

サービス連携のためのアーキテクチャ CONSORTS における WebSLIT

和泉憲明, 幸島明男, 車谷浩一, 中島秀之

概要

本稿では、デジタル世界のエージェントにより、実世界のユーザを支援するためのシステムのアーキテクチャとユーザインタフェースを提案する。具体的には、エージェントのタスクを細粒度の機能体として環境に配置し、ユーザを介して Web 上の部分コンテンツを受け渡すことにより、機能連携を実現する。そして、簡易な操作で Web 上の複数の部分コンテンツにかかるコピー&ペースト操作を行う仕組みとして WebSLIT を開発し、これをブラウザのプラグインとして実装した。これにより、次世代 Web のコンセプトをユビキタス計算環境に具現化させことを目指す。

WebSLIT Enabling Service Coordination with CONSORTS

Noriaki Izumi, Akio Sashima, Koichi Kurumatani and Hideyuki Nakashima

Abstract

In this paper, we propose the architecture and the user interface of a system for supporting the users in the real world by the agent in the digital world. In the concrete way, the agent's task is embedded in the environment as a functional object of fine-grained services. The user enables the service coordination by delivering the agent the partial contents on the Web. As an implementation, we have developed WebSLIT, which enables us to perform copy & paste action by simple operation, and that is embedded in the Web browser Mozilla as a plug-in module. Based on WebSLIT, we try to clarify the concept of the Web tomorrow in the Ubiquitous computing environment.

1. はじめに

WWW の発展系として期待されているセマンティック Web [1] は、インターネット網の急速な整備と電子商取引の普及にともない、より現実的なアプリケーションが検討されている。そして、ユビキタス計算環境や Web サービスなどに牽引される形で、Web の利用目的は、単なる情報流通基盤からサービス流通基盤へと移行しつつある。

上記のような期待の元に、近年、セマンティック Web とユビキタス計算の融合に関するパラダイムが提唱されている。具体的には、物理世界に埋め込まれた単位サービスを、ユーザの利用意図に沿って有機的に利用するための計算環境とその実現に関する意味的運用システムの検討がなされている。

以上のような観点から、近年、サービス連携の基盤として、いくつかの枠組みが提案されている。なかでも、Web サービスは、次世代 Web

の動的側面を支える分散計算環境の基盤として多方面で注目されている。

現時点で Web サービスは、米国 MS(マイクロソフト)社の .NET (ドットネット) や IBM 社を中心とする J2EE (JAVA 2 Enterprise Edition) 関連製品の共通仕様としての側面が強く、ユーザレベルの高次なサービス連携などは乖離したものである。

これに対して、次世代 Web に階層的な概念構造を導入する枠組みとして、米国国防総省 DARPA を中心としたセマンティック Web の関連プロジェクトとして、DAML-S が注目されているが、連携させる具体的な対象概念はまだ明確化されていない。

一方、サービス利用に冠する現実的な側面を検討すると、インターネット上の e-コマースやデジタル家電の協調利用のようなエンドユーザに直結した活動を支援するためには、単位サービスとしてのタスクを連携させる際に、ユ

所属: 産業技術総合研究所 サイバーアシスト研究センター/科学技術振興機構 CREST
Cyber Assist Research Center, AIST / CREST, JST
Email: niz@ni.aist.go.jp, sashima@carc.aist.go.jp, k.kurumatan@aist.go.jp, h.nakashima@aist.go.jp

ユーザの意図をいかに反映するかが重要となる。

一方、著者らは、最近、Web などのデジタル世界とユビキタス計算とのフレームワークを統合化する考えとして、位置に基づくコミュニケーションを提案している。このフレームワークでは、サイバー空間をデジタル世界と実世界の境界と位置づけ、実世界のユーザとデジタル世界のエージェントとでさまざまな資源を共有し、両者の活動の協調を目指している。

