



HAL
open science

Interopérabilité des réseaux de capteurs hétérogènes dans un appartement intelligent

Moutie Chehaider, Kévin Roussel, Ye-Qiong Song

► **To cite this version:**

Moutie Chehaider, Kévin Roussel, Ye-Qiong Song. Interopérabilité des réseaux de capteurs hétérogènes dans un appartement intelligent. UbiMob - 9èmes journées francophones Mobilité et Ubiquité - 2013, Jun 2013, Nancy, France. hal-00877451

HAL Id: hal-00877451

<https://inria.hal.science/hal-00877451>

Submitted on 28 Oct 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Interopérabilité des réseaux de capteurs hétérogènes dans un appartement intelligent

Moutie Chehaider
INRIA Nancy Grand-Est
moutie.chehaider@inria.fr

Kevin Roussel
INRIA Nancy Grand-Est
Université de Lorraine
kevin.roussel@inria.fr

Ye-Qiong Song
LORIA
Université de Lorraine
song@loria.fr

RESUME

L'utilisation efficace de réseaux de capteurs hétérogènes nécessite un effort considérable de développement (passerelles matérielles et logicielles), freinant ainsi les développeurs d'applications. Cet article présente une nouvelle architecture de passerelle permettant d'assurer l'interopérabilité et de faciliter l'accès transparent aux données de différents types de capteurs, utilisée dans le cadre de l'habitat intelligent pour le maintien à domicile des personnes âgées/dépendantes. Plus particulièrement, nous nous focalisons sur l'accès aux réseaux de capteurs sans fil via des services web pour des capteurs non pourvus de la pile CoAP/6LoWPAN. Cette passerelle offre aussi une interface homogène pour accéder aux données traitées par ces capteurs, grâce à l'intégration d'une couche *middleware* basée sur ROS (Robot Operating System).

ABSTRACT

Efficient use of heterogeneous sensors networks needs a huge development effort (hardware and software gateways), thus slowing down application developers. This article presents a new gateway architecture ensuring interoperability and facilitating transparent access to the data offered by various types of sensors, applied to the domain of smart housing for supporting the home care of the elderly/impaired people. More especially, we focus on the access to wireless sensor networks using web services in the case of devices devoid of the CoAP/6LoWPAN network stack. This gateway also provides an homogeneous interface for accessing the data processed by these different sensors, thanks to the integration of a middleware layer based on ROS (Robot Operating System).

Keywords

Web des objets, Réseaux de capteurs, CoAP, REST, Middleware, ROS.

1. INTRODUCTION

Les réseaux de capteurs sans fil (WSN : Wireless Sensor Networks) ont durant la dernière décennie pris un rôle sans cesse croissant, et désormais prépondérant, dans de nombreuses applications destinées à mettre en place des environnements dits « intelligents ».

L'une de ces applications concerne la mise en place de lieux de vie (appartements, maisons...) dotés de réseaux de capteurs permettant de suivre et d'analyser les faits et gestes de leurs occupants ; lorsque lesdits occupants souffrent d'une perte d'autonomie (personnes âgées, handicapées), cette surveillance permet d'améliorer leurs conditions de vie : en leur offrant une assistance réactive adaptée à leurs besoins, en facilitant le travail des différents intervenants à domicile (professions de santé, auxiliaires de vie) et surtout en permettant de donner l'alerte en cas de survenue d'une situation critique : malaise, chute... Un tel « domicile intelligent » peut permettre d'éviter ou du moins retarder le moment où la personne devra être retirée de son logis pour être placée en maison de retraite ou à l'hôpital. Le maintien à domicile est toujours préférable, humainement et économiquement, quand les conditions (notamment médicales et environnementales) le permettent.

Pour être efficaces, notamment dans des applications ainsi liées à la santé, de tels réseaux doivent le plus souvent faire intervenir un ensemble très hétérogène de capteurs, actionneurs, et autres appareils électroniques embarqués : ces systèmes hétérogènes reposent sur l'interopérabilité de ces différents éléments. La dite interopérabilité nécessite de suivre des standards précis et adaptés, mais aussi un certain travail de recherche et des développements importants pour mettre en synergie ces différents standards de façon optimale.

L'un des objectifs du travail présenté dans cet article est de mettre à disposition sur Internet, de façon simple, efficace et sûre, les données fournies par ces réseaux de capteurs hétérogènes, facilitant ainsi le développement des applications tierces pour le maintien à domicile (suivi et analyse des activités quotidiennes, de l'évolution de santé, assistance, ...). Nous avons ainsi été amenés à faire interopérer plusieurs standards : Ethernet, WiFi, Bluetooth, EIB/KNX et les standards des WSN (IEEE802.15.4, Zigbee, 6LoWPAN) au travers d'une passerelle munie d'une couche *middleware* appelé MPIGate (<http://mpigate.loria.fr/>) [1]. Cette passerelle est expérimentée dans le cadre d'une plateforme d'appartement intelligent au LORIA à Nancy (<http://infositu.loria.fr/>), ainsi que dans le cadre du projet PAL d'Inria (<https://pal.inria.fr/>). Grâce à cette passerelle, l'accès à des données hétérogènes est grandement simplifié, que ce soit pour des programmes via des API, ou pour le Web via Internet.

