

次世代サービスプラットフォーム向け サービスネットワーキング技術

Networking Technologies for Next-Generation Service Platform

あらまし

アプリケーション、コンピュータ資源、ストレージなど様々なものをネットワークを介して利用したいというお客様のニーズに応えるため、ネットワークサービス事業者や通信事業者は、種々のサービスに共通なネットワーク機能をサービスの部品として提供するプラットフォームであるSDP（Service Delivery Platform）に着目している。富士通では、システム構築で培った技術やデータセンタ運用で蓄積したノウハウを活用し、低コストかつ迅速に高信頼のサービスを提供できるサービスプラットフォームの開発に取り組んでいる。

本稿では、このプラットフォームを実現するサービスネットワーキング技術として、サービス部品が様々な端末やアプリケーションとの間でやり取りするメッセージを中継する技術、およびサービス部品の構造化技術について述べる。これらの技術によって、サービス部品のスケーラビリティを確保し、またアプリケーション開発者が端末やネットワーク依存の複雑な処理を意識せずにアプリケーションを開発し、迅速にサービス提供できるようになる。

Abstract

We introduce two service networking technologies of a service platform: these are the service element architecture and the messaging technology. The service elements are the components of network services and are used by the applications of the service providers. The messaging technology connects the service elements to applications or terminals. These technologies improve service element scalability and enable easy service creation by application developers. To meet the needs to utilize applications, computing resources, and storage over the network, network service providers and telecommunication carriers are investigating a service delivery platform that provides not only communication services but also service elements that provide commonly used functions for network services. Fujitsu is pursuing a service platform that can deliver highly reliable network services quickly and economically by leveraging the technological edge it has gained in building systems and its accumulated expertise in operating data centers.



阿比留健一（あびる けんいち）
ITシステム研究所 所属
現在、サーバフロントのネットワーク技術の研究開発に従事。



久保田 真（くぼた まこと）
ITシステム研究所サーバネットワーキング研究部 所属
現在、サーバフロントのネットワーク技術の研究開発に従事。



雨宮宏一郎（あめみや こういちろう）
ITシステム研究所サーバネットワーキング研究部 所属
現在、サーバフロントのネットワーク技術の研究開発に従事。

まえがき

ネットワークの高速化、プロセッサの性能向上を背景に、SaaS (Software as a Service)⁽¹⁾などアプリケーションのサービス化やコンピューティングリソースの仮想化により、多様なサービスをネットワークを介して提供できるようになりつつある。

これに伴い、エンドユーザや企業ユーザは自分の利用シーンや目的に応じて、いろいろな端末で、いろいろな場所から、コンテンツやアプリケーションを利用するだけでなく、コンピュータ資源、ストレージなど必要なものを必要なときに必要なだけ利用したいというニーズを持つようになった。

このニーズに応えるため、ネットワークサービス事業者や通信事業者は、単なる通信サービスだけでなく、種々のネットワークに共通な情報の収集や配信といった機能をサービスの部品として提供するプラットフォームであるSDP (Service Delivery Platform) に着目している。

SDPはネットワーク上の様々な端末やセンサとアプリケーションサービス提供者のアプリケーションや企業の業務アプリケーションとの間に位置する。これらアプリケーションの開発者は、サービス部品を組み合わせることで、サービスや業務アプリケーションを開発することで、ネットワークサービスと情報サービスを連携させた様々な新しいサービスを迅速に提供できるようになる。

本稿では、著者らが取り組んでいるサービスプラットフォームを実現する中核技術であるサービスネットワーク技術として、ネットワークサービス部品が、種々の通信プロトコルを用い、いろいろな端末やアプリケーションとの間でやり取りするメッセージの中継技術、およびこれらサービス部品の構造化技術を紹介する。

サービスプラットフォームの課題

情報サービスにおいては、サービスを部品化して共用するアーキテクチャとしてSOA (Service Oriented Architecture)^{(2),(3)}がある。SOAでは同一センタ内または企業内ネットワーク内のサーバ間で、部品化されたサービスが利用され、それらのサービス部品はSOAP (Simple Object Access Protocol) に代表される共通化されたWebサービスインタ

