchargé

Auparavant, le paramètre de chemin d'accès au dossier pour la mise à jour du programme était transmis à l'objet avec l'appel de méthode.

SIMATIC Automation Tool V3.1 induit une nouveauté en permettant à l'utilisateur de définir l'emplacement du dossier sur l'appareil.

La méthode définit les drapeaux suivants pour l'objet `ICPU`.

- `NewProgramFolder`
- `NewProgramName`
- `NewProgramNameIP`
- `NewProgramNameSubnetMask`
- `NewProgramNameGateway`
- `NewProgramNameIsValid`

Lorsque l'opération est effectuée sur un objet de sécurité, vous pouvez définir les éléments suivants :

- `NewProgramNameIsSafety`
- `NewProgramNameHasSafetyPassword`

L'exemple suivant montre comment définir le dossier programme sur un appareil.

```
Network myNetwork = new Network();
ulong targetMACAddress = 0x112233445566; // 11:22:33:44:55:66
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (retVal.Succeeded)
{
    //-----
    // Rechercher l'appareil à cette adresse MAC et SetProgramFolder
    //-----
    ICPU dev = scannedDevices.FindDeviceByMAC(targetMACAddress) as ICPU;
    if (dev != null)
    {
        retVal = dev.SetProgramFolder(@"C:\MyFolder");
        if (retVal.Error == ErrorCode.ProgramPasswordNeeded)
        {
            retVal = dev.SetPassword(new EncryptedString("Password"));
        }
    }
}
```

Remarque

Si le programme sélectionné possède un mot de passe, le `SetProgramFolder` retourne une erreur. Vous devez appeler la méthode `SetProgramPassword` et l'exécuter avec succès avant de pouvoir appeler la méthode `ProgramUpdate`.

7.12.4.4 Méthode SetProgramPassword

Type de retour	Nom de la méthode
Result	SetProgramPassword

Paramètres			
Nom	Type de données	Type de paramètre	Description
password	EncryptedString	In	Définit le mot de passe de la CPU pour le projet à transmettre à la CPU durant le ProgramUpdate.

Après la ProgramUpdate, l'application tente de se reconnecter à l'appareil. Si le programme placé sur la CPU est protégé par mot de passe, le fait de définir correctement le mot de passe de la CPU vous permet d'obtenir à nouveau l'accès à l'appareil, sans avoir besoin de définir un autre mot de passe pour la CPU.

La méthode définit les drapeaux suivants pour l'objet ICPU :

- NewProgramNamePasswordIsValid
- NewProgramNamePasswordIsSafety
- NewProgramNamePasswordLevel

L'exemple suivant montre comment définir un mot de passe de CPU actualisé sur un appareil.

```
Network myNetwork = new Network();
ulong targetMACAddress = 0x112233445566; // 11:22:33:44:55:66
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (retVal.Succeeded)
{
    //-----
    // Rechercher l'appareil à cette adresse MAC et SetProgramFolder
    //-----
    ICPU dev = scannedDevices.FindDeviceByMAC(targetMACAddress) as ICPU;
    if (dev != null)
    {
        retVal = dev.SetProgramFolder(@"C:\MyFolder");
        if (retVal.Error == ErrorCode.ProgramPasswordNeeded)
        {
            retVal = dev.SetPassword(new EncryptedString("Password"));
        }
    }
}
```

7.12.4.5 Méthode ProgramUpdate

Type de retour	Nom de la méthode
Result	ProgramUpdate

Cette méthode exécute la mise à jour d'un programme sur la CPU.

Les conditions préalables suivantes doivent être satisfaites pour pouvoir effectuer l'opération :

- L'appareil doit être sélectionné
 - S'il s'agit d'un appareil de sécurité, `SelectedConfirmed` doit être vrai
- Le `NewProgramFolder` pour l'objet doit être défini
 - Si le nouveau programme contient un mot de passe de CPU, le `NewProgramPasswordPresent` doit être vrai et sa valeur définie par `SetProgramPassword`
- La mise à jour du programme doit être prise en charge par l'appareil (`ProgramUpdateAllowed`)

Dans l'exemple suivant, on recherche une CPU à une adresse IP donnée dans l'`IProfinetDeviceCollection` et le programme de la CPU est mis à jour.

```
Network myNetwork = new Network();
ulong targetMACAddress = 0x112233445566; // 11:22:33:44:55:66
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (retVal.Succeeded)
{
    ICPU cpu = scannedDevices.FindDeviceByMAC(targetMACAddress) as ICPU;
    if (cpu != null && cpu.ProgramUpdateAllowed == true)
    {
        // Sélectionner la CPU à mettre à jour
        cpu.Selected = true;

        retVal = cpu.SetProgramFolder(@"C:\MyFolder");
        if (retVal.Error == ErrorCode.ProgramPasswordNeeded)
        {
            retVal = cpu.SetProgramPassword(new EncryptedString("Password"));
            if (retVal.Failed == true)
                return;
        }
        if (retVal.Failed == true)
            return;

        // Adresse IP unique ?
        if (cpu.DuplicateIP == true)
            return;

        // L'appareil est-il pris en charge ?
        if (cpu.Supported == false)
            return;
    }
}
```

```
// L'appareil est-il initialisé ?
if (cpu.Initialized == false)
    return;

// L'appareil est-il de sécurité ?
if (cpu.Failsafe == true)
{
    ConfirmationType type =
cpu.DetermineConfirmationMessage(FailsafeOperation.ProgramUpdateOperation);

    // Vérifier le type et confirmer
cpu.SelectedConfirmed = true;

    // Vérifier pour s'assurer que la mise à jour est faisable
if (cpu.HasSafetyProgram == true || cpu.Protected == true
|| cpu.NewProgramNameIsSafety == true)
{
    // L'appareil est-il protégé par mot de passe ?
    if (cpu.Protected == true)
    {
        // Un mot de passe valide a-t-il été fourni ?
        if (cpu.PasswordValid == false)
            return;

        // Sommes-nous légitimés pour le niveau de sécurité ?
        bool bSufficientAccess = cpu.PasswordProtectionLevel ==
ProtectionLevel.Failsafe;
        if (bSufficientAccess == false)
            return;

    }
}
}
else
{
    // Un programme de sécurité est-il sur le point d'être chargé ?
    if (cpu.NewProgramNameHasSafetyPassword == true)
        return;
}

// L'appareil est-il protégé par mot de passe ?
if (cpu.Protected == true)
{
    // Un mot de passe valide a-t-il été fourni ?
    if (cpu.PasswordValid == false)
        return;

    // Avons-nous un niveau de légitimation suffisant pour que l'opération
réussisse ?
    bool bSufficientAccess = cpu.PasswordProtectionLevel ==
ProtectionLevel.Failsafe || cpu.PasswordProtectionLevel == ProtectionLevel.Full;
```

```

        if (bSufficientAccess == false)
            return;
    }
    // Effectuer une mise à jour du programme
    retVal = cpu.ProgramUpdate();

    // Réinitialiser
    cpu.SelectedConfirmed = false;
}
}

```

Remarque

Le nom de dossier transmis à la méthode `ProgramUpdate` doit contenir un dossier nommé SIMATIC.S7S. Le dossier SIMATIC.S7S contient le programme à charger.

7.12.4.6 Méthode SetBackupFile

Type de retour	Nom de la méthode
Résultat	<code>SetBackupFile</code>

Paramètres			
Nom	Type de données	Type de paramètre	Description
<code>strFile</code>	string	In	Définit le lieu d'archivage du fichier de sauvegarde

Auparavant, le paramètre de chemin d'accès au fichier pour Restore était transmis à l'objet avec l'appel de méthode. SIMATIC Automation Tool V3.1 induit une nouveauté en vous permettant de définir l'emplacement du dossier sur la PG/PC.

La méthode définit les drapeaux suivants pour l'objet ICPU :

- `NewRestoreName`
- `NewRestoreFile`
- `NewRestoreNameIsValid`
- `NewRestoreNameIsSafety`
- `NewRestoreNameFSignature`

L'exemple suivant montre comment définir le chemin du fichier de sauvegarde.

```
Network myNetwork = new Network();
ulong targetMACAddress = 0x112233445566; // 11:22:33:44:55:66
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (retVal.Succeeded)
{
    ICPU cpu = scannedDevices.FindDeviceByMAC(targetMACAddress) as ICPU;
    if (cpu != null && cpu.RestoreAllowed == true)
    {
        // Sélectionner la CPU à mettre à jour
        cpu.Selected = true;

        retVal = cpu.SetBackupFile(@"C:\MyFolder\Backup.s7pbkp");
        if (retVal.Error == ErrorCode.ProgramPasswordNeeded)
        {
            retVal = cpu.SetBackupFilePassword(new EncryptedString("Password"));
            if (retValFailed == true)
                return;
        }

        if (retValFailed == true)
            return;

        // Adresse IP unique ?
        if (cpu.DuplicateIP == true)
            return;

        // L'appareil est-il pris en charge ?
        if (cpu.Supported == false)
            return;

        // L'appareil est-il initialisé ?
        if (cpu.Initialized == false)
            return;

        // L'appareil est-il de sécurité ?
        if (cpu.Failsafe == true)
        {
            ConfirmationType type =
            cpu.DetermineConfirmationMessage(FailsafeOperation.RestoreOperation);

            // Vérifier le type et confirmer
            cpu.SelectedConfirmed = true;

            // Vérifier pour s'assurer que la mise à jour est faisable
            if (cpu.HasSafetyProgram == true || cpu.Protected == true ||
            cpu.NewRestoreNameIsSafety == true)
            {
                // L'appareil est-il protégé par mot de passe ?

```

```

        if (cpu.Protected == true)
        {
            // Un mot de passe valide a-t-il été fourni ?
            if (cpu.PasswordValid == false)
                return;

            // Sommes-nous légitimés pour le niveau de sécurité ?
            bool bSufficientAccess = cpu.PasswordProtectionLevel ==
ProtectionLevel.Failsafe;
            if (bSufficientAccess == false)
                return;
        }
    }
else
{
    // Un programme de sécurité est-il sur le point d'être chargé ?
    if (cpu.NewRestoreNameIsSafety == true)
        return;
}

// L'appareil est-il protégé par mot de passe ?
if (cpu.Protected == true)
{
    // Un mot de passe valide a-t-il été fourni ?
    if (cpu.PasswordValid == false)
        return;

    // Avons-nous un niveau de légitimation suffisant pour que l'opération
réussisse ?
    bool bSufficientAccess = cpu.PasswordProtectionLevel ==
ProtectionLevel.Failsafe || cpu.PasswordProtectionLevel == ProtectionLevel.Full;
    if (bSufficientAccess == false)
        return;
}

// Effectuer une restauration
retVal = cpu.Restore();

// Réinitialiser
cpu.SelectedConfirmed = false;
}
}

```

Remarque

Si le fichier de sauvegarde sélectionné possède un mot de passe, le `SetBackupFile` retourne une erreur. La méthode `SetBackupFilePassword` doit être appelée et exécutée avec succès avant de pouvoir appeler la méthode `Restore`.

7.12.4.7 Méthode SetBackupFilePassword

Principe

Type de retour	Nom de la méthode
Result	SetBackupFilePassword

Paramètres			
Nom	Type de données	Type de paramètre	Description
password	EncryptedString	In	Définit le mot de passe pour le projet transmis à la CPU durant la restauration.

Après la restauration, l'application tente de se reconnecter à l'appareil. Si le programme chargé sur la CPU est protégé par mot de passe, le fait de paramétrer correctement la CPU vous permet d'obtenir à nouveau l'accès à l'appareil, sans avoir besoin de définir un autre mot de passe pour la CPU.

La méthode définit le drapeau suivant pour l'objet ICPU.

- NewRestoreNamePassword

L'exemple suivant montre comment définir un mot de passe de CPU actualisé sur un appareil.

```
Network myNetwork = new Network();
ulong targetMACAddress = 0x112233445566; // 11:22:33:44:55:66
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (retVal.Succeeded)
{
    //-----
    // Rechercher l'appareil à cette adresse MAC et SetBackupFilePassword
    //-----
    ICPU dev = scannedDevices.FindDeviceByMAC(targetMACAddress) as ICPU;
    if (dev != null)
    {
        if (dev.Protected)
        {
            retVal = dev.SetBackupFile(@"C:\MyFolder");
            retVal = dev.SetBackupFilePassword(new EncryptedString("Password"));
        }
    }
}
```

7.12.4.8 Méthode Restore (interface ICPU)

Type de retour	Nom de la méthode
Result	Restore

Cette méthode est utilisée pour restaurer les informations d'une sauvegarde précédente de la CPU. Certaines CPU ne prennent pas en charge la fonction Sauvegarde / Restauration.

Les conditions préalables suivantes doivent être satisfaites pour pouvoir effectuer l'opération.

- L'appareil doit être sélectionné.
 - S'il s'agit d'un appareil de sécurité, `SelectedConfirmed` doit être vrai.
- Le `NewProgramFolder` pour l'objet doit être défini.
 - Si le nouveau programme contient un mot de passe de CPU, le `NewProgramPasswordPresent` doit être vrai et sa valeur définie par `SetProgramPassword`
- `Restore` doit être pris en charge pour effectuer l'action (`RestoreAllowed`)

Dans l'exemple suivant, on recherche une CPU à une adresse IP donnée dans `IProfinetDeviceCollection`. Une fois trouvé, le programme vérifie que la CPU protégée par mot de passe prend en charge la fonction de restauration et appelle la méthode `Restore` sur un fichier de sauvegarde sans mot de passe CPU.

```
Network myNetwork = new Network();
uint targetIPAddress = 0xC0A80001; // 192.168.0.1
string bkFile = @"C:\MyCPUBackupFile.s7pbkp";
IProfinetDeviceCollection devices;

Result retVal = myNetwork.ScanNetworkDevices(out devices);
if (retVal.Succeeded)
{
    IProfinetDevice dev = devices.FindDeviceByIP(targetIPAddress);
    if (dev != null)
    {
        ICPU devAsCpu = dev as ICPU;
        if (devAsCpu != null && devAsCpu.RestoreAllowed)
        {
            retVal = devAsCpu.SetPassword(new EncryptedString("Password"));
            retVal = devAsCpu.SetBackupFile(bkFile);
            devAsCpu.Selected = true;
            if (devAsCpu.Failsafe)
                devAsCpu.SelectedConfirmed = true;

            retVal = devAsCpu.Restore();
        }
    }
}
```

7.12.4.9 Méthode Backup (interface ICPU)

Type de retour	Nom de la méthode
Result	Backup

Paramètres			
Nom	Type de données	Type de paramètre	Description
strFile	string	In	Chemin d'accès complet et nom sous lequel le fichier de sauvegarde doit être stocké.

Cette méthode est utilisée pour sauvegarder les données dans une CPU. Certaines CPU ne prennent pas en charge la fonction Sauvegarde / Restauration. Vous pouvez vérifier la propriété `BackupAllowed` pour vous assurer que la CPU concernée prend en charge cette fonction.

Dans l'exemple suivant, on recherche une CPU à une adresse IP donnée dans `IProfinetDeviceCollection`. Une fois trouvée, le programme vérifie que la CPU prend en charge la fonction de sauvegarde et appelle la méthode `Backup`.

```
Network myNetwork = new Network();
uint targetIPAddress = 0xC0A80001; // 192.168.0.1
string bkFile = @"C:\MyCPUBackupFile.s7pbkp";
IProfinetDeviceCollection devices;

Result retVal = myNetwork.ScanNetworkDevices(out devices);
if (retVal.Succeeded)
{
    foreach (IProfinetDevice dev in devices)
    {
        ICPU devAsCpu = dev as ICPU;

        if (devAsCpu != null && devAsCpu.IP == targetIPAddress &&
devAsCpu.BackupAllowed)
        {
            devAsCpu.Selected = true;
            retVal = devAsCpu.Backup(bkFile);
        }
    }
}
```

7.12.4.10 Méthode DownloadRecipe

Type de retour	Nom de la méthode
Result	DownloadRecipe

Paramètres			
Nom	Type de données	Type de paramètre	Description
strFile	string	In	Chemin d'accès complet et nom du fichier de recette à télécharger (transfert du PG/PC vers la carte mémoire de la CPU).

Cette méthode est utilisée pour ajouter ou remplacer un fichier CSV sur la carte mémoire de la CPU. Certaines CPU ne prennent pas en charge l'accès distant aux recettes. Vous pouvez vérifier la propriété `RemoteRecipesAllowed` pour vous assurer que la CPU concernée prend en charge cette fonction. L'exemple de code suivant montre l'écriture d'un fichier recette sur une carte mémoire de la CPU.

```

Network myNetwork = new Network();
uint targetIPAddress = 0xC0A80001; // 192.168.0.1
string rcpFile = @"C:\NewRecipe.csv";
IProfinetDeviceCollection devices;

Result retVal = myNetwork.ScanNetworkDevices(out devices);
if (retVal.Succeeded)
{
    foreach (IProfinetDevice dev in devices)
    {
        ICPU devAsCpu = dev as ICPU;
        if ((devAsCpu != null) &&
            (devAsCpu.IP == targetIPAddress) &&
            (devAsCpu.RemoteRecipesAllowed))
        {
            retVal = devAsCpu.SetPassword(new EncryptedString("Password"));
            IRemoteFolder recipes = devAsCpu.RecipeFolder;
            recipes.Selected = true;
            recipes.SetRemoteFile(rcpFile);
            retVal = devAsCpu.DownloadRecipe(rcpFile);
        }
    }
}

```

Remarque

Si une recette portant le même nom existe déjà sur la carte mémoire, elle est remplacée.

7.12.4.11 Méthode DeleteDataLog

Type de retour	Nom de la méthode
Result	DeleteDataLog

Paramètres			
Nom	Type de données	Type de paramètre	Description
strFileName	string	In	Nom du fichier Data log à supprimer de la carte mémoire d'une CPU.

Cette méthode est utilisée pour supprimer un fichier Data log de la carte mémoire de la CPU.

Certaines CPU ne prennent pas en charge l'accès distant aux Data logs. Vérifiez la propriété `RemoteDataLogsAllowed` pour vous assurer que la CPU concernée prend en charge cette fonction.

