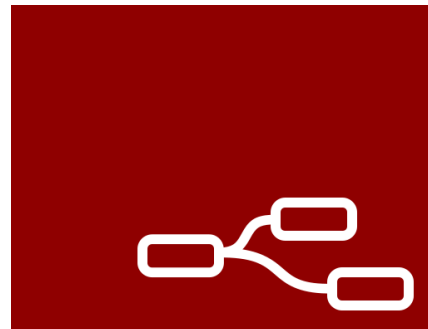


Mes 1^{er} pas avec Node-RED

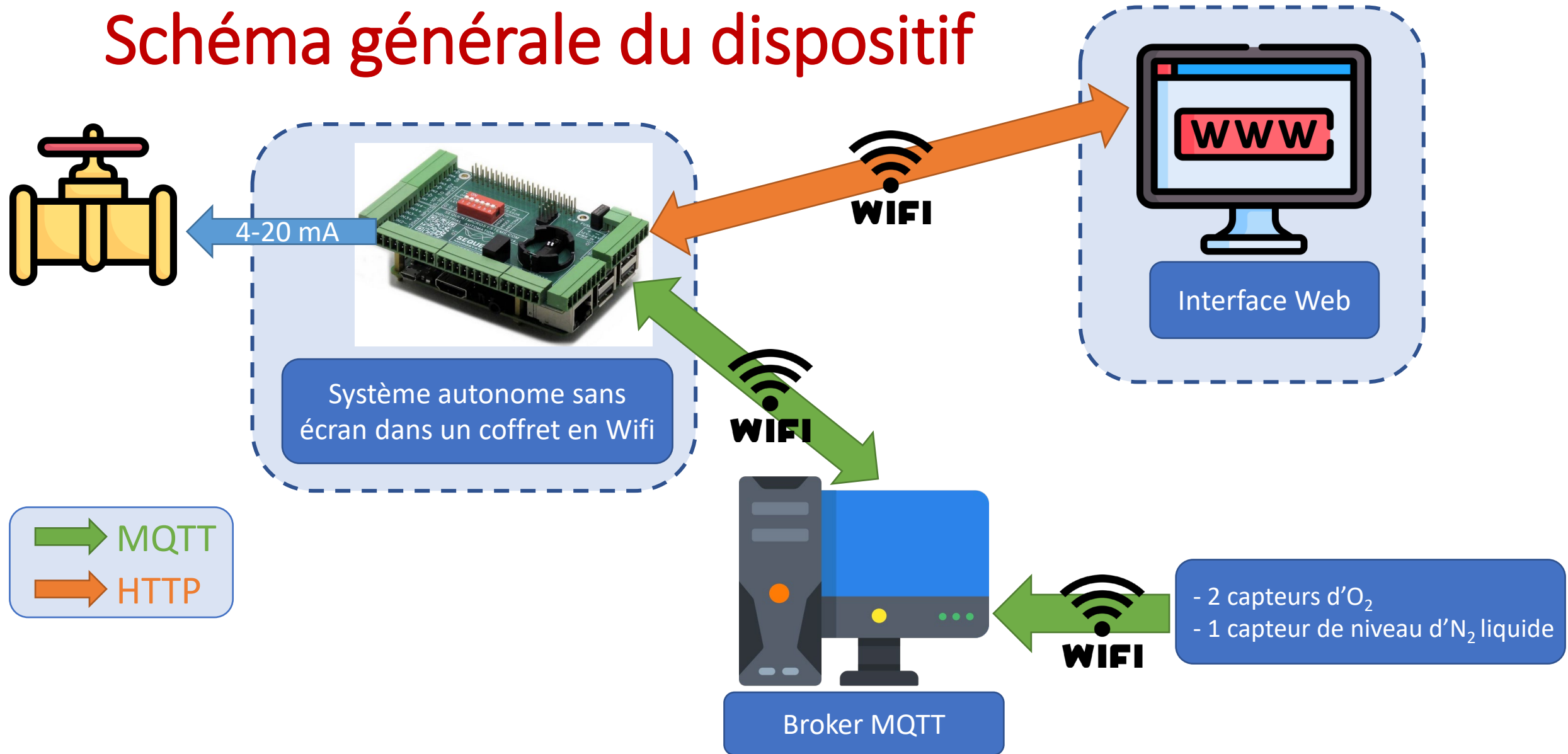


Node-RED

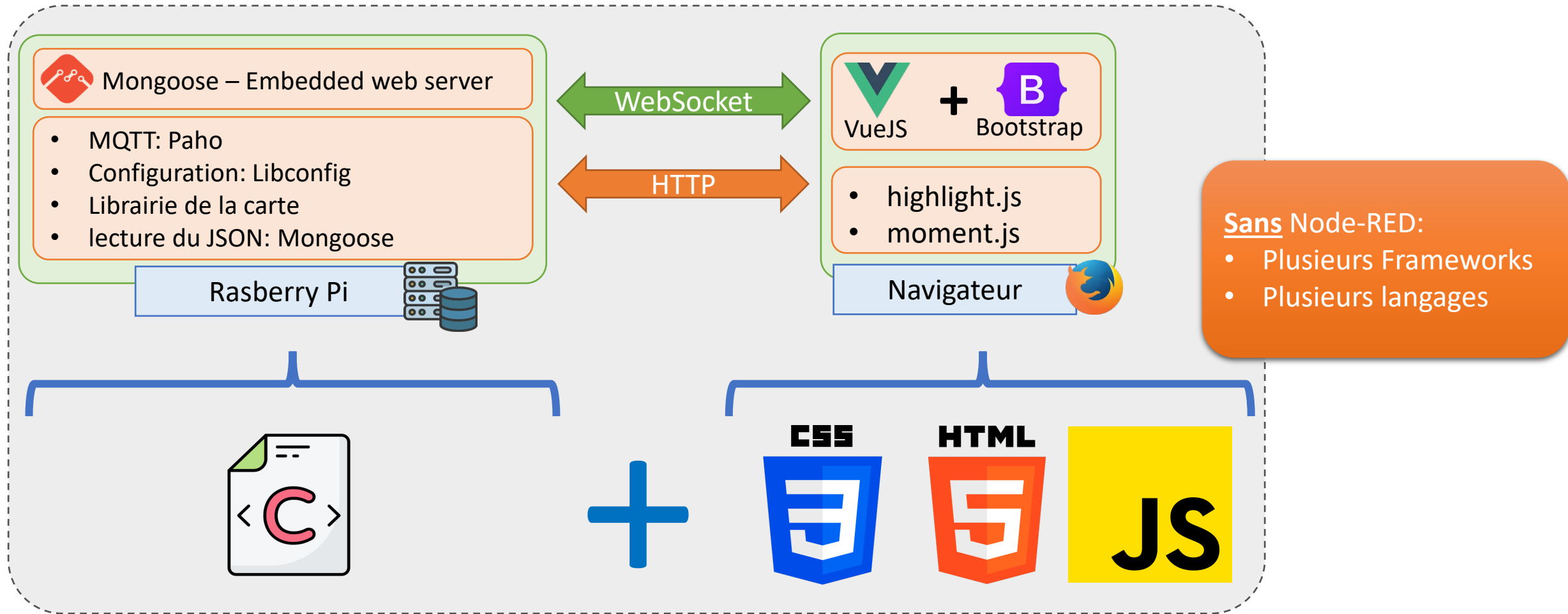
Sommaire

- Contexte
- Utilisation de Node-RED
- Dashboard 2.0
- Sous-Flux

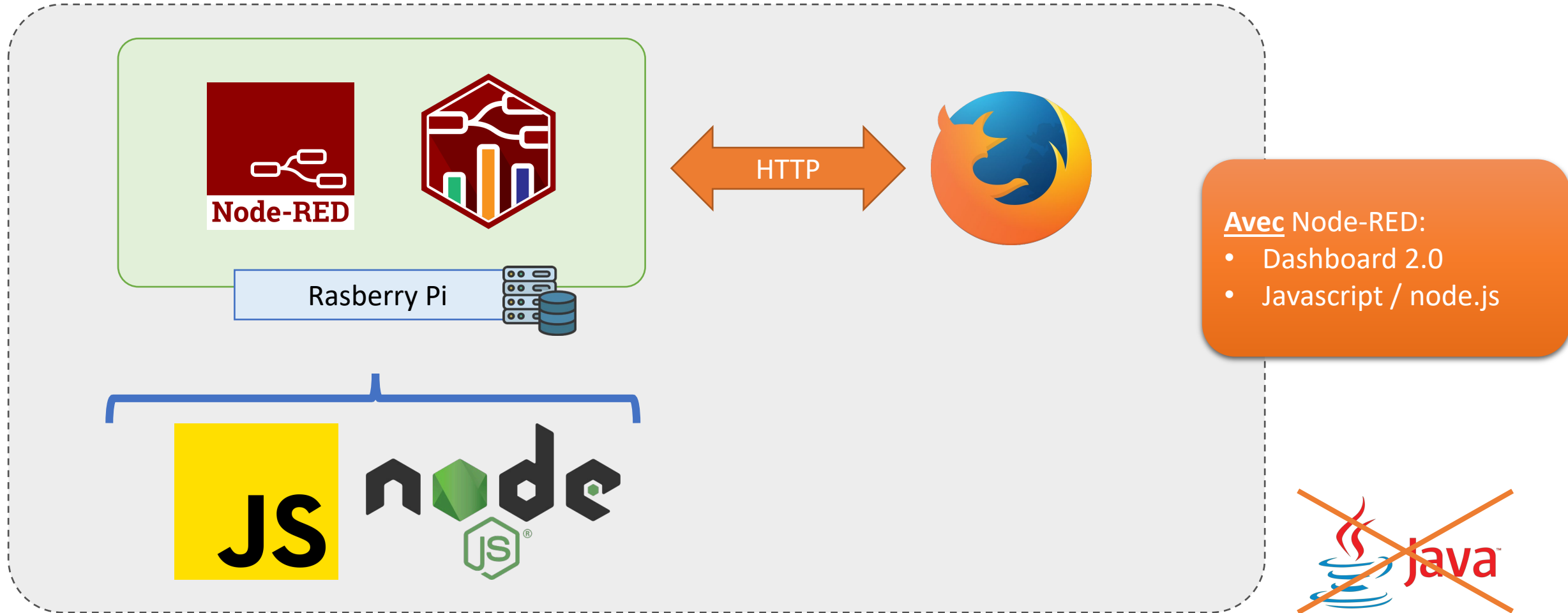
Schéma générale du dispositif



Pourquoi tester Node-RED? (1/2)



Pourquoi tester Node-RED? (2/2)



Interface

```

14 #ifndef ACTIVE_CARD
15 #define ACTIVE_CARD 1
16 #endif
17
18 static int keepRunning = 1;
19 static int clientWS = 0;
20
21 void intHandler()
22 {
23     keepRunning = 0;
24 }
25
26 static void cb(struct mg_connection *c, int ev, void *ev_data)
27 {
28     if (ev == MG_EV_HTTP_MSG)
29     {
30         struct mg_http_message *hm = (struct mg_http_message *) ev_data;
31         if(mg_match(hm->uri, mg_str("/websocket"), NULL))
32         {
33             mg_ws_upgrade(c, hm, NULL);
34             c->data[0] = 'L';
35         } else {
