

シームレスなモバイルアプリケーションのための 異種位置センサと異種地図の統合化方式

渡辺 亮嗣 村上 幸彦 細川 宜秀 高橋 直久

名古屋工業大学工学部電気情報工学科

あらまし 本稿ではモバイルアプリケーション(MA)設計者の持つ位置センサ, および地図の知識・技能を補い, その操作を仮想化して利用することを可能にする基盤ソフトウェアの実現方式を提案する. SEAMA の特徴は, 位置センサや地図の切り替え時や位置情報取得時, 利用している位置センサや地図の切り替え時を MA における動作起動タイミングを MA のイベントとして定義し, そのイベントとイベントに対するアクションの組み合わせとして記述することを可能にした点にある. これより, MA プログラムは利用する位置センサや地図に依存しない形で記述可能になる. その結果, MA の設計, 開発に必要な位置センサや地図に関する知識・技能を削減し, MA の設計, 開発に要するオーバーヘッドを削減することが可能となる. さらに実験より, SEAMA の有用性を検証する.

Integration of Heterogeneous Location-based Sensors and Maps for Seamless Mobile Applications

Ryoji Watanabe, Yukihiro Murakami, Yoshihide Hosokawa, Naohisa Takahashi

Department of Electrical & Computer Engineering
Nagoya Institute of Technology,

Abstract: In this paper, we present an integration of heterogeneous location-based sensor and maps for seamless mobile application.

There are three features in our system. The first feature is to define a mobile application program (MA-program) as a tuple of 6 essential actions. The second feature is to realize a mechanism for translating a given tuple of 6 actions into an equivalent and executable MA-program which manipulates several sensors and maps. The final feature is to realize a mechanism for extracting a sequence of current locations from several sensors which have different applicable scopes each other. By these features, our system makes it possible to develop various mobile applications without knowledge and skill for manipulating sensors and maps.

We clarify feasibility and effectiveness of our system by several experiments.

1. はじめに

現在, モバイル・コンピューティング技術の発展に伴い, カーナビゲーション・システムや SpaceTag[6] などといった多種多様なモバイル・アプリケーション(MA)が設計, 構築されている. 今後, 屋内 GPS[1]などのモバイル・コンピューティング技術の更なる発展に伴い, MA の社会での適用範囲は拡大していくものと予想される. 特に, 複数の位置センサおよび地図群を統合利用する MA がその適用範囲の拡大に貢

献する.

しかしそのように多種多様な MA の実現は, これまでの MA の設計開発よりも困難を伴う. それは次の問題点による.

問題点 複数の位置センサおよび地図を複合利用する MA の設計開発において, 利用する位置センサや地図を操作するための知識・技能を, 設計者が有している必要がある. しかし, 次の3つの異種性により, その設計者が修得しなければならない知識・技能は, これまでの MA よりも多く, MA の設計開発を困難

にする。

異種性-1 センサ間の操作系・データ表現形式・適用可能範囲の異種性

異種性-2 地図間の操作系・データ表現形式・利用可能範囲の異種性

異種性-3 位置センサと地図間のデータ表現形式・利用可能範囲の異種性

さらに、より多くの異種位置センサや異種地図群を統合利用する MA の設計開発において、その知識・技能の修得に要するオーバーヘッドは大きくなり、そのような MA の設計開発をより困難にする。

本稿では MA の設計者が先に述べた 3 つの異種性を知ることなく、複数の位置センサと地図を統合利用させる MA を記述させ、MA の設計開発を容易にすることを可能にする基盤ソフトウェア(これを SEAMA = Seamless Mobile Applications)の実現方式を提案する。

その実現方式の特徴は次の 3 点にある。

特徴-1 設計者の記述した MA プログラムを実行可能プログラムに変換するトランスレータの実現

SEAMA は、複数の位置センサと地図を統合利用する MA を、利用する位置センサおよび地図に依存しない形のプログラムとして記述可能にするものである。トランスレータは、その MA プログラムを、利用する位置センサと地図の操作のための制御コードを含む実行可能な MA プログラムに変換するものである。

