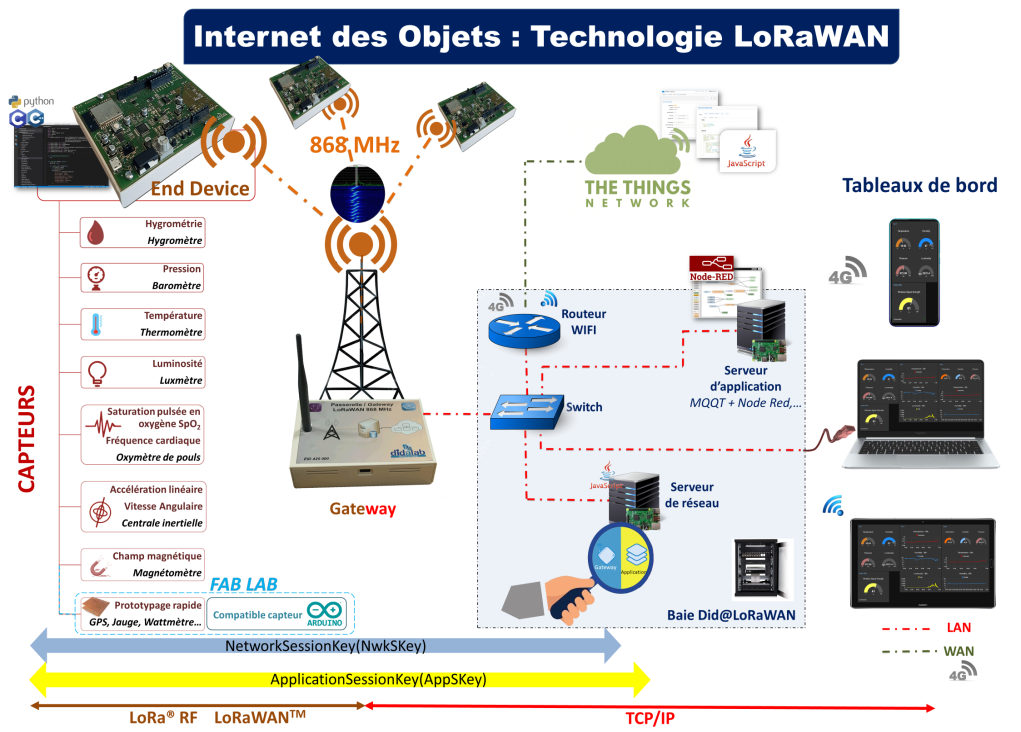
**Mise en œuvre d’une balance connectée**

Une définition possible de l’Internet des Objets ou IoT (Internet of Things) est donnée ci-après :

***« L’Internet des Objets est un dispositif pour lequel les Objets conçus par l’Homme, mais aussi les animaux et les personnes, se voient attribués d’identifiants uniques leur permettant de transmettre des données sur le réseau Internet sans nécessiter aucune action humain à humain ni humain à machine ».***

Le système support de l’épreuve de Travaux Pratiques, illustrant la thématique de l’Internet des Objets, est représenté ci-dessous :



Dans cet exemple, trois cartes « **End Device** » équipées de 7 capteurs chacune, transmettent à une passerelle (Gateway) et par voie hertzienne selon le protocole radio **LoRa**, les informations significatives des grandeurs physiques auxquelles ils sont sensibles.

La passerelle est connectée à la baie informatique **Did@LoRaWAN**, qui contient un switch, un routeur autorisant un accès au réseau Internet, un serveur de réseau et un serveur d’application.

Les données issues des capteurs implantés sur les cartes End Device sont transmises périodiquement (toutes les 2 secondes), soit au serveur de l’hébergeur « The Things Network » (TTN), via la passerelle, le switch et le routeur, soit au serveur de réseau intégré à la baie. Il est alors possible de visualiser ces données, soit en mode distant depuis le site de TTN, soit en mode local depuis le serveur de réseau.

