**Tutoriel : comment installer le driver VERA pour le serveur WES v2**

**INTRODUCTION**

Le Serveur WES est parfait pour la mesure et gestion de notre énergie domestique mais il peut être intéressant d’utiliser la VERA comme un point central dans votre système domotique afin d’intégrer les appareils de technologies différentes qui ont toutes leurs propres avantages.

Il est possible au moyen d’un plugin WES sur VERA d’exposer les diverses fonctions de mesure et de control de relai du serveur WES comme un périphérique VERA et utiliser le moteur de VERA pour les scènes, les triggers, les actions etc . Ainsi les relais apparaissent comme un switch On/OFF, les mesures de températures comme un senseur de température, etc… Rien n’empêche ensuite d’avoir des scènes et des conditions qui mêlent des périphériques zWave, RFX avec ceux du WES

Un tel « plugin » est décrit dans ce document dans une première version, des évolutions viendront au fil du temps

Amg0 est l’auteur de différents plugins pour VERA sur le store micasaverde dont : l’ipx800, le cube canal +, le IPhoneLocator , l’alarme KES Senia et ALTUI ( une interface utilisateur alternative à UI7 et recommandée car bien plus puissante que UI 7 ) .

Pour utiliser cette solution il vous faudra

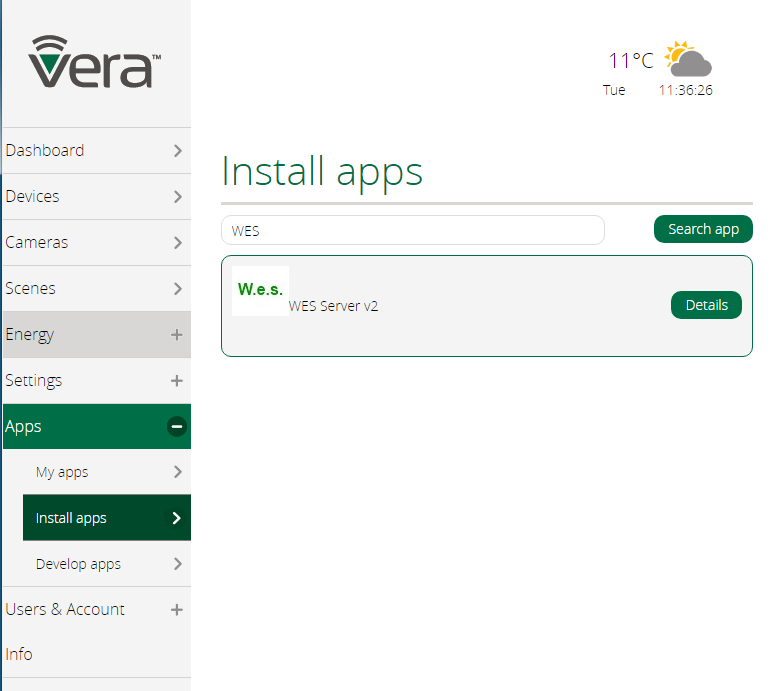
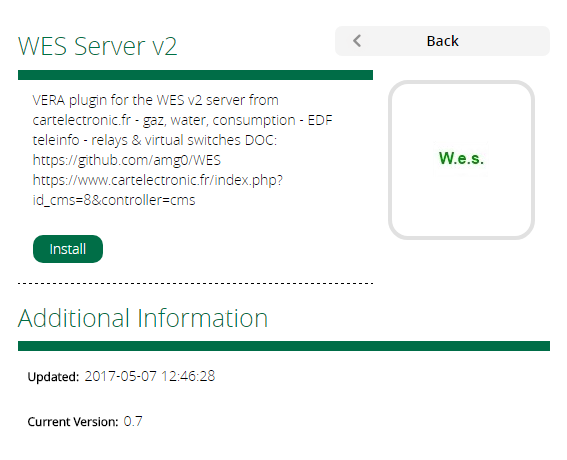
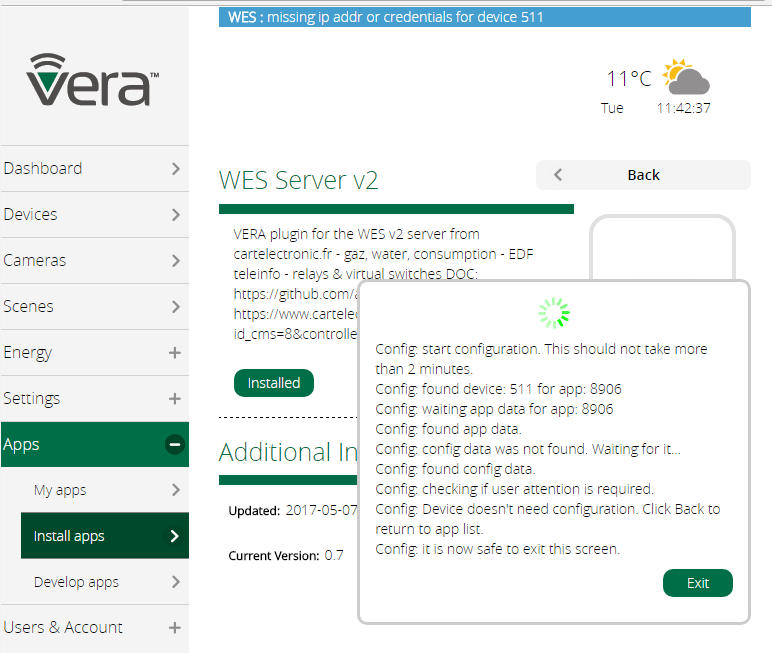
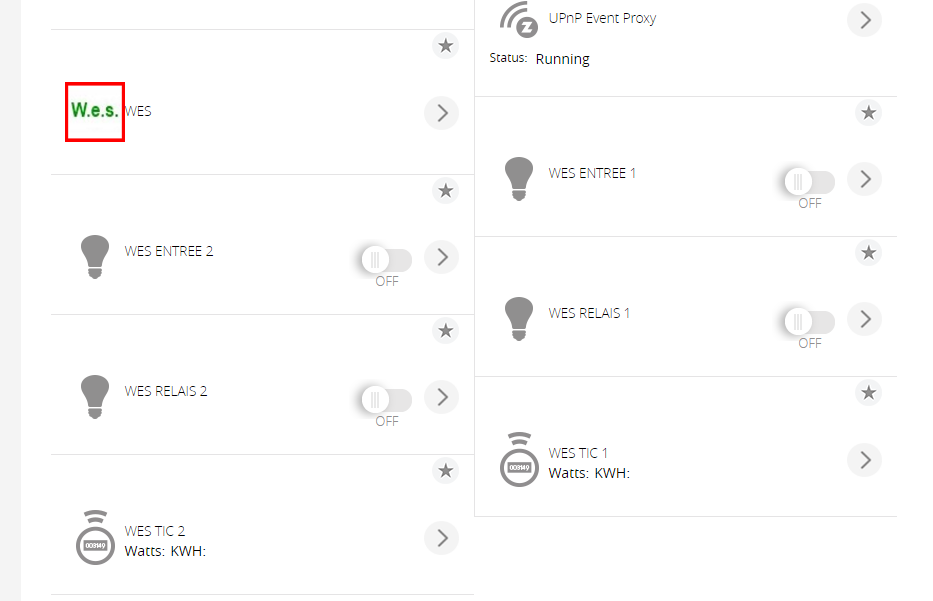
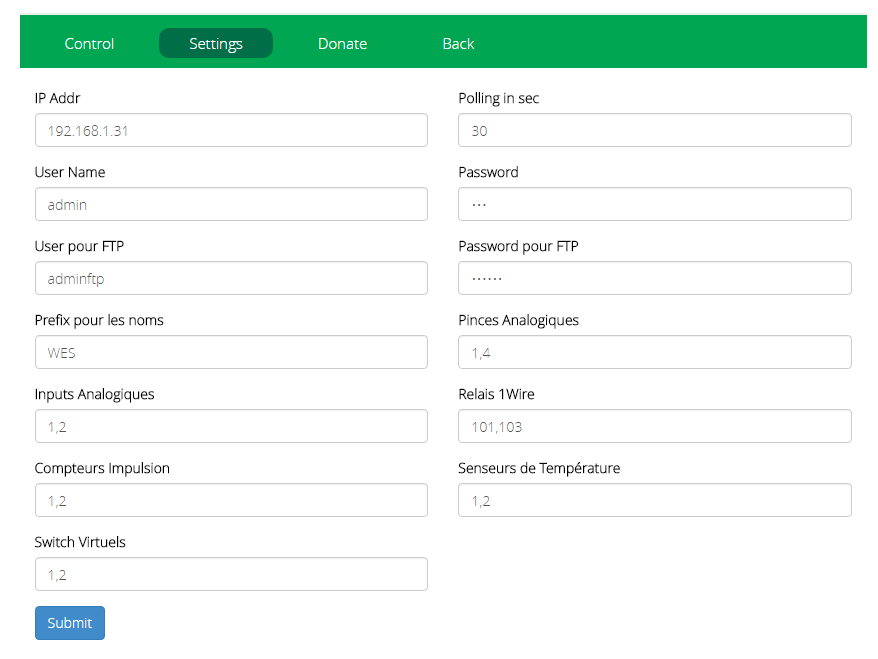
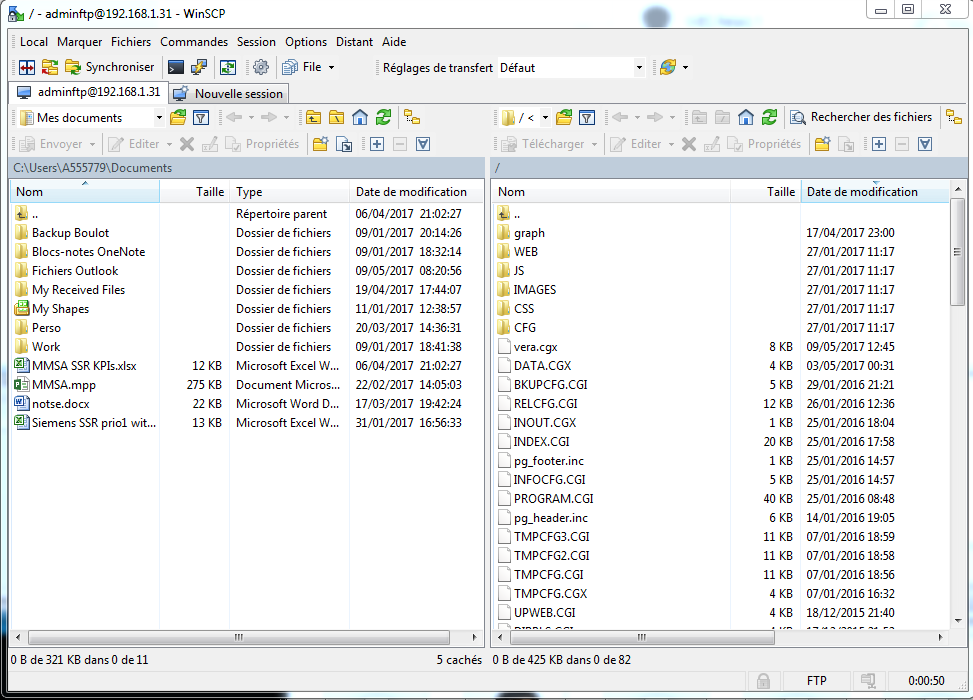
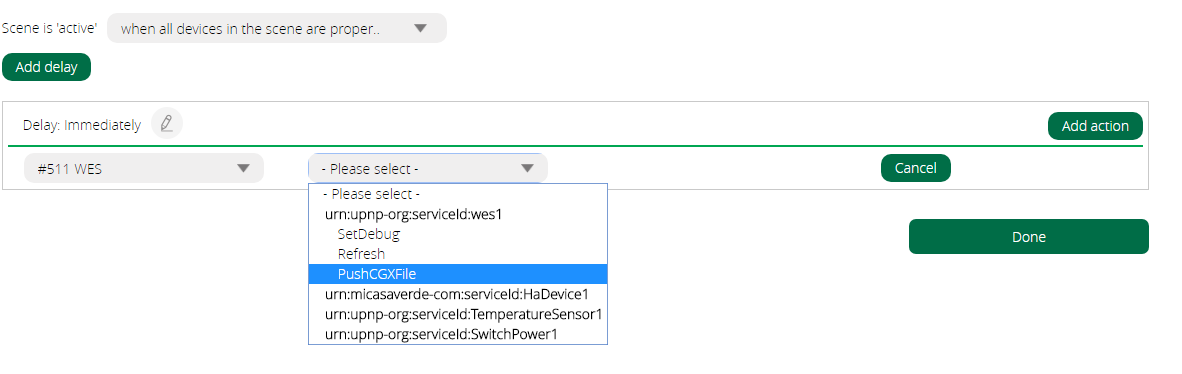
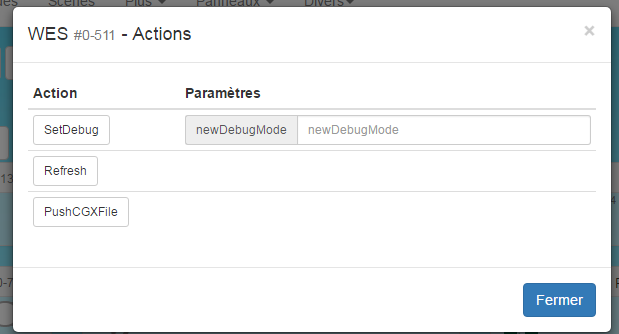
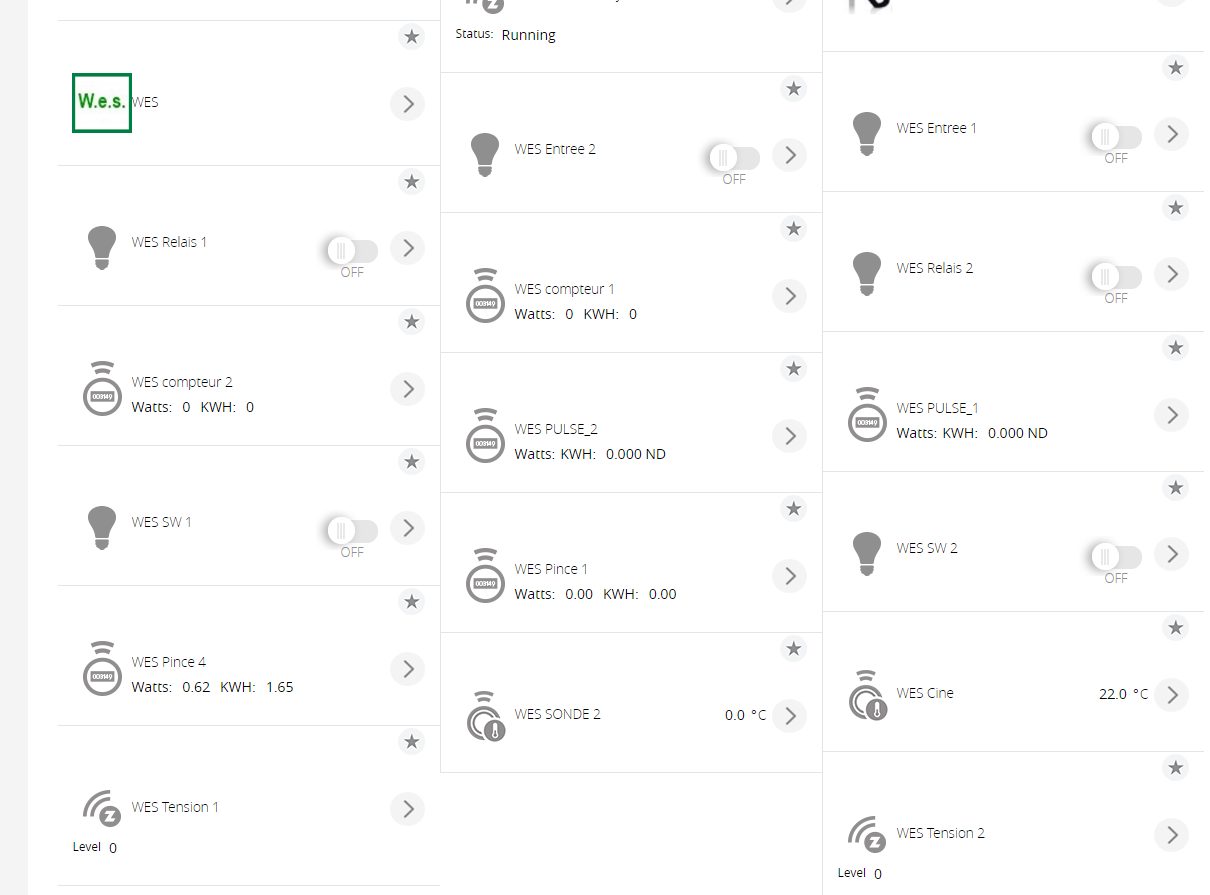
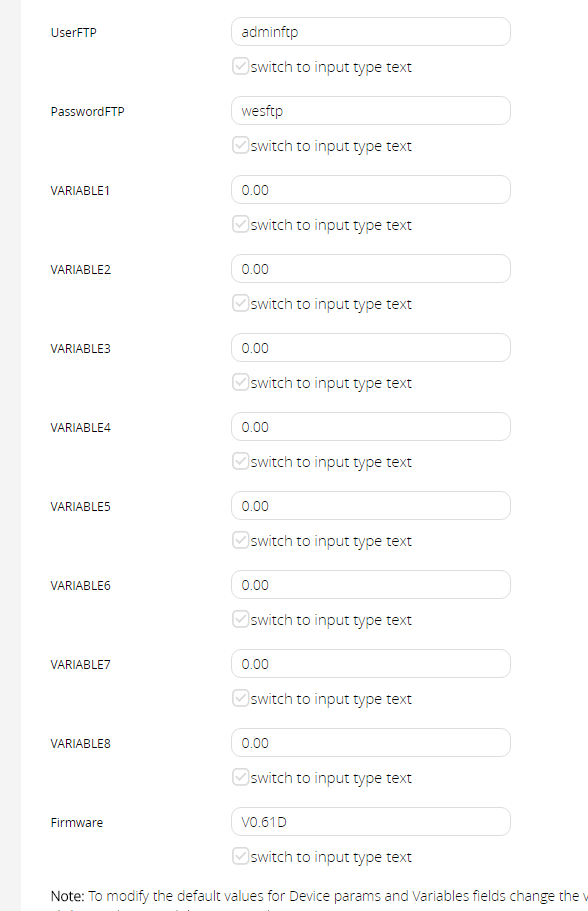
* + Une VERA sous UI7 ou bien une machine (raspberry ou VM ) exécutant openLuup qui est la version opensource compatible avec les API du moteur VERA
  + Un Serveur WES 2.0
  + Le plugin WES installable à partir de l’app store MCV : <http://apps.mios.com/plugin.php?id=8906> ou manuellement à partir du repository github <https://github.com/amg0/wes> voire à partir de l’app Store pour ALTUI/OpenLuup intégré à ALTUI