これにより MA 設計者は、位置センサおよび地図に関する知識・技能を有さずに MA の設計、開発を行うことが可能になる。

特徴-2 センサと地図の操作系の仮想化

SEAMA では、位置センサと地図を操作するための機能群を、SEAMA に組み込まれた位置センサと地図に依存せずに、仮想化して実現する。具体的には SEAMA をメディアータ・ラッパ・アーキテクチャによって構成することにより、この仮想化を実現する。ここで SEAMA における位置センサと地図を操作するための機能群を**標準操作機能群**と定義する。

これにより、標準操作機能群を用いることで SEAMA に接続された全ての位置センサと地図の操作が可能となるので、SEAMA のトランスレータを位置センサと地図に依存せずに実現することが可能になる。すなわち、新規の位置セン

サと地図を SEAMA に接続した場合においてもトランスレータの再構成を行う必要がなくなる。**特徴-3 センサと地図の自動切り替え機構の実現**

本機構は、複数の位置センサと地図を統合利用するが、それらの位置センサと地図に依存せずに記述された MA において、位置情報取得と地図操作を継続して行うことを可能にするものである。

これにより、新しい利用可能範囲を持つ位置センサや地図が SEAMA に接続された時に、MA を再構成することなく、MA の適用範囲を拡大することが可能となる。

さらに本機構の実現によって、利用する位置センサと地図の切り替えのための制御コードを MA 設計者が記述することがなくなる。すなわち MA の設計、開発に要するオーバーヘッドを削減させることが可能となる。

これらによって、SEAMA は先に述べた問題点を解決する。

2. 提案システム SEAMA の実現方式

SEAMA の特徴は、複数の位置センサや地図を統合利用する MA を、利用する位置センサおよび地図に依存しないプログラムとして記述可能にする点である。

位置センサや地図の切り替え、位置情報取得、利用者への提示地図の切り替えなどのイベントは、MA における特定の動作の起動タイミングとして重要である。そのため、利用する位置センサと地図に依存せずに記述される MA に対して、位置センサと地図に関するイベントを感知して対応する動作を起動するための機構を実現することが必要となる。

SEAMA では、位置センサと地図に関するイベントとして、次の 4 イベントを定義する。MA を 4 イベントに応じて起動するアクションの組み合わせとして記述する。

イベント 1 要求達成イベント : Goal

これは、MA の目的、要求が達成された時に発生するイベントを表す。

イベント 2 位置センサからの座標取得イベント : getLocation

これは、使用している位置センサから座標が取得された時に発生するイベントを表す。**イベント 3 位置センサ自動切り替えイベン**

ト : changeSensor

これは、現在使用している位置センサが適用範囲外になる、もしくは利用可能になる場合において、SEAMA により利用する位置センサが自動的に切り替えられた時に発生するイベントを表す。

イベント 4 地図自動切り替えイベント : changeMap

これは、現在使用している地図が適用範囲外になる場合において、SEAMA により利用する地図が自動的に切り替えられた時に発生するイベントを表す。

SEAMA が実行対象とする MA は次のアクションによって記述する。

アクション 1 : I_action

これは変数の初期値代入や、クラス宣言などの MA 実行のための初期化アクションである。

アクション 2 : T_action

MA の目的、要求達成イベント Goal が発生した場合において、要求達成条件が真である時に起動されるアクションである。

アクション 3 : F_action

MA の目的、要求達成イベント Goal が発生した場合において、要求達成条件が偽である時に、起動されるアクションである。

アクション 4 : L_action

MA において、getLocation イベントが発生した時に起動するアクションである。

アクション 5 : S_action

MA において changeSensor イベントが発生した時に起動するアクションである。

アクション 6 : M_action

MA において、changeMap イベントが発生した時に起動するアクションである。

SEAMA では MA を 4 イベントに対応する 6 アクションの組み合わせとして構成する。

```
MA = {I_action, T_action, F_action,
      L_action, S_action, M_action}
```