Dans cet article, nous allons présenter deux aspects récemment développés autour de cette passerelle : accès via le Web aux WSN non pourvu de la pile CoAP/6LoWPAN mais respectant l'architecture REST/HTTP (Representational State Transfer), et intégration de la couche *middleware* basée sur ROS.

En effet, bien que l'architecture RESTful soit rendue possible sur certains nœuds de WSN grâce à la pile CoAP/UDP/6LoWPAN [5], bon nombre d'autres capteurs existants s'appuient sur d'autres

protocoles tels que Zigbee ou Xbee. Notre premier objectif est donc de rendre possible l'accès via le Web à ces derniers, exactement de la même façon que pour accéder à un nœud muni d'un serveur CoAP. Cette transparence d'accès passe par le développement d'un composant dans notre passerelle.

L'intégration d'un *middleware* vise un objectif plus global. Il s'agit d'offrir une modèle unifié qui cache l'hétérogénéité et assure la distribution des données par un modèle « publisher-subscriber ». L'accès aux données est ainsi rendu facile, que ce soit pour les utilisateurs via un navigateur web, ou pour les programmeurs via des API adéquates.

Le premier travail consiste donc à créer une passerelle entre des réseaux de capteurs non munis de la pile CoAP/6LoWPAN et Internet. Cette passerelle doit notamment être pour cela capable d'assigner une adresse globale unique (IPv4/IPv6, ou URI) à chacun des nœuds des WSN, et de router correctement requêtes et réponses entre les deux mondes. De plus, comme les nœuds composant les WSN sont hétérogènes, et que 802.15.4 est un protocole de bas niveau (n'allant au-delà de la couche liaison, contrairement à la pile TCP/UDP/IP), la passerelle doit également gérer les différences d'implémentation qui en découlent au niveau des couches réseau, transport et application ; et ce de façon transparente pour qui l'interroge depuis Internet, par le biais d'une interface unique d'interrogation.

L'Internet Engineering Task Force (IETF) travaille actuellement à l'adaptation d'IP — dans sa version 6 — au monde de l'embarqué et à ses limitations : ces travaux ont notamment permis l'apparition de la norme 6LoWPAN (IPv6 for Low-power Wireless PAN) [2], [3]. Si cette norme ouvre la voie à l'utilisation d'IPv6 par dessus un protocole de plus bas niveau tel que 802.15.4, elle ne règle pas tous les problèmes : la nouvelle norme a notamment dû sacrifier la compatibilité avec IPv6 pour rester suffisamment légère ; les paquets 6LoWPAN ne sont pas directement routables sur Internet, une passerelle reste donc encore nécessaire pour interconnecter les deux mondes [5].

Plusieurs architectures de passerelles ont déjà été proposées, dans nombre de travaux antérieurs, afin de parvenir à une interopérabilité entre réseaux 6LoWPAN/802.15.4 et Internet. Les solutions existantes incluent notamment un proxy IPv6 pour les WSN [4], ou encore une passerelle Web proposant une architecture REST [5]. Malheureusement, aucune d'entre elles ne permet de travailler avec des nœuds incapables d'offrir une pile 6LoWPAN par dessus 802.15.4 : les nombreux capteurs reposant sur la norme ZigBee (une autre pile réseau « de haut niveau » s'appuyant sur 802.15.4, plus légère que 6LoWPAN mais totalement incompatible) nécessitent des passerelles spécifiques pour être atteints depuis Internet. L'importance de ces WSN basés sur ZigBee, notamment en terme d'applications industrielles déjà basées sur cette technologie, nous a conduit à développer une solution permettant de relier ces derniers à l'Internet des Objets, de la même façon que pour accéder les capteurs 6LoWPAN via CoAP/HTTP.

La solution que nous décrivons dans ce papier est conceptuellement très simple : nous relient des WSNs basés sur différents protocoles (802.15.4 seul ou avec 6LoWPAN, pile ZigBee complète) à Internet (ou tout autre réseau IP classique), grâce à un *middleware* — spécialement développé dans ce but — basé sur ROS. Ce *middleware* fournit une couche d'abstraction protocolaire, permettant de communiquer avec les différents nœuds des WSN de façon totalement transparente, en utilisant des services Web basés sur une architecture REST.

La souplesse de notre solution lui permet de s'adapter de façon autonome et efficace à toutes les combinaisons possibles de

WSNs, et d'offrir un mode d'accès simple et performant aux données mises à disposition par ces derniers. Nous espérons ainsi faciliter grandement le développement des applications d'environnement intelligent. Nous présentons notamment ci-dessous un cas d'utilisation de notre passerelle dans « l'appartement intelligent » présent au LORIA à Nancy, dans le cadre des projets visant à permettre le suivi des personnes âgées à domicile.

La section 2 du présent papier offrira un panorama rapide rappelant l'état de l'art actuel concernant l'interconnexion entre WSNs et Internet des objets. La partie 3 présentera ensuite l'architecture de notre solution, ainsi que les fonctionnalités offertes. La partie 4 se concentrera elle sur la façon dont sont implémentées les communications entre d'une part notre passerelle, et d'autre part les nœuds des WSNs et Internet. La partie 5 détaillera le cas de l'utilisation de notre solution au sein de « l'appartement intelligent » du LORIA. Enfin, l'article se terminera par une conclusion et une discussion sur les futurs développements de nos travaux.

