社団法人 情報処理学会 研究報告
IPSJ SIG Technical Reports

2007－ＳＬＤＭ－129／2007－ＥＭＢ－4（８）
2007／3／15

細粒度アクセス制御機構を有する超小型無線ノード向けJava仮想マシン

中本 幸一† 東山 修平† 千葉 匠峰† 古隅 陽子†

† 兵庫県立大学大学院応用情報科学研究科
〒650-0044 神戸市中央区東川崎町1-3-3
E-mail: †{nakamoto,aa05d212,aa05m211,ab06w202}@ai.u-hyogo.ac.jp

あらまし センサーネットワークやアクティブRFIDのような小型無線ノードが普及しつつある．我々は，小型無線ノード向けの汎用的でセキュアなプログラム実行環境であるActiveBeansを開発している．本研究では，小型無線ノード向けのアプリケーションプログラムをJavaプログラミング言語で記述するものとし，小型ノード内にはJava仮想マシンを実装する．管理対象物に小型無線ノードが装着されているとする．ActiveBeansでは，管理対象物の情報をXML形式でデータを保存する．細粒度アクセス制御機能によりこのデータアクセスへのXMLの要素単位でのアクセスを制御する．

キーワード Java仮想マシン，細粒度アクセス制御，小型無線ノード

A Java Virtual Machine with Fine-grain Access Control for Small Wireless Deives

Yukikazu NAKAMOTO†, Shuhei HIGASHIYAMA†, Narutaka CHIBA†, and Yoko FURUZUMI†

† School of Applied Informatics, University of Hyogo
1-3-33, Higashi-kawasaki-cho, Chuou-ku, Hyogo, 650-0044 Japan
E-mail: †{nakamoto,aa05d212,aa05m211,ab06w202}@ai.u-hyogo.ac.jp

Abstract Small wireless nodes in sensor networks are increasing being deployed in a wide variety of applications. In this report, we present ActiveBeans, a generic and secure software platform in a small wireless node employing the Java technologies for data management and sensor/actuator control applications inside a building. To deal with management data items and controlled devices in a node in a uniform manner, we introduce a node description in the XML format. ActiveBeans provides a Java software platform to access the node description securely. ActiveBeans has the following features: a small Java virtual machine with small footprint management for various input and output device using the generic framework, the data management, and the fine-grained access control.

Key words Java Virtual Machine, Fine-grain Access Control, Small Wireless Device

1. まえがき

超小型無線ノードが様々な分野で使用が始まっている．たとえば，センサーノードは環境データの情報取得に，アクティブRFIDタグは物流システムなどにに利用されている．このような超小型無線ノードにも，携帯電話やICカードのような他の組込みシステムで利用されている技術が普及すると考えられる．超小型無線ノードのCPU能力やメモリ容量も増加するであろう．その結果，超小型無線ノードで使用されるソフトウェア機能も複雑になり，規模も増大すると考えられる．超小型無線ノード向けのソフトウェアプラットフォームが必要とされてくる．

このような超小型無線ノードの初期の段階では，ハードウェア能力が低いために専用OSとプログラム実行環境が利用されてきた(例,[1],[?])．最近では，さらに汎用なOSやプログラム実行環境が利用されてきている[2],[?]．日本国内ではディペンダブルな機能を強化したLinuxの実装が進められている[3]．

Squawkのような無線センサーネットワーク用のJava実行環境が研究されている[4]．Squawkは無線センサーネットワークノード用に以下のような特徴をもっている．第一に，Squawkはプログラム分離による安全なマルチプログラム実行の機構を提供している．第二に，Javaプログラミング言語によって