ここで、I_action, F_action, L_action, S_action, M_action は省略可能である。

また SEAMA における MA の実行手順を次に示す。

Step-1 現在使用している位置センサから座標を取得する。

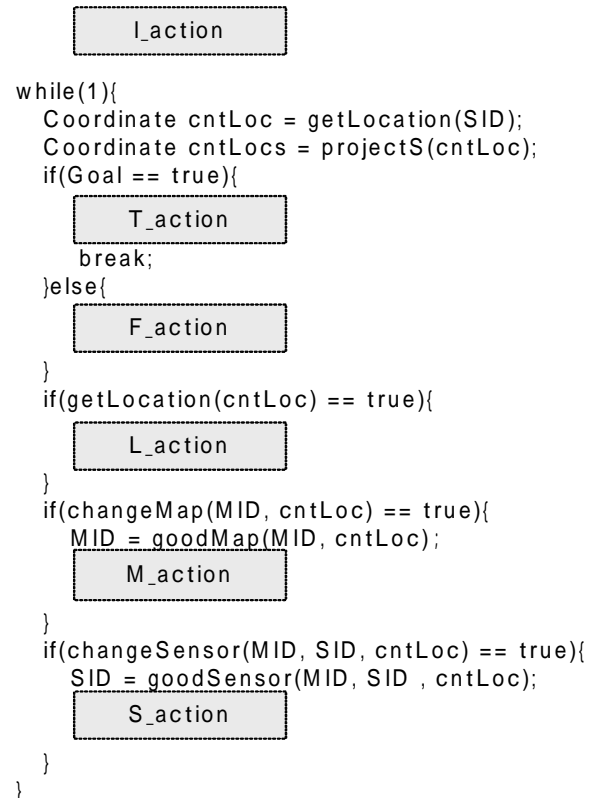


図 1 アクションを適用する実行可能プログラムのテンプレート

Step-2 MA の目的、要求達成イベントに対するアクションの要求達成条件が真であれば、T_action を実行して、MA を終了させる。

要求達成条件が偽であれば、F_action を実行して Step-3 へ進む。

Step-3 getLocation, changeSensor, changeMap のそれぞれのイベントに対するアクションを実行する。その後 Step-1 に戻り、手順を繰り返す。

図 1 は、MA から生成される実行可能プログラムを表す。

2.1 SEAMA の構成方式

MA とトランスレータを位置センサと地図に依存せずに構築するために、SEAMA をメディアエータ・ラッパ・アーキテクチャ[2]に基づいて、次の 4 サブシステムによって構成する。

(サブシステム - 1) トランスレータ

トランスレータは、位置センサと地図に依存せずに記述されたアクションを図 1 に示す実行可能プログラムのテンプレートに埋め込み実行可能プログラムを生成する

(サブシステム - 2) MA 実行部(mediator)

MA 実行部は、トランスレータによって生成された MA 実行プログラムを実行するものである。ここで MA 実行部は、位置センサと地図の標準操作機能を実現し、その機能を介して、MA が利用する位置センサと地図に対する操作の実行制御を行う。

さらに、MA 実行部では、地図と地図間、地図と位置センサ間、位置センサと位置センサ間のデータの受け渡しを標準空間を介して行う。

標準空間を介したデータの受け渡しの実現は、次の2つのメリットによるものである。

メリット-1 一般に任意のセンサを任意の地図上で利用する場合、センサと地図間のデータ変換機能をその組み合わせ数だけ実現する必要があるが、標準空間を定義することにより実現するデータ変換機構の数を、センサと地図の総和に抑えることができる。