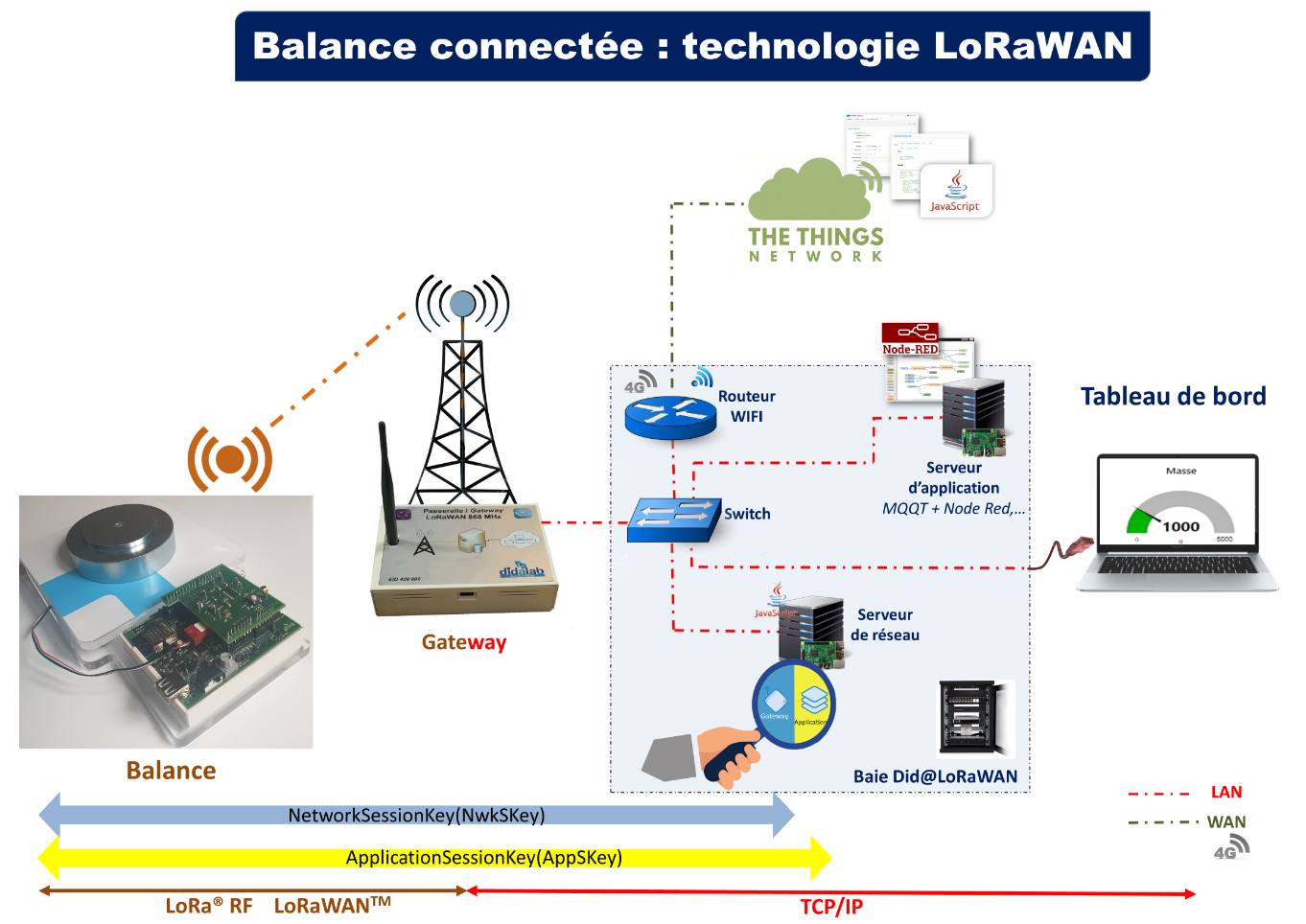
La baie contient par ailleurs un serveur d’application qui permet de récupérer les données brutes hébergées par le serveur de TTN ou bien par le serveur de réseau de la baie, afin de bâtir une application correspondant aux besoins d’un cahier des charges spécifique (comme par exemple, afficher des tableaux de bord).