L'exemple de code suivant utilise la propriété `DataLogFolder` pour itérer tous les Data logs sur la carte mémoire de la CPU. Tous les Data logs sont supprimés.

```
Network myNetwork = new Network();
IProfinetDeviceCollection devices;

Result retVal = myNetwork.ScanNetworkDevices(out devices);
if (retVal.Succeeded)
{
    foreach (IProfinetDevice dev in devices)
    {
        ICPU devAsCpu = dev as ICPU;
        if (devAsCpu != null)
        {
            if (devAsCpu.RemoteDataLogsAllowed)
            {
                //-----
                // Vérifier d'abord que la carte mémoire contient des Data logs
                //-----
                if (devAsCpu.DataLogFolder.Exists)
                {
                    devAsCpu.SetPassword(new EncryptedString("Password"));
                    //-----
                    // Rechercher tous les fichiers Data log
                    //-----
                    for (Int32 i=devAsCpu.DataLogFolder.Files.Count - 1; i>=0; i--)
                    {
                        IRemoteFile datalog = devAsCpu.DataLogFolder.Files[i];
                        if (datalog != null)
                        {
                            datalog.Selected = true;
                            //-----
                            // Supprimer le Data log.
                            //-----
                            devAsCpu.DeleteDataLog(datalog.Name);
                        }
                    }
                }
            }
        }
    }
}
```

7.12.4.12 Méthode DeleteRecipe

Type de retour	Nom de la méthode
Result	DeleteRecipe

Paramètres			
Nom	Type de données	Type de paramètre	Description
strFileName	string	In	Nom du fichier recette à supprimer de la carte mémoire.

Cette méthode est utilisée pour supprimer un fichier de recettes de la carte mémoire de la CPU.

Certaines CPU ne prennent pas en charge l'accès distant aux recettes. Vérifiez la propriété `RemoteRecipesAllowed` pour vous assurer que la CPU concernée prend en charge cette fonction.

L'exemple de code suivant utilise la propriété `RecipeFolder` pour itérer toutes les recettes sur la carte mémoire de la CPU. Toutes les recettes sont supprimées.

```
Network myNetwork = new Network();
IProfinetDeviceCollection devices;

Result retVal = myNetwork.ScanNetworkDevices(out devices);
if (retVal.Succeeded)
{
    foreach (IProfinetDevice dev in devices)
    {
        ICPU devAsCpu = dev as ICPU;
        if (devAsCpu != null)
        {
            if (devAsCpu.RemoteDataLogsAllowed)
            {
                //-----
                // Vérifier d'abord que la carte mémoire contient des recettes
                //-----

                if (devAsCpu.RecipeFolder.Exists)
                {
                    devAsCpu.SetPassword(new EncryptedString("Password"));
                    //-----
                    // Rechercher tous les fichiers Data log
                    //-----
                    for (Int32 i=devAsCpu.RecipeFolder.Files.Count - 1; i>=0; i--)
                    {
                        IRemoteFile recipe = devAsCpu.RecipeFolder.Files[i];
                        if (recipe != null)
                        {
                            recipe.Selected = true;
                            //-----
                            // Supprimer la recette.
                            //-----
                            devAsCpu.DeleteRecipe(recipe.Name);
                        }
                    }
                }
            }
        }
    }
}
```

7.12.4.13 Méthode GetCurrentDateTime

Type de retour	Nom de la méthode
Result	GetCurrentDateTime

Paramètres			
Nom	Type de données	Type de paramètre	Description
DateTime	System.DateTime	Out	Date et heure en cours sur la CPU

Cette méthode récupère l'horodatage de la CPU.

Dans l'exemple suivant, on recherche une CPU à une adresse IP donnée dans l'IProfinetDeviceCollection et on récupère l'heure et la date actuels.

```
Network myNetwork = new Network();
IProfinetDeviceCollection devices;

uint targetIPAddress = 0xC0A80001; // 192.168.0.1

Result retVal = myNetwork.ScanNetworkDevices(out devices);
if (!retVal.Succeeded)
    return;
foreach (IProfinetDevice dev in devices)
{
    ICPU devAsCpu = dev as ICPU;
    if ((devAsCpu != null) && (devAsCpu.IP == targetIPAddress))
    {
        devAsCpu.SetPassword(new EncryptedString("Password"));
        devAsCpu.Selected = true;

        DateTime curTime = new DateTime();

        retVal = devAsCpu.GetCurrentDateTime(out curTime);
    }
}
```

7.12.4.14 Méthode GetDiagnosticsBuffer

Type de retour	Nom de la méthode
Result	GetDiagnosticsBuffer

Paramètres			
Nom	Type de données	Type de paramètre	Description
Diagnos- ticsItems	List<DiagnosticsIte m>	Out	Une collection d'éléments de diagnostic : Chaque élément de la collection représente une entrée dans le tampon de diagnostic.

Cette méthode lit les entrées de diagnostic actuelles provenant de la CPU. Chaque entrée est représentée sous forme de `DiagnosticsItem`. Cette classe est décrite après l'exemple de code. Language enum est décrit dans la section Énumérations API. Dans l'exemple suivant, on recherche une CPU à une adresse IP donnée dans l'`IProfinetDeviceCollection`. Une fois trouvées, les informations de diagnostic sont lues à partir de la CPU.

```
Network myNetwork = new Network();
uint targetIPAddress = 0xC0A80001; // 192.168.0.1
List<DiagnosticsItem> aLogs = new List<DiagnosticsItem>();
IProfinetDeviceCollection devices;

Result retVal = myNetwork.ScanNetworkDevices(out devices);
if (retVal.Succeeded)
{
    foreach (IProfinetDevice dev in devices)
    {
        ICPU devAsCpu = dev as ICPU;
        if ((devAsCpu != null) && (devAsCpu.IP == targetIPAddress))
        {
            devAsCpu.SetPassword(new EncryptedString("Password"));
            devAsCpu.Selected = true;
            retVal = devAsCpu.GetDiagnosticsBuffer(out aLogs, Language.English);
            if (retVal.Succeeded)
            {
                for (int idxLog = 0; idxLog < aLogs.Count; idxLog++)
                {
                    string descr = aLogs[idxLog].Description1;
                }
            }
        }
    }
}
```

La classe DiagnosticsItem

Remarque

Modifiez le second paramètre dans la méthode `GetDiagnosticsBuffer`, pour obtenir les chaînes de caractères dans une autre langue prise en charge.

La méthode `GetDiagnosticsBuffer` renvoie une collection d'objets `DiagnosticsItem`.

Cette classe définit les éléments suivants :

Nom d'élément	Type de données	Description
TimeStamp	System.DateTime	Date et heure d'enregistrement de l'événement de diagnostic.
State	Byte	Message entrant ou sortant
Description1	String	Titre
Description2	String	Détails

7.12.4.15 Méthode MemoryReset

Type de retour	Nom de la méthode
Result	MemoryReset

Cette méthode réinitialise la mémoire sur la CPU

Dans l'exemple suivant, on recherche une CPU à une adresse IP donnée dans `IProfinetDeviceCollection` et on appelle `MemoryReset` pour cette CPU.

```
Network myNetwork = new Network();
uint targetIPAddress = 0xC0A80001; // 192.168.0.1
IProfinetDeviceCollection devices;

Result retVal = myNetwork.ScanNetworkDevices(out devices);
if (retVal.Succeeded)
{
    IProfinetDevice dev = devices.FindDeviceByIP(targetIPAddress);
    if (dev != null)
    {
        ICPU devAsCpu = dev as ICPU;
        if (devAsCpu != null)
        {
            devAsCpu.SetPassword(new EncryptedString("Password"));
            devAsCpu.Selected = true;
            retVal = devAsCpu.MemoryReset();
        }
    }
}
```

7.12.4.16 Méthode ResetToFactoryDefaults

Type de retour	Nom de la méthode
Result	ResetToFactoryDefaults

Paramètres			
Nom	Type de données	Type de paramètre	Description
password	EncryptedString	In	Cette méthode établit une connexion légitimée à l'appareil. De ce fait, un mot de passe peut être requis.

Cette méthode restaure les paramètres d'usine d'une CPU.

Dans l'exemple suivant, on recherche une CPU à une adresse IP donnée dans l'IProfinetDeviceCollection et on appelle la méthode ResetToFactoryDefaults. Vous définir le drapeau SelectedConfirmed sur TRUE pour les appareils de sécurité.

```
Network myNetwork = new Network();
ulong targetMACAddress = 0x112233445566; // 11:22:33:44:55:66
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (retVal.Succeeded)
{
    ICPU cpu = scannedDevices.FindDeviceByMAC(targetMACAddress) as ICPU;
    if (cpu != null && cpu.ResetToFactoryAllowed == true)
    {
        // Sélectionner la CPU à mettre à jour
        cpu.Selected = true;

        // Adresse IP unique ?
        if (cpu.DuplicateIP == true)
            return;

        // L'appareil est-il pris en charge ?
        if (cpu.Supported == false)
            return;

        // L'appareil est-il initialisé ?
        if (cpu.Initialized == false)
            return;

        // L'appareil est-il de sécurité ?
        if (cpu.Failsafe == true) { ConfirmationType type =
cpu.DetermineConfirmationMessage(FailsafeOperation.ResetToFactoryOperation);
        // Vérifier le type et confirmer
        cpu.SelectedConfirmed = true;

        // Vérifier que la mise à jour est faisable
    }
}
```

```
        if (cpu.HasSafetyProgram == true || cpu.Protected == true ||
cpu.NewRestoreNameIsSafety == true)
    {
        // L'appareil est-il protégé par mot de passe ?
        if (cpu.Protected == true)
        {
            // Un mot de passe valide a-t-il été fourni ?
            if (cpu.PasswordValid == false)
                return;

            // Sommes-nous légitimés pour le niveau de sécurité ?
            bool bSufficientAccess = cpu.PasswordProtectionLevel ==
ProtectionLevel.Failsafe;
            if (bSufficientAccess == false)
                return;
        }
    }
else
{
    // Un programme de sécurité est-il sur le point d'être chargé ?
    if (cpu.NewRestoreNameIsSafety == true)
        return;
}

// L'appareil est-il protégé par mot de passe ?
if (cpu.Protected == true)
{
    // Un mot de passe valide a-t-il été fourni ?
    if (cpu.PasswordValid == false)
        return;

    // Avons-nous un niveau de légitimation suffisant pour que l'opération
réussisse ?
    bool bSufficientAccess = cpu.PasswordProtectionLevel ==
ProtectionLevel.Failsafe || cpu.PasswordProtectionLevel == ProtectionLevel.Full;
    if (bSufficientAccess == false)
        return;
}

// Effectuer une restauration aux valeurs usine par défaut
retVal = cpu.ResetToFactoryDefaults();

// Réinitialiser
cpu.SelectedConfirmed = false;
}
```

7.12.4.17 Méthode SetOperatingState

Type de retour	Nom de la méthode
Result	SetOperatingState

Paramètres			
Nom	Type de données	Type de paramètre	Description
nRequestState	OperatingStateREQ	In	Le nouvel état de fonctionnement

Cette méthode est utilisée pour changer l'état de fonctionnement d'une CPU.

Certaines CPU ne prennent pas cette fonction en charge. Vérifiez la propriété `ChangeModeAllowed` pour vous assurer que la CPU concernée prend en charge cette fonction.

Dans l'exemple suivant, on recherche une CPU à une adresse IP donnée dans `IProfinetDeviceCollection`. Une fois trouvé, le programme vérifie que la CPU prend en charge la fonction de changement de mode et passe la CPU en RUN.

```

Network myNetwork = new Network();
uint targetIPAddress = 0xC0A80001; // 192.168.0.1
IProfinetDeviceCollection devices;

Result retVal = myNetwork.ScanNetworkDevices(out devices);
if (retVal.Succeeded)
{
    IProfinetDevice dev = devices.FindDeviceByIP(targetIPAddress);
    if (dev != null)
    {
        ICPU devAsCpu = dev as ICPU;
        if ((devAsCpu != null) && (devAsCpu.ChangeModeAllowed))
        {
            devAsCpu.SetPassword(new EncryptedString("Password"));
            devAsCpu.Selected = true;
            retVal = devAsCpu.SetOperatingState(OperatingStateREQ.Run);
        }
    }
}

```

7.12.4.18 Méthode SetCurrentDateTime

Type de retour	Nom de la méthode
Result	SetCurrentDateTime

Paramètres			
Nom	Type de données	Type de paramètre	Description
password	EncryptedString	In	Cette méthode établit une connexion légitimée à l'appareil. De ce fait, un mot de passe peut être requis.
time	System.DateTime	In	Nouvelle valeur pour l'heure actuelle de la CPU.

Cette méthode règle l'heure actuelle de la CPU. Les règles de changement de l'heure configurée ne sont pas affectées par cette action. De ce fait, la valeur `DateTime` spécifiée est basée sur l'heure UTC, pas sur l'heure locale.

L'exemple suivant concerne l'ensemble du réseau industriel et règle l'heure actuelle de chaque CPU sur l'heure du PG/PC.

```
Network myNetwork = new Network();
IProfinetDeviceCollection devices;

Result retVal = myNetwork.ScanNetworkDevices(out devices);
if (retVal.Succeeded)
{
    foreach (IProfinetDevice dev in devices)
    {
        ICPU devAsCpu = dev as ICPU;
        if (devAsCpu != null)
        {
            devAsCpu.SetPassword(new EncryptedString("Password"));
            devAsCpu.Selected = true;
            retVal = devAsCpu.SetCurrentDateTime(DateTime.UtcNow);
        }
    }
}
```

7.12.4.19 Méthode UploadDataLog

Type de retour	Nom de la méthode
Result	UploadDataLog

Paramètres			
Nom	Type de données	Type de paramètre	Description
strFileName	string	In	Le nom de fichier du Data log à charger depuis la carte mémoire SIMATIC amovible d'une CPU.
strDestinationFolder	string	In	Chemin d'accès complet sous lequel est stocké le fichier Data log chargé.

Cette méthode charge une copie du fichier Data log spécifié de la carte mémoire de la CPU vers votre PG/PC. Certaines CPU ne prennent pas en charge l'accès distant aux Data logs. Vérifiez la propriété `RemoteDataLogsAllowed` pour vous assurer que la CPU concernée prend en charge cette fonction.

L'exemple de code suivant utilise la propriété `DataLogFolder` pour itérer tous les Data logs sur la carte mémoire de la CPU. Une copie de chaque Data log est chargée dans le dossier C:\MyDataLogs.

```
Network myNetwork = new Network();
IProfinetDeviceCollection devices;

Result retVal = myNetwork.ScanNetworkDevices(out devices);
if (retVal.Succeeded)
{
    foreach (IProfinetDevice dev in devices)
    {
        ICPU devAsCpu = dev as ICPU;
        if (devAsCpu != null)
        {
            devAsCpu.SetPassword(new EncryptedString("Password"));

            if (devAsCpu.RemoteDataLogsAllowed)
            {
                //-----
                // Vérifier d'abord que la carte mémoire contient des Data logs
                //-----
                if (devAsCpu.DataLogFolder.Exists)
                {
                    //-----
                    // Rechercher tous les fichiers Data log
                    //-----
                    foreach (IRemoteFile datalog in devAsCpu.DataLogFolder.Files)
                    {
                        datalog.Selected = true;
                        //-----
                        // Charger une copie de chaque data log.
                        //-----
                        devAsCpu.UploadDataLog(datalog.Name, @"C:\MyDataLogs");
                    }
                }
            }
        }
    }
}
```

7.12.4.20 Méthode UploadRecipe

Type de retour	Nom de la méthode
Result	UploadRecipe

Paramètres			
Nom	Type de données	Type de paramètre	Description
strFileName	string	In	Le nom de fichier de la recette à charger depuis la carte mémoire de la CPU.
strDestinationFolder	string	In	Chemin d'accès complet sous lequel est écrit le fichier Recette chargé

Cette méthode charge une copie du fichier recette de la carte mémoire de la CPU. Certaines CPU ne prennent pas en charge l'accès distant aux recettes. Vérifiez la propriété `RemoteRecipesAllowed` pour vous assurer que la CPU concernée prend en charge cette fonction.

L'exemple de code suivant utilise la propriété `RecipeFolder` pour itérer toutes les recettes sur la carte mémoire de la CPU. Une copie de chaque recette est chargée dans le dossier C:\MyRecipes.

```
Network myNetwork = new Network();
IProfinetDeviceCollection devices;

Result retVal = myNetwork.ScanNetworkDevices(out devices);
if (retVal.Succeeded)
{
    foreach (IProfinetDevice dev in devices)
    {
        ICPU devAsCpu = dev as ICPU; if (devAsCpu != null)
        {
            devAsCpu.SetPassword(new EncryptedString("Password"));
            if (devAsCpu.RemoteRecipesAllowed)
            {
                //-----
                // Vérifier d'abord que la carte mémoire contient des recettes.
                //-----
                if (devAsCpu.RecipeFolder.Exists)
                {
                    //-----
                    // Rechercher tous les fichiers recette
                    //-----
                    foreach (IRemoteFile recipe in devAsCpu.RecipeFolder.Files)
                    {
                        recipe.Selected = true;
                        //-----
                        // Charger une copie de chaque recette.
                        //-----
                        devAsCpu.UploadRecipe(recipe.Name, @"C:\MyRecipes");
                    }
                }
            }
        }
    }
}
```

7.12.4.21 Méthode UploadServiceData

Type de retour	Nom de la méthode
Result	UploadServiceData

Paramètres			
Nom	Type de données	Type de paramètre	Description
strPath	string	In	Chemin d'accès complet au dossier regroupant le contenu de la carte programme.

Cette méthode peut charger les données de maintenance d'une CPU défectueuse.

Dans l'exemple suivant, on recherche une CPU à une adresse IP donnée dans l'`IProfinetDeviceCollection`. L'`OperatingState` actuel de la CPU est vérifié. Si la CPU est défectueuse, les données de maintenance sont chargées.

```
Network myNetwork = new Network();
IProfinetDeviceCollection devices;

uint targetIPAddress = 0xC0A80001; // 192.168.0.1
string strDiagFolder = @"c:\Diagnostics";

Result retVal = myNetwork.ScanNetworkDevices(out devices);
if (retVal.Succeeded)
{
    IProfinetDevice dev = devices.FindDeviceByIP(targetIPAddress);
    if (dev != null)
    {
        ICPU devAsCpu = dev as ICPU;

        if (devAsCpu != null)
        {
            devAsCpu.SetPassword(new EncryptedString("Password"));
            devAsCpu.Selected = true;
            if (devAsCpu.OperatingMode == OperatingState.Defective)
            {
                retVal = devAsCpu.UploadServiceData(strDiagFolder);
            }
        }
    }
}
```

7.12.4.22 Méthode FormatMemoryCard

Type de retour	Nom de la méthode
Result	FormatMemoryCard

Cette méthode est utilisée pour formater la carte mémoire amovible SIMATIC enfichée dans une CPU.

