

汎用クラウドサービスを利用したサーバシステム

Server Systems Utilizing General-Purpose Cloud Service

宮崎 剛 MIYAZAKI, Tsuyoshi

喜多村 卓 KITAMURA, Takashi

竹内 修 TAKEUCHI, Osamu

富士電機は、汎用クラウドサービスを利用して、サービスアプリケーションによる IoT サービスの実行環境を提供するサーバシステムを開発した。さまざまなクラウドベンダが提供する汎用クラウドサービスの差異を吸収し、サービスアプリケーションのクラウドベンダ間ポータビリティを実現する機能を持つ。また、エッジコントローラと通信し、収集したデータをデータベースに格納する機能、ならびにサービスアプリケーションの実行状態やリソースの負荷状態を監視する機能を持つ。サーバシステムを利用することで、顧客価値の高い IoT サービスを高品質かつ迅速に提供する。

Fuji Electric has developed a server system that offers an execution environment for IoT services using service applications on a general-purpose cloud service. It absorbs the difference between the cloud services provided by various cloud vendors, allowing service applications to have portability between cloud vendors. Other features include communicating with edge controllers, saving gathered data on the database and monitoring the states of running service applications and resource load. Utilizing server systems allows Fuji Electric to quickly deliver high-quality IoT services that increase customer value.

1 まえがき

富士電機は、IoT（Internet of Things）サービスビジネスを普及させるため、富士電機 IoT プラットフォームを開発した。その構成を図 1 に示す。富士電機 IoT プラットフォームにおいては、お客さまが必要とするサービスを実現するためのサービスアプリケーションの実行環境を提供する部分をサーバシステムと呼び、汎用クラウドサービスを利用して構築した。汎用クラウドサービスを利用することにより、顧客規模に応じてインフラやハードウェアのリソースを柔軟に追加、削除することができる。これにより、新しい IoT サービスを提供する際にはリードタイムを短縮できるため、初期コストやランニングコストを抑え

ることができる。また、運用時の障害対応をクラウドベンダに任せるので、運用保守の手間が軽減する。さらに、クラウドベンダが提供する複数の地域でバックアップを行うことで地震や水害などに対して耐災害性の高いシステムが実現する。

本稿では、富士電機 IoT プラットフォームにおけるサーバシステムの全体像、およびサーバシステムを構成する各機能について述べる。

2 サーバシステム

富士電機 IoT プラットフォームにおけるサーバシステムでは、次に示すことが要求される。

- (a) 小規模な構成でシステム構築が容易にできること
- (b) IoT サービスの追加や接続するエッジコントローラの追加などによるリソースの拡張が容易にできること
- (c) 他社製エッジコントローラや他社提供サービスと連携したサービスを展開できること
- (d) セキュリティが確保できること
- (e) 安定した運用ができ、運用保守が容易にできること
- (f) サービスを海外顧客にも展開できること
- (g) オンプレミス^(注)で提供する既存サービスを継続して提供できること

図 2 にサーバシステムの構造を示す。これらの要求を満たすため、サーバシステムでは、クラウドベンダが提供する汎用クラウドサービス（表 1）を採用し、次に示す機能を開発した。

