

学位論文題名

PadSpace : A New Framework for the Improvisational Service Federation of Web Resources and Local Resources

(パッドスペース: ウェブ上ならびにローカルなリソースの即応的サービス
連携構築のための新フレームワーク)

学位論文内容の要旨

Currently, resources over the World Wide Web are rapidly increasing their variety and volume. Such resources include not only static HTML documents but also Web applications and Web services. The maximum reuse of these existing resources needs to coordinate them to create new complex composite services. However, it is difficult for end-users to coordinate these resources to make them work together. It requires some programming skills which most end-users do not have.

Users may need to use not only a single Web resource, but also several Web resources including not only Web services, but also Web applications, and, in some cases, to use those Web resources in combination with their own local applications. To satisfy these requirements, users need to write program codes to interoperate those various types of resources. We may conclude that users need a simple and interactive environment for interoperating these various types of resources to satisfy their needs without writing any program codes to reduce the time for achieving such interoperation, and to reuse their interoperation.

First, this thesis proposes a new software architecture “PadSpace”. A PadSpace architecture focuses on how to provide end-user tools to establish *ad hoc* service federation of Web applications, Web services and end-users’ functional resources. A PadSpace architecture assumes that all these different types of resources are wrapped into visual components called pads before they are coordinated to work together. Pads are visual object-oriented components based on the meme media architecture. A PadSpace uses the 2D (two dimensional representation) meme media system IntelligentPad as its platform. A PadSpace also uses an XML-tuplespace as its coordination model. The XML-tuplespace is a distributed coordination model that extends the Linda coordination model with the ability to carry an XML element as the values of each tuple field. We call each tuple an XML-tuple. Linda represents the interface of every available service as a tuple and stores it in its tuplespace. It also represents each service request as a tuple, and matches this requesting tuple with a compatible registered tuple representing a service. Similarly, a PadSpace represents Web applications, Web services, and end-users’ local resources as service-providing XML-tuples that only describe their IO (input-output) interfaces, and stores these XML-tuples in its XML-tuplespace.

In IntelligentPad, we represent every compound object as a pad. Therefore, we want to represent a service request also as a pad even before it is matched with a specific service, so that we can define a composite application pad using this service request pad without instantiating it to one candidate of the requested service. This service request pad, when matched with some service, works as a proxy of this service. A PadSpace represents such a service request pad as a requesting XML-tuple, and matches this XML-tuple with an appropriate service XML-tuple stored in its XML-tuplespace. It establishes a connection between this service request pad and the matched service, and makes the service request pad to work as the proxy pad of this service. Using these mechanisms, a PadSpace provides an end user tool for the service composition of Web resources and local resources. This tool enables end users to register Web resources and local application resources into the XML-tuplespace and to use those shared resources in combination with their own local resources by defining service compositions in the IntelligentPad architecture without writing any program codes.

Second, this thesis proposes a new spreadsheet-based service-coordination tool for users to orchestrate the invocation of multiple Web resources and local resources. This service-coordination tool also extends the PadSpace federation environ-

ment architecture for users to federate legacy Microsoft Office applications with arbitrary Web resources. We use Microsoft Excel as our spreadsheet-based platform for resource orchestration, and provided the user-defined orchestration function that can access Web services and associate each of their output results to any cell on an Excel sheet. Users can define complex tasks using this orchestration function in combination with other formulae defined in other cells on an Excel sheet. We also provide the mechanism to embed Web applications into the Microsoft Office applications by wrapping the formers into pads. All these mechanisms of our federation environment enable users to make legacy Microsoft Office applications work together in real time with arbitrary Web resources.

Third, this thesis proposes a new software architecture called a PadLogSpace that extends the PadSpace architecture for federating services based on semantic relationships among them, and uses a Prolog-tuplespace as its coordination model. A Prolog-tuplespace presents each service as a Prolog fact using only its IO interfaces. The IO relationships of services stored in the same Prolog-tuplespace are represented as Prolog rules. The matching mechanism of a PadLogSpace provides the one-to-one matching mechanism for Prolog facts, and the one-to-many matching mechanism for Prolog rules. A PadLogSpace allows users to represent each service request as a special requesting pad. When more than one such service requesting pads are manually combined together, the PadLogSpace immediately recognizes this composition action, and registers the corresponding Prolog fact that represents this service composition into the Prolog-tuplespace. This mechanism of a PadLogSpace enables users to define a new relationship among services stored in the same Prolog-tuplespace just by manually pasting such service requesting pads together to define this service composition.