Dans l'exemple suivant, on recherche une CPU à une adresse IP donnée dans l'IProfinetDeviceCollection. La carte mémoire de l'appareil est alors formatée. Vous définir le drapeau SelectedConfirmed sur TRUE pour les appareils de sécurité.

```
Network myNetwork = new Network();
ulong targetMACAddress = 0x112233445566; // 11:22:33:44:55:66
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (retVal.Succeeded)
{
    ICPU cpu = scannedDevices.FindDeviceByMAC(targetMACAddress) as ICPU;
    if (cpu != null && cpu.FormatMCAllowed == true)
    {
        // Sélectionner la CPU à mettre à jour
        cpu.Selected = true;

        // Adresse IP unique ?
        if (cpu.DuplicateIP == true)
            return;

        // L'appareil est-il pris en charge ?
        if (cpu.Supported == false)
            return;

        // L'appareil est-il initialisé ?
        if (cpu.Initialized == false)
            return;

        // L'appareil est-il de sécurité ?
        if (cpu.Failsafe == true)
        {
            ConfirmationType type =
cpu.DetermineConfirmationMessage(FailsafeOperation.ResetToFactoryOperation);

            // Vérifier le type et confirmer
            cpu.SelectedConfirmed = true;

            // Vérifier que la mise à jour est faisable
            if (cpu.HasSafetyProgram == true || cpu.Protected == true ||
cpu.NewRestoreNameIsSafety == true)
            {
                // Is the device password protected?if (cpu.Protected == true)
                {

```

```
// Un mot de passe valide a-t-il été fourni ?
if (cpu.PasswordValid == false)
    return;

// Sommes-nous légitimés pour le niveau de sécurité ?
bool bSufficientAccess = cpu.PasswordProtectionLevel ==
ProtectionLevel.Failsafe;
if (bSufficientAccess == false)
    return;
}

}

else
{
    // Un programme de sécurité est-il sur le point d'être chargé ?
    if (cpu.NewRestoreNameIsSafety == true)
        return;
}

// L'appareil est-il protégé par mot de passe ?
if (cpu.Protected == true)
{
    // Un mot de passe valide a-t-il été fourni ?
    if (cpu.PasswordValid == false)
        return;

    // Avons-nous un niveau de légitimation suffisant pour que l'opération
réussisse ?
    bool bSufficientAccess = cpu.PasswordProtectionLevel ==
ProtectionLevel.Failsafe || cpu.PasswordProtectionLevel == ProtectionLevel.Full;
    if (bSufficientAccess == false)
        return;
}

// Effectuer une restauration
RetVal = cpu.FormatMemoryCard();

// Réinitialiser
cpu.SelectedConfirmed = false;
}
}
```

7.12.4.23 DetermineConfirmationMessage

Type de retour	Nom de la méthode
ConfirmationType	DetermineConfirmationMessage

Paramètres			
Nom	Type de données	Type de paramètre	Description
operation	FailsafeOperation	In	Opération évaluée.

Cette méthode détermine le contenu du message de confirmation lorsque l'utilisateur tente d'effectuer une opération relative à la sécurité.

La classe ConfirmationType comprend une série de codes Hamming qui représentent chacun un message d'erreur affiché comme suit pour l'utilisateur :

ConfirmationType	Message à afficher
SafetyPasswordIsBeingUsed	Une opération sur un programme standard est sur le point d'être initiée à l'aide du mot de passe de sécurité de la CPU.
DeletingExistingSafetyProgram	Un programme de sécurité existant est sur le point d'être supprimé.
ReplacingExistingSafetyProgram	Un programme de sécurité existant est sur le point d'être actualisé avec un autre programme de sécurité.
ReplacingExistingSafetyProgramWithNonSafetyProgram	Un programme de sécurité existant est sur le point d'être remplacé par un programme standard.
LoadingSafetyProgram	Un programme de sécurité est sur le point d'être chargé pour la première fois.

Dans l'exemple suivant, on recherche une CPU à une adresse IP donnée dans l'IProfinetDeviceCollection. Cette méthode est utilisée pour afficher un message concernant un état relevant de la sécurité dans une boîte de dialogue de confirmation utilisateur avant l'exécution d'une opération relative à la sécurité.

```
Network myNetwork = new Network();
uint targetIPAddress = 0xC0A80001; // 192.168.0.1
IProfinetDeviceCollection devices;

Result retVal = myNetwork.ScanNetworkDevices(out devices);
if (retVal.Succeeded)
{
    IProfinetDevice dev = devices.FindDeviceByIP(targetIPAddress);
    if (dev != null)
    {
        ICPU devAsCpu = dev as ICPU;
        if (devAsCpu != null)
        {
            devAsCpu.SetPassword(new EncryptedString("Password"));
            devAsCpu.Selected = true;
            if (devAsCpu.Failsafe)
            {
                devAsCpu.SelectedConfirmed = true;
            }
            ConfirmationType confirm =
            devAsCpu.DetermineConfirmationMessage(FailsafeOperation.FormatMCOOperation);

            if (confirm == ConfirmationType.DeletingExistingSafetyProgram)
                devAsCpu.FormatMemoryCard();
        }
    }
}
```

7.12.5 Propriétés RemoteInterfaces

7.12.5.1 Modules d'E/S décentralisés

Chaque CPU peut prendre en charge plusieurs interfaces E/S décentralisées. Les informations relatives aux appareils raccordés à ces interfaces distantes sont accessibles via la propriété `RemoteInterfaces`.

L'exemple suivant montre comment accéder à ces informations pour toutes les CPU d'un réseau.

```
Network myNetwork = new Network();
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (retVal.Succeeded)
{
    foreach (IProfinetDevice dev in scannedDevices)
    {
        ICPU devAsCpu = dev as ICPU;

        List<IRemoteInterface> decentalNets = devAsCpu.RemoteInterfaces;
        foreach (IRemoteInterface net in decentalNets)
        {
            //-----
            // Inspecter l'interface distante
            //-----
        }
    }
}
```

7.12.5.2 Propriétés IRemoteInterfaces

L'interface `IRemoteInterface` prend en charge les propriétés suivantes. Ces propriétés sont en lecture seule.

Nom de la propriété	Type de retour	Description
Devices	List<IBaseDevice>	Une liste de stations E/S décentralisées raccordées à cette interface distante
InterfaceType	RemoteInterfaceType	Le protocole de communication pour cette interface distante Voir aussi Énumération <code>RemoteInterfaceType</code> (Page 248)
Name	string	Nom configuré pour cette interface distante.

La propriété `Devices` peut être utilisée au travers d'un réseau décentralisé. Chaque appareil du réseau décentralisé est représenté par une interface `IBaseDevice`. Cette interface possède un sous-groupe de propriétés disponibles pour un `IProfinetDevice` et fournit la fonctionnalité limitée disponible pour ces appareils dans l'API SIMATIC Automation Tool.

Les propriétés suivantes sont disponibles dans l'interface `IBaseDevice`.

Nom de la propriété	Type de retour	Description
<code>ArticleNumber</code>	<code>string</code>	Numéro de référence du module. Également appelé MLFB ou numéro d'article.
<code>Comment</code>	<code>string</code>	Ceci permet à l'utilisateur de formuler un commentaire sur l'appareil. Cette propriété est utilisée dans l'interface utilisateur de SIMATIC Automation Tool. Elle ne concerne pas les opérations API.
<code>Configured</code>	<code>bool</code>	L'appareil possède-t-il une configuration valide ?
<code>Description</code>	<code>string</code>	Une description du matériel, sur la base du numéro d'article. Il s'agit de la même description que dans TIA Portal. (par ex. "CPU-1215 DC/DC/DC")
<code>Failsafe</code>	<code>FeatureSupport</code>	Sur la base de ce numéro d'article, s'agit-il d'un appareil de sécurité ?
<code>Family</code>	<code>DeviceFamily</code>	De quelle gamme est cet appareil ? Pour plus d'informations, référez-vous à la description de l'énumération de la <code>DeviceFamily</code> (Page 242).
<code>FirmwareUpdateAllowed</code>	<code>FeatureSupport</code>	Cet appareil prend-il en charge la mise à jour du firmware ?
<code>FirmwareVersion</code>	<code>string</code>	La versions de firmware actuelle de l'appareil
<code>HardwareInFirmwareOrder</code>	<code>IHardwareCollection</code>	Collection matérielle dans la commande de firmware
<code>HardwareInDisplayOrder</code>	<code>IHardwareCollection</code>	Matériel dans la commande affichée
<code>HardwareNumber</code>	<code>short</code>	Identifiant numérique
<code>ID</code>	<code>uint</code>	Un identifiant unique pour chaque appareil dans la station. Il est utilisé comme identificateur unique pour la mise à jour du firmware.
<code>Modules</code>	<code>IModuleCollection</code>	Une collection de modules locaux raccordés à la station. Cette propriété est décrite ici (Page 185) en détail.

Nom de la propriété	Type de retour	Description
Name	string	Le nom de l'appareil.
NewFirmwareFile	string	Chemin vers le nouveau fichier de firmware
NewFirmwareVersion	string	Cette propriété est utilisée dans l'interface utilisateur de SIMATIC Automation Tool. Elle ne concerne pas les opérations API.
NewFirmwareNameIsValid	bool	Le nouveau fichier de firmware est-il valide ?
Selected	bool	L'appareil est-il sélectionné ?
SerialNumber	string	Le numéro de série univoque de l'appareil.
Slot	uint	Le numéro d'emplacement du matériel.
SlotName	string	Cette propriété est utilisée dans l'interface utilisateur de SIMATIC Automation Tool. Elle ne concerne pas les opérations API.
StationNumber	uint	Le numéro de station de l'appareil.
SubSlot	uint	Le sous-emplacement de l'appareil. Ceci vaut pour les sous-modules enfichables tels que les SB-1200.
Supported	FeatureSupport	L'appareil réseau détecté est-il pris en charge par les opérations actuelles de l'API SIMATIC Automation Tool ?

En utilisant la propriété Devices de l'`IRemoteInterface`, il est possible d'inspecter toutes les stations sur le réseau décentralisé.

Pour élargir l'exemple précédent :

```
Network myNetwork = new Network();
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (!retVal.Succeeded)
    return;

foreach (IProfinetDevice dev in scannedDevices)
{
    ICPU devAsCpu = dev as ICPU;
    if (devAsCpu == null)
        continue;

    List<IRemoteInterface> decentalNets = devAsCpu.RemoteInterfaces;
    List<string> orderNumbers = new List<string>();

    foreach (IRemoteInterface net in decentalNets)
    {
        //-----
        // Inspecter l'interface distante
        //-----
        if (net.InterfaceType == RemoteInterfaceType.Profinet)
        {
            //-----
            // Consulter chaque station décentralisée
            //-----
            List<IBaseDevice> stations = net.Devices;

            foreach (IBaseDevice station in stations)
            {
                orderNumbers.Add(station.ArticleNumber);
            }
        }
    }
}
```

L'exemple suivant concerne toutes les interfaces PROFINET distantes et crée une liste des numéros d'article de toutes les stations décentralisées sur le réseau industriel.

Comme `IBaseDevice` prend également en charge la propriété `Modules`, il est aisément d'élargir encore l'exemple pour consulter non seulement les stations décentralisées, mais aussi tous les modules enfichés localement sur chaque station.

7.13 Interface IHMI

7.13.1 Interface IHMI

La méthode `ScanNetworkDevices` est appelée pour générer une `IProfinetDeviceCollection`. Cette collection contient un élément pour chaque appareil accessible sur le réseau industriel. Ces appareils peuvent être des CPU, des appareils IHM, des stations I/O décentralisées et d'autres appareils Siemens. L'interface `IProfinetDevice` fournit les propriétés et les méthodes qui s'appliquent à toutes les catégories d'appareils.

Cependant, certaines méthodes sont utilisées uniquement pour les appareils HMI. Ces propriétés et méthodes sont accessibles à l'aide de l'interface `IHM`.

Pour déterminer si une interface `IProfinetDevice` donnée représente effectivement un appareil IHM, placez-la dans un `IHM`. Si l'opération réussit, l'appareil est bien un appareil IHM, et vous pouvez utiliser les méthodes sur l'interface `IHM`. L'exemple suivant illustre la procédure.

```
Network myNetwork = new Network();
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (retVal.Succeeded)
{
    foreach (IProfinetDevice dev in scannedDevices)
    {
        IHMI devAsHmi = dev as IHMI;
        if (devAsHmi != null)
        {
            //-----
            // L'appareil est un appareil IHM.
            // L'interface IHM peut interagir avec cet appareil.
            //-----
        }
    }
}
```

Remarque

L'interface `IHM` hérite de l'`IProfinetDevice`. De ce fait, les propriétés et méthodes prises en charge sur le `IProfinetDevice` le sont aussi sur l'interface `IHM`. La rubrique sur les propriétés IHM suivante décrit seulement les propriétés/méthodes qui appartiennent uniquement à l'interface `IHM`.

Remarque

L'interface `IHMI` prend en charge la méthode `FirmwareUpdate`. Cependant, cette méthode renvoie toujours l'erreur `FirmwareUpdateNotSupported`. Pour mettre à jour le firmware pour un appareil `HMI`, il est nécessaire d'exécuter la méthode de mise à jour du programme (Page 234).

7.13.2 Propriétés et drapeaux IHMI**7.13.2.1 Propriétés IHMI**

L'interface `IHMI` V3.1 a également reçu les modifications de l'API SIMATIC Automation Tool V3.0 où de nombreux paramètres sont devenus des propriétés de l'objet.

Nom de la propriété	ReturnType	Description
<code>DeviceType</code>	<code>string</code>	Retourne le type d'IHM que l'objet représente
<code>FirmwareDeviceVersion</code>	<code>string</code>	Retourne la version de firmware présente sur l'IHM
<code>RuntimeDeviceVersion</code>	<code>string</code>	Retourne la version de runtime présente sur l'IHM

7.13.2.2 Drapeaux relatifs aux mises à jour de programmes

Vous pouvez utiliser ces drapeaux avec l'interface IHM V3.1.

Nom de la propriété	Type de retour	Description
<code>NewProgramNameIsValid</code>	<code>bool</code>	Vrai, lorsque la méthode <code>SetProgramFolder</code> est appelée avec un dossier programme valide. Faux, si le programme n'est pas valide.
<code>ProgramUpdateSucceeded</code>	<code>bool</code>	Vrai, lorsque la mise à jour du programme a réussi, même si une erreur peut être retournée depuis l'état de l'actualisation interne
<code>NewProgramName</code>	<code>string</code>	Nom du nouveau programme
<code>NewProgramFolder</code>	<code>string</code>	Emplacement du dossier pour le nouveau programme : la valeur est définie via la méthode <code>SetProgramFolder</code>
<code>NewProgramNameErrorCode</code>	<code>Result</code>	Code permettant d'identifier les problèmes qui peuvent survenir lors de la validation du nouveau programme, par exemple si le programme est invalide pour l'appareil ou si l'adresse IP affectée dans le programme existe déjà sur le réseau.

7.13.2.3 Restaurer les drapeaux

Vous pouvez utiliser ces drapeaux avec l'interface IHM V3.1.

Nom de la propriété	Type de retour	Description
NewRestoreNameIsValid	bool	Vrai, lorsque la méthode SetBackupFolder est appelée avec un fichier de sauvegarde valide. Faux, si le fichier de sauvegarde n'est pas valide.
RestoreSucceeded	bool	Vrai lorsque la restauration a réussi, même si une erreur a pu être retournée depuis l'état de l'actualisation interne
NewRestoreName	string	Quel est le nom du nouveau programme ?
NewRestoreFile	string	Quel est l'emplacement du fichier pour le nouveau programme ? La valeur est définie à l'aide de la méthode SetBackupFile.
NewRestoreNameErrorCode	Result	Un moyen simple d'identifier les problèmes qui peuvent survenir lors de la validation du nouveau programme, par exemple si le programme est invalide ou incompatible avec l'appareil.

7.13.2.4 Drapeaux de fonctionnalités

Principe

Vous pouvez utiliser ces drapeaux avec l'interface IHM V3.1.

Nom de la propriété	Type de retour	Description
BackupAllowed	bool	Vrai, si l'appareil autorise des sauvegardes
ProgramUpdateAllowed	bool	Vrai, si l'appareil autorise des mises à jour du programme
RestoreAllowed	bool	Vrai, si l'appareil autorise des restaurations

7.13.3 Méthode Backup (interface IHMI)

Type de retour	Nom de la méthode
Result	Backup

Paramètres			
Nom	Type de données	Type de paramètre	Description
strFile	string	In	Chemin d'accès complet et nom sous lequel le fichier de sauvegarde est stocké

Cette méthode est utilisée pour sauvegarder les données d'une IHM. Certaines IHM ne prennent pas en charge la fonction Sauvegarde / Restauration. La propriété `BackupAllowed` peut être vérifiée pour s'assurer que l'IHM concernée prend en charge cette fonction. Dans l'exemple suivant, on recherche une IHM à une adresse IP donnée dans la `IProfinetDeviceCollection`. Une fois trouvée, le programme vérifie que l'IHM prend en charge la fonction de sauvegarde et appelle la méthode `Backup`.

```

Network myNetwork = new Network();
uint targetIPAddress = 0xC0A80001; // 192.168.0.1
string bkFile = @"C:\MyCPUBackupFile.s7pbkp";
IProfinetDeviceCollection devices;

Result retVal = myNetwork.ScanNetworkDevices(out devices);
if (retVal.Succeeded)
{
    foreach (IProfinetDevice dev in devices)
    {
        IHMI devAsHmi = dev as IHMI;

        if ((devAsHmi != null) &&
            (devAsHmi.IP == targetIPAddress) &&
            (devAsHmi.BackupAllowed))
        {
            devAsHmi.Selected = true;
            retVal = devAsHmi.Backup(bkFile);
        }
    }
}

```

7.13.4 Méthode ProgramUpdate (interface IHMI)

Type de retour	Nom de la méthode
Result	ProgramUpdate

Cette méthode met à jour le système d'exploitation et le logiciel runtime de l'appareil IHM. Le paramètre `strPath` attribue un dossier contenant le programme à charger.