この考え方を Web 上のサービスとコンテンツに特化させると、仮想空間である Web 上のサービスに自由にアクセスするのではなく、特定の情報機器のインタフェースが物理的ななどの位置からアクセス可能で、どういう範囲でサービス受諾できるかを設計することが重要という考えに我々は至った。

そこで、本研究では、サービス連携のミドルウェアとして、先行研究であるマルチエージェントアーキテクチャである CONSORTS[3]を採用し、このエージェントが物理的な空間に表出する情報機器とのインタフェースを介して Web 上のサービスを動的に連携させる仕組みの確立を目指す[2]。

具体的には、サービス機器の機能を Web 上の CGI として実現し、CGI の表出を機能提供する機器のインタフェースと位置付け、CGI の入出力として Web 上のコンテンツの部分構造を自由にドラッグ&ドロップできるような意味処理機能を開発することにより、ユーザの意図を反映させたサービス機器の連携を試みる。

本研究は、次世代 Web のコンセプトを、マルチエージェントを基礎とするユビキタス計算環境として具現化させるもので、次世代 Web の情報家電などのインタフェースへの応用可能性を明確化できるものと期待できる。

2. ユビキタス計算を意味的に運用するための課題

デジタル世界と実世界のリンクに関する課題

デジタル世界および実世界をリンクするためには、セマンティック Web のような意味づけされた Web 上のリソースと、ユビキタス計算環境におけるさまざまなデバイス群とを、相互に運用できなければならない。

このためには、物理的環境に関する情報が、環境に埋め込まれたさまざまなデジタル ID を介して、適切にセンサとエフェクターにグラウ

ディングされなければならない。例えば、ユーザは、氏名だけでなく、さまざまな ID を有するが、これらは、ユビキタス計算のための RF-ID により提供されるかもしれないし、セマンティック Web とエージェントのための URI として提供されるかもしれない。

ユーザとエージェントの協調に関する課題

ユーザとエージェントが協調して実問題を解決するためには、サイバー空間のサービスは、両者に対して同様にアクセスできなければならない。

このことを Web 上のリソース群に特化させると、Web サービスサーバ上に構成されるサービスは、ユーザ向けのインタフェースとして HTML で定義された CGI サービスと、ソフトウェア(自動処理)向けのインタフェースとして WSDL で定義された Web サービスを同時に備えていなければならない。この考えを汎用的に実現するために、異種の API を同時に備えるサービス提供主体をエージェントとして構成し、エージェント間には FIPA-ACL を基礎として共通プロトコルをサポートするものとする。

サービス連携に関する動的要素と静的要素

ユビキタス計算の環境下でのサービス連携を考えると、そこで連携対象となるサービスは、デジタル機器や情報家電など、機器に強く関連したもののサービス連携が不可欠である。これらのサービス機器の連携にユーザの意図を柔軟に反映させるとき、たとえある物理的空間からアクセス可能なサービスは有限であっても、そのサービスの組み合わせは無限に考えられる。例えば、テレビの番組表のような情報にアクセス可能な場合に、テレビとビデオの連携を考える。このとき、番組表やテレビの予告、現在の番組など、さまざまなメディアを入力源として、番組の予約のパターンは無限に考えられる。このため、現時点の Web サービスの連携のような固定の連携パターンでは、ユーザの意図を完全に反映させることは難しい。

環境に埋め込まれた機器の能力とユーザが期待する能力

「サービス」とは一般に提供者の視点からの技術用語であるが、ユーザの意図はサービスの

合成だけで完全には達成されない。例えば、プリンタは紙を印刷することはできるが、紙を手取るのはユーザである。また、モニターはさまざまなメディアを表示することができるが、実際にそのメディアを見るかどうかはユーザが決める。このことは、定型的なサービス連携を安易にユーザに提案することは、ユーザの意図を制限することを意味する。したがって、環境側が提供するサービスは、できるかぎり、単位独立したものであり、かつ、ユーザ側で自由にかつ容易にサービス群を連携可能であることが、ユーザの意図を反映させるための要件である。