Squawk VM は実装されていること，Java バイトコードを実行時に内部表現に変換することにより，小型メモリで効率よく Java プログラムを実行されることが可能となっている．第三に，無線通信が汎用コネクションフレームワーク (genetic connection framework, GCF) を使って利用することが可能である．このような特徴はセンサーノードのプログラムを開発するのに適していると考えられる．本稿で提案する AcitiveBeans のターゲットとしているのは，屋内で利用されるデータ管理やセンサー/アクチュエータ制御アプリケーションである．小型無線ノードはあらゆるところで利用される．これにはデータそのものを維持管理，データを利用した操作，ノードに付加されたセンサー/アクチュエータ制御である．このようなアプリケーションはセンサーネットワークのように大量のノードを利用するのではなく，短距離無線通信を利用した中規模サイズのノードネットワークで構築される．ここでは，よりセキュアな信頼性の高いデータ管理が必要とされる (e.g., [5]).

屋内のデータ管理とデバイス制御は典型的な実社会でのアプリケーションである．屋内データ管理では，たとえば物流システムでの管理があげられる．ここでは 2 つのデータ管理が考えられる: 物流システムで管理対象となる “もの”に対するデータの管理と “もの”間の関係に関するデータの管理である．倉庫での “もの”のデータ管理を考えてみる．倉庫では多くのケージが置かれている．ケージには商品が入ったダンボール箱が置かれている．本稿では商品のことを “アイテム”，ダンボール箱を “コンテナ”と呼ぶことにする．小型無線ノードはアイテム管理のために，アイテム自身，あるいはコンテナに装着される．各コンテナの装着されたノードは，コンテナ自身のデータと，コンテナとその中に入っているコンテナあるいはアイテムの間のデータを管理する，と見ることができる．この 2 種類のデータは動的に，しかも簡単に変更できる必要がある．

小型無線ノードの第二の例は価格表示や店内のカメラ制御のような店舗管理である．価格表示の例では，小型無線ノードには商品の現在の価格を表示するディスプレイがついている．これにより，店舗内の表品管理や価格変更が簡単になる．

ActiveBeans の目的は，実社会でのアプリケーションでの利用を考えた，小型無線ノード向けに汎用でセキュアなデータアクセス制御機構を有するプログラミング手段を提供することである．ActiveBeans のプログラム実行環境は，簡単なプログラム実行環境とセキュアで柔軟なデータ管理機能を有する．今回の例では，無線通信は，IEEE 802.15.4 を利用することにする．

本稿の構成は以下のとおりである．2. 章では AcitveBeans での Java 仮想マシンの機能を述べる．3. 章ではデータ管理機構とアクセス制御機構について述べ，4. 章で試作の状況を報告する．

2. データ管理用プログラム実行環境

1. 章で述べたように，ActiveBeans の目的は，小型無線ノード向けに汎用でセキュアなデータアクセス制御機構を有するプログラミング手段を提供することである．ActiveBeans の仮想マシンである ActiveBeans VM の開発にあたり，以下のような設計方針をおいた．なお，ActiveBeans VM は携帯電話向けの Java 仮想マシンである Connected Limited Device Configuration [6] と IC カード向けの Java 仮想マシンである JavaCard [7] の中間に位置づけられる．

- Java 実行環境:

我々は小型無線ノード向けアプリケーションプログラムのプログラミング言語として Java プログラミング言語を選択した．Java プログラミング言語とそのプログラム実行環境は小型無線ノードのハードウェアの詳細を隠蔽し，プログラミングを容易にすると考えられるからである．さらに，3. 章で述べる，データ管理とアクセス制御用のクラスライブラリを容易に構築できる．

- マルチスレッド，ガーベッジコレクション機能の削除:

ActiveBeans VM は小型無線ノード内のデータにアクセスするための単純なプログラム実行環境を提供する．したがって，マルチスレッド機能は現時点では不要と考えている．このほか，小型無線ノードではオブジェクトの生成，削除することは多くないと考えられることから，ガーベッジコレクション機能は削除することとした．ActiveBeans VM は JavaCard の VM と類似したものになっている [7].

