

Web サービスを提供する GUI 部品を用いた 分散アプリケーションの構築手法*

5E-1

伊藤 公人† 田中 譲‡

北海道大学 知識メディア・ラボラトリー§

1 はじめに

近年, Internet の普及により, 商用/研究用を問わず様々な Web サービスが提供されている. Web サービスとは, Internet を介し HTTP (HyperText Transfer Protocol) によってデータを入出力できるプログラムのことを呼ぶ. Google 等の検索エンジンや Mapion 等の地図画像提供サービスも Web サービスの一例である. Web サービスでは, データの表現に HTML (HyperText Markup Language) や XML (eXtensible Markup Language) を用いることで, プログラミング言語やプラットフォームに依存しない分散アプリケーションの構築が可能である.

現在, 複数の Web サービスを組み合わせる事が重要な課題とされており, XML-RPC (XML Remote Procedure Calling) や SOAP (Simple Object Access Protocol) といった Web サービスに関する規格化が進んでいる. XML-RPC や SOAP により, サーバやアプリケーションの管理者が Web サービスを合成して提供することが可能となったが, エンドユーザが Web サービスを自由に合成し, リアルタイムに分散アプリケーションを構築するためのフレームワークは完成されていない.

そこで本稿では, Web サービス自身にネットワークからアクセス可能な GUI コンポーネントとしての機能を持たせることにより, エンドユーザがそれらをデスクトップ上で自由に貼り合わせる事により分散アプリケーションを構築する手法を提案する.

2 貼り合わせによる Web サービスの 合成

本稿で提案する Web サービスの形態を図 1 に示す.

各 Web サービスは GUI を持ち, ユーザのデスクトップに表示されている. これらのサービスはローカルあるいはリモートの計算機上で動作しており, その GUI がユーザのデスクトップに描画される. ユーザは, 部品として提供される Web サービスを画面上で貼り合わせる [2] ことによって合成する. この時, 貼り合わせた二つの Web サービスが, データの連携によって合成される.

*Interactive Construction of Distributed Application with GUI Components which Provide Web Service

†Kimihiro ITO, itok@meme.hokudai.ac.jp

‡Yuzuru TANAKA, tanaka@meme.hokudai.ac.jp

§Meme Media Lab., Hokkaido Univ.

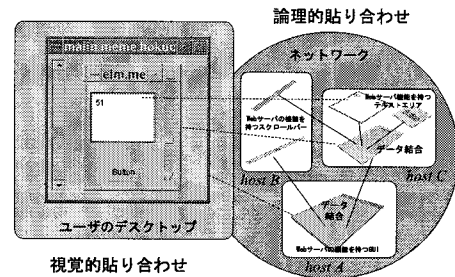


図 1: 貼り合わせによる Web サービスの合成

このように, 本フレームワークにおける各部品は, GUI を持った Web サービスであり, かつ Web サービスとしてアクセス可能な GUI である.

GUI の視覚的な位置関係を視覚的貼り合わせ, Web サービスのデータ連携の関係を論理的貼り合わせと呼ぶ.

2.1 視覚的貼り合わせ

視覚的貼り合わせでは, GUI レベルで部品を合成する. その様子を図 2 に示す.

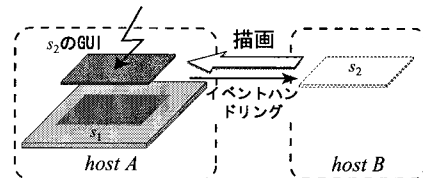


図 2: 視覚的貼り合わせ

Web サービス s_1 はホスト A 上で動作し, その GUI がローカルに描画されている. サービス s_2 は B 上で動作し, その GUI は A 上に描画されている. ホスト A の画面上で, ローカルのサービス s_1 とリモートのサービス s_2 が貼り合わせられている. すなわち, s_1 の GUI が画面上で移動すると, s_2 の GUI が付随して移動する. s_1 を s_2 の親, または, s_2 を s_1 の子と呼ぶ. 視覚的貼り合わせでは, リモート/ローカルを問わない GUI 部品のダイナミックな合成が必要である. XWindow 等のリモートへ Window 描画機能に, 画面上での移動等の GUI としての合成機能を付加することで実現する.