3. 埋め込み型意味処理インタフェースの設計と実装

前節の課題に対処するためには、ユビキタス環境である実世界に埋め込まれた任意構造のオブジェクトを、ユーザが置かれたコンテキストに応じて自由に選択し、任意の機器と自由にかつ簡単に受け渡しできることが必要である。

本稿では、ユーザの意図に応じてサービス機器を簡易に操作し、連携させるために、Web上の複数の任意構造のオブジェクトを容易に受け渡しさせるための仕組みを設計し、開発する。具体的には、ユーザの意図に応じてWeb上のリソースを部分的に抽出し、さまざまなサービスに仮想的にドラッグ&ドロップするための仕組みとして、WebSLITのアーキテクチャとその設計を述べる。

ここでは、簡単のために、Webブラウザ上での操作に特化するが、仕組みそのものは、汎用的に環境に埋め込み可能なものである。

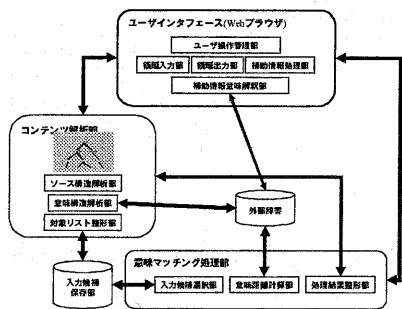


図1 WebSLITアーキテクチャ

WebSLIT アーキテクチャの概要

WebSLITは、Web上のリソースの部分オブジェクトを媒介として、Webアクセス可能なサービスの連携をユーザ駆動で実行するための統合インタフェースである。WebSLITアーキテクチャは図1に示すとおり、次のサブコンポーネントから構成される。

- ユーザインタフェース
- コンテンツ解析部
- 意味マッチング処理部
- 外部辞書
- 入力候補保存部

図1に示すように、ユーザインタフェースは、Webコンテンツを提示するためのブラウザ機能をフレームワークとして、ユーザ操作管理部や領域入力部、領域出力部、補助情報処理部、補助情報意味解釈部のサブモジュールがフレームワークにプラグインされる形で動作するものである。WebSLITは、Web上のコンテンツに特化するため、このフレームワークをMozillaとしている。

コンテンツ解析部は、MozillaのDom InspectorをコアとしてWebコンテンツをDOMツリー形式に解析するソース構造解析部、外部辞書を参照してDOMツリーのプリミティブに意味情報を付加する意味構造解析部、意味情報と操作情報を付加したソースとして整形する対象リスト整形部から構成される。対象リストとして整形されたWebコンテンツは、入力候補保存部に保存され、適宜、意味マッチング処理部から読み出される。

意味マッチング処理部は、入力候補保存部に保存されたマッチング処理の候補を選択する入力候補選択部、外部情報を参照しつつソースを構成するプリミティブ間の意味距離を計算する意味距離計算部、処理結果を整形する処理結果整形部からなる。

外部辞書は、DOMノードとして抽出された文字列に意味情報を付加するための概念辞書や同義語辞書、概念間の意味距離を定義するための概念階層などから構成される。

入力候補保存部は、意味情報と操作情報が付加して整形されたDOMツリーを保存する部分で、単一ホスト内での動作の場合はOSのカットバッファを利用するが、一般には、Webアクセス可能なSQLサーバ上に構成される。

WebSLIT の意味処理プロセス

ここでは、図 1 のモジュール構成に従い、WebSLIT が簡易な操作で任意の Web 上のオブジェクトをドラッグ&ドロップする手法について述べる。ここでは、任意構造のオブジェクトは HTML の形態でフォーマットされるものとし、ユーザに対してブラウザを介して提示されるものとする。ただし、HTML のフォーマットは、自由構造を許している。

図 2 は、図 1 の構成の WebSLIT が任意構造オブジェクトを意味処理するプロセスを模式化したものである。ここでは、ブラウザ上の部分コンテンツをユーザが選択し、これを環境に埋め込まれたサービス機器のインタフェースとしてのエージェントへ仮想的にドラッグ&ドロップするプロセスを示している。