36             struct mg_http_serve_opts opts = {.root_dir = "web_root"};
37             mg_http_serve_dir(c, ev_data, &opts);
38         }
39     } else if (ev == MG_EV_WS_MSG) {
40         // something in ws

```

Status **WebSocket** Manuel 2026-06-12T10:04:35.626+02:00 - diff: 0.604 s

0 % Send Start Auto Expert mode

Tension: 10.00 V

Channels	Value	Last Data (s)
Oxygene 1	24%	0s
Oxygene 2	22%	0s
Level N2	9 cm	0s
Vanne	100.0% (0.00 V)	
Error	No Error	
Last Data	a few seconds ago	
Timestamp	a few seconds ago	

JSON

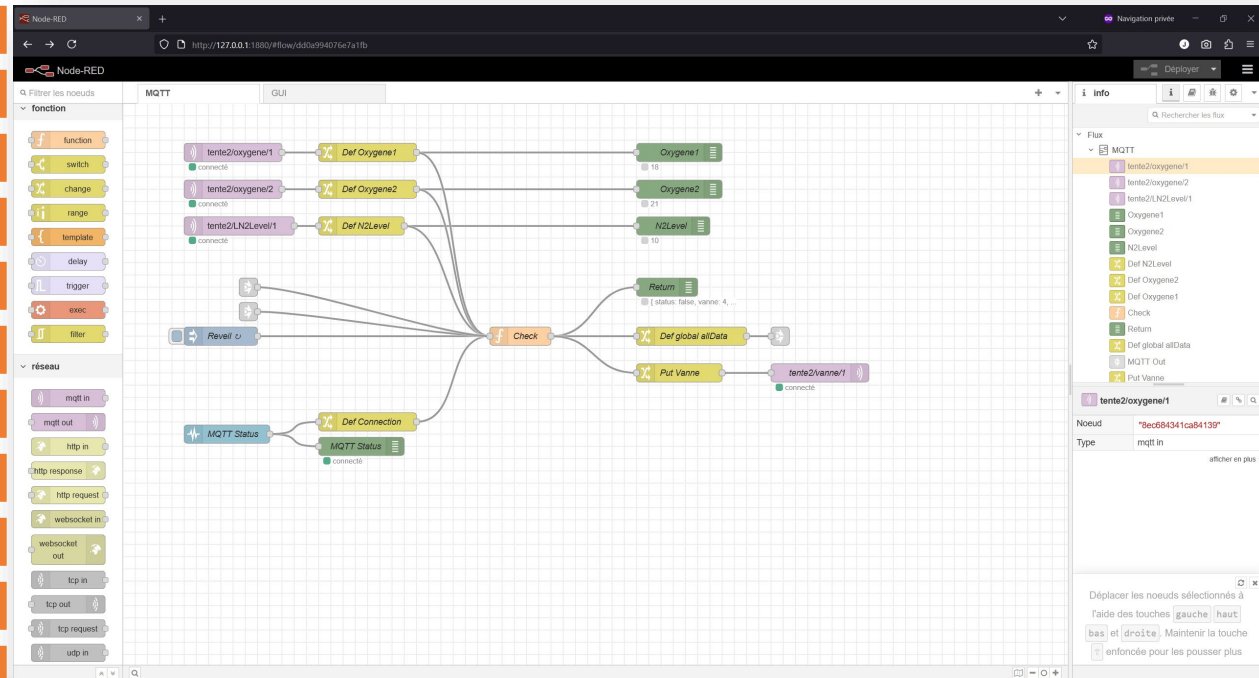
```

{
  "oxygene1": 24,
  "oxygene2": 22,
  "level": 9,
  "connection_mqtt": 1,
  "date": "2026-06-12 - 10:04:35",
  "tension": 0,
  "error": "No Error",
  "diff_oxygene1": 0,
  "diff_oxygene2": 0,
  "diff_level": 0,
  "oxygene1_timeout": 120,
  "oxygene2_timeout": 120,
  "level_timeout": 30,
  "oxygene1_limit": 19,
  "oxygene2_limit": 19,
  "level_limit": 20,
  "expert": 0,
  "clientWS": 1,

