

# Domotique

- [Introduction](#)
- [Serveur](#)
- [Capteurs & actionneurs](#)
- [Passerelles](#)
- [Home Assistant](#)

# Introduction

La domotique est un domaine très large avec beaucoup d'acteurs et de protocoles ... Mais les choses changent et on voit (enfin) des standards se profiler à l'horizon à des prix bien plus abordables.

## Architecture globale

Nous allons trouver différents acteurs dans notre installation domotique, faisons un petit tour ...

### Capteurs

On commence par le plus simple. Les capteurs vont nous permettre d'introduire des données dans notre système : température, humidité, pression, présence, mouvement, niveau d'eau, ouverture de porte/fenêtre ... Il en existe de toute sorte.

### Actionneurs

A l'inverse des capteurs, les actionneurs vont agir sur notre environnement : chauffage, électrovanne, haut-parleur, SMS, écran ... La aussi la liste est trop longue pour la faire ici !

A noter que évidemment on certains dispositifs peuvent adopter plusieurs rôles, il n'est donc pas rare qu'un capteur soit également un actionneur. Par exemple il existe des relais qui permettent de piloter une lampe ou une pompe par exemple qui sont également capable de mesurer la consommation électrique, ainsi ils ont un côté actionneur mais aussi capteur ...

### Serveur

Le serveur c'est le **chef d'orchestre** de votre installation domotique. Capteurs et actionneurs ne sont sensés que se cantonner à leur rôle (bien que comme évoqué avant, certains peuvent cumulé les deux rôles, il peuvent alors être un peu plus intelligent comme un thermostat).

De ce fait c'est lui qui va gérer un certain nombre de choses :

- Les **scénarios** : c'est les scénarios qui vont définir les automatisme de votre installation. Par exemple "coupe le chauffage si personne n'est dans la maison" ou "envois moi un

SMSsi tu détectes de l'eau dans la cave". Une autre manière de dire est de dire qu'il fait le lien entre les *capteurs* et les *actionneurs*.

- Les **données** :
- Le **pilotage** manuel : les scénarios c'est bien beau mais on ne peut pas toujours tout mettre en automatique et on veut aussi pouvoir suivre l'évolution de certaines données (sa consommation électrique par exemple). Un serveur domotique digne de ce nom permet donc d'afficher des graphiques, mais aussi de piloter "à la main" votre installation : allumer/éteindre les lumières, mettre en marche la filtration de votre piscine, passer en mode "vacances" ...

Ce serveur est dans la très large majorité des cas un serveur informatique, c'est à dire un ordinateur sans écran qui fait tourner des logiciels en permanence ... La domotique étant généralement peu gourmande en ressource, un serveur domotique tourne souvent sur des très petites plateformes surtout pour économiser de l'électricité, des nuisances sonores, de la chaleur et de l'argent. A ce titre vous croiserez souvent sur votre route le célèbre Raspberry Pi, que je vous déconseille néanmoins mais j'y reviendrai dans la page dédiée aux serveurs ...

Certaines solutions proposent une application sur smartphone ou tablette qui peut être bien pratique

## Communication/protocole

Pour que tout ce beau monde fonctionne il faut un lien entre eux. Et suivant les ça il existe toute une panoplie de protocoles : Wifi, Bluetooth, Zigbee, Z-Wave, X10, ethernet ... Comment choisir ? Tout le débat est là ... Plusieurs facteurs rentrent en compte :

- **Fil ou sans-fil** : souvent le facteur principal. Sans-fil en générale plus pratique car beaucoup plus facile à installer, mais peut-être moins fiable et nécessite souvent une phase de configuration. Si vous choisissez du sans-fil, la **portée** devient un facteur à étudier.
- **Consommation énergétique** : le protocole va énormément influencer la consommation électrique. Certains capteurs peuvent fonctionner des années sur une simple pile bouton, mais suivant le besoin il faudra une connexion permanente et donc une haute demande en énergie ...
- **Sécurité** : certains protocoles sont sécurisés par défaut, d'autres pas, à vous de voir ...
- **Fiabilité** : c'est un peu comme tout, certains protocoles marchent mieux que d'autres.
- **Prix** : certaines technologies (surtout celles qui sont fermées) vont coûter plus chères. A vous de trouver le rapport performance/prix qui vous convient.
- **Popularité** : on va aussi choisir un protocole parce qu'il est plus ou moins répandu, ce qui évite ensuite d'avoir à en mixer 20 chez soi ... Popularité qui va également jouer sur le prix.

