

ラピッドプロトタイピングツール Muse(1) *

1 R-4

神尾広幸† 雨宮美香† 松浦博† 新田恒雄†
松澤由香里† 山城明宏†

†株式会社 東芝 マルチメディア技術研究所

†株式会社 東芝 研究開発センター

1 はじめに

不特定多数のユーザが利用する社会情報システムの高機能化がはかられるに伴って、ユーザにはますます複雑な操作が要求されるようになってきている。このことから、より使いやすいユーザインタフェース (UI) の開発が求められており、従来の GUI に音声認識、音声合成などを取り入れたマルチモーダルユーザインタフェース (MUI) も、システムの使い勝手を向上させるものとして期待されている [1]。

そこで我々は、MUI の作成と評価・改良を迅速に行うことができるラピッドプロトタイピングツール Muse (Multimodal User-interface design Support Editor) を開発した [2]。Muse は MUI の作成と、作成された MUI の動作シミュレーションを実行するツールであり、複雑になりがちな MUI を GUI 環境で構築する。

本稿では、Muse の概要を説明したのち、マルチモーダル入出力の取り扱い方法について詳述する。

2 Muse の概要

Muse では MUI の設計および実行という 2 つのモードが存在する。設計モードでは、画面の設計と動作シナリオの設定を行う。画面の設計は図 1 のように画面 (カード) 上に部品 (UI-object) を配置することで行う。Muse には表 1 に示す UI-object が用意されている。一方、動作シナリオはカード

上に配置された UI-object から他のカードや UI-object にリンクを張ることによって設定する。リンクではリンク先とリンク先に送信するメッセージを保持する。実行モードでは、マウスやタッチパネルを用いて UI-object をクリックすると、その UI-object は設定されたリンク先にメッセージを送信する。メッセージを受けたカードや UI-object は、メッセージに従った動作を行う。

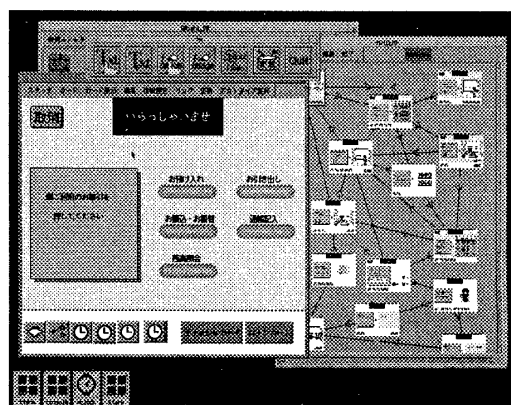


図 1: Muse による画面の設計

表 1: Muse で用意されている UI-object

画面を構成する UI-object	イメージ部品 テキスト部品 アニメーション部品
音声を出力する UI-object	サウンド部品 規則合成部品
マルチモーダル入力を行う UI-object	音声認識部品 文字認識部品
制御を行う UI-object	タイマ部品 条件分岐部品等

*Rapid Prototyping Tool Muse(1)
Hiroyuki Kamio, Mika Amamiya, Hiroshi Matsuura,
Tsuneo Nitta, Yukari Matsuzawa, Akihiro Yamashiro

†Multimedia Eng. Lab., Toshiba Corp.

‡Research&Development Center, Toshiba Corp.

3 Muse におけるマルチモーダル入出力の取り扱い

Muse では、すべての UI-object を同等に取り扱うことを目的としてオブジェクト指向開発を行った [3]。すなわち、すべての UI-object をひとつの抽象クラスから継承して作成することで、音声や画像という異なるモダリティを扱う UI-object でも、Muse 内では同じクラスのインスタンスとして同等に取り扱っている。

