

コアテクノロジー ネットワーク・コンピューティング・サービス

河井 淳

コンピュータ技術とネットワーク技術の進化により、ついに広域でのネットワークをひとつのアプリケーションの中で扱う時代が到来した。つまり、どんな人でも広域ネットワークを利用したアプリケーションを居ながらにして開発することが可能であるし、当然のことながらその開発されたアプリケーションを即座に利用できるわけである。まさにネットワーク・コンピューティングの時代の到来である。今後多くのサービスが、広域ネットワークを介して提供されていくのは必至である。しかも今までにない高い生産性でそれらが開発され、驚くべき速さで普及していくであろう。当社もこの時代の波頭に乘るべく、新しいパラダイムのネットワーク・コンピューティング・プラットフォームと優れたサービスの提供に全力を注いでいる。

ネットワーク・コンピューティングの動向

インターネットが普及したことで、それまで専用線による専用プロトコルでのみ通信可能であった広域でのネットワーク・コンピューティングの世界が、一般大衆のレベルに急速に展開されてきた。TCP/IPという世界共通の通信プロトコルが使われ、世界中のどんなデバイスとも通信が可能となった。WWW (World Wide Web) すなわち世界中をクモの巣のように張り巡らしたTCP/IPネットワークを介して、上位プロトコルであるHTTP (Hyper Text Transfer Protocol) を使い、HTML (Hyper Text Markup Language) で記述されるWebページにアクセスすることで、テキストや静止画だけでなく、音楽や動画を含む豊かな情報 (リッチ・コンテンツ) を得ることができる。また、WebページからCGI (Common Gateway Interface) を通して、サーバ側のサービスを起動することも可能である。Webブラウザを単なる情報の表示端末から、ネットワーク・コンピューティングへのアクセス端末として利用することで、インターネット上のサーバ・クライアント・システムを容易に実現できる。今まで、航空会社の専用端末を通してしかできなかつ

たフライトの予約が、自宅で居ながらにしてできるようになった。

1996年にサンマイクロ・システムズ社の開発したJava^{TM*1)} 言語¹⁾ (以下Java) が登場し、それまでサービスに対して簡単な入力を与えることしかできなかった、クライアント端末の機能が大きく向上し、サーバとの協調処理ができるようになった。インターネット・ワイドの分散処理に向けて大いなる前進となった。

Javaを使用して、サーバはアプレットという比較的小規模のプログラムをクライアントに送りこみ、これを実行させることにより、高度なサービスを提供することができる。逆に、クライアントの与えたイベントにより起動されるサーバ側でのプログラム (サーブレットと呼ぶ) を設置することもできる。また、Javaのオブジェクト間通信機能であるRMI¹⁾ (Remote Method Invocation) を使い、双方向性の高いサーバ・クライアント型アプリケーションを提供することで、OLTP (On-Line Transaction System) のようなセッションをもつサービスが可能となった。

さらに、Java上に開発されたJini^{TM*2)} 技術²⁾ (以下Jini) を利用することで、ネットワーク上にサービスを動的に登録したり、サービスを動的に検索して実行させることができる。サービスはネットワーク上に配置されたルックアップに登録される。一方クライアントは、マルチキャストあるいはユニキャスト通信によりルックアップを発見し、目的とするサービスを検索する。サービスが見つかった場合には、そのサービスの存在するURL (Uniform Resource Locator) が得られる。同時に、そのサービスとの通信に必要なプロキシを取得し、サービスへのアクセスを行う。サービスとの通信手段が確立された後は、クライアントとサーバの間で直接通信を行い、必要なサービスを受ける。このように、Jiniにより動的にサービスの登録、検索、およびサービスへのアクセスができることから、この機能はPlug & Workと呼ばれている。

*1) JavaTMは米国サンマイクロシステムズ社の登録商標 *2) JiniTMは米国サンマイクロシステムズ社の登録商標

ネットワーク・サービスの役割

ネットワーク・コンピューティング・サービスは、広域のネットワークを介して提供されるサービスである。インターネット・サービス・プロバイダ業者（以下ISP）の登場で、それまで企業や教育・研究機関などで独自に整備していたWebサーバやメール、ニュースなどのネットワーク・アプリケーションが、ネットワーク上の基本サービスとしてISPにより提供されるようになった。つまり、多くのクライアントが頻繁に利用するサービスが、ネットワーク内に配備されるようになったととらえることができる。このように、共通性の高いサービスを、ネットワーク・サービス・コンポーネントとして広域ネットワーク上に配置することで、ネットワークの付加価値を高め、かつ利用効率を向上させる利点が生まれる。そしてこれらのサービス・コンポーネントを使用することで、サービス層のアプリケーション自体の構成も軽量化と高品質化が期待できる。