## Passerelles

Elle permettent de passer d'un protocole à un autre. Et oui vous vous en rendrez rapidement compte, un protocole ne peut pas les gouverner tous. Pour une raison ou une autre il faudra mixer. Pour faire ça on va utiliser des passerelles (**gateway** en anglais) pour "convertir" un message d'un protocole vers un autre.

L'association à la mode est Zigbee et wifi. Votre serveur basiquement ne peut communiquer que sur des protocoles IP (Wifi et ethernet). Globalement c'est pratique, votre passerelle étant souvent votre box internet (et oui vous ne l'aviez pas vue comme ça !). Mais le wifi a un gros défaut, il consomme. Ce n'est pas un protocole conçu pour l'économie d'énergie. La quasi totalité des composants domotiques exploitant cette connexion se trouve donc souvent à côté d'une prise ... La on se dit qu'avoir un protocole plus économique serait bien pratique. Nous reviendrons en détail dessus mais le Zigbee fait une grosse percée depuis quelques temps, et à l'inverse du wifi, est vraiment orienté économie. J'ai des capteurs de température depuis deux ans, ils fonctionnent avec la même pile bouton depuis 2 ans et le niveau est toujours >90% ! Cependant ils ont besoin d'un petit dispositif qui peut prendre différentes formes (dans cette documentation nous développerons surtout le dongle USB couplé à Zigbee2Mqtt) qui va "traduire" le message en Zigbee vers un protocole IP pour que notre serveur récupère la donnée (ou en envoi si c'est le cas).

# Serveur

Grosso modo pour le serveur vous arrivez devant un grand choix à faire : **solution toute prête dédiée** à la domotique ou je développe ma **propre solution**. Alors attention, la seconde option est loin d'être aussi inaccessible qu'elle n'y paraît ...

Toutes les solutions évoquées ici fonctionnent sous **Linux**, donc assez rapidement on peut se retrouver à jouer avec et faire de l'administration système. Je vous préviens c'est tout :D

A noter que certaines solutions proposent directement des box toute prêtes. Sauf si vous êtes fainéant ou pressé, monter sa solution domotique est souvent le but du jeu et acheter une box gâche un peu le plaisir avec évidemment un surcoût notable.

## Logiciel

### Solution ready-to-go

La plupart des solutions vous sont proposées soit à installer sur un ordinateur déjà doté d'un système d'exploitation, soit de télécharger des images systèmes entières (typiquement pour Raspberry Pi). Cette deuxième option peut être un bon point de départ si on a jamais fait d'administration système, mais vous êtes tributaire de la plateforme pour laquelle elle a été préparée.