Comme indiqué sur le synoptique de la page précédente, les cartes « **End Device** » sont équipées de 4 connecteurs au format Shield Aduino.

Nous avons donc mis à profit ces connecteurs pour concevoir et réaliser une carte fille venant s’insérer sur la carte « **End Device** », permettant de réaliser une balance connectée :



Le synoptique de l’application **balance connectée** est alors le suivant :



**Objectif de l’épreuve de travaux pratiques**

On se propose de vérifier la capacité du dispositif à successivement :

* élaborer une grandeur numérique significative de la valeur de la masse du poids positionné sur la balance (**balance**) ;
* transmettre cette grandeur par voie radio à la passerelle, puis à destination du serveur de réseau intégré à la baie (**passerelle**, **switch**, **routeur**) ;
* convertir les données mémorisées dans le serveur de réseau en tableaux de bord (**routeur**, **switch**, **serveur d’application**, **terminal de supervision**).

1. **Caractérisation de la balance**

Documents ressources mis à disposition (dossier « Documents ressources » du bureau du PC) :

* Schéma\_carte\_balance.pdf ;
* PCB\_carte\_balance.pdf ;
* Documentation\_cellule\_5kg.pdf ;
* Documentation HX711.pdf.

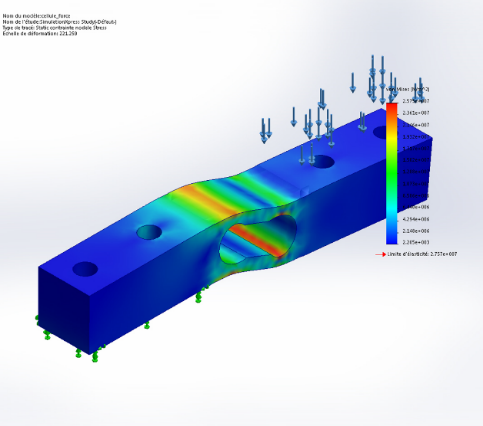
**A1. La jauge de déformation**

La mesure de la masse s’effectue de façon classique à l’aide d’un corps d’épreuve sur lequel sont collées 4 jauges de déformation notées ici J1 à J4.

Le montage, visible sur la balance, est réalisé comme suit :



On donne par ailleurs, ci-contre, une simulation sous Solideworks de la déformation du corps d’épreuve, lorsqu’il est soumis à un moment de forces :



On note respectivement R1 à R4, la résistance nominale des jauges J1 à J4 lorsque la balance n’est pas chargée (à vide).

On pose un objet de masse m et de poids P sur le plateau de la balance, juste au-dessus de la vis assurant la fixation du plateau supérieur sur le corps d’épreuve :



1. Sachant que la résistance des jauges en extension augmente (Ri + ΔRi), alors qu’inversement, la résistance des jauges en compression diminue (Ri - ΔRi), indiquer, lorsqu’on pose l’objet sur le plateau de la balance, comment varie la résistance de chacune des 4 jauges par rapport à leur valeur nominale R1 à R4.

**A2. Le pont de Wheatstone**

Les 4 jauges de déformation de résistances nominales R1 à R4 sont insérées dans un pont de Wheatstone comme suit :



On montre que la tension différentielle de sortie du pont de Wheatstone peut s’écrire sous la forme :

1. Indiquer le sens de variation de la tension Vdiff lorsque la masse mesurée augmente.

**A3. Caractérisation expérimentale de l’ensemble [ jauges + pont de Wheatstone ]**

Connecter l’ensemble (carte mère « End Device » + carte fille « balance ») sur un port USB du PC et mettre l’ensemble sous tension à l’aide de l’interrupteur M/A positionné sur la carte « End Device »).