例えば、出力を担当する UI-object には画面を構成するものと音声出力を行うものがあるが、音声という不可視なモダリティを可視化させるために UI-object を表すアイコンを用意し、カード下部にそれを配置する領域を設けた。また図 2 のように音声の属性もダイアログで設定できるようにしている。このため、設計モードでは、UI-object をすべて GUI で操作することができる。また、実行モードでは、すべての UI-object はリンク元からメッセージを受け取ることで各々の動作を行う。例えばイメージ部品やテキスト部品は表示/非表示という動作をメッセージに従って行うが、規則合成部品は同じメッセージで音声出力/停止という動作を行う。

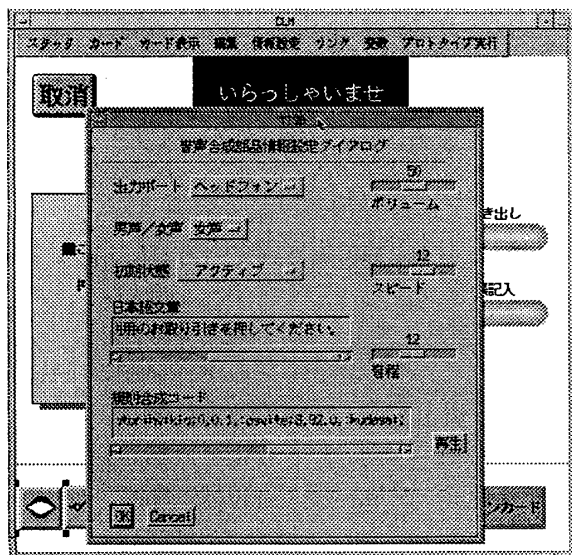


図 2: 規則合成部品の属性設定ダイアログ

入力を取り扱う UI-object の場合、イメージ部品など画面に表示される UI-object はユーザからマウスやタッチパネルでクリックされることをきっかけにしてリンク先にメッセージを送出する。一

方、音声認識部品や文字認識部品はそれぞれの認識結果を獲得したことをきっかけにして、リンク先にメッセージを送出する。このように UI-object によって異なったイベントが入力されるが、各 UI-object でイベントを受信してからはリンクに従って動作が行われるので、リンクの設定方法はすべての UI-object で統一されている。

また、音声認識部品などの認識結果を格納する手段として、Muse では変数オブジェクトが用意されている。図 3 は音声認識部品の処理を表したものである。音声認識部品は音声認識装置から認識結果を受け取ると変数オブジェクトに認識結果を格納して、条件分岐部品にメッセージを送出する。条件分岐部品は、条件とリンク先を格納したテーブルと認識結果を比較して、対応するリンク先にメッセージを送出し、認識結果に応じた動作を起こす。

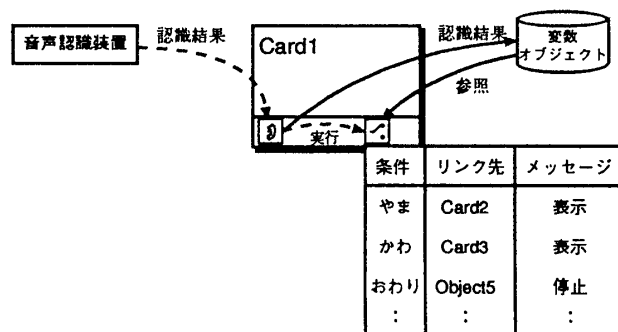


図 3: 音声認識部品の処理

4 おわりに

マルチモーダルインタフェースの評価・改良を容易に行うラピッドプロトタイプングツール Muse を開発した。本ツールによって、従来複雑な制御を必要としたマルチモーダル入出力機能を、容易に付加することが可能となった。今後は Muse を様々な社会情報システム開発に適用していく。

参考文献

- [1] 新田, “GUI からマルチモーダル UI (MUI) に向けて”, 情処誌, vol36, No.11 (1995-11).
- [2] 神尾他, “社会情報システムのためのラピッドプロトタイプングツール Muse の開発”, 信学技報, NLC95-50, SP95-85 (1995-12).
- [3] 松澤他, “ラピッドプロトタイプングツール Muse (2)-MVC モデルを利用したオブジェクト指向開発”, 情処全大 (1996-3).