

## ユビキタスコンピューティング環境のための 作業継続可能なユーザインタフェース切替機構

3K-04

守分 滋<sup>1</sup> 村瀬 正名<sup>2</sup> 永田 智大<sup>2</sup> 西尾 信彦<sup>2</sup> 徳田 英幸<sup>1,2</sup><sup>1</sup> 慶應義塾大学 環境情報学部 <sup>2</sup> 慶應義塾大学大学院 政策・メディア研究科

### 1 はじめに

近年、ユビキタスコンピューティング環境におけるユーザインタフェース (User Interface: UI) の研究が盛んである。Document-based Framework[2], MOCA[3], そして ICrafter[4] はいずれもユビキタスコンピューティング環境での UI フレームワークを提案している。

本研究は、こういった先行研究を踏まえ、ユビキタスコンピューティング環境において、ユーザが UI を切り替えても継続的にサービス进行操作できる新たな機構、i-face を提案する。サービスとは、ユーザが操作できるデバイスやアプリケーションを指す。

本稿では、UI の切り替えと作業の継続を考慮した UI フレームワーク、i-face の設計・実装を行った。UI を切り替えたときの作業の継続を可能にすることで、切り替え後に再設定する手間を省いた。また、UI が自由に切り替えられることで、ユーザは状況に合ったモダリティによる UI でサービスを利用できる。

### 2 ユビキタスコンピューティング環境の UI

従来、UI はサービスとセットで提供されてきた。テレビの UI はリモコンであり、ライトの UI は壁のスイッチといったように、ユーザがサービス进行操作する際の UI は一つに限定された。しかし、これでは無数のサービスが存在するユビキタスコンピューティング環境において、ユーザは無数の UI の利用を強いられてしまう。

これに対し、Universal Controller という全てのサービスの UI を一つにまとめる手法がある [1]。ユーザは一つの端末から GUI を利用して、全てのサービス进行操作する。Universal Controller の問題は、モダリティが一つに限定される点である。操作のための端末がないと、ユーザはサービス进行操作できなくなってしまう。

ユーザは状況に適したモダリティの UI を利用し、かつ継続的にサービス进行操作できなければならない。最適なモダリティは、使用できる機器、まわりの環境、そしてユーザの状態によって異なる。周りにディスプレイがなければ GUI は表示できず、満員電車ではジェスチャは迷惑で、電話をしているときは音声 UI を利用すると相手方が混乱する。

#### 解決すべき課題

ユビキタスコンピューティング環境における UI を実現するには、解決すべき課題が 3 つある。

#### A. サービス操作法の抽象化

サービスの操作法を UI に対して抽象化する必要がある。UI とサービスの結びつきが強いと、サービスを様々な UI から操作することができない。Document-based Framework[2] は、サービスの操作法をメソッドとその引数として抽象化している。

#### B. UI の動的生成

UI は、ユーザがサービス进行操作するときに動的に生成する必要がある。UI はモダリティごとに多数存在し、サービスの UI を全て事前に用意するのは現実的でない。サービスを記述した共通のドキュメントから、個々の UI を生成すれば、事前に UI を用意せずともサービスを操作できる [4]。

#### C. UI 切り替えの考慮

サービスを操作するときに、ユーザによって UI が切り替えられることを考慮する必要がある。ユーザは、状況が変わり、いままで使っていた UI のモダリティが新しい状況に適さなければ、別の UI に切り替える。本研究は C. に注目し、切り替えた後の UI に前の UI の状態を復元することで作業の継続を実現した。

### 3 本研究の概要

#### 3.1 UI の状態の復元

サービスとの依存関係から、UI 切り替えの際に復元する状態は二つに分類できる。

- サービス状態依存な UI 状態
- サービス状態非依存な UI 状態

本稿では、これらの二つの状態それぞれに対して復元方法を実現する。

##### 3.1.1 サービス状態からの算出による復元

一部の UI の状態はサービスの状態を反映して変化する。例えばビデオデッキをサービスとして、GUI からの操作を考える。ビデオデッキにビデオカセットが入っていないければ再生はできない。そのため、GUI の再生ボタンはグレーアウトして使えない状態にある。UI の「再生ボタンは使えない」という状態は、サービスの「ビデオカセットが入っていない」という状態から導くことができる。そのため、サービス状態依存な UI 状態は、UI の状態として明示的に保存せず、復元するときはサービスの状態から算出する。

##### 3.1.2 UI 状態の明示的な保存による復元

一部の UI の状態は、サービスの状態とはかかわりなく変化する。ビデオデッキの例で録画予約をする場合、チャンネルや時間を設定する際のテキストフィールドの中身がこれに当たる。全ての項目を決定し、送信するまでビデオデッキの状態は変化しない。そのため、サービス状態非依存な UI 状態は、サービスの状態とは別に保存し、それを利用して復元する。

複数のユーザが使えば、それぞれのユーザごとに状態が存在するため、サービス状態非依存な UI 状態は、一つのサービスに複数存在しうる。例えば、二人が別々に録画を予約する場合、それぞれが入力している時刻やチャンネルは互いに異なる。そのため、サービス状態非依存な UI 状態はユーザごとに保存する必要がある。

#### 3.2 UI 状態の記述

保存される UI の状態は、異なる UI に復元可能とするために、個々の UI に依存しない形で記述しなければならない。個々の UI は、2 の B. で述べたようにサービスの記述から生成される。この共通なサービス記述に属

User Interface Switching Mechanism with State Preservation for Ubiquitous Computing Environments

Shigeru Moriwake<sup>1</sup> Masana Murase<sup>2</sup> Tomohiro Nagata<sup>2</sup>

Nobuhiko Nishio<sup>2</sup> Hideyuki Tokuda<sup>1,2</sup>

<sup>1</sup>Faculty of Environmental Information, Keio University. SFC, Endo5322, Fujisawa, Kanagawa 252, JAPAN

E-Mail: t99919sm@sfc.keio.ac.jp

<sup>2</sup>Graduate School of Media and Governance, Keio University

Deep thanks to Takeshi Iwamoto for all the support.