1. Mesurer à l’oscilloscope la tension différentielle Vdiff = VA+ - VA- (voir schéma structurel et schéma d’implantation de la carte) avec une précision au 10ème de mV, en faisant varier la masse m entre 0 et 5 kg par pas de 1 kg. Vérifier la conformité des résultats expérimentaux par rapport à la réponse apportée à la question précédente. Exprimer la relation Vdiff = f(m).

Aide : le signal étant bruité, on configurera l’oscilloscope de telle façon qu’il effectue une mesure de la valeur moyenne (mean en anglais) de la voie Math = C2 – C1, en réglant la base de temps sur 500 ms/div.

1. Mesurer la tension d’alimentation AVDD du pont.

**A4. Numérisation du capteur de masse**

La tension différentielle Vdiff est convertie en une grandeur numérique par le composant HX711 dont la documentation est donnée en annexe. Cette grandeur numérique sera notée Nb dans ce qui suit.

1. Compte tenu des valeurs de Vdiff mesurées précédemment, choisir la valeur du gain du composant HX711 permettant d’obtenir le quantum de conversion analogique-numérique le plus faible possible.

Lancer l’environnement de développement **Visual Studio Code**  depuis la barre des tâches.

Charger l’application « **balance** » disponible sur U:\didalab\eid430\Balance.

Pour cela, sélectionner à l’aide des onglets ***File*** puis ***Open Folder*** le dossier U:\didalab\eid430\Balance.

Vérifier dans la barre des tâches que GCC 5.2.0 xtensa-esp32-elf est bien activé (si tel n’est pas le cas, cliquer 2 fois sur la mention no kit selected et sélectionner dans le menu déroulant proposé GCC 5.2.0 xtensa-esp32-elf).

Lancer la construction de l’application en cliquant sur Build (barre des tâches), et vérifier qu’elle se temine par la mention Build finished with exit code 0 .

Il reste à charger le code de l’application « **balance** » dans la mémoire flash du µC de la carte.

Pour cela, sélectionner l’onglet **TERMINAL** de la barre d’outils de la fenêtre de commandes et de débogage, puis compléter comme suit :PS U:\didalab\eid430\Balance> eid430.py flash.

Une fenêtre s’ouvre alors quelques secondes, le temps que le code soit mémorisé dans le µC.

On se propose de compléter une partie du code C permettant d’acquérir le nombre Nb significatif de la masse positionnée sur le plateau de la balance.

Editer la fonction **hx711.c** (il suffit de cliquer sur l’onglet **C** hx711.c sur la barre des tâches de la fenêtre d’édition).

On s’intéresse dans un premier temps à la fonction « **lire-jauge()** » dont le rôle consiste à procéder à la conversion analogique-numérique de la tension Vdiff.

Cette fonction renvoie, via la variable entière **count** codée sur 32 bits, le nombre Nb défini précédemment.

1. En vous appuyant sur la documentation du composant HX711, compléter le code de la fonction « **lire\_jauge()** » permettant de satisfaire le cahier des charges précisé à la question précédente.
2. Visualiser à l’oscilloscope les sorties du composant HX711 **PD\_SCK** et **DOUT**, afin de vérifier la pertinence du code proposé. Observer, pour différentes valeurs de masses, la sortie **DOUT** et vérifier que le nombre Nb augmente au fur et à mesure que la masse augmente. Justifier la raison pour laquelle, pour une masse donnée, le nombre Nb n’est pas stable.

**Aide** : la mesure étant délicate à effectuer, on suggère de déclencher l’oscilloscope en mode normal et de le synchroniser sur une durée (width) au niveau bas du signal d’horloge supérieure à 100 ms.

Puisque le nombre Nb n’est pas stable, nous décidons d’implanter la fonction « **lire\_moyenne()** » dont le rôle consiste à effectuer 5 lectures de Nb successives, puis à en calculer la valeur moyenne.