2. ETAT DE L'ART

L'intégration des WSNs à l'Internet ne date pas d'aujourd'hui. Il existe de nombreux travaux de définition et de développement de passerelles selon deux approches. L'approche la plus récente considère que les nœuds sont munis de la pile HTTP/TCP/UDP/6LoWPAN et développe des passerelles légères pour accéder à ces capteurs [4], [5], [7], [8]. La seconde approche s'adresse à l'interconnexion à l'Internet des WSNs dont les nœuds ne sont pas pourvus de 6LoWPAN, par le développement de passerelles faisant la conversion entre protocoles (e.g. Zigbee gateway et bridge) [9], [10].

Gopinath et al. [4] présentent une architecture de communication point-à-point entre des machines sous IPv6 et des capteurs 6LoWPAN. Des tests de performances montrent une latence entre 100ms et 135ms, jugée satisfaisante pour des applications courantes. Cette solution est néanmoins limitée à un nombre fixe de clients pouvant interroger les capteurs. Les requêtes des clients, arrivant sous la forme de paquets IPv6, sont traduites par la passerelle en des trames 802.15.4. Un autre inconvénient de cette proposition est qu'elle répond uniquement à des besoins spécifiques, et n'est pas compatible avec d'autres plateformes ou systèmes n'utilisant pas l'IPv6.

Pour y remédier, une architecture intéressante a été proposée par Colitti et al. [5]. Il s'agit d'une passerelle permettant d'intégrer des capteurs 6LoWPAN avec l'Internet des Objets, en se basant sur des applications Web avec architecture REST (Representational State Transfer) et le protocole HTTP (Hypertext Transfer Protocol). La passerelle joue le rôle d'un serveur proxy qui permet de relayer les requêtes HTTP des clients directement vers les nœuds. Sur chaque nœud, un serveur HTTP allégé est installé pour répondre de façon autonome aux requêtes des clients. L'interaction entre les capteurs et les applications web est assurée par une machine qui agit en tant que serveur web et proxy : le serveur web de la passerelle permet d'interroger les serveurs HTTP implémentés sur les nœuds, et le proxy établit la communication avec le réseau de capteurs via un routeur de bordure RPL. L'adaptation et l'intégration de cette architecture à notre solution nous a permis d'assurer l'interopérabilité de nos capteurs Wasmote avec notre passerelle.

Le *middleware* DIGIHOME, décrit par Romero et al. en [11], offre une plate-forme ambitieuse, basée sur la technologie REST, permettant l'intégration de données issues de capteurs très divers (notamment ceux faisant partie de WSN) collectées à l'aide des protocoles variés (TCP/IP, ZigBee, UPnP, etc.) Toutefois, ce

projet s'intéresse principalement à la domotique « pure », et ne prévoit pas d'interconnecter les réseaux hétérogènes ainsi gérés à Internet ; la notion d'IoT est absente de ce projet, dont le but est en fait de permettre la gestion d'un habitat intelligent via une interface adaptée aux terminaux mobiles (notamment les Smartphones). Les choix techniques retenus par le projet sont par conséquent différents des nôtres, DIGIHOME ne se limitant pas aux appareils limités que sont les capteurs WSN, et ayant recours à des technologies (Java, XML, SOAP, ESB, etc.) trop exigeantes en termes de puissance matérielle ; il nous était par conséquent impossible de réutiliser les travaux fournis par ce projet.

En lien direct avec l'application de maintien à domicile que nous visons, Jara et al. [6] présentent une solution basée sur des capteurs sans fil mobiles 6LoWPAN connectés à une base de données médicale déployée sur une passerelle, pour le suivi de santé de personnes dépendantes. Les informations des capteurs sont envoyées via le protocole GPRS. Malheureusement, cette solution assure juste la collecte de données : elle ne présente pas de détails quant au fonctionnement de la passerelle.

Notre étude d'état de l'art montre que toutes les solutions proposées fournissent un accès soit à des réseaux sans fil basés sur le protocole 6LoWPAN, soit à des réseaux de capteurs non pourvus de la pile 6LoWPAN, mais aucune ne supporte les deux mondes à la fois. Dans ce papier, nous présentons une passerelle vers des réseaux de capteurs hétérogènes, implémenté sur une couche *middleware* basée sur le paradigme « Producteur / Consommateur ». L'objectif de cette couche *middleware* est d'assurer l'accès aux données des capteurs depuis Internet via des services Web, ou encore des services de distribution de données, pour permettre à d'autres équipements de communiquer avec nos capteurs via le réseau IP. La figure 1 montre la communication des capteurs avec un réseau IP via la passerelle.

