

F-8

拡張性を備えた対話システム開発支援ツールの構築 Expandable Design Tool for Spoken Dialogue System

伊藤 和明† 山口 由紀子‡ 河口 信夫‡ 松原 茂樹‡ 稲垣 康善†

Kazuaki Ito Yukiko Yamaguchi Nobuo Kawaguchi Shigeki Matsubara Yasuyoshi Inagaki

1. はじめに

電話音声認識技術の向上に伴い、電話を介した音声対話システムが普及し、メーカーのコールセンターや、様々な情報を提供する音声ポータルサイトなどに利用されている。これらの CTI サービスを構築するための商用環境が様々な企業から提供されているが、対話システム研究の促進のために汎用的に使用可能なオープンな対話システム開発環境を提供することが望まれる。

本論文では、電話を利用する音声対話システムを GUI 上で設計でき、システム開発者が必要に応じて自由に機能を拡張できる、開発支援ツールの構築について述べる。本開発支援ツールでは、音声対話システムの基本コンポーネントをブロックとして表現し、対話システム設計者がそれらをキャンパス上で接続していくことによって対話システムを設計できる。また、対話システム設計者がスクリプトを記述することにより、機能を追加することができる。さらに、各コンポーネントをオブジェクト指向におけるクラスとして実現しているため、クラスの継承を利用することによって効率的な機能拡張が可能である。本ツール上で開発した対話システムは、実際に電話を介した動作が可能である。

2. 拡張性の高い開発支援環境

電話を用いた対話システムの評価実験や音声認識装置のテストなどの研究を目的とした、汎用的に利用可能な対話システムの開発支援環境には、以下のような要素を備えていることが求められる。

- 対話システム開発に集中できるように、一般的な対話システムの基本機能を全て備えること
- 直感的にわかりやすいインタフェースを備え、対話システムの設計段階から、動作、メンテナンスまでをサポートすること
- 必要に応じた機能の入れ替え・追加・変更を効率よく行うことができ、汎用的に利用できること
- シンプルでわかりやすい内部設計であること

対話システムの開発支援環境として、これまでに LOTOS[1]、CSLU Toolkit[2]などが構築されている。これらはいずれも、GUI 上で音声認識、音声合成などの基本要素を組み立てることによって、対話システムを設計できるツールである。しかし、前者は市内案内システムという特定の対話システム開発のために作成されたものであり、汎用性を備えていない。また、後者は広く公開されており、研究目的にも利用できるものの、既存の機能を変更することや効率よく拡張することは難しい。

†名古屋大学大学院工学研究科

‡名古屋大学情報連携基盤センター

E-mail:kazu_ito@inagaki.nuie.nagoya-u.ac.jp

そこで我々は、対話システム開発者が効率よく機能を追加でき、対話システム研究に汎用的に利用できる設計支援ツールの開発を行った。

3. オブジェクト指向に基づく開発支援ツール

3.1 開発支援ツールの概要

本開発支援ツールは、設計、動作、変更・改良など、対話システムの開発サイクルをサポートする。音声認識・音声合成などの対話システムの一般的な機能をコンポーネントと呼ぶ基本単位に分割し、コンポーネントのインスタンスをブロックとして GUI キャンパス上に表示する。ブロックをキャンパス上でマウス操作により接続することにより、対話システムの処理の流れを設計できる。このことにより、それぞれのブロックの機能を明確化でき、ブロックの入れ替えなどによる変更が容易になる。また、処理の流れを可視化することにより、全体の構成を把握しやすくなる。また、本ツールは機能の拡張性の高さに重点を置いて構築しており、コンポーネントをオブジェクト指向におけるクラスとして表現し、ブロックをそのクラスのオブジェクトとして実現した。また、シンプルな実行方式を採用することによって、柔軟な機能の変更や追加が可能となっている。

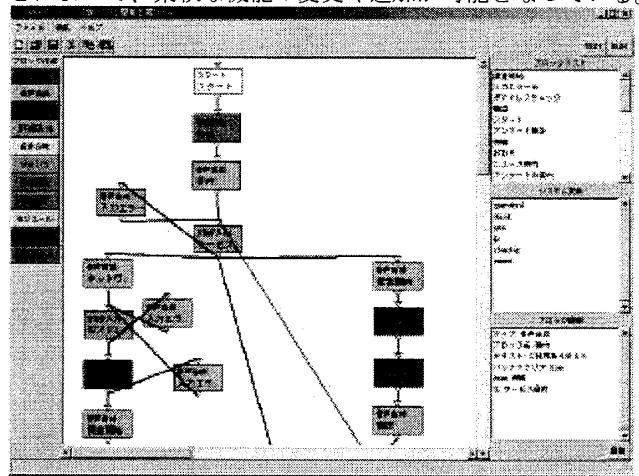


図1. 開発支援ツール

3.2 開発支援ツールの実装

開発支援ツールは Windows2000 上でオブジェクト指向スクリプト言語 Ruby[3]を用いて実装した。本設計支援ツールは Ruby インタプリタ上で動作し、Dialogic 社製電話音声ボード D/4PCI によって、実際に電話を介した動作が可能である。

図1に開発支援ツールのインタフェースを示す。対話システム開発者が、左のコンポーネントリストから必要な機能を選択すると、そのインスタンスがキャンパス上にブロックとして表示される。キャンパス上に表示されたブロック