メリット-2 新しい位置センサや地図を SEAMA に組み込む際に今までに実現されている位置センサや地図の操作系を変更する必要がない。

(サブシステム - 3) 位置センサ・ラップ

位置センサ・ラップは、位置センサ毎の固有の操作を位置センサに対する標準操作機能としてラップする。さらに位置センサと標準空間の間の値の変換を行う。

(サブシステム - 4) 地図ラップ

地図ラップは、地図と標準空間の間の座標の変換を行うための機構を実現する。

また SEAMA の標準操作機能は、次のとおりである。

なお、位置センサおよび地図の座標は、 x, y, z の3次元で表す。

標準操作機能-1 getLocation(SID) lsv{x,y,z}

現在利用している位置センサ(SID はその識別子)から現在の座標 lsv{x,y,z}を取得する。そのセンサの動作が不安定の場合は座標にはエラー値が書き込まれる。

標準操作機能-2 projectS(SID,lsv{x,y,z}) sv{x,y,z}

現在利用している位置センサから取得された座標 lsv{x,y,z}を標準空間の座標 sv{x,y,z}に変換する。

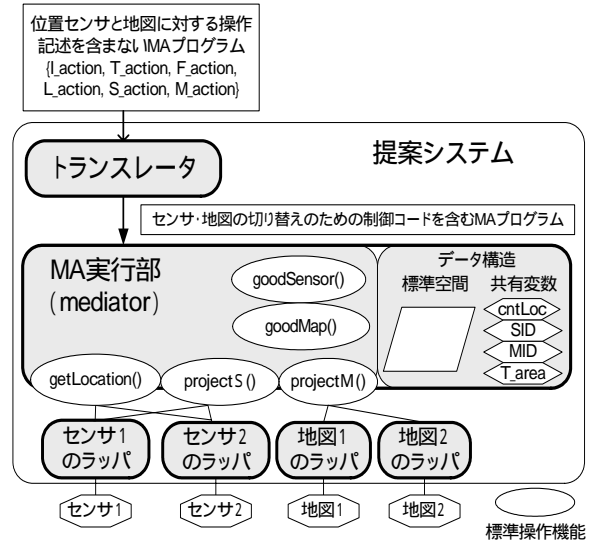


図 2 SEAMA の構成

標準操作機能-3 projectM(MID,lmv{x,y,z}) sv{x,y,z}

地図(MID はその識別子)上の座標 lmv{x,y,z}を標準空間の座標に変換する。

標準操作機能-4 goodMap(MID,sv{x,y,z}) NewMID

与えられた標準空間上の座標 sv{x,y,z}が地図(MID はその識別子)の利用可能範囲外であるとき、その座標を利用可能範囲に含む地図(NewMID はその識別子)を選択する。複数候補があるときはその中で最適な地図を選択する。

標準操作機能-5 goodSensor(MID,,SID,sv{x,y,z}) NewSID

現在利用中の地図(MID はその識別子)、現在利用中のセンサ(SID はその識別子)ならびに、現在位置の座標 sv{x,y,z}から、使用するのに最適な位置センサ(NewSID はその識別子)を選択する。複数候補があるときはその中で最適な位置センサを選択する。

goodMap および goodSensor の実現方式の詳細は、第 2.3 節において述べる。

2.2 トランスレータの実現方式

2.2.1 SEAMA と MA 間のデータ共有方式

MA の多くは、位置センサからの位置情報や地図情報を利用して実行されている。そのため、SEAMA と MA の間で位置センサからの位置情報や地図情報を共有するための機構を実現することは本質的である。

SEAMA では、この機構を位置センサの位置情報や地図情報を格納する変数を共有することによって実現する。それらの変数は次のとおりである。

- MID: 現在 MA が使用している地図の ID
- SID: 現在 MA が使用している位置センサの ID
- cntLoc[0~2]: 現在の位置センサの座標 (cntLoc [0]=x, cntLoc[1]=y, cntLoc[2]=z)
- T_areamin[0~2]: 判定領域を構成する矩形の角の座標のうち最小の座標 (T_areamin[0]=x, T_areamin[1]=y, T_areamin[2]=z)
- T_areamax[0~2]: 判定領域を構成する矩形の角の座標のうち最大の座標 (T_areamax[0]=x, T_areamax[1]=y, T_areamax[2]=z)

