



WebdynSunPM

Note d'application

Power regulation
&
InjectionAndProductionRegulation

Introduction

Cette note d'application décrit comment mettre en œuvre les scripts de régulation de puissance active : **ActivePowerRegulation** et **InjectionAndProductionRegulation**

Ces scripts permettent de commander la puissance active des onduleurs d'un site de production photovoltaïque en fonction d'une ou plusieurs mesures de puissance active afin de respecter une consigne d'injection ou de consommation donnée.

Le script **InjectionAndProductionRegulation** permet en plus de brider la production des onduleurs.

Ces deux scripts permettent également la prise en compte du démarrage de groupes électrogène en coupant la production des onduleurs.

Le seuil de régulation est modifiable en direct par requête d'écriture Modbus ou par commandes FTP ou MQTT ou par requêtes post.



L'utilisation de ces scripts nécessite l'achat d'une licence, merci de vous rapprocher du service commercial Webdyn (<https://www.webdyn.com/contact>) afin d'obtenir la licence appropriée.

Principe de fonctionnement

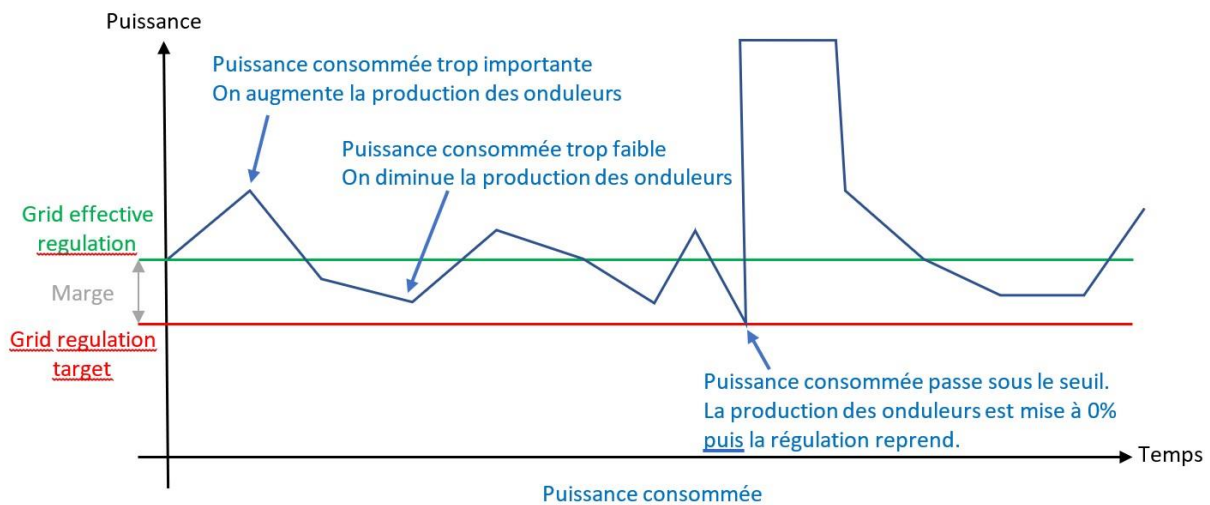
Le script de la WebdynSunPM envoie des commandes de limitation de production de puissance active équitablement réparties entre tous les onduleurs d'une centrale afin de respecter une consigne appelée **Grid regulation target**. Cette consigne peut être définie, aussi bien, comme une consigne de soutirage ou une consigne d'injection définie par le paramètre **Grid regulation type**.

Une boucle de régulation dynamique permet l'adaptation de la commande envoyée aux onduleurs en fonction de leur production courante et de la consommation locale du site de production.

Pour ce faire le script tient compte d'un ou plusieurs compteurs d'énergie reflétant la puissance active au point d'injection et calcul l'écart avec le point de régulation qui correspond à la consigne **Grid regulation target (kW)** à laquelle on ajoute une marge **Grid effective regulation (%)** exprimée en pourcentage de la puissance installée **Total plant solar power (kW)**

Dans le cas d'une régulation dite « zéro injection » il faudra choisir **Grid regulation type** en consommation et **Grid regulation target** à zéro.

Le pourcentage de marge **Grid effective regulation** qu'il faut choisir dépend des variations subies par le réseau du site de production. Un site avec des machines industrielles qui peuvent être arrêtées ou redémarrées fréquemment nécessitera un pourcentage assez important (30%). Alors qu'un site sans variation pourra supporter un pourcentage faible (2%). Par défaut nous préconisons la valeur 5%.



Si la puissance au point d'injection est supérieure au seuil ciblé, on considère que la production photovoltaïque n'est pas suffisante : on augmente la production des onduleurs en conséquence

Si la puissance au point d'injection est inférieure au seuil ciblé, on considère que la production photovoltaïque est trop élevée : on diminue la production des onduleurs en conséquence.

Si la puissance au point d'injection passe sous le seuil **Grid regulation target** alors la régulation photovoltaïque passe à 0%.

La convention de signe utilisée pour les compteurs doit être la suivante : Les valeurs positive indiquent un soutirage et les valeurs négatives une injection.

Le paramètre « **Regulation speed** » permet de limiter l'envoi de commande aux onduleurs afin de permettre au(x) compteur(s) de refléter la commande précédente et ne pas créer un système divergent. Cette latence n'est pas respectée en cas d'urgence (perte de communication avec le compteur ou valeur inférieur au seuil de régulation **Grid regulation target**)

Le script respecte des contraintes de régulation imposant une correction en cas de défaut inférieur à 2s.

Le paramètre « **Phase control** » permet d'affiner la régulation en prenant en compte le détail de la puissance de chaque phase pour les systèmes triphasés. Pour ce faire il faut utiliser la valeur « **Min of the 3 phases** » de ce paramètre. Dans ce cas, le script de la WebdynSunPM tiendra compte de la puissance de la phase la plus proche du seuil de régulation. En cas de déséquilibre entre les phases cela pourra entraîner des réductions de production qui peuvent paraître plus importantes que nécessaire.

L'utilisation de la valeur « **single phase or sum of the 3 phases** » permettra de réduire les pertes de production mais n'empêchera pas l'injection sur une phase ne particulier.

Pour les installations monophasées, le paramètre « **Phase control** » du script doit impérativement être en mode « **single phase or sum of the 3 phases** ».

En cas de perte de communication avec le compteur ou en cas d'arrêt du script, le comportement choisi via le paramètre « **On error** » est appliqué.

Quand le défaut disparaît, le script recommence la régulation.

Lorsqu'une perte de communication avec le compteur est détectée, un fichier d'alarme est instantanément émis vers le portail.



Fonctionnement particulier

Au démarrage du script, une phase transitoire de 60s a lieu, elle correspond au temps moyen de démarrage des onduleurs. Aucune commande n'est envoyée pendant cette période.

Le redémarrage ou la mise à jour de la WebdynSunPM provoque un arrêt puis un redémarrage du script. Qui applique la stratégie définie par le paramètre « **On error** »

Prérequis

La mise à jour de la WebdynSunPM en version de firmware 4.6.5 ou supérieur est nécessaire.

Le script « ActivePowerRegulation » est présent dans la bibliothèque de script de la WebdynSunPM à partir de la version 5.0.10.

Toutefois il peut être récupéré en suivant le lien ci-après et importé via l'interface Web ou le serveur <http://www.webdyn.com/download/ActivePowerRegulation.zip>

Le script « InjectionAndProductionRegulation » est présent dans la bibliothèque de script de la WebdynSunPM à partir de la version 5.0.10.

Toutefois il peut être récupéré en suivant le lien ci-après et importé via l'interface Web ou le serveur <https://www.webdyn.com/download/InjectionAndProductionRegulation.zip>

Une licence spécifique à la WebdynSunPM utilisée est nécessaire

Merci de vous rapprocher du service commercial (<https://www.webdyn.com/contact>) pour l'obtenir, l'identifiant de votre passerelle vous sera demandé.

La connaissance des principes de base du fonctionnement de la WebdynSunPM est fortement recommandée.

Reférez-vous au manuel d'utilisation de la WebdynSunPM pour acquérir les connaissances suivantes :

- Chapitre §3.2.3.2.2.1 Ajouter un équipement page 102
- Chapitre §3.1.2.2.2 Contenu du fichier de définition page 66
- Chapitre §3.2.4.1 Importer un service ou une licence page 151
- Chapitre §3.1.2.1.4 Fichier « <UID>_scl.ini » page 63
- Chapitre §3.2.3.3.4 MQTT page 130
- Chapitre § 5.2.1.1 Format Json du fichier de commandes page 212
- Chapitre § 3.4 Modbus esclave TCP (Modbus slave TCP)
 - § 3.4.2.2 Variables utilisateurs

Les paramétrages décrit ci-dessous sur les fichiers de définition des onduleurs et des compteurs sont déjà réalisés dans la plupart des fichiers intégrés dans la bibliothèque interne de la WebdynSunPM.

Dans de tel cas, l'utilisation du script ne nécessite pas de paramétrage spécifique supplémentaire sur les fichiers de définition.

Paramétrage des compteurs

Il est conseillé d'utiliser des compteurs communiquant via une liaison Ethernet (Modbus TCP) pour la rapidité d'interrogation, et de meilleures performances de la régulation.