Pour exécuter correctement la méthode `ProgramUpdate` sur l'interface IHM, vous devez vérifier les éléments suivants :

- L'appareil est sélectionné
- `NewProgramFolder` a été défini (effectué via `SetProgramFolder`)

Dans l'exemple suivant, on recherche une IHM à une adresse IP donnée dans la `IProfinetDeviceCollection` et le programme de cette IHM est mis à jour.

```
Network myNetwork = new Network();
uint targetIPAddress = 0xC0A80001; // 192.168.0.1
IProfinetDeviceCollection devices;

Result retVal = myNetwork.ScanNetworkDevices(out devices);
if (retVal.Succeeded)
{
    IProfinetDevice dev = devices.FindDeviceByIP(targetIPAddress);
    if (dev != null)
    {
        IHMI devAsHMI = dev as IHMI;
        if (devAsHMI != null)
        {
            devAsHMI.Selected = true;
            devAsHMI.SetProgramFolder(
                @"c:\myFolder\ProgramUpdate\Simatic.HMI\RT_Projects\Project1");
            retVal = devAsHMI.ProgramUpdate();
        }
    }
}
```

Le dossier programme doit contenir les fichiers suivants pour que l'opération se termine correctement :

DownloadTask.xml
ProjectCharacteristics.rdf

Ces fichiers se trouvent généralement dans un dossier créé (à l'aide de TIA Portal) au format suivant :

{DeviceName}\Simatic.HMI\RT_Projects\{ProjectName}.{DeviceName}

Par exemple :

"C:\Desktop\hmim14000100a\Simatic.HMI\RT_Projects\DasBasicUndMobilePanelen.hmim14000100a[KTP700 Mobile]"

Remarque

Mises à jour du système d'exploitation et du logiciel runtime d'appareils IHM

ProgramUpdate pour une HMI est différent de la méthode pour une CPU. Cette méthode permet de mettre à jour le firmware, le système d'exploitation et le logiciel d'exécution pour des appareils IHM. Vous ne pouvez pas choisir une mise à jour partielle. SIMATIC Automation Tool met à jour tous les composants de données si nécessaire afin d'assurer un téléchargement cohérent. Une carte de mise à jour du programme IHM peut comporter plusieurs projets nécessitant la saisie d'un dossier sous \Simatic.HMI\RT_Projects\ pour le téléchargement.

7.13.5 Méthode Restore (interface IHMI)

Type de retour	Nom de la méthode
Result	Restore

Utilisez cette méthode pour restaurer les données d'appareil IHM d'une sauvegarde précédente de l'appareil. Certaines IHM ne prennent pas en charge la fonction Sauvegarde / Restauration. La propriété `RestoreAllowed` peut être vérifiée pour s'assurer que l'IHM concernée prend en charge cette fonction.

Pour exécuter correctement la méthode `Restore` sur l'interface `IHMI`, vous devez vérifier les éléments suivants :

- L'appareil est sélectionné
- `BackupFile` a été défini (effectué via `SetBackupFile`)

Dans l'exemple suivant, on recherche une IHM à une adresse IP donnée dans la `IProfinetDeviceCollection`. Une fois trouvée, le programme vérifie que l'IHM prend en charge la fonction de restauration et appelle la méthode `Restore`.

```
Network myNetwork = new Network();
ulong targetMACAddress = 0x112233445566; // 11:22:33:44:55:66
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (retVal.Succeeded)
{
    IHMI hmi = scannedDevices.FindDeviceByMAC(targetMACAddress) as IHMI;
    if (hmi != null && hmi.RestoreAllowed == true)
    {
        // Sélectionner la CPU à mettre à jour
        hmi.Selected = true;

        retVal = hmi.SetBackupFile(@"C:\MyFolder\Backup.s7pbkp");
        if (retValFailed == true)
            return;

        // Adresse IP unique ?
        if (hmi.DuplicateIP == true)
            return;

        // L'appareil est-il pris en charge ?
        if (hmi.Supported == false)
            return;

        // Effectuer une restauration
        retVal = hmi.Restore();
    }
}
```

7.13.6 Méthode SetProgramFolder

Type de retour	Nom de la méthode
Result	SetProgramFolder

Paramètres			
Nom	Type de données	Type de paramètre	Description
strFolder	string	in	Définit l'emplacement du dossier où la source de téléchargement du programme est stockée.

Auparavant, le paramètre de chemin d'accès au dossier pour la mise à jour du programme était transmis à l'objet avec l'appel de méthode. SIMATIC Automation Tool V3.1 induit une nouveauté en permettant à l'utilisateur de définir l'emplacement du dossier.

L'exemple suivant montre comment définir le dossier programme sur un appareil IHM.

```

Network myNetwork = new Network();
ulong targetMACAddress = 0x112233445566; // 11:22:33:44:55:66
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (retVal.Succeeded)
{
    IHMI hmi = scannedDevices.FindDeviceByMAC(targetMACAddress) as IHMI;
    if (hmi != null && hmi.ProgramUpdateAllowed == true)
    {
        // Sélectionner la CPU à mettre à jour
        hmi.Selected = true;

        retVal = hmi.SetProgramFolder(@"C:\MyFolder");
        if (retValFailed == true)
            return;

        // Adresse IP unique ?
        if (hmi.DuplicateIP == true)
            return;

        // L'appareil est-il pris en charge ?
        if (hmi.Supported == false)
            return;

        // Effectuer une mise à jour du programme
        retVal = hmi.ProgramUpdate();
    }
}

```

Le dossier spécifié pour "strPath" doit contenir les fichiers suivants pour que l'opération se termine correctement :

- DownloadTask.xml
- ProjectCharacteristics.rdf

Ces fichiers se trouvent généralement dans un dossier créé (à l'aide de TIA Portal) au format suivant :

{DeviceName}\Simatic.HMI\RT_Projects\{ProjectName}.{DeviceName}

Par exemple :

"C:\Desktop\hmim14000100a\Simatic.HMI\RT_Projects\DasBasicUndMobilePanelen.hmim14000100a[KTP700 Mobile]"

Remarque

ProgramUpdate pour une HMI est différent de la méthode pour une CPU. Pour un appareil IHM, cette méthode met à jour le système d'exploitation et le logiciel runtime. Vous ne pouvez pas choisir une mise à jour partielle. Si nécessaire, SIMATIC Automation Tool met à jour tous les composants afin d'assurer un téléchargement cohérent. Une carte de mise à jour du programme IHM peut comporter plusieurs projets nécessitant la saisie d'un dossier sous \Simatic.HMI\RT_Projects\.

7.13.7 Méthode SetBackupFile

Type de retour	Nom de la méthode
Result	SetBackupFile

Paramètres			
Nom	Type de données	Type de paramètre	Description
strFile	string	in	Définit l'emplacement du dossier où la source de fichier de sauvegarde est stockée.

Auparavant, le paramètre de chemin d'accès au fichier pour l'opération de restauration était transmis à l'objet avec l'appel de méthode. SIMATIC Automation Tool V3.1 introduit une nouveauté en vous permettant de définir l'emplacement du dossier.

La méthode définit les drapeaux suivants pour l'objet ICPU :

- NewRestoreName
- NewRestoreFile
- NewRestoreNameisValid

L'exemple suivant montre comment définir le chemin du fichier de sauvegarde sur un appareil IHM.

```
Network myNetwork = new Network();
ulong targetMACAddress = 0x112233445566; // 11:22:33:44:55:66
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (retVal.Succeeded)
{
    IHMI hmi = scannedDevices.FindDeviceByMAC(targetMACAddress) as IHMI;
    if (hmi != null && hmi.RestoreAllowed == true)
    {
        // Sélectionner la CPU à mettre à jour
        hmi.Selected = true;

        retVal = hmi.SetBackupFile(@"C:\MyFolder\Backup.s7pbkp");
        if (retValFailed == true)
            return;

        // Adresse IP unique ?
        if (hmi.DuplicateIP == true)
            return;

        // L'appareil est-il pris en charge ?
        if (hmi.Supported == false)
            return;

        // Effectuer une restauration
        retVal = hmi.Restore();
    }
}
```

7.14 Prise en charge des classes

7.14.1 Diagnosticsitem

Un élément de diagnostic contient des informations de diagnostic pour un seul événement.

Nom de la propriété	Type de retour	Description
TimeStamp	DateTime	Horodatage de l'événement de diagnostic
State	byte	0 = événement sortant 1 = événement entrant
Description1	string	Description sommaire
Description2	string	Description détaillée

7.15 Exceptions

7.15.1 CriticalInternalErrorException

Principe

L'interface API V3.1 a ajouté une exception personnalisée en cas de détection d'un état critique.

Lorsque cette exception est déclenchée, il est recommandé de fermer l'application utilisant l'API. Lorsqu'une erreur critique survient dans le code, cette exception est appelée.

```
Network myNetwork = new Network();
try
{
    uint targetIPAddress = 0xC0A80001; // 192.168.0.1
    IProfinetDeviceCollection devices;

    Result retVal = myNetwork.ScanNetworkDevices(out devices);
    if (retVal.Succeeded)
    {
        IProfinetDevice dev = devices.FindDeviceByIP(targetIPAddress);
        if (dev != null)
        {
            ICPu devAsCpu = dev as ICPu;
            if (devAsCpu != null)
            {
                devAsCpu.SetPassword(new EncryptedString("Password"));
                devAsCpu.Selected = true;
                if (devAsCpu.Failsafe) devAsCpu.SelectedConfirmed = true;
                retVal = devAsCpu.ResetToFactoryDefaults();
            }
        }
    }
}
catch (CriticalInternalErrorException e)
{
    // Une erreur critique est survenue dans l'API
}
catch (Exception e)
{
    // Une exception s'est produite dans l'API
}
```

7.16 Enumérations API

7.16.1 DataChangedType

Cette énumération définit les valeurs d'argument possibles pour le DataChangedEventHandler (Page 182).

```
Invalid
OperatingState
RackInformation
Folders
File
ProfinetName
IPAddress
Password
```

7.16.2 DeviceFamily

Cette énumération indique la gamme de produits d'un matériel.

```
CPU1200
CPU1500
ET200AL
ET200ECO
ET200M
ET200MP
ET200PRO
ET200S
ET200SP
HMI
NetworkDevice
None
SITOPUPS
Unsupported
```

7.16.3 ConfirmationType

L'énumération précise les types de confirmation utilisateur pour les opérations relative à la sécurité.

```
Invalid = 0
SafetyPasswordIsBeingUsed = 0x2f161717
DeletingExistingSafetyProgram = 0x40232122
ReplacingExistingSafetyProgram = 0x492a282b
ReplacingExistingSafetyProgramWithNonSafetyProgram = 0x4a2c2b2d
LoadingSafetyProgram = 0x46292728,
```

7.16.4 ErrorCode

Cette énumération indique toutes les valeurs de retour possibles pour un objet Result.

```
OK
AccessDenied
ServiceTimeout
Disconnected
FailedToDisconnect
ServiceNotConnected
TooManySessions
SessionDelegitimated
NotChangableInRun
InvalidFileName
MultiESNotSupported
ServiceAborted
MultiESLimitExceeded
MultiESIncompatibleOtherESVersion
MultiESConflict
WriteProtected
DiskFull
InvalidVersion
Failed
CPUFailedToEnterRunMode
MACAddressIsNotValid
IPAddressIsNotValid
SubnetMaskIsNotValid
GatewayIsNotValid
ProfinetNameIsNotValid
NewIPAddressIsNotValid
NewSubnetMaskIsNotValid
NewGatewayIsNotValid
NewProfinetNameIsNotValid
InvalidPointer
SetIPErrorDueProjectSettings
UnsupportedDevice
SetNameErrorDueProjectSettings
OperationNotSupportedByThisDevice
DeviceNotOnNetwork
FirmwareVersionMatch
FirmwareFileNotCompatibleToNew
FirmwareFileNotCompatibleToOld
FirmwareFileNotCompatibleNotSame
FirmwareFileNotCompatibleSame
FirmwareFileNotCompatibleBuildType
FirmwareFileNotCompatible
FirmwareModuleNotReachable
FirmwareModuleNotAccepted
FirmwareIDNotFound
WriteBlockFailed
InvalidProjectVersion
DeviceIsNotAcceptingChanges
InvalidSignature
```

```
ParameterOutOfRange
FailedToZipFolderContents
ErrorWritingToFile
ErrorCreatingFile
ErrorCreatingFolder
NoSATLicensePresent
InvalidTimeoutValue
NoDataToBackup
ErrorWritingToStream
ErrorReadingFromStream
InvalidProjectPath
ProjectNotCompatibleWithDevice
FailedToSetProfinetName
FailedToSetIPAddress
DownloadInvalidRecipe
IdentityFailure
DeviceMismatch
InvalidInterface
DeviceNotSelected
FailsafeAccessRequired
InternalApplicationError
InvalidPassword
DuplicateIPAddress
DuplicateProfinetName
SafetyDeviceMustBeConfirmed
NoSDCardPresent
InvalidProgramFolder
FSignaturesDoesNotMatch
FSignaturesMatch
DeviceDoesNotSupportProject
ProjectsUpdateIPNotReachable
RestoreIPNotReachable
ProjectIPNotUnique
SafetyProjectDownloadedToStandardNotAllowed
PasswordDiversityFailed
InvalidBackupFile
IncompatibleBackupFile
InvalidFirmwareFile
OperationWasNotSuccessful
CouldNotValidatePassword
IPAddressAlreadyExistsOnNetwork
MissingProgramFilePassword
InvalidProgramFilePassword
OperationCancelledByUser
InvalidProgramForDevice
InvalidProgramFilePasswordLegitimizationLevel
RestoreAuthenticationWarning
InvalidCPUPassword
MissingPAOM
DeviceNotFound
DeviceAlreadyExists
IPAddressAlreadyOnNetwork
```

```
ProfinetNameAlreadyOnNetwork
FailedToConnect
DeviceNotInitialized
CPUNewerVersionNotSupported
IPSuiteNotValid
IPAddressChanged
ScanNoDevicesFound
DeviceCannotBeInserted
InsertDeviceDuplicateIP
InvalidImportLine
IPNotReachable
CouldNotReadFSignature
InvalidNetworkInterface
InsufficientLegitimizationLevel
NoProgramPassword
UnhandledExceptionOccured
ErrorLoadingOfflineDatabase
ProjectVersionV1NotSupported
ProjectOpenCanceled
ProgramPasswordNeeded
InvalidImportFile
FailsafeControlObjectNotFound
RestoreError
IncompatibleProgramFile
UnsupportedProgramFile
ProgramFileFamilyMismatch
DuplicateNewIPAddress
PLCSimAdvancedIsRunning
UnexpectedOperatingSystemError
ServiceActive
RemoteTransferDisabled
HardwareSoftwareNotComplete
LogicalVolumeMissing
LogicalVolumeOutOfSpace
Abort
FirmwareTypeNotSupported
FirmwareTypeNotInstalled
StoreReadFailed
StoreWriteFailed
RescueBackupNotPossible
RescueRestoreNotPossible
ConnectionRequired
ObjectNotFound
BufferTooSmall
InvalidArguments
AttributeNotFound
InvalidPath
TypeConversionFailed
FileReadFailed
FileWriteFailed
OutOfResources
OutOfSpace
```

```
UnknownAddon
IncompatibleAddon
AddonsUnsupported
LicenseFailed
UnknownApp
UnknownAppAddon
UnknownReferenceApp
RuntimeMissing
RuntimeBroken
SignatureRequired
SignatureInvalid
SignatureFailure
CertificateInvalid
CertificateFailure
CertificateNotReady
CertificateExpired
CertificateRevoked
SecurityLib
WrongRuntimeVersion
MajorRuntimeDowngrade
MajorRuntimeUpgrade
MajorImageDowngrade
MajorImageUpgrade
WrongRuntime
NotEnoughMemory
ProjectCharacteristicsMissing
ProjectCharacteristicsInvalid
PanelOrientationIsPortrait
PanelOrientationIsLandscape
WrongDevicetype
NoRuntimeInstalled
RuntimeCorrupt
```

7.16.5

Language

L'énumération Language permet de définir la langue des chaînes de caractères retournées. Elle contient les valeurs suivantes :

```
English
German
French
Spanish
Italian
```

7.16.6 OperatingState

Cette énumération définit les états possibles pour la propriété OperatingState.

```
NotSupported
StopFwUpdate
StopSelfInitialization
Stop
Startup
Run
RunRedundant
Halt
LinkUp
Update
Defective
ErrorSearch
NoPower
CiR
STOPwithoutODIS
RunODIS
```

7.16.7 OperatingStateREQ

Cette énumération définit les transitions d'état qui peuvent être demandées dans un appel de la méthode SetOperatingState (Page 215).

```
Stop
Run
```

7.16.8 ProgressAction

Cette énumération définit les valeurs d'argument possibles qui peuvent être envoyées au ProgressChangedEventHandler (Page 183).

```
Invalid
Connecting
Reconnecting
Disconnecting
Initializing
Updating
Processing
Downloading
Uploading
Deleting
Reseting
Rebooting
Verifying
Formatting

Refreshing
Finished
UpdatingFirmware
InstallingRuntime
InstallingAddOns
UninstallingAddOns
UpdatingProgram
```

7.16.9 RemoteInterfaceType

Cette énumération définit les états qui peuvent être retournés à partir de l'appel de la propriété `InterfaceType` sur l'interface `IRemoteInterfaces` (Page 226).

```
None
Profinet
Profibus
ASi
```

7.16.10 FeatureSupport

SIMATIC Automation Tool fournit cette énumération pour indiquer les fonctions prises en charge par chaque appareil.

```
Uninitialized
BackupAllowed
ChangeModeAllowed
FirmwareUpdateAllowed
MemoryResetAllowed
PasswordAllowed
ProgramUpdateAllowed
ResetToFactoryAllowed
FormatMCAAllowed
NotFailsafe
RestoreAllowed
RemoteDataLogsAllowed
RemoteRecipesAllowed
Supported
FormatMCAAllowed
Failsafe
ServiceDataAllowed
SetTimeAllowed
DiagBufferAllowed
```

Pour savoir si un appareil prend en charge une fonction données, comparez la valeur de la propriété concernée à la valeur `FeatureSupport` définie pour cette fonction.

