

MODE D'EMPLOI de la carte SD TIPI Version 3 et ultérieures

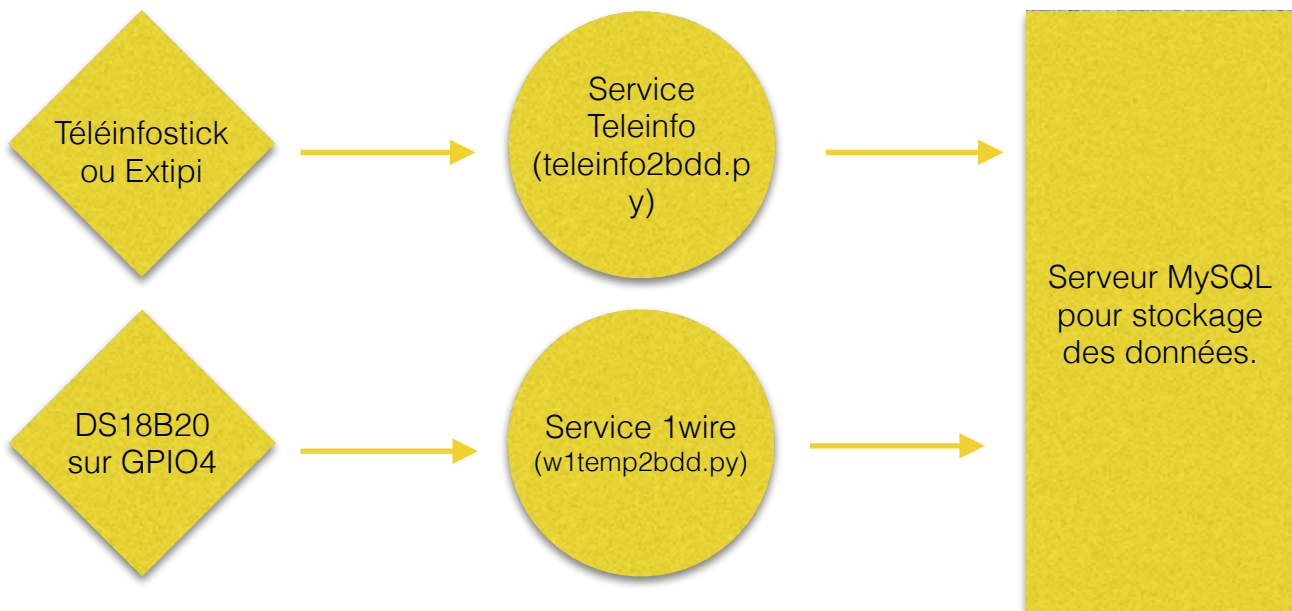
Ce mode d'emploi insiste uniquement sur les différences avec les versions antérieures.
Reportez-vous donc au manuel de l'ancienne version pour avoir les connaissances de base.

FONCTIONNEMENT DES VERSIONS ANTERIEURES

Un service était chargé de collecter les températures des sondes DS18B20

Un service était chargé de collecter les données téléinfo EDF

Ces 2 services étaient chargés de stocker ensuite les données dans la base MySQL



FONCTIONNEMENT DE LA VERSION 3

Le fonctionnement a été totalement revu :

Un programme serveur APIscTeleinfo se charge de récupérer les données téléinfo, uniquement sur demande.