ここで判定領域とは、MA の要求達成条件を表し、T_areamin, T_areamax で与えられる矩形領域によって表現する。SEAMA では、位置センサから取得された現在位置が判定領域に含まれたとき、MA の要求達成条件を真とする。SEAMA における要求達成条件の判定は、cntLoc[i]および T_areamin[i], Tarea_max[i]に格納されている座標を写像した後、その写像された標準空間上の座標を用いて行う。

2.2.2 トランスレータの実行手順

SEAMA におけるトランスレータの実行手順は次の通りである。

Step-1 MA の記述内容から適用するテンプレートを選擇する。

ここでテンプレートとは、MA の記述内容より実行可能プログラムを生成するためのベースとなるコードで、MA における位置センサや地図の一連の操作を形式化したものである。図 1 はテンプレートの一例である。

Step-2 MA のアクションをテンプレートの規定した位置に挿入する。図 3 は I_action, T_action, M_action からなる MA を図 1 のテンプレートに挿入して生成される実行可能プログラムを表す。

2.3 位置センサと地図の自動切り替え機構の実現方式

SEAMA の特徴の 1 つは、複数の異種位置センサおよび異種地図を統合利用する MA を、利用している位置センサや地図に依存せずに記述、

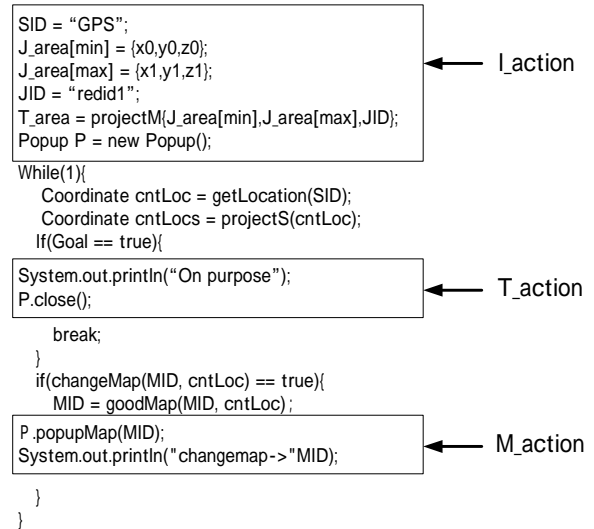


図 3 トランスレータによる実行可能プログラムの生成例

実行可能にする点にある。位置センサと地図の自動切り替え機構は、そのような MA プログラムにおいて位置情報取得と地図操作を継続して行うことを可能にするものである。goodMap と goodSensor は、位置センサと地図の自動切り替えのための基本機能である。

goodMap(MID, lmv{x,y,z})の実現方式

SEAMA は、地図の自動切り替えを行うために、地図について、次の 4 項目のデータを保持する：(1)地図の識別子、(2)利用可能範囲、(3)詳細度、(4)建物内か建物外かを表すデータ。

表 1 は地図の登録例を示す。

なお、これらのデータは、地図ラッパ作成者が SEAMA に事前に登録するものとする。

goodMap の実行手順は次のとおりである。

Step-1 位置センサから得られた座標 sv{x,y,z}が指定された地図(MID はその識別子)の利用可能範囲内にあるかどうかを判定する。

その判定結果として、利用可能範囲内に sv{x,y,z}が含まれている場合は、入力された MID を返す。すなわち、地図の切り替えは行わない。

また、判定結果として、利用可能範囲内に sv{x,y,z}が含まれていない場合は、sv{x,y,z}を利用可能範囲に含む地図群を選択する。

Step-2 Step-1 で選擇された地図群から、それらの詳細度を比べ、もっとも詳細度の高い

表1 地図の登録データ

地図の識別子	利用可能範囲	詳細度	建物内外
stand	{E0,N0,Z0 E1,N1,Z1}	8	True
campus	{cx0,cy0,cz0,cx1,cy1,cz1}	10	True
redid1	{rx0,ry0,rz0,rx1,ry1,rz1}	11	false