Cette fonction renvoie donc, au travers de la variable entière **moy** codée sur 32 bits, le nombre Nb moyen.

1. Proposer le codage de la fonction « **lire\_moyenne()** ».

Editer la fonction **Gestion\_Balance.c**.

On trouve dans cette fonction, le codage de la tâche permettant :

* d’assurer la gestion du dispositif de tarage de la mesure, via le bouton poussoir **BP1** ;
* de calculer et d’exprimer la masse mesurée en kg (variable réelle **m\_kg**).

L’instruction printf(" m=%x conv=%x ref=%x m\_kg=%1.5f\n ", mass, moy, ref\_val, m\_kg) ; permet par ailleurs de visualiser le contenu des variables **mass**, **moy** (Nb moyen), **ref\_val** et **m\_kg**.

1. Indiquer à quoi correspondent les grandeurs associées aux variables **mass**, **ref\_val** et **m\_kg**.

On propose désormais d’étalonner la balance.

Pour cela, entrer la commande PS U:\didalab\eid430\Balance> eid430.py monitor ayant pour effet d’ouvrir une fenêtre de débogage permettant de visualiser le contenu des variables **mass**, **moy**, **ref\_val** et **m\_kg**.

1. Agrandir la fenêtre de débogage, positionner une masse de 2 kg, et modifier la valeur du rapport de division de l’instruction m\_kg = ((float)mass)/500000; (ligne de programme n°82) jusqu’à ce que la fenêtre de débogage indique m\_kg=2.000xx (on demande une précision au gramme près).

**Remarque :** après modification du code de la fonction Gestion\_Balance.c (ici, plus précisément, de la valeur du rapport de division fixé initialement à la valeur 500000), il faut successivement :

* enregistrer la modification (File puis Save) ;
* fermer la fenêtre de débogage (le cas échéant) ;
* charger le programme modifié (commande PS U:\didalab\eid430\Balance> eid430.py flash) ;
* retirer la masse de 2 kg et faire la tare par appui sur le bouton poussoir BP1 ;
* repositionner la masse sur la balance ;
* réouvrir la fenêtre de débogage (commande PS U:\didalab\eid430\Balance> eid430.py monitor) ;
* lire la valeur de la variable m\_kg affichée dans la fenêtre de débogage.

