

モデル駆動アーキテクチャに基づく ホームネットワークサービスの開発支援

坪井 正徳[†], 沢田 篤史[‡], 河原 達也[†]

[†] 京都大学 情報学研究科
〒 606-8501 京都市左京区吉田二本松町
[‡] 南山大学 数理情報学部

本稿では、ホームネットワークサービスの開発支援を目的としたモデル駆動アーキテクチャに基づくツール基盤を提案する。提案するツール基盤では、サービスの記述形式からメタモデルを定義し、定義したメタモデルによりデータベーススキーマを生成する。そして、変換規則の定義により変換規則に応じたモデル変換をデータベース上で行う。提案するツール基盤の有効性を示すために、情報通信研究機構で開発された「ゆかりカーネル」と呼ばれるホームネットワーク基盤への適用を行って変換ツールを作成し、シミュレーション環境での動作確認を行って有効性を示す。

キーワード：ネットワーク家電，ホームネットワーク，モデル駆動アーキテクチャ，開発支援

Supporting Service Development of Home Network Service Based on Model-driven Architecture

Masanori TSUBOI[†], Atsushi SAWADA[‡], Tatsuya KAWAHARA[†]

[†] Graduate School of Informatics, Kyoto University
Yoshida-nihonmatsu, Sakyo, Kyoto 606-8501, Japan

[‡] Mathematical Sciences and Information Engineering, Nanzan University

In this paper, we propose a supporting platform based on model-driven architecture for service development of networked home appliances. In this platform, we define database schemata from meta models which have made based on the syntax and semantics of description. Model elements are stored to a database and model transformation is handled on the database.

In “UKARI-Kernel” developed at NICT, we support service development by using this proposed platform. We confirm that the generated scenario model is valid and executable in UKARI-Kernel.

Keywords: Networked Appliance, Home Network, Model-driven Architecture, Supporting Development

1 はじめに

近年、ユビキタス社会への移行により、ホームネットワークの普及が進んでいる。ホームネットワークの環境を利用することにより、ネットワーク家電同士を連携させたサービスの利用が可能となる。このようなサービスは、一般的にあらかじめ作成されたサービスアプリケーションをユーザが実行することにより実現される。サービスの開発には、ネットワーク家電の組み合わせとその複雑な動作手順や状態遷移をプログラム等の抽象度の低い形式で記述する必要がある。また、抽象度の低いプログラムなどの形式で記述されたサービス論理では仕様変更への追従や保守などが困難となっていることも共通の課題であり、分析、設計、実装、保守を统一的に支援する環境が求められている。

本稿では、モデル駆動アーキテクチャというソフトウェア開発手法の概念をホームネットワークにおけるサービス開発の支援を目的として適用し、モデル変換ツール基盤を提案する。サービスの記述形式から定義したメタモデルに基づいてデータベーススキーマを生成し、データベース上に記述要素を格納してデータベース上でモデル変換を行う。ツール基盤の利用により、サービス開発の部分的な自動作成が可能となり、データベース上で記述要素を扱うことでサービス仕様変更への追従や保守の負担が軽減する。また、本稿で提案するツール基盤を情報通信研究機構において開発されたゆかりカーネルへ適用する。ゆかりカーネルにおけるサービス開発ではシナリオ記述の作成が開発者にとって負担となっており、サービス仕様変更への追従や保守などが困難となっている。シナリオ記述の基本構造がゆかりカーネルの仕様である通信オペレーションの指定により決定されることを確認し、通信オペレーションを付与したトポロジ記述を開発者が作成することでシナリオ記述を自動生成するツールの実装を行い、生成されたシナリオ記述がゆかりカーネル上で動作することを確認し、有効性を示す。

また、2つのシナリオ記述を同期箇所を指定して1つのシナリオ記述として合成することを考え、同期箇所についての考察と合成についての考察を行う。

2 背景

本節では、ホームネットワークの現状、ソフトウェア開発の現状についても述べ、本稿の位置づけについて述べる。そして本稿で扱うゆかりカーネルの概要を述べ、ゆかりカーネルにおいてサービスアプリケーションを開発する際に問題となっている点について述べる。