Des requêtes sont envoyées en permanence par la WebdynSunPM et peuvent empêcher d'autres équipements de collecter des données sur le compteur (bien qu'il soit en TCP). Il est préférable que le compteur soit dédié à la WebdynSunPM et ne soit pas utilisé par un autre système.

Dans le cas où il serait préféré une communication via une liaison série, il est recommandé de dédier cette interface à la communication exclusive avec le compteur seul.

Dans chaque fichier de définition utilisés par les compteurs connectés au concentrateur, il faut renseigner les éléments suivants :

• Catégorie (identification des équipements)

Dans l'en-tête des fichiers de définition, le champs catégorie (première ligne, 2eme colonne) doit être défini avec le nom « **meter** ». C'est cette dénomination qui permet d'identifier tous les compteurs à prendre en compte par le script.

Généralement un seul compteur fournissant la puissance active au point d'injection est nécessaire.

Toutefois dans certaines conditions il peut être nécessaire de cumuler les données de plusieurs compteurs afin d'obtenir la puissance au point d'injection.

Le script additionne automatiquement les puissances de tous les équipements déclarés avec la catégorie « **meter** ».

Tips : Si un compteur doit être soustrait, dans le cas d'un compteur de production et d'un compteur de consommation par exemple, il suffit de modifier le signe du gain (coef A : colonne H) des variables utilisées dans le fichier du compteur à soustraire.

• Tags (identification des variables)

Dans le(s) fichier(s) de définition des compteurs connectés au concentrateur, il faut identifier les variables de puissance, vérifier qu'elles sont exprimées en kW, ajuster le coefficient A (gain) si besoin, et leur affecter les tags suivants :

- Si le paramètre de script « **Phase control** » vaut « **Min of the 3 phases** »

Puissance active de la phase L1 en kW : tag « **ActivePow1kW** »

Puissance active de la phase L2 en kW : tag « **ActivePow2kW** »

Puissance active de la phase L3 en kW : tag « **ActivePow3kW** »

- Si le paramètre de script « **Phase control** » vaut « **single phase or sum of the 3 phases** ».

Puissance active des 3 phases en kW : tag « **ActivePowSumkW** »

Les « Tag » doivent être renseignés dans la colonne G (champs 7) du fichier de définition de l'équipement. Si besoin référez-vous au manuel du fabricant de l'équipement pour identifier les variables nécessaires.



Les puissances des compteurs doivent être exprimé en **kW**

La convention de signe utilisée pour les compteurs doit être la suivante :

Les valeurs positive indiquent un soutirage et les valeurs négatives une injection.

Exemple :

catégorie « meter »

Tag des variables de puissance instantanées

```
modbusTCP;meter;Janitza;UMG-604-TCP
1;4;19020;F32;;Real power L1-N;ActivePow1kW;0.001000;0.000000;kW;4
2;4;19022;F32;;Real power L2-N;ActivePow2kW;0.001000;0.000000;kW;4
3;4;19024;F32;;Real power L3-N;ActivePow3kW;0.001000;0.000000;kW;4
4;4;19026;F32;;Psum3=P1+P2+P3;ActivePowSumkW;0.001000;0.000000;kW;4
```

Paramétrage des onduleurs

Dans chaque fichier de définition utilisé par les onduleurs connectés au concentrateur, les éléments suivants sont requis :

- **Catégorie (identification des équipements)**

Dans l'en-tête du fichier de définition, le champs catégorie (première ligne, 2eme colonne) doit être défini avec le nom « **Inverter** ». C'est cette dénomination qui permet d'identifier tous les onduleurs à contrôler.

Tips : Pour exclure des onduleurs de la régulation il suffit de définir une autre catégorie dans leur fichier de définition. Cela peut nécessiter la duplication et le renommage du fichier si celui-ci est aussi utilisé par un équipement à piloter.

- **Tags (identification des variables)**

Tous les équipements identifiés par la catégorie « **Inverter** » devront avoir les tags suivants :

Tag « **RealPower** » Permet d'identifier la variable contenant la puissance active instantané. Cette variable n'est pas utilisée directement par l'algorithme de régulation mais elle est utilisée pour tester la communication avec l'onduleur. De plus l'information apparait dans les logs du script (cf chapitre Log de script) et permet de confirmer l'application des commandes de réduction de puissance.

Tag « **cmdPwrPercent** » : Permet d'identifier la variable recevant les commandes de réduction de puissance

Tag « **WMaxLim_Ena** » (optionnel selon les onduleurs) : Permet d'identifier la variable d'activation de la commande de puissance.

Tag « **WMaxLimPct_RvrtTms** » (optionnel selon les onduleurs) permet d'annuler la consigne de rétablissement de la puissance d'origine en l'absence de nouvelle commande

Tag « **WMaxLimPct_RmpTms** » (optionnel selon les onduleurs) permet le réglage des temps d'exécution des consignes de modification de puissance.

Les Tags **cmdOn** et **CmdOff** ne sont nécessaires qu'en cas d'utilisation du paramètre **cmdOnOffINV** (cf paramétrage du script)

Tag « **cmdOn** » Permet d'identifier la variable qui recevra la valeur 1 lors d'une commande de couplage des onduleurs et 0 lors d'une commande de découplage.

Tag « **CmdOff** » : Permet d'identifier la variable qui recevra la valeur 0 lors d'une commande de couplage des onduleurs et 1 lors d'une commande de découplage.

Il n'est pas nécessaire que ces deux tags soient présents. Si l'onduleur ne présente qu'un registre de pilotage il conviendra de lui associer le tag adapté :

Si la logique est « directe » : 1 pour activer et 0 pour désactiver alors le tag « **cmdOn** » sera approprié.

Si la logique est « inverse » : 0 pour activer et 1 pour désactiver alors le tag « **cmdOff** » devra être utilisé.

Le Tag **NominalPower** n'est nécessaire qu'en cas d'utilisation du paramètre **LoadBalancing=0** (cf paramétrage du script)

Le Tag **NominalPower** permet d'identifier le registre qui indique la puissance nominale de l'onduleur.

Les « Tag » doivent être renseignés dans la colonne G (champs 7) du fichier de définition de l'équipement.

Exemple :

categorie « Inverter »

Tag « WMaxLim_Ena » pour activation de la commande

```
modbusRTU; Inverter; Sungrow; Generic
...
109;3;5006;U16;;Power limitation switch; WMaxLim_Ena 1.000000;0.000000;;4
110;3;5007;U16;;Power limitation setting; cmdPwrPercent 0.100000;0.000000;%;4
|...
```

Tag « cmdPwrPercent » pour la commande de puissance

Paramétrage des IO de la WebdynSunPM

Il est possible d'effectuer la configuration des IO de la WebdynSunPM directement depuis l'interface Web

• Digital input

La présence d'un tag « **DIN1** » sur l'une des entrées numériques configurées en lecture d'état (**dry contact**) permet de mettre en « veille » les onduleurs et ainsi prendre en compte le démarrage d'un *groupe électrogène* en réglant la production des onduleurs à 0% tant que ce contact est actif.

Par défaut la fermeture du contact (passage à un) provoque l'arrêt des onduleurs, mais il est possible d'inverser la logique de l'entrée numérique en configurant l'entrée numérique avec un gain à -1 et un offset à 1.

Les tags **DIN2** et **DIN3** peuvent être utilisées afin de permettre une régulation à respectivement 30% du seuil d'origine et 60% du seuil d'origine lorsque le contact correspondant est fermé.

Ces actions n'ont pas d'effet dans le cas d'une régulation configurée à zéro (zéro injection), en revanche si on autorise par exemple une injection de 100kW, la fermeture du contact correspondant à DIN2 limitera l'injection à 30kW et la fermeture du contact correspondant à DIN3 limitera l'injection à 60kW

Si plusieurs contacts sont fermés au même instant, le contact avec la plus forte limitation sera prioritaire.

• Sortie relais

La configuration du tag « **RelayOutput** » lié à la sortie relai de la webdynSunPM permet au script de couper la centrale indépendamment de commandes émises vers les onduleurs, il est nécessaire lorsque :

-le paramètre de script « **onError** » est configuré avec la valeur « **Stop with contactor relay** ».Ce dispositif permet d'assurer l'absence d'injection, en cas de défaillance du système de régulation.

-le paramètre « **CmdONOFF** » est utilisé. Ce paramètre est recommandé lorsque l'on souhaite empêcher l'injection résiduelle des onduleurs avec une commande à zéro, ou lorsque la commande de contrôle de l'onduleur n'est pas assez rapide.

Le pilotage d'un contacteur de puissance type « *Schneider Electric LC1D115004P7* » permettant la coupure de la production nécessite l'utilisation d'un relai intermédiaire type Finder 55.32.9.024.0000, le relai interne de la WebdynSunPM n'ayant pas un pouvoir de coupure suffisant pour assurer une commande directe.