沖のネットワーク・サービス・アーキテクチャ

当社ではHA-Linux（High Availability Linux：高可用性Linux）を搭載するサーバ上に、図1に示すようなコンポーネントを開発している。Java、JiniおよびNet Liaison™技術³⁾（以下Net Liaison）により、いつでもどこでも様々なクライアント・デバイスから、様々な通信媒体を介してネットワーク上のアプリケーションにアクセスするための、総合的なネットワーク・コンピューティング環境を提供している。さらに、サービス管理などの

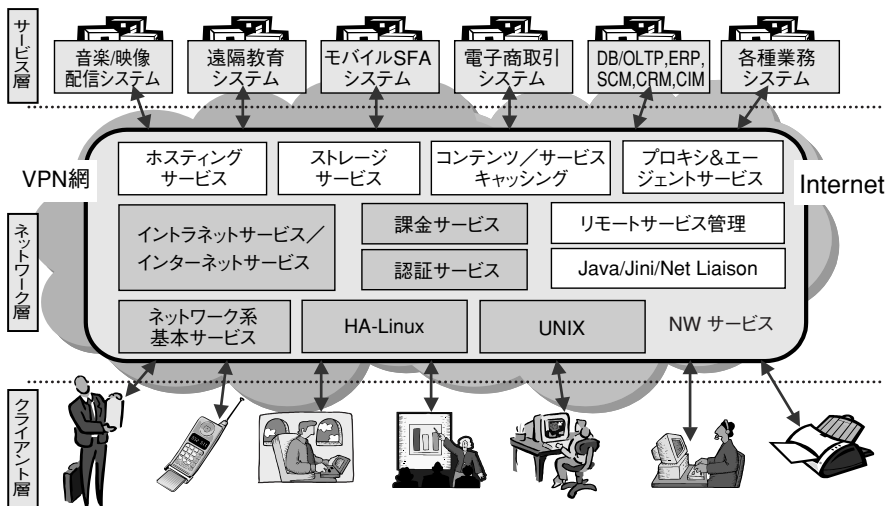


図1 沖のネットワークサービス

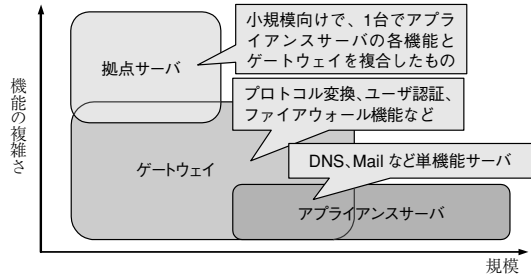


図2 Linuxサーバとゲートウェイサーバ

管理コンポーネントを用意し、通信層のみならずサービス層でのリモート監視を可能としている。

Linuxサーバとネットワーク・サーバ製品

ネットワーク・コンピューティング環境のプラットフォームとしてLinuxサーバを選択した。Linuxサーバには、OSカーネルのみならず、オープンソースながらも高品質の基本ネットワーク・サービスが整備される。

ネットワーク・サーバ製品としては、(1) SOHO (Small Office Home Office) など比較的小規模の企業を対象とした、小型ながらも一台で多くのネットワーク・サービス機能をもつ拠点サーバ、(2) ネットワークのエッジに設置し、プロトコル変換、ユーザ認証、およびファイアウォール機能などを提供するゲートウェイ・サーバ、および(3) メール、DNS (Domain Name Service)、HTTPサーバなどの単機能サービスをそれぞれ単体のサーバに割り付けた、いわゆるアプリケーション・サーバを提供する。また、複数のサーバをスタックすることで、処理

負荷に対するスケラビリティを確保している。さらにHA-Linuxを搭載するモデルを用意し、クラスタリングによる負荷分散、およびサーバ代替によるフェイル・オーバー機能を提供し、24時間x365日サービスへの適応性を向上させている。図2にこれらの製品の特徴を示す。ゲートウェイ・サーバとしては、Linuxサーバ上にNAT (Network Address Translation) やIPマスカレードなどのIPアドレス変換機能を搭載し、これによりローカルIPアドレスとグローバルIPアドレスとの間の変換機能をもたせたVPN (Virtual Private Network) ゲートウェイ・サーバ

