

状態遷移を用いたコンテキストウェアサービスの開発方法の提案

2014SE083 坂口 樹奈 2014SE106 鷲頭 瑠奈

指導教員 青山 幹雄

1 研究背景

現在、ユビキタスコンピューティング環境が普及し、いつでもどこでも高度なサービスを利用できるようになっている。また、センシングデバイスの普及により、ユーザの置かれている状況やユーザの状態をセンサから収集可能となり、ユーザの状況に応じたサービスを提供するコンテキストウェアサービスの実現が求められている。

2 研究課題

本研究では背景を踏まえ、以下の2点を課題とする。

- (1) コンテキストモデルとそれを用いたコンテキストウェアサービスの開発方法の確立
- (2) コンテキストモデルの完全性、一貫性の保証

3 関連研究

3.1 コンテキストとコンテキストウェア

- (1) コンテキストとは

Dey はコンテキストを「エンティティ(entity)の状況の特徴づける情報である。」と定義している[1]。

本研究では、コンテキストをユーザの振舞い、ユーザ内面、環境、時空間として定義した。

- (2) コンテキストウェアとは

Dey はコンテキストウェアを、「ユーザのタスクに依存し、ユーザに対し情報および/またはサービスを提供するためにコンテキストを使用するシステム」と定義した[3]。

3.2 コンテキストモデル

コンテキストモデルとはコンテキストを表現するモデルのことで、以下の6つに分類される[2]。

- (1) キーバリューモデル
- (2) マークアップスキーマモデル
- (3) グラフィカルモデル
- (4) オブジェクト指向モデル
- (5) 論理モデル
- (6) オントロジモデル

3.3 状態遷移モデル

振舞いは状態遷移で表現される。オブジェクト指向の状態遷移の表現方法として UML のステートマシン図がある。ステートマシン図はシステムやクラスの「状態」と「状態遷移」を表現する[4]。

4 アプローチ

研究課題を解決するためのアプローチを以下に示す。

- (1) ステートコンテキスト図の提案

ユーザの振舞いに着目し、その変化を状態遷移で表現する。UML のステートマシン図に基づいて、コンテキストの変化に応じてユーザの状態の遷移を表現するステート

トコンテキスト図を提案する。

- (2) サービス主導のコンテキストウェアサービス開発方法の提案

ステートコンテキスト図を用いたコンテキストウェアサービスのサービス主導の開発方法を提案する。

5 サービス主導の開発方法

5.1 コンテキストウェアサービス開発プロセス

次の3つのプロセスにおいてコンテキストウェアサービスを開発するための新しい方法を提案する。

- (1) 要求定義

コンテキストウェアサービスの提供者の要求をもとにステートコンテキスト図を作成する。

- (2) 単体試験

個別ステートコンテキスト図を用いて検証する。

- (3) システム試験

全体のステートコンテキスト図を用いてシステムが要求通りに動くか妥当性確認を行う。

5.2 状態遷移型のコンテキストモデル

- (1) コンテキストのメタモデル

本研究ではコンテキストを以下のように定義する(図 1)。

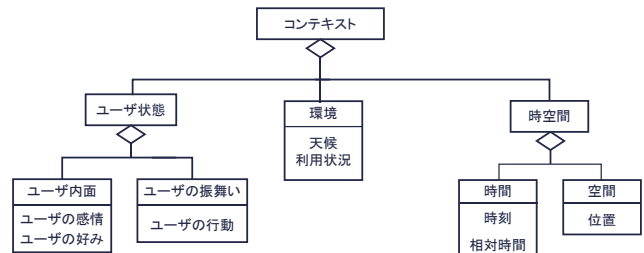


図 1 コンテキストのメタモデル

- (2) ステートコンテキスト図

状態遷移型のコンテキストモデルを定義する図として、状態遷移を表現するステートマシン図に基づいた「ステートコンテキスト図」を提案する。ステートコンテキスト図は、「コンテキスト状態」、「コンテキストイベント」、「サービス定義」の3つの要素で構成される。図 2 にステートコンテキスト図の定義を示す。