表2 位置センサの登録データ

センサの識別子	利用可能範囲	優先度
GPS	{E0,N0,Z0 E1,N1,Z1}	1
Cart	{rx0,ry0,rz0,rx1,ry1,rz1}	2

地図を最適な地図として選択する。

goodSensor(MID,SID,sv{x,y,z})の実現方式

SEAMA は、位置センサの自動切り替えを行うために、位置センサについて次の3項目のデータを保持する：(1)位置センサの識別子、(2)利用可能範囲、(3)優先度。

表2は位置センサの登録例を示す。

なお、これらのデータは、位置センサ・ラッパ作成者がSEAMAに事前に登録するものとする。はその識別子の動作が安定しているかどうか、ならびに位置センサから得られた座標sv{x,y,z}その位置センサの利用可能範囲にあるかどうかを判定する。

goodSensorの実行手順は次のとおりである。

Step-1 現在利用している位置センサ(SIDはその識別子)その判定結果として、位置センサの動作が安定している、もしくは、利用可能範囲内にsv{x,y,z}が含まれている場合は、入力されたSIDを返す。すなわち位置センサの切り替えは行わない。

また、その判定結果として、その位置センサの動作が不安定である、もしくは、利用可能範囲内にsv{x,y,z}が含まれていない場合は、sv{x,y,z}を利用可能範囲内に含む位置センサ群を選択する。

Step-2 Step-1の結果から現在利用している地図の利用可能範囲に現在の座標があり現在利用されている地図と重複部分がある利用可能範囲を持つ位置センサ群を選択する。

Step-3 Step-2の結果から、もっとも優先度の高い位置センサを最適な位置センサとして選択する。

2.4 標準操作系の実現方式

SEAMAにおける標準操作機能の仮想化方式は、現在までに提案されている異種センサ群の統合利用のためのシステム[8]や異種情報源の統

合利用のためのシステム[2][3]の実現方式と同様である。詳細はそれらの文献を参考にされたい。

3. 提案システム SEAMA の評価

本節では、次の項目を検証することによりSEAMAの有効性を明らかにする。

検証項目 SEAMAによって、MAの設計、開発に必要な位置センサや地図の知識、技能が削減されること

3.2 実験方法

本実験では、先に述べた検証項目を検証するために、同一の要求達成条件を持ち、MA実行中に利用する位置センサおよび地図の切り替えが発生する2つのMAを構築し、比較を行う。

(MA-1)SEAMAを用いて構築されたMA

(MA-2)SEAMAを用いずに構築されたMA

検証項目については、MA-1を実行させて与えられた要求を達成すること、またMA-1の中位位置センサおよび地図の操作のための制御コードの有無を比較する事、さらにMA-1のプログラムコードが削減されていることを確認する事によって検証する。

3.3 実験環境

本実験では、MA実行中に3つの地図と2つの位置センサの切り替えが発生するMA(実験MA)を設計した。このMAは、提供者の異なる地図や利用可能範囲の異なる位置センサを、1つの活動空間内で利用するMAの代表的な例として位置づけられる。

実験MAに対する要求は、「JR鶴舞駅から名古屋工業大学1号館の高橋研究室に案内する」というものである(図4)。

実験MAでは、鶴舞駅周辺広域地図、名古屋工業大学構内地図、および名古屋工業大学1号館館内地図の3種類の地図を利用した。また、

位置センサとして、屋外の位置取得を行う GPS と屋内の位置取得を行う手押し車の 2 つの異なる利用可能範囲を持つ位置センサを利用した。

実験 MA における地図の切り替えについては、広域地図から構内地図、および構内地図から 1 号館館内地図へ、2 回要するものとした。また位置センサの切り替えについては、鶴舞駅周辺、および名古屋工業大学構内において GPS を用い、1 号館内に入ったときに手押し車に自動的に切り替えを要するものとした。

