

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

沢田 篤史^{†‡}, 多鹿 陽介^{*}, 山崎 達也[‡], 美濃 導彦^{†‡}

[†] 京都大学 学術情報メディアセンター

〒 606-8501 京都市左京区吉田二本松町

[‡] 独立行政法人 情報通信研究機構

*(株) 東芝研究開発センター

本稿では、「ゆかりコア」と呼ばれる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の上で、家電、センサ、ロボットなどのアプライアンスがその機能を互いに協調させながらサービスを構築するためのシナリオ記述方式を提案する。提案するシナリオ記述は、Web サービス連携のための言語 BPEL4WS の考え方に基いており、ユーザおよび環境とのインタフェースに、アプライアンスの機能発見と制御のインタフェースを統合したサービスフローの記述を可能とする。本稿では、提案するサービスシナリオ記述方式の設計について説明するとともに、単純なアプリケーション記述の例を通じて、今後取り組むべき課題について考察する。

キーワード：ネットワーク家電，ホームネットワーク，機能協調，シナリオ記述，サービス構築プラットフォーム

A Service Scenario Description Framework for Functional Cooperative Environment of Networked Appliances

Atsushi SAWADA^{†‡}, Yosuke TAJIKA^{*}, Tatsuya YAMAZAKI[‡], Michihiko MINOH^{†‡}

[†] Academic Center for Computing and Media Studies, Kyoto University

Yoshida-nihonmatsu, Sakyo, Kyoto 606-8501, Japan

[‡] National Institute of Information and Communications Technology

*R&D Center, Toshiba Corporation

In this paper, we propose a new service scenario description framework for constructing home network services based on the distributed cooperative appliance platform named “UKARI-Core”. Proposed framework is based on BPEL4WS — a web service harmonization language, and we introduce several new constructs to handle and integrate user interface, environmental interface, and appliance function interface. In this paper, we overview the design of our description framework. Description examples are shown for some simple home applications to discuss the merits and demerits of our framework.

Keywords: Networked Appliance, Home Network, Cooperative Function, Scienario Description, Service Construction Platform

1 はじめに

近年、TV放送のデジタル化など様々な要因に支えられて、多様なデジタル家電が開発されている。また、高速ネットワーク技術の一般化に伴い、今後家庭内の多種多様な機器がホームネットワークに接続された環境が普及することが見込まれる。

このような背景のもと、情報通信研究機構(NICT)が主導して遂行中のゆかりプロジェクト(UKARI: Universal Knowledgeable Architecture for Real-life appliances)プロジェクト[1]では、家電、センサ、ロボットなどのアプライアンスがホームネットワーク上で互いに連携しながら、家庭環境に適応したサービスを提供するための基盤ミドルウェア[2]と、それを利用した知的でインタラクティブなサービスアプリケーション群の構築を目指して研究開発を行っている。

家電、センサ、ロボットなど、異種のアプライアンスが互いに連携するアプリケーションサービスを構築するためには、サービスに参加する機器を統合的に制御するソフトウェアプロキシの存在が不可欠であり、参加する機器が多種多様となり、機器間の連携が密になればなるほどそのソフトウェアの構成は複雑で大規模になる。このように、ネットワーク家電向けサービスアプリケーションにおけるソフトウェアの比重は今後も高まり、短期化する製品サイクルとあいまって、ソフトウェアの生産性向上への取り組みが急務になってきているといえる。

本稿では、ゆかりプロジェクトにおいて開発中の連携基盤「ゆかりコア」[3]上のサービスシナリオ記述方式を提案する。ネットワーク機器の連携を目指した既存の通信基盤に対するゆかりコアの特長は、各種アプライアンスの持つ機能を単位に発見、接続することで、機器をまたいで柔軟な接続、協調、補完を可能とする点にある[4]。提案するシナリオ記述方式では、ホームネットワーク上に分散した機能が互いに協調しながらサービスを提供するための動作仕様を、BPEL4WS[5, 6]に基づく、形式的で了解性の高い記法により記述する。アプライアンス機能間のトポロジに関する抽象的な記述[7]と