フェースで接続される。

サービスプラットフォームが提供するネットワークサービスでは、SOAPに加えて、SIP (Session Initiation Protocol) などネットワーク固有のプロトコルを扱う。また多種多様な端末・センサが携帯網やインターネットなど様々なネットワーク経由でアクセスしてくる。これら様々なプロトコルを用いるデバイスやネットワークと既存のSOAの世界とをつなげることがサービスプラットフォームの特長である。

このような機能を提供するサービスプラットフォームであるが、共通利用に耐え得るプラットフォームであるためには、機能を提供するだけでなく、スケーラビリティやシンプルで使いやすいインタフェースの公開などのプラットフォーム要件を満たす必要がありそこに課題がある。

端末・アプリケーションやセンサの数が増加すると、サービスプラットフォームで提供するサービス部品の処理がボトルネックとなる。スケーラビリティ確保のためには、同じ機能のサービス部品を複数のサーバに分散配備し、その前段に負荷分散装置を配置した負荷分散構成とすることが有効である。

しかし、複数のプロトコルを扱うサービスプラットフォームのサービス部品を負荷分散構成とした場合、従来の負荷分散装置では異なるプロトコルのリクエストメッセージを関連付けて同じサーバに送信することができない。これはプロトコルに跨^{また}がってリクエストを関連付けることはアプリケーションやサービス部品が制御しており、従来の負荷分散装置ではその関連付けを知ることができないためである。

また、サービスプラットフォームのサービス部品には、SOAのサービス部品と異なり、様々なデバイスやネットワーク上の端末と通信できることが求められる。

しかし、ネットワークを介してデバイスや端末と通信するための複雑な処理をアプリケーション開発者に意識させてしまうと、その部品はアプリケーション開発者にとって使いにくいことが問題となる。例えば携帯網と家庭内LANの両方に接続することが可能な携帯端末への情報配信サービスを実現するときに、アプリケーション開発者は、端末がつながれているネットワークを意識することなくサービス

部品を呼び出し、端末およびネットワークに依存した処理をサービス部品に任せられることを期待している。

サービスプラットフォームアーキテクチャ概要

著者らが考えるサービスプラットフォームのアーキテクチャSPACE (Service Platform for Awareness, Creation and Experience)⁽⁴⁾を図-1に示す。

メッセージ中継層はサービス部品に対して負荷分散機能を提供するとともに、様々なプロトコルを解釈して、端末、アプリケーションやサービス部品にメッセージを届ける役割を担う⁽⁵⁾

サービス部品層は基本サービス部品層、抽象化サービス部品層の2階層に構造化され、ネットワークや端末に関する複雑な処理を行いつつアプリケーション開発者にとって使いやすいインタフェースを提供する役割を担う。すなわち基本サービス部品はネットワークや端末に依存するSIP、HTTPなどの通信プロトコルを終端してSOAPなどの共通なプロトコルに変換することでプロトコルの違いを吸収する。抽象化サービス部品は基本サービス部品を使って端末に対する情報の収集・配信や端末およびネッ

トワークの制御を行うとともに、端末とやり取りした膨大な情報を分析・整形することで、サービス部品の操作のしやすさ、およびサービス部品が提供する情報の扱いやすさを向上する。

管理機能は、SPACE内のサービス部品およびメッセージ中継層を管理するとともに、アプリケーション開発者に対し、サービス部品利用のための各種情報を提供する。

SPACEでは、前章で述べたサービスプラットフォームの課題を解決する仕組みをメッセージ中継層、サービス部品層、および管理機能内に設けており、以下にその技術を紹介する。

メッセージ中継技術

複数のプロトコルを扱うサービス部品を負荷分散構成要素とした場合には、従来の負荷分散装置では異なるプロトコルを用いて送受信されるメッセージを関連付けて、同じサーバに送信することができなかった。この課題を解決するためにSPACEのメッセージ中継層は、プロトコルに跨がるリクエストメッセージの関連付けをアプリケーションやサービス部品から取得するためのサービス経路制御機構、プロトコル非依存の共通情報を付与したメッセージ

