
Approche IDM pour le développement des services basés capteurs

Marin Cristina, Donsez Didier, Lalanda Philippe

Equipe Adèle, LSR-IMAG, Université Joseph Fourier, Grenoble

Résumé

De nombreux capteurs communicants sont déployés dans notre environnement. Leur rôle est de collecter des mesures sur l'environnement surveillé. Le développement des applications à base de capteurs/ actionneurs demande des connaissances technologiques multiples et complexes. Pour le faciliter, nous proposons une approche IDM (Ingénierie Dirigée par les Modèles). L'approche proposée dans cet article modélise les concepts du domaine à l'aide d'un métamodèle métier. Ce métamodèle indépendant d'une plate-forme d'exécution est ensuite transformé dans un modèle spécifique à une plate-forme d'exécution. Une validation de notre approche est en cours de développement pour une application à base de capteurs sur OSGi.

1. Introduction

Les réseaux de capteurs sont aujourd'hui de plus en plus utilisés pour surveiller en temps réel des environnements hétérogènes. Qu'il s'agisse de suivre la circulation automobile d'une ville pour pouvoir régler les feux afin de régulariser le trafic, de contrôler le fonctionnement des équipements d'une usine de distribution d'énergie électrique, de surveiller un patient en convalescence à son domicile ou de suivre le parcours d'un produit dès sa conception jusqu'à son arrivée dans le chariot de la ménagère, la technologie utilisée pour capturer des événements sur l'environnement repose sur le même principe. George Lawton [1] mentionne que par rapport à 2003 où la production des équipements comportant un capteur était de 15 millions pour l'année, l'estimation pour l'année 2010 donne le résultat de 880 millions de tels équipements, un chiffre énorme mais significatif pour montrer les besoins d'outils fiables qui aident à l'avancement dans le domaine.

Au dessus des capteurs résident des applications. Nous proposons de concevoir ces applications par assemblage de services - les services basés capteurs (SBC) [2]. Ces unités logicielles se proposent d'acquérir, collecter, filtrer, d'agréger, d'analyser et de réagir aux mesures acquises par les capteurs disséminés dans le monde réel. Ces services s'articulent principalement autour du traitement de flux de mesures acquises en début de chaîne par une myriade de capteurs.

Cet article propose une approche IDM (Ingénierie Dirigée par les Modèles) [3] pour la conception des applications à base de services basés capteurs. L'approche IDM est un concept apparu il y a quelques années qui propose de faciliter le développement des applications par l'utilisation des modèles. A partir des modèles décrivant les applications, toute ou partie de l'application peut être générée. L'approche consiste de la définition d'un modèle d'application indépendant de la plate-

forme d'exécution. Le modèle peut ensuite être rendu spécifique à une plate-forme d'exécution par des techniques de transformation de modèles.

Cet article est organisé comme suit. La section 2 présente la difficulté de la mise en place d'une architecture distribuée d'une application à base de capteurs/ actionneurs. La section 3 présente l'approche que nous proposons. La section 4 fournit une conclusion et développe les perspectives de notre travail.

2. Services basés capteurs

Le rôle d'un capteur est de récolter des informations sur l'environnement surveillé [4]. Ces mesures sont ensuite remontées vers une application - *application à base de capteurs/ actionneurs*. Elle est réalisée par l'assemblage de plusieurs unités logicielles – des services basés capteurs. Ceux ci ont pour but d'acquérir, de collecter, de filtrer, d'agréger, d'analyser et enfin de réagir aux mesures acquises par les capteurs disséminés dans le monde réel d'une organisation ou d'une entreprise. Les mesures ainsi remontées peuvent être présentées à un utilisateur (un technicien d'une usine de distribution électrique) ou enregistrées dans un espace de stockage (une base de données, un dépôt de données) .

Il est à peu près admis que les solutions centralisées dans lesquelles le serveur collecte directement les mesures relevées par les capteurs [5] ne permettent pas la mise en place rapide et à grande échelle de services performants et simples à administrer, indispensables à la satisfaction conjointe des entreprises et des fournisseurs d'équipements et des services associés [6]. Aujourd'hui les nouvelles tendances localisent la logique applicative au plus près des équipements par l'installation de passerelles Internet intelligentes et normalisées entre les équipements et les serveurs IT. Les services basés capteurs déployés au niveau d'une passerelle effectuent une première chaîne de traitements sur les mesures collectées. Seules ces données traitées, agrégées, filtrées sont envoyées vers les services basés capteurs du système d'information de l'entreprise, le coût des échanges réseau étant ainsi diminué. Cela permet aussi un fonctionnement autonome de la passerelle qui agrandit la résistance aux pannes.

Une telle architecture se caractérise par l'hétérogénéité des plates-formes d'exécution où les SBC doivent être mis en place. Le développement des services logiciels sur de telles architectures se complexifie d'une part à cause de la grande variété de plates-formes d'exécution pouvant être utilisées sur les passerelles (J2ME, OSGi, J2EE, .NET) et d'intergiciels (Data Distribution Service de l'OMG, WireAdmin d'OSGi [7], standard NIST IEEE 1451.1) mis en oeuvre et d'autre part à cause du caractère fluctuant (dynamique) de l'environnement en cours de production (branchement/retrait/reconfiguration de nouveaux équipements, défaillance transitoire/définitive).