#### Jeedom

<https://www.jeedom.com/site/fr/>

#### Domoticz

<https://www.domoticz.com/>

#### Home Assistant

<https://www.home-assistant.io/>

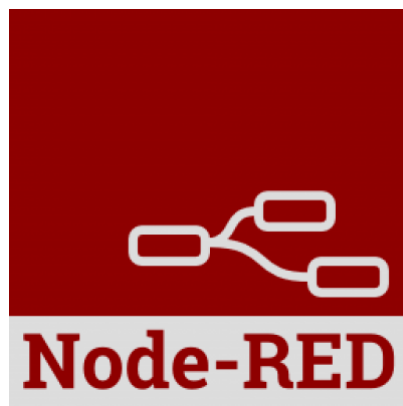
#### Gladys

# Solution homemade

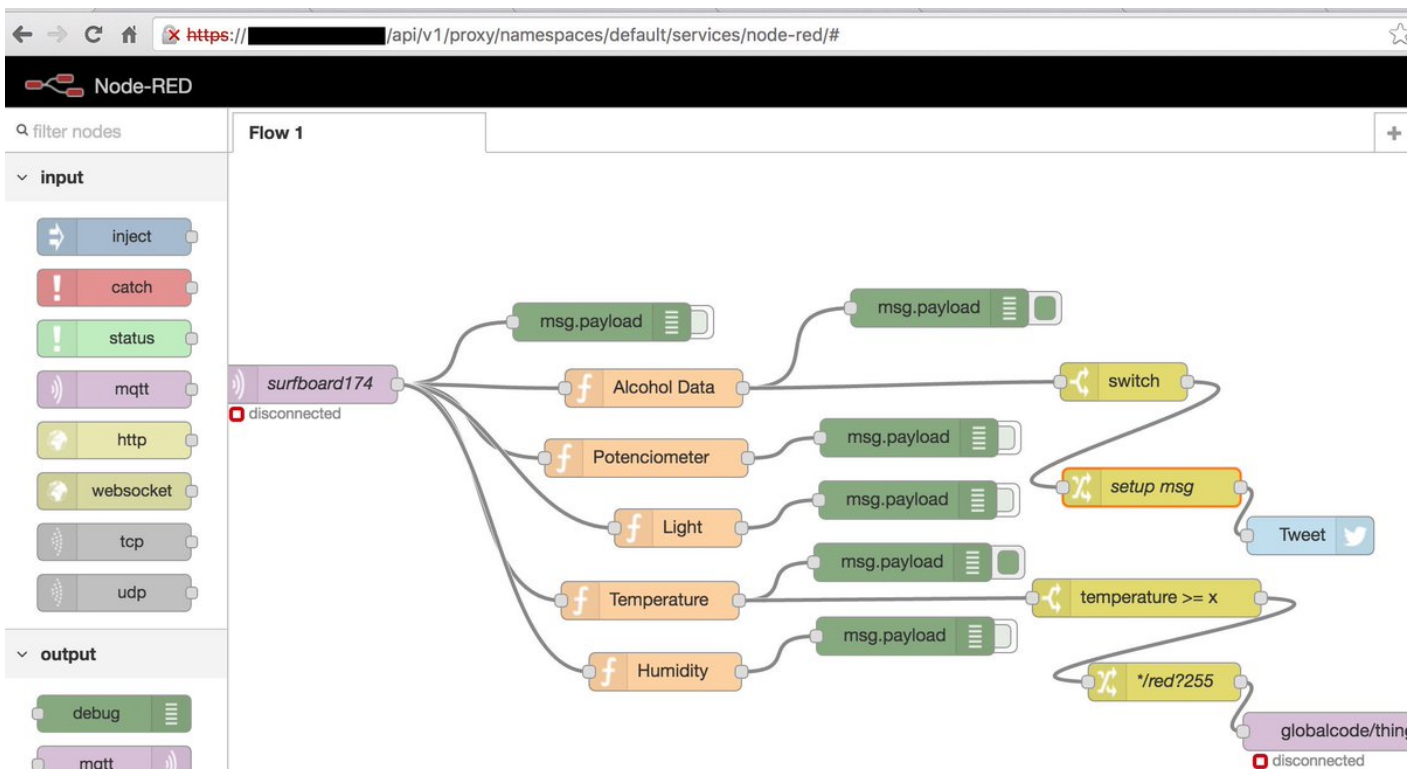
Si vous êtes développeurs vous pouvez évidemment faire ce que vous voulez dans le langage de votre choix, tout est possible ... Je ne vais donc traiter que de NodeRed qui permet de coder un serveur (pas que domotique vous pouvez faire ce que vous voulez avec) de manière graphique et de Grafana qui est un outil qui permet d'afficher des données.