学位論文審査の要旨

主 査 教 授 田 中 讓
副 査 教 授 原 口 誠
副 査 教 授 有 村 博 紀

学 位 論 文 題 名

PadSpace : A New Framework for the Improvisational Service Federation of Web Resources and Local Resources

(パッドスペース: ウェブ上ならびにローカルなリソースの即応的サービス
連携構築のための新フレームワーク)

近年, ウェブ上に公開されているリソースの種類と数量は急速に増加している. このようなリソースには, HTML 文書だけでなく, ウェブ・アプリケーションやウェブ・サービスも含まれる. このよう
なリソースを最大限に利用するためには, これらを相互に機能的に連携させ, 視覚的にも組み合わ
せて新しい複雑な合成サービスやツールを構築することが必要である. ユーザの PC 環境にも多種
類のツールが存在しており, これらをも連携対象として合成サービスやツールが自在に即座に構築
できることが望まれている. しかし, エンドユーにとっては, このような合成連携を構築することは
容易ではない. 近年普及しつつあるウェブマッシュアップ技術では, 連携定義の対象として扱える
サービスやウェブ・アプリケーションに制限があったり, 連携定義にプログラミングを必要する場
合があり, あらゆるウェブ上のリソースに即座に自在に連携合成を構築できる技術にまでは育って
いない.

本論文は, 知識メディア技術とタブルスペース技術を基盤技術として用い, 以下の技術を提案し確立
した.

- (1) ウェブ上に公開されている任意のウェブ・アプリケーションやウェブ・サービス, 任意の
Windows アプリケーションや知識メディア・アプリケーションを, 連携可能なサービスとして抽出
しラッピングし, タブルスペースに登録するユーザ環境を知識メディア・オブジェクトとして提供.
- (2) 登録されているサービスへの要求をインタフェースのみが定義された空の知識メディア・オブ
ジェクトとして実現し, これも部品として用いて合成アプリケーションを構築することにより, 特定
サービスとのマッチング以前に, アプリケーション開発を独立に行えるフレームワークを確立.
- (3) タブルスペースを, サービス要求とサービス提供のマッチングだけでなく, マッチングによって
形成された両者間のチャンネルを介したメッセージ交換にも用いることにより, (1) に述べたリソース
群と知識メディア部品の間の連携合成を自在に定義できる基盤連携アーキテクチャを確立. 合成ア
プリケーションの実行時に, サービス要求が呼び出されると対応する空の知識メディア・オブジェ
クトがマッチング相手のサービスへのプロキシ・オブジェクトへと自動的に変身し, マッチングで
形成されたチャンネルを用いて登録サービスやウェブ・アプリケーションとの連携動作を実行する仕
組みを確立.

本研究における成果は以下の 4 点である:

(a) ウェブ上のアプリケーションとサービス, さらには利用者のローカル環境のツールをも含めた多様なリソースを連携可能部品としてリポジトリに登録すると共に, 知識メディア・アプリケーションの開発において, 登録サービスのインタフェースのみを指定するサービス要求を部品オブジェクトとして利用可能にし, 合成アプリケーション実行時に必要に応じて個々の要求を登録サービスとマッチングし, 複雑な連携を実現する基盤アーキテクチャの提案.

(b) これら多様なリソース間の複雑な連携定義を, 知識メディア・オブジェクトの貼り合わせ操作による合成により実現し, 即興的連携定義を可能にしたこと.

(c) 多様なサービスやツールからなるリソースを連携可能対象として抽出登録する機能と, アプリケーションを部品の組み合わせ合成で開発する機能と, 合成アプリケーション中のサービス要求と登録リソースとを動的に実行時にマッチングする機能の3つを独立に分離し, リソースや, アプリケーションを独立に更新可能なアーキテクチャを提案したこと.

(d) これらすべての機能が, 簡便な直接操作のみで利用可能な形式で提供されるユーザ環境を提案したこと.

本論文の構成は以下の通りである. 2章では, マッシュアップや SOA, スプレッドシートを用いたオーケストレーション技術に関して現状を解説し, 3章では本研究で用いる基盤技術を解説し, 4章で本論文が提案する PadSpace アーキテクチャを詳述している. 5章では Window アプリケーションとの連携を含めた拡張機構の提案を詳述し, 6章では連携定義をマクロ化して規則として登録することができる PadLogSpace への拡張を詳述している.

これを要するに, 著者は, 知識メディア技術とタプルスペース技術を基盤技術として用い, (1) 多様なウェブ上のツールやサービスからなるリソースと, 利用者のローカル環境上のアプリケーションから必要な機能を任意に抽出してこれをサービスとしてラッピングして登録する技術, (2) サービス要求を知識メディア部品として利用可能にする技術, (3) サービス要求とサービスを実行時に動的に結合する技術の3つの技術を新たに確立することにより, 多様なリソース間の自在かつ即興的連携合成機構に関する新知見を得たものであり, ウェブ工学, ソフトウェア工学に対して貢献するところ大なるものがある. よって著者は, 北海道大学博士 (情報科学) の学位を授与される資格あるものと認める.