2.1 ホームネットワークの現状

近年、さまざまなホームネットワークシステムが研究開発され、実際に実用化されている。ホームネットワークを普及させるための問題点として規格の統一が挙げられ、現在、AV機器を対象としたHAVI¹、ソフトウェアシステム、家電機器などを対象としたJini²、パソコン関連のほか白物家電を対象としたUPnP³、白物家電を対象としたECHONET⁴といった規格が提唱されている。だが、これらの様々な分野の機器にまたがる統一的な規格は提案されていない。また、同じ分野の機器であっても各メーカーごとに採用する規格が違う。

例えば、ECHONETに準拠した家電製品としては、松下電器産業株式会社の「くらしネット」⁵、株式会社東芝の「FEMINITY」⁶等が発売されているが、松下電工株式会社の「エミット・ホームシステム」⁷では、エミットホームプロトコルという独自のプロトコルを提案している。機器間やメーカー間における互換性はほとんど考慮されていない。よって、ユーザが購入したネットワーク家電がそれぞれ異なるプロトコルを実装していた場合、それらと同じネットワークで利用することは困難となる。

NTTサイバーソリューション研究所では、OSGi (Open Services Gateway Initiative) の仕様を利用した「ホームサービスハーモニー」の開発を行い、異なるプロトコルで実装されたネットワーク家電間の接続を可能とした[1]。また、家電利用者の認証、通信データの保護、ネッ

¹<http://www.havi.org/>

²<http://jp.sun.com/jini/>

³<http://www.upnp.org/>

⁴<http://www.echonet.gr.jp/>

⁵<http://national.jp/appliance/product/kurashinet/>

⁶<http://feminity.toshiba.co.jp/feminity/>

⁷<http://www.mew.co.jp/emit/>

トワーク内にある家電の把握，異なる制御プロトコルを持つ家電へのアクセスといったホームネットワーク環境構築の問題点は，グリッドを構築する際の問題点でもあるということに着目した[2]の研究では，Globus Alliance⁸が開発を行っている Globus Toolkit を利用して家電を制御する「グリッド家電」の提案を行っている。

現在のホームネットワークを扱う研究としては，異なる制御プロトコルを統一して扱う部分に焦点が当てられ，その解決のためのミドルウェア提案に関するものが多数である。だが，最近では[3]に見られるようにホームネットワークにおけるサービスに焦点を当てた研究も行われてきていることから，ホームネットワークの普及が進み，ホームネットワーク環境を利用したサービスの多様化，複雑化が進んでくると，サービスアプリケーションの開発支援に研究の焦点が移行してくるものと考えられる。本稿では，このようなホームネットワークのサービスアプリケーションの開発支援に焦点を当てる。

2.2 モデル駆動アーキテクチャに基づくソフトウェア開発

近年，UML (Unified Modeling Language) [4] の普及が進み，モデルを中心としたソフトウェア開発が広く採用されるようになった。このことにより，仕様を自然言語で記述した場合と比較して曖昧性を取り除くことができるようになった。

このような背景のもと，開発の段階で作成した UML の図式からプログラムのコードを自動生成するような枠組みに着目され，モデル駆動アーキテクチャ (MDA: Model-Driven Architecture) [5] という概念の提唱が行われた。この概念は，PIM (Platform Independent Model: プラットフォーム非依存モデル) から PSM (Platform Specific Model: プラットフォーム依存モデル) に変換規則に基づいて自動変換し，更にはプログラムコードの生成まで変換規則に基づいて自動変換を行うというものである。MDA の概念により，分析・設計，開発，保守といった一連のサイクルすべてをモデルを使って実行することが可能となる。

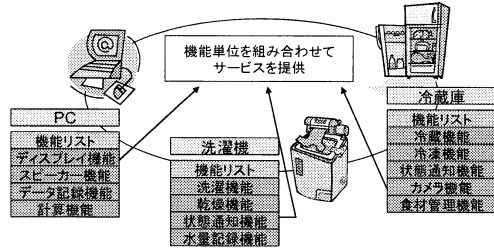


図 1: サービスの構成要素の例

[6], [7] の研究では，UML の図式を記述することにより Java のプログラムコードを自動生成する手法を提案している。[8], [9] では，複雑な認証プラットフォームの理解が必要であったセキュリティの設定を，モデル上でセキュリティを設定することでモデル変換によりセキュリティポリシーを作成する手法を提案することで容易に設定できるようにしている。