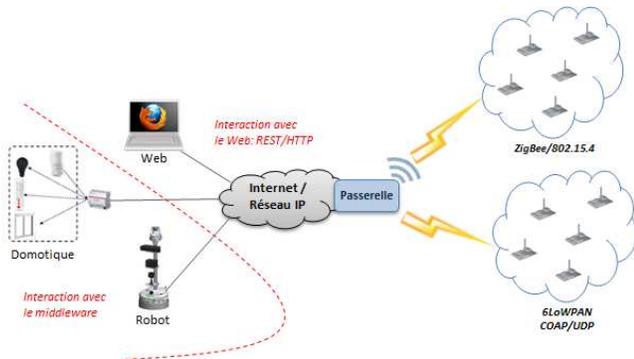


Figure 1 : Interaction du réseau de capteur avec le réseau IP

3. ARCHITECTURE DE LA PASSERELLE

La passerelle est conçue pour supporter de bout en bout des communications entre des applications Web (depuis Internet) et des réseaux de capteurs sans fils hétérogènes déployés dans un habitat intelligent. Deux types de communication peuvent être traités par la passerelle :

- Communication par interrogation : Un client Web interroge directement les ressources d'un capteur sans fil 802.15.4 déployé dans l'appartement.
- Communication spontanée : Les capteurs sans fil envoient, soit périodiquement, soit en cas d'événement, leurs données à la passerelle. Le client web peut alors récupérer la dernière valeur envoyée par le capteur.

Quelle que soit la méthode de communication employée, la passerelle permet d'assurer les fonctionnalités suivantes :

- Identification des capteurs : reconnaissance de chaque capteur de façon unique via son/ses ID, ainsi que du type de capteur et de sa localisation (pièce dans l'appartement) ; fourniture de la liste des capteurs actifs et enregistrement de leur adresse MAC, pour faciliter les interrogations ultérieures.
- Capture des données : récupération des informations présentes dans l'environnement, avec leur contexte.
- Interconnexion des réseaux : connexion des réseaux de capteurs hétérogènes basés sur le protocole 802.15.4 avec le réseau IP public ; utilisation possible d'Ethernet comme réseau de transport pour transférer les données aux clients situés sur des sites distants.
- Intégration et déploiement : grâce au service de découverte, la passerelle peut s'adapter de manière autonome à chaque nouveau capteur arrivant dans les réseaux de capteurs sans fil installés dans l'appartement ; génération dynamique (sans besoin de reconfiguration par l'administrateur) de liens Web pour accéder aux données des nouveaux capteurs via Internet.
- Abstraction vis-à-vis du format des données : lors de la récupération de trames (Zigbee / Xbee / 802.15.4 / ...), la passerelle traite celles-ci afin de fournir une couche d'abstraction, où tous les capteurs sont présentés sous forme d'objets virtuels dans un espace de données partagé ; cette couche facilite l'accès aux données de façon transparente.

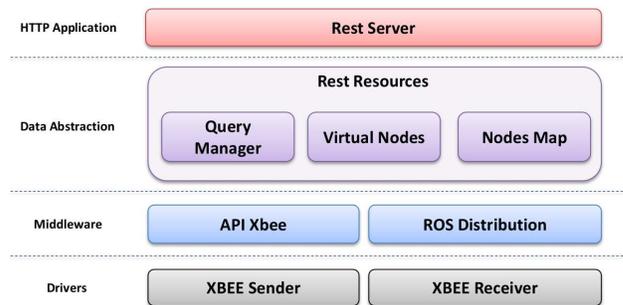


Figure 2: Architecture de la passerelle

L'architecture de la passerelle globale étant déjà présentée dans [1], nous nous focaliserons ici uniquement sur la partie Xbee / IEEE 802.15.4. La passerelle est découpée en quatre couches (Figure 2), détaillées du bas en haut comme suit :

- **Couche Drivers:** cette couche connecte deux modems sans fil de type Xbee. Ces modems assurent une communication bidirectionnelle avec les capteurs. Le premier est un coordinateur du réseau 802.15.4 servant à écouter le réseau en permanence pour récupérer les données et les remonter à la couche *middleware*. Le deuxième modem sert à envoyer des requêtes pour interroger les capteurs. Ce modem est relié à une API Xbee transformant une demande de ressource en une trame d'interrogation. L'utilisation du double modem est nécessaire à cause du mode « half-duplex » imposé : nous devons éviter la perte de données / d'événements pendant l'interrogation d'autres capteurs.

- Couche Middleware** : cette couche est une interface de liaison entre l'entrée du système (trames Zigbee/Xbee récupérées) et la couche d'abstraction de données dans laquelle les capteurs sont présentés sous forme d'objets virtuels. L'intégration d'une couche *middleware* permet en effet de gérer la distribution de données entre les composants logiciels de la passerelle. L'implémentation de la couche *middleware* se divise en deux étapes : premièrement, un module ROS jouant le rôle d'ordonnanceur est relié au modem de réception. Ce module permet d'analyser et traiter les trames Zigbee / Xbee récupérées, pour créer des « publishers » publiant des instances des messages reçus sous forme de « topics » ROS. Chaque « topic » correspond à l'ensemble des messages envoyés par un capteur pour une type de valeur précis, formant ainsi un flux de données homogènes. La liaison entre « topics » et capteurs se fait via les ID des capteurs. La deuxième étape consiste à créer des « subscribers » s'inscrivant aux « topics » de façon adéquate, et à attacher chaque « topic » à un capteur virtuel pour mettre à jour les variables de cet objet à chaque réception d'une trame. Le choix inhabituel de ROS est justifié principalement pour deux raisons : 1) la facilité d'intégration avec le monde robotique, car la passerelle assurera ainsi aussi bien les interactions avec les robots d'assistance qu'avec les capteurs environnementaux ; 2) le besoin d'un modèle de distribution « Publisher / Subscriber » pour les données / services, afin d'accéder de manière transparente à ces derniers.