- 多様な入出力デバイスのための汎用フレームワーク:

小型無線ノードには多様な入出力デバイスが装着することが考えられる．ActiveBeans VM では，このような多様な入出力デバイスへのアクセスと操作を単一の形態で行うため，‘ノードオブジェクトファイル’で抽象化することにした．この目的は，デバイスの多様性を隠蔽し，プログラム実行環境を単純化するためである．記述方法としては，汎用コネクションフレームワーク (Generic Connection Framework, GCF) を採用した [6]. GCF は URL により指定されたコネクションを確立する手段を提供する．GCF の URL の一般的な形式は以下である．

`{scheme}:[{target}] [{params}]`.

`{scheme}` は実装依存であり，実際のデバイスを指定する．`{scheme}` の例として，バイトストリームストレージやレコード型ストレージなどが考えられる．`{target}` はデバイスのファイル名であり，`{params}` はオプションおパラメタである．データストレージへのアクセスは以下のようになる:

```
Storage io
  = Connector.openIODevice
    ("storage:///dev/storage");
```

`Connector` は GCF のインタフェースであり，このメソッドは `IOStorage` クラスの，`{scheme}` で指定されたデバイスに対応するインタフェースを実装する，実装オブジェクトを返す．GCF を利用することにより，多様なデバイスアクセス手段を提供することができると同時に，無線小型ノードへのデバイスアクセスを少量のメモリ量で実装することができる．

3. データ管理とアクセス制御機能

小型無線ノードを利用しての，物流システムなどでのデータ管理やセンサー/アクチュエータ制御での要求事項をまとめると次のようになる．第一に，1. 章で述べたように，小型無線

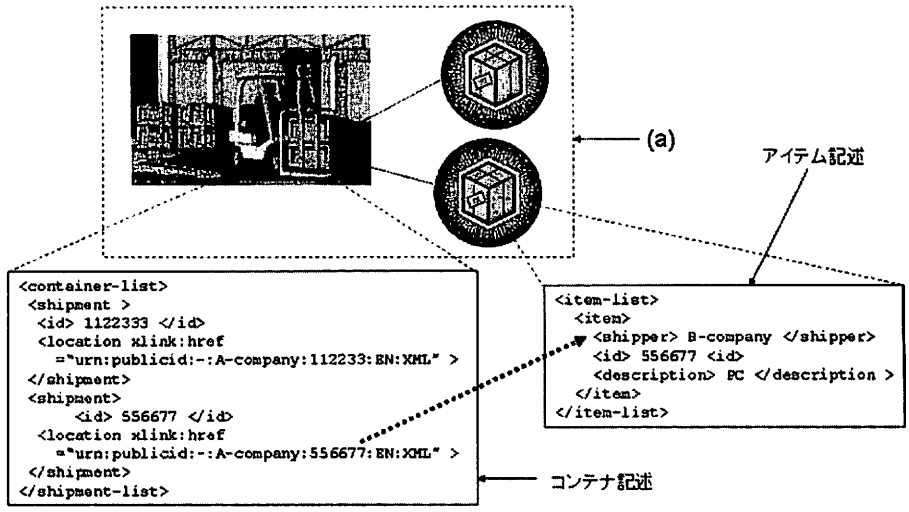

図 1　物流システムでのデータ管理

ノード内のデータ管理はノードに付随するアイテムのデータだけでなく，他のアイテムに装着したノードとの関係記述が必要となる．EPCglobal は類似のデータ管理フレームワークを提供している [8] が，中央で集中管理をする形態であり，アイテムの動的な変動に対する柔軟性に欠けると考えられる．

第二に，アイテムデータやノードに装着したデバイスへのアクセスは制御される必要があるという点である．アイテムデータへのアクセスでは，例えば段ボール箱中のアイテムを輸送する会社は，そのアイテムに関係する情報は他社には知られたくないであろう．各会社のアイテムの情報は他社にはアクセス可能であってはならない．しかし，税関のような機関はその段ボール箱中のアイテムを全て把握する必要がある(注1)