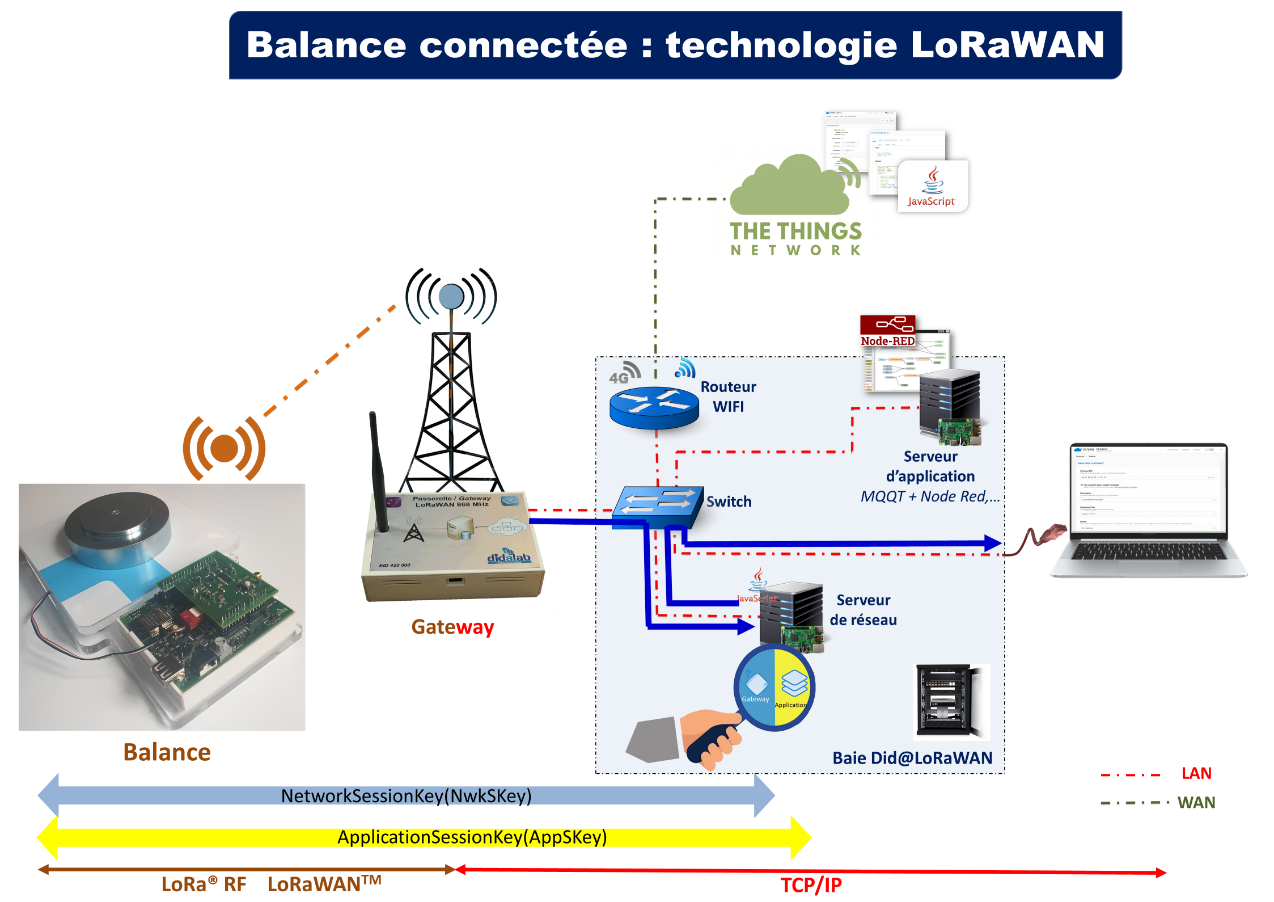
1. Une fois le réglage effectué, faire varier la masse m de 0 à 5 kg par pas de 1 kg, et vérifier que la mesure demeure à chaque fois précise au gramme près.
2. Modifier, pour terminer, la valeur de la constante (10) de l’instruction data\_balance=(int)(10\*m\_kg); (ligne n°92) permettant d’affecter à la variable **data\_balance**, la masse mesurée exprimée en g.
3. **Caractéristiques spectrales du signal radio LoRa**

La norme radio **LoRa** permet une transmission sur 8 canaux distincts positionnés de part et d’autre des fréquences centrales 867,1 MHz / 867,3 MHz / 867,5 MHz / 867,7 MHz / 867,9 MHz / 868,1 MHz / 868,3 MHz / 868,5 MHz, chacun d’entre eux occupant une largeur de 125 kHz.

1. Visualiser à l’analyseur de spectre le spectre d’amplitude du signal radio en couvrant la bande allant de 866 MHz à 869 MHz, et repérer visuellement le canal de communication utilisé. Indiquer la raison pour laquelle l’affichage du spectre d’amplitude s’effectue par intermittence.

Afin d’améliorer le confort de lecture, on configure l’affichage de l’analyseur de façon à mémoriser l’enveloppe du spectre. Pour cela, il suffit d’appuyer sur le bouton « **Trace** » de l’analyseur, puis sur le bouton « **Trace Config** » de l’écran tactile, et de sélectionner le mode « **Max Hold** » de la « **Trace 1** ».