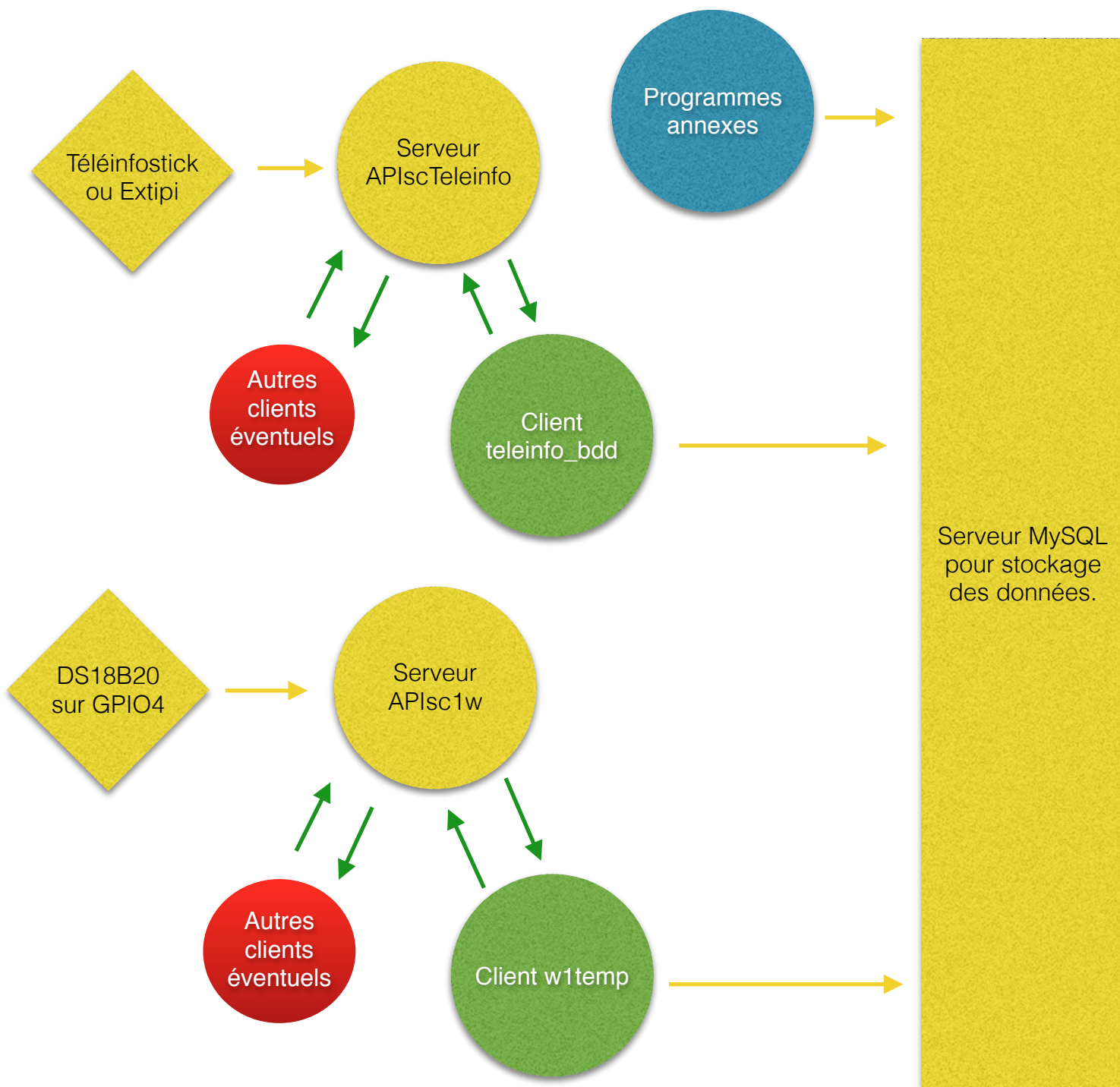
Un programme serveur APIsc1w se charge de récupérer les températures, uniquement sur demande.

De multiples petits programmes (aux noms assez évocateurs) se chargent ensuite de :

- modifier les données dans la base MySQL si nécessaire (par exemple pour poser des repères quotidiens à chaque nouvelle journée au sens des horaires EDF. Ainsi à 6h00, les données sont agrémentées d'un T, et à minuit d'un N. Ces repères permettent de récupérer les index quotidiens pour les calculs de consommations quotidiennes et/ou mensuelles.
- effacer les vieilles données (sauf celles qui portent un repère N ou T) afin de ne pas garder trop d'informations inutiles dans la base MySQL
- faire des sauvegardes hebdomadaires
- gérer l'affichage des LED des cartes ExTipi
- gérer l'envoi de courriels en cas d'alerte

Tous ces petits programmes sont exécutés à intervalles de temps réguliers (en fonction de leur tâche) par CRON (Pour les experts : voir la crontab de l'utilisateur tipi et la crontab du système)

Si vous voulez écrire un programme pour un besoin qui vous est spécifique, inspirez-vous de mes programmes comme bon vous semble.



Les programmes serveurs répondent à des requêtes HTTP (GET) pour obtenir les données auprès des serveurs. La réponse sera au format JSON. (Ces transferts HTTP et JSON correspondent aux flèches vertes sur le diagramme ci-dessus).

Ce sont donc des programmes clients qui se chargent d'envoyer ces requêtes, puis de stocker les données reçues en réponse.

L'avantages de cette architecture, c'est qu'un système externe peut également accéder aux données. Il lui suffit d'émettre une requête HTTP GET et de traiter la réponse au format JSON. La très grandes majorités des centrales domotiques savent le faire nativement (alors qu'interroger la base MySQL est plus complexe).

Vous pouvez donc avoir les données téléinfo sur votre système Tipi, et programmer les « scénarios » de votre centrale domotique pour réagir en fonction des données (actionner un appareil en heures creuses, couper certains appareils durant les jours Tempo Blanc ou Rouge).

ARRÊT et DÉMARRAGE DES SERVICES

Ce sont des services, comme n'importe quel service Debian

`sudo service nom_du_service start` pour le démarrage

`sudo service nom_du_service stop` pour l'arrêt

`sudo service nom_du_service status` pour connaître l'état

`sudo service nom_du_service restart` pour redémarrer

nom_du_service peut être : *APIscTeleinfo* (serveur), *teleinfo_bdd* (client), *APIsc1w* (serveur) ou *w1temp* (client, correspondant au fichier *w1temp2bdd.py*).

Ces 4 services sont automatiquement démarrés après le boot du Raspberry Pi.

