

JADE を用いた Web Services 連携エージェントの設計と実装

The Design and Implementation of JADE based Web Services Integration Agents

福田 直樹^{*1}
Naoki Fukuta

和泉 憲明^{*2}
Noriaki Izumi

山口 高平^{*1}
Takahira Yamaguchi

^{*1} 静岡大学情報学部
Faculty of Information, Shizuoka University

^{*2} 産総研サイバーアシスト研究センター
Cyber Assist Research Center, AIST

The Semantic Web technology is an emerging technology that provides a basic infrastructure for automatic integration of many resources and services on the Web. We propose the basic design and its implementation of Web services integration agents. The JADE framework, a JAVA based FIPA compliant agent framework, is widely used in the research and development of intelligent agents. We show how agents and interactions among them on the JADE framework are mapped into the corresponding integration process of services. We present an automatic code generator for such agents by using DAML-S, a DAML based service description language proposed by the Semantic Web community. In the DAML-S, services are described by three aspects: profile, process, and grounding. In the code generator, a process description in the DAML-S is transformed into the corresponding cooperation process among agents. We show a preliminary implementation of the code generator, and then clarify the differences from existing works.

1. はじめに

Web サービス(Web Services)は、アプリケーション統合の協力的な手段の1つとして注目されている。Web サービスでは、アプリケーション間の疎な結合を前提としているため、アプリケーション構築基盤を自由に選択できる利点があり、連携のためのプロトコル標準化が進んでいるため、企業を中心に導入が進んでいる。今後、企業内アプリケーション統合から企業間アプリケーション統合(EAI, Enterprise Application Integration)へ発展させる際に重要となるのが、Web サービス連携である。

複数のサービスを知的に連携させることをめざして、セマンティック Web コミュニティを中心にセマンティック Web サービス [McIlraith 2001] が提唱されている。セマンティック Web サービスでは、Web サービスの意味記述を用意することで、複数の Web サービスを知的に連携させることを目指しており、このためにサービス記述言語 DAML-S [DAML-S Coalition 2002] が開発されている。DAML-S では、1つのサービスを複数のサービスの組み合わせによって実現するための、サービスプロセス記述を用意している。

DAML-S におけるプロセス記述を利用して Web サービスを連携させるソフトウェアを実装する場合、DAML-S の記述内容を解釈し、実際に Web サービスを呼び出すためのコーディングが必要であり、開発者にとって負担となっていた。本論文では、Web サービス連携の記述言語として DAML-S に着目し、DAML-S で定義されたサービス連携を実装したエージェントのコードを自動生成するための枠組みを提案する。

2. WS 連携コード生成

2.1 DAML-S とプロセス

DAML-S では、サービスを3つの側面(プロファイル、プロセス、接地)から記述する。プロファイルでは、そのサービスが持つ静的な性質(例えば、サービスのカテゴリ、入出力およびサービス提供者に関する情報など)を記述する。プロセスでは、そのサ

ービスの実現方法を、プロセスの構成(Composition)として記述する。接地(Grounding)では、サービスを構成する個々のプロセスの Web サービスとの対応づけについて、WSDL(Web Service Description Language)を補助的に用いながら記述する。

```
<daml:Class rdf:ID="BravoAir_Process">
  <daml:subClassOf rdf:resource="sprocess;#CompositeProcess"/>
  <daml:subClassOf>
    <daml:Restriction>
      <daml:onProperty rdf:resource="sprocess;#composedOf"/>
      <daml:toClass>
        <daml:Class>
          <daml:intersectionOf rdf:parseType="daml:collection">
            <daml:Class rdf:about="process:Sequence"/>
            <daml:onProperty rdf:resource="sprocess;#components"/>
            <daml:toClass>
              <daml:Class>
                <process:listOfInstancesOf
                  rdf:parseType="daml:collection">
                  <daml:Class rdf:about="#GetDesiredFlightDetails"/>
                  <daml:Class rdf:about="#SelectAvailableFlight"/>
                  <daml:Class rdf:about="#BookFlight"/>
                </process:listOfInstancesOf>
              </daml:Class>
            </daml:toClass>
          </daml:Restriction>
        </daml:intersectionOf>
      </daml:toClass>
    </daml:Restriction>
  </daml:subClassOf>
</daml:Class>
```