Exemple :

```
1 io;Io;Webdyn;ioSunPM;;;
2 0;2;1;1;;digital1;DIN1;1;0;;4
3 1;2;2;1;;digital2;DIN2;1;0;;4
4 2;2;3;1;;digital3;DIN3;1;0;;4
5 3;1;1;1;;analog1;;1;0;;4
6 4;1;2;1;;analog2;;1;0;;4
7 5;1;3;1;;analog3;;1;0;;4
8 6;1;4;1;;analog4;;1;0;;4
9 7;3;1;;;output;RelayOutput;1;0;;4
```

Tag pour la surveillance de contact sec

Tag « RelayOutput » pour le pilotage du relais

Script

Chargement du script et de la licence

Le script est présent dans la bibliothèque de script de la WebdynSunPM à partir de la version 5.0.10. Toutefois il peut être récupéré via le lien suivant :

<http://www.webdyn.com/download/ActivePowerRegulation.zip>

Depuis la page **control** vous pouvez charger le script en cliquant sur le bouton « *Add script/licence file* »

The screenshot shows the 'Services' page in the Webdyn interface. A table lists several services, with the 'ActivePowerRegulation' service highlighted. The 'License' column for this service shows 'Missing/Invalid'. A red box highlights the 'Add script/licence file' button at the bottom right of the table. Below the table, a modal window titled 'Add script/licence file' is open, showing a 'Choose file' section with a 'Script or licence file' button and 'Cancel' and 'Add' buttons.

Name	Description	Version	License	Status
ActivePowerRegulation	Active power regulation	6.0	Missing/Invalid	Disabled
Decouplage	Decouplage	8	Missing/Invalid	Disabled
GenSet-V1_04	Generator	1.04	Missing/Invalid	Disabled
LocalDisplay	Local Display	8	Not required	Disabled
RelayControl	Relay Control	2.0	Not required	Disabled
SendCommand	Send Command	1.0	Not required	Disabled

Vérification de l'intégration de la licence :

Si la licence n'est pas chargée dans le produit le message « Missing/Invalid » apparaît dans la colonne License.

Ajoutez la licence en cliquant sur le bouton « *Add script/licence file* »

The screenshot shows the 'Services' page in the Webdyn interface. The 'ActivePowerRegulation' service is now shown with an 'Active' license, highlighted by a red box. The 'Status' column shows 'Disabled'.

Name	Description	Version	License	Status
ActivePowerRegulation	Active power regulation	6.0	Active	Disabled

Le champ « License » doit indiquer « Active »

Paramétrage du script et démarrage depuis l'interface Web :

Cliquer sur les 3 points en fin de ligne pour accéder au menu étendu puis cliquez sur **Script arg**

Name	Description	Version	License	Status	
ActivePowerRegulation	Active power regulation	6.0	Missing/Invalid	Disabled	⋮
Decouplage	Decouplage	8	Missing/Invalid	D	⋮
GenSet-V1_04	Generator	1.04	Missing/Invalid	D	⋮
LocalDisplay	Local Display	8	Not required	Disabled	⋮

ActivePower

Total plant solar power (kW): 200

Grid regulation type: Consumption

Grid regulation target (kW): 0

Grid effective regulation (%): 5

Regulation speed (s): 5

Phase control: Single phase or sum of the 3 phases

On error: None

Buttons: Compute speed, Cancel, Save

Un point de régulation est calculé à partir des paramètres renseignés.

Il correspond à la consigne **Grid regulation target (kW)** à laquelle on ajoute une marge **Grid effective regulation (%)** exprimée en pourcentage de la puissance installée **Total plant solar power (kW)** (voir le paragraphe *principe de fonctionnement plus haut*)

-Le paramètre « **Grid regulation type** » permet de choisir une consigne d'injection ou de consommation car la valeur du paramètre **Grid regulation target (kW)** doit toujours être positif.

Le paramètre « **Regulation speed** » permet de limiter l'envoi de commande aux onduleurs afin de permettre au(x) compteur(s) de refléter la commande précédente et ne pas créer un système divergent. Cette latence n'est pas respectée en cas d'urgence (perte de communication avec le compteur ou valeur inférieur au seuil de régulation **Grid regulation target**)

-Le paramètre « **Phase control** » permet de choisir si la surveillance d'injection s'effectue sur la somme des 3 phases (ou sur une seule phase dans le cas d'une installation monophasée). « **single phase or sum of the 3 phases** » ou sur chaque phase indépendamment: « **Min of the 3 phases** »

-Le paramètre « **On error** » permet de choisir le comportement du script lors de l'arrêt de la WebdynSunPM (perte d'alimentation, mise à jour) ou lors de la perte de communication avec le compteur.


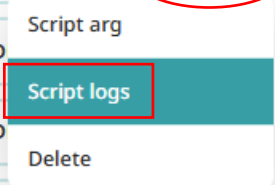
Il est possible de n'effectuer aucune action ; « **none** » ou de forcer un pourcentage prédéfini pour les onduleurs : « **set inverter to** » suivi d'une valeur dans le champ approprié.

On error	Set Percentage
Set inverters to (%) ▼	100

Nous recommandons l'utilisation de ce dernier avec la valeur zéro, c'est la configuration la plus sûre pour prévenir l'injection lorsqu'il n'y a pas de contacteur installé sur site.

Finalement il est possible de choisir le paramètre « **stop with contactor relay** », dans ce cas la configuration du tag du relai est impérative et un contacteur de puissance type « *Schneider Electric LC1D115004P7* » permettant la coupure de la production doit être installé, il nécessite l'utilisation d'un relai intermédiaire type Finder 55.32.9.024.0000, le relai interne de la WebdynSunPM n'ayant pas un pouvoir de coupure suffisant pour assurer une telle fonctionnalité.

Après avoir activé le script au moyen du bouton en fin de ligne vous pouvez accéder au log du script :


Name	Description	Version	License	Status
ActivePowerRegulation	Active power regulation	6.0	Active	Enabled 
Decouplage	Decouplage	8	Missing/Invalid	D 
GenSet-V1_04	Generator	1.04	Missing/Invalid	D

```

Logs
2024-10-31 10:53:19 [ActivePowerRegulation.luaw 96] ACTIVE CONTROL V6.0 started
2024-10-31 10:53:19 [ActivePowerRegulation.luaw 170] missing tag DIN1
2024-10-31 10:53:19 [ActivePowerRegulation.luaw 189] missing tag RelayOutput
2024-10-31 10:53:19 [ActivePowerRegulation.luaw 224] 3 inverters found
2024-10-31 10:53:19 [ActivePowerRegulation.luaw 239] Inverter 1(INV2) has tag: cmdPwrPercent
2024-10-31 10:53:19 [ActivePowerRegulation.luaw 254] Inverter 1(INV2) has tag: NominalPower
2024-10-31 10:53:19 [ActivePowerRegulation.luaw 270] Inverter 1(INV2) has tag: RealPower
2024-10-31 10:53:19 [ActivePowerRegulation.luaw 239] Inverter 2(INV3) has tag: cmdPwrPercent
2024-10-31 10:53:19 [ActivePowerRegulation.luaw 254] Inverter 2(INV3) has tag: NominalPower
2024-10-31 10:53:19 [ActivePowerRegulation.luaw 270] Inverter 2(INV3) has tag: RealPower
2024-10-31 10:53:19 [ActivePowerRegulation.luaw 239] Inverter 0(INV1) has tag: cmdPwrPercent
2024-10-31 10:53:19 [ActivePowerRegulation.luaw 254] Inverter 0(INV1) has tag: NominalPower
2024-10-31 10:53:19 [ActivePowerRegulation.luaw 270] Inverter 0(INV1) has tag: RealPower
2024-10-31 10:53:19 [ActivePowerRegulation.luaw 311] Error reading Nominal Power on inverter 1
2024-10-31 10:53:19 [ActivePowerRegulation.luaw 311] Error reading Nominal Power on inverter 2
2024-10-31 10:53:19 [ActivePowerRegulation.luaw 311] Error reading Nominal Power on inverter 3
2024-10-31 10:53:19 [ActivePowerRegulation.luaw 319] Warning: nominal power :(calc Vs Declared) 0 / 200.0
2024-10-31 10:53:19 [ActivePowerRegulation.luaw 350] 1 meters found
2024-10-31 10:53:19 [ActivePowerRegulation.luaw 375] Meter 0(Meter1) has tag: ActivePowSumkW
2024-10-31 10:53:19 [ActivePowerRegulation.luaw 148] Power control initialized
2024-10-31 10:53:19 [ActivePowerRegulation.luaw 149] SolarRatedPowerkW200.0
2024-10-31 10:53:19 [ActivePowerRegulation.luaw 150] Regulation limit -0.0 kW
2024-10-31 10:53:19 [ActivePowerRegulation.luaw 151] Regulation target 10.0 kW (5.0 %)
2024-10-31 10:53:19 [ActivePowerRegulation.luaw 152] Regulation speed 5.0
2024-10-31 10:53:19 [ActivePowerRegulation.luaw 351] Meter 1 reading fail
  
```

Au démarrage du script, le fichier de log recapitule les équipements trouvés et les tags présents dans les fichiers associés à ces équipements.