## NodeRed

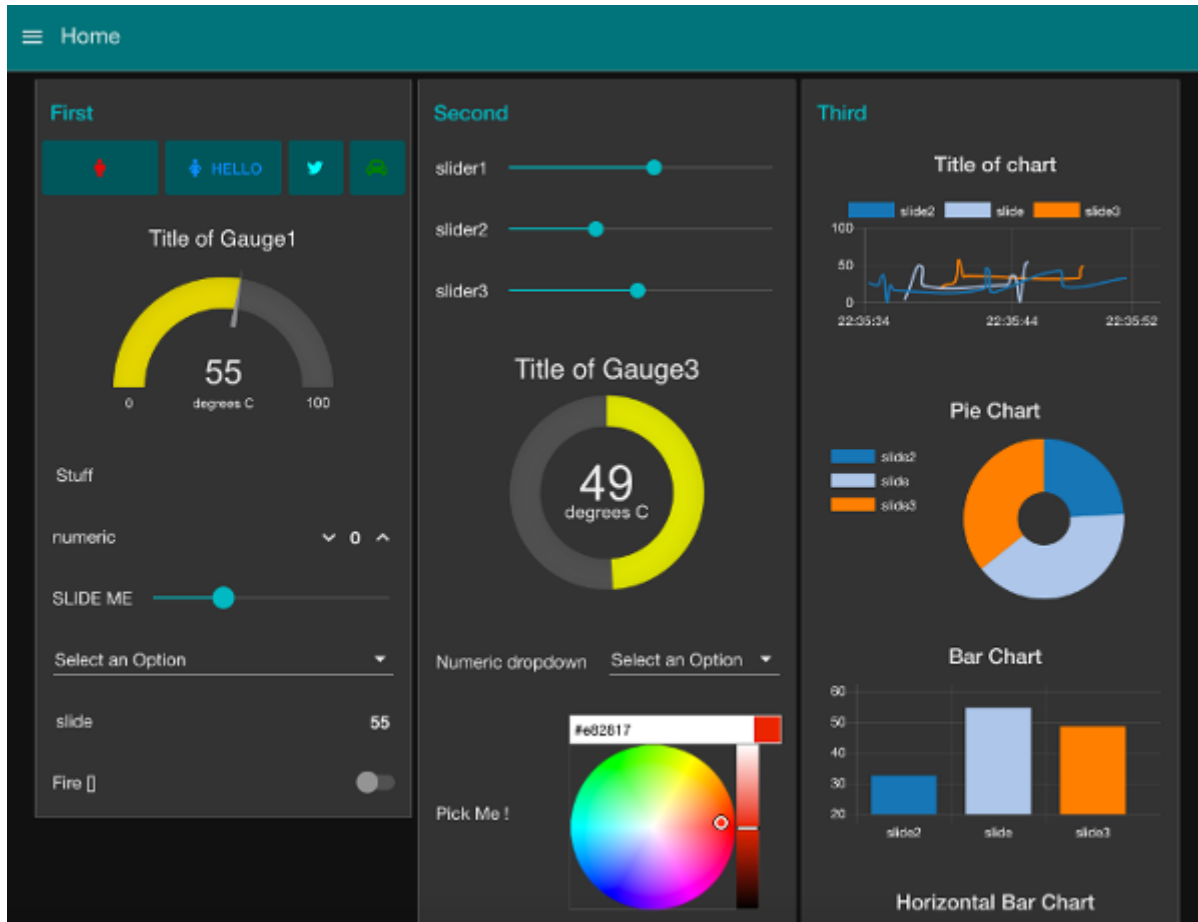
<https://nodered.org/>



Développé par *IBM* initialement, elle fonctionne sous *NodeJS*, est Open Source et est assez simple à prendre en main. On accède à l'interface de développement via son navigateur et on a un système de bloc fonctionnel (les **nodes**) que l'on relie entre eux :



Il y a un système d'extension qui permet d'ajouter d'autres nodes, comme par exemple **dashboard** qui permet très facilement de créer une interface :



Grafana



<https://grafana.com/>



# Matériel

Vous avez choisi votre solution, super ! On l'installe sur quoi maintenant ? Comme mentionné en intro, on va souvent chercher au plus petit ... et au plus simple à utiliser aussi. Les **SBC (Single Board Computer)** sont des petites cartes qui embarquent tout ce dont un ordinateur a besoin. En tête de gondole, le *Raspberry Pi*, mais il en existe bien d'autres : Asus Tinker, Orange Pi, Banana Pi, RockPi etc ... **Mais laquelle choisir ?**

La aussi beaucoup de facteurs entre en jeu (prix, puissance, connectivité consommation ...) comme un PC classique en somme. Par contre je vais m'attarder sur un point en particulier.

