

## 異なる言語を用いる設計チームのための 要求仕様定義支援システム

脇屋 達<sup>†</sup> Claude Moulin<sup>‡</sup> Laurent Wouters<sup>††</sup> 真部 雄介<sup>†††</sup> 藤田 茂<sup>†††</sup> 菅原 研次<sup>†††</sup>

千葉工業大学大学院 情報科学専攻<sup>†</sup>

Compiègne University of technology Heudiasyc CNRS UMR 6599<sup>‡</sup>

Compiègne University of technology Génie informatique<sup>††</sup>

千葉工業大学 情報科学部<sup>†††</sup>

### 1. はじめに

近年、異なる言語を使用する設計者を含むチームによってシステム開発を行う必要性が増加している。そのため、これまでの協調設計支援機能に多言語翻訳機能を加えた異文化コラボレーション支援システムの研究が進められている[1]。本システムは異なる言語を用いるチームが母国語で協働開発を行うために、UMLを用いて要求仕様を定義する作業を支援するシステムを開発する。更に、チャット機能と翻訳機能を用いることによってリアルタイムに質問や回答、説明を行いながら協働作業を行っていくことができる。

### 2. 多言語による要求仕様定義支援システム

設計や分析、要求定義を行う時には、高いチームワークおよびコミュニケーション能力が必要となる。しかし、異なる言語を用いるメンバーから構成されたチームでは、言語や文化の壁があり、コミュニケーションをとることも難しい[2]。そのため、多言語の協働設計システムは異なる言語を用いるチームのコミュニケーションやチーム間で理解を深めるためにも必要となる。本システムでは多言語のチームで協働作業を行えるように、それぞれ母国語を使用して作業を行えるようにする。また、協働設計作業を非同期作業と同期作業の 2 種類に分けてそれぞれの支援を行う。

### 3. 非同期作業と同期作業

仕様書の開発は、新しいドキュメントを作成し、更新、他のチームが前のチームの設計書を理解してからまた新しいドキュメントを開発す

るといった作業をシステムが完成するまで交互に行う。作成するファイルのうち、少なくとも 1 つはプロジェクトの UML を記述するために使用される。また、自然言語で書かれたプロジェクトの状態を記したドキュメントもある。

設計作業中、チームは同時に異なった場所で作業を行う可能性も、違う時間に他のチームが作成したドキュメントに取り組む可能性もある。このことから図 1 に示すように設計作業を非同期作業と同期作業の 2 つに分けることができる。

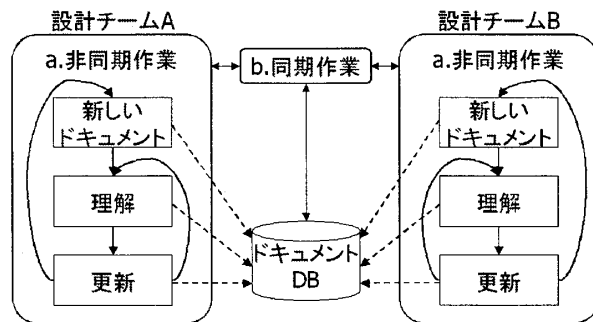


図 1. 作業の手順

#### a. 非同期作業

非同期作業とは、それぞれの設計チームが独立して行う作業である。例えば、日本のチームとフランスのチームで協働作業を行う場合、時差があるため同期作業を行うことが難しい。そのような場合、非同期で開発を行う必要がある。

#### b. 同期作業

同期作業とは、チームがリアルタイムに質問や回答、説明、意見の交換などを行う作業である。非同期作業と同様にファイル交換の作業も行われる。これらの作業を行うことによりUML表現で設計されている仕様に関する補足説明なども行うことができる。

### 4. 多言語要求仕様定義支援システムの設計

本システムは大きく分けて 2 つの機能を持つ。1 つは翻訳を行う機能、もう 1 つはユーザイン

Requirements Specification Definition Support System for Design Team using a Different Language

<sup>†</sup>Tatsuru WAKIYA · Graduate School of Information Science, Chiba Institute of Technology

<sup>†††</sup>Yusuke MANABE and Shigeru FUJITA and Kenji SUGAWARA · Faculty of Information and Computer Science, Chiba Institute of Technology