Si l'un des tags est manquant le script ne démarrera pas et le fichier de log indiquera les tags manquants :

Name	Description	Version	License	Status
ActivePowerRegulation	Active power regulation	6.0	Active	Error 

<p>2024-10-31 13:09:41 [ActivePowerRegulation.luaw 96] ACTIVE CONTROL V6.0 started</p> <p>2024-10-31 13:09:41 [ActivePowerRegulation.luaw 224] 3 inverters found</p> <p>2024-10-31 13:09:41 [ActivePowerRegulation.luaw 239] Inverter 1(INV2) has tag: cmdPwrPercent</p> <p>2024-10-31 13:09:41 [ActivePowerRegulation.luaw 254] Inverter 1(INV2) has tag: NominalPower</p> <p>2024-10-31 13:09:41 [ActivePowerRegulation.luaw 273] inverter 1(INV2) missing tag: RealPower</p> <p>2024-10-31 13:09:41 [ActivePowerRegulation.luaw 276] Load balancing mandatory</p> <p>2024-10-31 13:09:41 [ActivePowerRegulation.luaw 239] Inverter 2(INV3) has tag: cmdPwrPercent</p> <p>2024-10-31 13:09:41 [ActivePowerRegulation.luaw 254] Inverter 2(INV3) has tag: NominalPower</p> <p>2024-10-31 13:09:41 [ActivePowerRegulation.luaw 273] inverter 2(INV3) missing tag: RealPower</p> <p>2024-10-31 13:09:41 [ActivePowerRegulation.luaw 276] Load balancing mandatory</p> <p>2024-10-31 13:09:41 [ActivePowerRegulation.luaw 239] Inverter 0(INV1) has tag: cmdPwrPercent</p> <p>2024-10-31 13:09:41 [ActivePowerRegulation.luaw 254] Inverter 0(INV1) has tag: NominalPower</p> <p>2024-10-31 13:09:41 [ActivePowerRegulation.luaw 273] inverter 0(INV1) missing tag: RealPower</p> <p>2024-10-31 13:09:41 [ActivePowerRegulation.luaw 276] Load balancing mandatory</p> <p>2024-10-31 13:09:41 [ActivePowerRegulation.luaw 311] Error reading Nominal Power on inverter 1</p> <p>2024-10-31 13:09:41 [ActivePowerRegulation.luaw 311] Error reading Nominal Power on inverter 2</p> <p>2024-10-31 13:09:41 [ActivePowerRegulation.luaw 311] Error reading Nominal Power on inverter 3</p> <p>2024-10-31 13:09:41 [ActivePowerRegulation.luaw 319] Warning: nominal power :(calc Vs Declared) 0 / 200.0</p> <p>2024-10-31 13:09:41 [ActivePowerRegulation.luaw 350] 1 meters found</p> <p>2024-10-31 13:09:41 [ActivePowerRegulation.luaw 375] Meter 0(Meter1) has tag: ActivePowSumkW</p> <p>2024-10-31 13:09:41 [ActivePowerRegulation.luaw 426] Config error</p> <p>2024-10-31 13:09:41 [ActivePowerRegulation.luaw 427] STOP</p>	
--	--



En cours de fonctionnement la perte de communication avec le compteur entraine l'application de la stratégie définie par le paramètre « **on error** » et une connexion vers le serveur distant est initié afin de déposer un fichier alarme.

Paramétrage et démarrage du script depuis le serveur distant

Depuis le serveur distant, le fichier « <uid>_scl.ini » permet la configuration et l'activation des scripts, il est présent dans le répertoire /Config

Le paramètre **SCRIPT_Enable[n]** indique l'état de fonctionnement et permet l'activation (=1) et la désactivation (=0) du script identifié par le paramètre **SCRIPT_File[n]** qui vaut ActivePowerRegulation.luaw dans ce cas.

Le paramètre **SCRIPT_Args[n]** utilise par défaut les mêmes éléments de configuration que ceux décrit dans l'interface Web, auxquelles peuvent s'ajouter les paramètres optionnels décrit ci-après.

Exemple de fichier.UID_scl.ini

```
SCRIPT_Args[n] = {"solarRatedPowerKW": 200,"gridRegulationType": "injection","gridRegulationTargetKW": 0,"gridEffectiveRegulationPercent": 5,"regulationSpeedS": 5,"phaseControl": "sum","errorAction": "none","setToPercent": 100,"DataFreq": 0,"MeterType": 1,"maxGetxCount": 2,"EnableConfirm": 0,"cmdOnOff": 0,"cmdOnOffINV": 0,"AlarmOn": 1,"AlarmDelayed": "false","InvErrorAction": 0,"repeatCmdFreq": 120,"TargetCriticity": 0,"LoadBalancing": 1}
```

Certains paramètres ne sont pas disponibles depuis la page web de configuration du script.

Pour modifier les paramètres suivants, il est nécessaire d'accéder au fichier de configuration des scripts « <uid>_scl.ini » sur le serveur.

DataFreq : Ce paramètre permet d'activer (>0) ou de désactiver (=0) l'écriture des variables du script dans le fichier de donnée émis vers le serveur. La liste des données disponible est décrite dans le fichier de définition du script qui est considéré comme un équipement virtuel. (cf § suivant : principe de fonctionnement)

Lorsque l'option est activée, la valeur du paramètre permet de définir la période d'acquisition données. Par défaut se paramètre est à 0 (aucune donnée émise).

MeterType : Ce paramètre permet de modifier la source de la mesure de puissance qui sert de référence à la régulation.

0 (historique) : Les mesures sont relevées sur un compteur unique déclaré avec le nom « main meter »

1 (par défaut) : Les mesures sont relevées via l'ensemble des compteurs déclaré avec la catégorie « Meter ».

2 (usage alternatif): Les mesures sont émises par un équipement tiers type GTB sur une variable de l'équipement virtuel associé au script.

MaxGetxCOUNT : Ce paramètre permet de définir le nombre de communication en échec avec un équipement avant de considérer qu'il est défaillant. Il est défini à 2 par défaut, un nombre plus important entrainera des délais dans la prise en compte d'un défaut de communication. Une valeur trop faible peut provoquer des coupures intempestives de la centrale si une perte de communication intempestive est signalée.

EnableConfirm : Ce paramètre permet d'activer l'envoi d'une commande d'acquiescement de la commande de régulation précédemment émise. Ce comportement est parfois attendu par certains onduleurs.

CmdOnOffINV : Le paramètre cmdOnOffINV lorsqu'il est activé (=1) permet au script de piloter le relai interne des onduleurs. ayant les tags nécessaires (CmdOn/CmdOff) en complément d'une mise à zéro de la consigne de régulation. Cette fonctionnalité permet un arrêt plus rapide et évite les productions résiduelles des onduleur concernées.

Attention : Un temps de synchronisation avec le réseau est nécessaire lors de la reprise de la production (généralement 60s) automatiquement géré par le script.

CmdOnOff : Le paramètre cmdOnOff permet au script d'agir sur le relai interne de la WebdynSun PM en complément d'une mise à zéro de la consigne de régulation lorsque la puissance au point d'injection passe sous le seuil **Grid regulation target** Cette fonctionnalité permet un arrêt plus rapide et évite les productions résiduelles des onduleur concernées. Attention : Les onduleurs étant coupé suite à l'activation du relai, un temps de resynchronisation avec le réseau est nécessaire lors de la reprise de la production (généralement 60s) automatiquement géré par le script.

AlarmOn, : Ce paramètre permet d'activer ou désactiver l'émission de fichier d'alarmes vers le répertoire correspondant du serveur

En cas d'activation la valeur 1 limite l'envoi à l'alarme relative à la perte de communication avec le compteur, la valeur 2 autorise l'envoi de toutes les alarmes possibles : perte de communication avec le compteur, erreur de paramétrages, arrêt et activation du script lors des pertes d'alimentation et autres causes de reboot de l'équipement

AlarmDelayed : Lorsque le paramètre vaut :

« true » les alarmes sont émises uniquement lors des connexions programmées vers le serveur

« false ». les alarmes sont émises immédiatement, L'apparition d'une alarme provoque une connexion immédiate vers le serveur. Par défaut cette variable est à false.

InvErrorAction : Ce paramètre vient en complément du paramètre OnError, la stratégie définie par ce dernier ne s'applique qu'en cas de perte de communication avec le compteur (InvErrorAction=0) Lorsque InvErrorAction vaut 1, un défaut de communication sur les onduleurs aura les mêmes conséquences qu'une

perte de communication avec le compteur. Ce paramètre est surtout recommandé lorsqu'il n'y a qu'un onduleur sur le site dans le cas contraire les autres onduleurs vont compenser l'absence de communication avec l'onduleur défaillant

repeatCmdFreq : Ce paramètre permet de définir la période de répétition de la commande de mise à zéro ou à 100% lorsque la fonction de régulation est arrêtée par l'utilisation de l'IO DIN1 ou par la variable CmdStop. Par défaut ce paramètre est défini à 60s.

TargetCriticality : Ce paramètre permet de modifier le comportement du script lorsque la valeur de référence passe sous la valeur seuil critique : **Grid regulation target**

0 :criticality low : Aucune action spécifique le comportement de régulation habituel s'applique

1 criticality medium : La fréquence de mise à jour de la commande est augmentée, la nouvelle commande s'applique toute les secondes, le paramètre **regulationSpeed** n'est plus utilisé.

2 criticality high : C'est le comportement par défaut, si la valeur du compteur n'est pas correcte après la première tentative de régulation, les onduleurs sont mis à 0% et la régulation reprend normalement.