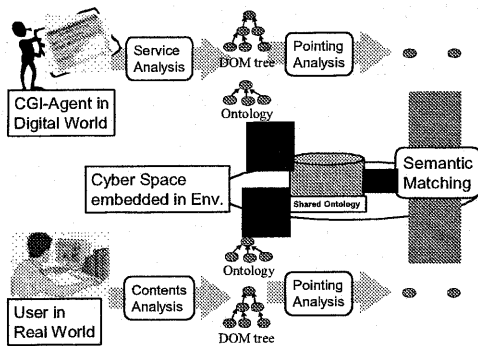


図 2 WebSLIT における意味処理プロセス

ここで、例として、Web 上のリソースとして街角の広告や案内を考え、が内部的に HTML にて整形されていると仮定する。また、その部分コンテンツをドラッグ&ドロップする対象のサービス機器として、カーナビゲーションシステムなどを考える。Web コンテンツは図 3 のようなものを仮定し、サービス機器のインタフェースとして、図 4 のような FORM タグを利用した HTML をインタフェースとしてもつ CGI を考える。

以下、図 2 のプロセスに従い説明する。まず、コンテンツ解析のプロセスとして、図 3 のようにブラウザを介してユーザに対して提示された Web コンテンツは、ブラウザとしてのインタフェースからコンテンツ解析部へ送られ、そこで、ソース構造解析部にて DOM ツリーが生成される。ここでは、図 5 の DOM Inspector が用いられ、図 3 のソースは、図 6 のような DOM ツリーの木構造形式で解析される。

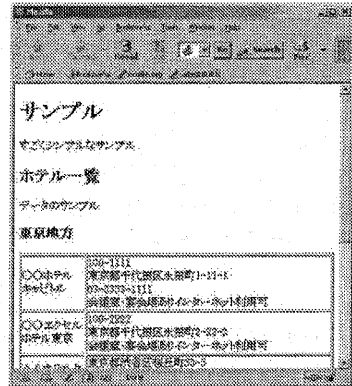


図 3 Web コンテンツの例

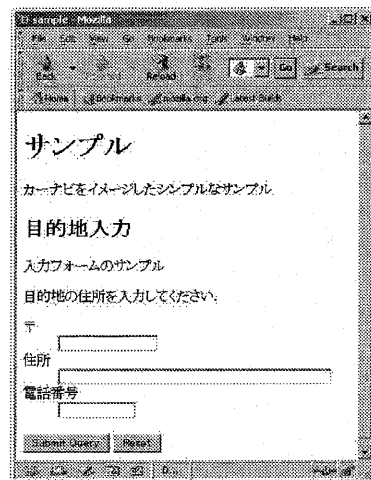


図 4 機器インタフェースの例

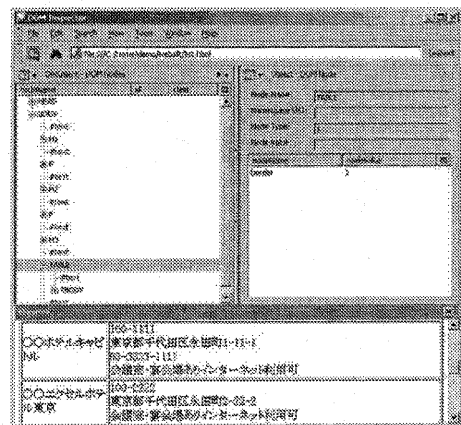


図 5 DOM Inspector による構造解析

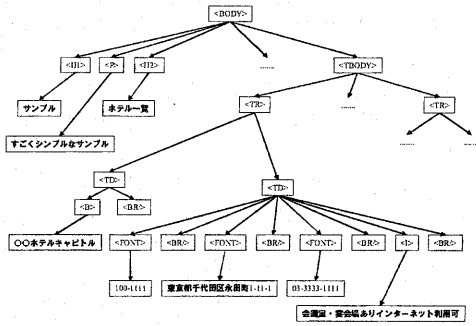


図 6 Web コンテンツの解析結果

ソース構造解析部では、同時に、外部辞書を参照し、DOM ツリーのプリミティブに意味解釈の結果を付加する。図 3 のコンテンツは、図 6 の解析結果を経て、図 7 のように意味解釈される。