Par exemple, le code suivant vérifie si un appareil prend en charge la fonction RAZ mémoire avant de commencer l'opération :

```
Network myNetwork = new Network();
IProfinetDeviceCollection scannedDevices;

Result retVal = myNetwork.ScanNetworkDevices(out scannedDevices);
if (retVal.Succeeded)
{
    List<ICPU> cpus = scannedDevices.FilterOnlyCpus();
    foreach (ICPU cpu in cpus)
    {
        if (cpu.MemoryResetAllowed)
        {

        }
    }
}
```

7.16.11 ProtectionLevel

L'énumération `ProtectionLevel` donne le niveau de protection d'un mot de passe de la CPU :

```
Unknown
Failsafe
Full
Read
HMI
NoAccess
NoPassword
```

7.16.12 ConfirmationType

Cette énumération est utilisée pour indiquer le statut de CPU de sécurité.

Description	Valeur
Invalid	0
SafetyPasswordIsBeingUsed	0x2f161717
DeletingExistingSafetyProgram	0x40232122
ReplacingExistingSafetyProgram	0x492a282b
ReplacingExistingSafetyProgramWithNonSafetyProgram	0x4a2c2b2d
LoadingSafetyProgram	0x46292728

7.16.13 FailsafeOperation

L'énumération `FailSafeOperation` indique des opérations relative à la sécurité.

Description	Valeur
Invalid	0
ResetToFactoryOperation	0x2f161717
FormatMCOperation	0x46292728
ProgramUpdateOperation	0x43252224
RestoreOperation	0x45262427

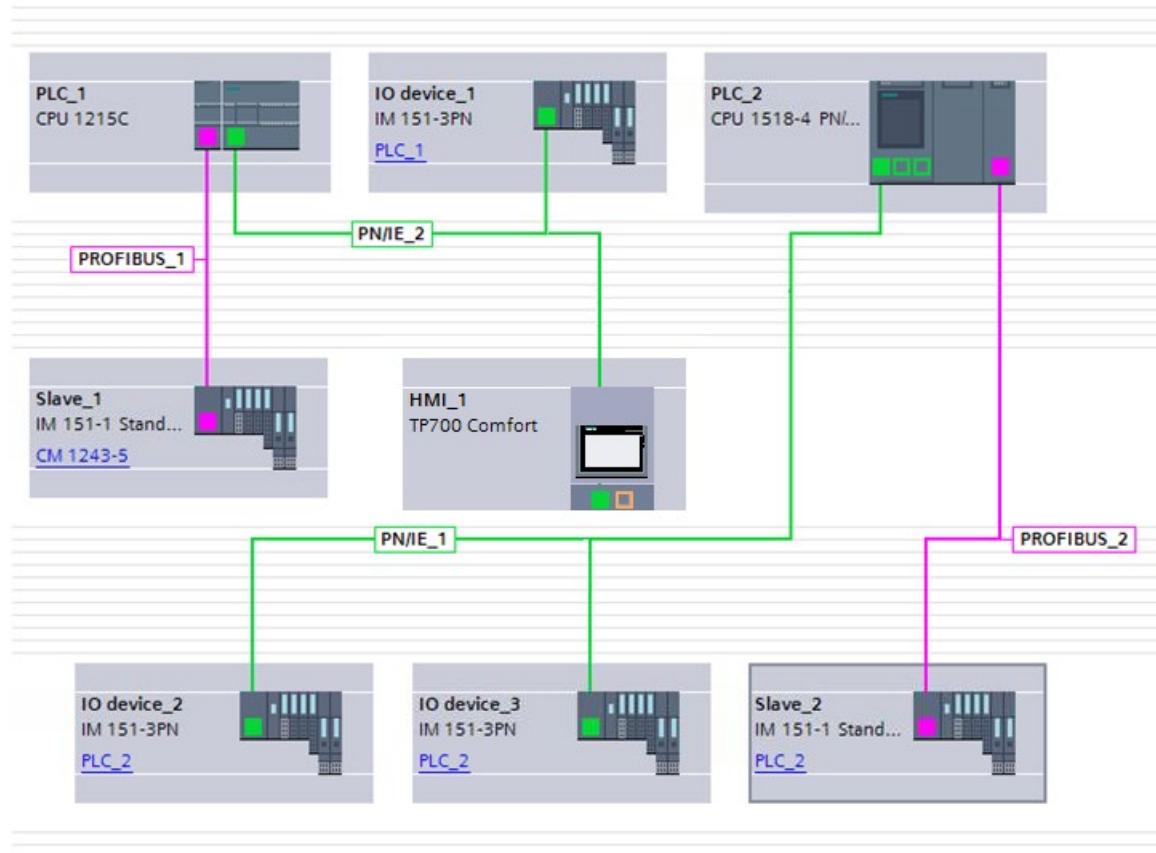
7.16.14 RemoteFolderType

L'énumération `RemoteFolderType` indique le type de dossier distant.

Description	Valeur
None	0
Recipe	1
Datalog	2

7.17 Exemple de réseau

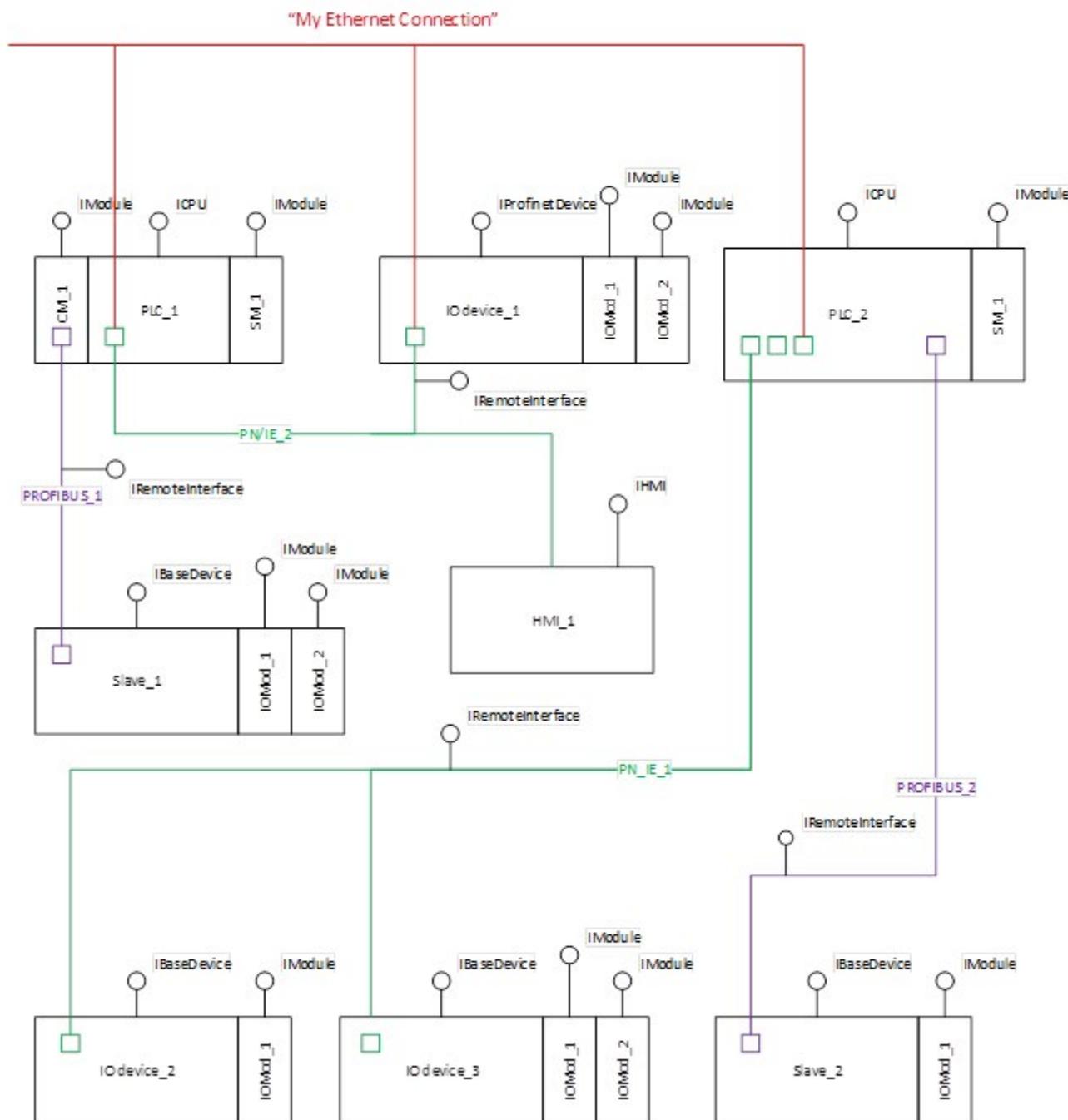
Cet exemple montre une configuration réseau TIA Portal et les interfaces API qui représentent les appareils mis en réseau.



On postule que tous les appareils de la première ligne (PLC_1, IO device_1 et PLC_2) sont raccordés à un réseau Ethernet externe (non affiché) et peuvent donc être accédés directement par l'API SIMATIC Automation Tool. On postule également que le sous-réseau PROFINET raccordé au PLC_2 n'est pas raccordé au réseau externe.

L'API SIMATIC Automation Tool peut fournir des informations et des opérations pour tous les API et les stations E/S de cette configuration.

Le diagramme suivant montre la même configuration réseau et les appareils physiques du réseau.



Dans le diagramme ci-dessus, le symbole en sucette (lollipop) montre la classe d'interface API SIMATIC Automation Tool qui représente le mieux chaque composant du réseau.

- Les CPU raccordées directement au réseau externe sont représentées par l'interface `ICPU`
- Les stations I/O raccordées directement au réseau externe sont représentées par l'interface `IProfinetDevice`
- Les sous-réseaux provenant d'une CPU sont représentés par l'interface `IRemoteInterface`
- Les stations I/O non raccordées directement au réseau externe (mais accessibles par une CPU) sont représentées par l'interface `IBaseDevice`.
- Les modules E/S ou les modules de communication raccordés à une CPU ou une station E/S sont représentés par l'interface `IModule`

8.1

Versions de firmware et appareils non reconnus

Appareils Siemens pris en charge

Les tables de prise en charge des appareils montrent les appareils Siemens pris en charge, les versions de firmware prises en charge et les opérations prises en charge.

Appareils Siemens et de tiers non pris en charge

Si vous connectez un appareil non pris en charge à votre réseau, deux cas de figure sont possibles.

- SIMATIC Automation Tool prend en charge l'appareil Siemens, mais la version de firmware est plus récente que la dernière version de firmware prise en charge.
 - L'appareil est affiché dans une ligne de la table Appareil.
 - L'icône "Appareil pris en charge" s'affiche dans la ligne.
 - Les opérations possibles sont limitées aux opérations prises en charge dans une version de firmware antérieure prise en charge.
- SIMATIC Automation Tool ne prend pas en charge l'appareil Siemens ou de tiers.
 - L'appareil est affiché dans une ligne de la table Appareil avec une partie de ses informations.
 - L'icône point d'interrogation s'affiche dans la ligne pour indiquer que l'appareil n'est pas pris en charge.
 - Les opérations DCP basées sur l'adresse MAC (Scanner le réseau complet, Identifier, Mettre à jour l'adresse IP et Mettre à jour le nom PROFINET) ne sont possibles que pour les appareils connectés au même sous-réseau que le PG/PC qui exécute SIMATIC Automation Tool.

8.2 ET 200

8.2.1 ET 200AL

8.2.1.1 Prise en charge ET 200AL IM

Prise en charge des opérations ET 200AL IM et version de firmware

Si elle est cochée (✓), l'opération est prise en charge.

Si la colonne PROFINET est prise en charge, ces opérations PROFINET sont prises en charge :

- Recherche d'appareils
- Identifier l'appareil
- Configurer l'adresse IP
- Définir le nom PROFINET

Référence	Nom du module	Version de firmware	PROFINET	Restauration des paramètres d'usine	Mise à jour du firmware
6ES7 157-1AA00-0AB0	IM 157-1 DP	V1.0			✓
6ES7 157-1AB00-0AB0	IM 157-1 PN	V1.0	✓	✓	✓

8.2.1.2 Prise en charge ET 200AL SM et IO-Link

ET 200AL SM, prise en charge IO-Link et version de firmware

Si elle est cochée (✓), l'opération est prise en charge.

Référence	Nom du module	Type de module	Version de firmware	Mise à jour du firmware
6ES7 142-5AF00-0BA0	DQ 8x24VDC/2A 8xM12	SM	V1.0	✓
6ES7 143-5AF00-0BA0	DIQ 4+DQ 4x24VDC/0.5A 4xM12	SM	V1.0	✓
6ES7 143-5AH00-0BA0	DIQ 16x24VDC/0.5A 8xM12	SM	V1.0	✓
6ES7 143-5BF00-0BA0	DIQ 4+DQ 4x24VDC/0.5A 8xM8	SM	V1.0	✓
6ES7 144-5KD00-0BA0	AI 4xU/I/RTD 4xM12	SM	V1.0	✓
6ES7 145-5ND00-0BA0	AQ 4xU/I 4xM12	SM	V1.0	✓
6ES7 147-5JD00-0BA0	CM 4xIO-Link 4xM12	IO-Link	V1.0	✓

8.2.2 Prise en charge de ET 200eco

Prise en charge des opérations ET 200eco et version de firmware

Si elle est cochée (✓), l'opération est prise en charge.

Si la colonne PROFINET est prise en charge, ces opérations PROFINET sont prises en charge :

- Recherche d'appareils
- Identifier l'appareil
- Configurer l'adresse IP
- Définir le nom PROFINET

Référence	Nom du module	Version de firmware	PROFINET	Restauration des paramètres d'usine	Mise à jour du firmware
6ES7 141-6BF00-0AB0	8DI x 24VDC 4xM12	V6.0, 7,0	✓	✓	✓
6ES7 141-6BG00-0AB0	8DI x 24VDC 8xM12	V6.0, 7,0	✓	✓	✓
6ES7 141-6BH00-0AB0	16DI x 24VDC 8xM12	V6.0, 7,0	✓	✓	✓
6ES7 142-6BF00-0AB0	8DO x 24VDC / 1.3A 4xM12	V6.0, 7,0	✓	✓	✓
6ES7 142-6BF50-0AB0	8DO x 24VDC / 0.5A 4xM12	V6.0, 7,0	✓	✓	✓
6ES7 142-6BG00-0AB0	8DO x 24VDC / 1.3A 8xM12	V6.0, 7,0	✓	✓	✓
6ES7 142-6BH00-0AB0	16DO x 24VDC / 1.3A 8xM12	V6.0, 7,0	✓	✓	✓
6ES7 142-6BR00-0AB0	8DO x 24VDC / 2.0A 8xM12	V6.0, 7,0	✓	✓	✓
6ES7 144-6KD00-0AB0	8AI x 4U/I + 4RTD/TC 8 x M12	V6.0, 7,0	✓	✓	✓
6ES7 144-6KD50-0AB0	8AI x RTD/TC 8xM12	V7.0	✓	✓	✓
6ES7 145-6HD00-0AB0	4AO x 4U/I 4 x M12	V6.0, 7,0	✓	✓	✓
6ES7 147-6BG00-0AB0	8DI/8DO x 24VDC / 1.3A 8xM12	V6.0, 7,0	✓	✓	✓
6ES7 148-6JA00-0AB0	4IO-L + 8DI + 4DO x 24VDC / 1.3A 8xM12	V6.1, 7,0	✓	✓	✓
6ES7 148-6JD00-0AB0	4IO-L 4xM12	V1.0	✓	✓	✓

8.2.3 Prise en charge ET 200M IM

Prise en charge des opérations ET 200M IM et version de firmware

Si elle est cochée (✓), l'opération est prise en charge.

Si la colonne PROFINET est prise en charge, ces opérations PROFINET sont prises en charge :

- Recherche d'appareils
- Identifier l'appareil
- Configurer l'adresse IP
- Définir le nom PROFINET

Référence	Nom du module	Version de firmware	PROFINET	Restauration des paramètres d'usine	Mise à jour du firmware
6ES7 153-1AA03-0XB0	IM 153-1			✓	✓
6ES7 153-2BA02-0XB0	IM 153-2			✓	✓
6ES7 153-2BA10-0XB0	IM 153-2	V6.0		✓	✓
6ES7 153-2BA70-0XB0	IM 153-2 OD	V6.0		✓	✓
6ES7 153-2BA82-0XB0	IM 153-2 OD			✓	✓
6ES7 153-2BB00-0XB0	IM 153-2 FO			✓	✓
6ES7 153-4AA01-0XB0	IM 153-4 PN	V2.0, 3.0, 4.0	✓	✓	✓
6ES7 153-4BA00-0XB0	IM 153-4 PN	V3.0, 4.0	✓	✓	✓
6ES7 360-3AA01-0AA0	IM 360 IM S			✓	✓
6ES7 361-3CA01-0AA0	IM 361 IM R			✓	✓
6ES7 365-0BA01-0AA0	IM 365 IM S-R			✓	✓

8.2.4 Prise en charge ET 200MP IM

Prise en charge des opérations ET 200MP IM et version de firmware

Si elle est cochée (✓), l'opération est prise en charge.

Si la colonne PROFINET est prise en charge, ces opérations PROFINET sont prises en charge :

- Recherche d'appareils
- Identifier l'appareil
- Configurer l'adresse IP
- Définir le nom PROFINET

Référence	Nom du module	Version de firmware	PROFINET	Restauration des paramètres d'usine	Mise à jour du firmware
6AG1 155-5AA00-7AB0	IM 155-5 PN ST SIPLUS	V1.0, 2.0, 3.0	✓	✓	✓
6ES7 155-5AA00-0AA0	IM 155-5 PN BA	V4.0	✓	✓	✓
6ES7 155-5AA00-0AB0	IM 155-5 PN ST	V1.0, 2.0, 3.0	✓	✓	✓
6ES7 155-5AA00-0AC0	IM 155-5 PN HF	V1.0, 3.0	✓	✓	✓
6ES7 155-5BA00-0AB0	IM 155-5 DP ST	V2.0, 3.0			✓

8.2.5 ET 200S

Prise en charge des opérations ET 200S et version de firmware

Si elle est cochée (✓), l'opération est prise en charge.