LoadBalancing : Ce paramètre modifie le comportement du script afin d'optimiser les temps de réaction. Lorsqu'il est activé (=1 valeur par défaut) les variations de puissance sont en permanence équitablement répartie sur l'ensemble des onduleurs ainsi tous les onduleurs on en permanence le même pourcentage de commande de puissance.

Lorsqu'il est désactivé (=0) toute nouvelle commande tient compte de la production actuelle de chaque onduleur ainsi les niveaux de production de tous les onduleurs sont différents. Cette option nécessite la présence du tag NominalPower pour identifier la puissance nominale de chaque onduleur.

Paramètres spécifiques au script **InjectionAndProductionRegulation**

ProductionLimit : Ce paramètre permet de définir la valeur de bridage souhaité pour la production maximale des onduleurs.

DynLimit : Ce paramètre permet de choisir le type de bridage souhaité

- Si le paramètre vaut zéro, le bridage est statique et un pourcentage maximum est calculé en fonction de la puissance de la centrale et du paramètre précédent **ProductionLimit**. Ce pourcentage fixe la limite maximum qui peut être envoyée aux onduleurs
- Si le paramètre vaut un, le bridage est dynamique et la valeur limite de production pourra évoluer en fonction de la somme des productions déclarées par chaque onduleur.

Comme pour la régulation d'injection, dans le cas d'un bridage dynamique de la production, une marge équivalente à celle de la régulation d'injection est appliquée. Cette marge qui correspond à un pourcentage de la capacité de la centrale (**Grid effective regulation**) est retirée de la valeur du paramètre **ProductionLimit**. Par exemple, sur une centrale de 300KW, avec un paramètre Grid effective regulation de 5% et une limite de production de 250KW, la régulation de production se fera autour de $250 - 15 = 235KW$.

Description des paramètres du script :

Nom du paramètre du script dans l'interface web	Nom du paramètre dans le fichier « <uid>_scl.ini »	Description	Type	Valeur par défaut
Total plant solar power	solarRatedPowerKW	Puissance maximum en kW de la centrale solaire	Nombre entier positif	200
Grid regulation type	gridRegulationType	Type de régulation : <ul style="list-style-type: none"> • injection : La valeur cible de la régulation (gridRegulationTargetKW) est une valeur d'injection (positive) : voir ci-dessous. • consumption : La valeur cible de la régulation (gridRegulationTargetKW) est une valeur de consommation (positive) : voir ci-dessous. 	Liste : <ul style="list-style-type: none"> • injection • consumption 	injection
Grid regulation target	gridRegulationTargetKW	Valeur cible en kW pour la régulation, le fonctionnement de cette valeur dépend du type de régulation (gridRegulationType) <ul style="list-style-type: none"> • injection : La valeur cible de la régulation (gridRegulationTargetKW) représente la valeur maximale qu'on s'autorise à injecter. • consumption : La valeur cible de la regulation (gridRegulationTargetKW) est la valeur minimale qu'on s'autorise à consommer. 	Nombre entier positif	0
Grid effective regulation	gridEffectiveRegulationPercent	en % de la puissance solaire installée, cette valeur permet de calculer le point de fonctionnement de la régulation. Cette information indique quelle marge est calculée par rapport à la cible (top margin) et quelle est la valeur effective de la régulation.	Nombre entier positif	5
Regulation speed	regulationSpeedS	Temps de chaque pas en seconde de la gestion de la régulation	Nombre entier positif	5

Phase control	phaseControl	<p>La gestion de la régulation peut être faite de 2 manières différentes :</p> <ul style="list-style-type: none"> • Single phase or sum of the 3 phases (sum) : Sur l'ensemble des phases en triphasé ou sur une phase en monophasé. • Min of the 3 phases (min) : Sur la phase la plus faible (possible uniquement pour une installation triphasée) 	<p>Liste :</p> <ul style="list-style-type: none"> • sum • min 	sum
On error	errorAction	<p>En cas d'erreur sur les équipements ou arrêt du script, on peut choisir 3 scénarios :</p> <ul style="list-style-type: none"> • none : Régulation actuelle • Set inverters to (setTo) : Régulation en pourcentage par rapport à une valeur renseignée dans le paramètre « setToPercent » • Stop with contactor relay (stop) : Ouverture du relais (paramétrer le relais avec le tag « RelayOutput ») 	<p>Liste :</p> <ul style="list-style-type: none"> • none • setTo • stop 	none
	setToPercent	<p>Pourcentage de la puissance désirée en cas d'erreur. (Uniquement si le paramètre « errorAction » est en mode « setTo »)</p>	<p>Nombre entier positif</p>	100
DataFreq		<p>Permet de définir la fréquence d'enregistrement des données de l'équipement virtuel dans le fichier transmis au serveur (si 0 : pas de donnée transmises)</p>	<p>Nombre entier positif</p>	0
MeterType		<p>MeterType désigne la source de référence de mesure de la puissance active.</p>	<p>List:</p> <ul style="list-style-type: none"> 0 : MainMeter 1 : Meter category 2 : Modbus Slave 	1
maxGetxCount		<p>maxGetxCount permet de définir le nombre d'échec de communication avec un équipement qui déclenche un défaut.</p>	<p>Nombre entier positif</p>	2

EnableConfirm		Envoi une commande d'acquiescement permettant la prise en compte de la commande de régulation de puissance.	Nombre entier positif	0
cmdOnOff		Lorsque la consigne 0% est émise, le relai interne de la PM est utilisé pour assurer la non-production.	Liste : <ul style="list-style-type: none"> 0 fonction désactivée 1 fonction activée 	0
cmdOnOffINV		Lorsque la consigne 0% est émise, les relais interne des onduleurs sont utilisés pour assurer la non-production.	Liste : <ul style="list-style-type: none"> 0 fonction désactivée 1 fonction activée 	0
AlarmOn		Permet de configurer la stratégie d'émission des alarmes <ul style="list-style-type: none"> 	Liste : 0 : pas d'alarme 1 : Alarme perte de com compteur uniquement 2 : toutes	1
AlarmDelayed		Par défaut (false), les alarmes sont émises instantanément. Si le paramètre vaut « true » les alarmes ne seront émises qu'à la prochaine connexion programmée vers le serveur.	Booléen	false
InvErrorAction		Permet d'appliquer en cas de défaut de com avec un onduleur, la même stratégie qu'en cas de défaut de com avec le compteur. Sinon les défaut de com onduleurs sont ignorés.	Liste : 0 : inactif 1 : actif	0
repeatCmdFreq		Définit la période de répétition de la commande de mise à zéro dans les modes ou le script est en pause. DIN1 activé ou cmdStop à 1.	Nombre entier positif	120
TargetCriticity		Définit le comportement du script lorsque le seuil de régulation Grid regulation target est atteint	Liste : 0 : aucune action spécifique 1 : commandes émises toute les secondes 2 : une commande corrective avant mise à zéro.	2

LoadBalancing		Defini la strategie de répartition des variations de puissance sur les onduleurs.	Liste: 1 : puissance équitablement répartie sur l'ensemble des onduleurs. 0 : commande différentielle tient compte de la production actuelle de chaque onduleur	1
ProductionLimit		Défini la valeur de bridage souhaité pour la production maximale des onduleurs.	Nombre entier positif	
DynLimit		Permet de choisir le type de bridage de la production.	Liste : 0 : bridage statique 1 : bridage dynamique	

Principe de fonctionnement

Après le démarrage, le script met à disposition une série de variables qui sont décrites dans le fichier « **WPMXXXXX_Script_ActivePowerRegulation.csv** » déposé dans le répertoire /DEF du serveur à la première connexion suivant son activation.

none	Script	Script	ActivePowerRegulation						
1			U8	Alarm	Alarm	1.000000	0.000000		4
2			U8	CmdStop	CmdStop	1.000000	0.000000		4
3			I32	SetTarget	SetTarget	1.000000	0.000000		4
4			U32	TotInvPower	TotInvPower	1.000000	0.000000		4
5			U32	Watchdog	Watchdog	1.000000	0.000000		4
6			U16	inverterCommand	inverterCommand	1.000000	0.000000		4
7			I16	meterValueKW	meterValueKW	1.000000	0.000000		4
8			I16	variation	variation	1.000000	0.000000		4

Il décrit les variables du script au même titre que les autres fichiers de ce répertoire décrivent les variables accessibles sur un équipements physique. Dans le cas du script on parle d'équipement virtuel.

Ces variables peuvent être intégrées aux fichiers de données émis vers le serveur en utilisant le paramètre **DataFreq**

La variable **Alarm** informe sur l'état de fonctionnement du script et de l'installation en remontant des codes correspondant aux divers défauts observés

TABLEAU DES CODES ALARMES :

- 0 Pas d'erreur
- 1 Configuration erronée
- 2 Perte de com avec le compteur
- 3 Perte de com avec un onduleur (seulement si paramètre InvErrorAction=1)
- 100 Initialisation

La variables **CmdStop** permet de piloter le fonctionnement du script.

L'écriture de la valeur 1 permet d'arrêter la régulation en mettant les onduleurs à 0% de production

L'écriture de la valeur 100 permet d'arrêter la régulation en mettant les onduleurs à 100% de production

La variable **SetTarget** permet de modifier la consigne de régulation, elle reçoit des puissances en KW positive pour un seuil de consommation ou négative pour un seuil d'injection.