MDA の概念を利用した研究の対象は，UML の図式からプログラムコードへの自動生成を扱うものが多数を占めるが，本稿ではホームネットワークにおけるサービスの開発を支援することを目的とした適用を考えている。

2.3 機能協調サービス基盤ゆかりカーネル

本節では，本稿で扱う機能協調サービス基盤ゆかりカーネル [10] についての説明を行う。ゆかりカーネルは情報通信研究機構 (NICT) のゆかりプロジェクト [11] で開発されたホームネットワーク基盤である。ゆかりカーネルの特長は，各種ネットワーク家電の持つ機能を単位に発見，接続して，サービスを提供する点にある。ゆかりカーネルにおいて，機能の単位は FE (Function Element: 機能要素) と呼ぶ。例えば，図 1 では，PC のディスプレイ機能と洗濯機の状態通知機能を組み合わせて洗濯機の状態通知を PC のディスプレイ上で通知するというサービスの構成要素，PC のディスプレイ機能と冷蔵庫のカメラ機能を協調させて冷蔵庫の中を PC の画面上で表示するというサービスの構成要素を示している。

ゆかりカーネルにおいてサービスを実現するには，トポロジ記述とシナリオ記述の 2 種類の

⁸<http://www.globus.org/>

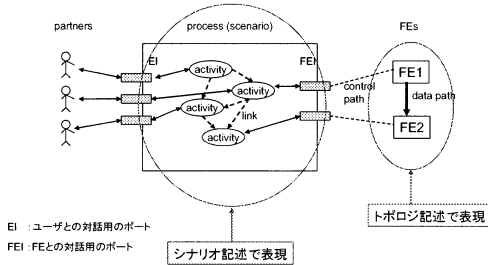


図 2: トポロジ記述とシナリオ記述の記述対象

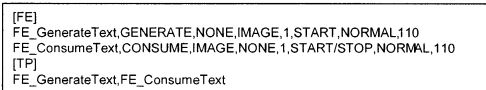


図 3: トポロジ記述の例

記述を用意する必要がある [12].

トポロジ記述とシナリオ記述で表現する対象を図 2 に示す。トポロジ記述は、サービスが利用する FE と利用する FE 間のデータベースの関係を記述したものである。シナリオ記述は、FE の発見手順、ユーザへの応答といった FE の動作手順や状態遷移をアクティビティ (activity) とリンク (link) の関係によって記述したものである。アクティビティには属性として、合流条件 (joinCondition) が定義されており、この合流条件にリンクを複数指定して AND 条件を付与することで、指定したリンクがすべて遷移した後にアクティビティを実行させるという制御が可能となる。

図 2 におけるプロセス (process) が、一つのサービスアプリケーションに対するシナリオに相当する。アクティビティとは、データの入力、出力、FE の発見といった作業を行うものであり、リンクによりそれらの遷移関係を示している。

トポロジ記述の例を図 3 に、シナリオ記述の例を図 4 に示す。シナリオ記述は複雑な状態遷移を表現する必要があり、またサービスの仕様変更への追従や保守が容易ではない。シナリオ記述の作成がゆかりカーネルにおけるサービス開発で負担となっている。

ホームネットワーク開発、ソフトウェア開発のこのような背景をふまえ、本稿ではモデル駆



図 4: シナリオ記述の例

動アーキテクチャの概念をホームネットワークサービスの開発支援を目的として適用することを考え、モデル変換ツール基盤を作成し、ゆかりカーネルへの適用を行う。

3 モデル変換ツール基盤

本節では、ホームネットワークサービスの開発支援を目的として設計したモデル変換ツール基盤について述べる。そして、一例としてゆかりカーネルにおけるサービスの開発支援に、設計したモデル変換ツール基盤を利用することを考えて作成したシナリオ記述作成を支援するツールについて述べる。

