

Web サービス機能を搭載した組み込み機器のためのアクセス管理方式 An Access Control Method for Embedded Devices which provide Web Services

炭崎竜平† 荒井兼秀† 井上勝行† 阿部一裕† 遠藤義雄†
Ryuhei Sumisaki Kanehide Arai Katsuyuki Inoue Kazuhiro Abe Yoshio Endoh

1. はじめに

近年、インターネット/イントラネットを介し、組み込み機器の情報を取得するニーズが増加している。そのニーズに対しては、Web サービス技術を用いて、組み込み機器をサーバ化することが有効である。そこで、我々は、組み込み機器に搭載可能な小型の SOAP[1]サーバである μ SOAP サーバ[2]を開発している。 μ SOAP サーバを用いることにより、フィールドネットワーク系からビジネス系のシステムまでをシームレスに統合することを目指している。

フィールドネットワーク上の組み込み機器が提供するサービスに対して、インターネット/イントラネットを介してアクセスする場合、クライアントの種類・特徴・権限などに基づいてアクセス管理を行い、各々のクライアントに対して適切なサービスを提供する必要がある。

本稿では、組み込み機器にサーバを搭載して構成したシステムにおけるアクセス管理方式を提案する。

2. 従来の方式と課題

従来から行われているアクセス管理を行う方式の一つに、各サーバ上にサービスに対するアクセス管理機能を持たせる方式がある。しかし、組み込み機器では、リソースの制限が厳しく機能の付加が困難な場合が一般的である。機能を付加できたとしても、各機器のコストが増加することで、結果的にシステム全体のコストの大幅な増加に繋がる可能性がある。また、組み込み機器は、機器ごとのリソース(CPU、メモリ、通信インターフェイス、ソフトウェア(OS、通信ミドルウェアなど)の違いが大きい。そのため、クライアントから、各機器にアクセスする時点で、各機器の呼び出しインターフェイスを知っておく必要があり、クライアント側でこれらの情報を収集、管理する機能を持たせるには、大きな開発工数を要すると思われる。

別のアクセス管理方式として、各機器の外部にアクセスを管理するための機能を設ける方式が挙げられる。例えば、Web サービスにおいては、UDDI レジストリ[3]が規定されている。しかし、UDDI レジストリでは、全てのクライアントに対し同じ内容のインターフェイス情報が提供される。クライアントが目的のサービスを検索する必要がある、サーバの障害などによりサービスが利用可能な状態である保証はできないなどの課題がある。

3. サービス利用情報の提供によるアクセス管理方式

3.1 サービス利用情報

筆者らが提案する方式では、クライアントがサーバのサービスを利用するためのサービス利用情報を提供する装置

をクライアント、サーバが存在するシステム内に設ける。この装置をサービス利用情報提供装置と呼ぶ。ここでサービス利用情報とは、サーバのアドレス情報、サーバの持つサービスの種類、各サービスを利用するための入力情報・出力情報からなる。図1にサービス利用情報の一例を示す。

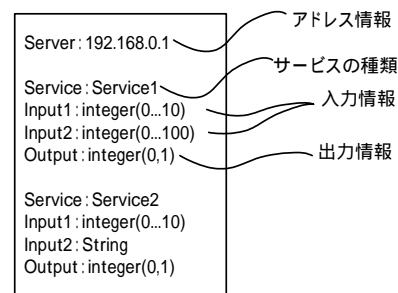


図1 サービス利用情報

3.2 サービス利用の流れ

クライアントは、サーバ上のサービスを利用する際に、サービス利用情報提供装置に対し、サービス利用情報をリクエストし、取得する。取得した情報に従って、サービスをリクエストし、サービスの結果を得る。図2にサービス利用時の流れを示す。

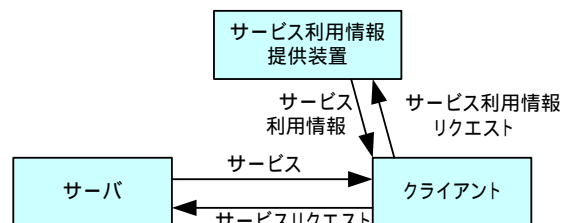


図2 サービス利用時の流れ

3.3 サービス利用情報の提供方法

本方式により、図2のサービス利用時の流れのにおいて、サービス利用情報提供装置が、クライアントに対し適したサービス利用情報を提供できる。そのためのサービス利用情報提供装置の動作を以下に示す。

(1) クライアントコンテキストの取得

サービス利用情報提供装置は、クライアントからのサービス利用情報のリクエストと共に、クライアントの種類・特徴・権限などの情報であるクライアントコンテキストを取得する。

(2) サービス利用情報の作成

サービス利用情報提供装置は、装置上にサービス利用情報の作成のために必要となるアクセス管理情報を保持している。アクセス管理情報とは、サービスのインターフェイス情報と、そのサービスへのアクセスを許可するクライア

ントコンテキストの条件を示したアクセス管理条件からなる。アクセス管理情報の一例を図3に示す。

サーバA		アクセス管理条件		
インターフェイス情報			アクセス管理条件	
サービス名	入力	出力	IPアドレス	ユーザ名
Service1	integer(0..10) integer(0..100)	integer(0,1)	192.168.0.2.. 192.168.0.254	ALL
Service2	Integer(0..100) string	integer(0,1)	192.168.0.2.. 192.168.0.254	USER1, USER2, USER3.