図 1: DAML-S によるサービスプロセス記述例(部分)

図 1 に、DAML-S による航空券販売サービスのプロセス記述例¹の一部を示す。図 1 では、航空券販売が、GetDesiredFlightDetails, SelectAvailableFlight, および BookFlight の3つのプロセスから構成されることを示している。図 1 で定義されたサービスのプロセスを実行する場合には、これら3つのプロセスを利用する。

サービスプロセスを実際に行うには、プロセスの実行以外に、プロセスと次のプロセスの実行の間に推論やユーザとの対話を行う場合がある。例えば、航空券に関する希望の詳細を伝え予約可能な便の一覧を得るプロセスから、予約すべき便名を選択してその航空便の予約プロセスに進むためには、予約可能な便の一覧から予約すべき便を選び出す必要がある。本研究では、サービスの実行プロセス全体のひな形と、その構成要素となる Web サービスの呼び出し部分についてコードを自動生

連絡先: 福田 直樹, 静岡大学情報学部, 〒432-8011 静岡県浜松市城北 3-5-1,
Tel/Fax:053-478-1478 e-mail: fukuta@cs.inf.shizuoka.ac.jp

¹ <http://www.daml.org/services/daml-s/0.9/BravoAirProcess.daml> より一部を引用した

成する機構を用意することにより、開発者の負担を軽減することを目指す。開発者は、自動生成されたコードに推論やユーザとの対話のためのコードを追加することにより、目的のサービスを実行するソフトウェア(エージェント)の実装が可能となる。

2.2 JADE に基づくエージェントのコード生成

JADE [Bellifemine 2001] は FIPA 仕様に基づくエージェントを JAVA 言語で実装するためのフレームワークであり、エージェント開発コンペティション Agentcities 等で主要なエージェント開発フレームワークの一つとして用いられている。図 2 に、本論文で扱う Web サービス連携コードの構成(すなわち、本システムで生成したいコードのシステム構成)を示す。本コードでは、複数のエージェントを用いて Web サービス連携を実現する。具体的には、1つの Web サービスに対して1つの JADE ラッパーエージェント(ACL Wrapper)を割り当て、Web サービス連携プロセスを管理するエージェント(Coordination Agent)として別途 JADE エージェントを1つ割り当てる。Web サービス連携管理エージェントは、JADE に内蔵された FIPA-ACL 互換のエージェント間通信機構を用いてラッパーエージェントと通信し、間接的に Web サービスにアクセスすることで、複数の Web サービスを連携させたサービスプロセスを遂行する。この構成をとることにより、Web サービスアクセスのためのコードを Web サービス連携のコードから効果的に分離することができ、生成したコードの加工およびメンテナンスに役立つ物と考えられる。

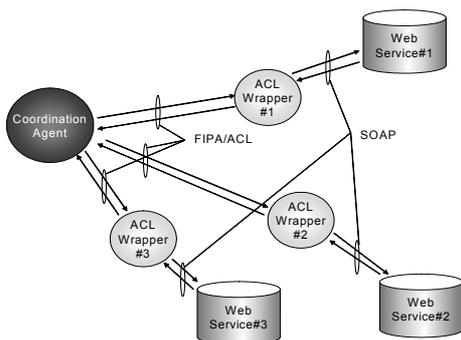


図 2: JADE エージェントによる Web サービス連携

DAML-S で記述されたサービスプロセスは、各プロセス構成要素に対応する状態を用意することで、1つの状態遷移図として記述できる。JADE では、エージェントの振る舞い(Behavior)が一連の複雑なプロセスとなる場合には、複数の振る舞いに分割し、状態を示す変数によってそれらの振る舞いを切り替えるようにコーディングを行う必要がある。本システムでは、DAML-S による記述からサービスプロセスを表現した状態遷移図を構築し、各状態における処理を振る舞いに対応させる。