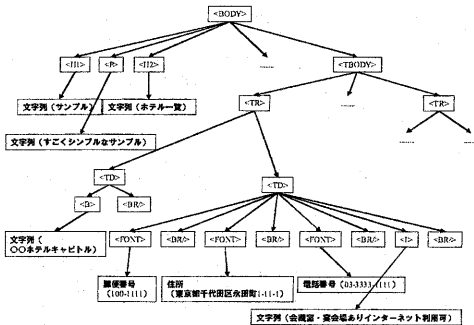


図 7 Web コンテンツの意味解析結果

次に、指示解析プロセスとして、解析結果の DOM ツリーはユーザインタフェースに部に戻され、ユーザ操作管理部にて、ユーザの指示部分を領域入力部にて DOM ツリー上に照合し、該当する Web コンテンツの部分と同定し、領域出力部にてその部分コンテンツをハイライトさせ、ユーザに提示する。ここで、マウス操作や音声操作などによりコピー操作およびドラッグ操作がなされた場合、この操作は提示部分に対してなされたものと考え、その操作意味は、物理的な操作を管理する補助情報処理部とその操作(特に発話など)の意味解釈を行う補助情報意味処理部にて解析される。そして、対象リスト整形部にて意味解析情報を付加された形に整形された後、操作意味に従って入力

候補保存部にストアされる。

次に、サービス解析プロセスとして、サービス機器のインタフェースも Web コンテンツ同様に DOM Inspector にて解析され、図 4 の機器インタフェースは、図 8 のように解析され、続いて、指示解析プロセスとして、同様に入力対象部分が同定されハイライトされる。

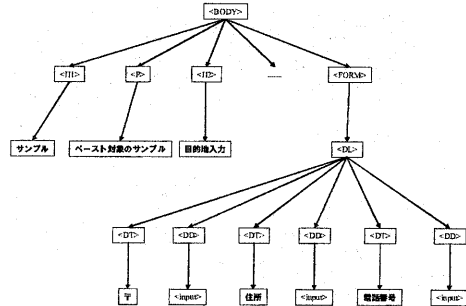


図 8 機器インタフェースの解析結果

さらに、Web コンテンツ処理プロセス同様、ユーザによるドラッグ操作またはペースト操作がなされた場合、まず、メニュー指示や音声指示などによるユーザの指示情報を補助情報処理部にて処理し、これを受けて、入力候補選択部にてその操作対象オブジェクトを同定する。この後、具体的な操作行為を操作情報処理部にて判定し、これを、部分オブジェクト間のマッチングとして外部辞書を参照しつつ意味距離計算部にて概念辞書上のノード距離を計算する。対象としている DOM ツリー上のプリミティブ間に関して、対応の積算が最も小さい組み合わせ、すなわち、全体的なマッチングの尤度のもっとも高いものをマッチング結果とする。そして、この結果を処理結果整形部に渡し、HTML の形式に変換し、ユーザインタフェースにて、その結果を提示する。

以上のプロセスの結果、図 3 の部分コンテンツが選択され、図 4 の機器インタフェースの入力としてドラッグ&ドロップされる。

以上の仕組みを Web コンテンツに特化させたものである WebSLIT は、上述の通り、Mozilla のプラグインとして実装されている。ここでは、外部辞書は組み込み可能なオントロジー定義として与えられている。また、入力候

補保存部は、インターネット上の SQL サーバを参照するものとしており、WebSLIT にて得られたユーザ ID をキーとしてドラッグ&ドロップやコピー&ペーストなどの操作が行われる。図 9 は、単一画面内でのドラッグ&ドロップされた結果画面であるが、この操作は、異なる機器間でも可能である。