動作仕様記述を組み合わせることで、サービスとユーザや環境とのやりとり、機器の機能発見や接続などの手順を制御するソフトウェアの開発負担を軽減することが本研究の目的である。

本稿では、以下2節で分散機能協調型家電サービス基盤 ゆかりコアについて概説し、サービスアプリケーションを形式的に記述する枠組に求められる要件について整理する。3節ではシナリオ記述の設計について説明する。さらに4節では実家電を用いたサービスとそれを制御するシナリオ記述の例について検討し、5節で本稿のまとめをする。

2 ゆかりコアとサービス記述

本節では、現在我々がプロトタイプの実装と改良を進めているゆかりコアについて概説する。ゆかりコアは実家電を用いたサービス構築基盤として試用されているが、その過程を通じて明らかになった改善項目と、サービス記述に求められる要件について整理する。

2.1 ゆかりコアとその実装

ゆかりコアは、家庭内に存在するアプライアンスをそれが持つ機能を単位に接続、協調、補完させることにより、家庭全体で機能重複の無駄を省き、ユーザの状況に応じた最適なサービスを提供することを目的として開発されたサービス構築基盤である。

ゆかりコアの構成概略を図1に示す。ネット

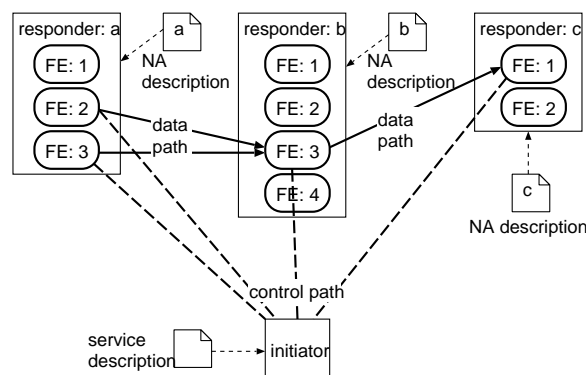


図 1: ゆかりコアの構成概略

ワークに接続されるアプライアンス (NA: Networked Appliance と呼ぶ) は, サービス実行上の役割によって, イニシエータ (initiator) とレスポнда (responder) に区別される. イニシエータはサービス記述 (service description) に基づいて機能の発見, 接続を行う. 一方, レスポндаはアプライアンスの持つ個々の機能 (FE: Function Element と呼ぶ) の接続や実行を管理する. NA が持つ FE の仕様は, NA 記述 (NA description) としてあらかじめ機器毎に記述され, レスポндаはこの記述に従って FE の管理をする. つまり, ゆかりコアは, サービス記述と NA 記述に基づいて必要な FE を発見し, イニシエータ・レスポнда間の制御パス (control path), およびレスポнда・レスポнда間 (FE 間) のデータパス (data path) を接続するまでの手順をサポートするプラットフォームであるといえる.

イニシエータによる機能の発見は FE の種類 (FE タイプ) を指定して行われる. FE タイプは, 我々がこれまでに提案してきた AMIDEN アーキテクチャ[8, 9] の考え方に基いており, 「音声生成機能」あるいは「画像消費機能」などのように, メディアとそれに対する操作を組み合わせた形に分類される. 表 1 に FE タイプの種類を示す.

表 1: メディアと操作による FE の分類

	生成	消費	合成	変換	蓄積
ブーリアン	Bg	Bc	Bm#	B?t	Bs
数値	Ng	Nc	Nm#	N?t	Ns
文字	Tg	Tc	Tm#	T?t	Ts
音声	Ag	Ac	Am#	A?t	As
画像	Ig	Ic	Im#	I?t	Is
映像	Mg	Mc	Mm#	M?t	Ms

は合成するインターフェース数

? は B,N,T,A,I,M のいずれか

イニシエータに与えられるサービス記述の概要を図 2 に示す. サービス記述は, ロール (role) とパス (path) に関する記述により構成される. ロールは, サービスを構成する機能の役割に対応するものであり, ロール名に FE タイプを組み合わせたものとして記述する. また, ロール間の接続関係はパスで示される. 機能発見手続きでは, ロールに指定された FE タ