1. Configurer l’analyseur tel qu’indiqué précédemment, et modifier la largeur de bande fréquentielle d’analyse de façon à mesurer précisément la fréquence centrale **fc** ainsi que la largeur de bande **BW** du canal de communication radio (on utilisera les marqueurs de l’analyseur 🡪 bouton **Mkr**). Préciser alors les valeurs de ces deux paramètres **fc** et **BW**.
2. **Transmission des données au serveur de réseau**



L’objectif consiste, conformément au synoptique de la page précédente, à vérifier la capacité du dispositif à accéder, depuis le serveur de réseau de la baie, aux valeurs de masses mesurées.

Mettre le serveur de réseau de la baie sous tension, attendre une vingtaine de secondes que la LED verte en face avant clignote, puis mettre sous tension la passerelle positionnée sur la baie.

Lancer un navigateur depuis l’ordinateur de commande, et accéder au serveur de réseau d’adresse 192.168.1.20:8080 (ou bien double cliquer sur le raccourci bureau ).

Vérifier que la passerelle communique avec la balance, en cliquant successivement sur les boutons « Gateways », « gateway\_dorian » puis « LIVE LORAWAN FRAMES ».

1. Cliquer sur quelques rubriques défilantes, et retrouver les valeurs des paramètres de l’interface radio mesurées à la question précédente.

Nous allons désormais accéder aux données transmises par la balance au serveur de réseau.

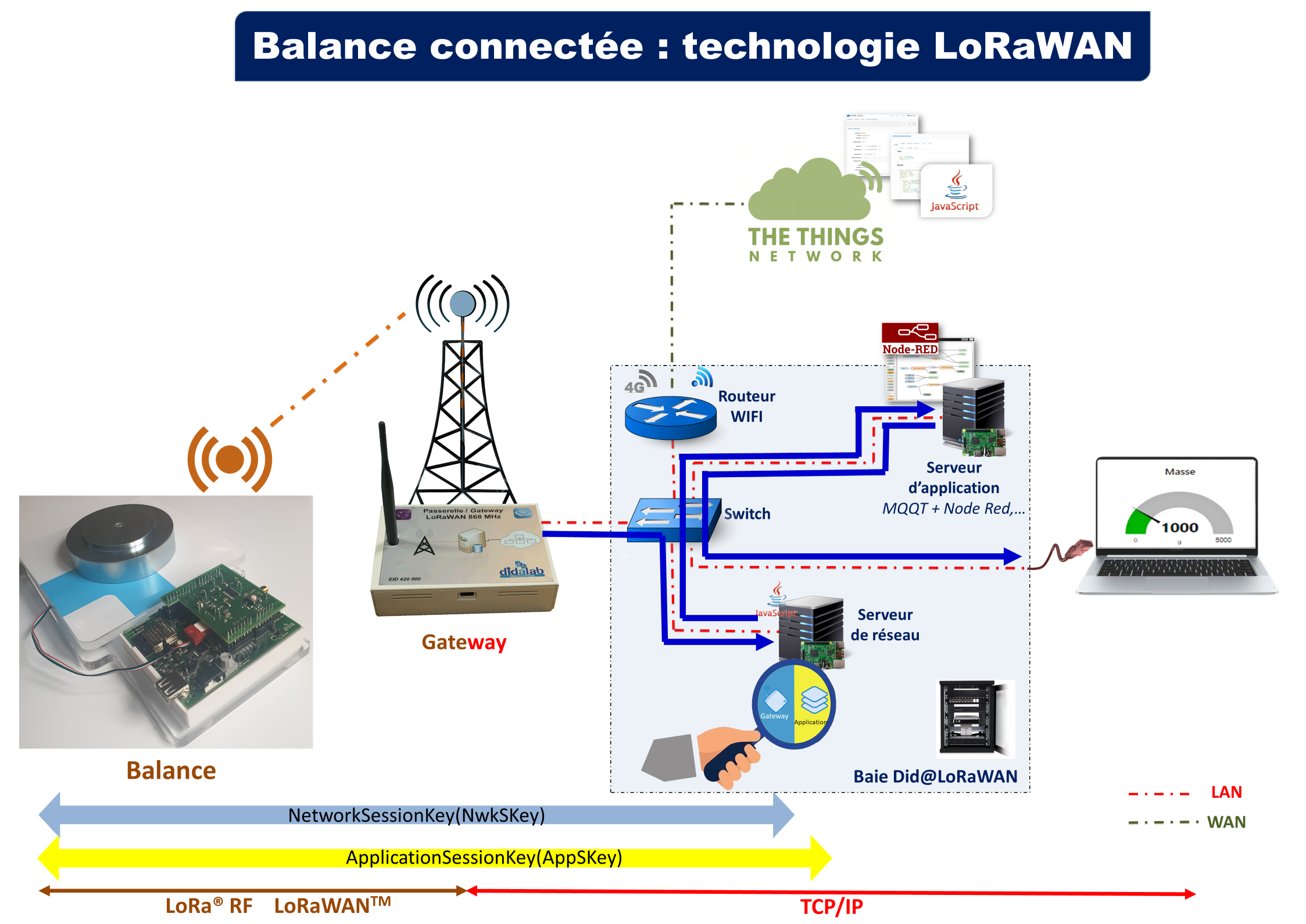
On précise au préalable que l’application transmet non seulement la valeur de la masse mesurée en g (contenu de la variable **data\_balance** précisée à la question 13), mais également la chaine de caractères « Balance » (voir codage de la fonction **main.c**).