```

15/06/2026

Sans Node-RED



Vanne Dashboard

No Error Last data: a few seconds ago

Current: 4 Send clear

Oxygene 1 22 % Oxygene1: 0 s

Oxygene 2 23 % Oxygene2: 0 s

Level N2 21 cm N2Level: 0 s

Vanne 4 mA

Custom

```

{
  "status": true,
  "vanne": 4,
  "error": "No Error",
  "connection": true,
  "oxygene1": 22,
  "oxygene1_timestamp": 1780396571910,
  "oxygene1_diff": 0,
  "oxygene2": 23,
  "oxygene2_timestamp": 1780396571912,
  "oxygene2_diff": 0,
  "n2level": 21,
  "n2level_timestamp": 1780396571916,
  "n2level_diff": 0,
  "number_client": 1,
  "date": 1780396572794
}

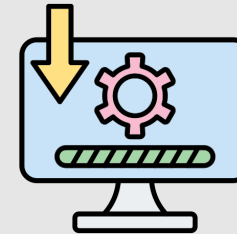
```

Avec Node-RED

6 / 23

Installation/Configuration de Node-RED

- Sur **Windows**:
 - Installer node.js + npm: <https://nodejs.org/en/download>
 - Dans un terminal: `npm install -g node-red`
- Sur **Linux**:
 - **Ubuntu/Raspbian/Debian**:
Dans un terminal:
`sudo apt install nodejs npm`
`npm install -g node-red`
 - **Fedora**:
Dans un terminal:
`sudo dnf install nodejs nodejs-npm`
`npm install -g node-red`



La **configuration de node-RED** se trouve:

- Sur **Linux**: `~/.node-red/settings.js`
- Sur **Windows**: `C:\Users\nomUtilisateur\.node-red\settings.js`



Interface de Node-RED

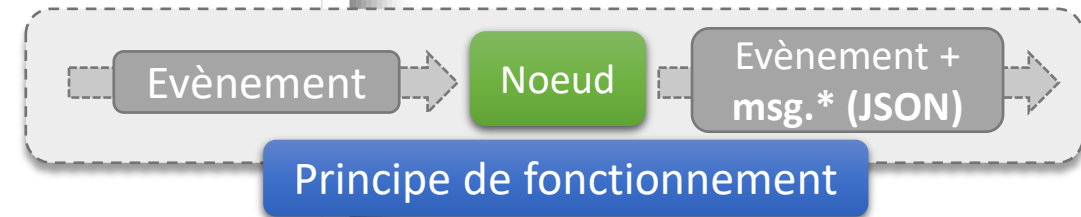
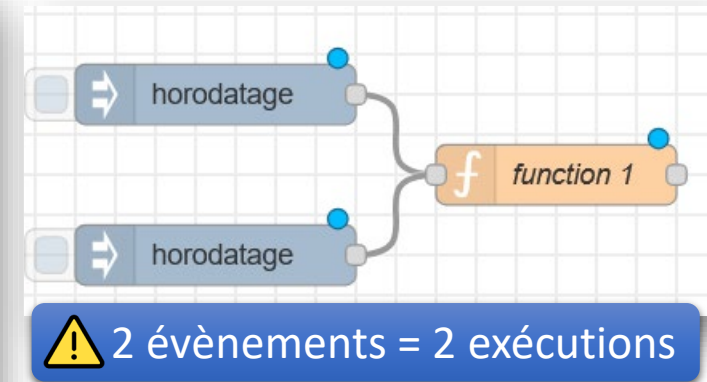
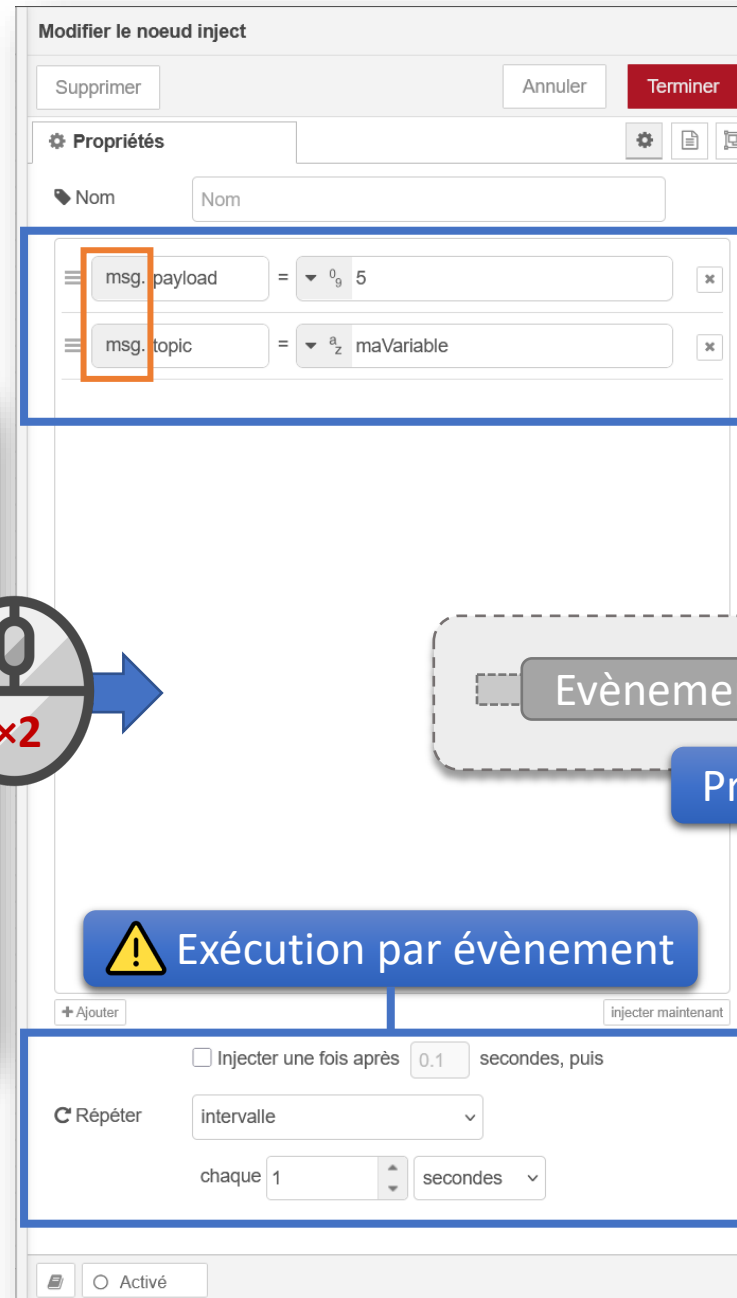
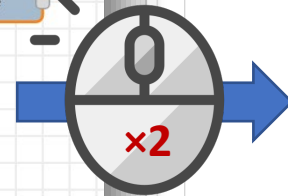
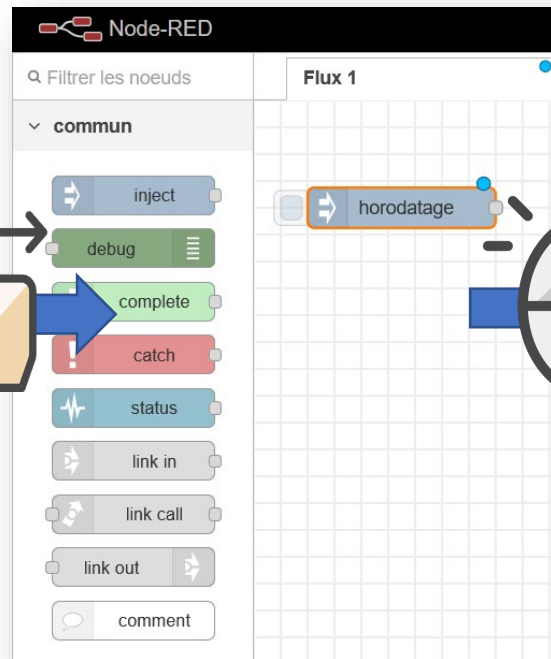
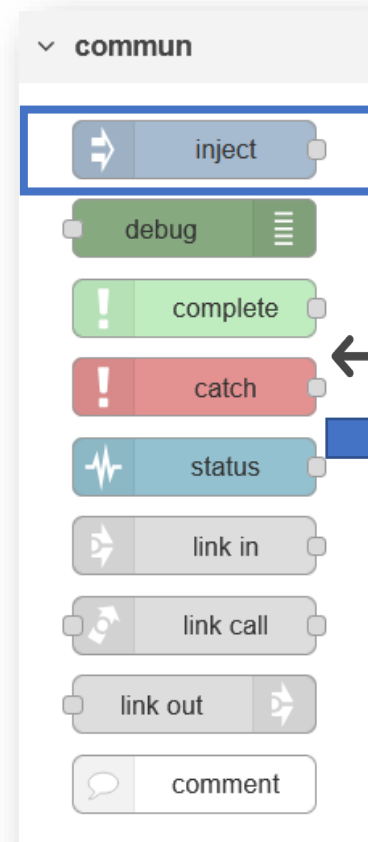
Node-RED fonctionne dans un navigateur Web