間を、マウス操作によって接続することによって、対話システムの処理の流れを設計する。本ツールでは、一般的な電話を用いた対話システムの開発に必要な機能を分割し、以下にあげる9つのコンポーネントとして構成した。

- ◇ 音声再生：PCM 音声ファイルを電話で再生する。
- ◇ 音声合成：音声合成エンジンによって日本語のテキストを PCM 音声ファイルに変換して再生する。
- ◇ 音声認識：システム利用者が電話で話した発話を音声認識エンジンを通して日本語文字列に変換する。(現在は録音機能のみ)
- ◇ DTMF 入力：利用者が行った DTMF 入力を文字列として変数に格納する。
- ◇ 条件分岐：複数ブロックへの分岐条件を指定する。
- ◇ ジャンプ制御：離れたブロックなどに制御を移す。
- ◇ サブモジュール呼び出し：別個に保存した対話システム情報ファイルをモジュールとして呼び出す。
- ◇ 電話待機：電話がかかってくるまで待機し、電話ボードで電話を検知したら次の処理へ移行する。
- ◇ 電話切断：電話を切断する処理を行う。

ブロックには、コンポーネントごとの設定事項が用意されており、開発者がカスタマイズすることが可能である。

3.3 内部構造とオブジェクト指向に基づく機能拡張

本開発ツールにおけるコンポーネントは、オブジェクト指向におけるクラスとして表現され、コンポーネントのインスタンスであるブロックはオブジェクトとして実現される。それぞれのブロックは、以下の3つの機能を持つオブジェクトから構成されている。

- 情報オブジェクト：ブロックの名称、コンポーネントタイプ、ブロックごとの設定、次ブロック・遷移条件対応リストなどを保持
- GUI 情報オブジェクト：GUI キャンバス上での位置、色などの表示情報を保持
- 動作定義オブジェクト：ブロックの動作定義をメソッドとして保持

このうち、GUI 情報オブジェクトは全てのコンポーネントに共通するクラスのインスタンスであり、他の2つのオブジェクトは、共通機能定義のためのクラスを継承した、コンポーネント固有クラスのインスタンスである。

実行の際には、処理スレッドがブロックの動作定義オブジェクトの実行メソッドを実行し、情報オブジェクトにある次ブロック情報に従い、ブロックをたどることにより全体の処理が実行される。この様子を図2に示す。

本開発支援ツールでは、前述した基本コンポーネント以外に、機能拡張のための機構が2つのコンポーネントとして用意されている。

- ◇ スクリプト記述：基本コンポーネントで記述されていない処理を、システム開発者が Ruby スクリプトを記述することによって追加できる
- ◇ クラス定義：動作定義オブジェクトのクラスを新たに定義することにより、新たなコンポーネントを作成することができる

クラス定義コンポーネントを用いて、既存のコンポーネントの動作定義クラスを継承するクラスを作成することによって、効率よくコンポーネントを定義できる。例として、音声認識コンポーネントを拡張した場合のクラス関係を図3に示す。

また、Ruby では、C 言語によりライブラリを作成することが可能であり、機能を柔軟に拡張することができる。

4. まとめ

本論文では、対話システムを GUI 上で設計できる対話システム開発支援ツールについて述べた。本ツールでは、オブジェクト指向に基づくコンポーネントの実現と、シンプルな実行方式の採用により、必要な機能を開発者が効率よく柔軟に拡張・追加することが可能である。

現在、開発支援ツールに音声認識機能を実装している。また、本開発支援ツールを用いた対話システム研究として、ネットワーク状況自動応答システムを開発し、その運用テストの実施をしている。

参考文献

- [1] Tomas Nouza, Jan Nouza: Graphic platform for designing and developing practical voice interaction systems, Eurospeech 2001, pp.1287-1290 (2001)
- [2] <http://cslu.cse.ogi.edu/toolkit/>
- [3] <http://www.ruby-lang.org/>

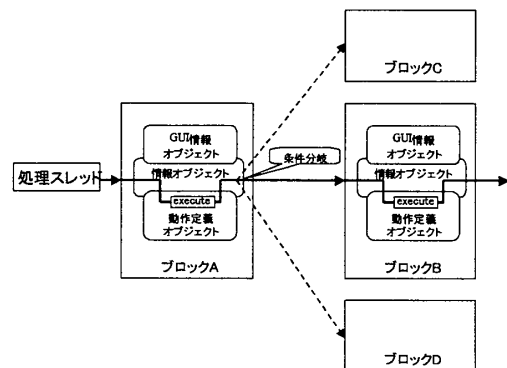


図2. ツール内部での動作の実現

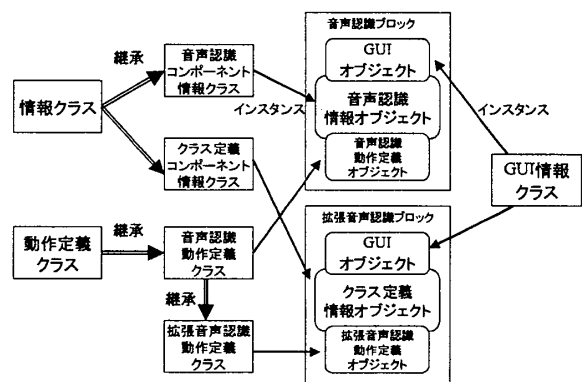


図3. 継承を用いたクラス定義