2.2 論理的貼り合わせ

論理的貼り合わせでは、データ連携のレベルで Web サービスを合成する。その様子を図3に示す。

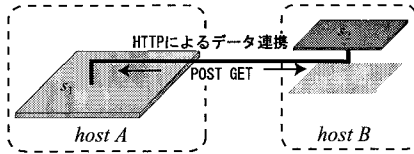


図3: 論理的貼り合わせ

Web サービス s_1 はホスト A 上で、サービス s_2 は B 上で動作している。GUI がどのホスト上に描画されているかは規定しない。

論理的貼り合わせでは、各サービス間のリモート/ローカルを問わないデータ連携を実現する。各サービスにスロットと呼ばれるデータフォルダを仮定する。スロットは XML 中の特定のタグで囲まれるデータとして実現する。Web サービス間のデータを同期させる機構として、各サービスに親子関係を与え、IntelligentPad の Set Gimme Update による手法 [2] を用いる。

以下に、Set Gimme Update メッセージの概略を示す。

Message	引数	機能	返り値
Set	スロット名, 値	子が親のスロットに値をセットする	なし
Gimme	スロット名	子が親のスロット値を要求する	値
Update	なし	親が自身の更新を子に伝える	なし

データ連携の通信プロトコルとして HTTP を用いる。HTTP の GET により Gimme を実現し、POST により Set と Update を実現する。

2.3 視覚的かつ論理的貼り合わせの実現

GUI の貼り合わせ関係と Web サービス間のデータ連携の関係が一致した状況を図4に示す。

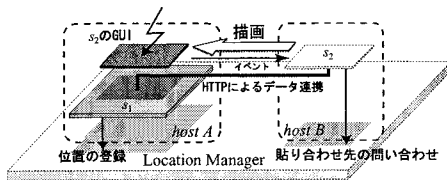
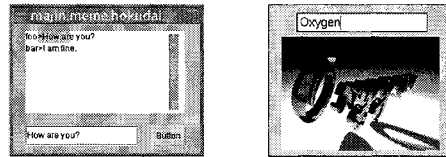


図4: 視覚的かつ論理的貼り合わせ

各ディスプレイごとに LocationManager を用意する。LocationManager は、ディレクトリサービスの一種で各サービスの GUI の位置関係を管理する。これにより、A 上でリモートサービス s_2 の GUI がローカルサービス s_1 の GUI にドロップされると、 s_1 と s_2 のスロット間にデータ連携が確立される。サービス間の親子関係は、木構造であることが条件であり、特に階層の深さに制限はしない。

3 実装とアプリケーション例

本手法により、CHIP(Corroborating Host-Independent Pads)を実装した。実装言語には、Java, C, Ruby を用い、異なる言語で実装された部品を自由に組み合わせることが可能である。CHIP による分散アプリケーションの構築例を図5に示す。



(a) チャット (b) レイトレーシング

図5: アプリケーション例

(a) は、テキスト入力サービスとテキストを保持するサービスを組み合わせで構築したチャットのシステムである。(b) では、テキスト入力、テキストの3次元レンダリングサービス、画像表示サービスを組み合わせ、テキストからロゴを作成するアプリケーションを構築した。3次元レンダリングサービスを高速な計算機上で実行させるといった分散アプリケーションの構築が、画面上での部品の貼り合わせによって可能となる。

4 おわりに

本稿では、遠隔の様々なコンピュータ上で提供されるサービスを、ユーザが自分のコンピュータ上で動作する部品と同様に組み合わせることが可能なシステムを提案した。ユーザは部品を画面上で貼り合わせることによって、リアルタイムに分散アプリケーションを構築することが可能である。

本稿では触れなかったが、同一のサービスが、異なるユーザにアクセスされる場合は、通常の Web サーバと同様にサービスがコピー (プロセスが fork) されて提供される。

今後の課題として、XMLSpace 等を用いてサービスの定式化を行うこと、サービスの検索を実現すること、ユーザの求めるサービスを既存のサービスから自動的に合成することが挙げられる。

参考文献

- [1] Didier Martin, etc. : *Professional XML*, Wrox Press Ltd. (2000)
- [2] 長崎 祥, 田中 謙: シンセティック・メディアシステム: IntelligentPad, コンピュータソフトウェア, Vol. 11, No.1, pp.36-48.(1994)
- [3] ミームメディア・アーキテクチャ IntelligentPad とその応用, 情報処理学会誌, vol.38, No.3. PP.222-231. (1997)
- [4] J. Pritchard: *COM and CORBA Side by Side*, Adison Wesley Longman, Inc. (1999)
- [5] B. Schneiderman, "Direct Manipulation: A step beyond programming languages", *IEEE Computer*, 16(8), pp.57-69 (1983)