Onglets

Palette de nœud

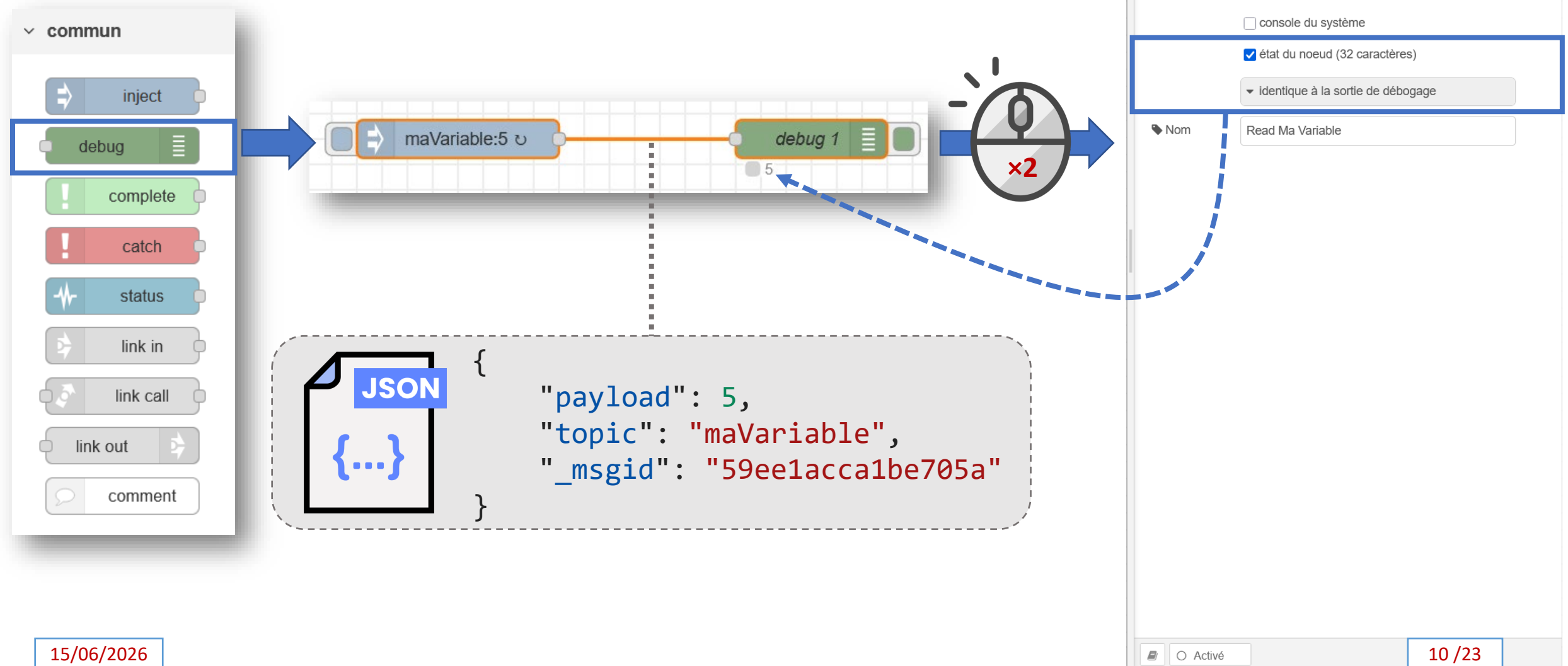
Liste des nœuds utilisés dans l'onglet

Inject Node



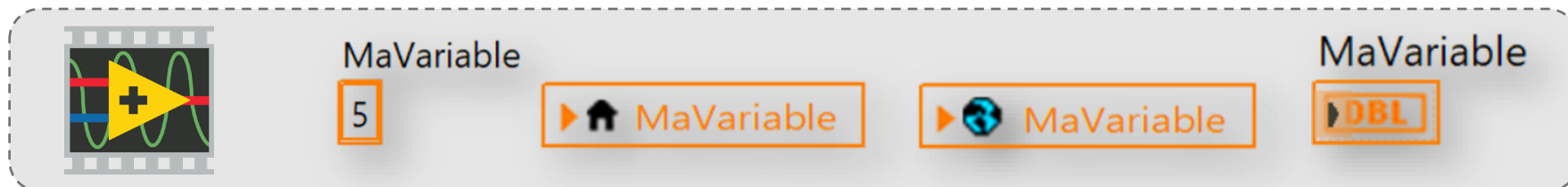
Exécution par évènement

Debug Node



Portée (scope) des variables sur Node-RED

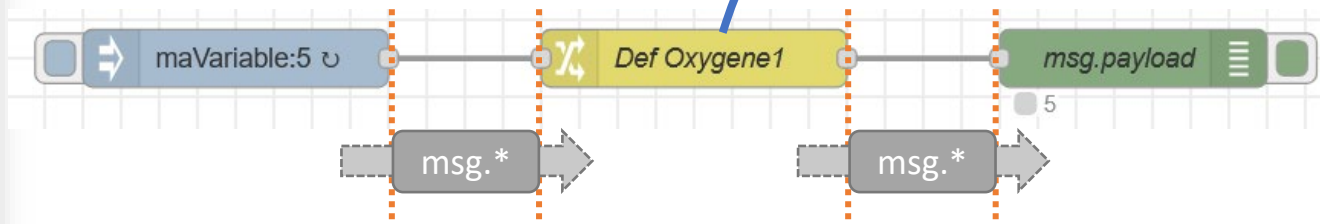
- Dans **l'ensemble de Node-RED** (variable de Node-RED):
 - **msg.*** => passage uniquement d'un nœud à l'autre
 - **flow.*** => existe uniquement dans l'onglet courant
 - **global.*** => existe de partout
 - **context.get**("...") et **context.set**("...", 2) => existe au niveau d'un nœud
 - **env.get**("...") => existe au niveau d'un sous-flux
- Dans le cas spécifique des **fonctions "custom"** (spécifique à node.js/JS):
 - **let** et **const** : la portée est au niveau du bloc "{}" et à l'intérieur de la fonction
 - **var** : la portée est au niveau de la fonction



Change Node

▼ fonction

- function
- switch
- change**
- range
- template
- delay
- trigger
- exec
- filter



▼ flow. Oxygene1

- msg.
- flow.
- global.

Modifier le noeud change

Supprimer Annuler Terminer

Propriétés

Item Def Oxygene1

Règles

Définir ▼ ▼ flow. Oxygene1

sur la valeur ▼ msg. payload

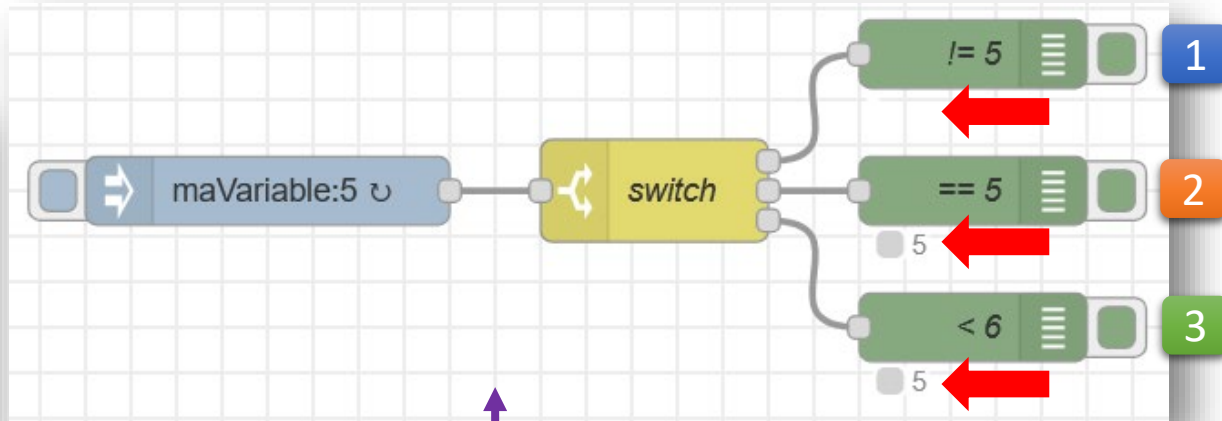
☐ Copie profonde de la valeur

flow.Oxygene1 = msg.payload

Ajouter

Activé

Switch Node



```
if(maVariable != 5)
{
    // Aller à 1
} else if(maVariable == 5)
{
    // Aller à 2
} else if(maVariable < 6)
{
    // Aller à 3
}
```

vérifier toutes les règles
vérifier toutes les règles
arrêter après la première concordance