図 2 ステートコンテキスト図

本研究では、コンテキストイベントを、「事象の変化によりコンテキストに変化をもたらすもの」と定義している。状態遷移型のコンテキストモデルを、「コンテキストを状態とみなし、コンテキストイベントによりある状態からある状態へ遷移するモデル」と定義する。遷移後にその状態に適

合するコンテキストウェアサービスの提供を行う。このコンテキストモデルを用いてサービスの開発を支援するため、モデルには「提供されるサービス」を組み込んだ。

ステートコンテキスト図は、ステートマシン図に基づいている。ステートコンテキスト図はステートマシン図と表 1 のように対応している。

表 1 ステートマシン図とステートコンテキスト図の対応

ステートマシン図	ステートコンテキスト図	
	定義	意味
イベント	コンテキスト イベント	ユーザ内面、 環境、時空間
状態名	コンテキスト状態	ユーザの振舞い
アクション	サービス定義	状態遷移に伴う サービス
アクティビティ		サービス提供

ステートマシン図では、システムの振舞いを状態遷移で記述しているが、ステートコンテキスト図ではユーザのコンテキストをイベントや状態名とすることで、コンテキストの変化を表現することが可能である。

ステートコンテキスト図は階層構造を持つ。ステートコンテキスト図の作成プロセスは、上位からレベル 1、レベル 2、レベル 3 のようにこれ以上分割できなくなるまで、段階的に詳細化していく。

5.3 サービス主導によるコンテキストウェアサービスの開発方法

(1) サービス主導による開発

サービス主導による開発とは、開発者が初めにサービスを発案し、そのサービス利用時のユーザの状態を特定し、特定した状態へのコンテキストイベントを定義するという方法である。サービス主導による開発ではステートコンテキスト図を用いることにより、コンテキストウェアサービスの開発を効率化でき、ユーザ状態の遷移を表現できる。

(2) ステートコンテキスト図レベル 1

ステートコンテキスト図レベル 1 の作成プロセスを図 3 に示す。

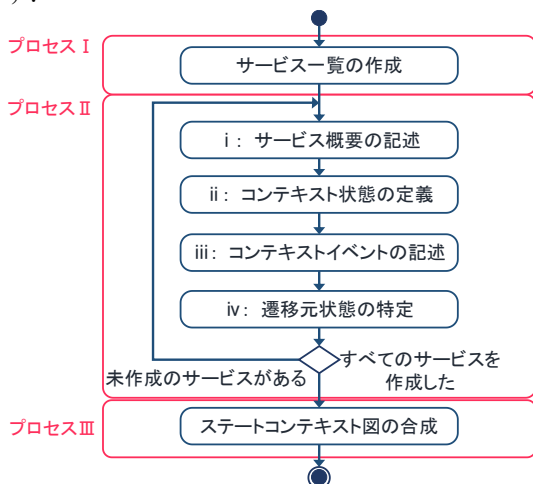


図 3 ステートコンテキスト図レベル 1 の作成プロセス

(プロセス I) サービス一覧の作成

実現したいサービスを複数あげ、サービスの一覧を作成する。

(プロセス II) 個別ステートコンテキスト図の作成

サービス一覧の中からサービスの一つを取り上げ、個別ステートコンテキスト図を作成する。

(プロセス III) ステートコンテキスト図の合成

各個別ステートコンテキスト図に関して、同一状態があれば一つの状態として融合し、すべての図を合成する。

融合とは個別ステートコンテキスト図に同一状態がある場合にその状態を一つの状態とすることである。合成とは同一状態がすべて一つの状態になるまで融合を繰り返すことである。単一状態であるときに限り融合可能であるとする。合成アルゴリズムを以下に示す(図 4)。