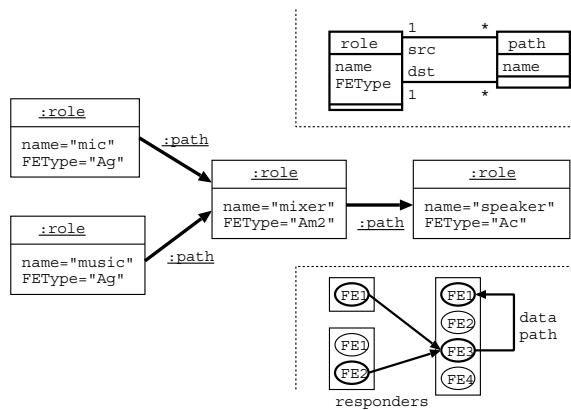


図 2: ゆかりコアのサービス記述

イプをキーに FE が発見され, 発見された FE の間には, サービス記述に示されたパスに従ってデータパスが構築される.

現在, ゆかりコアを通じて機能間接続可能な家電は表 2 の通りであり, これらのうちの一部

表 2: ゆかりコア対応機器と機能

機器名	FE タイプ
冷蔵庫	Tg, Ig
洗濯機	Tg
プラズマテレビ	Ic, Mc, Tc, Ac
時計	Tg, Tc, TAt, Ac
スピーカ	Ac, Bc
デジタルカメラ	Ig, Mg, Ag, Is, Ms, As

は, NICT けいはんな情報通信融合センター内に建設されたユビキタスホーム (図 3) に設置, 稼働状況にある.



図 3: NICT ユビキタスホーム

2.2 サービス記述に求められる要件

前項で述べたように、ゆかりコアでは、NA 記述とサービス記述を導入することで、次の手順を実現する際に必要なプログラミングのコストを軽減することができる。

- サービスを提供するために必要となる家電の機能を FE として発見
- サービス制御を担うイニシエータと FE との間に制御パスを確立
- FE と FE との間に適切なデータパスを確立

このことはすなわち、現状のゆかりコアが、異種の NA を FE 単位で組み合わせたサービスの初期化手順のみを主にサポートしており、初期化が完了してから FE をどのように制御し、FE 間のデータ通信をどの条件で行うのかといった、サービスの振舞いに関しては関知していないことを示している。実際、ユビキタスホームでのサービス構築経験からも、機器を連携するサービスを実現するにあたり、設計者はただ NA 記述とサービス記述を用意すれば良い訳ではなく、サービスの振舞いに関する詳細な論理をプログラムとしてイニシエータに作り込まなければならず、その手間は当初予想していたより大きいことが明らかになっている。

サービス実行中のイニシエータは、サービスに参加する複数の FE を統合的に制御するソフトウェアプロキシの役割を担うことになるが、参加する FE の数が多くなり、FE 間の連携が密になればなるほど、構成は複雑で大規模になる。またこういったプロキシは、個々のアプリケーション毎にその論理に依存した形で構築しなければならないため、新たなサービスを追加する毎に開発負担が嵩むことになる。このような問題に対し、サービス記述の要素として機能間のネットワーク的な接続関係（トポロジ）だけでなく、サービスの振舞いの側面をシナリオとして適切な抽象度で形式的に記述でき、その機械的な実行が可能であれば、異機種家電を連携したサービス開発にかかる負担の大幅な軽減が期待できる。

このように、ゆかりコアのサービス記述には、サービスの振舞いをシナリオとして記述するための方式が必要となる。このシナリオには、FE の実行制御や、FE 間のデータ受渡しを記述できることがまず求められる。ただし、生活支援サービスに関わる要素は、家電だけではなく、ユーザインタフェース機器や環境センサ、ロボットなど多様であり、それらを画一的に FE に分類することはできない場合も多い。このため、シナリオには、ユーザ、ロボット、環境センサのようなサービス構成要素が FE との間にどのような働きかけを行うのかについても、FE の振舞いと同様に記述できることが求められている。

