

環境適応型サービス連携フレームワークにおける ストリーム指向アプローチの提案

岩田 哲弥 徳元 誠一 大石 哲矢 武本 充治[†]
日本電信電話株式会社 NTT ネットワークサービスシステム研究所[†]

1. はじめに

近い将来、身の回りの様々なものにコンピュータが組み込まれ、ネットワークを介して相互に接続が可能なユビキタスコンピューティング環境（以降、ユビキタス環境）が実現される。ユビキタス環境では、ユーザサイドで、その場の環境や、その時のユーザの嗜好に適切な機器・機能を発見し、合成してサービスを提供する環境適応型サービス連携フレームワークが期待されている[1]。その合成はテンプレートという設計図を用いて行うが、本稿では、その合成に用いるテンプレートの記述方法として、従来のオペレーション指向とは異なる、ストリーム指向の記述方法を提案し、また、その実現手法について述べる。この方法は、利用者にとってより簡便に直感的に記述でき、ユビキタス環境への親和性の高い方法である。

2. サービス合成フレームワーク

2.1 サービス合成とは？

本サービス合成技術は、サービスの設計図であるサービステンプレート(Service Template: ST)に従い、ユビキタス環境に偏在する機器や機能を総称するサービス構成要素(Service Elements: SE)を結びつけて合成サービスを実現する。STには、SEそのものではなく、同等のサービス(片)を提供するSE群であるサービスカテゴリのレベルで、そのフローが書かれているところに特徴がある。ユーザはサービス要求時にまず自分の希望に合致するSTを自作するか探してダウンロードした後、次にSTに記載のサービスカテゴリに合致するSEを探索するという動作をする。このようにサービス提供の直前にSEを探索する動作により、ユーザ個人の嗜好に合致し、かつその時点で利用可能なSEの取得が可能となる。[1]

2.2 ユビキタス環境でのサービス合成の要求条件

- 流通性：STの流通性が高いこと。いつでもどこでも誰でもどんな環境でもサービス合成に適応可能なこと。
- 適応性：その場その場の環境に適応し、ユーザ満足度が最大になる形で、継続的にサービス合成可能なこと。
- 容易性：ST記述が極めて容易で、理解性が高いこと。

2.3 BPEL

サービス合成フレームワークの実現にあたっては、すでに実装、標準化、普及している技術をベースにすることが望ましい。このような技術として、B2B環境がターゲットではあるが、複数のWebサービスを合成するBPEL4WS(BPEL)がある。BPELでは、構成各Webサービスのどのメソッドをどの順序で呼び出すか、呼び出しパラメータにどんな値を入れるか、呼び出し結果パラメータの値をどのように利用するのかを記述し、それで合成することができる。

2.4 BPELのユビキタスへの適用

BPELでは、構成各Webサービスについて、メソッド名、パラメータ名、型の全てが厳密に一致する必要がある。しかし、ユビキタス環境では、要求条件(A)より、その場その時にSEを発見し、その時点で初めてその具体的なメソッド名、パラメータ名、型がわかるため、BPELのように

個々のSEを特定せず、抽象的に記述する必要がある。したがって、筆者らは、SEはカテゴリで表現し、パラメータ名は共通的なオントロジで表現する方法を提案している。その具体的なイメージを図1・2に示す。

```
<sequence>
  <invoke category="検索サービス"/>
  <copy from="検索サービス.検索結果" to="翻訳サービス.翻訳語"/>
  <invoke category="翻訳サービス"/>
</sequence>
```

図1 STの記述イメージ

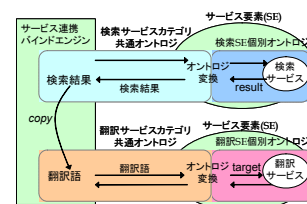


図2 サービス合成の動作イメージ

3. 現状のサービス合成フレームワークの問題点

3.1 オペレーション指向でのST記述での問題点

上記の例のように、プログラムライクに、各SEのどのメソッドをどの順序で呼び出し、呼び出しパラメータにどんな値を入れるかに着目しSTを記述する（以降、オペレーション指向記述）方法では、ユビキタス環境への適応性が低い改善すべき点がある。次の図3に示すサービスを例に考える。近くのTV電話の映像を近くのディスプレイに大きく映し、部屋にあるビデオで録画し、その音声は同時通訳され、手元のヘッドセットで聞こえるというものである。部屋を離れるときは、携帯電話に表示もできる。