ターフェースと連携する機能である。また、本システムは統合開発環境 Eclipse 上で実装した。実装言語は Java である。

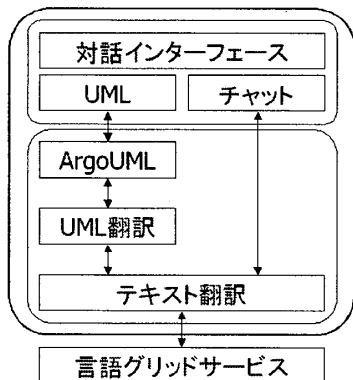


図2. 要求仕様定義支援システム

翻訳機能としてテキスト翻訳を行う。テキスト翻訳では翻訳精度を改善するために、必要に応じてユーザ辞書を使用する。テキスト翻訳インターフェースはユーザが ArgoUML[3]ファイルを含むファイルを翻訳するために用いる。ArgoUML インターフェースは zargo（圧縮した ArgoUML のプロジェクト）ファイルの管理機能を提供する。これは操作をキャッシュすることによって ArgoUML データにアクセスを許可する。例えば、解凍と圧縮はデータにアクセスする命令である。対話インターフェースはプラグイン用のユーザインターフェースを含んでいる。チャットインターフェースは異なる言語を用いるチーム同士でプロジェクトについての議論を可能にするインターフェースである。チャット画面は図3のようないくつかのフィールドと2個のボタンからなる。

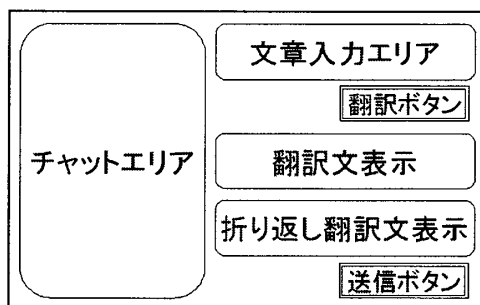


図3. チャット画面

左のチャットエリアで名前や対話の履歴を表示して会話をを行う。ユーザは文章入力エリアに文を入力し、翻訳ボタンを押すことで翻訳を行う。翻訳された文と同様に折り返し翻訳された文も表示される。その結果、ユーザは翻訳され

た文の精度を確認することができる。この行為を行うことで相手のチームに誤訳された文章を送信することを防ぐ。右下の送信ボタンを押すことにより翻訳された文を相手のチームに送信する。

## 5. 言語グリッド

翻訳サービスには京都大学と NICT で提供されている言語グリッドプロジェクトのサービスを用いた。このサービスには機械翻訳や折り返し翻訳などがある。折り返し翻訳とは原文から翻訳対象の言語への翻訳、および翻訳された言語から原文の言語への翻訳を行う。これは通常の翻訳のフィードバックをとることができるので翻訳の精度を確認することができる。折り返し翻訳が原文とまったく違った結果を表示した場合、翻訳結果に不備があるということが分かる。従って、異なる表現で文章を明確に記述し直すか、文章を単純化しなければならないということが分かる。

## 6. おわりに

本稿では、異なる言語を用いるチームが母国語で協働開発を行うことを支援するシステムの開発について報告した。

協調作業で作成された UML モデルはユースケース図やクラス図などを用いたものになる。ユースケース図など開発の間に作成されたドキュメントやチャットを利用した質疑応答を日本語と英語の間で行うことができた。

翻訳の結果、十分な結果を得るためには両チームが厳密なガイドラインに従う必要があるということが確認できた。例えば、文は短くしなくてはならないということや、折り返し翻訳で翻訳文を確認しなければならないというものである。これらの制約はユースケース図の場合だけでなく、チャットシステムなどの言語翻訳にも関係するということが分かった。

## 参考文献

- [1] 杉山香織, 船越要, 神田智子, 藤代祥之, 石田亨, “セマンティックアノテーションに基づく多言語コラボレーション支援,” 情報処理, Vol.2005, No.24, pp.157-162, Mar.2005.
- [2] 林田尚子, 石田亨, “日本-中国共同ソフトウェア開発の観察—異文化コラボレーション支援に向けて—,” 情報処理, Vol.2005, No.49, pp.25-30, May.2005.
- [3] <http://argouml.tigris.org/>, [argouml.tigris.org](http://argouml.tigris.org)