- Couche d'abstraction vis-à-vis des données** : celle-ci prend la forme d'un espace de données partagé entre le serveur web (RESTful) et la couche *middleware*. Dans l'architecture de la passerelle, cet espace partagé est un ensemble de ressources REST se présentant sous forme de nœuds virtuels. Ces nœuds sont des objets dynamiques qui forment une représentation à l'identique des capteurs au sein de la passerelle ; plus précisément : chaque objet correspond à un capteur, et est alimenté par les ressources envoyées par ce capteur. Dans cette couche, on trouve aussi une table de correspondance entre les ID et les adresse MAC des capteurs présents sur le réseau. Elle est utilisée pour lister les capteurs actifs et pour les interroger facilement et rapidement.
- Couche applicative** : au plus haut niveau se trouve le serveur REST permettant d'analyser et d'exécuter les requêtes des clients. Pour chaque requête HTTP reçue, le serveur la transfère vers le capteur ciblé s'il est directement interrogeable, sinon il renvoie la dernière valeur transmise par le capteur déjà présente dans les objets-ressources REST.

L'implémentation du *middleware* ROS dans une passerelle vers l'Internet des Objets nous permet de présenter les données issues des capteurs sous forme de services. Ces services simplifient la communication des capteurs sans fil reliés à la passerelle avec d'autres systèmes embarqués basés sur le système ROS (comme les robots Wi-Fi). D'autre part, l'intégration de la couche intergicelle ROS permet d'assurer la robustesse d'une communication temps réel entre les services Web et les capteurs : en effet, une architecture classique reposant sur une base de données introduit souvent un temps de latence lié à la mise à jour de cette base. Comme nous le verrons par la suite, la couche ROS permet aussi d'enrichir les fonctionnalités de la passerelle par l'adjonction de services assurant la supervision des réseaux de capteurs.

4. SYSTEME DE COMMUNICATION DE LA PASSERELLE

Parmi ses points forts, la passerelle est capable de s'adapter dynamiquement à l'ajout des nouveaux capteurs aux réseaux qu'elle gère. Ces capteurs doivent impérativement suivre une structure de données standard définie par notre plateforme (structure héritée du projet d'appartement intelligent). Le format défini est le suivant :

```
{CapteurID, AdresseMAC, Localisation, Date,
NomdeVariable(1), ValeurdeVariable(1),
....
NomdeVariable(n), ValeurdeVariable(n)}
```

Où *NomdeVariable(i)* et *ValeurdeVariable(i)* forment un couple de données qui identifie la nature de variable et sa valeur. Actuellement les *i* variables utilisées dans l'appartement sont :

- 1 - Température,
- 2 - Luminosité,
- 3 - Pression,
- 4 - Ouverture des portes,
- 5/6/7 - Accélération X/Y/Z,
- 8 - Présence,
- 9 - Niveau de batterie.

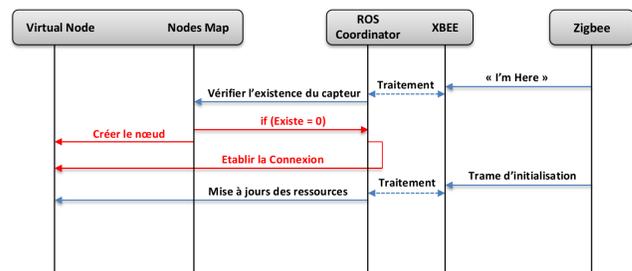


Figure 3 : Service de découverte de la passerelle

La figure 3 illustre le fonctionnement du service de découverte de la passerelle. Lors du démarrage d'un capteur, deux trames de découverte sont envoyées à la passerelle. La première trame avec le tag « I'm Here » permet à un capteur nouvellement arrivé d'informer la passerelle de sa présence. Lors de la réception de cette trame, la passerelle vérifie l'existence de l'ID fourni dans la « Nodes Map » ; s'il n'existe pas, un nouveau nœud virtuel correspondant à l'ID de ce capteur est créé, ensuite un lien Web d'interrogation de ce capteur est généré. La deuxième trame permet d'initialiser les ressources du nœud virtuel avec les premières données issues du capteur. Ces ressources sont enregistrées par la passerelle via la méthode décrite ci-dessous.

Dans notre plate-forme, la passerelle admet une adresse IP fixe, accessible depuis Internet, au sein du réseau LORIA. A partir de sa page d'accueil (<http://mpigate.loria.fr/sensors/>) la passerelle liste les capteurs présents et permet leur interrogation. Dans cette liste, chaque capteur est spécifié par son ID, sa localisation et ses ressources : luminosité, température, détection de présence, ouverture de porte, pression sur une chaise ou un lit. Pour chaque capteur, une URL (par ex. : <http://localhost/sensors/14/readings>) est générée automatiquement lors de la détection. La figure 4 illustre l'interaction entre les différentes couches constituant la passerelle pour assurer la communication de bout en bout entre un client Web et un capteur sans fil.