Cet article explique, pas à pas l’installation et la configuration du plugin ainsi que les différentes « variables » des périphériques VERA exposée afin de remonter les informations sur la VERA et de s’en servir dans les scènes

# Prérequis

* Connaitre l’adresse ip de son serveur WES
* Connaitre les comptes
  + User /password pour l’accès web
  + User / password pour l’accès ftp
  + Avoir coche l’option FTP dans le serveur WES
* Avoir configuré son serveur WES, avec les noms des différents éléments (sondes, relais etc.)
* Avoir une VERA sous UI7 ou ALTUI (<http://apps.mios.com/plugin.php?id=8246>). L’ancienne interface UI5 n’est plus supportée
* Avoir décidé quels éléments on voulait rendre visible sous vera, cela peut être changé à tout moment plus tard

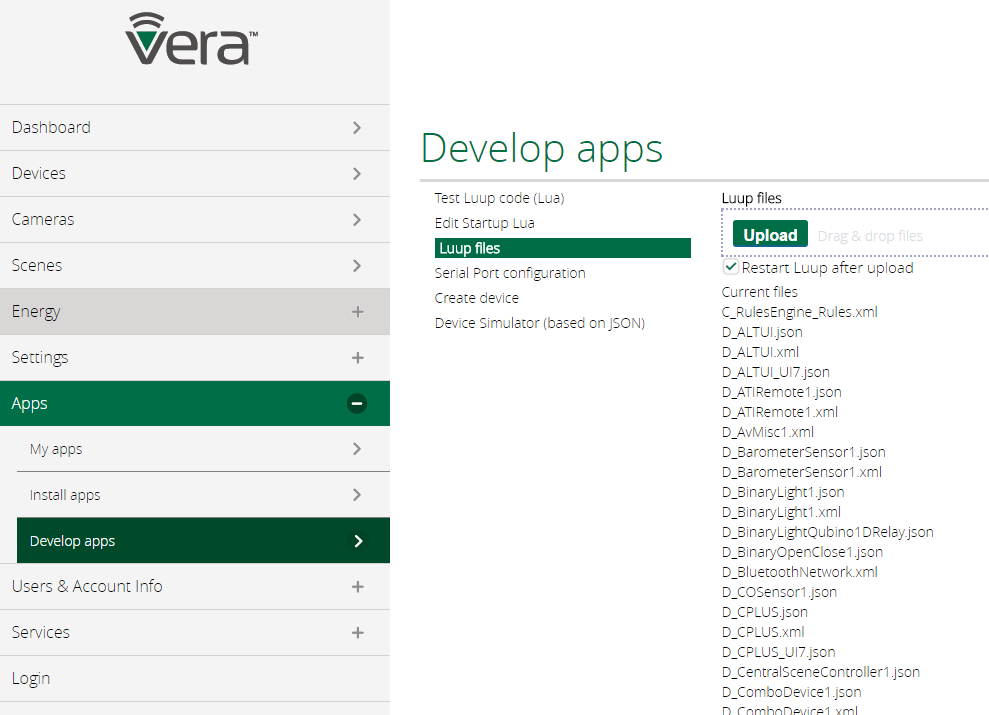
# Installer le plugin vera

* Aller sur la page Install App de UI7 et rechercher WES
  + 
* Cliquer **Détails** puis **Install**
  + 
* Suivre les instructions, la VERA doit redémarrer son moteur Luup
* Un message d’information doit être visible en haut de l’écran, ce message est normal tant que l’application n’est pas configurée
  + 
* Aller dans la page « Devices » et trouver le serveur WES qui doit être compose d’une icône entourée en rouge ainsi que des périphériques « enfants » créés par default. Le rouge indique un défaut de connectivité
  + 
* Aller dans les « Settings » du périphérique « WES »
  + 
* Choisir un préfix a utiliser pour le nom des périphériques « enfants » (ou rien si on préfère mais une convention de nommage rend les choses un peu plus facile).
* Choisir une fréquence de rafraichissement des données (30s ou plus est recommandé. Au besoin il y a une action UPNP « **Refresh** » pour forcer un rafraichissement instantané des donnes – dans le cadre d’une scène par exemple)
* Choisir les périphériques que l’on veut exposer  grâce à une liste de numéro, sépares par des virgules. Par exemple
  + Pinces Analogique : 1,4 pour avoir la 1 et la 4
  + Senseur Température : 1 pour avoir que le 1
  + Etc.
* Les relais, virtual switches et entrées digitale sont actionnable depuis la Vera comme le serait un switch normal ( zwave ou autre )
* Cliquer sur « Submit » après les changements, ceci va provoquer un rechargement du moteur luup pour prendre en compte les nouveaux paramètres et créer les nouveaux périphériques « enfants »
* Si l’accès par FTP est activé sur le WES, la VERA va pouvoir s’occuper de charger un nouveau fichier vera.cgx sur le WES automatiquement par FTP comme vous pouvez le constater par un accès via une session FTP ouverte sur le WES serveur
  + 
  + Si d’aventure le fichier vera.cgx n’était pas créé,  il est possible de forcer sa création à tout moment par le biais d’une action UPNP supportée par le périphérique « WES » avec un script lua ou par le biais d’une scène qui appelle l’action UPNP du device
    - luup.call\_action("urn:upnp-org:serviceId:wes1", "PushCGXFile", {}, nnn) ou nnn est le numéro de device VERA
  + OU, par l’advanced scene editor de VERA
    - 
  + OU, par la dialog box ACTION d’ALTUI
    - 
* Si tout est conforme, et après un ou deux redémarrage du moteur lua sur la VERA , l’icône du périphérique WES est maintenant verte et les périphériques demandés selon les « settings » sont maintenant créés sur la VERA
  + 
* Les 8 variables variables1 à 8 sont présentes sous la forme de « variable » sur le périphérique WES
* 