まとめると、サービスの振舞い仕様記述方式としてのシナリオ記述には、

- サービスの振舞いを制御の流れとして記述できる
- ユーザやロボット、環境センサとのやりとりを記述できる
- FE 発見手順を含め、FE とのやりとりを記述できる
- 記述の操作的意味が明確である

などの要件が求められているといえる。

3 シナリオ記述の設計

前節で述べた要件を満たすシナリオ記述を設計するにあたり、本研究では BPEL4WS を参考にした。BPEL4WS は複数の Web サービスを連携させてビジネスプロセスを構成するための言語で、記述の自動実行が可能である。BPEL4WS の仕様には、接続、分岐、繰り返しといった通常の制御の流れの他に、長期トランザクションや相関定義、補償プロセスなど、Web サービスを連携させるアプリケーションにとって利用価値の高いものを含め、多くのレベルの言語要素が定義されている。

このように BPEL4WS は、Web サービスに対する多種多様な要求を実現するために設計されたリッチな記述体系であるが、家庭におけ

るアプリケーションの観点では過剰な仕様と考えることもできる。家庭内サービスの構成要素は、実際にユーザの手が届くものがほとんどであり、直接機器に触れ、場合によっては物理的な作用を加えることも可能である。また、ユーザも家庭という声の届く範囲に基本的には存在し、仮にサービスが競合するような場面でも、ユーザ間の対話により調停することもできる。

本研究では、このような家庭内での利用を前提に、BPEL4WS 仕様限定を加えるとともに、FE、ユーザ、環境のためのインタフェースを加えたシナリオ記述を提案する。つまり、家庭向けサービスにとって適度なレベルの記述力を持ち、組込み系の家電機器上で軽快な動作が可能となる程度に簡潔な記述方式の整備がシナリオ記述設計の目的であるといえる。

3.1 シナリオ記述の位置づけ

図 4 にゆかりコアでのシナリオ記述の位置づけを示す。図中央部に示したプロセス (process)

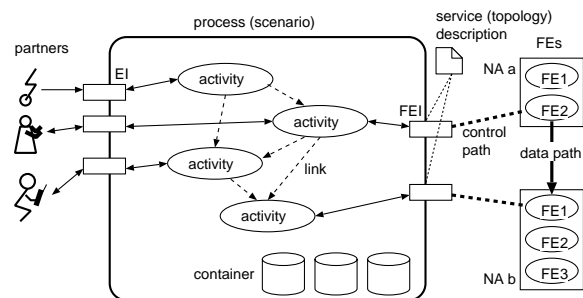


図 4: ゆかりコアにおけるシナリオ記述

cess) 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し、アクティビティ (activity) とアクティビティ間の制御依存関係を表すリンク (link) により、制御の遷移手順を記述することができる。プロセスと外部とのデータのやりとりの結果得られるデータはコンテナ (container)¹ に格納され、プロセス内の複数のアクティビティによって参照可能となる。シナリオ記述に関するこれらの考え方は BPEL4WS と同等のものである。

図左部に示す FE インタフェース (FEI) は、

¹BPEL4WS のバージョン 1.1 では変数 (variable)。

プロセスと FE との間でやりとりを行うためのポートであり、FE の発見や FE タイプ毎の通信パターンを含め、FE の制御インタフェースがここに定義される。一方、図右部の環境インタフェース (EI) は、FE に分類することのできないサービス要素とのやりとりを行うためのポートであり、主にセンサやロボット、ユーザとの間で制御やデータを受け渡すために用いられる。BPEL4WS では、プロセスと外とのインタフェースを WSDL ポートタイプで表現し、ポートを通じて入出力されるデータの仕様については個別に定義される。これに対して提案するシナリオ記述では、プロセスの外部インタフェースが FE インタフェースと環境インタフェースに大別されている点で異なる。

3.2 FE インタフェース

FE インタフェースは、ゆかりコアの持つ動的機能発見機構や、機能単位での FE 接続といった手順をシナリオから駆動するために設けられたポートである。

ゆかりコアでは FE 間の接続関係 (topology) を示したサービス記述にしたがって FE の発見や接続を行う。一つの FE インタフェースはこのサービス記述 (図 2 参照。シナリオ記述との役割分担を明確にするため、以下、トポロジ記述と呼ぶ) に指定されたロール一つに対応する。すなわち、FE インタフェースはトポロジ記述のロール名によって区別され、例えば FE インタフェースの「FE 発見」オペレーションを呼び出すと、ロールに指定された FE タイプをキーに FE を発見し、FE インタフェースとの間に制御パスを構築し、関連する FE との間にデータパスを構築する。