図4にTV電話⇒ディスプレイ部分の動作イメージを示す。

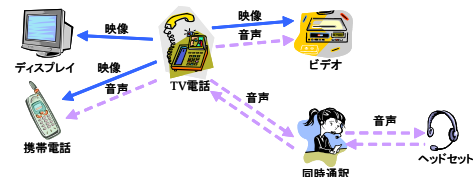


図3 例として考える合成サービス

す。このようにSE側にバインドエンジンとのインタフェースやストリーミング制御・送受信機能があるものとする。

- カテゴリごとオントロジの使用：（要求条件(C)が不十分。）
TV電話の映像を各装置に入力するという点では同じでも、その対象装置のカテゴリごとに、インタフェースを調査し、STを記述しなければならぬ。
例) ディスプレイではdisplay(), 携帯電話ではshowImage(), ビデオではrecord().
- その場の同等機能を持つ他のSEへの適応不可：（要求条件(B)が不十分。）
(ア) 近くにディスプレイはないが携帯電話がある場合、ディスプレイ出力用のSTは適用できない。
(イ) 近くにヘッドセットはないがスピーカーがある場合、このSTは適用できない。
- コンテンツフォーマット等意識した記述の必要性：コンテンツフォーマット (MPEG2 / MPEG4 / WMA /

Stream-oriented Service Coordination Framework on Ubiquitous and Context-aware Computing Environments

[†]Tetsuya Iwata, Seiichi Tokumoto, Tetsuya Oh-ishi and Michiharu Takemoto

[†]NTT Network Service Systems Laboratories, NTT Corporation

- MP3)・転送方式(ストリーミング/ファイル)・プロトコル(RTSP/SIP)等についてSE間で整合をとる必要がある。しかし、それらに応じて、通常、SEのインタフェースは異なっており、
- (ア) ST記述の際にそれらを意識する必要がある、わずらわしい。(要求条件(C)が不十分。)
 - (イ) 環境ごとに多様なSEが存在するユビキタス環境では、その分だけSTの適応性が低くなる。(要求条件(B)が不十分。)

4) 終了処理記述の必要性:

TV電話終了時にビデオ録画も中止するためには、STで明示的に、終了イベントを受け、ビデオ録画中止メソッドを呼ぶ処理を明示的に記述する必要がある。

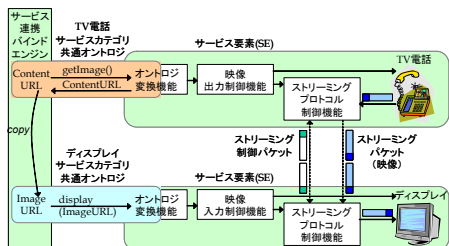


図4 オペレーション指向ST記述でのサービス合成(要求条件(C)が不十分)

4. 提案するST記述・合成方法

4.1 ストリーム指向ST記述・サービス合成方式

上記の問題点を解決するため、UNIXパイプ、C++ストリーム演算子/Java入出力ストリームクラスを参考に、以下の特徴を持つST記述・サービス合成方式を提案する。

【ST記述】

1. コンテンツの流れ(ストリーム)のみ記述する。
2. コンテンツフォーマット、転送方式、プロトコル、端末能力に関する内容は記述しない。
3. 送受信端末でのストリームハンドリング処理に関する内容は記述しない。
4. 「映像出力」「音声加工」等、コンテンツ処理能力の観点に基づくカテゴリを追加して利用する。