〈注〉 オンプレミス（on-premises）：サーバやソフトウェアなどの情報システムを使用者（通常は企業）が管理する設備内に設置・導入し、運用すること

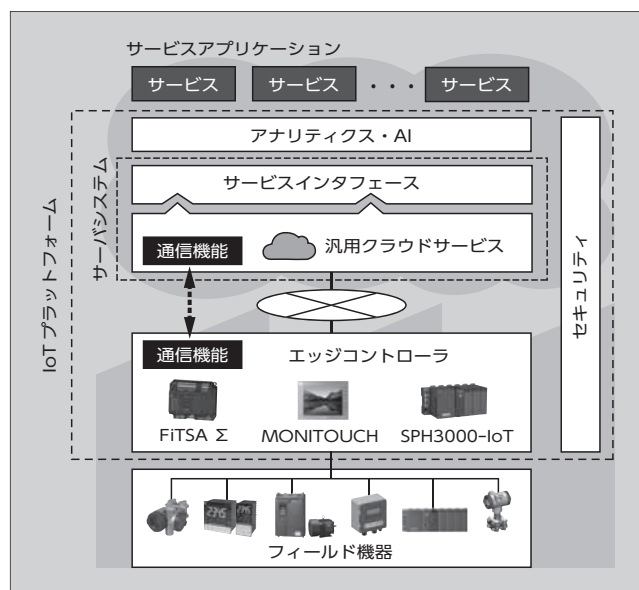


図 1 富士電機 IoT プラットフォームの構成

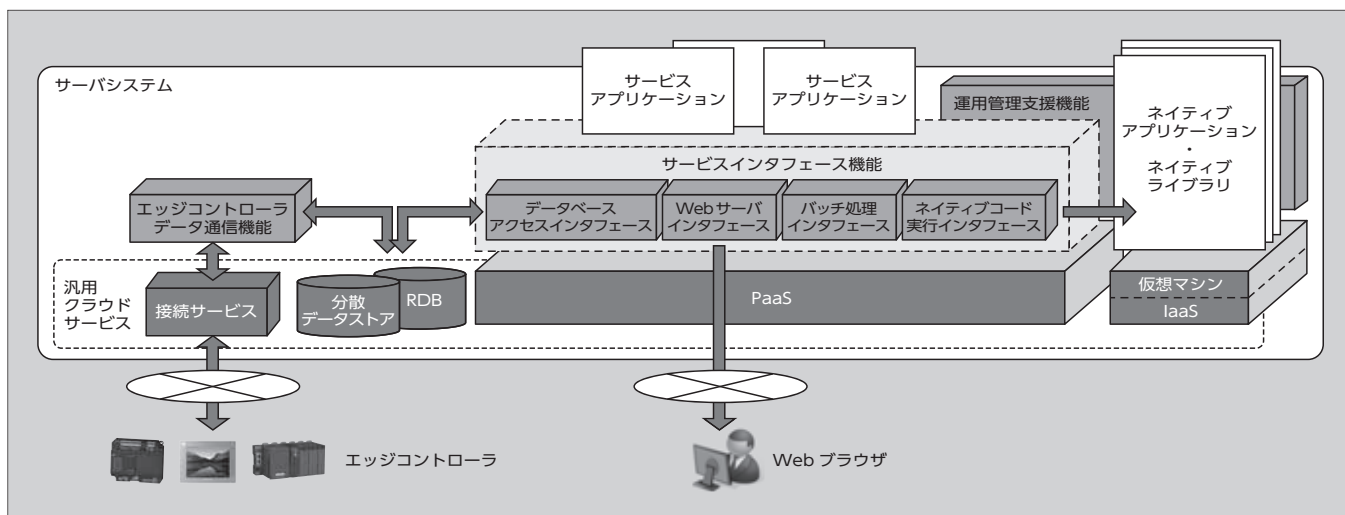


図2 サーバシステムの構造

表1 汎用クラウドサービス一覧

サービス名称	サービスの説明
RDB (Relational Database)	- 高い可用性を持ったマネージドデータベース - ディスク容量拡張などのスケールアップが可能
分散データストア	- テキストデータやバイナリデータなどを蓄積するマネージドデータベース - ディスク容量拡張やクラス追加などのスケールアップが可能
PaaS (Platform as a Service)	- サービスアプリケーションが稼動するためのハードウェアやOSなどのプラットフォーム - オープンソースのCloud Foundryを利用 - 使用するハードウェアのリソース拡張が可能
IaaS (Infrastructure as a Service)	- 情報システムの稼動に必要なハードウェアやネットワークなどのインフラ環境 - 使用するハードウェアのリソース拡張や複数のネットワーク階層に分割したセキュアな通信が可能
仮想マシン	- IaaS上でユーザが占有できるコンピュータ環境 - 使用するハードウェアのリソース拡張が可能
接続サービス	- エッジコントローラからのデータを簡単かつセキュアに収集する環境 - エッジコントローラの追加に伴うリソース拡張や収集したデータに対するアクセス制限が可能

(1) サービスインタフェース機能

サービスアプリケーションのクラウドベンダ間ポータビリティを実現し、ユーザに提供するサービスアプリケーションを効率的に開発するための共通的なインタフェースを提供する。