La variable **TotInvPower** contient la somme des productions des onduleurs.

La variable **Watchdog** permet de surveiller la communication Modbus avec une GTC lorsque celle-ci fourni la valeur de reference de regulation en lieu et place du meter. La GTC doit modifier la variable **Watchdog** très régulièrement (<5s) afin de confirmer la bonne communication. (cf fonctionnement alternatif)

La variable **InverterCommand** met à disposition la dernière commande émise vers les onduleurs (en %)

La variable **meterValueKW** met à disposition la dernière valeur de référence lue sur le meter.

Alternativement elle peut recevoir la valeur de référence transmise par la GTC (cf fonctionnement alternatif).

La variable **variation** met à disposition met à disposition la valeur de la variation lors de la dernière commande emise vers les onduleurs.

Pilotage de la centrale à distance

La variable **SetTarget** précédemment décrite permet de modifier à distance la variable **Grid regulation target** sans modification de la configuration et donc sans redémarrage du script.

Pour ce faire il faut utiliser la fonction associée : **SetGridRegulationTarget**

Cette fonction peut être appelée par tous les moyens mis à disposition sur la WebdynSunPM

- Fichier de commande via le répertoire /CMD du serveur FTP
- Requêtes via le topic « command » du serveur MQTT
- Requêtes POST http

En retour le script répond avec la valeur effective de regulation qui tient compte de la puissance de la centrale et du pourcentage de marge : **Grid effective regulation (%)**

Pilotage par envoi de commande via un serveur FTP

En déposant le fichier de commande suivant sur le serveur FTP dans le répertoire /CMD.

Le fichier de commande doit avoir le format suivant :

```
<uid>_cmd.json
```

Ci-dessous la modification de la variable SetTarget afin de fixer le seuil à 50KW d'injection :

```
{  
  "rpcName": "ActivePowerRegulation.SetGridRegulationTarget",  
  "parameters": -50,  
  "callerId": "1"  
}
```



- L'utilisation de la commande via un serveur FTP ne permet pas une application immédiate de l'ordre émis. La prise en compte de la commande est dépendante de la fréquence de connexion au serveur. Pour une prise en compte immédiate il faut privilégier les requêtes post ou le serveur MQTT. Une connexion permanente est nécessaire pour ces types d'utilisation.

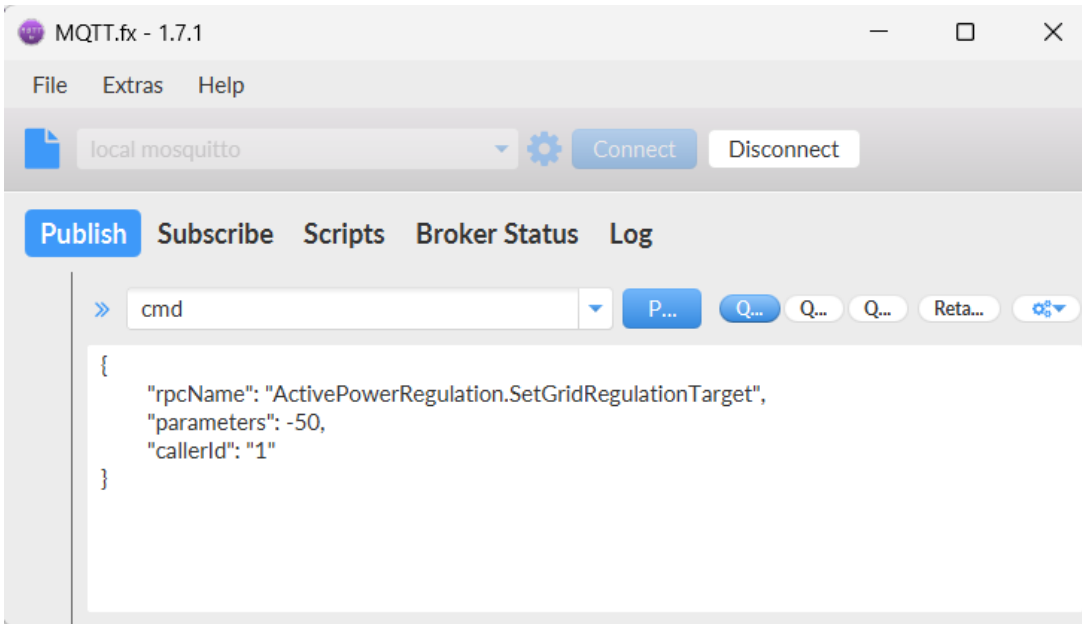
Pilotage par envoie de commande via un serveur MQTT

Pour configurer la connexion de la WebdynSunPM vers un serveur MQTT referez vous au manuel d'utilisateur §3.2.3.3.4 MQTT page 130. La connexion MQTT ne permettant pas la supervision de la configuration du datalogger, seul le serveur 2 peut etre configuré avec ce type de connexion.

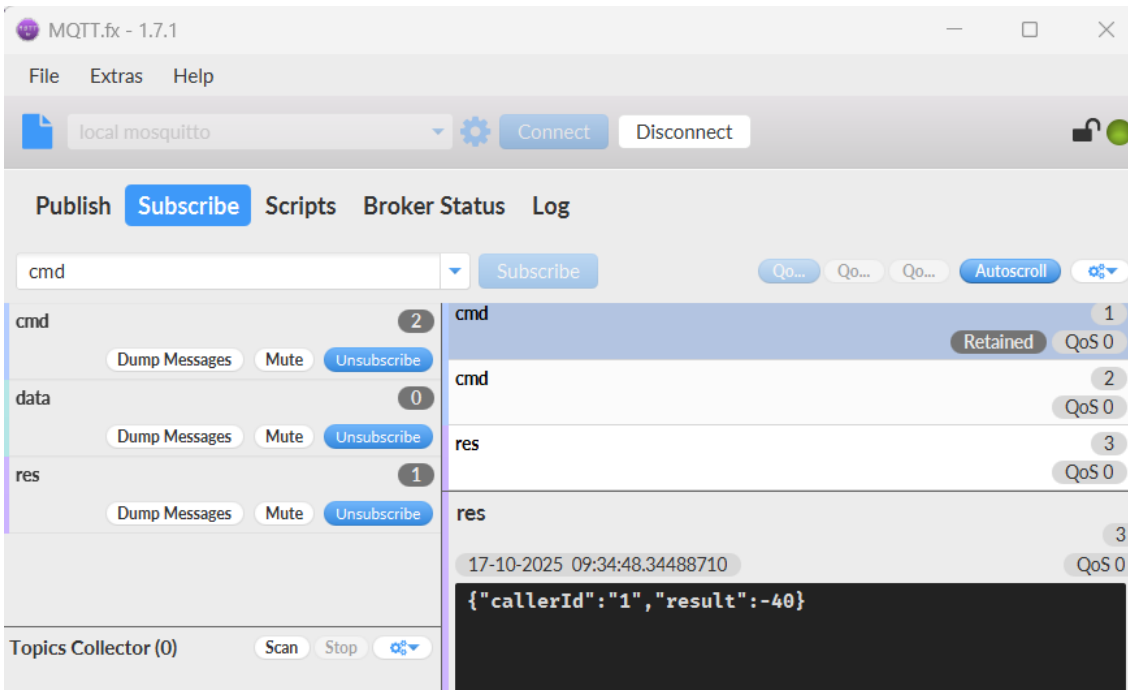
Assurez vous d'avoir definit un **Commandtopic** et un **ResultTopic** afin de permettre la reception de commandes et l'envoi du resultat.

La commandes peut etre transmises via un logiciel permettant de souscrire a un broker MQTT type MQTTfx (<https://www.softblade.de/download/>).

Ci-dessous la modification de la variable SetTarget afin de fixer le seuil à 50KW d'injection :



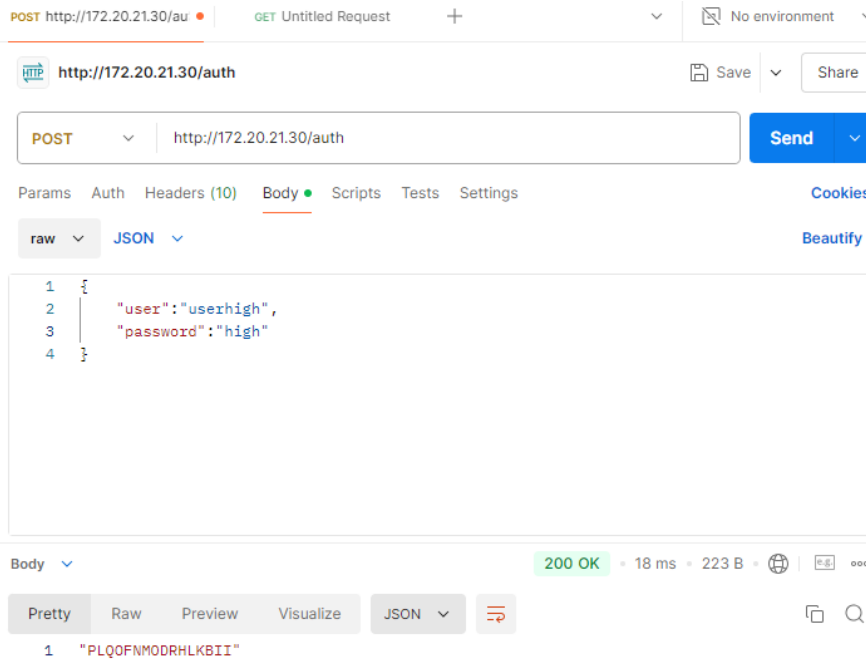
En retour, le résultat de la commande indique que la valeur effective de régulation sera de 40Kw d'injection (5% d'une centrale de 200KW)



Pilotage par envoi d'une requête POST

Avant d'envoyer une commande post vers la WebdynSunPM, il est nécessaire d'ouvrir une session en envoyant une requête d'authentification comme suit en utilisant l'API Postman par exemple (<https://web.postman.co/>).

