

コンテキストを活用したモバイルアプリケーション開発の要求定義を支援するパターンと適用手法

大橋恭子^{†1} 栗原英俊^{†1} 中里克久^{†1} 山本里枝子^{†1}

ビジネス環境の変化によって様々な環境でコンピュータシステムが使われるようになった。また一方でコンピュータシステムが、ユーザにより良いユーザ体験(UX)を提供する動きも注目されている。UX を高める技術としてコンテキストウェア技術がある。本論文では、コンテキストウェア・アプリケーション開発の要求定義工程を支援するパターンとその適用手法を示す。このパターンを用いることで、コンテキストの知識が乏しい要求定義の担当者であってもコンテキストを活用したアプリケーションの要求を抽出することができる。また、このパターンの有効性を確認した結果も報告する。

The patterns and application method for requirements definition for mobile application development using contexts

Kyoko Ohashi^{†1} Hidetoshi Kurihara^{†1}
Katsuhisa Nakazato^{†1} Rieko Yamamoto^{†1}

A transition of business environments has caused usage of computer systems under various surroundings. On the other hand, we hope to provide good User Experience (UX) to computer systems. Context is one of the technologies to improve good UX. In this paper, we show the "context-aware application patterns" and the application process of the patterns. Even if a requirement engineer who does not have enough context knowledge, he/she can elicit context aware requirements. In this paper, we also report a result of evaluation of these patterns' effectiveness.

1. はじめに

近年の情報化社会の拡大に伴いビジネス環境が変化しており、出張先や移動中といった社外、事務所以外の店頭や倉庫等で業務を行うモバイルワークが求められている[1].

一方コンピュータシステムの分野では、ユーザ体験(User Experience, UX)やコンテキスト注目され始めている[2,3]. ユーザ体験とはユーザがある製品やシステムを利用した時に得られる経験や体験である。Deyらは、コンテキストとはユーザ自身やユーザとインタラクションを行う人や場所やアプリケーション等の状況であり、コンテキストウェア・アプリケーションとはコンテキストを利用してユーザに適切なサービスや情報を提供するアプリケーションだと定義している[4]. 代表的なコンテキストは、ユーザの居場所である。コンテキストに応じた情報やサービスをユーザに提供することは、ユーザにより良いUXを提供することに繋がる。

これまでユーザのコンテキストを推定することは困難であったが、近年の携帯端末や各種センサーの普及によりコンテキストの推定に必要な情報を容易に取得する環境が整いつつある。例えば、多数のユーザを持つRuntastic[5]というコンシューマ向けのアプリケーションは、スマートフォン等に内蔵されているGPSや加速度センサー等を用いて走行ルートや走行ペースといったコンテキスト推定し、さら

にそれらを利用した情報をユーザに提供している。

一方、ビジネス向けには広く知られているコンテキストウェア・アプリケーションはない。我々はその理由を、ビジネスアプリケーションの開発者、特に要求定義の担当者にとって、コンテキストの認知度がまだ低く、コンテキストの知識や活用した経験が乏しいからだと推測した。要求定義には業務知識が必要であるため、業務知識がなくてコンテキストのみに詳しい者が要求定義を担当することはできない。そのため要求定義の担当者にコンテキストやコンテキストウェア・アプリケーションに関する知識を習得してもらうことが必要である。

知識を伝える手段としてパターンが有効であることは知られている。またパターンはパターン名、問題、解法、結果の4つの基本的な構成要素を有するといわれている[6]. 我々はコンテキストウェア・アプリケーション開発の要求定義作業を支援するために、これらの基本構成要素を具体化し、コンテキストやその使い方を示したパターンを作成する取り組みを行った。そのパターンがコンテキストウェア・アプリケーション・パターン(以後、単にアプリパターンと呼ぶ)である。今回このアプリパターンの有効性を評価したので報告する。

本稿では、アプリパターンの適用手法の概要を2.で、アプリパターンの定義と例を3.で説明し、4.でアプリパターンの有効性を評価した結果を示す。最後に5.で関連研究を示し、6.で我々の取り組みをまとめる。

^{†1}(株)富士通研究所
Fujitsu Laboratories Ltd.

2. アプリパターン適用手法の概要

要求を定義するプロセスは①要求抽出、②要求分析、③④要求仕様化、④要求の確認という4つのプロセスから構成されている[7].

我々の提案するアプリパターンは①の要求抽出プロセスで利用する。このプロセスでは、新業務や新システムの利用方法を示したシナリオ(ToBeシナリオと呼ぶ)を作成する。シナリオには、ユーザがある目標を達成するために行う行動(以後、ユーザ行動と呼ぶ)が記述される。

我々の提案するアプリパターン適用手法では、アプリパターンを適用した ToBe シナリオを次の手順で作成する。

- 作業1: コンテキストを意識せずに理想的な ToBe シナリオを作成
- 作業2: アプリパターンを参照しながら ToBe シナリオ中のユーザ行動のUXを向上させるためにパターンを適用できる部分を見つける
- 作業3: 作業2で見つけた部分にアプリパターンを適用して ToBe シナリオをブラッシュアップ