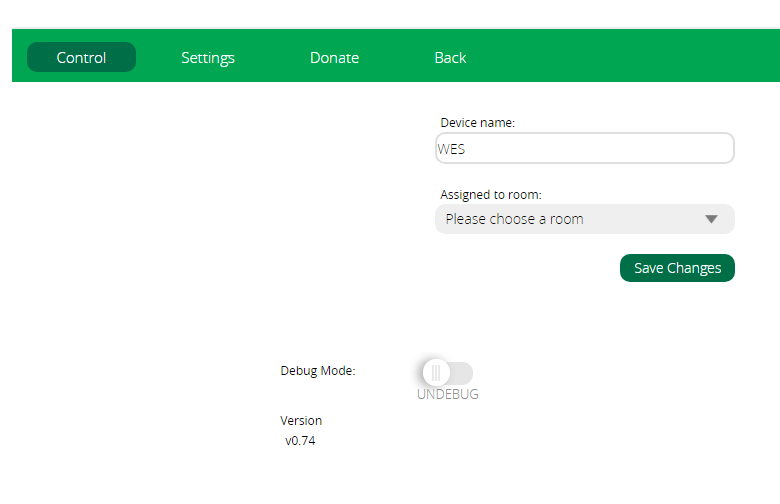
# Installer une version plus récente

Apres une première installation par le store MCV, il est possible d’upgrader à une version plus récente a partir des sources disponibles sur <https://github.com/amg0/WES>

Pour cela il suffit de télécharger les fichiers et de les installer sur la vera par l’intermédiaire de l’écran Apps/Develop Apps de UI7 et de demander un reload. Seuls les fichiers .XML, .Lua, .JS, et .JSON sont nécessaires.



Le numéro de version est visible dans la variable « Version » du périphérique ou sur l’écran de contrôle



# Variables VERA du périphérique principal WES

|  |  |  |
| --- | --- | --- |
| **variable** | **Exemple** | **Explication** |
| AnalogClamps | 1,4 | numéro des pinces ampérométriques séparés par une "," |
| AnalogInputs | 1,2 | numéro des inputs analogiques séparés par une "," |
| Credentials | YWRtaW46d2Vz | user:pwd en base64 |
| Debug | 1 | mode debug (1) ou normal (0) |
| Firmware | V0.61D | version du firmware de la WES |
| IconCode | 100 | 100 si le WES est joignable, 0 sinon |
| NamePrefix | WES | préfix a utiliser pour les noms des périphériques enfants |
| PasswordFTP | wesftp | pwd ftp |
| PulseCounters | 1,2 | numéro des compteurs a impulsion séparés par une "," |
| RefreshPeriod | 30 | refresh toutes les x seconds |
| Relais1W | 101,103 | numéro des relais 1 Wire séparés par une "," |
| TempSensors | 1,2 | numéro des senseurs de température séparés par une "," |
| UI7Check | true | true |
| UserFTP | adminftp | user pour l'accès FTP |
| VARIABLE1 | 0.00 | valeur de la variable VARIABLE1 |
| VARIABLE2 | 0.00 | valeur de la variable VARIABLE2 |
| VARIABLE3 | 0.00 | valeur de la variable VARIABLE3 |
| VARIABLE4 | 0.00 | valeur de la variable VARIABLE4 |
| VARIABLE5 | 0.00 | valeur de la variable VARIABLE5 |
| VARIABLE6 | 0.00 | valeur de la variable VARIABLE6 |
| VARIABLE7 | 0.00 | valeur de la variable VARIABLE7 |
| VARIABLE8 | 0.00 | valeur de la variable VARIABLE8 |
| Version | v0.76 | version du plugin |
| VirtualSwitches | 1,2 | numéro des switch virtuels séparés par une "," |
| nCartesRelais1W | 1 | nombre de carte relais 1W détectées par le WES |

