

M-91 ウェアラブルコンピューティングのためのルール処理システムの設計と実装  
Design and Implementation of a Rule Processing System for Wearable Computing

宮前雅一\*1 Masakazu Miyamae  
中村聡史\*1 Satoshi Nakamura  
寺田 努\*2 Tsutomu Terada  
塚本昌彦\*3 Masahiko Tsukamoto  
橋本隆之\*4 Takayuki Hashimoto  
青木功介\*4 Kousuke Aoki  
堀 雅和\*4 Masakazu Hori  
西尾章治郎\*3 Shojiro Nishio

1. はじめに

近年、マイクロエレクトロニクス技術の発展による計算機の小型化・軽量化に伴って、ウェアラブルコンピューティングに対する注目が高まっている。我々の研究グループでは、ウェアラブルコンピューティングの実現に向け、サービス提供の基盤となるシステムの構築に関する研究を行っている [1]。

ウェアラブルコンピュータには、他の作業を行いながらマニュアルなど各種の情報を閲覧する、現在位置に関する情報を取得しながらナビゲーションを行うなど、ユーザの状況に応じたさまざまなサービスの提供が求められる。また、ユーザは常に自分の端末を携帯して行動するため、目的によって機器やシステムを取り替えるのではなく、場所や目的に応じて端末の機能が柔軟に変化することが望ましい。

そこで本研究では、ウェアラブルコンピューティング環境において、さまざまなサービスを提供するための基盤システムの構築を目的とする。提案するシステムは、各種のイベントを処理するルールによって、ウェアラブルコンピューティング環境におけるさまざまなサービスを提供する。

システムの動作をルールの集合として表現することで、ルールの追加・削除によるカスタマイズや場所に応じたルールの変更が可能になるため、柔軟なサービスが提供できる。

2. ECA ルール

提案システムの動作記述言語としては、アクティブデータベースの動作記述言語である ECA ルールを用いる。アクティブデータベースとは、従来のデータベースと異なり、データベース内部で起こる事象の発生に反応して自動的に更新等の操作を行なうデータベースシステムであり、その動作は発生する事象 (イベント)、実行させるための条件 (コンディション)、イベントによって発火する操作 (アクション) の 3 つの組からなる ECA ルールによって記述される [2]。

一般にアクティブデータベースでは、イベントやアクションとしてデータの挿入や削除といったデータベース操作しか記述できないため、本システムでは、ウェアラブルコンピューティング環境におけるさまざまなサービスを記述できるように、言語仕様の拡張を行う。

2.1 ECA ルール言語仕様

本システムでは従来のアクティブデータベースがサポートする ECA ルール記述に拡張を加え、任意のイベント、アクションを定義できるようにした。本システムで提供する主なイベント、アクションを表 1, 2 に示す。ルールの記述フォーマットは、XML (eXtensible Markup Language) 仕様に準拠させ、システム内での取り扱いを容易にした。

\*1: 大阪大学大学院工学研究科, Graduate School of Engineering, Osaka University

\*2: 大阪大学サイバーメディアセンター, Cybermedia Center, Osaka University

\*3: 大阪大学大学院情報科学研究科, Graduate School of Information Science and Technology, Osaka University

\*4: インテック・ウェブ・アンド・ゲノム・インフォマティクス株式会社, INTEC Web and Genome Informatics Corporation

表 1: イベント一覧

名称	内容
SELECT	データベースに対するデータ参照
INSERT	データベースに対するデータの追加
DELETE	データベースからのデータの削除
UPDATE	データベースのデータの更新
GPS_MOVE	GPS による移動検出
RECEIVE	データパケットの受信
END_FILE_RECEIVE	ファイルの受信終了
BEGIN_FILE_SEND	ファイルの送信開始
TIMER	設定したタイマの発火
MM_END	マルチメディアコンテンツの再生終了

表 2: アクション一覧

名称	内容
QUERY	データベース操作
SET_TIMER	タイマの設定
KILL_TIMER	タイマの削除
MULTICAST_SEND	指定したホストへパケットを送信
BROADCAST_SEND	パケットをブロードキャスト送信
FILE_SEND	指定したホストへファイルを送信
MM_PLAY	マルチメディアコンテンツの再生
BROWSER_SHOW	Web ページの表示

本研究で提案する ECA ルールの記述構文を図 1 に示す。以下、それぞれの要素について簡単に説明する。

- <ECARULE>: 一つの ECA ルールを表す。ID 属性にはルールを識別する ID を、SCOPE 属性にはルールの有効範囲を指定する。
- <VARIABLE>: ルールで使用する変数を宣言する。NAME 属性には変数を識別する名前を、TYPE 属性には変数の型を指定する。
- <EVENT>: イベントを記述する。TYPE 属性にはイベントの種類を、TARGET 属性には検出するイベントの対象を指定する。
- <CONDITION>: コンディションを記述する。実際のコンディション要素は <CITEM> タグで記述し、CITEM 間の関係は <AND> タグ、<OR> タグで記述する。<AND> タグはすべての子要素が成り立つ場合、<OR> タグはいずれかの子要素が成り立っている時に、コンディションが成立する。<AND> タグ、<OR> タグは入れ子状に使用できる。
- <CITEM>: コンディション内容を記述する。左辺式および右辺式をタグのテキストとして記述し、比較演算子を TYPE 属性値として指定する。
- <ACTION>: アクションを記述する。実際のアクションは <AITEM> タグで記述する。
- <AITEM>: アクション内容を記述する。<ACTION> タグの子要素としていくつでも記述できる。TYPE 属性にはアクションの種類、RET 属性にはアクションの返り値の格納先を指定する。