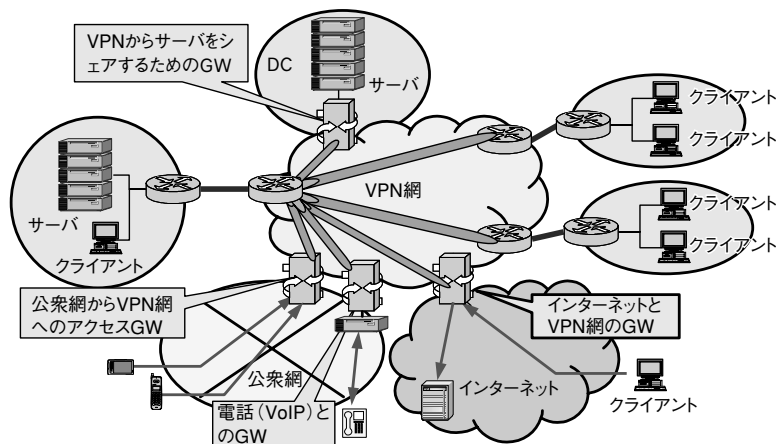


図3 各種VPNゲートウェイ

を開発した。図3に各種VPNゲートウェイ・サーバを示す。

VPNゲートウェイとして、既にインターネットとVPN網間のゲートウェイ・サーバを製品化した。現在、公衆網とVPN網間、およびVoIP（Voice over IP）網とVPN網間のゲートウェイ・サーバを開発中である。今後、IPネットワークを通信媒体とした、広域の業務アプリケーション・サービスの需要が、急速に立ち上がると予想される。当社では、これらVPNゲートウェイ・サーバにより、VPN網と他のIPネットワーク網間のセキュアな接続を提供していく。

M-NetLiaison

Linuxサーバ上にはJava, Jini, および当社で開発したNet Liaison³⁾が搭載される。Net Liaisonを使用することで、様々な通信媒体が利用される広域ネットワークでの、サービス・アクセス環境が大きく向上する。Net Liaisonは、Jiniが想定している企業内LANだけでなく、ファイアウォールを越えてのイントラネット・アクセスや、通信速度の低い電話やISDN回線、さらに、通信速度が低く接続が瞬断することの多い携帯電話やPHSなどからのサービス・アクセスに対して強みを発揮する³⁾。当社ではNet Liaisonの上にコンポーネントを配信するための技術（ICDS: Intelligent Components Delivery Service）を新たに開発した。コンポーネントはコンテンツだけでなくサービスも含む。したがって、クライアントは目的とするデータの参照だ

けでなく、そのデータを操作するためのプログラムの配信も受けることができる。当社ではこのICDSのほか、セキュリティ、データベース・アクセス、メッセージングおよびコンテンツ変換などの基本サービス・コンポーネントで構成されるM-NetLiaison（モバイル・ネット・リエゾン）を製品化している。

M-NetLiaisonによる モバイル・アプリケーション・サービス

図4にモバイル・アプリケーション・サービス方式を示す。外出先からオンラインでサービスを受ける場合には、オフィスにいるときと同様に、ノートPCやPDAなどのクライアント（以下軽量端末）から、プロキシを介してICDSを搭載するサーバ内のサービスとコンテンツにアクセスする。この時点で、オフライン時に利用する、サービスの一部は既にプロキシとして端末内にロードされる。一方、オフラインでサービスにアクセスする場合には、必要とするコンテンツを、コンテナを使用して軽量端末内に予めキャッシュしておく。これで、オフラインで必要とするサービスを独立して実行するための準備が整う。オフライン処理で更新された軽量端末内のデータは、その