Cela montre le besoin de disposer d'outils de développement pour faciliter la tâche du développeur des SBC qui peut être un expert métier non spécialiste des technologies nécessaires pour mettre en place une telle architecture distribuée de services. Pour cela, nous proposons une approche qui suit l'approche IDM. Cette approche consiste en la définition d'un métamodèle d'application à base de services basés capteurs indépendant d'une plate-forme d'exécution. La section suivante présente l'approche proposée et sa mise en oeuvre.

3. Meta-modèle des applications à base de capteurs/ actionneurs

L'IDM définit des approches génératives pour le développement des applications. Le point central de l'approche IDM n'est plus le logiciel mais son modèle. Le modèle est exprimé par des concepts décrivant le problème à résoudre. Ainsi, les modèles sont beaucoup plus compréhensibles, il est plus facile de les spécifier et de les maintenir. Il est même possible que ce soit l'expert métier qui spécifie le modèle et non l'expert de la technologie nécessaire pour sa mise en oeuvre [8].

Notre approche consiste en plusieurs métamodèles (Figure 1 gauche) que par manque d'espace nous

ne présentons pas dans ce papier. Comme l'application à base de capteurs/ actionneurs sera conçue dans une architecture orientée services, un métamodèle (Métamodèle SOA) abstrait les concepts caractérisant une telle architecture. Une réification de ce métamodèle est le métamodèle des architectures orientées services dans le domaine des applications à base de capteurs/actionneurs (Métamodèle Application dans le domaine). Ce métamodèle raffine les interactions qui peuvent apparaître entre les services basés capteurs. Le modèle d'application conforme au métamodèle Application dans domaine reste indépendant d'une plate-forme d'exécution (ç.a.d PIM). Le modèle donne à tout moment une vision correcte et abstraite de l'application. L'utilisation des modèles comme base de développement permet une évolution flexible de l'application, le modèle pouvant subir plusieurs incréments. La plate-forme d'exécution est prise en compte seulement au moment de la génération de code où le modèle spécifié, indépendant de la plate-forme (PIM), est transformé dans un modèle spécifique à la plate-forme (PSM). Toutefois, la logique applicative de l'application doit être ajoutée par l'expert .

La Figure 1 (droite) présente le métamodèle métier qui fait l'objet de cet article. Il décrit les concepts les plus importants du domaine et les moyens d'interactions entre ces concepts.

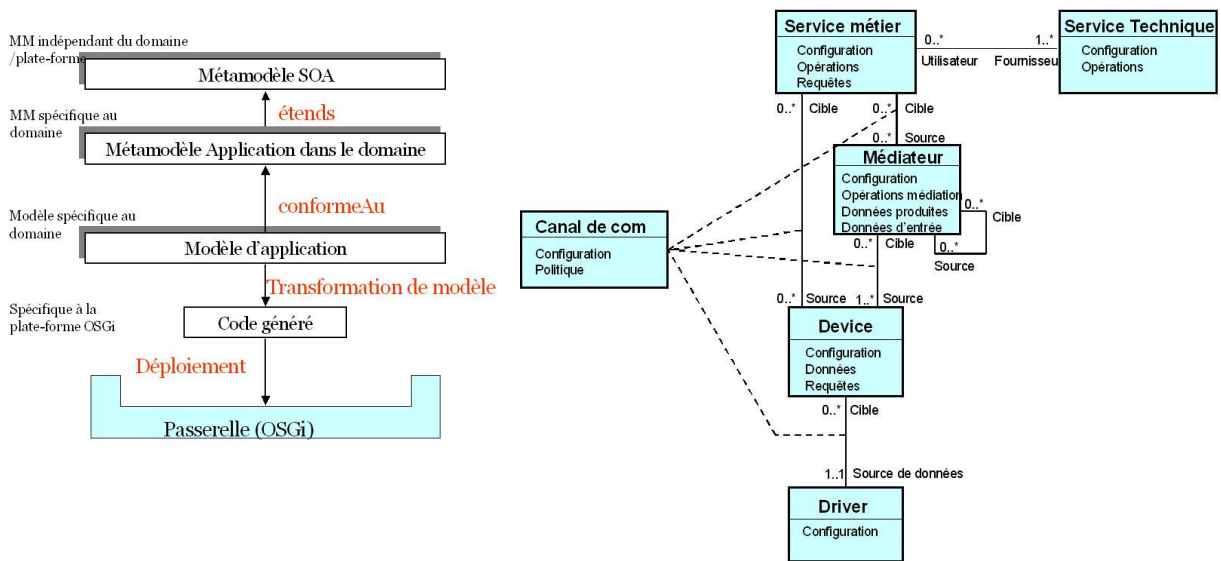


Figure 1 Approche IDM pour les applications à base de capteurs/ actionneurs

Driver - Ce concept abstrait les modalités d'interaction avec les capteurs. L'interaction se fait en employant des protocoles bien définis qui spécifient les couches de liaison physique et les mécanismes d'échange. Un driver peut également optimiser les requêtes issues de la passerelle vers les équipements et redirige les réponses vers les composants de type **Device**.