Modifier le noeud switch

Supprimer Annuler Terminer

Propriétés

Nom switch

Propriété msg. payload

!=	5	→ 1
==	5	→ 2
<	6	→ 3

+ Ajouter

vérifier toutes les règles

☐ recréer des séquences du messages

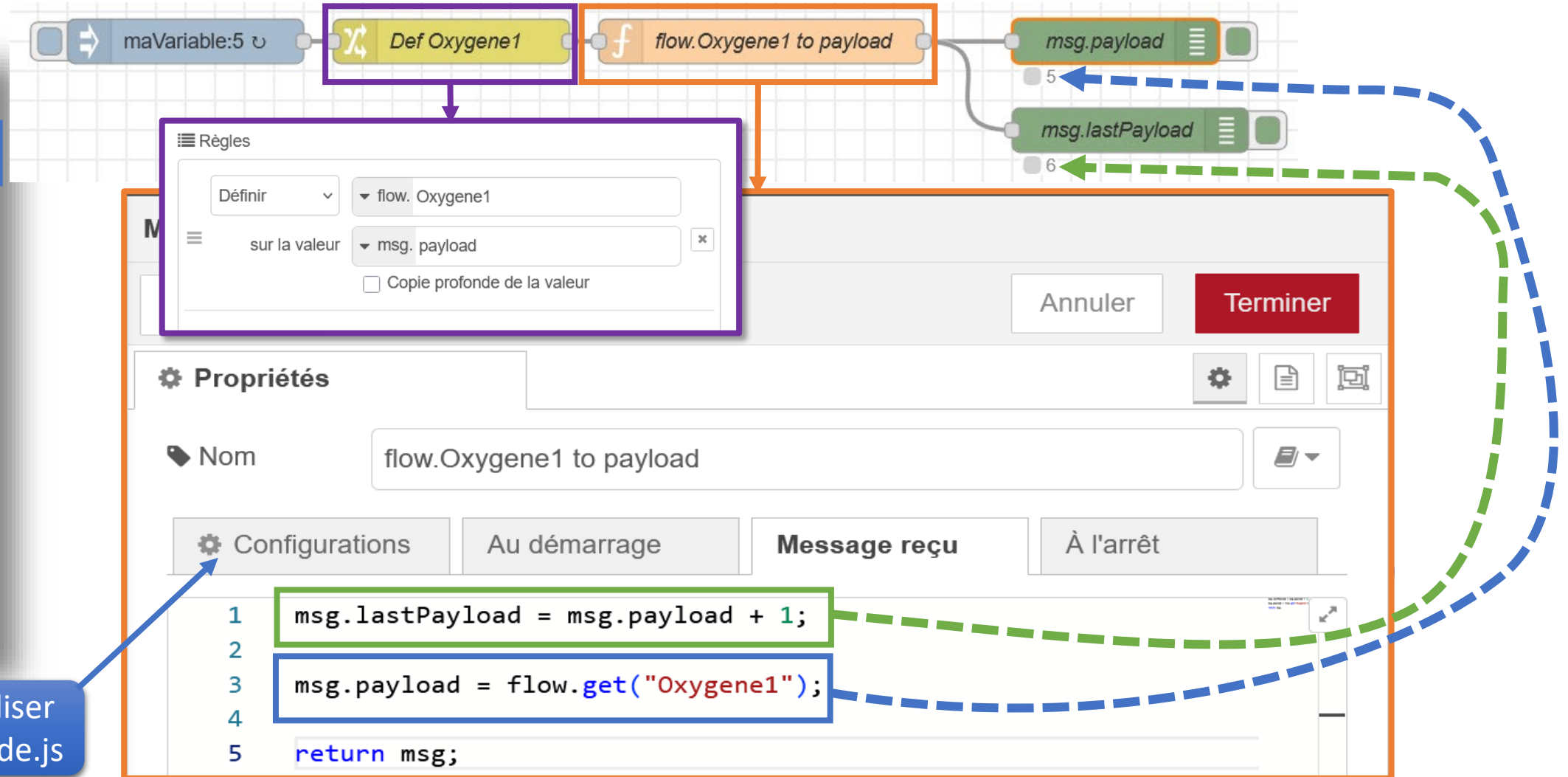
Function Node



```
node.warn("mon debug")
```

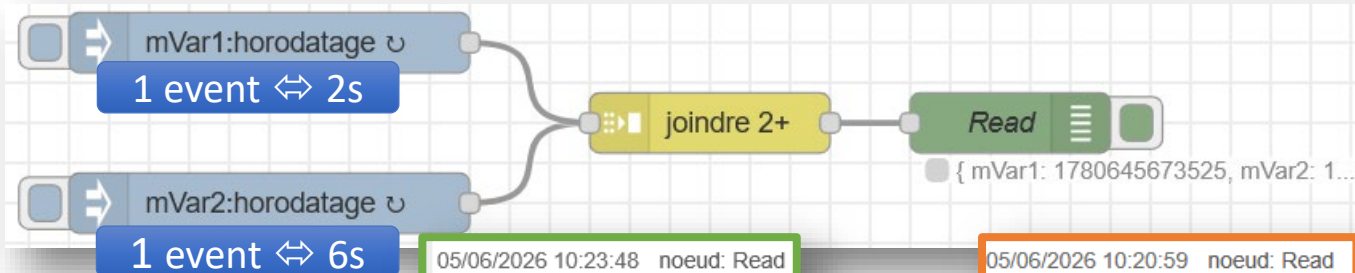


```
node.status({fill: "red", shape: "dot", text: "timeout"})
```

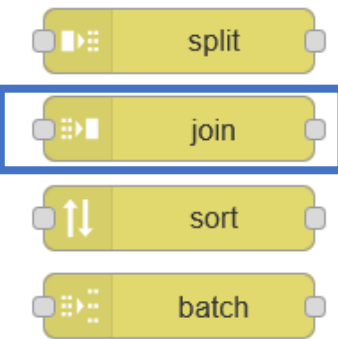


Possibilité d'utiliser
des modules node.js

Join Node



séquence



☒ et tous les messages suivants.



05/06/2026 10:23:48 noeud: Read
mVar1 : msg.payload : Object

▼ object

mVar1: 1780647828718

mVar2: 1780647828703

05/06/2026 10:23:50 noeud: Read
mVar1 : msg.payload : Object

▼ object

mVar1: 1780647830729

mVar2: 1780647828703

05/06/2026 10:23:52 noeud: Read
mVar1 : msg.payload : Object

▼ object

mVar1: 1780647832734

mVar2: 1780647828703

05/06/2026 10:23:54 noeud: Read
mVar2 : msg.payload : Object

▼ object

mVar1: 1780647832734

mVar2: 1780647834712

2s

6s

05/06/2026 10:20:59 noeud: Read
mVar2 : msg.payload : Object

▼ object

mVar1: 1780647657092

mVar2: 1780647659054

05/06/2026 10:21:05 noeud: Read
mVar2 : msg.payload : Object

▼ object

mVar1: 1780647663123

mVar2: 1780647665058

05/06/2026 10:21:11 noeud: Read
mVar2 : msg.payload : Object

▼ object

mVar1: 1780647669157

mVar2: 1780647671072

05/06/2026 10:21:17 noeud: Read
mVar2 : msg.payload : Object

▼ object

mVar1: 1780647675190

mVar2: 1780647677078

Modifier le noeud join

Supprimer

Annuler

Terminer

Propriétés

Mode

manuel

Combine each

msg. payload

créer

un Objet clé/valeur

en utilisant la valeur de

msg. topic

comme la clé

Envoyer le message :

Après un certain nombre de parties du

• message

2

☒ et tous les messages suivants.

Après un délai d'attente après le premier

• message

secondes

• Après un message avec la propriété **msg.complete** définie

Nom

Nom

Autre Node

commun

- inject
- debug
- complete
- catch
- status**
- link in
- link call
- link out
- comment

analyseur

- 1,2 CSV
- html
- json
- xml
- yaml

Modifier le noeud status

Supprimer Annuler Terminer

Propriétés

Signaler l'état de noeuds sélectionnés

Sélectionner les noeuds...

☒	tente2/oxygene/1	mqtt in
☒	tente2/oxygene/2	mqtt in
☒	tente2/LN2Level/1	mqtt in
☐	Oxygene1	debug
☐	Oxygene2	debug
☐	N2Level	debug
☐	Def N2Level	change
☐	Def Oxygene2	change
☐	Def Oxygene1	change
☐	Check	function
☐	Return	debug

réseau

- mqtt in
- mqtt out
- http in
- http response
- http request
- websocket in
- websocket out
- tcp in
- tcp out
- tcp request
- udp in
- udp out