【サービス合成フレームワーク】

5. フレームワークで、やりとり可能なコンテンツフォーマット、転送方式、プロトコル、帯域、ストリームハンドリング処理等の決定、整合性判定を行う。各SEやカテゴリにそれらの能力情報を持たせ、それを参照することで判定する。その際、流れる情報の量がよく質が高くなるように留意する。
6. ストリームのリソース管理(close等)、それに伴うSEの動作終了等はフレームワークで自動的に行う。
7. 整合性が解決できないとき、アダプテーションSEを探し、解決することも可能である。

```
<stream>
<link from="TV電話" to="映像出力" content="movie"/>
<link from="TV電話" to="映像記録" content="movie"/>
<link from="TV電話" to="音声変換" content="voice"/>
<link from="音声変換" to="音声出力" content="voice"/>
</stream>
```

図5 ストリーム指向ST記述イメージ

```
<category name="TV電話">
<capability type="output">
<content type="movie"/>
<content type="voice"/>
</capability>
</category>
<category name="映像出力">
<capability type="input" content="movie"/>
<capability type="output" content="movie-for-human"/>
</category>
<category name="映像記録">
<capability type="input" content="movie"/>
<capability type="output" content="movie"/>
</category>
```

図6 能力定義ドキュメントのイメージ

このST、能力情報の記述イメージを図5・6に示す。

4.2 提案方式の効果

提案方式では、能力情報の記述や参照が必要だが、STは簡潔に利用者の直感に近い形で記述でき、理解性が高い。また、特定の装置やプロトコル等に依存した処理、メソッド名やパラメータ名等の情報を記述していないため、流通性も高い。そして、「映像出力」のようなコンテンツ処理能力で表現されているため、ディスプレイがあるところではディスプレイで、携帯しかないところでは携帯に出力するということが可能であり、適応性が高い。一方、欠点としては、フレームワークで解決する内容が多い分だけ実装量が多く、重いこと、各SEでストリーム入出力に対応した標準処理しか行えず複雑な制御は不可能なこと、コンテンツ関連処理以外への適用はやや難しいことがある。つまり、従来方式を補強するもので排他するものではない。

4.3 実現方式

図7に実現方法を示す。4.1項の特徴5,6,7の機能がフレームワークで提供される。SEごとに実装が必要なのは、フレームワークへのデバイスドライバ的な位置づけの、図中のアダプタ部分のみで、図中の信号もすべてフレームワーク中で自動的に実現され、ST提供者はごく簡易に記述されたST以外意識する必要がない。SE提供者はSEまた

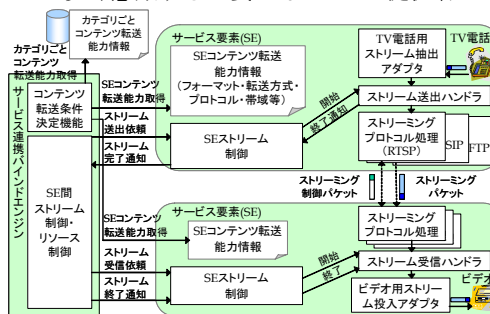


図7 ストリーム指向合成の実現イメージ
はどこかにその能力情報の記述が必要である。

5. 他の研究との比較

STでサービス合成を行うものとして、他にSTONE[2]がある。STONEは、SEにInput/mp3/UDP/IP/Ethのようにインタフェースで名前を付与し、それにより適合するSEを探し合成する。ただし特殊なネットワークアーキテクチャが必要である。提案手法は、WebServices, UPnP等市中技術との整合性が高く、実用性面で優れている。また、能力情報を実装と独立に規定でき汎用性が高い点やインタフェース名称付与の制約がなく自由度が高い点で優れている。

6. まとめ

記述が容易で流通性が高いストリーム指向ST記述について提案した。実装による実現性の検証が重要であり、進めていきたい。またさらにコンテンツタイプ(映像/音声)についてもSTで記述せずともフレームワークで判定する等、より一層の環境適応性を目指し検討していきたい。

7. 謝辞

本研究の一部は、平成15年度総務省「ユビキタスネットワーク認証・エージェント技術の研究開発」の研究助成によるものである。

8. 参考文献

[1] M. Takemoto, et al., "A Service-Composition and Service-Emergence Framework for Ubiquitous-Computing Environments", SAINT2004 Workshop, Jan. 2004
[2] 森川他, "STONE:環境適応型ネットワークアーキテクチャ", 信学技報, IN2001-12, May 2001.