Device - Un **Device** est une abstraction d'un équipement (capteur) qui lui donne une vue simplifiée au niveau de la passerelle. Pour obtenir les mesures collectées par l'équipement, un **Device** doit formuler des requêtes vers un **Driver** et doit formater dans un langage évolué les mesures obtenues. Il définit également une stratégie de collecte des informations (type des échanges, période d'interrogation). Il contient une description des données caractérisant l'équipement et le processus contrôlé par l'équipement.

Médiateur - Un médiateur effectue des opérations complexes (transformation, agrégation, filtrage, intégration) sur des mesures issues d'un ou plusieurs équipements. Il contient des méthodes permettant d'accéder aux mesures (médiation) et de définir la stratégie de collecte et d'intégration.

Service Métier - Un service métier implante une fonctionnalité reposant généralement sur les mesures contenues dans les **Devices** et/ou les **Médiateurs**. Un exemple est un service métier utilisé par un superviseur de la passerelle pour connaître les moyennes des mesures collectées dans les

deux derniers jours par certains équipements. Il contient également des méthodes permettant d'accéder aux données et de définir leur stratégie de collecte.

Service technique - Ce concept modélise les services qui implantent une certaine fonctionnalité considérée comme non-fonctionnelle par rapport aux services métiers (des services d'historisation, de distribution, d'ordonnancement).

Les liaisons existant entre les concepts présentés ont associé un **canal de communication**. Il est chargé d'assurer l'échange de données suivant la politique caractérisant la remontée de l'information. Car l'initiative de la remontée de l'information peut être double : *push* (les informations disponibles peuvent être envoyées à l'initiative du concept qui les détient) ou *pull* (requête explicite).

Pour valider notre approche nous utilisons les outils proposés par le projet Eclipse EMF [10] et nous cibons comme plate-forme d'exécution une passerelle OSGi (OSCAR [11]). L'outillage conçu est destiné à l'utilisation d'un expert métier. Un éditeur obtenu avec l'aide d'EMF (génération de code) lui permet de définir le modèle de son application dans les termes présentés. Le modèle ainsi décrit est indépendant de la plate-forme d'exécution cible, des détails spécifiques à la plate-forme d'exécution sont ignorés par le modèle pour permettre une portabilité maximale. L'application est obtenue par une deuxième phase de génération de code dans laquelle nous prenons en compte la plate-forme d'exécution. Pour la réaliser, nous avons définis des *templates* JET spécifiques pour chaque concept du métamodèle présenté. Cette phase de génération résulte dans la création de plusieurs projets Eclipse (un pour chaque élément). La logique applicative doit être ajoutée par l'expert métier mais toutefois son travail est beaucoup diminué par le fait que les détails parfois complexes des technologies nécessaires pour le développement de son application pour une plate-forme spécifique lui sont complètement masqués.

4. Conclusions et perspectives

Cet article présente une approche IDM pour le développement d'applications exploitant des mesures récoltées par un réseau de capteurs. Nous pensons que l'approche est innovante car le développeur d'une telle application ne doit plus être un spécialiste des technologies nécessaires pour la mise en oeuvre de l'application. Nous validons actuellement cette approche dans le cadre du projet RNRT PISE avec Schneider Electric et France Telecom pour le développement des applications avec une architecture composée de plusieurs services basés capteurs pour une passerelle OSGi. Une des perspectives de nos travaux est la validation de l'approche pour une deuxième plate-forme (J2EE).

5. Bibliographie

- [1] G. Lawton. Machine-to-machine Technology gears up for Growth. *IEEE Computer*, 37(9):13–15, September 2004.
- [2] M. Desertot, C. Marin, and D. Donsez. Sensorbean : Un modèle à composants pour les services basés capteurs. Journées Composants JC'05, Avril 2005.
- [3] Object Management Group. Model Driven Architecture. <http://www.omg.org/mda/specs.htm>.
- [4] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. A Survey on Sensor Networks. *IEEE Communication Magazine*, 40:102–114, 2002.
- [5] Supervisory Control and Data Acquisition. <http://ref.web.cern.ch/ref/CERN/CNL/2000/003/scada/>.
- [6] P. Lalanda. E-services Infrastructure in Power Distribution. *IEEE Internet Computing*, May/June 2005.
- [7] OSGi Alliance. OSGi Service Platform (3d release). <http://www.osgi.org>, March 2003.
- [8] B. Selic. The Pragmatics of Model-Driven Development. *IEEE Software*, pages 19–25, September 2003.
- [9] E. Seidewitz. What Models Mean. *IEEE Software*, pages 26–32, September 2003.
- [10] Projet Eclipse. Eclipse Modeling Framework. <http://download.eclipse.org/tools/emf/scripts/home.php>.
- [11] R.S. Hall and H. Cervantes. An OSGi Implementation and Experience Report. Proceedings of the IEEE Consumer Communications and Networking Conference, January 2004.