3.1 モデル変換ツール基盤の設計

一般的にホームネットワーク環境を利用したサービスアプリケーションを開発する際、開発者はサービスが利用するネットワーク家電の組み合わせとその動作手順、状態遷移の関係を記述することが必要となる。記述方法は各ホームネットワーク基盤により異なるが、ネットワーク家電間の複雑な連携が必要となるサービスを開発する際には特に開発者にとってこれらの記

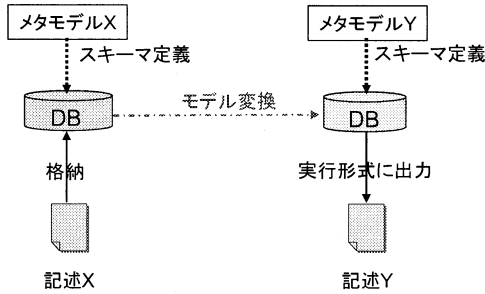


図 5: モデル変換ツール基盤の設計

述が大きな負担となっている。アプリケーションを作成する際の部分的な自動生成、仕様変更や保守の際のサービス記述の容易な変更といった要望がある。

そこで、本稿ではモデル駆動アーキテクチャの概念に基づいて、図 5 で示すようなモデル変換ツール基盤を提案する。記述要素のメタモデルを作成し、それに基づいてデータベーススキーマの定義を行い、記述要素をデータベース上に格納してモデル変換を行う枠組みである。メタモデルは、クラス、属性、関連、継承の 4 つの要素に限定したクラス図で記述することとし、メタモデルからデータベーススキーマを自動定義するツールの作成を行い、そのツールを利用してデータベーススキーマの定義を行う。

このモデル変換ツール基盤を利用することで、開発者がモデル変換する際の変換ルールを作成すればさまざまな応用が可能となる。例えば、セキュリティ機能の追加を行う変換ルールを定義することでセキュリティ追加ツールの作成を行うということが可能となる。

サービス開発の部分的な自動生成、サービス記述の容易な変更による仕様変更や保守が可能となり、ホームネットワークアプリケーション開発者の負担が軽減される。

3.2 トポロジ記述からシナリオ記述への自動生成ツール

ゆかりカーネルにおいて機能協調サービスを開発する際に負担となっているシナリオ記述の作成を支援することを考える。

シナリオ記述を調査すると、シナリオ記述の

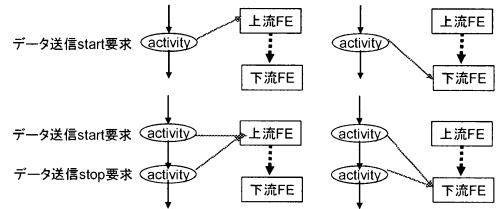


図 6: シナリオ記述の構造パターン

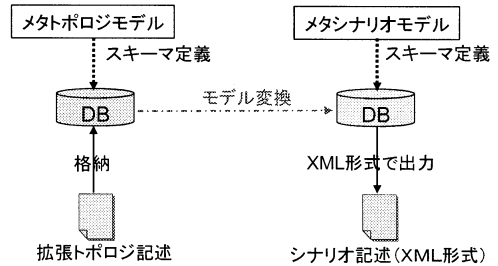


図 7: シナリオ記述の自動生成ツール

基本構造はゆかりカーネルの仕様である 2 つの FE 間のデータ通信の方法を指定する通信オペレーションの指定により決定されることを確認した。通信オペレーションは、上流 FE に指示するタイプと下流 FE に指示するタイプに分類される。また、指定する通信オペレーションの種類により、1 回のアクティビティが必要なタイプと 2 回のアクティビティが必要なタイプに分類される。これらにより、図 6 で示すように 4 種類のシナリオ記述の構造パターンに分類される。指定した通信オペレーションに対応したこれらのシナリオ記述の構造パターンを自動的に選択することによりシナリオ記述の基本構造を決定する方式をとった。このようにシナリオ記述の基本構造を決定した後にシナリオ記述中のトポロジ記述の要素と対応している部分にトポロジ記述の要素を挿入しくことでシナリオ記述を完成させるという方式をとった。