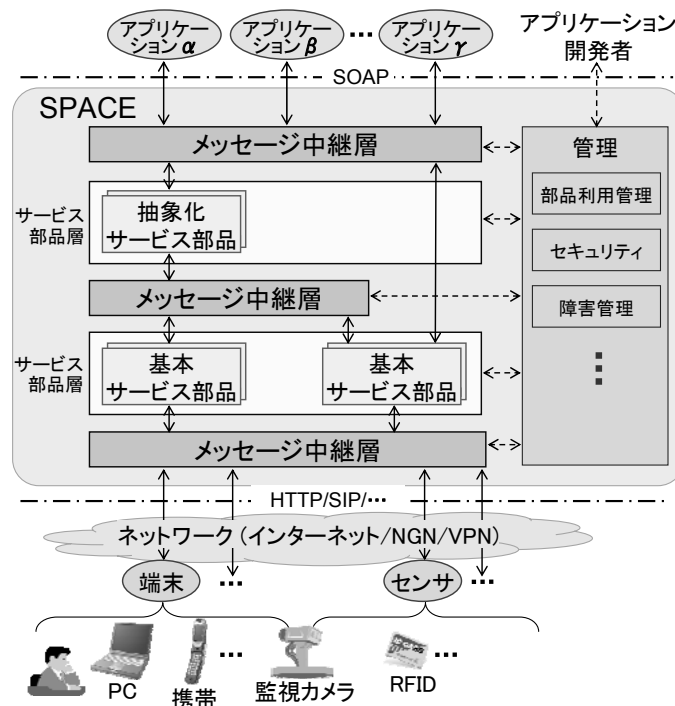


図-1 サービスプラットフォーム基本アーキテクチャ SPACE
Fig.1-Basic architecture of service platform SPACE.

を用いてメッセージの宛先を制御する宛先決定機構、様々なプロトコルのメッセージに対応してメッセージからプロトコル非依存の共通情報を生成するためのアダプタを備える。このメッセージ中継層のアーキテクチャを図-2に示す。

サービス経路制御機構は、サービス部品の扱う様々なプロトコルにおいて、端末またはアプリケーションとサービス部品間でセッションが確立したときに、サービス部品からセッションを特定するセッション識別子と、そのセッションをサービス部品が配備されたどのサーバで処理すべきかを示す情報（転送先サーバのアドレス）を収集する。それに基づいてサービス部品名、収集したセッション識別子と転送先サーバのアドレスで構成されるメッセージルーティングテーブルのエントリを生成する。

受信アダプタは受信したメッセージからその宛先に記されたサービス部品名と識別子を抽出して、宛先決定機構内で利用する共通情報を作り、メッセージに付与する。サービス部品名や識別子はプロトコルごとにその格納場所が異なる。そこで受信アダプタを前段に配置してサービス部品名および識別子の抽出というプロトコル依存の処理を吸収することにより、宛先決定機構ではサービス部品名および識別子に基づいた、プロトコルに依存しない共通化された処理を実現している。

宛先決定機構は受信アダプタにより抽出され、共

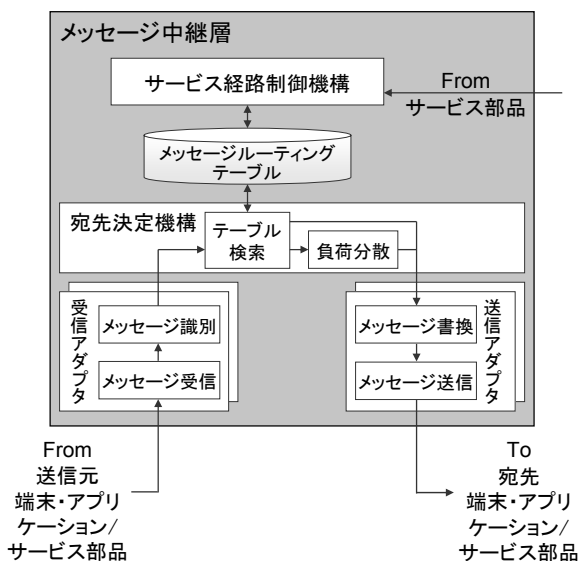


図-2 メッセージ中継層アーキテクチャ
Fig.2-Architecture of messaging function.