ゆかりコアでは FE 発見とネットワーク接続に関する手順を自動化することができるが、FE インタフェースではこれらの手順に対するインタフェースの他に、サービス実行中の制御インタフェースについても提供できるように設計した。FE のトポロジが決まれば、FE が受け付ける制御やデータ通信のパターンをある程度類形化できる。そのパターンを FE インタフェースに

において提供することで、実際に発見されて利用される機器の種類には依存しないシナリオの記述が可能となる。

FEの実行制御に対するオペレーションは、機能実行の開始、終了、中断、再開の四つであり、FEインタフェースにはそれぞれのオペレーションに対するポートを定義するとともに、FEからの返答のためのポートが定義される。

また、FE間の通信制御に対するオペレーションは、図5に示す通信パターンをサポートする。すなわち、下流側のFE (FE2) から上

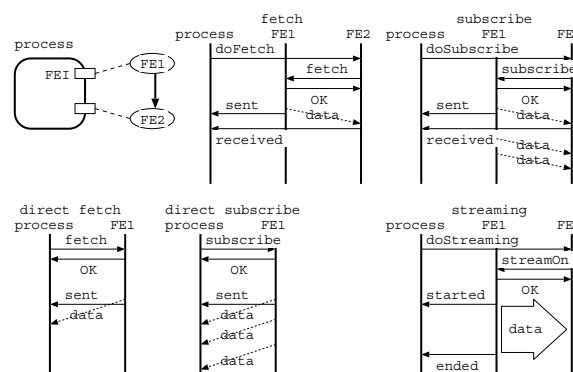


図 5: FEI での通信パターン

流側 FE (FE1) に対するデータ取得 (fetch)、下流側から上流側へのデータプッシュの依頼 (subscribe)、下流側から上流側へのストリーミング開始依頼 (streaming) のパターンに対し、プロセスの操作および FE からプロセス向けのイベントに対するインタフェースが定義される。また、プロセスから FE に対して直接データを取得するパターン (direct fetch) と直接プッシュを依頼するパターン (direct subscribe) に対応するインタフェースも定義される。

FEインタフェースは、一定のオペレーションに対して FE が一定の振舞いをするのが規定されているという意味で、サービスに依存するオペレーションを個別に定義する必要がない。このような制限を加えることで、サービス設計と FE 設計 (機器設計) の独立性を高めることができる。

3.3 環境インタフェース

環境インタフェースは、センサやロボット、ユーザといった、FEには分類できない対象とプロセスとのインタフェースを記述するものであり、オペレーションとそれに対する振舞いが自由に定義できるという点で、WSDLのポートタイプにより近い要素であるといえる。

ただし、環境インタフェースには完全な自由定義を認めるのではなく、NA や FE に関する情報が FE インタフェースを介して流通可能となるように、オペレーションのメッセージには、パートナーとしての実体をホームネットワーク上で一意に表す NA-ID と NA 名 (ユーザ理解可能な名前) が一律付与されるものとする。

BPEL4WS のポートを介してプロセス外部からシナリオの持つオペレーションを呼び出す際は、エンドポイントとしてプロセスの実体を指定して行われる。このモデルを家庭内のサービス提供形態に適用することを考えると、例えばユーザが利用したいサービスを指定して起動等のオペレーションを行う形態には向いているが、センサから何らかのイベントが発生した場合に自律的にサービスが起動するといった形態には向かない。

後者のような形態でのシナリオを記述するため、環境インタフェースでは、エンドポイントとオペレーション名を指定した通信形態とともに、エンドポイントもオペレーションも指定しない通信形態もサポートする。