Si la colonne PROFINET est prise en charge, ces opérations PROFINET sont prises en charge :

- Recherche d'appareils
- Identifier l'appareil
- Configurer l'adresse IP
- Définir le nom PROFINET

Référence	Nom du module	Version de firmware	PROFINET	Restauration des paramètres d'usine	Mise à jour du firmware
6ES7 151-3AA22-0AB0	IM 151-3 PN	V5.0	✓	✓	✓
6ES7 151-3AA23-0AB0	IM 151-3 PN	V6.0, 6.1, 7.0	✓	✓	✓
6ES7 151-3BA22-0AB0	IM 151-3 PN	V5.0	✓	✓	✓
6ES7 151-3BA23-0AB0	IM 151-3 PN	V6.0, 6.1, 7.0	✓	✓	✓
6ES7 151-3BA60-0AB0	IM 151-3 PN	V3.0	✓	✓	✓
6ES7 151-3BB22-0AB0	IM 151-3 PN	V5.0	✓	✓	✓
6ES7 151-3BB23-0AB0	IM 151-3 PN	V6.1, 7.0	✓	✓	✓
6ES7 138-4FB04-0AB0	4 F-DO DC24V/2A				✓

Remarque

CPU ET 200S non prise en charge

La CPU ET 200S n'est pas prise en charge par SIMATIC Automation Tool

8.2.6 ET 200pro

8.2.6.1 Prise en charge de la CPU ET 200pro (basée sur S7-1516)

Prise en charge des opérations de la CPU ET 200pro et version de firmware

Si elle est cochée (✓), l'opération est prise en charge. Les CPU standard affichent uniquement la version de firmware dans l'en-tête de la colonne. Les CPU de sécurité affichent "Sécurité" dans l'en-tête de la colonne.

CPU 1516pro-2 PN, CPU1516pro F-2 PN	V2.0	V2.1	Sécurité	
			V2.0	V2.1
Recherche d'appareils	✓	✓	✓	✓
Identifier l'appareil	✓	✓	✓	✓
Configurer l'adresse IP	✓	✓	✓	✓
Définir le nom PROFINET	✓	✓	✓	✓
Mettre la CPU en MARCHE/ARRÊT	✓	✓	✓	✓
Régler l'heure sur l'heure PG/PC	✓	✓	✓	✓
Mise à jour du programme	✓	✓	✓	✓
Accès distant aux recettes	✓	✓	✓	✓
Accès distant aux fichiers journaux	✓	✓	✓	✓
Sauvegarde	✓	✓	✓	✓
Restauration	✓	✓	✓	✓
Chargement Service Data	✓	✓	✓	✓
Lecture du tampon de diagnostic	✓	✓	✓	✓
Restaurer la mémoire de la CPU	✓	✓	✓	✓
Restauration des paramètres d'usine	✓	✓	✓	✓
Formater la carte mémoire	✓	✓	✓	✓
Mise à jour du firmware	✓	✓	✓	✓

8.2.6.2 Prise en charge ET 200pro IM

Prise en charge des opérations ET 200pro IM et version de firmware

Si elle est cochée (✓), l'opération est prise en charge.

Si la colonne PROFINET est prise en charge, ces opérations PROFINET sont prises en charge :

- Recherche d'appareils
- Identifier l'appareil
- Configurer l'adresse IP
- Définir le nom PROFINET

Référence	Nom du module	Version de firmware	PROFINET	Restauration des paramètres d'usine	Mise à jour du firmware
6ES7 154-4AB10-0AB0	IM 154-4 Cu	V5.0, 6.0, 7.0, 7.1	✓		✓
6ES7 154-6AB00-0AB0	IM 154-6 IWLAN	V1.0	✓		✓
6ES7 154-6AB50-0AB0	IM 154-6 IWLAN	V1.0	✓		✓

8.2.6.3 ET 200pro IO-Link, prise en charge RFID

ET 200pro IO-Link, prise en charge RFID et version de firmware

Si elle est cochée (✓), l'opération est prise en charge.

Référence	Nom du module	Type de module	Version de firmware	Mise à jour du firmware
6ES7 147-4JD00-0AB0	CM 4xIO-Link 4xM12	IO-Link	V1.0	✓
6GT2 002-0HD00	RF170C	RFID	V1.0	✓
6GT2 002-0HD01	RF170C	RFID	V3.0	✓

8.2.7 ET 200SP

8.2.7.1 Prise en charge de la CPU ET 200SP (basée sur S7-151x)

Prise en charge des opérations CPU ET 200SP et version de firmware

Si elle est cochée (✓), l'opération est prise en charge. Les CPU standard affichent uniquement la version de firmware dans l'en-tête de la colonne. Les CPU de sécurité affichent "Sécurité" dans l'en-tête de la colonne.

CPU 1510SP-1 PN CPU 1510SP F-1 PN CPU 1512SP-1 PN CPU 1512SP F-1 PN	V1.6	V1.7	V1.8	V2.0	V2.1	Sécurité				
						V1.6	V1.7	V1.8	V2.0	V2.1
Recherche d'appareils	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Identifier l'appareil	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Configurer l'adresse IP	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Définir le nom PROFINET	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Transition MARCHE/ARRÊT	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Régler l'heure sur l'heure PG/PC	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Mise à jour du programme	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Accès distant aux recettes	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Accès distant aux fichiers journaux	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Sauvegarde		✓	✓	✓	✓	✓	✓	✓	✓	✓
Restauration		✓	✓	✓	✓	✓	✓	✓	✓	✓
Chargement Service Data	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Lecture du tampon de diagnostic	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Restaurer la mémoire de la CPU	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Restaurer les paramètres d'usine	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Formater la carte mémoire	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Mise à jour du firmware	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

8.2.7.2 Prise en charge ET 200SP IM et module serveur

Prise en charge ET 200SP IM et module serveur

Si elle est cochée (✓), l'opération est prise en charge.

Si la colonne PROFINET est prise en charge, ces opérations PROFINET sont prises en charge :

- Recherche d'appareils
- Identifier l'appareil
- Configurer l'adresse IP
- Définir le nom PROFINET

Référence	Nom du module	Versions de firmware	PROFINET	Restauration des paramètres d'usine	Mise à jour du firmware
6AG1 155-6AU00-7BN0	IM 155-6 PN ST SIPLUS	V1.0, 1,1, 3,1, 3,3	✓	✓	✓
6AG1 193-6PA00-7AA0	Module serveur SIPLUS	V1.0, 1,1	✓		✓
6ES7 155-6AR00-0AN0	IM 155-6 PN BA	V3.2	✓	✓	✓
6ES7 155-6AU00-0BN0	IM 155-6 PN ST	V1.0, 1,1, 3,1, 3,3	✓	✓	✓
6ES7 155-6AU00-0CN0	IM 155-6 PN HF	V2.1, 2,2, 3,0, 3,1, 3,3	✓	✓	✓
6ES7 155-6AU00-0DN0	IM 155-6 PN HS	V4.0	✓	✓	✓
6ES7 155-6AU01-0BN0	IM 155-6 PN ST	V4.1	✓	✓	✓
6ES7 155-6BU00-0CN0	IM 155-6 DP HF	V1.1, 3,0, 3,1			✓
6ES7 193-6PA00-0AA0	Module serveur	V1.0, 1,1			✓

8.2.7.3 Prise en charge ET 200SP SM, AS-i, CM, CP, TM, IO-Link, départs-moteurs

Prise en charge des ET 200SP, départs-moteurs et version de firmware

Si elle est cochée (✓), l'opération est prise en charge.

Référence	Nom du module	Version de firmware	Type de module	Mise à jour du firmware
3RK1 308-0AB00-0CP0	DS 0.3 - 1A HF 3DI/LC	V1.0, 1.1	Départ-moteur	✓
3RK1 308-0AC00-0CP0	DS 0.9 - 3A HF 3DI/LC	V1.0, 1.1	Départ-moteur	✓
3RK1 308-0AD00-0CP0	DS 2.8 - 9A HF 3DI/LC	V1.0, 1.1	Départ-moteur	✓
3RK1 308-0AE00-0CP0	DS 4 - 12A HF 3DI/LC	V1.1	Départ-moteur	✓
3RK1 308-0BB00-0CP0	RS 0.3- 1A HF 3DI/LC	V1.0, 1.1	Départ-moteur	✓
3RK1 308-0BC00-0CP0	RS 0.9- 3A HF 3DI/LC	V1.0, 1.1	Départ-moteur	✓
3RK1 308-0BD00-0CP0	RS 2.8- 9A HF 3DI/LC	V1.0, 1.1	Départ-moteur	✓
3RK1 308-0BE00-0CP0	RS 4 - 12A HF 3DI/LC	V1.1	Départ-moteur	✓
3RK1 308-0CB00-0CP0	F-DS 0.3 - 1A HF 3DI/LC	V1.0	Départ-moteur	✓
3RK1 308-0CC00-0CP0	F-DS 0.9 - 3A HF 3DI/LC	V1.0	Départ-moteur	✓
3RK1 308-0CD00-0CP0	F-DS 2.8 - 9A HF 3DI/LC	V1.0	Départ-moteur	✓
3RK1 308-0CE00-0CP0	F-DS 4 - 12A HF 3DI/LC	V1.0	Départ-moteur	✓
3RK1 308-0DB00-0CP0	F-RS 0.3 - 1A HF 3DI/LC	V1.0	Départ-moteur	✓
3RK1 308-0DC00-0CP0	F-RS 0.9 - 3A HF 3DI/LC	V1.0	Départ-moteur	✓
3RK1 308-0DD00-0CP0	F-RS 2.8 - 9A HF 3DI/LC	V1.0	Départ-moteur	✓
3RK1 308-0DE00-0CP0	F-RS 4 - 12A HF 3DI/LC	V1.0	Départ-moteur	✓
3RK7 136-6SC00-0BC1	F-CM AS-i Safety ST	V1.0	ASi	✓
3RK7 137-6SA00-0BC1	CM AS-i Master ST	V1.0, 1,1	CP	✓
6AG1 131-6BF00-7BA0	DI 8x24VDC ST SIPLUS	V1.0, 1,1	SM	✓
6AG1 131-6BH00-7BA0	DI 16x24VDC ST SIPLUS	V1.0	SM	✓
6AG1 132-6BD20-7BA0	DQ 4x24VDC/2A ST SIPLUS	V1.0, 1,1	SM	✓
6AG1 132-6BF00-7BA0	DQ 8x24VDC/0.5A ST SIPLUS	V1.0, 1,1	SM	✓
6AG1 132-6BD20-7BA0	DQ 4x24VDC/2A ST SIPLUS	V1.0, 1,1	SM	✓
6AG1 132-6BF00-7BA0	DQ 8x24VDC/0.5A ST SIPLUS	V1.0, 1,1	SM	✓
6AG1 132-6BH00-7BA0	DQ 16x24VDC/0.5A ST SIPLUS	V1.0	SM	✓
6AG1 134-6GD00-7BA1	AI 4xI 2- 4-wire ST SIPLUS	V1.0, 1,1	SM	✓
6AG1 134-6HD00-7BA1	AI 4xU/I 2-wire ST SIPLUS	V1.0, 1,1	SM	✓
6AG1 134-6JD00-2CA1	AI 4xRTD/TC 2- 3- 4-wire HF SIPLUS	V1.0, 1,1, 2,0	SM	✓
6AG1 135-6HD00-7BA1	AQ 4xU/I ST SIPLUS	V1.0, 1,1	SM	✓
6ES7 131-6BF00-0AA0	DI 8x24VDC BA	V1.0	SM	✓
6ES7 131-6BF00-0BA0	DI 8x24VDC ST	V1.0, 1,1	SM	✓
6ES7 131-6BF00-0CA0	DI 8x24VDC HF	V1.0, 1,1, 1,2, 2,0	SM	✓
6ES7 131-6BF00-0DA0	DI 8x24VDC HS	V1.0	SM	✓
6ES7 131-6BF60-0AA0	DI 8x24VDC SRC BA	V1.0	SM	✓
6ES7 131-6BH00-0BA0	DI 16x24VDC ST	V1.0, 1,1	SM	✓
6ES7 131-6FD00-0BB1	DI 4x120..230VAC ST	V1.0	SM	✓
6ES7 131-6TF00-0CA0	DI 8xNAMUR HF	V1.0	SM	✓
6ES7 132-6BD20-0BA0	DQ 4x24VDC/2A ST	V1.0, 1,1	SM	✓
6ES7 132-6BD20-0CA0	DQ 4x24VDC/2A HF	V1.0, 2,0	SM	✓

Référence	Nom du module	Version de firmware	Type de module	Mise à jour du firmware
6ES7 132-6BD20-0DA0	DQ 4x24VDC/2A HS	V1.0	SM	✓
6ES7 132-6BF00-0AA0	DQ 8x24VDC/0.5A BA	V1.0	SM	✓
6ES7 132-6BF00-0BA0	DQ 8x24VDC/0.5A ST	V1.0, 1,1	SM	✓
6ES7 132-6BF00-0CA0	DQ 8x24VDC/0.5A HF	V1.0, 1,1, 1,2, 2,0	SM	✓
6ES7 132-6BF60-0AA0	DQ 8x24VDC/0.5A SNK BA	V1.0	SM	✓
6ES7 132-6BH00-0BA0	DQ 16x24VDC/0.5A ST	V1.0, 1,1	SM	✓
6ES7 132-6FD00-0BB1	DQ 4x24..230VAC/2A ST	V1.0	SM	✓
6ES7 132-6GD50-0BA0	RQ 4x24VUC/2A CO ST	V1.0	SM	✓
6ES7 132-6HD00-0BB0	RQ 4x120VDC/230VAC/5A NO ST	V1.0, 1,1	SM	✓
6ES7 132-6HD00-0BB1	RQ 4x120VDC/230VAC/5A NO ST	V1.1	SM	✓
6ES7 132-6MD00-0BB1	RQ 4x120VDC/230VAC/5A NO MA ST	V1.0	SM	✓
6ES7 134-6FB00-0BA1	AI 2xU ST	V1.0	SM	✓
6ES7 134-6FF00-0AA1	AI 8xU BA	V1.0	SM	✓
6ES7 134-6GB00-0BA1	AI 2xI 2- 4-wire ST	V1.0	SM	✓
6ES7 134-6GD00-0BA1	AI 4xI 2- 4-wire ST	V1.0, V1.1	SM	✓
6ES7 134-6GF00-0AA1	AI 8xI 2- 4-wire BA	V1.0	SM	✓
6ES7 134-6HB00-0CA1	AI 2xU/I 2- 4-wire HF	V1.0, 2,0	SM	✓
6ES7 134-6HB00-0DA1	AI 2xU/I 2- 4-wire HS	V1.0, 1,1, 2,0	SM	✓
6ES7 134-6HD00-0BA1	AI 4xU/I 2-wire ST	V1.0, 1,1	SM	✓
6ES7 134-6JD00-0CA1	AI 4xRTD/TC 2- 3- 4-wire HF	V1.0, 1,1, 2,0	SM	✓
6ES7 134-6JF00-0CA1	AI 8xRTD/TC 2-wire HF	V2.0	SM	✓
6ES7 134-6PA00-0BD0	AI EnergyMeter ST	V1.0, 2,0	SM	✓
6ES7 134-6PA01-0BD0	AI EnergyMeter 400VAC ST	V3.0	SM	✓
6ES7 134-6PA20-0BD0	AI EnergyMeter 480VAC ST	V4.0	SM	✓
6ES7 134-6TD00-0CA1	AI 4xI 2-wire 4..20mA HART	V1.0	SM	✓
6ES7 135-6FB00-0BA1	AQ 2xU ST	V1.0	SM	✓
6ES7 135-6GB00-0BA1	AQ 2xI ST	V1.0	SM	✓
6ES7 135-6HB00-0CA1	AQ 2xU/I HF	V1.0	SM	✓
6ES7 135-6HB00-0DA1	AQ 2xU/I HS	V1.0, 1,1, 2,0	SM	✓
6ES7 135-6HD00-0BA1	AQ 4xU/I ST	V1.0, 1,1	SM	✓
6ES7 136-6BA00-0CA0	F-DI 8x24VDC HF	V1.0	SM	✓
6ES7 136-6DB00-0CA0	F-DQ 4x24VDC/2A PM HF	V1.0	SM	✓
6ES7 136-6DC00-0CA0	F-DQ 8x24VDC/0.5A PP HF	V1.0	SM	✓
6ES7 136-6PA00-0BC0	F-PM-E 24VDC/8A PPM ST	V1.0	PM	✓
6ES7 136-6RA00-0BF0	F-RQ 1x24..48VDC/24..230VAC/5A	V1.0	SM	✓
6ES7 137-6AA00-0BA0	Point à point	V1.0	CM	✓
6ES7 137-6BD00-0BA0	CM 4xIO-Link	V1.0, 2,0, 2,1	IO-Link	✓
6ES7 138-6AA00-0BA0	TM Count 1x24V	V1.0, 1,1, 1,2	TM	✓
6ES7 138-6BA00-0BA0	TM PosInput 1	V1.0, 1,1, 1,2	TM	✓
6ES7 138-6CG00-0BA0	TM Timer DIDQ 10x24V	V1.0	TM	✓
6ES7 138-6DB00-0BB1	TM Pulse 2x24V	V1.0	TM	✓
6ES7 545-5DA00-0AB0	CM DP		CP	✓
6GK7 542-6UX00-0XE0	CP 1542SP-1	V1.0	CP	✓
6GK7 542-6VX00-0XE0	CP 1542SP-1 IRC	V1.0	CP	✓

Référence	Nom du module	Version de firmware	Type de module	Mise à jour du firmware
6GK7 543-6WX00-0XE0	CP 1543SP-1	V1.0	CP	✓
7MH4 138-6AA00-0BA0	SIWAREX WP321	V1.0	TM	✓

8.3 S7-1200

8.3.1 Prise en charge de la CPU S7-1200

Prise en charge des opérations S7-1200 et version de firmware

Si elle est cochée (✓), l'opération est prise en charge. Les CPU standard affichent uniquement la version de firmware dans l'en-tête de la colonne. Les CPU de sécurité affichent "Sécurité" dans l'en-tête de la colonne.