## Raspberry Pi = faux ami ?

Il faut bien comprendre un truc : une fois votre logement domotisé, votre serveur devient le lien entre tout. Vous appuyer sur un interrupteur, le message par au serveur, il sait que sur tel interrupteur il doit allumer telle lumière, il l'envoi à l'ampoule et hop ça s'allume. Si votre serveur n'est pas fonctionnel => RIEN ne fonctionne. C'est votre tour de contrôle, sans elle y'a plus un avion qui décolle ... Elle doit donc être d'une fiabilité à toute épreuve. Et la le bas blesse avec notre copain le Raspberry .. En effet, le "disque dur" de votre Raspberry Pi est une **carte SD**. Or une carte SD c'est fait pour stocker des photos ou des mp3, mais **surtout pas pour supporté le**

**stresse généré par un système d'exploitation ... Elles crament donc souvent ...**

## Un bon conseil ?

Privilégier les plateformes à mémoire dites eMMC : c'est une "vraie" mémoire informatique. Elle n'est pas aussi fiable qu'un SSD mais comparé à une SD on est à des années lumières.

Dans la catégorie la famille des RockPi offre des prestations sympas à bas prix, et le développement est assez bien suivi (ce qui n'est pas le cas de tous les SBC ...).

**Mais je vais vous donner mon astuce ultime :**



Vous vous souvenez des netbooks il y a quelques années ? Ces tout petits ordinateurs à pas cher avec des Intel Atom ? Ils ont presque tout pour plaire !

- **Pas cher** : si vous ne le récupérez pas dans votre placard, on en trouve pour une bouchée de pain sur le bon coin ou aux puces ...
- **Econome** : même si vous prenez un modèle assez ancien (je vous déconseille cependant d'en prendre un trop vieux car vous pourriez avoir des soucis pour installer une version de Linux à jour) les processeurs Atom étaient déjà très économes (c'était d'ailleurs un des points forts de ces petits PC).
- **Plus puissant** : là aussi, même les vieux sont toujours plus puissants qu'un Raspberry
- **Une vraie mémoire** : du fait d'un besoin en compacité, ces petites bestioles étaient très souvent équipées en ... **eMMC** ! Alors des fois on trouve des versions avec disques durs, c'est pas mal non plus ... Alors bien évidemment, l'espace est souvent très faible comparé

à ce qu'on a aujourd'hui (16 go, 32go, 64go avec de la chance) mais c'est largement suffisant ! N'oubliez pas, on est ici pour faire tourner des logiciels assez simples, sur un système d'exploitation (Linux pour rappel) connu pour sa faible utilisation de ressources. Globalement 8 go c'est le minimum, à 16 go on commence à être largement tranquille ...

Autre possibilité : recycler un vieux PC. N'importe quoi avec un processeur x86 fait l'affaire ! Alors n'allez pas chercher votre Pentium 2 non plus hein ...

# Capteurs & actionneurs

## SonOff

<https://www.itead.cc/> mais disponibles dans plein d'autres crèmeries ...

SonOff ça tourne surtout autour du wifi (ils font un peu de Zigbee maintenant).

Un mot clé = pas cher

La plupart des produits sont basés sur de l'ESP8266, un microcontrôleur qu'on peut facilement reprogrammer pour y mettre un firmware open source comme [espurna](#) ou [tasmota](#).

## Xiaomi/Aqara

<https://www.banggood.com/search/aqara.html?from=nav>

Une très bonne gamme en zigbee et pas chère du tout. Tous les modules semblent compatible avec [zigbee2mqtt](#)

## Ikea

Ikea s'appuie sur le zigbee pour ses objets domotiques. On peut causer avec grâce (toujours) à [zigbee2mqtt](#)

Prix imbattables, fiables ... Que demander de plus ?