Quand le serveur REST reçoit une requête GET, il récupère l'adresse MAC du capteur depuis la table de correspondance (« *Nodes Map* ») puis transmet la requête sur le modem émetteur (« *Xbee Sender* ») pour interroger le capteur. Le capteur reçoit la trame Xbee et renvoie la réponse vers le modem récepteur (« *Xbee Receiver* »). La communication entre les capteurs et la station de base se fait en unicast basé sur un adressage 16 bits. Une fois la réponse du capteur reçue et traitée, la passerelle met à jour les ressources du nœud virtuel, gardant ainsi en mémoire la dernière valeur, et renvoie les données au client. En cas d'interrogation d'un capteur devenu indisponible, ou bien incapable de répondre à une requête directe, la passerelle peut ainsi répondre en fournissant la dernière valeur récupérée.

La passerelle offre deux modes de communication avec Internet : le premier est basé sur une architecture REST, et emploie le protocole HTTP pour supporter la conversation entre client Web et passerelle ; le deuxième est basé sur les fonctionnalités de communication large distance qu'offre le *middleware* ROS. En effet, grâce à ROS les données Zigbee ou Xbee sont reproduites sous forme de « *topics* », puis récupérées par le module standard ROSbridge, qui peut ensuite les retransmettre (via l'usage de « *websockets* ») vers des clients Web distants. Cette procédure permet de présenter les données issues des capteurs — de façon transparente — sous forme d'un service Web public, utilisable par des robots et plus généralement par tout équipement supportant l'API ROS.

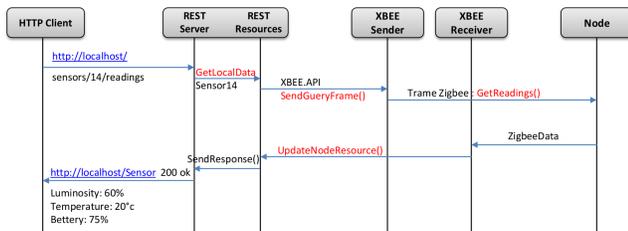


Figure 4 : Communication de bout en bout entre un client web et le réseau de capteurs

Dans la section suivante, nous présentons des scénarios et des cas d'utilisations réels de la passerelle dans l'appartement intelligent du LORIA, d'une part pour gérer une communication distante entre des capteurs sans fil et des clients Web via Internet, et d'autre part pour présenter l'interaction de la couche *middleware* avec d'autres technologies.

5. IMPLEMENTATION DE LA PASSERELLE

La passerelle a été à l'origine développée pour assurer l'intégration des capteurs sans fil déployés dans un appartement intelligent avec des applications Web. Des scénarios de test sont réalisés dans notre plateforme d'expérimentation pour valider l'architecture de la passerelle et montrer les solutions que peuvent fournir la passerelle pour l'assistance et le suivi des personnes âgées et/ou dépendantes.

5.1 Plateforme d'expérimentation

La figure 5 illustre la structure de l'appartement intelligent du LORIA/INRIA à Nancy [1].

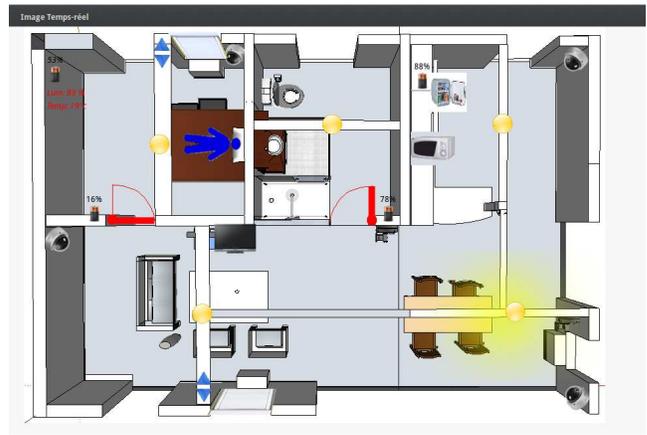


Figure 5 : Image temps réel de l'appartement de Nancy

Dans cet appartement, deux types de capteurs sans fil (issus de concepteurs différents) sont déployés :

- capteurs Waspnote, utilisant le protocole 802.15.4 : certains sont installés pour détecter la présence de personnes sur les fauteuils et sur le lit, d'autres pour détecter l'ouverture et la fermeture des portes des chambres et du frigo ; les capteurs de pression et d'ouverture de portes envoient automatiquement leurs données à la passerelle lors de la détection d'événements (afin d'économiser l'énergie, leur émetteur / récepteur radio est par défaut mis en veille, pour être activé seulement par la détection d'un événement à transmettre, ici : la détection d'un contact ou d'une pression) ; les autres capteurs Waspnote (dédiés à la détection de présence, de température et luminosité) envoient eux leurs données en réponse à une requête (et ne bénéficient donc pas du même mécanisme d'économie d'énergie) ;
- capteurs TelosB sous OS Contiki : dédiés à la mesure de la luminosité et la température des pièces ; ces capteurs sont divisés en deux catégories : ceux qui utilisent la pile 6LoWPAN et communiquent avec la passerelle via le « *RPL border router* », et ceux qui utilisent uniquement la couche MAC de Contiki (« *rime* ») en envoyant leurs données au modem « *XBee Receiver* » de la passerelle.