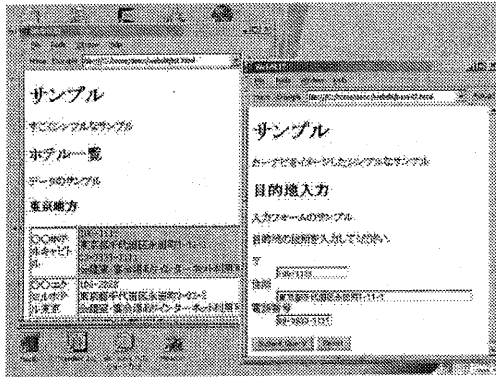


図 9 WebSLIT の動作結果

さらに、以上の操作の後、サービス実行した上で、その結果の出力が HTML にて提供されると、上記の図 2 のプロセスをユーザの利用目的に応じて繰り返すことができる。この結果、ユーザの操作意図を Web の部分コンテンツのドラッグ&ドロップとして捉えることにより、機器間の連携の管理がユーザ主導で可能となる。

4. 高度な運用への検討と課題

WebSLIT では、任意構造の複数オブジェクトとして、Web 上の部分コンテンツを考えた。そして、これを、ユーザの指示動作を解析して指定部分をハイライトさせることにより、ユーザが意図する操作対象の解釈とした。

同様に、サービス機器のインタフェースもハイライトさせることにより、操作目標も部分的に抽出することができる。このため、Web コンテンツやサービス機器などをあらかじめ固定的かつ共有された大規模な意味体系を仮定することなく、柔軟に機器間の連携が行えるようになった。

逆に、あらかじめサービスの体系がある程度想定でき、かつ、その入出力のオブジェクトの体系も想定できるのであれば、その意味解釈は、あらかじめ、属性タグを用いて HTML に埋め

込むことができる。さらに、ブラウザにて解釈されない属性タグは無視されることと規定されているため、ユーザとユビキタスサービス機器との間でのコンテンツ共有も可能となり、かつ、ユーザのブラウジング活動を邪魔しないことになる。

固定的な意味解釈タグは、閉じた体系でのサービス連携に有効で、WebSLIT では、「class = "タグ解釈"」の形式で HTML の任意の単位タグに埋め込まれた解釈を優先するものとしている。この結果、ある程度、不確実な解釈が発生するコンテンツには、意味解釈タグを埋め込むことで、より、正確なサービス機器の連携が運用可能となる。

5. おわりに

本研究では、次世代 Web のコンセプトをサービス連携環境へ応用することによりユビキタス計算環境を具現化させることを目標とした。そして、Web コンテンツの任意構造の複数オブジェクトを簡易にドラッグ&ドロップする仕組みとして WebSLIT を設計し、実装した。そして、サービス機器のインタフェースを CGI として実現した上で、CGI の入出力として Web 上のコンテンツの部分構造を自由にドラッグ&ドロップできるような意味処理機能を開発した。

これにより、ユーザの意図を反映させたサービス機器の連携が可能となった。また、WebSLIT に固定的な意味解釈を HTML に埋め込むことができ、より、ユビキタス環境での機器操作の高度な運用を検討した。

今後は、音声インタフェースなどの展開やマウス操作などをより一般の機器などに応用する予定である。

参考文献

- [1] Berners-Lee, T., Hendler, J., and Lassila, O., The Semantic Web. Scientific American, 2001.
- [2] N.Izumi, A.Sashima, K.Kurumatani, H.Nakashima: Semantic Service Coordination For Human and Agent Communities, In Proc. of the Third International Workshop on Workshop : Challenges in Open Agent Systems, (2003.7)
- [3] Sashima, A., Izumi, N., and Kurumatani, K., Location-Mediated Service Coordination in Ubiquitous Computing. In Proceedings of the Third International Workshop on Ontologies in Agent Systems, pp.39-46, 2003.