stockage

- écrire le fichier
- lire le fichier
- watch

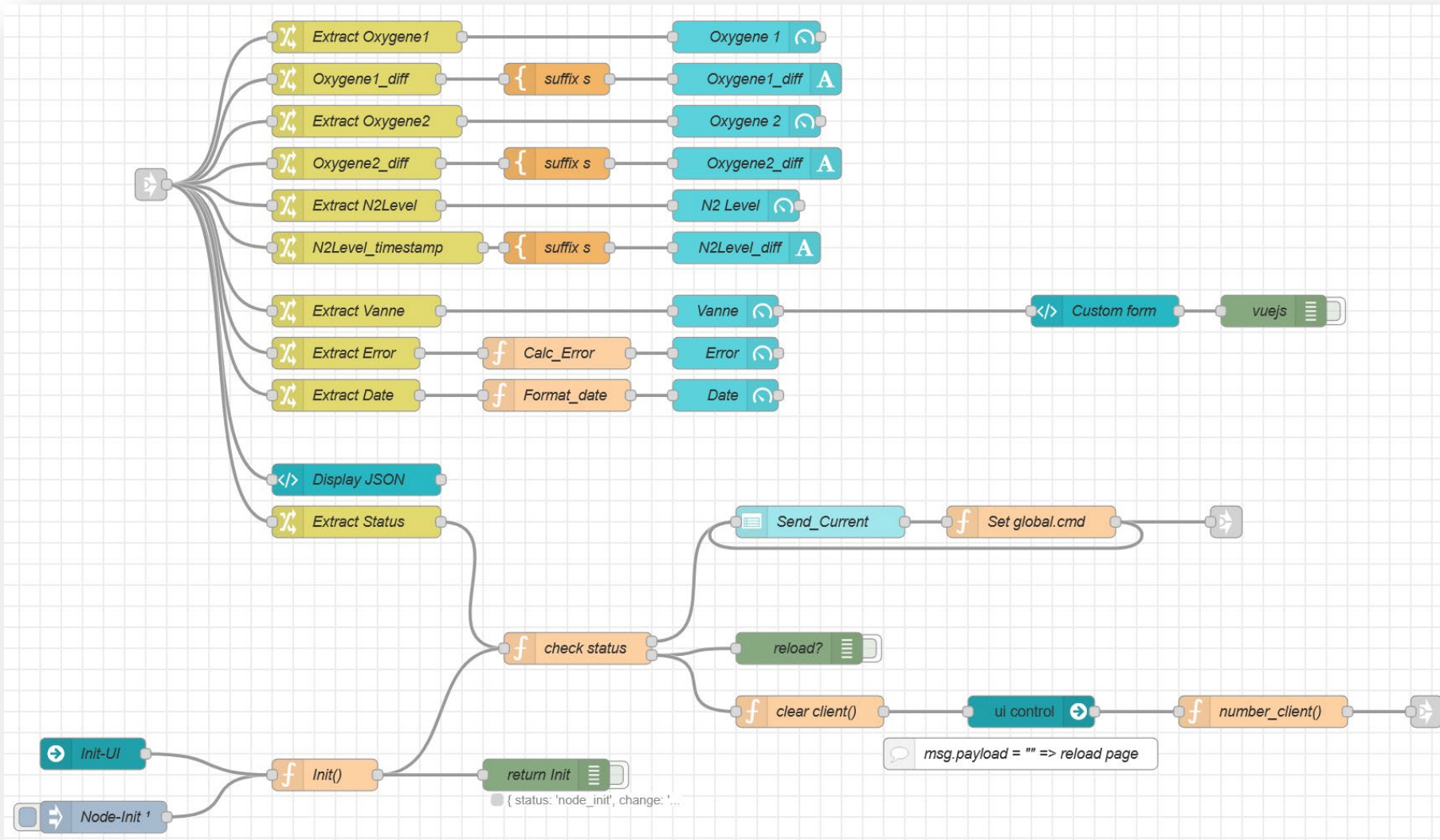
Réception des msg.* venant d'un autre onglet

Envoie des msg.* vers un autre onglet

Possibilité d'ajouter d'autres nœuds
Exemple: Dashboard 2.0

>- @flowfuse/node-red-dashboard

Dashboard 2.0 (1/4)



Dashboard 2.0 (2/4)

dashboard 2

form

text input

number input

file input

button

button group

dropdown

radio group

slider

switch

text

table

chart

gauge

notification

audio

markdown

template

event

ui control

progress

Vanne Dashboard

http://127.0.0.1:1880/dashboard/page1

No Error

Last data: a few seconds ago

Current

4

Send

clear

Oxygene 1

22

%

Oxygene 2

23

%

Level N2

21

cm

Vanne

4

mA

Oxygene1: 0 s

Oxygene2: 0 s

N2Level: 0 s

```
{
  "status": true,
  "vanne": 4,
  "error": "No Error",
  "connection": true,
  "oxygene1": 22,
  "oxygene1_timestamp": 1780396571910,
  "oxygene1_diff": 0,
  "oxygene2": 23,
  "oxygene2_timestamp": 1780396571912,
  "oxygene2_diff": 0,
  "N2Level": 21,
  "N2Level_timestamp": 1780396571916,
  "N2Level_diff": 0,
  "Number_client": 1,
  "Date": 1780396572794
}
```

Custom

Half Gauge

Tile

3/4 Gauge

Half Gauge

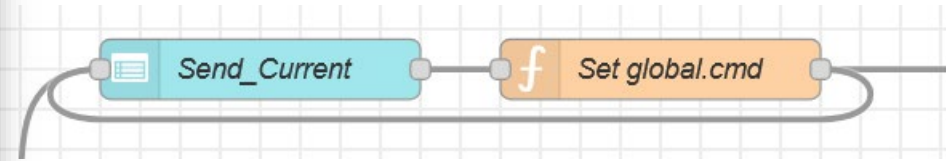
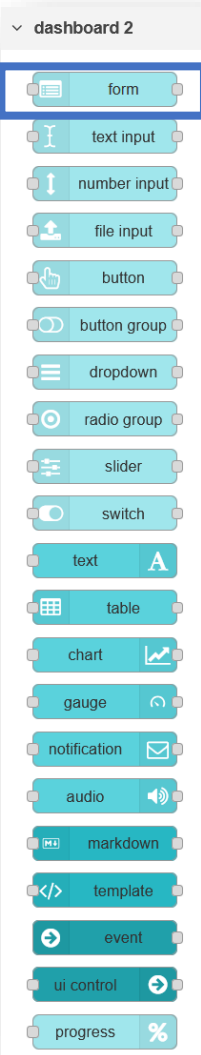
Battery Level

Tank Level

15/06/2026

18 /23

Dashboard 2.0 (3/4)



* Current

4

Send clear

Propriétés

Nom: Send_Current

Group: [Vanne Dashboard] Command 2

Size: auto

Label: optional label

Class: Optional CSS class name(s)

Form

Form elements

Label	Nom	Type	Required	Rows
Current	Current	Number	☒	1

Ajouter

Buttons

Send clear

☒ Place the form elements in two columns

☐ Reset the form when submitted

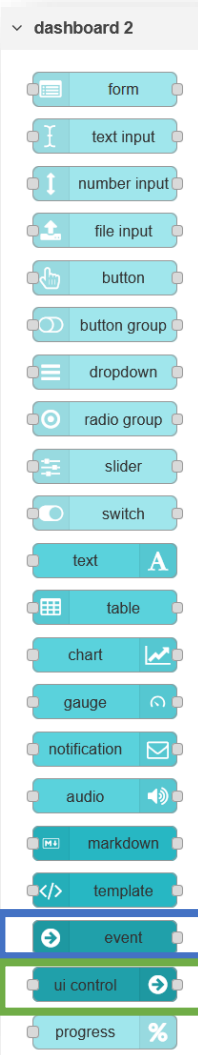
Topic: msg. topic

N'existe pas dans les "form" dans le Dashboard 2.0 ???



```
<input type="number" max="20" min="4" name="Current" />
```