Ci-dessous la commande d'ouverture de session :

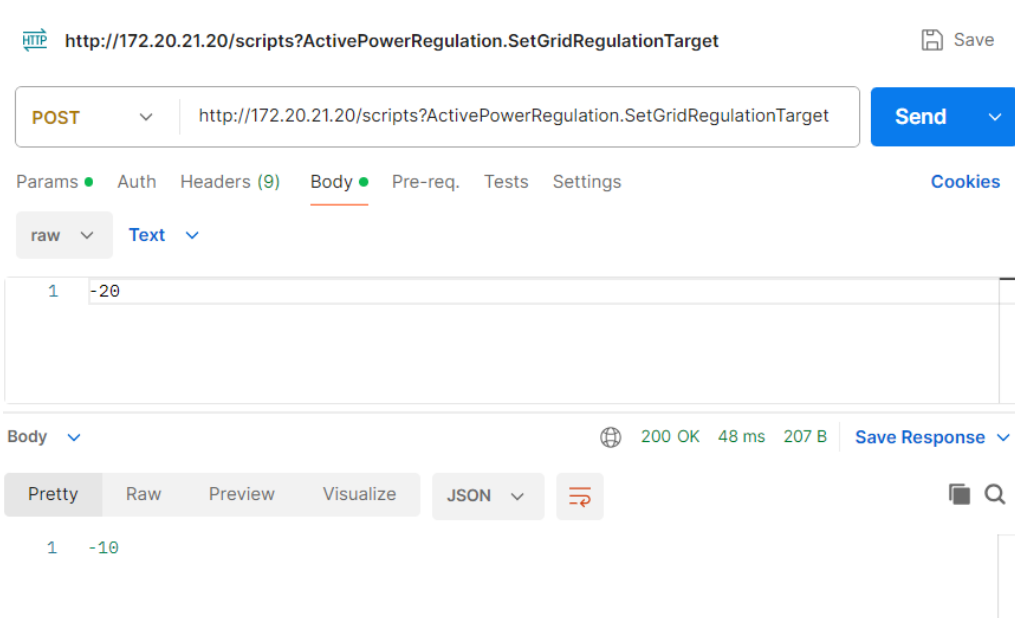


The screenshot shows a Postman interface for a POST request to `http://172.20.21.30/auth`. The request body is a JSON object:

```
1 {
2   "user": "userhigh",
3   "password": "high"
4 }
```

The response is `200 OK` with a status of `18 ms` and `223 B`. The response body is displayed as `"PLQOFNMODRHLKBII"`.

Après authentification vous pourrez alors appeler les commandes précédemment décrites. Ci-dessous la modification de la variable `SetTarget` afin de fixer le seuil à 20KW d'injection :



The screenshot shows a Postman interface for a POST request to `http://172.20.21.20/scripts?ActivePowerRegulation.SetGridRegulationTarget`. The request body is the text `-20`. The response is `200 OK` with a status of `48 ms` and `207 B`. The response body is displayed as `-10`.

```
2025-10-16 16:00:40 [ActivePowerRegulation.luaw 543] received new order : -20.0 kW
2025-10-16 16:00:40 [ActivePowerRegulation.luaw 547] New Grid Regulation Target -20.0 kW
2025-10-16 16:00:40 [ActivePowerRegulation.luaw 548] New effective regulation target -10.0 kW
2025-10-16 16:00:40 [ActivePowerRegulation.luaw 1176] meter value is 0.00
2025-10-16 16:00:40 [ActivePowerRegulation.luaw 1176] meter value is 0.00
```

Pilotage local de la centrale

Pilotage par envoi de requêtes Modbus

La consultation ou la modification des variables décrites ci-dessus peut se faire par envoi de requêtes Modbus.

Pour se faire il faut « mapper » les variables dans l'esclave modbus au moyen du fichier de description de la table de registre interne à la WebdynSunPM. Vous pourrez ensuite envoyer des requêtes de lecture et/ou d'écriture sur les registres choisis.

Par exemple :

	A	B	C	D	E	F	G	H	I	J	K
1	ModbusSlave										
2	1		0		ActivePowerRegulation		meterValueKW				4
3	2		1		ActivePowerRegulation		CmdStop				4
4	3		2_4		ActivePowerRegulation		Watchdog				4
5	4		4		ActivePowerRegulation		Alarm				4
6	5		5		ActivePowerRegulation		inverterCommand				4
7	6		6		ActivePowerRegulation		variation				4
8	7		7_4		ActivePowerRegulation		TotInvPower				4
9	8		9		ActivePowerRegulation		SetTarget				4
10											

Les variables du script peuvent être déclarées dans n'importe quels registres réservés aux clients et disponibles aux adresses comprises entre 0 et 32000

Reférez-vous au manuel § 3.4 Modbus esclave TCP (Modbus slave TCP) page 183 pour la configuration des registres de l'esclave Modbus.

Ainsi une écriture sur la variable **SetTarget** entrainera une modification de la consigne de régulation.

Pilotage avancé par requêtes Modbus via SCADA (GTB/GTC)

Il est possible de piloter ce script depuis un SCADA local au site en utilisant les variables décrites dans le chapitre **Principe de fonctionnement** correctement reportée dans la table Modbus esclave

Ainsi il n'est pas nécessaire de connecter un nouveau compteur s'il en existe déjà un qui communique avec le SCADA ou de configurer le SCADA pour qu'il mette à disposition la valeur qu'il collecte. Pour rappel il est fortement déconseillé d'interroger le compteur de référence avec 2 équipements.

En effet en utilisant le paramètre **MeterType** avec la valeur 2 cela permet au SCADA d'écrire directement la valeur de référence de régulation sur la variable **meterValueKW**

Dans ce cas une surveillance de la communication avec le SCADA est nécessaire afin d'identifier les défauts d'émission de consigne, pour ce faire il est nécessaire que le SCADA rafraichisse périodiquement (5s) la valeur de la variable **Watchdog** avec une valeur différente de la précédente.

Les autres défauts sont signalés dans la variable **Alarm**

Pilotage par script SendCommand

Les variables accessibles en écriture précédemment décrites peuvent être modifiées depuis l'interface web de la webdynSunPM en utilisant le script SendCommand avec le paramétrage approprié.

Ci-dessous la modification de la variable SetTarget afin de fixer le seuil à 50KW d'injection :

Add arguments

Script arguments

Cancel Save

Régulation d'injection et découplage

Dans le cas où un dispositif de découplage simplifié (couplage et découplage par commandes impulsionnelles du compteur) doit être utilisé en parallèle du script de régulation il n'est pas possible d'utiliser le script de découplage en parallèle de ce script de régulation.

Pour pallier cette problématique, nous conseillons d'utiliser un relai temporisé type : FINDER 80.61 (https://cdn.findernet.com/app/uploads/IB8061_82FR.pdf) afin de permettre la conversion des impulsions en un état interprétable par une entrée numérique configurée avec le tag **DIN1** (cf **Paramétrage des IO de la WebdynSunPM**) habituellement utilisée pour recevoir l'état de démarrage d'un groupe électrogène. L'entrée numérique ainsi configurée permet d'arrêter la régulation et positionne la production des onduleurs à zéro.