	V1.x	V2.x	V3.x	V4.0	V4.1	V4.2	Sécurité	
							V4.1	V4.2
Recherche d'appareils		✓	✓	✓	✓	✓	✓	✓
Identifier l'appareil		✓	✓	✓	✓	✓	✓	✓
Configurer l'adresse IP		✓	✓	✓	✓	✓	✓	✓
Définir le nom PROFINET		✓	✓	✓	✓	✓	✓	✓
Transition MARCHE/ARRÊT		✓	✓	✓	✓	✓	✓	✓
Régler l'heure sur l'heure PG/PC		✓	✓	✓	✓	✓	✓	✓
Mise à jour du programme		✓	✓	✓	✓	✓		✓
Accès distant aux recettes					✓	✓	✓	✓
Accès distant aux fichiers journaux					✓	✓	✓	✓
Sauvegarde						✓		✓
Restauration						✓		✓
Chargement Service Data		✓	✓	✓	✓	✓	✓	✓
Lecture du tampon de diagnostic		✓	✓	✓	✓	✓	✓	✓
Restaurer la mémoire de la CPU		✓	✓	✓	✓	✓	✓	✓
Restaurer les paramètres d'usine		✓	✓	✓	✓	✓		✓
Formater la carte mémoire						✓	✓	✓
Mise à jour du firmware				✓	✓	✓	✓	✓

8.3.2 Prise en charge de S7-1200 I/O et autres modules

D I/O, A I/O, SB, CM, CP, et IO link

Si elle est cochée (✓), l'opération est prise en charge.

Référence	Nom du module	Version de firmware	Type de module	Mise à jour du firmware
021620-B	CM CANopen	V1.0	CM	✓
021730-B	CM CANopen (consolidé)	V1.0	CM	✓
3RK7243-2AA30-0XB0	CM 1243-2	V1.1	CM	✓
6AG1 221-1BF32-2XB0	DI 8x24VDC SIPLUS	V2.0	SM	✓
6AG1 221-1BF32-4XB0	DI 8x24VDC SIPLUS	V2.0	SM	✓
6AG1 221-1BH32-2XB0	DI 16x24VDC SIPLUS	V2.0	SM	✓
6AG1 221-1BH32-4XB0	DI 16x24VDC SIPLUS	V2.0	SM	✓
6AG1 221-3AD30-5XB0	DI 4x5VDC SIPLUS	V1.0	SignalBoard	
6AG1 221-3BD30-5XB0	DI 4x24VDC SIPLUS	V1.0	SignalBoard	
6AG1 222-1AD30-5XB0	DQ 4x5VDC SIPLUS	V1.0	SignalBoard	
6AG1 222-1BD30-5XB0	DQ 4x24VDC SIPLUS	V1.0	SignalBoard	
6AG1 222-1BF32-2XB0	DQ 8x24VDC SIPLUS	V2.0	SM	✓
6AG1 222-1BF32-4XB0	DQ 8x24VDC SIPLUS	V2.0	SM	✓
6AG1 222-1BH32-2XB0	DQ 16x24VDC SIPLUS	V2.0	SM	✓
6AG1 222-1BH32-4XB0	DQ 16x24VDC SIPLUS	V2.0	SM	✓
6AG1 222-1HF32-2XB0	DQ 8xRelay SIPLUS	V2.0	SM	✓
6AG1 222-1HF32-4XB0	DQ 8xRelay SIPLUS	V2.0	SM	✓
6AG1 222-1HH32-2XB0	DQ 16xRelay SIPLUS	V2.0	SM	✓
6AG1 222-1HH32-4XB0	DQ 16xRelay SIPLUS	V2.0	SM	✓
6AG1 222-1XF32-2XB0	DQ 8xNO/NC Relay SIPLUS	V2.0	SM	✓
6AG1 222-1XF32-4XB0	DQ 8xNO/NC Relay SIPLUS	V2.0	SM	✓
6AG1 223-0BD30-4XB0	DI 2/DQ 2x24VDC SIPLUS	V1.0	SignalBoard	
6AG1 223-0BD30-5XB0	DI 2/DQ 2x24VDC SIPLUS	V1.0	SignalBoard	
6AG1 223-1BH32-2XB0	DI 8/DQ 8x24VDC SIPLUS	V2.0	SM	✓
6AG1 223-1BH32-4XB0	DI 8/DQ 8x24VDC SIPLUS	V2.0	SM	✓
6AG1 223-1BL32-2XB0	DI 16/DQ 16x24VDC SIPLUS	V2.0	SM	✓
6AG1 223-1BL32-4XB0	DI 16/DQ 16x24VDC SIPLUS	V2.0	SM	✓
6AG1 223-1PH32-2XB0	DI 8x24VDC/DQ 8xRelay SIPLUS	V2.0	SM	✓
6AG1 223-1PH32-4XB0	DI 8x24VDC/DQ 8xRelay SIPLUS	V2.0	SM	✓
6AG1 223-1PL32-2XB0	DI 16x24VDC/DQ 16xRelay SIPLUS	V2.0	SM	✓
6AG1 223-1PL32-4XB0	DI 16x24VDC/DQ 16xRelay SIPLUS	V2.0	SM	✓
6AG1 223-1QH32-2XB0	DI/DQ 8x120VAC/DQ 8xRelay SIPLUS	V2.0	SM	✓
6AG1 223-1QH32-4XB0	DI/DQ 8x120VAC/DQ 8xRelay SIPLUS	V2.0	SM	✓
6AG1 223-3AD30-5XB0	DI 2/DQ 2x5VDC SIPLUS	V1.0	SignalBoard	
6AG1 223-3BD30-5XB0	DI 2/DQ 2x24VDC SIPLUS	V1.0	SignalBoard	
6AG1 231-4HD32-4XB0	AI 4x13BIT SIPLUS	V2.0	SM	✓
6AG1 231-4HF32-4XB0	AI 8x13BIT SIPLUS	V2.0	SM	✓
6AG1 231-5ND32-4XB0	AI 4x16BIT SIPLUS	V2.0	SM	✓
6AG1 231-5PD32-2XB0	AI 4xRTD SIPLUS	V2.0	SM	✓

Référence	Nom du module	Version de firmware	Type de module	Mise à jour du firmware
6AG1 231-5PD32-4XB0	AI 4xRTD SIPLUS	V2.0	SM	✓
6AG1 231-5PF32-2XB0	AI 8xRTD SIPLUS	V2.0	SM	✓
6AG1 231-5PF32-4XB0	AI 8xRTD SIPLUS	V2.0	SM	✓
6AG1 231-5QD32-4XB0	AI 4xTC SIPLUS	V2.0	SM	✓
6AG1 231-5QF32-4XB0	AI 8xTC SIPLUS	V2.0	SM	✓
6AG1 232-4HA30-4XB0	AQ 1x12BIT SIPLUS	V1.0	SignalBoard	
6AG1 232-4HA30-5XB0	AQ 1x12BIT SIPLUS	V1.0	SignalBoard	
6AG1 232-4HB32-4XB0	AQ 2x13BIT SIPLUS	V2.0	SM	✓
6AG1 232-4HD32-2XB0	AQ 4x14BIT SIPLUS	V2.0	SM	✓
6AG1 232-4HD32-4XB0	AQ 4x14BIT SIPLUS	V2.0	SM	✓
6AG1 234-4HE32-2XB0	AI 4x13BIT/AQ 2x14BIT SIPLUS	V2.0	SM	✓
6AG1 234-4HE32-4XB0	AI 4x13BIT/AQ 2x14BIT SIPLUS	V2.0	SM	✓
6AG1 241-1AH32-2XB0	CM 1241 (RS232) SIPLUS	V2.1, V2.2	CM	✓
6AG1 241-1AH32-4XB0	CM 1241 (RS232) SIPLUS	V2.1, V2.2	CM	✓
6AG1 241-1CH30-5XB1	CB 1241 (RS485) SIPLUS	V1.0	CommunicationBoard	
6AG1 241-1CH32-2XB0	CM 1241 (RS422/485) SIPLUS	V2.1	CM	✓
6AG1 241-1CH32-4XB0	CM 1241 (RS422/485) SIPLUS	V2.1	CM	✓
6AG1 242-5DX30-2XE0	CM 1242-5 SIPLUS	V1.0	CM	
6AG1 242-7KX30-4XE0	CP 1242-7 GPRS SIPLUS	V1.4	CM	
6AG1 243-1JX30-7XE0	CP 1243-1 DNP3 SIPLUS	V1.1	CP	✓
6AG1 243-5DX30-2XE0	CM 1243-5 SIPLUS	V1.3	CM	
6AG1 278-4BD32-2XB0	4SI IO link SIPLUS	V2.0	SM	✓
6AG1 278-4BD32-4XB0	4SI IO link SIPLUS	V2.0	SM	✓
6AT8 007-1AA10-0AA0	Système de surveillance d'état SIPLUS 1281	V1.0	SM	✓
6ES7 221-1BF30-0XB0	DI 8x24VDC	V1.0	SM	
6ES7 221-1BF32-0XB0	DI 8x24VDC	V2.0	SM	✓
6ES7 221-1BH30-0XB0	DI 16x24VDC	V1.0	SM	
6ES7 221-1BH32-0XB0	DI 16x24VDC	V2.0	SM	✓
6ES7 221-3AD30-0XB0	DI 4x5VDC	V1.0	SignalBoard	
6ES7 221-3BD30-0XB0	DI 4x24VDC	V1.0	SignalBoard	
6ES7 222-1AD30-0XB0	DQ 4x5VDC	V1.0	SignalBoard	
6ES7 222-1BD30-0XB0	DQ 4x24VDC	V1.0	SignalBoard	
6ES7 222-1BF30-0XB0	DQ 8x24VDC	V1.0	SM	
6ES7 222-1BF32-0XB0	DQ 8x24VDC	V2.0	SM	✓
6ES7 222-1BH30-0XB0	DQ 16x24VDC	V1.0	SM	
6ES7 222-1BH32-0XB0	DQ 16x24VDC	V2.0	SM	✓
6ES7 222-1HF30-0XB0	DQ 8xRelay	V1.0	SM	
6ES7 222-1HF32-0XB0	DQ 8xRelay	V2.0	SM	✓
6ES7 222-1HH30-0XB0	DQ 16xRelay	V1.0	SM	
6ES7 222-1HH32-0XB0	DQ 16xRelay	V2.0	SM	✓
6ES7 222-1XF30-0XB0	DQ 8xNO/NC Relay	V1.0	SM	
6ES7 222-1XF32-0XB0	DQ 8xNO/NC Relay	V2.0	SM	✓
6ES7 223-0BD30-0XB0	DI 2/DQ 2x24VDC	V1.0	SignalBoard	
6ES7 223-1BH30-0XB0	DI 8/DQ 8x24VDC	V1.0	SM	

Référence	Nom du module	Version de firmware	Type de module	Mise à jour du firmware
6ES7 223-1BH32-0XB0	DI 8/DQ 8x24VDC	V2.0	SM	✓
6ES7 223-1BL30-0XB0	DI 16/DQ 16x24VDC	V1.0	SM	
6ES7 223-1BL32-0XB0	DI 16/DQ 16x24VDC	V2.0	SM	✓
6ES7 223-1PH30-0XB0	DI 8x24VDC/DQ 8xRelay	V1.0	SM	
6ES7 223-1PH32-0XB0	DI 8x24VDC/DQ 8xRelay	V2.0	SM	✓
6ES7 223-1PL30-0XB0	DI 16x24VDC/DQ 16xRelay	V1.0	SM	
6ES7 223-1PL32-0XB0	DI 16x24VDC/DQ 16xRelay	V2.0	SM	✓
6ES7 223-1QH30-0XB0	DI/DO 8x120VAC/DQ 8xRelay	V1.0	SM	
6ES7 223-1QH32-0XB0	DI/DO 8x120VAC/DQ 8xRelay	V2.0	SM	✓
6ES7 223-3AD30-0XB0	DI 2/DQ 2x5VDC	V1.0	SignalBoard	
6ES7 223-3BD30-0XB0	DI 2/DQ 2x24VDC	V1.0	SignalBoard	
6ES7 226-6BA32-0XB0	F-DI 8/16x24VDC	V2.0	SM	✓
6ES7 226-6DA32-0XB0	F-DQ 4x24VDC	V2.0	SM	✓
6ES7 226-6RA32-0XB0	F-DQ 2xRelay	V2.0	SM	✓
6ES7 228-1RC51-0AA0	Module porteur amplificateur de signal de puissance	V2.0, V2.2	SM	
6ES7 228-1RC52-0AA0	Module segment amplificateur de signal de puissance	V2.0, V2.2	SM	
6ES7 231-4HA30-0XB0	AI 1x12BIT	V1.0, V2.0	SignalBoard	
6ES7 231-4HD30-0XB0	AI 4x13BIT	V1.0	SM	
6ES7 231-4HD32-0XB0	AI 4x13BIT	V2.0	SM	✓
6ES7 231-4HF30-0XB0	AI 8x13BIT	V1.0	SM	
6ES7 231-4HF32-0XB0	AI 8x13BIT	V2.0	SM	✓
6ES7 231-5ND30-0XB0	AI 4x16BIT	V1.0	SM	
6ES7 231-5ND32-0XB0	AI 4x16BIT	V2.0	SM	✓
6ES7 231-5PA30-0XB0	AI 1xRTD	V1.0, V2.0	SignalBoard	
6ES7 231-5PD30-0XB0	AI 4xRTD	V1.0	SM	
6ES7 231-5PD32-0XB0	AI 4xRTD	V2.0	SM	✓
6ES7 231-5PF30-0XB0	AI 8xRTD	V1.0	SM	
6ES7 231-5PF32-0XB0	AI 8xRTD	V2.0	SM	✓
6ES7 231-5QA30-0XB0	AI 1xTC	V1.0, V2.0	SignalBoard	
6ES7 231-5QD30-0XB0	AI 4xTC	V1.0	SM	
6ES7 231-5QD32-0XB0	AI 4xTC	V2.0	SM	✓
6ES7 231-5QF30-0XB0	AI 8xTC	V1.0	SM	
6ES7 231-5QF32-0XB0	AI 8xTC	V2.0	SM	✓
6ES7 232-4HA30-0XB0	AQ 1x12BIT	V1.0	SignalBoard	
6ES7 232-4HB30-0XB0	AQ 2x14BIT	V1.0	SM	
6ES7 232-4HB32-0XB0	AQ 2x14BIT	V2.0	SM	✓
6ES7 232-4HD30-0XB0	AQ 4x14BIT	V1.0	SM	
6ES7 232-4HD32-0XB0	AQ 4x14BIT	V2.0	SM	✓
6ES7 234-4HE30-0XB0	AI 4x13BIT/AQ 2x14BIT	V1.0	SM	
6ES7 234-4HE32-0XB0	AI 4x13BIT/AQ 2x14BIT	V2.0	SM	✓
6ES7 238-5XA32-0XB0	AI Compteur d'énergie	V2.0	SM	✓
6ES7 241-1AH30-0XB0	CM 1241 (RS232)	V1.0	CM	
6ES7 241-1AH32-0XB0	CM 1241 (RS232)	V2.0, 2,1, 2,2	CM	✓
6ES7 241-1CH30-0XB0	CM 1241 (RS485)	V1.0	CM	

Référence	Nom du module	Version de firmware	Type de module	Mise à jour du firmware
6ES7 241-1CH30-1XB0	CB 1241 (RS485)	V1.0	CommunicationBoard	
6ES7 241-1CH31-0XB0	CM 1241 (RS422/485)	V1.0	CM	
6ES7 241-1CH32-0XB0	CM 1241 (RS422/485)	V2.0. 2.1, 2.2	CM	✓
6ES7 278-4BD32-0XB0	4SI IO link	V2.0	SM	✓
6ES7 972 0MD00 0XA0	Module TS RNIS	V1.0	CP	✓
6ES7 972 0MG00 0XA0	Module TS GSM	V1.0	CP	✓
6ES7 972 0MM00 0XA0	Module TS Modem	V1.0	CP	✓
6ES7 972 0MS00 0XA0	Module TS RS232	V1.0	CP	✓
6GK7 242-5DX30-0XE0	CM 1242-5	V1.0	CM	
6GK7 242-7KX30-0XE0	CP 1242-7	V1.0, V1.3, V1.4	CM	
6GK7 242-7KX31-0XE0	CP 1242-7	V2.1, V3.0	CP	✓
6GK7 243-1BX30-0XE0	CP 1243-1	V2.0, V2.1, 3.0	CP	✓
6GK7 243-1HX30-0XE0	CP 1243-1 PCC	V2.0	CP	✓
6GK7 243-1JX30-0XE0	CP 1243-1 DNP3	V1.0, V1.1	CP	✓
6GK7 243-1PX30-0XE0	CP 1243-1 IEC	V1.1, V1.2	CP	✓
6GK7 243-7KX30-0XE0	CP 1243-7 LTE	V2.1, V3.0	CP	✓
6GK7 243-7SX30-0XE0	CP 1243-7 LTE	V2.1, V3.0	CP	✓
6GK7 243-8RX30-0XE0	CP 1243-8 IRC	V2.1, V3.0	CP	✓
6GT2 002-0LA00	RFC120C	V1.0	CM	
7MH4 960-2AA01	SIWAREX WP231		TM	
7MH4 960-4AA01	SIWAREX WP241		TM	
7MH4 960-6AA01	SIWAREX WP251		TM	

8.4 S7-1500

8.4.1 Prise en charge de la CPU S7-1500

Prise en charge des opérations S7-1500 et version de firmware

Si elle est cochée (✓), l'opération est prise en charge. Les CPU standard affichent uniquement la version de firmware dans l'en-tête de la colonne. Les CPU de sécurité affichent "Sécurité" dans l'en-tête de la colonne.