デバイスアクセスでは，屋内の監視カメラの例では，権限をもった管理者はカメラの制御は可能とすべきだが，権限をもたない人間はカメラを制御してはいけない．

この問題を解決するために，我々はデータ間の関係記述とアクセス制御のためのクラスライブラリを開発した．

XML 形式のノード記述: アイテムのデータとノードに装着されたデバイスを統一的に扱うために，XML 形式でのノード記述を導入した．ActiveBeans VM では XML によるノード記述へのアクセスメソッドのみ提供する．ノード記述には，アイテムの管理データやノードに装着されたデバイス管理が記述される．XML 形式によるコンテナとアイテムの記述は，コンテナ記述，アイテム記述と各々呼ぶことにする (図 1 参照)．コンテナ・アイテム間を階層関係を記述するために，各コンテナとアイテムに URN を付随させる．コンテナにアイテムが入っている場合に，コンテナのコンテナ記述にはアイテムへのリンクが XLink 形式 [9] により書かれる．このときにアイテムの指定先を URN で記述する．

小型無線ノードに URN とネットワークにおけるアドレスの対応を管理する名前サーバを実装する容量がある場合，小型無線ノードのネットワークへのジョインを検出し，名前サーバ内の対応の更新が容易になる．

細粒度アクセス制御: コンテナ記述やアイテム記述内の要素へのアクセスを制御するために，XML 形式の要素毎へのアクセスを制御する細粒度アクセス制御 [10] を採用した．細粒度アクセス制御では，コンテナ記述とアイテム記述の要素や属性へのアクセス権限が要素毎に割り当てられる．記述中の要素や属性は細粒度アクセスポリシーの権限をもっているユーザにより読まれたり，更新されたりする．権限は要素をルートとする部分木内の要素，属性に伝播する．ある要素の権限は，その要素の先祖の権限と衝突する場合がある．この衝突を解決するために，権限の優先度の導入を検討している．

4. 実　　装

現在，我々は Renesas H8 3069F ボードを使って試作を行っている．このボードには 2 MB の RAM と 512 KB のフラッシュメモリが搭載されており，Freescale IEEE 802.15.4 ボードをつないでいる．ActiveBeans VM の実装には waba 仮想マシン(注2)と TOPPERS プロジェクト(注3)で提供されている ITRON 仕様リアルタイムを使用している．

ActiveBeans VM の開発にあたり，waba を以下の方針の元に小型化を行った．

- waba の機種依存部を TOPPERS の ITRON 仕様 OS の

(注1)：これは米国の Homeland Security 省による Automated Custom Environment と International Trade Data System での要求である．

(注2)：http://waba.sourceforge.net/

(注3)：http://www.toppers.jp/en/index.html

カテゴリ	クラスライブラリ
今回の試作で実装したクラス	
lang	Object, String
sys	Time, Vm
ui	Control, Control Event, Event, Ikey, Key Event, MainWindow, Timer, Window
fx	Graphics, ISurface
今回の試作で削除したクラス	
lang	StringBuffer
io	Catalog, File, SerialPort, Socket, Stream
sys	Convert
ui	Button, Check, Container, Edit, Label, PenEvent, Radio, Tab, TabBar, Welcome
util	Vector
fx	Color, Font, FontMetrics, Image, Rect, Sound, SoundClip

図 2　ActiveBeans VM のクラスライブラリ

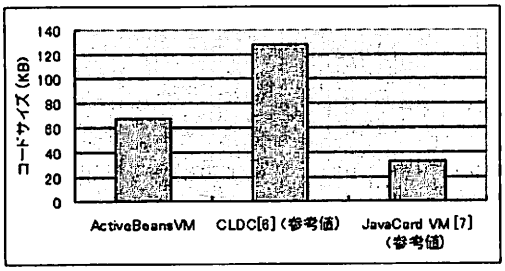

図 3　ActiveBeans VM のコードサイズ

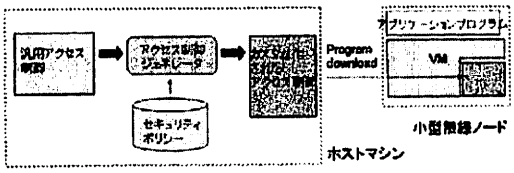

図 4　アクセス制御のカスタマイズ

API で置き換える.