Exploitation des logs

Voir chapitre 4.1.8.2 : les logs de scripts » du manuel WebdynSunPM

Les logs du script accessible depuis l'interface web du concentrateur sont disponible sur le serveur distant ou un fichier est déposé à chaque connexion dans le répertoire /LOG avec le format suivant : WPM000000_LUA_ActivePowerRegulation_241031_150000.log.gz

Avec l'exemple ci-dessous :

```
24/10/04-08:52:59;RegulationPoint:1.80kW ;RegulationLimit:0.00kW ;loop:5.0 s
24/10/04-08:52:59;tune;state:running;meterValue:40.61596875 kW ;requested variation:21.564427083333 % ;inv command:100 %
24/10/04-08:53:00;meter value is 40.31971484375
24/10/04-08:53:01;meter value is 25.328072265625
24/10/04-08:53:01;meter value is 25.328072265625
24/10/04-08:53:01;no new meter value
24/10/04-08:53:02;meter value is 24.556755859375
24/10/04-08:53:02;meter value is 24.0140078125
24/10/04-08:53:03;meter value is 24.26965234375
24/10/04-08:53:03;meter value is 24.26965234375
24/10/04-08:53:03;no new meter value
24/10/04-08:53:04;meter value is 23.78146484375
24/10/04-08:53:05;meter value is 22.245443359375
24/10/04-08:53:05;meter value is 21.78450390625
24/10/04-08:53:06;meter value is 21.78450390625
24/10/04-08:53:06;no new meter value
24/10/04-08:53:06;meter value is 22.246576171875
24/10/04-08:53:06;variation(21.564427083333->11.359208984375 %)
24/10/04-08:53:06;command(100->100 %)
24/10/04-08:53:06;command sent:100(100%) to inverter 1(43538/60KW)
24/10/04-08:53:06;command sent:100(100%) to inverter 2(40272/60KW)
24/10/04-08:53:06;command sent:100(100%) to inverter 3(39766/60KW)
24/10/04-08:53:06;RegulationPoint:1.80kW ;RegulationLimit:0.00kW ;loop:5.0 s
24/10/04-08:53:07;tune;state:running;meterValue:22.246576171875 kW ;requested variation:11.359208984375 % ;inv command:100 %
24/10/04-08:53:07;meter value is 24.85178515625
24/10/04-08:53:08;meter value is 21.26494921875
24/10/04-08:53:08;meter value is 18.313375
24/10/04-08:53:09;meter value is 18.313375
24/10/04-08:53:09;no new meter value
24/10/04-08:53:09;meter value is 19.121578125
24/10/04-08:53:10;meter value is 17.48408984375
24/10/04-08:53:10;meter value is 17.095189453125
24/10/04-08:53:11;meter value is 17.095189453125
24/10/04-08:53:11;no new meter value
24/10/04-08:53:12;meter value is 19.07495703125
24/10/04-08:53:12;meter value is 21.75724609375
24/10/04-08:53:13;meter value is 20.2193359375
24/10/04-08:53:13;variation(11.359208984375->10.232964409722 %)
24/10/04-08:53:13;command(100->100 %)
24/10/04-08:53:13;command sent:100(100%) to inverter 1(43573/60KW)
24/10/04-08:53:13;command sent:100(100%) to inverter 2(40292/60KW)
24/10/04-08:53:13;command sent:100(100%) to inverter 3(39876/60KW)
24/10/04-08:53:13;RegulationPoint:1.80kW ;RegulationLimit:0.00kW ;loop:5.0 s
24/10/04-08:53:13;tune;state:running;meterValue:20.2193359375 kW ;requested variation:10.232964409722 % ;inv command:100 %
```

On observe à chaque boucle de régulation la répétition des 2 lignes suivantes

```
24/10/04-08:52:59;RegulationPoint:1.80kW ;RegulationLimit:0.00kW ;loop:5.0 s
24/10/04-08:52:59;tune;state:running;meterValue:40.61596875 kW ;requested variation:21.564427083333 % ;inv command:100 %
```

La première recapitule le paramétrage :

-RegulationPoint correspond au point de régulation effectif (seuil +pourcentage de la puissance totale)

-RegulationLimit correspond au seuil défini par **Grid regulation target** dans ce cas il s'agit de zero injection

-loop fait référence au paramètre « **Regulation speed** » de la configuration. On observe d'ailleurs que ces 2 lignes se répètent environs toutes les 5s.

Alors que la deuxième donne le résultat de l'analyse de l'algorithme de régulation

-Les 2 premiers champs indiquent l'état de la régulation

-Le premier champ vaut « *tune* » en fonctionnement normal et « *warn* » si la valeur du compteur est inférieure au seuil limite.

-Le second champ indiquera *state* : *running* dans le premier cas et *state* : *limit* dans le second.

-*MeterValue* indique la valeur référence fourni par le(s) compteur(s).

-*Requested variation* donne l'évolution du pourcentage de puissance souhaité par rapport à la valeur de régulation effective.

-Finalement le champ *inv command* indique le pourcentage de commande envoyé a chaque onduleur.

Entre 2 boucles de régulation la surveillance du compteur continue et les valeurs relevées sont affichées.

Le log « *no new meter value* » indique que la valeur du compteur n'a pas évoluée, la table de registre du compteur n'a probablement pas été mise à jour coté compteur.

Ci-dessous un exemple de log ou l'on peut observer une réduction de puissance des onduleurs ainsi qu'un envoi de commande immédiate à la suite d'une lecture du compteur inférieur au seuil limite

La temporisation de 5s est alors ignorée.

```
24/10/04-09:35:08;RegulationPoint:1.80kW ;RegulationLimit:0.00kW ;loop:5.0 s
24/10/04-09:35:08;tune;state:running;meterValue:1.1867084960937 kW ;requested variation:-0.34071750217014 % ;inv command:61.743374892341 %
24/10/04-09:35:09;meter value is 1.1760515136719
24/10/04-09:35:10;meter value is 1.6058214111328
24/10/04-09:35:10;meter value is 1.7040106201172
24/10/04-09:35:11;meter value is 1.7040106201172
24/10/04-09:35:11;no new meter value
24/10/04-09:35:11;meter value is 3.1224379882812
24/10/04-09:35:12;meter value is 2.8238583984375
24/10/04-09:35:12;meter value is 2.7799497070313
24/10/04-09:35:13;meter value is 2.7799497070313
24/10/04-09:35:13;no new meter value
24/10/04-09:35:13;meter value is 1.1767595214844
24/10/04-09:35:14;meter value is 35.79412890625
24/10/04-09:35:15;meter value is 11.716805664063
24/10/04-09:35:15;variation(-0.34071750217014->5.5093364800347 %)
24/10/04-09:35:15;command(61.743374892341->67.252711372376 %)
24/10/04-09:35:15;command sent:67.252711372376(67.252711372376%) to inverter 1(40727/60KW)
24/10/04-09:35:15;command sent:67.252711372376(67.252711372376%) to inverter 2(40724/60KW)
24/10/04-09:35:15;command sent:67.252711372376(67.252711372376%) to inverter 3(40848/60KW)
24/10/04-09:35:15;RegulationPoint:1.80kW ;RegulationLimit:0.00kW ;loop:5.0 s
24/10/04-09:35:15;tune;state:running;meterValue:11.716805664063 kW ;requested variation:5.5093364800347 % ;inv command:67.252711372376 %
24/10/04-09:35:15;meter value is 11.763559570313
24/10/04-09:35:16;meter value is 11.763559570313
24/10/04-09:35:16;no new meter value
24/10/04-09:35:17;meter value is 6.1694033203125
24/10/04-09:35:17;meter value is 0.58923120117187
24/10/04-09:35:18;meter value is -6.8441318359375
24/10/04-09:35:18;variation(5.5093364800347->-4.8022954644097 %)
24/10/04-09:35:18;command(67.252711372376->62.450415907966 %)
24/10/04-09:35:18;command sent:62.450415907966(62.450415907966%) to inverter 1(40727/60KW)
24/10/04-09:35:18;command sent:62.450415907966(62.450415907966%) to inverter 2(40724/60KW)
24/10/04-09:35:18;command sent:62.450415907966(62.450415907966%) to inverter 3(40723/60KW)
24/10/04-09:35:18;RegulationPoint:1.80kW ;RegulationLimit:0.00kW ;loop:5.0 s
24/10/04-09:35:18;warn;state:limit;meterValue:-6.8441318359375 kW ;requested variation:-4.8022954644097 % ;inv command:62.450415907966 %
24/10/04-09:35:19;meter value is -6.957080078125
24/10/04-09:35:19;tempo
24/10/04-09:35:19;meter value is 22.1135546875
24/10/04-09:35:19;variation(-4.8022954644097->11.285308159722 %)
24/10/04-09:35:19;command(62.450415907966->73.735724067688 %)
24/10/04-09:35:19;command sent:73.735724067688(73.735724067688%) to inverter 1(42212/60KW)
24/10/04-09:35:19;command sent:73.735724067688(73.735724067688%) to inverter 2(40724/60KW)
24/10/04-09:35:19;command sent:73.735724067688(73.735724067688%) to inverter 3(40723/60KW)
24/10/04-09:35:19;RegulationPoint:1.80kW ;RegulationLimit:0.00kW ;loop:5.0 s
24/10/04-09:35:19;tune;state:running;meterValue:22.1135546875 kW ;requested variation:11.285308159722 % ;inv command:73.735724067688 %
```

Script alternatif :

Il est possible sur demande au support technique de Webdyn (support@webdyn.com) d'ajuster le comportement du script.

Il est possible par exemple de configurer une source alternative comme référence de puissance tel qu'une entrée analogique 4-20mA ou un registre de la fonction Modbus Esclave.

Il est également possible d'activer l'envoi d'alarme sur perte de communication avec les onduleurs, par défaut seul le compteur provoque cet envoie.

Il est également possible d'envoyer des données issues du script dans les fichiers de donnée standard de la WebdynSunPM afin qu'elles soient exploitées par le portail comme celle d'un équipement supervisé.

Matériel recommandé

Nous pouvons actuellement confirmer le bon fonctionnement du script avec les équipements suivants :

Onduleurs:

Sungrow; Huawei; Goodwee (MT, HT); Sofarsolar; Growatt; SMA (core2, STPX),Kaco

Compteur :

Janitza (UMG604, UMG96RM); Lettel (MCX4 34V); Schneider (IEM32xx, PM55xx)

Pour la compatibilité avec d'autres équipements vous pouvez nous consulter en écrivant a support@webdyn.com.