エンドポイントを指定しない通信形態では、受け渡されるメッセージの型がキーとなりエンドポイントとそのオペレーションの名前が解決されるものとする。これにより、特定の型を持つセンサイベントをトリガとしたシナリオの発火が可能となる。この形態の通信を可能とするためには、メッセージをその型に応じて適切なプロセスに仲介する機構が必要になる。サービスシナリオ記述の仕様としては仲介機構の実現方式まで規定はしないが、例えばブローカサーバにプロセスがデータプッシュを依頼する方式や、黒板 (ブロードキャスト / マルチキャスト) に投げられたメッセージを各プロセスが監視する方式が考えられる。

3.4 アクティビティ

本研究で提案するシナリオ記述言語では、プロセスの構成要素となるアクティビティの種類を表3に示す八種類とした。BPEL4WS との

表 3: アクティビティの種類

receive	FEI, EI を通じてデータを入力
reply	FEI, EI を通じてデータを出力
invoke	FEI, EI を通じてオペレーションを起動
assign	コンテナにメッセージの内容を代入
terminate	プロセスの実行を停止
pick	FEI, EI を通じて入力されたデータ種に応じたアクティビティに遷移
findNA	FEI を通じ、キーワードなどを指定して NA を発見
findFE	FEI を通じ、トポロジ記述上の FE を発見

大きな相違は、pick 以外の構造要素を排除し、アクティビティのシーケンスを制御依存リンクのみで表すようにした点にある。このことでプロセスの実行解釈論理が単純になり、組込みシステム上での軽快な動作が可能となることが期待できる。

また、NA や FE の発見論理を一括して扱うためのアクティビティ (findNA, findFE) を新たに導入した。これらのアクティビティはゆかりコアの機能発見機構に対する定型的な手順をマクロ的に用意したものであり、発見手順の詳細な制御が必要な場合は、他のアクティビティの組合せで記述することも可能である。

アクティビティ間の制御依存リンクの意味やリンクの遷移条件、発火条件は BPEL4WS と同様に指定できる。

4 サービスシナリオの記述例

本節では、前節で設計したシナリオ記述方式を用いて、NICT ユビキタスホームにおいて既に稼働している簡単なサービスのシナリオを記述する。

図6は、洗濯機の状態通知サービスを示す。シナリオは、洗濯機からの状態変化(洗濯終了など)イベントをEIを通じてreceiveアクティビティにより受け取り、イベント発生元である

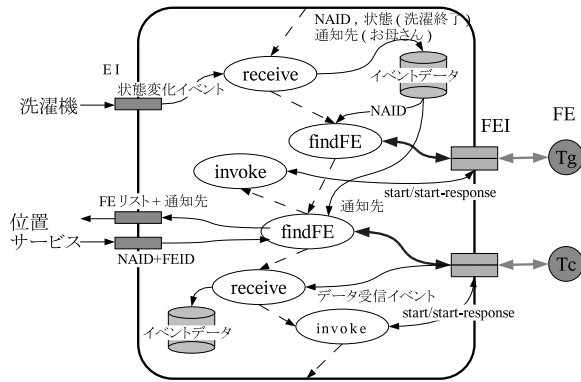


図 6: 洗濯機状態通知

洗濯機の状態を通知する FE (Tg) を findFE アクティビティにより発見する。さらに、その状態を表示する FE (Tc) を発見するが、その際に位置サービスを EI 経由で呼び出し通知先のユーザ(お母さん)に近い FE を選択する。しかる後に、各 FE の機能を invoke アクティビティにより開始させ、データ通信を行うことで、サービスが完了する。このサービスのトリガとなる洗濯機からのイベントは、匿名での(エンドポイントを指定しない)通信形態で実現されることを想定している。

同様に図7は、冷蔵庫の中身画像を表示するサービスである。これはロボット (Robovie)