通情報としてメッセージに付与されたサービス部品名およびセッション識別子に基づいて、メッセージの転送先サーバを決定する。宛先決定機構のテーブル検索機能はメッセージから抽出したサービス部品名およびセッション識別子をキーとして、メッセージルーティングテーブルを参照し、転送先サーバを求める。またセッションが確立されておらず、任意のサーバに転送してよい場合は、負荷分散機能が自律振分アルゴリズムに従って、メッセージの転送先サーバを決定する。

送信アダプタは、宛先決定機構で決定したメッセージの転送先アドレスを各プロトコル固有のフォーマットに合わせてメッセージに書き戻し、送信する。送信アダプタは受信アダプタと対を成してプロトコル依存の処理を担い、必要に応じて追加・削除を可能にしているため、SPACEに收容される様々なプロトコルを用いて通信するサービスに対応したメッセージの中継が可能である。

以上の構成により、多様な構成およびプロトコルを有するサービス部品に対応しつつ、サービス部品と端末・アプリケーションの間を、アプリケーションおよびサービス部品の要求を取り込んだ上でつなぐことができる。

サービス部品構造化技術

アプリケーション開発者に対し、端末やネットワークに関する複雑な処理を意識させることのないシンプルで使いやすいサービス部品の利用インタフェースを提供するために、SPACEでは各端末をラベルにより識別される仮想的なオブジェクトとして見せるアプローチをとる。そのために、まず名前管理機能は各端末に、SPACE内で一意になるようにラベルを割り振り、アプリケーションに対して各端末のラベルを公開する。アプリケーションはサービス部品に処理を依頼する際の対象名を、このラベルで指定する。そして抽象化サービス部品は、ラベルが指し示す端末がどのネットワーク上にあり、どの基本サービス部品を使って通信するのが最適かを判断し、ネットワーク上の端末との通信を基本サービス部品に依頼する。

以上を実現するアーキテクチャを図-3に示す。

收容端末監視機能は、端末からの通知により現在自身と通信できる端末を把握し、その端末のラベル

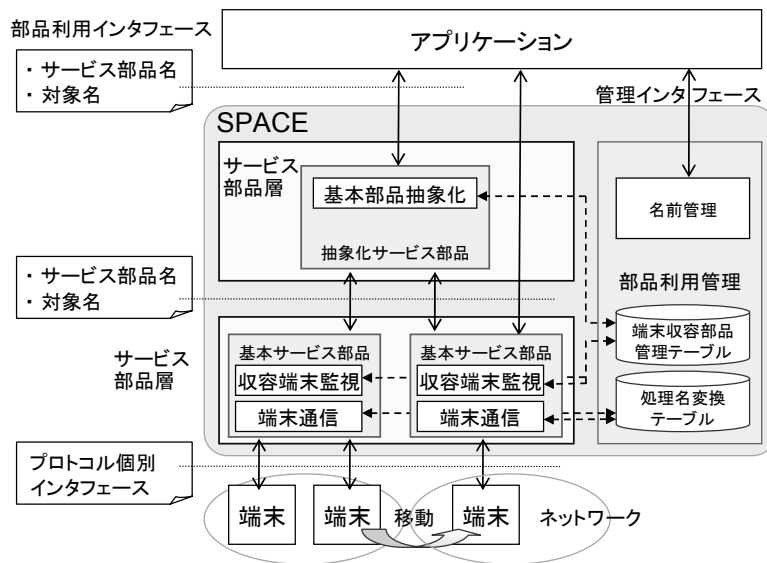


図-3 サービス部品アーキテクチャ
Fig.3-Architecture of service element.

