

SCEにおける検証結果のサービス仕様への 逆変換法の検討

3N-8

岡本光浩* 藤田浩** 新津善弘*

*NTTネットワークサービスシステム研究所

**北陸先端科学技術大学院大学

1. まえがき

NTTでは、高度インテリジェントネットワーク（高度IN）における、新しい電話サービスに必要な各種サービスソフトウェアの効率的な開発を目的とした、サービス生成環境（SCE: Service Creation Environment）を開発中である[1]。SCEでは、サービス仕様修正の容易性、およびサービス仕様とサービスソフトウェアとの間の整合性維持の点から、開発の下流工程で検出されたサービスソフトウェアレベルのエラーも、最上流工程である図形式のサービス仕様レベルで修正可能なことが必要である。このため、エラー情報を図形式のサービス仕様へ反映させるための支援方法が必要とされる。

本報告では、サービスソフトウェアレベルのエラー情報を上位レベルのサービス仕様へ反映させるための逆変換の考え方と、その一部機能に関するプロトタイプ作成の結果について述べる。

2. SCEでのサービス開発

SCEでのサービス開発[2]（図1）では、サービスソフトウェア自動生成ツール[1]（図2）に入力されたユーザ・網情報シーケンス図（MSDL）や、リソ-

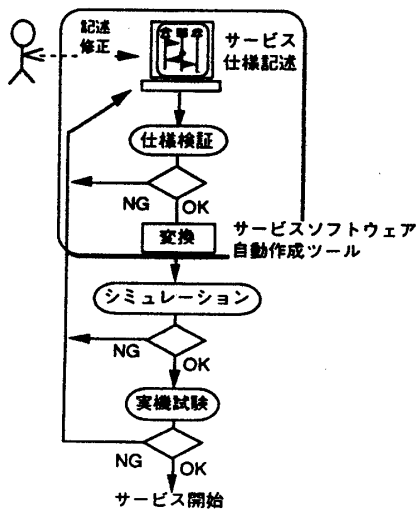


図1 SCEでのサービス開発作業の流れ

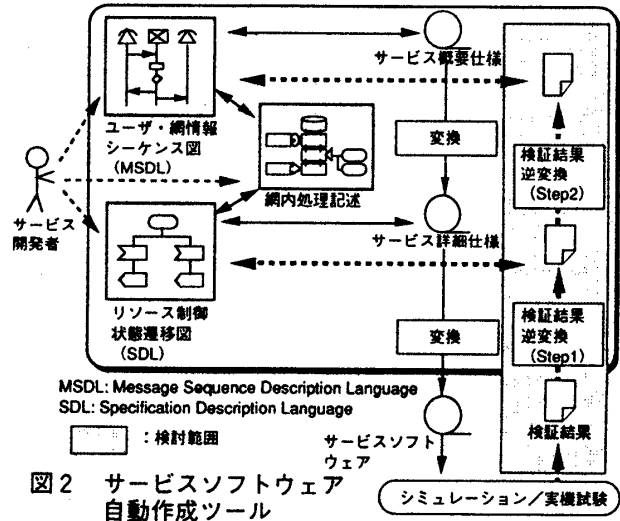


図2 サービスソフトウェア自動作成ツール

ス制御状態遷移図（SDL）を、サービスソフトウェアに変換し（以下、順変換と呼ぶ）、その後シミュレーション/実機試験で検証する。サービスソフトウェアレベルで検出されたエラーはSDLに、SDLでのエラーはMSDLに反映され、修正される。

3. 検証結果逆変換法

本検討では、検証結果逆変換を「サービスソフトウェアまたはサービス仕様レベルで検出されたエラーの情報をもとに、上位レベルのサービス仕様記述の修正に必要な情報を作成すること」と定義する。また、サービスソフトウェアレベルからSDLレベルへの逆変換をStep1、SDLレベルからMSDLレベルへの逆変換をStep2と呼ぶ（図2）。

3.1 逆変換に必要な情報

検証結果として得られるエラーは発生位置と内容で特定できるため、逆変換ではこれらに関する情報を扱う。

(1) エラー発生位置の情報

- ・ 下位レベルの情報：エラーの発生位置
- ・ 上位レベルの情報：修正すべき位置
- ・ 逆変換処理：上下レベル間の位置の対応づけ

(2) エラー内容の情報

- ・ 下位レベルの情報：エラーの現象や原因
- ・ 上位レベルの情報：エラー内容と修正方法
- ・ 逆変換処理：下位レベルの情報から上位レベルでの修正方法を判断

A Reverse Conversion Method of Service Software Verification Results

Mitsuhiro Okamoto and Yoshihiro Niitsu

*NTT Network Service Systems Laboratories

Hiroshi Fujita

**Japan Advanced Institute of Science and Technology

また、逆変換時には下位レベルからの情報に加えて、以下のものが必要である。

- (3) 順変換履歴：順変換時の上下レベル間行番号の対応。順変換時に作成。
- (4) 順変換知識：順変換の規則となるパラメータ間や記述要素セット間の対応。
- (5) 逆変換知識：検証結果と上位レベルでの表示情報との対応。