図3 アクセス管理情報

サービス利用情報提供装置は、取得したクライアントコンテキストとアクセス管理情報から、条件に合うサービスを選択し、そのサービスのインターフェイス情報を含むサービス利用情報を作成する。

(3)サービス利用情報の送信

サービス利用情報提供装置は、作成したサービス利用情報をクライアントに送信する。

3.4 提案方式の応用例

提案方式において、サービス利用情報提供装置の機能を拡張することにより、以下のような応用例が考えられる。

- ・クライアントコンテキストとして、位置情報を取得可能とすることで、クライアントの場所ごとに適切なサービスを割り当てることができる。
- ・サービス利用情報提供装置に、サーバの動作を確認する機能を設けることにより、その時に動作しているサーバの情報をクライアントに提供することができる。
- ・複数サーバが同じサービスを提供している場合、サービス利用情報提供装置にサーバの負荷状況を調査する機能を付加することにより、負荷の軽いサーバのサービス利用情報をクライアントに提供することができる。これにより、サーバの負荷分散の効果が期待される。この負荷分散への応用例を図4に示す。

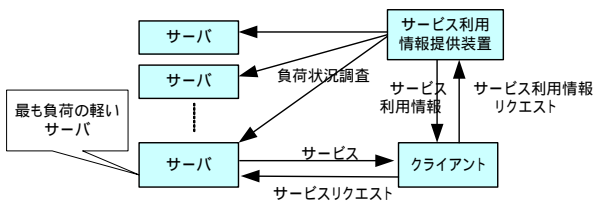


図4 負荷分散への応用例

4. 組み Web サービスシステムへの適用例

本章では、提案方式を、我々が開発しているμSOAPサーバを用いた組み Web サービスシステムに適用した場合について述べる。

提案方式におけるサービス利用情報は、Web サービスシステムにおいてサービスのインターフェイス情報を記述している WSDL[4]ファイルからも取得が可能である。サービス利用情報を提供する Web サービス利用情報提供装置を、μSOAPサーバを用いた組みサーバから構成されるシステムに適用する例を図5に示す。

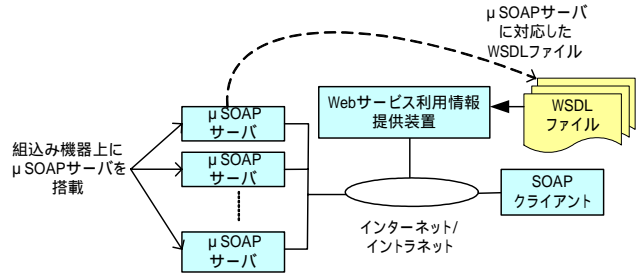


図5 μSOAPサーバを含むシステムへの適用例

4.1 UDDIレジストリとの比較

UDDIレジストリと Web サービス利用情報提供装置を比べた場合、次のような利点がある。Web サービス利用情報提供装置では、クライアントごとにサービス利用情報の内容を変化させてクライアントに提供するため、クライアントごとに適切なサービス利用情報が提供され、クライアントがサービスのインターフェイス情報を検索する手間が軽減される。また、3.4節に示すように、サーバの動作状況を調査する機能を、Web サービス利用情報提供装置上に設けることにより、サーバの故障時にも対応可能となる。

4.2 組みシステムへの適用上の利点

組み機器においては、機器ごとにいろいろな呼び出しインターフェイスが存在するが、Web サービス利用情報提供装置を設けた場合には、クライアントは、事前に組み機器のインターフェイスについての情報を知っておく必要は無く、様々なインターフェイスを持つサーバに対しても Web サービス利用情報提供装置を利用することにより、統一した方法でアクセスを行うことが可能となる。また、サービス利用情報を提供する仕組みを一ヶ所に集約することで、リソースの制限の厳しい機器に対しても、機能を付加することなくアクセス管理が可能であり、システム全体のコスト低減にも繋がる。システム内の組み機器の構成の変更がある場合にも、動作中の機器の情報を的確に提供することが可能となる。

5. おわりに

本稿では、組み機器のための、アクセス管理方式として、機器の外部に設けたサービス利用情報提供装置が、クライアントコンテキストに従って、クライアントに適したサービス利用情報を提供することでアクセスを管理する方式について述べた。

本方式によれば、システム内の各組み機器には追加の機能を付加することなく、多数の組み機器に対して統一された柔軟な方法でアクセス管理を行うことが可能となる。

参考文献

[1]W3C Recommendation: SOAP Version 1.2, 24 June 2003
 [2]阿部他:組み Web サービス構築環境 μWS-Suite, FIT2004
 [3]OASIS: UDDI Version 3.0, 19 July 2002
 [4]W3C Working Draft: Web Services Description Language (WSDL) Version 2.0, 26 March 2004