要求抽出プロセスのアウトプットである ToBe シナリオは、②の要求分析プロセスにおける要求の必要性や優先順位付けなどの議論を経て、③の要求仕様化プロセスで行う画面レイアウトやシステム機能仕様作成時のインプットになる。

3. アプリパターン

我々はパターンの基本構成要素であるパターン名、問題、解法、結果のうち、解法と結果を要求定義で利用しやすいように具体化した。UMLのクラス図[8]で記述した具体化後のモデルを図1に示す。図中の点線枠はパターンの4つの基本構成要素とアプリパターンの構成要素との対応を示したものである。

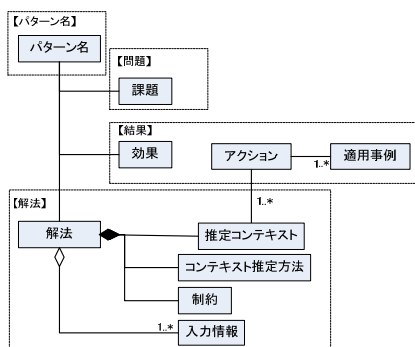


図1: アプリパターンのモデル

基本構成要素の解法は、パターンで推定する推定コンテキスト、コンテキスト推定に必要な入力情報、入力情報からコンテキストを推定する方法を示したコンテキスト推定方法、制約に具体化した。また、もう一つの基本構成要素である結果は、効果とアクションと適用事例に具体化した。効果には、パターン適用によって向上する品質と品質向上の根拠を示す。アクションは推定したコンテキストを用い

てユーザに提供できるサービスの概要を示すもので、適用事例はそのサービスをより具体的に示したものである。

図2は、前記のモデルに従って作成したアプリパターンの定義例である。この定義では制約を事前条件や事後条件、考慮すべき例外処理に分けて記述できるようにしている。またコンテキスト推定方法は、推定の処理ステップを記述できるようにした。我々は、推定コンテキストから少なくとも「主体(誰/何)」が、対象(何)に対して、どのような状況であるか」を取得できるようにしなければならないと考えている。アプリパターンの定義では、状況は推定コンテキスト欄に記載し、主体や対象等はコンテキスト推定方法等に記載する。

項目	記述内容
パターン名	距離を利用した情報表示
効果	・作業忘れを防止することによるユーザビリティの向上 ・端末操作数の削減によるユーザビリティの向上
課題	・利用者と対象(*)の間の距離によって、利用者に必要な情報が異なる。 (*)例: 人、モノ、場所等 これまででは、利用者が対象との距離を推測し、距離に応じて適切な情報を表示していた。そのため、情報を取得をし損ねることがあった。
アクション	利用者と想定するモノや場所との離れ具合に応じた情報を提示する
入力情報	・利用者の位置情報 ・対象の位置情報
推定コンテキスト	近接コンテキスト: 遠隔、近接
事前条件	対象(モノ・場所)とその座標を取得できること
例外処理	端末利用者の位置と対象の位置の両方もしくは片方が不明な場合
事後条件	—
コンテキスト推定方法	Step1 端末の位置を取得する 取得する情報: 現在座標
	Step2 全対象物の位置を取得する 取得する情報: 対象座標リスト
	Step3 現在座標と対象座標リスト中の座標との距離を算出。距離の値を用いて近接コンテキストを作成する
適用事例	近接時に、対象の詳細な情報の確認を促すメッセージを表示する
	近接時に、対象の詳細な情報を表示する コンテキストが変わった時にアプリケーションを起動

図2: アプリパターン定義の例

我々はこれまでに主に業務の実施忘れの防止や端末操作を作業負荷軽減に寄与する18個のアプリパターンを定義した。その一部を示す。

- 利用者と対象の距離に応じたサービス提供
- 日付や時刻を利用した適切なサービス提供
- イベント登録情報と利用者の位置情報を利用したサービス提供
- 選択履歴を利用した選択肢の順序付け

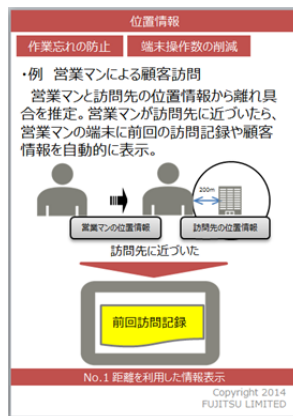


図3: アプリパターンカードの例

図2の書式で記述された定義は、アプリパターンの理解には適していたが、パターン選択時に参照するにはあまり適していなかった。そこで、図3に示すようなアプリパターン選択に最低限必要な情報をイラストを交えて記述したカードも作成した。

4. 評価

我々はアプリパターンの有効性を確かめるために、アプリパターンを用いると ToBe シナリオ内のユーザ行動に対してより多くの改善案が提案可能かを測定した。その結果を報告する。

4.1 被験者と評価に用いたデータ

評価を実施した被験者は計4名で、アプリパターンの知識を持っていない2名のグループ A、知識を持っている2名のグループ B の2グループで構成した。コンテキストや評価に用いた業務に対する知識は全員同程度である。