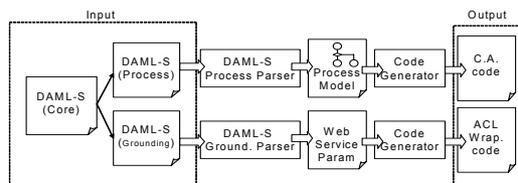


図 3: 試作システムの構成

図 3 に、本システムの構成を示す。本システムの試作には、JAVA 言語および JAVA 上で動作する論理型言語 MiLog[Fukuta 2001]を用いている。

3. 関連研究

DAML-S におけるプロセス記述ツールとして、ペトリネットに基づくプロセス記述・視覚化ツール KarmaSIM [Narayanan 2002] が提案されている。KarmaSIM では、ペトリネット理論に基づいたグラフ記述から DAML-S プロセスの生成、および生成したプロセスのシミュレーションを行うことが可能である。本研究では、DAML-S におけるプロセスの記述そのものを支援するのではなく、記述されたプロセスを実際に行うためのコード生成を支援する点で異なっている。

JADE に基づく Web サービスエージェント構築支援ツールとして WSDL Tool [SZTAKI 2003] が提案されている。WSDL Tool では、WSDL 定義から Web サービスにアクセスするための JADE コードを生成することが可能となっている。本研究では、Web サービス自身に対するアクセスよりもむしろ、複数の Web サービスを連携させるためのプロセス記述のサポートに比重を置いており、この点で本研究は WSDL Tool と異なる。

4. まとめ

本論文では、JADE エージェントによる Web サービス連携のためのコード自動生成機構の設計およびその試作について述べた。本機構では、サービス連携に関するプロセス記述を参照し、適切なサービス連携コードを生成可能である。本機構では、サービス連携のためのひな形コードを生成可能であるが、あらかじめサービス連携プロセスが与えられている必要がある。また、サービス連携プロセスを含んだ完全な形のアプリケーションを自動生成できるわけではなく、開発者によるコーディングが別途必要となっている。動的に必要なサービスを見つけ、それらのサービス連携プロセスを自動構築すること、およびサービス連携プロセス自身の記述を支援することは、本研究の扱う範囲を超える内容であるが、重要な課題である。これらの課題と本提案手法との連携については今後の課題としたい。

参考文献

- [DAML-S Coalition 2002] DAML-S Coalition: “DAML-S: Web Service Description for the Semantic Web”, In Proc of 1st International Conference on Semantic Web (ISWC2002), pp.348—363. (2002)
- [McIlraith 2001] S. A. McIlraith, T. C. Son, and H. Zeng: “Semantic Web Services”, IEEE Intelligent Systems, pp.46—53, March/April, 2001. (2001)
- [Bellifemine 2001] F. Bellifemine, A. Poggi, and G. Rimassa: “JADE: a FIPA2000 compliant agent development environment”, In Proc. of International Conference on Autonomous Agents 2001 (Agents2001), pp.216—217. (2001)
- [SZTAKI 2003] System Development Department, Computer and Automation Research Institute (SZTAKI): “Tool for Deploying Web Service Wrapper Agents”, In Proc. of 2003 Agentcities Aent Technology Exhibition, pp.44—46. (2003)
- [Narayanan 2002] S. Narayanan, and S. A. McIlraith : “Simulation, Verification and Automated Composition of Web Services”, In Proc. of 11th International World Wide Web Conference (WWW2002), pp.77—88. (2002)
- [Fukuta 2001] N. Fukuta, T. Ito, and T. Shintani: “An Approach to Building Mobile Intelligent Agents Based on Anytime Migration”, In R. Kowalczyk et, al. (Eds.) LNAI, Vol2112, pp.219—228. (2001)