

## 適応型ユーザインタフェースを実現するシステムの構築

5 Q-4

長崎 等 木村泰己 野中 誠 東 基衛  
早稲田大学理工学部

### 1. はじめに

一連の適応型ユーザインタフェース研究の一環として、本発表では以下の適応の概念及び具体的な実現方法について言及する。1つはUIを構成するオブジェクトの構成データを書き換えることによって適応を実現する部品適応、もう1つはオブジェクトのメソッドの置き換えによって操作プロセスの適応を実現するプロセス適応である。またその整合性をとるためのメカニズムについても言及する。

### 2. システムアーキテクチャモデル

本システムは図1のシステムアーキテクチャモデル[1]をベースとしている。このモデルはユーザインタフェースの動的な変更をサポートすることが出来るように設計されている。このモデルにおける Presentation Component, Presentation Control, Dialogue Control の各部分が適応の対象となる。

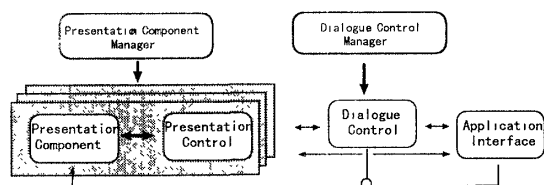


図1：システムアーキテクチャモデル

### 3. UI 適応の概念

#### 3.1. 部品適応

部品とは Window, Frame, Dialog 等に配置される Button などのコンポーネントである。

部品適応は Presentation Component で取り扱う部品を対象とした適応で、以下の様な適応をおこなう。

1. 部品の変更による適応
2. 部品の外観の変更による適応
3. 部品のレイアウトの変更による適応

またUI部品の適応を支援するために、GUIにおける操作方法を[2]を参考に、メニュー選択操作、空欄入力操作、直接操作の3つに分類した。部品適応はこれらのカテゴリーの範囲を超えてはおこなわない。

また部品適応を行う際の適応の影響範囲について、次の2種類に分類できる。

1. 単一の部品のみでの適応
2. 同一の部品の全てに対する適応

#### 3.2. プロセス適応

Presentation Control, Dialogue Control の両者に対する適応である。前者に対する適応ではアプリケーションのユーザイベントに対する振る舞いに変化する。また後者に対する適応ではタスクの実行手順を変化させることができる。

Presentation Control に対する適応は Window (Frame, Dialog) 内におけるイベント処理の変更による適応であり、Dialogue Control に対する適応は、1つの Window の範囲を超えた制御の変更による適応である。

### 4. 適応実現のためのメカニズム

#### 4.1. メカニズムの構成要素

図1の概念モデルを実際のアプリケーションに適用した具体的な構成のイメージが図2である。

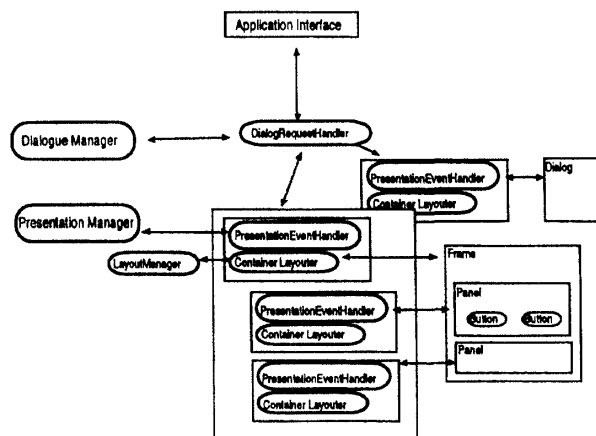


図2：メカニズムの構成要素

Development of Adaptive User Interface Systems

Hitoshi Nagasaki, Yasuki Kimura, Makoto Nonaka, and Motoei Azuma

School of Science and Engineering, Waseda University

以下に構成要素の説明を示す。

#### Dialog Manager

Dialog Request Handler を生成・置換・消去する。

#### Presentation Manager

Presentation Event Handler を生成・置換・消去する。

#### Layout Manager

Container Layouter を生成・消去する。また各 Container Layouter 間の整合性の管理も行う。

#### Dialog Request Handler

対話制御を行うがイベント処理は行わず各オブジェクトの要求を処理する形で制御を進める。

#### Presentation Event Handler

Frame 等の各コンテナのイベント処理を行う。

#### Container Layouter

各コンテナの配置，外観，部品の管理を行う。

### 4.2. 部品適応

部品適応実現の流れを図2を用いて説明すると、まず適応管理部分からの要求を受けた Layout Manager が対象部品を直接管理している Container Layouter に対して部品変更，配置変更，外観変更の要求を出す。Container Layouter が現在の部品を消去して、新しい部品のインスタンスを生成することによって部品変更を実現する。また部品のインスタンスの持つ属性を変更し、表示を更新することによって外観や配置の変更を行う。

### 4.3. プロセス適応

プロセス適応実現の流れを図2を用いて説明すると、まず適応管理部分より要求を受けた Dialog Manager もしくは Presentation Manager が Dialog Request Handler ， Presentation Event Handler のイベント処理メソッドを置き換える。その際、野中らによるメソッド適応の機構 [3] を用いてメソッドの置き換えを行う。 Presentation Event Handler の変更にとどまる場合の適応は単一のイベント処理メソッドの置き換えで完了するが、 Dialog Request Handler の要求処理メソッドを置き換える必要がある場合は、それに関連する Presentation Event Handler の変更や新たな Presentation Event Handler や Container Layouter の作成が Presentation Manager や Layout Manager に依頼して行われる場合がある。

### 4.4. 適応の整合性

#### 部品適応の整合性

全体，単一の2種類の適応を Layout Manager が管理することによって整合性を保証する。また既存の部品を前述の3つのカテゴリーに分類し，適応を行うために開発時においては直接UI部品を用いずにこれらの方式を選択して（デフォルトの部品を指定することは可能）コーディングを行なうことによって同一の操作となることを保証する。

#### プロセス適応の整合性

イベントに対するシステムの振る舞いを変化させることに関しては， Presentation Event Handler 単位で管理しており，適応がメソッドの置換によって行われるため保証される。

新たなダイアログが追加されるなどのプロセスの変化については， Dialog Request Handler の範疇だけではなく， Presentation Event Handler ， Container Layouter の構成状態を含めて変化するので，現時点では予め設計された対話への変更しか保証できない。

## 5. 考察及び今後の課題

本メカニズムで適応を実現することが出来るが，そのプログラミングは本構造に則った形で行う必要がある。そのためその構成はかなり複雑なものとなる。そこで，デザインパターンやフレームワークの提供によって適応型ユーザインタフェースを持つアプリケーションの作成を支援する必要がある。

自動適応を実現するために適応エンジンやセンサーの埋め込みをサポートしていく必要がある。

### 参考文献

[1] 長崎，東 「適応型ユーザインタフェースを実現するためのシステムアーキテクチャ」 情報処理学会，第52回全国大会講演論文集，pp.5-181-182

[2] Shneiderman, Ben, *Designing the User Interface Second Edition*, Addison-Wesley Publishing Company, 1992 (邦訳：東ら監訳 ユーザインタフェースの設計 第2版，日経BP社，1993)

[3] 野中，木村，長崎，東「適応型システムのためのアーキテクチャと支援サービス」 情報処理学会，第54回全国大会講演論文集