と自身のサービス部品名の対を端末收容部品管理テーブルに時々刻々記録する。抽象化サービス部品の基本部品抽象化機能は、このテーブルを参照することでラベルが指し示す端末と通信可能な基本サービス部品を知り、さらに通信回線容量などの付随情報をもとに最適な基本サービス部品を判断して端末との通信を依頼する。この機構により、アプリケーションは、処理の対象としたい端末がどのネットワーク上にあるかに依存せずに、同じ抽象化サービス部品に、同じラベルで、処理を依頼することが可能となる。

また、基本サービス部品はネットワーク上の端末とSIPやHTTPなどのプロトコルで通信を行う。そのために端末通信機能は、対象名として指定されたラベルをSIP-URIやURLなどのプロトコル依存のアドレスに変換し、プロトコル固有のメッセージに宛先として付与した上で、端末と通信する。その際に、端末の状態に応じ、例えば端末が電源ONであればすぐ通信し電源OFFであればONになるまで情報を一時蓄積するなど、通信の仕方を適切に判断する。このように端末の状態やプロトコルに応じた処理を基本サービス部品に局所化しているため、抽象化サービス部品を使うアプリケーションに、端末の種別や状態に応じた複雑な処理や、端末の種別および対応プロトコルの追加や変更を意識させないことが可能となる。

名前管理機能は、アプリケーション開発者に対し、基本サービス部品と抽象化サービス部品のインタフェース情報（WSDLなど）やサービス部品名を公開するためのAPI（Application Program Interface）、および各端末に割り振られたラベルを公開するためのAPIを、管理インタフェースとして提供する。アプリケーション開発者は、実現したいサービスの内容に応じて基本サービス部品と抽象化サービス部品のどちらを使うかを決めた後に、管理インタフェースを使ってそのサービス部品利用のためのスタブをWSDLから生成し、また目的の端末のラベルを取得する。アプリケーションはこのスタブ経由でサービス部品名および端末のラベルを指定したメッセージを送信することで、サービス部品に対する処理の依頼が可能となる。

以上のインタフェースおよび制御機構により、アプリケーション開発者に対し、端末やネットワークに関する複雑な処理を意識させることのない、シンプルで使いやすいサービス部品を実現できる。

サービスプラットフォームの利用シーン

以上で紹介した技術を備えたSPACEを利用して提供できるサービスの典型例を紹介する。図-4に示したのは、保護者や地域が高齢者や子どもを遠隔から見守ることのできるサービスである。このサービスでは見守り対象者に携帯端末を持ってもらい、外