## Shelly

Le concurrent de SonOff, avantages par rapport à SonOff

<https://shop.shelly.cloud>

- Firmware basé sur [moongose-os](#)

- API documenté : <https://shelly-api-docs.shelly.cloud/#shelly-family-overview>
- MQTT intégré par défaut
- Cloud 100% facultatif
- Reprogrammation possible sans modification pour les relais (par contre les broches sont très petites)
- Version du relai (Shelly 1L) ,sans neutre nécessaire, (il faut un bypass pour les ampoules consommant moins de 20w)
- Livraison par DHL depuis l'Europe (2 jour ouvrables)

# Passerelles

## Zigbee2mqtt

<https://www.zigbee2mqtt.io/>

Comme son nom l'indique, permet de passer des messages en Zigbee au travers d'un serveur MQTT.

La forme "classique" est le dongle USB qu'il faut reprogrammer (j'ai un programmeur si vous voulez) et insérer dans son serveur. On installe ensuite le soft qui va bien, on le configure et on peut coupler ses devices Zigbee.



Pour ~6\$ vous pouvez vous ouvrir une grande porte ...

## SonOff bridge

Pour passer du 433 MHz à l'IP.



# OpenMQTT Gateway

Passerelle pour passer du BLE / LoRa / GSM / Infrarouge / 433mhz vers du MQTT

Compatible ESP8266 / ESP32 / SonOff Bridge / M5Stack / M5Stick C / Sonoff RFR3 / TTGO LORA32

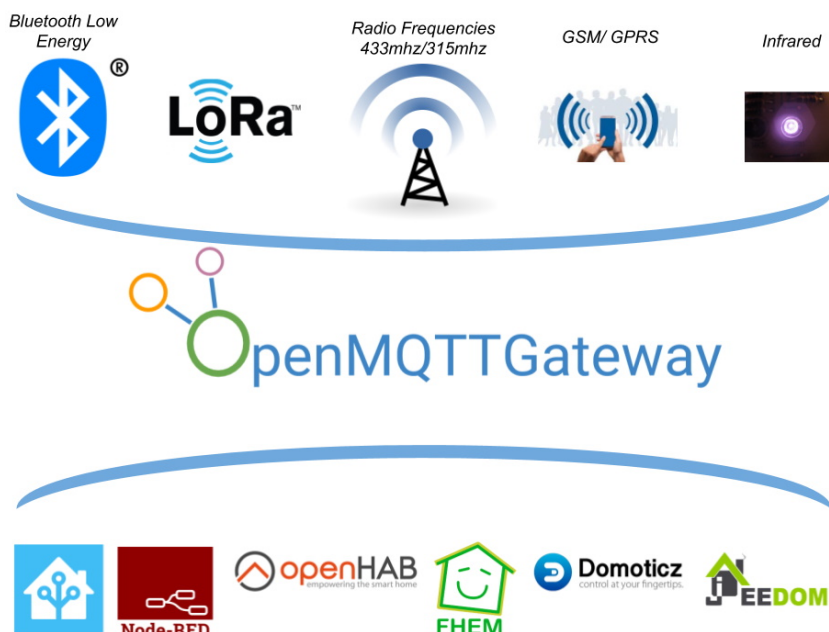
- Liste des appareils compatibles :

[https://docs.google.com/spreadsheets/d/1\\_5fQjAixzRtepykML-3uN3G5bLfQ0zMajM9OBZ1bx0/edit#gid=2126158079](https://docs.google.com/spreadsheets/d/1_5fQjAixzRtepykML-3uN3G5bLfQ0zMajM9OBZ1bx0/edit#gid=2126158079)

- Listes des cartes compatibles :

<https://docs.openmqttgateway.com/prerequisites/board.html#diy-boards>

- Documentation : <https://docs.openmqttgateway.com/>





# Home Assistant