S7-1500	V1.0	V1.1	V1.5	V1.6	V1.7	V1.8	V2.0	V2.1	Sécurité					
									V1.5	V1.6	V1.7	V1.8	V2.0	V2.1
Recherche d'appareils	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Identifier l'appareil	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Configurer l'adresse IP	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Définir le nom PROFINET	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Mettre la CPU en MARCHE/ARRÊT	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Régler l'heure sur l'heure PG/PC	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Mise à jour du programme	✓	✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓
Accès distant aux recettes	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Accès distant aux fichiers journaux	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Sauvegarde					✓	✓	✓	✓		✓	✓	✓	✓	✓
Restauration					✓	✓	✓	✓		✓	✓	✓	✓	✓
Chargement Service Data	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Lecture du tampon de diagnostic	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Restaurer la mémoire de la CPU	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Restaurer les paramètres d'usine	✓	✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓
Formater la carte mémoire				✓	✓	✓	✓	✓		✓	✓	✓	✓	✓
Mise à jour du firmware	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Remarque

Les CPU S7-1500 ODK (Open development kit) ne sont plus prises en charge par SIMATIC Automation Tool

Non pris en charge :

6ES7 518-4AP00-3AB0 CPU 1518-4 PN/DP ODK

6ES7 518-4FP00-3AB0 CPU 1518F-4 PN/DP ODK

6ES7 518-4FP00-3AB0 CPU 1518F-4 PN/DP ODK

6ES7 518-4FP00-3AB0 CPU 1518F-4 PN/DP ODK

8.4.2 Prise en charge de S7-1500 I/O et autres modules

D I/O, A I/O, CP, TM et PM

Si elle est cochée (✓), l'opération est prise en charge.

Prise en charge de la version de firmware de l'appareil et de la mise à jour du firmware

Modules d'alimentation électrique S7-1500	Opération de mise à jour du firmware
6EP1 332-4BA00	✓
6EP1333-4BA00	✓

Module TIM S7-1500	Opération de mise à jour du firmware
6GK7 543-1MX00-0XE0	✓ pour V0.1, V1.0

S7-1500 I/O et autres modules	V1.x	V2.x
Mise à jour du firmware	✓	✓

8.5 SIMATIC HMI (interface homme-machine)

8.5.1 Prise en charge des pupitres HMI Basic

Les groupes d'appareils SIMATIC HMI suivants sont pris en charge par SIMATIC Automation Tool.

Groupe de pupitres SIMATIC HMI	Modèles de pupitres pris en charge (modèles PROFINET uniquement)
Basic	KTP400 Basic
	KTP700 Basic
	KTP900 Basic
	KTP1200 Basic

Prise en charge de la version de firmware et des opérations des pupitres HMI

Opération	Version de firmware IHM supérieure ou égale à V13.0.0.0
Mise à jour du programme	✓ pour les versions de firmware V13.0.0.0 et suivantes SIMATIC Automation Tool peut mettre à jour le système d'exploitation et le logiciel exécutif de l'appareil IHM avec l'opération Mise à jour le programme.
Sauvegarde	✓ pour les versions de firmware V13.0.1.0 et suivantes
Restauration	✓ pour les versions de firmware V13.0.1.0 et suivantes

SIMATIC Automation Tool prend en charge les opérations PROFINET DCP :

- Recherche d'appareils
- Identifier l'appareil
- Configurer l'adresse IP
- Définir le nom PROFINET

8.5.2 Prise en charge des pupitres HMI Comfort

Les groupes d'appareils SIMATIC HMI suivants sont pris en charge par SIMATIC Automation Tool.

Groupe de pupitres SIMATIC HMI	Modèles de pupitres pris en charge
Comfort	KP400, KTP400 Comfort
	KP700, TP700 Comfort
	KP900, TP900 Comfort
	KP1200, TP1200 Comfort
	KP1500, TP1500 Comfort
	TP1900 Comfort
	TP2200 Comfort

Si elle est cochée (✓), l'opération est prise en charge.

Opération	Version de firmware IHM
Mise à jour du programme	✓ pour les versions de firmware V13.0.0.0 et suivantes La mise à jour du programme de SIMATIC Automation Tool pour les appareils IHM peut mettre à jour le firmware IHM, le système d'exploitation et les données de projet d'exécution.
Sauvegarde	✓ pour les versions de firmware V13.0.1.0 et suivantes
Restauration	✓ pour les versions de firmware V13.0.1.0 et suivantes

SIMATIC Automation Tool prend en charge les opérations PROFINET DCP :

- Recherche d'appareils
- Identifier l'appareil
- Configurer l'adresse IP
- Définir le nom PROFINET

8.5.3 Prise en charge des pupitres HMI Mobile

Les groupes d'appareils SIMATIC HMI suivants sont pris en charge par SIMATIC Automation Tool.

Groupe de pupitres SIMATIC HMI	Modèles de pupitres pris en charge
Mobile	KTP700 Mobile
	KTP900 Mobile

Prise en charge de la version de firmware et des opérations des pupitres HMI

Opération	Versions de firmware IHM V13.0.1.0 et suivantes
Mise à jour du programme	✓ SIMATIC Automation Tool peut mettre à jour le système d'exploitation et le logiciel exécutif de l'appareil IHM avec l'opération Mise à jour le programme.
Sauvegarde	✓
Restauration	✓

SIMATIC Automation Tool prend en charge les opérations PROFINET DCP :

- Recherche d'appareils
- Identifier l'appareil
- Configurer l'adresse IP
- Définir le nom PROFINET

8.6 SITOP (Alimentations)

8.6.1 Prise en charge de SITOP (alimentation)

Si elle est cochée (✓), l'opération est prise en charge.

Référence	Nom du module	Version de firmware	Type de module	Restauration des paramètres d'usine	Mise à jour du firmware
6EP3 436-8MB00-2CY0	PSU8600	V1.1, V1.2	IM		✓
6EP3 436-8SB00-2AY0	PSU8600	V1.1, V1.2	IM		✓
6EP3 437-8MB00-2CY0	PSU8600	V1.0, V1.1, V1.2	IM		✓
6EP3 437-8SB00-2AY0	PSU8600	V1.1, V1.2	IM		✓
6EP4 134-3AB00-2AY0	UPS1600 10A PN	V1.14, V1.22, V2.0, V2.1, V2.2	SITOP	✓	✓
6EP4 136-3AB00-2AY0	UPS1600 20A PN	V1.14, V1.22, V2.0, V2.1, V2.2	SITOP	✓	✓
6EP4 137-3AB00-2AY0	UPS1600 40A PN	V2.0, V2.1	SITOP	✓	✓
6EP4 293-8HB00-0XY0	BUF8600	V1.1, V1.2	SITOP		✓
6EP4 295-8HB00-0XY0	BUF8600	V1.1, V1.2	SITOP		✓
6EP4 297-8HB00-0XY0	BUF8600	V1.0, V1.1, V1.2	SITOP		✓
6EP4 297-8HB10-0XY0	BUF8600	V1.0, V1.1, V1.2	SITOP		✓
6EP4 436-8XB00-0CY0	CNX8600	V1.0, V1.1, V1.2	SITOP		✓
6EP4 437-8XB00-0CY0	CNX8600	V1.0, V1.1, V1.2	SITOP		✓

8.7 RFID et MOBY (modules de communication)

8.7.1 RFID (identification par radiofréquence)

Si elle est cochée (✓), l'opération est prise en charge.

Référence	Nom du module	Version de firmware	Type de module	Mise à jour du firmware
6GT2 002-0EF00	RF160C CM	V1.0	RFID	✓
6GT2 002-0HD00	RF170C CM	V1.0	RFID	✓
6GT2 002-0HD01	RF170C CM	V3.0	RFID	✓
6GT2 002-0JD00	RF180C CM	V2.0	RFID	✓

8.7.2 MOBY (interface DeviceNet)

Si elle est cochée (✓), l'opération est prise en charge.

Référence	Nom du module	Version de firmware	Type de module	Mise à jour du firmware
6GT2 002-0EB00	Interface MOBY ASM 450	V3.0	IM	✓
6GT2 002-0ED00	MOBY CM ASM 456	V5.0	IM	✓
6GT2 002-0GA10	MOBY CM ASM 475		SM	✓

Index

A

Adresse IP
 configuration requise, 28
 masque de sous-réseau, 33
 paramètre d'appareil, 56
 virtuelle, 32
Afficher les diagnostics de CPU, 96
API (interface d'application)
 AutomationToolAPI.dll, 24
 compatibilité des versions, 125
 conception de l'interface utilisateur, 128
 fichiers et installation, 140
 fonctionnalités de sécurité fournies, 127
 présentation de l'architecture, 137
Appareil
 insertion dans la table Appareil, 55
Assistance, 3
Assistance client, 3
Assistance technique, 3
Assistance technique Siemens, 3

B

Boutons, barre d'outils, 122

C

Classe DataChangedEventArgs (API), 146
Classe Diagnostics (API), 145
Classe EncryptedString (API), 141
Classe IProfinetDeviceCollection (API)
 itération des éléments dans la collection, 157
 Méthode GetEnumerator, 158
 Propriété [], 158
 Propriété count, 158
Classe ProgressChangedEventArgs (API), 146
Classe Result (API), 143
Code couleurs pour les champs de sécurité dans le développement de l'interface utilisateur, 131
 données d'appareil, 133
 dossier programme, 135
 Icône d'appareil pour la CPU, 132
 Mot de passe de la CPU, 134
 mot de passe de programme, 136
Codes de Hamming, 137

Commandes

 afficher les diagnostics de CPU, 96
 clignotement LED, 61
 Data log (chargement ou suppression), 77
 enregistrer/enregistrer sous, 99
 Formater la carte mémoire, 91
 importer/exporter, 100
 Installation de nouveaux firmware, 78
 lecture du tampon de diagnostic, 97
 modification des adresses IP, 56
 modification du nom PROFINET, 58
 ordre d'exécution, 98
 RAZ mémoire, 88
 recette (ajouter, remplacer et supprimer), 73
 Récupérer les données de maintenance, 93
 Régler l'heure, 95
 Restaurer l'appareil depuis un fichier, 83
 restaurer les paramètres d'usine, 90
 RUN/STOP, 60
 Sauvegarder l'appareil dans un fichier, 83
 scan, 49
Configuration dans STEP 7, 27
Configuration de la communication, 30
Configuration de TIA Portal, 27
Configuration nom PROFINET requise, 28
Configuration requise, 27
Consignes de programmation pour l'utilisation de l'API, 128
Constructeur Network (API), 151
CPU
 Configuration IP requise, 28
 Configuration nom PROFINET requise, 28
CriticalInternalErrorException, 241

D

 Data log (chargement ou suppression), 77
 Démarrage de SIMATIC Automation Tool, 27
 DiagnosticsItem, 240
 Données de maintenance, récupération, 93

E

 Éléments de la collection IProfinetDevice (API), 159
 Énumération ConfirmationType (API), 242
 Énumération DataChangedType (API), 242
 Énumération DeviceFamily (API), 242

Énumération d'opérations FailSafe (API), 250
Énumération ErrorCode (API), 243
Énumération FeatureSupport (API), 249
Énumération OperatingState (API), 247
Énumération ProgressAction (API), 248
Énumération ProtectionLevel (API), 250
Énumération RemoteFolderType (API), 250
Énumération RemoteInterfaceType (API), 248
Événement DataChanged (API), 182
Événement ProgressChanged (API), 183
Event Log, 45
Exemple de réseau, 251, 251
Exporter un tableau Appareil vers un fichier .csv, 100
extension de fichier sat, 99

F

fichier csv, 100
Fichier de projet SAT, 99
Filtrer des lignes de table, 44
Firmware
 mise à jour, 78
 non reconnu, 255
Formater la carte mémoire, 91

H

Heure, réglage, 95

I

Icônes de la barre d'outils, 122
Icônes, barre d'outils, 122
ICPU (API)
 drapeaux de fonctionnalités, 192
 Drapeaux relatifs aux mises à jour de programmes, 189
 interface, 187
 propriétés, 188
 restaurer les drapeaux, 191
Identifier l'appareil, 61
IHMI (API)
 drapeaux de fonctionnalités, 232
 drapeaux relatifs aux mises à jour de programmes, 231
 interface, 230
 propriétés, 231
 restaurer les drapeaux, 232
IModule (API)
 interface, 186

IModule (API)
 modules property and IModuleCollection class, 185
Importer un tableau Appareil sous forme de fichier .csv, 100
Informations pour nous contacter, 3
Insérer un appareil, 55
Installation du logiciel, 26
Interface IBaseDevice (API), 150
Interface IHardware (API), 148
Interface IHardwareCollection (API), 150
Interface IModule (API), 149
Interface IModuleCollection (API), 150
Interface IRemoteFile (API), 147
Interface IRemoteFolder (API), 147
Interface IremotelInterface (API), 148
IProfinetDevice (API)
 propriétés, 169

L

Language enumeration (API), 246
Ligne d'assistance, 3
Logiciel de mise à jour, 13

M

Menu
 aide, 121
 fichier, 103
 modifier, 104
 opérations, 105
 Options (communication), 111
 Options (data log), 118
 options (event log), 119
 Options (général), 109
 options (importer/exporter), 120
 Options (mise à jour du firmware), 113
 Options (mise à jour du programme), 114
 Options (projets), 112
 Options (recettes), 117
 Options (sauvegarde/restauration), 116
 Options (Service Data), 115
 outils, 120
Méthode Backup (interface ICPU), 204
Méthode Backup interface IHMI (API), 233
Méthode CheckValidLicense (API), 156
Méthode Clear (API), 168
Méthode COPYUserData (API), 167
Méthode DeleteDataLog (API), 206
Méthode DeleteRecipe (API), 208
Méthode DetermineConfirmationMessage (API), 224

Méthode DownloadRecipe (API), 205
 Méthode FilterByDeviceFamily (API), 159
 Méthode FilterOnlyCPUs (API), 160
 Méthode FindDeviceByIP (API), 161
 Méthode FindDeviceByMAC (API), 162
 Méthode FirmwareUpdate (API), 174
 Méthode FormatMemoryCard (API), 222
 Méthode GetCommunicationsTimeout (API), 155
 Méthode GetCurrentDateTime (API), 210
 Méthode GetDiagnosticsBuffer (API), 211
 Méthode GetEmptyCollection (API), 156
 Méthode Identify (API), 176
 Méthode InsertDeviceByIP (API), 165
 Méthode InsertDeviceByMAC (API), 166
 Méthode MemoryReset (API), 212
 Méthode ProgramUpdate (API), 197
 Méthode ProgramUpdate interface IHMI (API), 234
 Méthode QueryNetworkInterfaceCards (API), 151
 Méthode ReadFromStream (API), 164
 Méthode RefreshStatus (API), 173
 Méthode Remove (API), 168
 Méthode Reset (API), 177
 Méthode ResetToFactoryDefaults (API), 213
 Méthode Restore (interface ICPU), 203
 Méthode Restore interface IHMI (API), 236
 Méthode ScanNetworkDevices (API), 153
 Méthode SetBackupFile (API), 199
 Méthode SetBackupFile interface IHMI (API), 238
 Méthode SetBackupFilePassword (API), 202
 Méthode SetCommunicationsTimeout (API), 154
 Méthode SetCurrentDateTime (API), 216
 Méthode SetCurrentNetworkInterface (API), 152
 Méthode SetIP (API), 178
 Méthode SetOperatingState (API), 215
 Méthode SetPassword (API), 193
 Méthode SetProfinetName (API), 179
 Méthode SetProgramFolder (API), 194
 Méthode SetProgramFolder interface IHMI (API), 237
 Méthode SetProgramPassword (API), 196
 Méthode UploadDataLog (API), 217
 Méthode UploadRecipe (API), 219
 Méthode UploadServiceData (API), 221
 Méthode ValidatePROFINETName (API), 181
 Méthode WriteToStream (API), 163
 Mettre à jour le programme d'un appareil, 63
 Mise à jour du programme, 63
 Mode RUN, 60
 Mode STOP, 60
 Modification du mode, 60
 Modification du mode de fonctionnement, 60
 modification du nom de station PROFINET, 58
 Modifier le mode de fonctionnement, 60

Modules décentralisés, 226
 Mot de passe
 CPU, 37
 Fichier de projet .sat, 99
 Mot de passe :
 CPU,
O
 OperatingStateReq (API), 247
 Opérations liées à la sécurité en conception d'interface utilisateur avec l'API, 127, 128
P
 Présentation Automation Tool, 15
 Prise en charge des appareils
 Alimentation SITOP, 277
 appareils non reconnus, 255
 Appareils RFID, 278
 CPU ET 200pro, 261
 CPU ET 200SP, 263
 CPU S7-1200, 267
 CPU S7-1500, 272
 ET 200AL IM, 256
 ET 200AL SM, IO-Link, 256
 ET 200eco, 257
 ET 200M IM, 258
 ET 200MP IM, 259
 ET 200pro IM, 262
 ET 200pro IO-Link, RFID, 262
 ET 200S, 260
 ET 200SP IM, serveur, 264
 ET 200SP SM, ASi, CM, CP, TM, IO-Link, départs-moteurs, 265
 Interface MOBY, 278
 Pupitres HMI Basic, 274
 Pupitres HMI Comfort, 275
 Pupitres HMI Mobile, 276
 S7-1200 I/O et autres modules, 268
 S7-1500 I/O et autres modules, 273
 Prise en charge des logiciels de protection, 25
 Prise en charge des logiciels VM, 25
 Prise en charge des systèmes d'exploitation, 25
 Programme de sécurité, 124
 Propriété CurrentNetworkInterface (API), 153
 Propriétés IRemoteInterface (API), 226
 Protocole de l'interface réseau, 32

R

Raccourcis clavier, 124, 124
RAZ mémoire, 88
Recettes (ajouter, remplacer et supprimer), 73
Récupérer les données de maintenance, 93
Régler l'heure, 95
RemotelInterfaces (API), 226
Réseau
 accès, 19
 options, 22
Restaurer la mémoire, 88
Restaurer l'appareil, 83
Restaurer les paramètres d'usine, 90

S

Sauvegarder l'appareil, 83
Scanner un réseau, 49
Service et assistance, 3

T

Table Appareil
 scanner un réseau, 49
 vue d'ensemble, 42
Tampon de diagnostic, 96, 97
TIA software updater, 13
Traitement multifil (multithreading), 46, 98