2.2 ECA ルール記述例

ECA ルールの記述例として、ユーザの見ている方向にある建物のホームページを表示するルールを図 2 に示す。このルールは、ユーザの移動に伴って付近の建物を検索されたときに発火する。GPS と地磁気センサからユーザの現在位置と見ている方向を検出し、その方向に建物があるときに検出された建物のホームページを表示する。

```

<ECARULES>
<ECARULE ID="[ルール ID]" SCOPE="[ルール適用範囲]">
  <VARIABLE NAME="[変数名]" TYPE="[変数の型]" />
  <EVENT TYPE="[イベントタイプ]"
    TARGET="[イベント対象]" />
  <CONDITION>
    (<AND> や <OR> などコンディション間制約)
    <CITEM TYPE="[演算子]">
      [コンディション記述]
    </CITEM>
    <CITEM>...
  </CONDITION>
  <ACTION>
    <AITEM TYPE="[アクションタイプ]"
      RET="[返り値格納変数]">
      [アクション記述]
    </AITEM>
    <AITEM>...
  </ACTION>
</ECARULE>
</ECARULES>
    
```

図 1: ECA ルールの構成

```

<ECARULE ID="Search Building" SCOPE="ENTIRE">
  <EVENT TYPE="SELECT" TARGET="OSAKADB" />
  <CONDITION>
    <CITEM TYPE="?">
      MAP.EXIST(%GPS.X%;%GPS.Y%;%NEW.X%;%NEW.Y%;
        %MOTION_SENSOR.ALPHA_NORTH%;100.0;20.0)
    </CITEM>
  </CONDITION>
  <ACTION>
    <AITEM TYPE="BROWSER_SHOW">
      %NEW.URL%
    </AITEM>
  </ACTION>
</ECARULE>
    
```

図 2: ECA ルールの記述例

### 3. システム構成

3章で述べた仕様に基づいてルールの処理を行うルールエンジンのシステム構成を図3に示す。以下、それぞれの機能について説明する。

- ルール管理部: XML形式のルールを解析し、内部形式に変換してルールを保持する。
- プラグイン管理部: システムに追加して機能の拡張を行うプラグインの制御を行う。プラグインは、独自のイベント・アクションをシステムに追加することでシステムの機能を拡張するモジュールである [3]。
- ルール処理部: ルールの評価・実行を行う。イベントが発生すると、ルール処理部は、ルール管理部が保持するルールを参照して発生したイベント ID に適合するルールを検索し、検索されたルールのコンディションの判定を行う。コンディションが満たされていれば、そのルールのアクションを実行する。

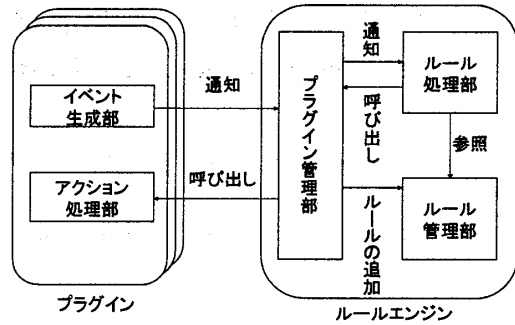


図 3: システム構成

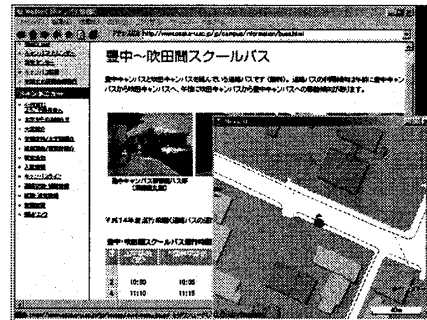


図 4: システムの表示例

### 4. 実装

以上に述べたルールエンジンを実装した。実装は Microsoft 社の Visual C++ .NET Enterprise Architect を使用して、SONY の PCG-C1MRX 上で行ない、ルールエンジンのソースコードは約 5000 行となった。

本システムを利用して構築した建物案内サービスの表示例を図4に示す。このサービスでは、ユーザの移動に伴って、ユーザが見ている方向にある建物のホームページを次々に表示する。このサービスは、3個のルールからなり、ユーザの現在位置をマップに表示するルール、ユーザの移動に伴って付近の建物を検索するルール、検索された建物のホームページを表示するルールによりサービスが実現されている。

### 5. 最後に

本稿では、ウェアラブルコンピューティング環境のための拡張可能なルール処理システムの構築について述べた。提案システムは ECA ルールによりその動作を記述するため、ルールの追加・削除による処理のカスタマイズが可能となる。本システムを用いることでウェアラブルコンピューティング環境における柔軟なアプリケーションが提供できる。

今後は、Windows CE や Palm など、軽量 PDA への対応や、ルールエンジンの機能強化を行う予定である。

### 参考文献

- [1] 塚本ほか: ウェアラブルコンピューティングのためのシステム基盤, FIT2002 論文集 (2002 掲載予定)。
- [2] Lohman, G, et al.: Extensions to starburst: object, types, functions, and rules, *Communications of the ACM*, Vol. 34, No. 10, pp. 94-109 (1991)。
- [3] 中村ほか: ウェアラブルコンピューティングのためのルール処理システムを用いたサービス, FIT2002 論文集 (2002 掲載予定)。