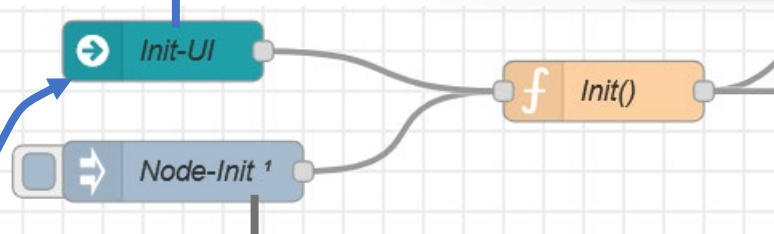
Dashboard 2.0 (4/4)



Initialisation du Dashboard

- `$pageview` - Emits whenever a user views a page.
- `$pageleave` - Emits whenever a user leaves a page.

msg.topic



☒ Injecter une fois après secondes, puis

↳ Répéter

Initialisation de Node-RED

↳ Output

All Events

Page/Tab Change Events Only

Connection Events Only

ui-control

Avec ces différents nœuds:

- Il est possible de savoir le nombre de "client"
- Un nouveau "client"
- Un "client" quitte la page
- Quand Node-RED et le Dashboard 2.0 est chargés



Initialisation Node-RED



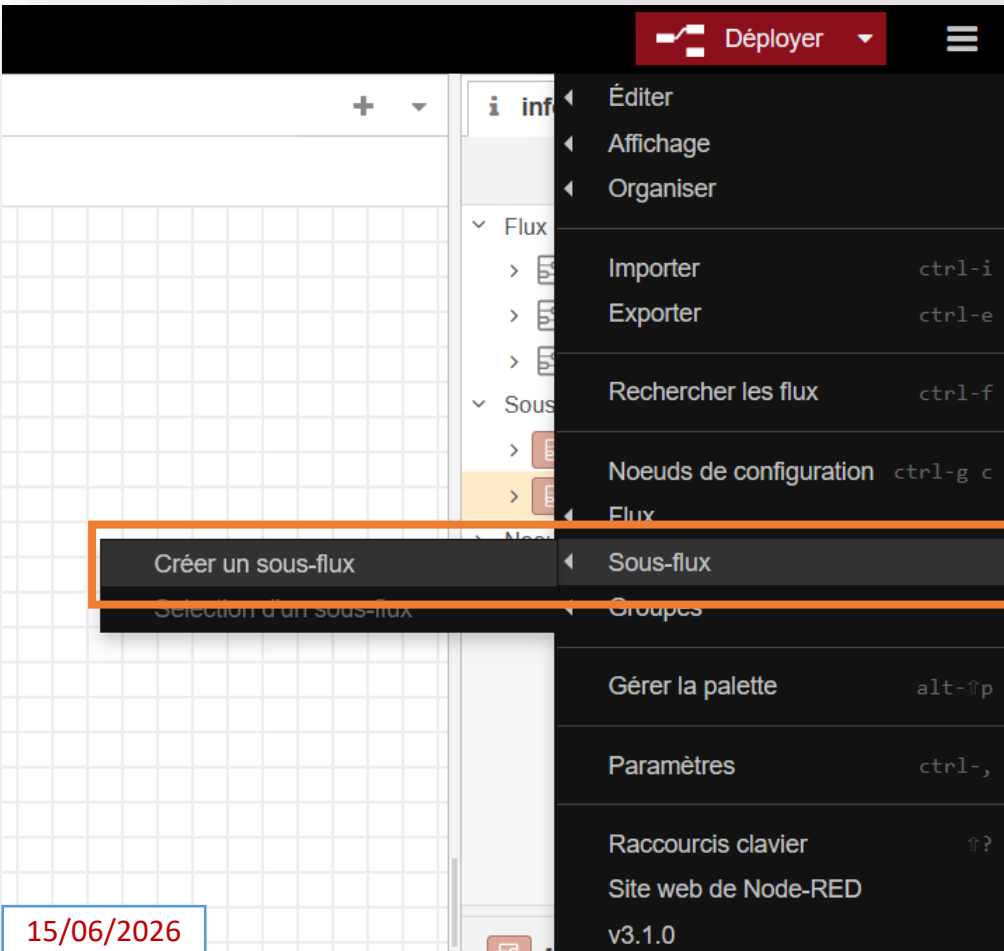
Initialisation Dashboard 2.0

Sous-Flux (1/2)

 modifier les propriétés

entrées:

sorties: ☐ statut du noeud  supprimer le sous-flux

 Déployer

Créer un sous-flux

Selection d'un sous-flux

Sous-flux

Grouper

Gérer la palette alt-tp

Paramètres ctrl-,

Raccourcis clavier tr?

Site web de Node-RED

v3.1.0

Modifier le modèle du sous-flux : Average

Annuler Terminer

Propriétés

Nom Average

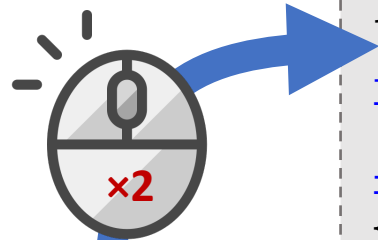
Variables d'environnement Aperçu de l'interface utilisateur

Nom	Valeur par défaut
size	5

+ Ajouter

21 / 23

Sous-Flux (2/2)



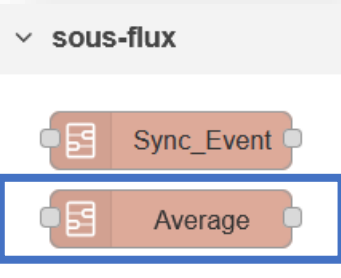
```
if(typeof msg.payload !== "number")
{
  node.warn("Invalid type");
  return null;
}

let data;

if(context.get("data") === undefined)
{
  context.set("data", Array());
  data = Array();
} else {
  data = context.get("data");
}

if ((data.length + 1) > env.get("size"))
{
  data.shift();
}

data.push(msg.payload);
msg.payload = data.reduce((sum, value) => sum + value, 0)/data.length;
msg.topic = "average";
context.set("data", data);
return msg;
```



Conclusion

- Node-RED permet un **prototypage très rapide**
- Nécessite une **gymnastique d'esprit** avec le système d'évènement
- Pour le **Dashboard 2.0 parfois perfectible**, chose simple en HTML5 pas facile avec le Dashboard
- Pour un projet plus gros, le **débogage me paraît infernale**
- J'utilise **beaucoup de nœud "function"** => problème de maîtrise ou mauvaise habitude ???
- Utiliser le mode "projet" (<https://nodered.org/docs/user-guide/projects/>)
- Une **installation qui peut paraître inhabituelle** sur Windows (npm, node.js)