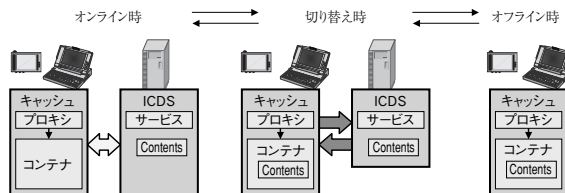


図4 モバイルアプリケーションサービス

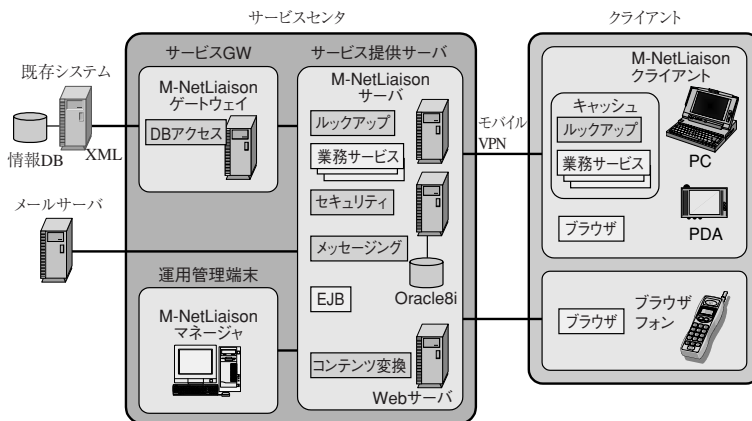


図5 モバイルサービスシステム構成例

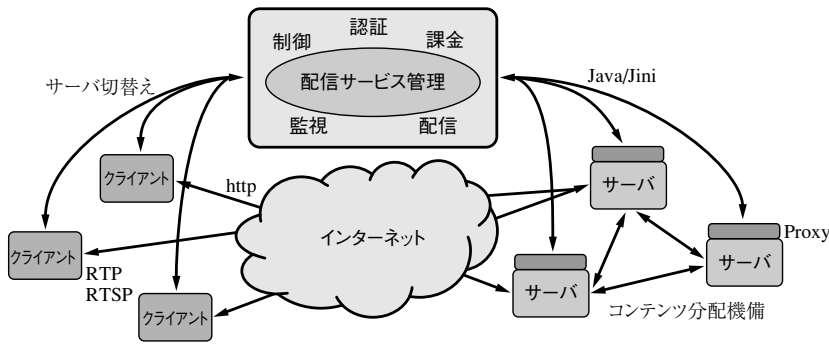


図6 コンテンツ配信サービスシステム

後のオンライン接続がされた時点でサーバに転送され、同期がとられる。

現在、さらに上位の業務アプリケーションのための、スケジュー管理、在庫管理、受発注管理、印刷サービスなど、業務支援のためのサービス・コンポーネントを開発中である。これらの業務向けコンポーネントを使用することで、ユーザはユーザ固有のビジネスロジックを容易に開発することができる。図5に業務支援コンポーネントを使用したモバイル・サービス・システムの構成例を示す。

コンテンツ配信サービス

ADSLやケーブルモデム、さらにはIMT2000などの次世代携帯電話向け通信など、アクセス系の通信帯域が着々と向上しつつある。一般ユーザにとって通信容量増加の恩恵を受けるためには、これを活かしたサービスの提供が必須である。音楽配信や映像配信など、いわゆるリッチ・コンテンツの配信が大いに期待される。当社では、ICDSを活用したサービスとしてコンテンツ配信サービスを開発中である。広域ネットワークにおける大容量データの効率良い配信を目的として、動的な最適化技術を組み込んでいる。当社は、従来方式の巨大なアーカイバを備えた拠点集中型のサーバではなく、沢山の複製（リプリケーション）あるいはキャッシュを備えた配信サーバを分散配置する、分散方式を採用している。分散配置された配信サーバに対して、最適な位置を与えるための制御機能を開発している。これはNet Liaisonを活用してJiniを広域ネットワークで利用する方式である。

図6に当社で開発中のコンテンツ配信サービス・システムの例を示す。

図で、分散配置された各配信サーバへのコンテンツの事前の複製、あるいはユーザからのアクセス要求に基づくオンデマンドの複製は、サーバどうしの連携により、コ

ンテンツ毎に独立して、平行して実行する方式を開発した。Jiniを活用することで、複製元、複製先のサーバどうしの組み合わせを、分散して自立的に最適化することが可能となった。ユーザからのアクセス要求に対しては、サーバ切り替え機構により最適な配信サーバが選択される。そしてこのような配信サービス管理は、提供するコンテンツの種類、ネットワークの混雑度、およびサービスの契約情報などに基づいて、きめ細かく行うことができる。

今後の展開

当社では、ネットワーク・コンピューティング・サービスのプラットフォームとして、Linuxを搭載するサーバを製品化している。拠点アプライアンス・サーバやSOHO向けシンサーバ、および各種VPNゲートウェイ・サーバを、基本ネットワーク・サーバとして順次提供する予定である。また、Linux上にJavaおよびJiniを搭載し、さらに広域ネットワーク上で高度なサービス・アクセスを行うためのNet Liaisonを開発済みである。さらに、Net LiaisonをベースとしたM-NetLiaison（モバイル・ネット・リエゾン）を製品化した。今後、M-NetLiaison上のモバイル・アプリケーション・サービスを構築するための業務支援コンポーネント、およびコンテンツ配信サービスを、高付加価値サービス・コンポーネントとして提供していく予定である。

当社ではこれらのコンポーネントを利用した、応用サービスの開拓も積極的に行っている。様々なサービスのユーザ提案や試作評価を行いながら、ユーザの声を積極的に取り入れて、既存プロダクツの改良や、新しいコンポーネントの開発を進めていく所存である。◆◆

参考文献

- 1) Bill Joy et. Al.: The Java Language Specification, Second Edition, Addison-Wesley Pub Co., June 2000
- 2) Ken Arnold et. Al.: The Jini™ Specification, Addison-Wesley Pub Co., June 1999
- 3) 森, 福田, 小山: ネットワークアプリケーションインタフェース「Net Liaison」, 沖電気研究開発第183号Vol.67, No.2, pp79-84, 2000

● 筆者紹介

河井 淳: Atushi Kawai. ネットワークシステムカンパニー ネットワーク・サービス事業推進センタ 技術企画部長