3.2 逆変換処理

3.2.1 エラー発生位置の逆変換法

逆変換では、下位レベルの情報である発生位置（具体的には行番号）をもとに、順変換知識と順変換履歴を参照し、上位レベルにおける修正位置を決定する。MSDLからSDL、またはSDLからサービスソフトウェアへの順変換では、記述要素間の1対1の変換ではなく、記述要素のあるまとまり（以下、変換記述要素セット）を単位とした変換方法をとっている[3]。このため上位レベルでは複数の記述要素が修正対象になる場合もある。

3.2.2 エラー内容の逆変換法

検証の種類により扱う内容が異なる。

- (1) **構文検証**（サービス仕様記述の文法誤りを検出）
→ 文法誤りは、サービス仕様の記述時に検出するので、逆変換の対象とはならない。
- (2) **論理検証**（網で許容されていない状態遷移の定義など、サービスの動作規定に反するものを検出）
→ SDLの検証では、誤り検出の基準となる検証知識が存在するので、この検証知識の条件を満たさなかった項目をエラー情報として出力する。シミュレーション/実機試験では、プログラムの動作が停止した時点での状態とプログラムからのログファイルへの出力を検証結果とする。逆変換では、出力されたエラー情報をもとに、順変換および逆変換知識を参照し、上位レベルでの表示内容を決定する。
- (3) **意味検証**（要求と動作との不一致を検出）
- (4) **操作性検証**（サービスの使い勝手の悪さを検出）
→ これらの検証では、誤りは作業によって判断され、何が誤りであるかを形式的なエラー情報として出力するのは、現状では困難である。しかし、ある程度の修正のための指示（手順の変更/追加/削除）や、誤り原因となったパラメータ名の指定などは、何らかのインタラクティブな方法によってエラー情報を作成できると考えられる。
- (5) **性能検証**（規定以上の処理時間を要する箇所を検出）
→ サービスソフトウェアレベルでの関数ライブラリの修正になるため、逆変換の対象としない。

4. 検証結果逆変換Step1機能の設計

3.2.2の検証のうち、検証工程での占める割合が高く、逆変換の効果が期待できる、論理検証結果についての逆変換Step1機能を以下のように検討した。

[エラー発生位置] シミュレータは検証結果として、エラーが発生したプログラム行番号を出力する。逆変換機能は、この行番号と順変換履歴である行番号対応表をもとに、SDLの対応位置を決定する。

[エラー内容] シミュレータは、以下の項目を検証結果として出力する。

- ・終了コード（正常/異常/中断）
- ・関数の戻り値
- ・エラー発生時のパラメータ値
- ・エラーと判断されたパラメータ名

逆変換機能は、上記情報、順変換知識、および逆変換知識を参照し、SDLに以下の情報を出力する。

- ・命令実行結果
- ・エラーの内容
- ・修正すべきパラメータ名

5. プロトタイプの作成

前章の検討にもとづき、逆変換Step1機能のプロトタイプシステムを作成した（図3）。逆変換知識等は今後の検討であるが、従来人手に頼る作業であった、修正すべき位置やパラメータ名の判定においては、本システムの有効性を確認できた。

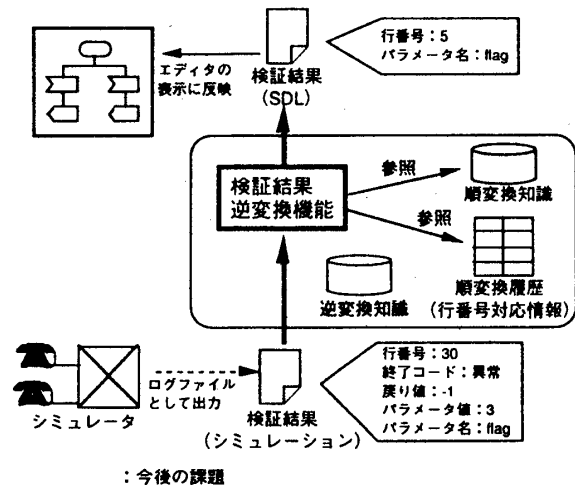


図3 プロトタイプシステムの構成

6. むすび

SCEにおいて、検証結果を上位レベルのサービス仕様へ反映するための、エラーの発生位置と内容に関する情報を用いた逆変換方法と、プロトタイプの作成結果について述べた。今後の課題を以下に示す。

- ・Step2の逆変換法
- ・意味検証、操作性検証の検証結果逆変換法
- ・逆変換知識の構成法と上位レベルでの誤り表示方法

文献

- [1]新津、岡本、高見：“高度INにおける統合サービス生成環境構成法”、信学技報、SSE94-173(1995-1)
- [2]岡本、新津：“高度INサービス生成環境におけるサービスソフトウェア検証法”、信学技報、SSE94-85(1994-7)
- [3]新津：“サービスソフトウェア自動作成における検証結果逆変換法の検討”、1993信学会春季大会、B-531