Cliquer successivement sur les boutons « Applications », « Appli\_Balance » et « EID430A » (correspondant à la carte End Device associée à la balance).

Cliquer ensuite sur le bouton « ACTIVATION », remettre à zéro les deux compteurs « Uplink » et « Downlink », cliquer sur le bouton « (RE)ACTIVATE DEVICE », sur le bouton « EID430\_A » et enfin sur le bouton « DEVICE DATA ».

1. Cliquer sur quelques rubriques défilantes, et identifier celles qui sont associées à la transmission de la masse mesurée et celles liées à la transmission de la chaine de caractères.
2. Vérifier, en positionnant sur le plateau de la balance, des masses variant de 0 à 5 kg par pas de 1 kg, que la masse mémorisée sur le serveur de réseau est bien exprimée en g, avec une précision au gramme près, comme souhaité. Si tel n’est pas le cas, en donner la raison.
3. **Transmission des données au serveur d’application**

L’application consiste simplement à proposer un affichage de la masse mesurée plus convivial que celui obtenu en réponse à la question précédente :



Pour cela, nous allons mettre en œuvre le logiciel de gestion de flux graphique **Node-Red**.

Mettre le serveur d’application de la baie sous tension.

Lancer l’exécution du script **node\_red-local** (raccourci sur le bureau : Une image contenant texte

Description générée automatiquement) et laisser la fenêtre ouverte pour la suite du TP (ou la réduire dans la barre des tâches de Windows).

Ouvrir une nouvelle fenêtre de navigation, et accéder au serveur d’application d’adresse **localhost:1880** (ou bien double cliquer sur le raccourci bureau ).

Cliquer sur les boutons successifs « Flow 1 », « Dashboard » (situé dans le coin supérieur droit de la page) et enfin sur le bouton situé juste en-dessous (donc à droite du bouton « Thème »).

1. Vérifier, en positionnant sur le plateau de la balance, des masses variant de 0 à 5 kg par pas de 1 kg, que la masse affichée à l’écran est bien exprimée en g, avec une précision au gramme près. Si tel n’est pas le cas, procéder au réétalonnage de la balance (attention, après chaque étalonnage, il est nécessaire de I/ réactiver la carte balance 🡪 bouton « **ACTIVATION** », II/ remettre les compteurs « **Uplink**» et « **Downlink** » à 0 et III/ réactiver la communication 🡪 bouton « **(RE)ACTIVATE DEVICE** »).