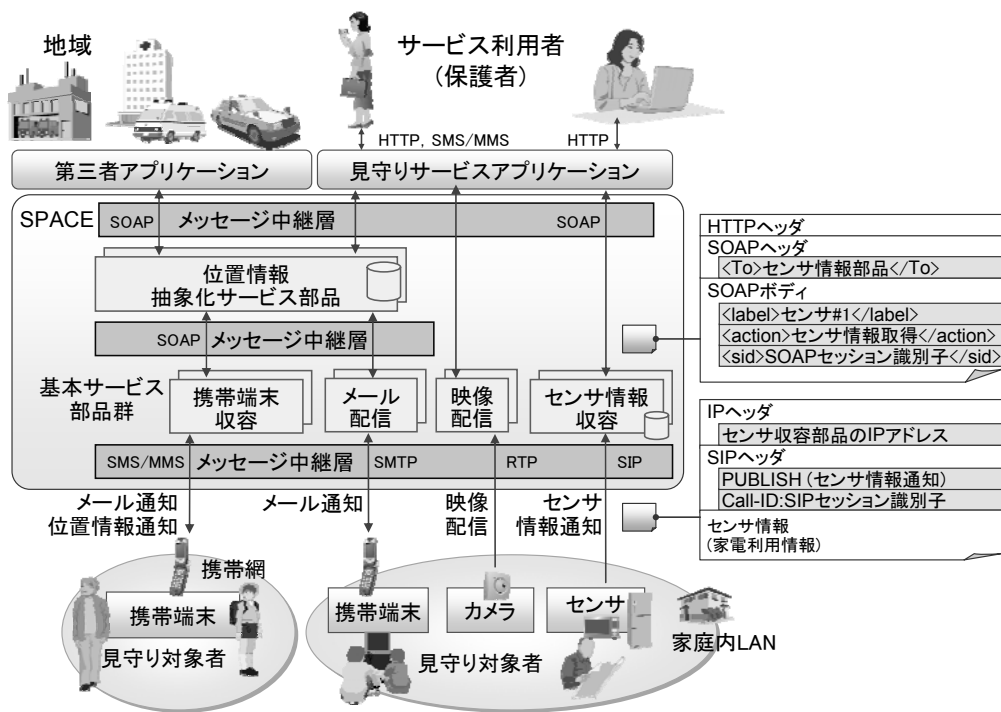


図-4 サービスプラットフォームの利用シーン
Fig.4-Affectionately watching service using service platform.

では携帯端末から、屋内では携帯端末に加えて、カメラや家電に組み込まれた各種センサからイベントとして上がってくる情報を使って対象者の状況を収集する。保護者は収集した情報を閲覧するアプリケーションにアクセスし、対象者の居場所や行動、健康状態などを知ることができる。また、緊急時には、対象者の位置情報を地域の第三者にも公開し、最寄りの人が対応に当たることもできる。

このサービスでは見守り対象者の家電利用情報がSIPを用いてセンサ情報収容サービス部品に登録され、その情報をアプリケーションがSOAP経由で取得する。センサ情報収容サービス部品は様々な対象者の多数のセンサを扱うため負荷が高く、さらにSIPとSOAPを利用するためSPACEのメッセージ中継技術が必須である。メッセージ中継層は見守り対象者の家電利用情報がSIPを用いてセンサ情報収容サービス部品に登録された際に、その部品から登録した情報を参照するためのSOAPセッション識別子(図-4中sid)と転送先サーバのアドレスを取得する。その取得情報に基づいてSIPメッセージで家電利用情報が登録されたサーバに、アプリケーションからのSOAPメッセージを届けることができる。

またこのサービスでは見守りサービスアプリケー

ションは位置情報抽象化サービス部品を使って携帯端末の位置情報を取得する。この位置情報抽象化サービス部品は、携帯端末の移動に伴い適切な基本サービス部品を判断し切り替えながら各端末から位置情報を収集する。このように構造化されたサービス部品を使うことで、アプリケーション開発者は、携帯端末が携帯網に接続されている場合と家庭内LANに接続されている場合とを区別することなく、アプリケーションを開発できる。

む す び

本稿では、サービスプラットフォームのネットワークサービス部品を利用するためにやり取りするメッセージの中継技術、およびサービス部品の構造化技術を紹介した。これらの技術は、ネットワークサービス事業者やキャリアのSDP向けだけではなく汎用性があると考えており、データセンタのクラウド化などを見据え、より複雑化し分散したアプリケーションの利用環境に耐え得るプラットフォーム実現を目指していく。

参考文献

- (1) 佐竹功二：富士通のSaaSへの取組み, FUJITSU,

- Vol.59, No.1, p.14-19 (2008).
- (2) 藪田和夫ほか：富士通が提唱するSOAの概要.
FUJITSU, Vol.57, No.6, p.564-569 (2006).
- (3) R. High, Jr. et al. : IBM SOA Foundation : An architectural introduction and overview.
<http://download.boulder.ibm.com/ibmdl/pub/software/dw/webservices/ws-soa-whitepaper.pdf>
- (4) 中条孝文ほか：サービス提供プラットフォーム技術. *FUJITSU*, Vol.60, No.4, p.369-374 (2009).
- (5) K. Matoba et al. : Architecture and key technologies of the next generation service platform. WTC 2008, 2008.