LES REQUÊTES HTTP

Il faut faire des requêtes de type GET, avec une adresse (URL) suivant ce format :

Pour la téléinformation EDF, sur le port 100001 :

http://adresse_IP_du_raspberry:10001/API/port=/dev/ttyUSB0

ou

http://adresse_ip_du_raspbrry:1000/API/PORT=/dev/ttyUSB0

Attention à la casse dans l'adresse URL :

« port » donnera les valeurs numériques sur forme de chaînes de caractères dans la réponse

« PORT » donne ces mêmes valeurs sous forme de nombres

Le port de communication doit être de ce genre : */dev/ttyUSBx*, */dev/teleinfox*, */dev/ttyACMx*, *extipi1* ou *extipi2* (x est un nombre compris entre 0 et 3)

Le serveur *APIscTeleinfo* récupère au démarrage la liste des ports configurés par l'utilisateur (page Paramètres de l'interface web). Si ces réglages ont changé, il faut lui dire se mettre à jour avec la requête suivante :

http://ip_du_raspberry:10001/API/refresh_port

Pour les sondes DS18B20, sur le port 10000 :

http://adresse_IP_du_raspberry:10000/API/id=28-aaaa2564def

Le numéro id correspond au numéro de série de la sonde, tel qu'il est détecté par le système, et tel qu'il est affiché dans la page « Paramètres »

Il faut éviter les requêtes simultanées, et la réponse est parfois un peu retardée (quelques secondes), surtout avec la carte ExTipi. Cela est du à la façon dont est gérée l'absence éventuelle d'un des 2 compteurs que gère cette carte.

RÉPONSE AU FORMAT JSON

Pour la téléinformation, la réponse contiendra toutes les données envoyées par le compteur, le nom du port utilisé, et pour les systèmes gérant plusieurs compteurs, le numéro du compteur. (Sinon Num vaut 0)
Quelques exemples :

Pour un Teleifostick sur le port /dev/ttyUSB0 :

```
{"IINST": 1, "OPTARIF": "BASE", "ADCO": 40621215363,
"MOTDETAT": 0, "PAPP": 150, "BASE": 473210, "Num": 0,
"ISOUSC": 15, "IMAX": 4, "PTEC": "TH..", "Port": "/dev/
ttyUSB0"}
```

Pour une carte ExTiPi, compteur n°1 (Version 1) :

```
{"DEMAIN": "ROUG", "ADCO": 31678434483, "BBRHPJR": 262202,
"ISOUSC": 45, "IMAX": 53, "PAPP": 860, "IINST": 4, "OPTARIF":
"BBR,", "BBRHCJR": 330916, "MOTDETAT": 0, "BBRHPJB": 6044066,
"BBRHCJW": 705893, "Num": 1, "BBRHPJW": 748962, "BBRHCJB":
5402461, "PTEC": "HCJR", "Port": "extipi1", "HHPHC": "Y"}
```

Pour une sonde 1wire DS18B20 :

```
{"Temperature": 2.3}
```

En cas de commande « refresh_port » :

La réponse indique « OK », la liste des ports configurés et détectés, et si une carte ExTiPi a été détectée :

```
{"port2": "/dev/ttyUSB0", "extipi": "presente", "refresh":
"OK", "port1": "extipi1"}
```

ERREURS POSSIBLES DANS LA RÉPONSE JSON

Port de communication (ou sonde DS18B20) inexistant ou non paramétré :

```
{"erreur": 1}
```

Absence de données Téléinfo :

```
{"Num": 0, "Port": "/dev/ttyUSB0", "erreur": 2}
```

CHOIX DES PORTS

Pour chacun de vos compteurs vous devez indiquer le port utilisé (soit un Teleinfostick, soit une carte Extipi)

Cas de la carte extipi :

Elle gère 2 compteurs, donc il y a 2 ports : extipi1 ou extipi2 (correspondant aux borniers teleinfo1 et teleinfo2)

Cas d'un Teleinfostick V2

C'est un port /dev/ttyUSBx (x étant un nombre)

Cas d'un Teleinfostick V3

c'est un port /dev/ttyACMx (x étant toujours un nombre)

Dans les 2 cas, x peut varier (surtout s'il y a plusieurs Teleinfosticks ou que d'autres adaptateurs Série/USB sont utilisés) et il y a donc des liens (des raccourcis) qui sont créés par le système (règle udev). Ces liens sont les ports /dev/teleinfox

Ces liens /dev/teleinfox sont dépendants de la prise USB utilisée sur le Raspberry

Raspberry A et B :

/dev/teleinfo0 prise du bas

/dev/teleinfo1 prise du haut

Raspberry A+ et B+ :

/dev/teleinfo0 prise en bas à gauche

/dev/teleinfo1 prise en haut à gauche

/dev/teleinfo2 prise en bas à droite

/dev/teleinfo3 prise en haut à droite

Attention : de vieux Teleinfostick V2 ne sont pas programmés au niveau du composant qui gère l'USB : ils apparaissent comme des FT232 au lieu d'apparaître comme des Teleinfostick V2. Ça ne change rien au fonctionnement, mais pour eux, les raccourcis /dev/teleinfox ne sont pas créés. Voir le site www.suiviconso.fr pour savoir comment les reprogrammer.

Dans une session SSH, les commandes suivantes vous permettront de connaître les noms des ports que l'OS Debian a reconnu :

*ls /dev/tty**

*ls /dev/teleinfo**