そこで、通信オペレーションを付加情報としてトポロジ記述を作成する（これらを合わせて以後、拡張トポロジ記述と呼ぶ）ことでモデル変換によりシナリオ記述を自動生成するツールの構築を行う。本稿では、拡張トポロジ記述からシナリオ記述を自動生成する際、図 7 で示すような枠組みで設計を行った。

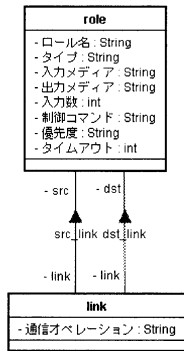


図 8: メタトポロジモデル

モデル変換ツール基盤の適用にあたり、新たに定義しなければならないのは、メタトポロジモデル、メタシナリオモデル、変換規則である。

トポロジ記述の仕様を基にして、図 8 に示すようにメタトポロジモデルを定義した。

シナリオ記述の仕様を基にして、図 9 で示すようにシナリオ記述のメタモデルであるメタシナリオモデルを定義した。

変換規則は、指定した通信オペレーションに応じてシナリオ記述の構造パターンを選択し、その中にトポロジ記述の要素を挿入するというものである。

3.2.1 モデル変換の手順

拡張トポロジ記述を作成することによりシナリオ記述を自動生成するツールについて、その手順を以下に示す。

1. 開発者が拡張トポロジ記述を作成する。
2. 作成した拡張トポロジ記述をデータベースに自動挿入する。
3. 指定した通信オペレーションを参照し、対応するシナリオ記述の構造パターンを選択する。
4. 選択したシナリオ記述の構造パターンに、トポロジ記述の要素を自動挿入する。
5. トポロジ記述の要素を挿入し終了データベースを実行形式に自動変換する。
6. シナリオ記述が完成する。

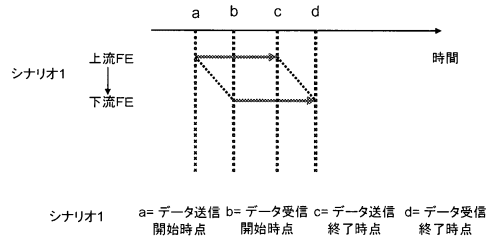


図 10: 指定可能な同期箇所

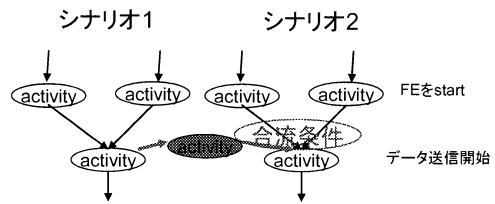


図 11: シナリオの合成方法

3.3 同期箇所を指定したシナリオ記述合成ツール

例えば、洗濯の終了通知をテレビに字幕として表示するだけでなく音楽と共に知らせるサービスを実現する際、2つのアプリケーションを同期箇所を指定して1つのアプリケーションとして合成することで実現できる。

そこで、モデル変換ツール基盤の適用として、2つのシナリオ記述を同期箇所を指定することで1つのシナリオ記述として合成するツールの構成について考える。便宜上、シナリオ1とシナリオ2を同期箇所を指定して新しいシナリオに合成することを考え、考察を行う。まず、指定できる同期箇所についての考察を行う。

ゆかりカーネルにおいては、上流FEのデータ送信開始時点と終了時点、下流FEのデータ受信開始時点と終了時点にイベントが発生する。これらにより、ゆかりカーネル上で指定できる同期のタイミングは、図 10 で示すように4箇所となる。シナリオ1とシナリオ2の入れ替えを考えて8箇所となる。

これらの同期の実現にあたっては、図 11 の中央で示すようにアクティビティの追加をして合流条件とすればよい。このアクティビティはシナリオ1のイベントを取得するものである。