La passerelle a bénéficié de nos développements à différents niveaux : un module ROS a été développé en C++ et installé en aval du modem « *XBEE Receiver* » pour récupérer, traiter, et ensuite publier les trames ZigBee reçues sous forme de « *topics* » ROS dans l'espace de données partagé local. Un autre module ROS a été développé en Java et installé au sein du serveur REST, afin de relier l'espace de données partagé du *middleware* avec les ressources du serveur REST : ce module Java se charge en effet de représenter les « *topics* » ROS sous forme de nœuds virtuels. Le *middleware* ROS assure l'interopérabilité entre ces modules développés avec des langages différents.

Notre objectif est de relier le réseau de capteurs sans fil de l'appartement aux réseaux IP (et notamment Internet), afin de fournir accès aux données suivant plusieurs schémas bien définis, ces schémas étant les suivants :

- Accès local : fourniture d'une interface web de supervision pour relever l'état des capteurs.

- Accès distant offert aux professions de santé : fourniture d'une interface d'actimétrie accessible via Internet, pour lister les activités quotidiennes des personnes supervisées pendant une journée. Ces données sont publiées en temps réel sur Google Agenda.
- Accès distant aux données brutes : fourniture d'un service Web et d'un service ROS pour transférer les données des capteurs à des écosystèmes externes.

5.2 Résultats préliminaires

Cette section liste des cas d'utilisations de la passerelle et des applications développées dans l'appartement du LORIA. Nos résultats sont également accompagnés des mesures de performance de la couche *middleware* afin de valider l'efficacité de notre architecture.

L'interrogation des capteurs reliés à la passerelle se fait par le biais d'URI devant respecter un format précis : le préfixe (URL de la passerelle : <http://mpigate.loria.fr>) doit être suivi de :

- `/sensors/list` : renvoie la liste des capteurs présents sur le réseau
- `/sensors/{capteurID}/readings` : renvoie l'ensemble de toutes les valeurs lues par le capteur.
- `/sensors/{capteurID}/{variable}` : renvoie la valeur d'une seule variable lue par le capteur.

La figure 6 montre un exemple d'accès aux capteurs via la passerelle. On voit à gauche la page d'accueil listant les capteurs connectés au réseau Zigbee, avec un lien d'accès ; à droite se trouve un exemple de réponse à une requête envoyée à un capteur de température.

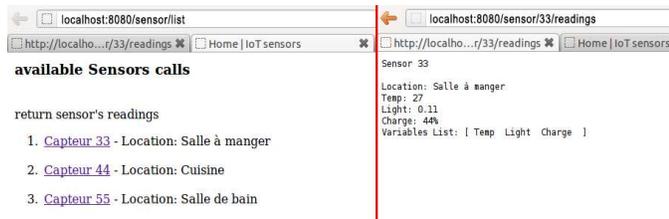


Figure 6 : Liste des capteurs et exemple d'interrogation http

Pour valider l'architecture de la passerelle et montrer l'interaction de la couche *middleware* avec d'autres services, nous présentons maintenant des exemples d'applications développées et testées dans l'appartement intelligent du LORIA.

Dans une plateforme web, toutes les données Zigbee sont interprétées et représentées sous forme graphique. En effet, grâce au module ROSBridge publiant les « *topics* » ROS sous forme de services Web, chaque détection d'événement aboutit à la fourniture automatique des données issues des capteurs sur le Web. Comme on le voit sur la figure 5, on peut visualiser de façon graphique l'état des capteurs domotiques (Zigbee et autres) depuis un simple navigateur Web. Cette interface permet ainsi de suivre les activités de la personne supervisée ainsi que l'état de son environnement en temps réel ; par exemple, la figure 5 nous montre que notre patient est dans son lit, mais a laissé la porte du frigo ouverte et la lumière de la salle à manger allumée.

Pour tester le fonctionnement de la passerelle dans un environnement réel, nous nous sommes focalisés sur l'interaction des capteurs sans fil avec un service domotique et des robots d'assistance. Dans ce but, nous avons réalisé les deux scénarios suivants :

- *Capteurs sans fil et domotique* : le matin, lorsque le patient se lève (cela est détecté par un capteur de pression sous le lit), la passerelle va d'abord allumer la lumière de la chambre puis ouvrir le volet roulant. Ensuite, à l'aide d'un capteur de luminosité installé dans la chambre, le système prend la décision d'éteindre ou non la lumière afin d'économiser l'énergie (cf. <http://mpigate.loria.fr/>).
- *Capteurs sans fil et robot* : un robot Wi-Fi, équipé d'une caméra infrarouge, communique avec la passerelle pour se faire relayer les données du capteur de pression situé sous le lit. Ainsi, le robot est programmé pour se diriger vers le lit si le capteur sans fil détecte une présence anormale (tardive) du patient dans son lit. (cf. <http://mpigate.loria.fr/demonstration/interactionRobot.avi>)

5.3 Etude de performances

Le *middleware* basé sur ROS introduit des délais supplémentaires. La figure 7 montre les traitements nécessaires entre l'arrivée d'une trame Zigbee, sa publication en tant que « *topic* » ROS (T1), sa distribution vers le « *subscriber* » via un réseau local Ethernet (T2), et sa prise en compte par le « *subscriber* » (T3).

Dans ce scénario, la rapidité moyenne de publication du « *topic* » ROS est excellente : T1=0,4 ms. Le délai de bout en bout (T1 + T2 + T3) est de l'ordre de 4,5 ms, ce qui est satisfaisant pour toutes nos applications.