- 高度なグラフィックス機能やファイル入出力などのクラスライブラリを削除した. ActiveBeans VM でのクラスライブラリと waba のクラスライブラリの差分を図 2 に示す.

上記の小型化の結果, H8 ボード上でのコードサイズを図 3 に示す. なお, CLDC と JavaCard VM のコードサイズは仕様書から各々の VM が目標とするコードサイズを抜粋したものである.

5. ディスカッション

アクセス制御を小型無線ノードに具備させることに関しては解決すべき問題点がいくつかある. 第一に, アクセス制御のアクセスポリシーをどのようにして分散し展開された小型無線ノードに誤りなく配布するかという点である. システムの運用においていくつかの中継サーバがその配下の小型無線ノードを管理するなど階層型に管理を行い, アクセスポリシーを確実に配布するなどの方法が考えられる.

第二に, アクセス制御の実装には多量のメモリが必要となることが分かっている [11]. このようなメモリ消費を避けるために, アクセス制御のモジュールをカスタマイズするジェネレータを開発することを考えている. カスタマイズされたアクセス制御機構は, ActiveBeans VM の一部として無線通信を利用してダウンロードされる.

謝　辞

IEEE 802.15.4 と ZigBee に関して情報提供頂いた KCS 株式会社社長の松村博士に御礼申し上げます.

本研究は, 兵庫県立大学特別教育研究助成金の支援を受けている.

文　献

[1] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler and K. Pister: "System architecture directions for networked sensors", Proc. the 9th international conference on Architectural support for programming languages and operating systems, pp. 93–104 (2000).

[2] J. Eswaran, A. Rowe and R. Rajkumar: "Nano-RK: An Energy-Aware Resource-Centric RTOS for Sensor Networks", Proc. 26th IEEE International Real-Time Systems Symposium, pp. 256–265 (2005).

[3] 徳田, 高汐, 中澤, 岩井, 由良：“マイクロユビキタスノード用ディペンダブル OS の実現に向けて”, 情報処理学会研究報告 (2006-UBI-12), **2006**, 116, pp. 53–60 (2006).

[4] D. Simon, C. Cifuentes, D. Cleal, J. Daniels and D. White: "Java on the Bare Metal of Wireless Sensor Devices: the Squawk Java Virtual Machine", Proc. the 2nd international conference on Virtual execution environments, pp. 78–88 (2006).

[5] P. Dutta, J. Hui, J. Jeong, S. Kim, C. Sharp, J. Taneja, G. Tolle, K. Whitehouse and D. Culler: "Trio: Enabling Sustainable and Scalable Outdoor Wireless Sensor Network Deployments", Proc. The 5th International Conference on Information Processing in Sensor Networks, pp. 407–415 (2006).

[6] Sun Microsystems: "Connected Limited Device Configuration, Specification version 1.1,Java 2 Micro Edition" (2003).

[7] Sun Microsystems: "The Java Card specifications, version 2.2.2" (2006).

[8] EPCglobal: "The EPCglobal Architecture Framework" (2005).

[9] W3C: "XML Linking Language (XLink) Version 1.0" (2001).

[10] E. Daminiani, S. D. C. di Vimercati and P. S. S. Paraboschi: "A fine-grained access control system for XML documents", ACM Transaction on Information and System Security, **5**, 2, pp. 169–202 (2002).

[11] 神田, 才田, 中山, 千嶋, 中本：“携帯端末向け Linux のリソース管理の実現”, 情報処理学会論文誌コンピューティングシステム, **46**, SIG 3(ACS 8), pp. 1–11 (2005).