本実験で設計した MA-1 のプログラムは、図 4 のに示すとおりである。

3.4 結果と考察

結果 1 提案システム SEAMA によって MA-1(図 5)から生成された実行可能プログラム(図 6)を実行した時に、本実験で設定した MA の要求を達成するように動作し、結果としてその要求を達成した事を確認した。

結果 2 図 6 中の*は、MA-1 のコードに含まれないが、MA-1 において利用する位置センサと地図の操作のための制御コードを表している。これより、SEAMA 上での MA 開発において、位置センサと地図の利用可能範囲に関する知識が不要であることを明らかにした。

さらに、MA-2 では、図 6 の getLocation() の代わりに、センサを直接操作するためのコードが埋め込まれていることから、MA-1 の開発において、利用するセンサのコマンド・セットやデータ表現形式に関する詳細な知識が不要であることを確認した。

これらの結果より、SEAMA における MA 開発において、その MA が利用するセンサと地図に関する知識、技能が、従来の MA 開発と比較して削減されることを明らかにした。

また MA-1 と MA-2 についてコード量の比較を行った。その結果を表 3 に示す。これより SEAMA を用いることでコード量を削減することが可能になっていることが分かる。すなわち MA 設計者の MA の設計、開発に関するオーバーヘッドが削減されることを明らかにした。

表 3 MA-1 と MA-2 のコード量の比較

実験 MA	設計者が記述したコード量
MA-1	11 行
MA-2	611 行

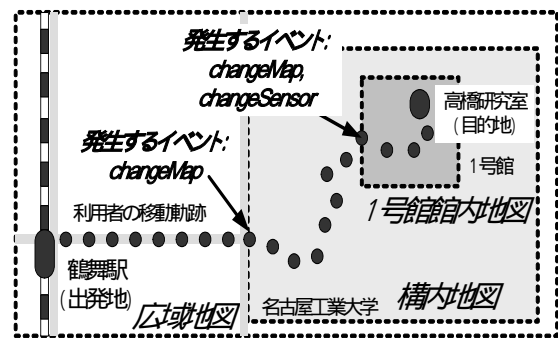


図 4 実験 MA の概要。

```
{Filename = "MA";
Translator tr = new Translator(Filename);
tr.judge(SID,T_area[min],T_area[max],MID,T_action,M_action,S_action);
}
```

```
実験MA = {T_action, M_action, S_action}
T_action = {System.out.println("到着しました。")}
M_action = {popupMap(MID)}
S_action = {System.out.println("センサが切り替わりました。")}
```

図 5 トランスレータ適用前の実験 MA プログラム(MA-1)

```
SID = goodMap();
T_area = projectM(T_area[min],T_area[max],JID);*
popup P = new popup()
while(1){
Coordinate cntLocs = getLocation(SID); *
Coordinate cntLoc = projectS(cntLocs); *
Goal = judge(cntLoc, T_area); *
if(Goal = true){ *
System.out.println("到着しました");
break;
}
else{
}
if(getLocation() == true){ *
}
if(changeMap(MID, cntLoc) == true){ *
MID = goodMap(MID, cntLoc) *
P.popupmap(MID);
}
if(changeSensor(MID, SID, cntLoc) == true){ *
SID = goodSensor(MID, SID, cntLoc) *
System.out.println("センサが切り替わりました");
}
Thread.sleep();
}
```