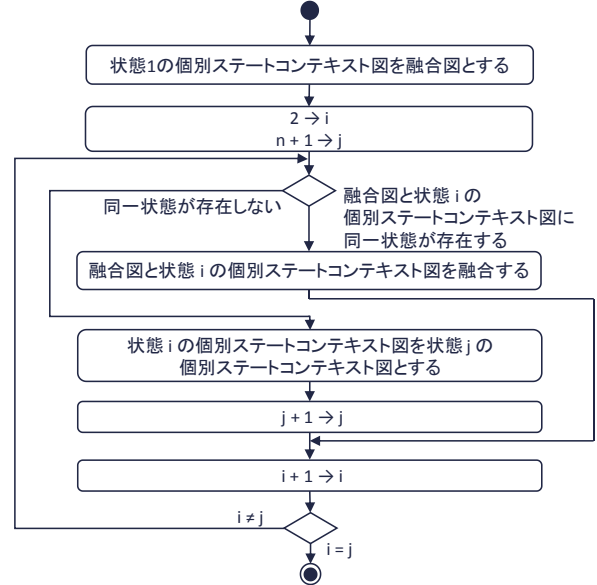


図 4 プロセス III のステートコンテキスト図の合成プロセス
融合の例を以下の図 5 に示す。

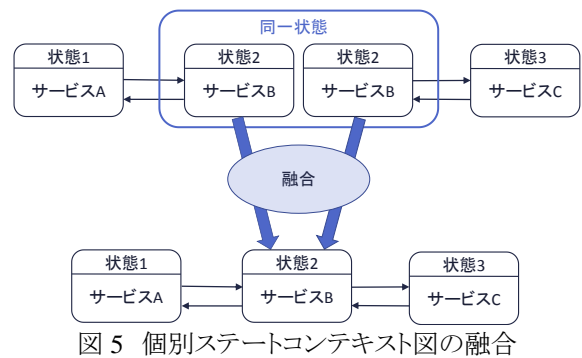


図 5 個別ステートコンテキスト図の融合

(3) ステートコンテキスト図レベル 2

以下にステートコンテキスト図レベル 2 の作成プロセスを記述する(図 6)。

1) サービスの分割

レベル 1 で作成した個別ステートコンテキスト図のサービス定義を分割し、記述する。また、サービスを分割した結果、サービス実行時に必要な処理が新たに見つかった場合も記述する。

2) 入場アクションを entry, 退場アクションを exit として記述

分割したサービスから入場アクションと退場アクションを抽出して記述する。入場アクションと退場アクション以外に個別ステートコンテキスト図の状態へ遷移するための処理が存在する場合、作用(effect)として記述する。

態を定義する。分割したサービスは並行して実行されるので直交状態として記述する(図 11)。

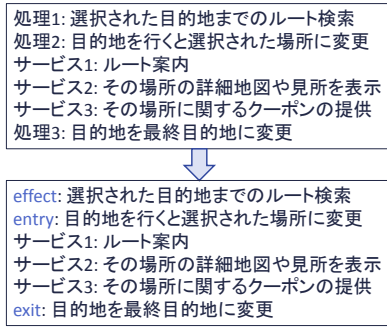


図 10 サービスの分割

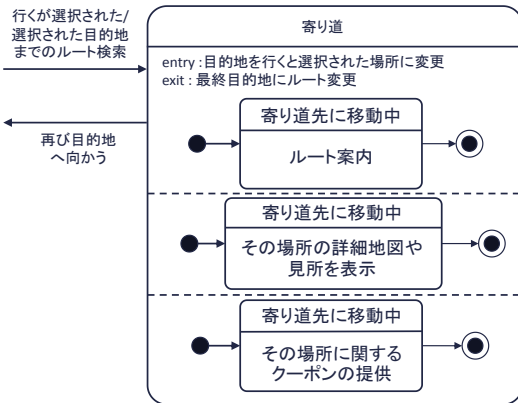


図 11 寄り道のレベル 2 の状態コンテキスト図

7 評価

課題として以下の 2 点をあげ、評価を行った。

7.1 コンテキストモデルとそれを用いたコンテキストウェアサービスの開発方法の確立

(1) コンテキストモデルの確立

状態コンテキスト図レベル 1 では、サービスを定義し、そのサービスが提供されるときにユーザ状態を定義した。その後、コンテキストの変化をコンテキストイベントとして定義した。レベル 2 ではレベル 1 で定義したサービスを詳細化し、その時のユーザ状態を定義した。よって、状態コンテキスト図により、詳細化されたサービス提供時のユーザ状態を遷移で表現することが可能となった。