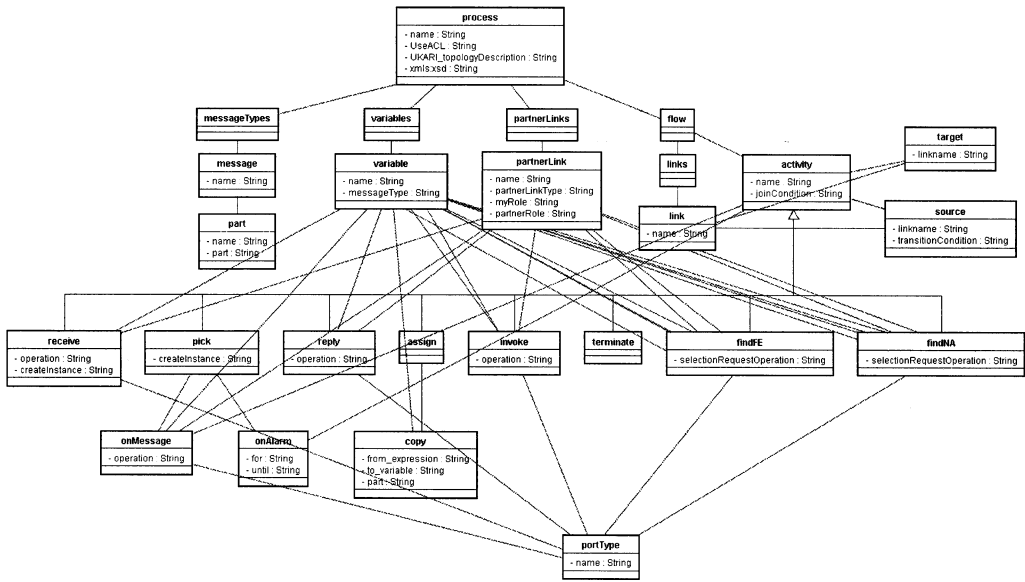


図 9: メタシナリオモデル

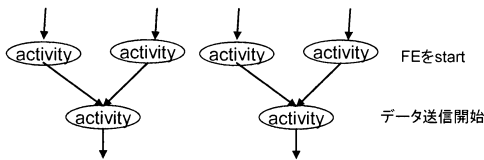


図 12: 事前状態の一例

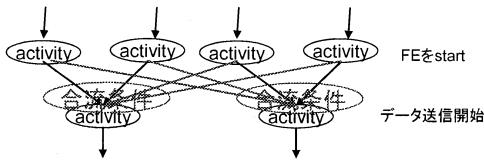


図 13: 事前状態の指定方法の例

この部分に、図 10 における同期させたい箇所を対応させればよい。

次に事前状態の条件としては、以下の 4 種類が考えられる。シナリオ 1、シナリオ 2 の進行について共に制約がない場合、シナリオ 1 の進行についてのみ制約がある場合、シナリオ 2 の進行についてのみ制約がある場合、シナリオ 1、シナリオ 2 の進行について共に制約がある場合である。

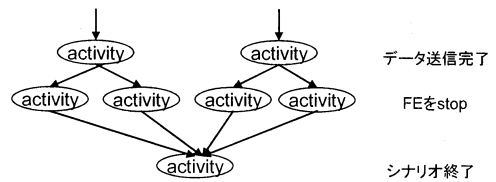


図 14: 事後状態の一例

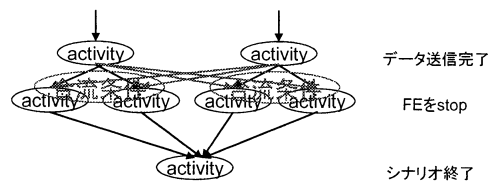


図 15: 事後状態の指定方法の例

事後状態についても同様に、シナリオ 1、シナリオ 2 の進行について共に制約がない場合、シナリオ 1 の進行についてのみ制約がある場合、シナリオ 2 の進行についてのみ制約がある場合、シナリオ 1、シナリオ 2 の進行について共に制約がある場合の 4 種類が挙げられる。

事前状態、事後状態は、図 13、図 15 のように合流条件を指定したリンクを付与により実現

できる。

4 動作確認

モデル変換ツール基盤の有効性を示すため、トポロジ記述からシナリオ記述への自動生成ツールから生成したシナリオ記述をPC上に構築したゆかりカーネルのシミュレーション環境上での動作確認を行った。