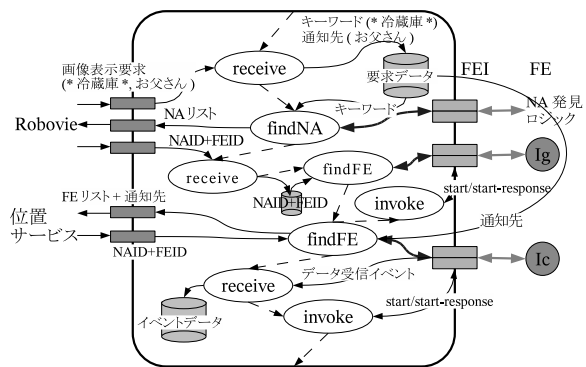


図 7: 冷蔵庫画像表示

から明示的に起動されるサービスである点と、FE インタフェースを通じて「冷蔵庫」という名前を持つ NA を発見してから、その NA が持つ FE を接続している点で上のサービスと異なるが、他の FE 発見や通信、位置サービスの利用手順についてはほぼ同様である。

5 おわりに

本稿では、家電ネットワーク向け機能協調基盤ゆかりコアの上でサービスの振舞い仕様を記述するためのシナリオ記述方式を提案した。提案する記述は BPEL4WS の考え方に基づいているが、機能発見のためのインタフェースを加えるなど機能協調家電ネットワーク向けの言語要素を新たに加えている。また、構造要素を排除するなど、軽量な実装を指向した単純化を行っている点も一つの特長となっている。

今後の課題は、まず提案するシナリオを解釈実行するエンジンを実装し組み込みシステム上での稼働を確かめ、性能を評価することが挙げられる。本稿で提案したシナリオ記述では、構造要素を排除して単純化を図ったが、構造化のための要素を今後導入するか否かについては、実装を評価した結果を精査し、記述性・可読性と性能とのバランスで決定する予定である。また、シナリオ記述によるサービス設計を支援するツールの整備も今後の課題となる。設計支援の一環として、モデル検査検証などの技術を利用した検査ツールなどについても、中島 [10] の研究成果などを参考に今後取り組んでいきたい。

謝辞

本研究の一部は、科学研究費補助金 若手 (B) 16700031 の補助を受けて行った。

参考文献

- [1] 美濃導彦：“ゆかりプロジェクトの目的と概要 — UKARI プロジェクト報告 No.1 —”，情報処理学会第 66 回全国大会講演論文集，3TTC-2，2004。
- [2] 山崎達也，沢田篤史，多鹿陽介，大倉計美，中尾敏康，マハダドヌリシラジ，佐野睦夫，金田重郎：“ゆかりプロジェクトにおける分散協調基盤ミドルウェア — UKARI プロジェクト報告 No.2 —”，情報処理学会第 66 回全国大会講演論文集，3TTC-3，2004。

- [3] 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。(to appear)
- [4] 多鹿陽介，沢田篤史，中尾敏康，美濃導彦：“分散型機能協調基盤上の家電・センサ機能単位化およびその統合”，電子情報通信学会 2004 年総合大会講演論文集，A-16-9，2004。
- [5] Curbera, F., Golland, Y., Klein, J., Leymann, F., Roller, D., Thatte, S., and Weerawarana, S.: Business Process Execution Language for Web Services, Version 1.0, 2002.
- [6] Andrews, T., Curbera, F., Dholakia, H., Golland, Y., Klein, J., Leymann, F., Liu, K., Roller, D., Smith, D., Thatte, S., Trickobic, I., and Weerawarana, S.: Business Process Execution Language for Web Services, Version 1.1, 2003.
- [7] 多鹿陽介，沢田篤史，美濃導彦，釜江尚彦：“単機能分散型ネットワークドアプリケーションにおける統合サービス記述の検討”，電子情報通信学会技術研究報告，ソフトウェアサイエンス研究会 (SS2003-53)，Vol. 102，No. 704，pp. 1-6，2003。
- [8] 多鹿陽介，安次富大介，中村素典，美濃導彦，釜江尚彦：“ホームネットワークに適した単機能分散型ネットワークドアプリケーションアーキテクチャ”，情報処理学会論文誌，Vol. 44，No. 9，pp. 2320-2333，2003。
- [9] AMIDEN 研究会：AMIDEN アーキテクチャ仕様書 第 2 版，財団法人イメージ情報科学研究所，2004。
- [10] 中島震：“Web サービスフロー記述のモデル検査検証”，情報処理学会論文誌，Vol. 44，No. 3，pp. 942-952，2003。