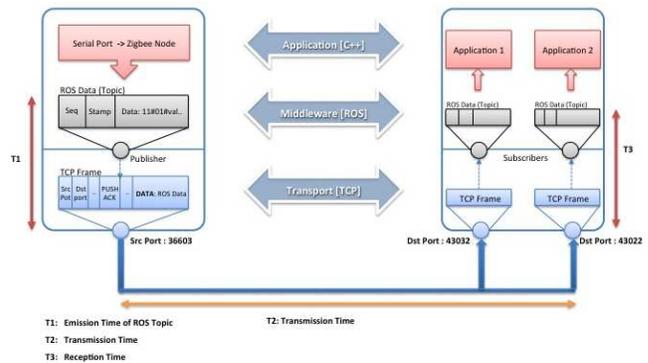


Figure 7 : Délai de bout en bout au travers le *middleware* ROS et un réseau local Ethernet

6. CONCLUSION

L'utilisation efficace des capteurs hétérogènes pour l'habitat intelligent, et plus particulièrement dans le but du maintien à domicile de personnes âgées et/ou dépendantes passe par un effort de développement de solutions d'interopérabilité. Dans ce papier, nous avons présenté nos développements récents qui ajoutent à la passerelle MPIGate (<http://mpigate.loria.fr/>) la capacité d'adresser des capteurs sans fil via les protocoles REST/HTTP, que ces capteurs soient ou non munis de la pile 6LoWPAN. L'interopérabilité est aussi renforcée par le développement d'une couche *middleware*, permettant ainsi un accès homogène transparent aux données des différents types de capteurs.

Deux résultats originaux sont présentés dans ce papier. Le premier consiste en la couche *middleware* basée sur ROS (Robot Operating System). Ce choix est surtout motivé par l'intégration des robots d'assistance dans nos réseaux de capteurs, constituant ainsi un véritable réseau de capteurs et actionneurs opérationnel malgré l'hétérogénéité des matériels et protocoles sous-jacents.

Les premières mesures ont montré des performances très satisfaisantes. Le deuxième est d'ordre technique et pragmatique : il s'agit du développement d'une composante de la passerelle permettant d'accéder via le Web non seulement aux capteurs sans fil munis de la pile CoAP/6LoWPAN, mais aussi aux capteurs qui en sont dépourvus. Ainsi ce dernier résultat étend les solutions de passerelles existantes qui ne se focalisent que sur l'un ou l'autre de ces types de capteurs.

Nos travaux futurs porteront sur l'étude des interactions entre les capteurs sans fil et les robots d'assistance, ainsi que sur la gestion par notre passerelle de la qualité de service (QoS), dans le cadre du projet LAR (« *Living Assistant Robots* »).

7. REFERENCES

- [1] Cruz-Sanchez H., Havet L., Chehaider M., Song Y.-Q., "MPiGate: A Solution to use Heterogeneous Networks for Assisted Living Applications", The 9th IEEE International Conference on Ubiquitous Intelligence and Computing (UIC 2012) (2012) 104-111
- [2] Kushalnagar, N. Montenegro, G, and Schumarer, C. "IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals" RFC 4919
- [3] Shelby, Z and Bormann, C. Wiely, 2009 "6LoWPAN: The Wireless Embedded Internet"
- [4] Gopinath, R. Zeldi, S. Reza, K Usman, S. Mazlan, A and Sureswaran, 2012. "IPv6 Wireless Sensor Network Gateway Design and End-to-End Performance Analysis". SENSORCOMM
- [5] Colitti, W. Steenhaut, K, De Caro, N. Bogdan, B and Virgil, D. 2011. "REST Enabled Wireless Sensor Networks for Seamless Integration with Web Applications". IEEE MASS 2012
- [6] Jara, A. J. Zamora M. a and A. F. G. Skarmeta, , 2009. "HWSN6: Hospital Wireless Sensor Networks Based on 6LoWPAN Technology: Mobility and Fault Tolerance Management", 2009 International Conference on Computational Science and Engineering, pp. 879-884
- [7] Da Silva Campos, B., Rodrigues, J.J.P.C., Mendes, L.D.P., Nakamura, E.F., Figueiredo, C.M.S., "Design and Construction of Wireless Sensor Network Gateway with IPv4/IPv6 Support", ICC2011, Tokyo, 5-9 June 2011.
- [8] Joel J. P. C. Rodrigues and Paulo A. C. S. Neves, "A survey on IP-based wireless sensor network solutions", Int. J. Commun. Syst. 2010; 23:963–981, Wiley InterScience.
- [9] Zigbee alliance, "Network device: gateway specification, version 1.0, March 23rd, 2011
- [10] Peng Q., Yi Z., Ung H., Di Z., Jaeho C., "Gateway architecture for zigbee sensor network for remote control over IP network", 8th Asia-Pacific Symposium on Information and Telecommunication Technologies (APSITT), 2010.
- [11] Romero D., Hermosillo G., Taherkordi A., Nzekwa R., Rouvoy R. and Eliassen F, "RESTful Integration of Heterogeneous Devices in Pervasive Environments", 10th IFIP International Conference on Distributed Applications and Interoperable Systems (DAIS'10), Amsterdam, Netherlands, pp. 1-14.