図 6 トランスレータ適用後の実験 MA プログラム

4. 関連研究

Active Mobile DBMS[7]は、移動体端末が交信

可能状態になった時に、アクションを起動するための機構を実現している。その特徴は、交信可能イベント connect を定義し、そのイベントに対するアクションを実行するための機構を実現する点にある。一方、SEAMA は、異種地図・異種位置センサの統合利用を行う MA において、実行中に発生する4つの本質的なイベントを定義し、MA をそれらの4イベントに対する6つのアクションとして記述・実行可能にするものである。

アクティブデータベースシステム[5]は、能動的に動作可能なデータベースシステムである。ECA ルールは、アクティブデータベースにおいて、能動的な動作記述を行うものである。このシステムは SEAMA と異なり、データベースシステム中に発生するイベント（更新など）に対してアクションを起動するものである。

文献[8]は、異種センサ統合利用システムを実現している。複数のセンサ・ネットワーク群から独立に送られてくる位置情報を品質を落さずに収集するための手法を実現している。

SEAMA は、現在の位置情報が、現在使用しているセンサの利用範囲外になった時、他の利用可能なセンサに自動的に切り替えるための機構を実現することによって、MA 実行中に継続して位置情報を獲得することを可能にする。

5. おわりに

本稿ではモバイルアプリケーション(MA)設計者の持つ位置センサ、および地図の知識・技能を補い、その操作を仮想化して利用することを可能にする基盤ソフトウェアの実現方式を提案する。この SEAMA 用いることにより、MA 設計者は位置センサおよび地図の操作系に関する知識・技能を有することなく MA を設計、開発することが可能となった。

今後の課題として、現在の SEAMA では複数観測点および複数判定領域の複合的な処理を行うことが可能となっていない。よってこの点を実装できるようにする必要がある。

また現在、MA の実行処理および位置センサや地図のオペレーションは一台のマシンで行われている。今後 MA の適用範囲を向上させるためには、位置センサなどの観測点の処理を行うマシンと、MA 本体の処理を行うマシンを切り離し、ネットワーク化していくことが必要となると

考えられる。

さらに、現在は領域を座標のみで指定している SEAMA のオペレーションを、建物や道路などのオブジェクトを用いて指定することが出来るようにする必要がある。それにより、設計者は判定空間や目的物をより直感的に指定する事が可能となり、MA 設計の幅を広げることが出来る。そのためには、オブジェクトと座標の間の対応付けを行なう方式を検討したい。

参考文献

- [1] 天野真家: さまざまな次世代 GPS 測位方式, 情報処理, Vol.43, No.8, pp.827-859 (2002).
- [2] Wiederhold, G.: Mediators in the architecture of future information systems, IEEE Computer, Vol.25, No.3, pp.38-49 (1992).
- [3] 細川宜秀, 石橋直樹, 八代夕紀子, 清木康: マルチデータベース環境における時間的・空間的関連性評価によるデータ結合方式, 情報処理学会論文誌: データベース Vol.40, No.SIG8(TOD4), pp.99-111 (1999).
- [4] 細川宜秀, 清木康: 関数型計算によるマルチデータベースシステムの間合せ処理方式, 情報処理学会論文誌 Vol.39, No.7, pp.2217-2230 (1998).
- [5] 石川博: アクティブデータベース, 情報処理 Vol.35, No.2, pp.120-129 (1994).
- [6] 森下健, 中尾恵, 垂水浩幸, 上林弥彦: 時空間限定オブジェクトシステム SpaceTag プロトタイプシステムの設計と実装: 情報処理学会論文誌, Vol.40, NoSIG10, pp.2689-2697 (2002).
- [7] Murase, T., Tsukamoto, MandNishio, S.: Active Mobile Database System for Mobile Computing Environments, IEICE Transaction on Information and Systems, Vol. E81-D, No. 5, pp. 427-433 (May 1998).
- [8] 白石陽, 安西祐一郎: 位置情報に基づくセンサデータ統合のための逐次データ提供方式, 情報処理学会データベースと Web 情報システムに関するシンポジウム (DBWeb2002) 予稿集, pp.153-160, (2002).