(2) 開発方法の確立

サービス主導で開発を行うことにより、考慮すべき状態の数が減り、コンテキストの複雑さを軽減することができる。従来のコンテキストの定義では図 1 の 8 つの要素それぞれについて考慮することになり、それらを状態変数とするとその組み合わせの数は $2^8 - 1 = 255$ 通りとなる。これに対してサービス主導の開発ではサービス提供時のコンテキスト状態の決定に必要な状態変数の組み合わせのみを考えればよい。例題では個別状態コンテキスト図の数である 6 通りとなる。

7.2 コンテキストモデルの完全性、一貫性の保証

(1) 完全性の保証

状態コンテキスト図は、状態を定義するとき、必ずイベントや遷移元状態も特定している。さらに、個別状態コンテキスト図を融合するときに記述漏れや不完全な記

述がないか確認できる。

(2) 一貫性の保証

提案した状態コンテキスト図では、一つのサービスを提供するために必ず一つの状態を定義しているため、状態は常に一意に決定される。また、レベル 1 の作成プロセスにおいて、状態コンテキスト図の合成は状態や遷移を基に融合している。そのため同じ状態から同じイベントで異なる状態へ遷移するなどの矛盾の特定が容易になり、状態遷移の一貫性を保つことができる。

(3) ユビキタスコンピューティング環境への対応

従来の開発方法では、ユーザの状態を特定し、その状態に合ったサービスを提供するため、用いるセンサの数、種類が増大する。しかし、サービス主導の開発方法では、考慮すべきユーザ状態が減るため、必要となるセンサも減らすことができる。

8 今後の課題

今後の課題を以下に示す。

(1) 状態コンテキスト図の詳細な定義

サービスごとの分割基準や、状態を特定するセンサ値の決定方法、状態の分割基準などのルール追加や改善が必要である。

(2) 他のサービスへの適用

本研究で対象としたカーナビゲーションシステム以外のコンテキストウェアサービスにも適用し、提案方法の妥当性や有効性を検証する必要がある。

9 まとめ

本研究ではユーザの振舞いの変化を状態遷移とする状態コンテキスト図を用いた開発方法を提案した。状態コンテキスト図は、ユーザの振舞いの状態遷移を全体の遷移から個別の状態の詳細まで階層的に表現した。提案したモデルをカーナビゲーションシステムのコンテキストウェアサービスの開発に適用して、モデルの有効性を検証した。状態コンテキスト図を用いて動的に変化するコンテキストによりユーザ状態の遷移を表現できた。サービス主導の開発では、サービスの定義後にユーザ状態を定義するので考慮すべきコンテキストの数を削減できる。また、完全性、一貫性に関して、記述漏れや不完全な記述の特定が容易となる。今後は、サービスの分割基準の確立、センサの値の決定方法の定義が課題である。

10 参考文献

- [1] A. K. Dey, et al., Toward a Better Understanding of Context-Awareness, Handheld and Ubiquitous Computing, Springer, 1999, pp. 304-307.
- [2] T. Strang, et al., A Context Modeling Survey, Proc. of 1st Int'l Workshop on Advanced Context Modeling, Reasoning and Management, UbiComp 2004, Sep. 2004, 8 pages.
- [3] 鈴木 健太 ほか, コンテキストの類似度を用いた動的コンテキストウェアサービス提供アーキテクチャの提案と評価, 情報処理学会第 76 回全国大会 講演論文集, Vol. 3, 4X-10, Mar. 2014, pp. 411-412.
- [4] 竹政 昭利 ほか, かんたん UML 入門, 改訂 2 版, 技術評論社, 2017.