# Variables VERA d’un périphérique TIC

Le nom de la variable correspond au tag XML reporté par le serveur WES http://ip/vera.cgx

|  |  |
| --- | --- |
| **Variable** | **Valeur** |
| ADCO | Pas Dispo |
| BASE | 0 |
| BBRHCJB | 0 |
| BBRHCJR | 0 |
| BBRHCJW | 0 |
| BBRHPJB | 0 |
| BBRHPJR | 0 |
| BBRHPJW | 0 |
| DEMAIN | ---- |
| EJPHN | 0 |
| EJPHPM | 0 |
| HCHC | 0 |
| HCHP | 0 |
| IINST | 0 |
| IINST1 | 0 |
| IINST2 | 0 |
| IINST3 | 0 |
| IMAX | 0 |
| IMAX1 | 0 |
| IMAX2 | 0 |
| IMAX3 | 0 |
| ISOUSC | 0 |
| KWH | 0 |
| Log | 0,0,0,1496863800,1 |
| OPTARIF | . |
| PAP | 0 |
| PEJP | 0 |
| PTEC | TIC NDERR. |
| Pulse | Pas Dispo |
| Watts | 0 |

# Variables VERA d’un périphérique Compteur a Impulsion

|  |  |  |
| --- | --- | --- |
| **Variable** | **Valeur** |  |
| Daily | 338.25 l | Consommation journalière |
| DayBefore | 463.75 l | Consommation de la veille |
| DisplayLine1 | 338.25 l | ligne spéciale pour ALTUI |
| DisplayLine2 | 463.75 l | ligne spéciale pour ALTUI |
| Monthly | 1.98 m3 | Consommation Mensuelle |
| Pulse | 1353 | Compteur d’unité |
| PulsePerUnit | 4 | Impulsion par unité |
| Yearly | 2.43 m3 | Consommation Annuelle |

# Variables VERA d’un périphérique Pince Ampérométrique

|  |  |  |
| --- | --- | --- |
| **Variable** | **Valeur** |  |
| Amps | 0.00 | Valeur de l’intensité mesurée par WES |
| Daily | 42.13 | Consommation journalière |
| KWH | 42.13 | Pour l’affichage vera. Base sur la consommation journaliere |
| Log | 0,0,0,1496863800,1 | Créé et géré par VERA |
| Monthly | 239.97 | Consommation Mensuelle |
| Pulse | 271317 | Compteur d’unité |
| Watts | 0 |  |
| Yearly | 271.32 | Consommation annuelle |

# Troubleshooting

* Vérifier que le WES tourne et est joignable par http://addr\_ip \_du\_wes
* Vérifier la présence du fichier vera.cgx sur le WES
* Vérifier que [http:// addr\_ip \_du\_wes /vera.cgx](http://wes/vera.cgx) retourne un XML avec des données
* Il est possible de mettre un mode DEBUG (ne pas oublier de l’enlever) pour capturer des logs plus généreux lors de l’utilisation du plugin WES
* Si le Transfer du fichier vera.cgx ne se fait pas sur le WES
  + Vérifier que l’accès ftp est autorisé sur le WES
  + Essayer un redémarrage du WES (configuration / Info Serveur / Reset)
  + En dernier recours, le placer à la main par FTP sur le WES

# ALTUI

* Ce plugin est entièrement supporté avec l’interface alternative ALTUI <http://apps.mios.com/plugin.php?id=8246>