評価に用いたデータは、携帯端末を持った医療関係者が患者宅を訪問して診療を行う業務を題材にした AsIs シナリオである。

4.2 測定項目と測定方法

我々はアプリパターンが有効であれば、2章に示した作業2において、グループ B はグループ A よりも多くの改善案を提案可能だと考えた。

そこで、ToBe シナリオ内のユーザ行動に対してコンテキストを用いた改善案を作成し、各グループが作成した改善案の数 $\text{Prop}(t)$ でアプリパターンの有効性を判断することにした。t は改善案を作成したグループを示しており、 $\text{Prop}(A)$ や $\text{Prop}(B)$ はグループ A または B のみが作成した改善案の数、両グループに共通の改善案の数は $\text{Prop}(\text{Both})$ で示す。

$\text{Prop}(t)$ の測定は次の3手順で行った。手順①②は2章の作業2、3に該当し、手順③は測定結果の分析作業である。

①AsIs シナリオ内のユーザ行動から改善すべき課題の抽出と抽出した課題を解決した ToBe シナリオの作成を被験者全員で実施

②作成した ToBe シナリオに対し、コンテキストを用いた改善案をグループ別に作成。この時グループ B は、アプリパターンを参照する

③被験者全員で各改善案の内容の妥当性と改善案の同一性の判断を行い、妥当な改善案を作成グループ別に分類

4.3 測定結果と考察

改善案の分類結果を表1に示す。グループ A のみの改善案($\text{Prop}(A)$)は2個、グループ B のみの改善案($\text{Prop}(B)$)は9個、両グループ共通の改善案($\text{Prop}(\text{Both})$)は4個であった。

表1: 改善案の分類結果

グループ名(t)	A 知識無し	Both	B 知識有り	総数
改善数($\text{Prop}(t)$)	2	4	9	15

グループ別の改善案の作成数の割合は、 $\text{Prop}(t) + \text{Prop}(\text{Both})$ で表すことができる。グループ B の改善案の総数は13個、グループ A の改善案の総数は6個で、アプリパターンの知識が無いグループと比べ、知識があるグループは約2.2倍の改善案を作成している。各グループ独自の改善案の数で比較するとその差はさらに広がり、グループ B はグループ A の4.5倍の改善案を作成している。グループ

間にはコンテキストや業務の知識の差はないので、これらの差はアプリパターンの知識の差によるものと考えられる。我々はこれらの差からアプリパターンの有効性を示すことができたと判断した。

グループ A は2個の独自の改善案を作成している。これらは、異常発生時の作業やシナリオ範囲外でのユーザ行動に対する改善案であった。グループ B は9個の独自の改善案を作成している。そのうちの6個は3種類のアプリパターンを適用したものであった。この6個のうち、3個には距離を利用したアプリパターンを適用していた。これは評価に用いたシナリオがユーザの移動を伴う業務であったことによると思われる。また、既存のパターンに該当しない改善案が3個あった。これらはユーザが携帯する端末には搭載されていないセンサーの情報やセンサーの情報と画像解析等の技術を組み合わせたものであった。

5. 関連研究

コンテキストの使い方に合わせた小さなアーキテクチャ(micro-architecture)を設計パターンとして提供している研究がある[9]。パターンの一つとして、コンテキスト値毎に異なるサービスを提供する時のアーキテクチャを示している。これは、Gamma らの提案しているデザインパターンで構造の項に示されている抽象的な構造に該当するものである。ユーザに提供するサービスや情報の検討に利用するものではない点が我々の研究と異なる[6]。

6. まとめ

本稿では、コンテキストの知識や活用経験が乏しい要求定義の担当者にも利用可能なアプリパターンとその適用手法、およびアプリパターンの有効性も示した。今後は、より多くのアプリパターンの収集や、アプリパターンのサンプルコードを開発することで、コンテキストウェア・アプリケーションのプロトタイプや実装の開発期間を短縮に取り組む。

- 1) ワークスタイル変革,
<http://jp.fujitsu.com/solutions/offerings/workstyle/>
- 2) 川西裕幸, 潮田浩, 栗山進, UXデザイン入門, 日経BP社(2012).
- 3) 牧慶子, 中道上, 青山幹雄, 意図に応じたコンテキストウェアサービス提供モデルの提案と評価, 情報処理学会研究会報告, Vol.2013-SE-179, No.28, pp.1-8.
- 4) A.K.Dey, "Understanding and using context", Personal and Ubiquitous Computing, pp.4-7(2001).
- 5) Runtastic, <https://www.runtastic.com/ja/about>
- 6) E. Gamma, R. Helm, R. Johnson, J. Vissides 著, 本位田真一, 吉田和樹監訳, デザインパターン, ソフトバンクパブリッシング, pp.15 (1999).
- 7) 山本修一郎, 連載 要求工学 第4回要求工学プロセス, 月刊ビジネスコミュニケーション,
<http://www.bcm.co.jp/site/youkyu/index.html>
- 8) Object Management Group 著, 西原裕善監訳, UML2.0 仕様書, オーム社 (2007).
- 9) Gustavo Rossi1, Silvia Gordillo, Fernando Lyardet, Design Patterns for Context-Aware Adaptation, SAINT-W'05.