冷蔵庫の中の様子をテレビの画面に表示するというサービスを例にとる。冷蔵庫の画像生成するFEとテレビの画像出力するFEを利用してサービスを実現する必要がある。ゆかりカーネル上でこれらのFE間でデータの送信、受信が行われていることを確認した。

5 おわりに

本稿では、ホームネットワークサービスの開発支援として、モデル駆動アーキテクチャの概念をホームネットワークサービスの開発支援を目的として適用し、モデル変換ツール基盤の提案を行った。メタモデルによりデータベーススキーマを定義し、データベース上でのモデル変換を行う設計をした。さまざまな記述形式で適用可能であり、サービス開発の部分的な自動化、仕様変更への追従や保守などが可能となる。本稿ではゆかりカーネルへの適用を行った。

ゆかりカーネルにおいてシナリオ記述の作成が負担となっていることから、シナリオ記述の作成を支援するツールの作成を行った。シナリオ記述の基本的構造は通信オペレーションの指定により決定されることを確認し、トポロジ記述に付加情報として通信オペレーションを付与した拡張トポロジ記述を作成することでシナリオ記述が自動生成するツールの作成を行った。メタトポロジモデル、メタシナリオモデルをそれぞれ定義し、変換規則としてシナリオ記述の構造パターンの選択、選択したシナリオ記述の構造パターンへのトポロジ記述要素への挿入を定義した。

また、2つのシナリオ記述を同期箇所を指定して1つのシナリオ記述として合成するツールの構成方法についての検討を行い、同期箇所、事前状態、事後状態の考察を行った。

ツールにより生成されたシナリオ記述をゆかりカーネルのシミュレーション環境上で動作することを確認し、有効性を示した。

参考文献

- [1] 小河原成哲, 小林英嗣, 依田育生: ネットワークサービス管理技術, NTT 技術ジャーナル, Vol. 16, No. 11, pp. 12-15 (2004).
- [2] 澤田淳二, 下坂久司: グリッド家電メタホームネットワーク環境の構築一, 情報処理推進機構:事業成果報告集 (2004).
- [3] パツアラ・リーラープルット, 中村匡秀, 井垣宏, 松本 健一, 菊野亨: ホームネットワークシステムにおけるサービス競合の分類と解消について, 電子情報通信学会技術研究報告, Vol. 105, No. 628, 電子情報通信学会, pp. 55-60 (2006).
- [4] James Rumbaugh, Ivar Jacobson, G. B.: *The Unified Modeling Language Reference Manual (2nd Edition)*, Addison-Wesley Professional (2004).
- [5] MDA, <http://www.omg.org/mda/>.
- [6] Gitzel, R. and Schwind, M.: Experiences with hierarchy-based code generation in the J2EE context, *PPPJ '06: Proceedings of the 4th international symposium on Principles and practice of programming in Java*, pp. 216-223 (2006).
- [7] Harrison, W., Barton, C. and Raghavachari, M.: Mapping UML designs to Java, *OOPSLA '00: Proceedings of the 15th ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications*, pp. 178-187 (2000).
- [8] Tatsubori, M., Imamura, T. and Nakamura, Y.: Best-Practice Patterns and Tool Support for Configuring Secure Web Services Messaging, *icws*, p. 244 (2004).
- [9] 佐藤史子, 中村祐一, 小野康一: モデル駆動型開発に基づくセキュリティ設定ーWeb サービスのための認証設定ー, 情報処理学会研究報告, No.2006-SE-151, pp. 151-158 (2006).
- [10] 沢田篤史, 多鹿陽介, 山崎達也, 美濃導彦: ゆかりコア: ネットワーク家電のための分散協調型サービス構築基盤, 電子情報通信学会記述研究報告, ソフトウェアサイエンス研究会 (2004).
- [11] 美濃導彦: ゆかりプロジェクトの目的と概要ーUKARIプロジェクト報告No.1ー (2004). 3TTC-2.
- [12] 沢田篤史, 多鹿陽介, 山崎達也, 美濃導彦: 機能協調型家電ネットワークのためのサービスシナリオ記述方式, 情報処理学会ソフトウェア工学研究会, No. 2004-SE-145 (2004).