(2) エッジコントローラデータ通信機能

フィールドでデータ収集を担うエッジコントローラと通信し、エッジコントローラが収集したデータを分散データストアに格納する。

(3) 運用管理支援機能

サービスアプリケーションの実行状態やリソースの負荷状態を監視したり、異常発生時には異常を通知したりすることでサービス運用時の品質を向上する。

③ サービスインタフェース機能

本機能では次に示すことが要求される。

- 顧客都合により特定のクラウドベンダを指定されるケースが想定されるため、マルチベンダに対応すること
- サービスアプリケーションの追加やユーザ数の増加などに迅速に対応できること
- ユーザが安心してサービスアプリケーションを利用できること
- さまざまな Web ブラウザを使用できること
- さまざまな形式（数値、テキスト、画像、音声など）のデータを高速に扱うことができること
- 既存サービスを利用できること

これらの要求を満たすため、次に示す 4 種類のインタフェースを開発した。

3.1 データベースアクセスインタフェース

データベースアクセスインタフェースは、汎用クラウドサービスが提供する RDB (Relational Database) と分散データストアにアクセスするためのインタフェースである。

クラウドベンダが提供する汎用クラウドサービスは、クラウドベンダごとに採用する RDB や分散データストアが異なっており、データベースアクセスインタフェース内でデータベースの差異を吸収する必要がある。そこで、サービスアプリケーションからデータベースにアクセスするためのインタフェースは共通的なインタフェースとし、インタフェース内で各種データベースにアクセスする処理を個々に実装することにより、データベースの差異を吸収した。この結果、特定のクラウドベンダに依存せず、さまざまな汎用クラウドサービスの利用を可能とした。また、データベースとして、テキストデータやバイナリデータなどを蓄積することが可能な分散データストアを利用することで、さまざまな形式のデータを扱うことができる。

3.2 Web サーバインタフェース

Web サーバインタフェースは、顧客に公開する Web

アプリケーションを効率的に開発するためのインタフェースである。

Web アプリケーションを効率的に開発するために OSS (Open Source Software) で提供される Model-View-Controller フレームワークを採用し、ユーザ認証やユーザごとのアクセス制限を行うセキュリティ機能、ユーザ数の増加に対応したセッション管理機能などを開発した。この結果、ユーザは自分が利用したい Web ブラウザを使用し、安心して Web アプリケーションを利用できる。また、ユーザ数の増加にも迅速に対応できる。

3.3 バッチ処理インタフェース

バッチ処理インタフェースは、集計や帳票作成などのバッチ処理アプリケーションを効率的に開発し、実行するためのインタフェースである。

各バッチ処理を高速に実行し、かつ、複数のバッチ処理を同時に実行するためのマルチスレッド・マルチインスタンスにおける並行処理実行機能や分散処理実行機能などを開発した。この結果、新たなサービスアプリケーションを追加した場合にも、既存のサービスアプリケーションへの影響を最小限にとどめることができる。

3.4 ネイティブコード実行インタフェース

ネイティブコード実行インタフェースは、C 言語で記述されたアプリケーションやライブラリを実行するためのインタフェースである。

通常、サービスアプリケーションは PaaS (Platform as a Service) 上で実行されるため、オンプレミスで提供している C 言語で記述された既存サービスやアナリティクス・AI (Artificial Intelligence) サービスを活用することができない。そこで、本サーバシステムでは、IaaS (Infrastructure as a Service) 上の仮想マシンに配備された既存サービスやアナリティクス・AI サービスを PaaS 上のサービスアプリケーションから実行する機能を開発し、これらのサービスを実現できるようにした。

4 エッジコントローラデータ通信機能

本機能では次に示すことが要求される。

- (a) 他社製のエッジコントローラや IoT 機器とも通信できること
- (b) 接続台数が増えても、容易に通信できること
- (c) エッジコントローラとサーバシステム間で安全に通信できること
- (d) 今後の用途拡張に備えて、さまざまな形式（数値、テキスト、画像、音声など）のデータを通信できること

4.1 デファクトスタンダードプロトコルの採用

エッジコントローラとサーバシステム間の通信では、デファクトスタンダードとして、広く利用されている

MQTT (Message Queue Telemetry Transport) プロトコルを採用した。この結果、他社製のエッジコントローラや IoT 機器と通信でき、また、接続台数が増えた場合でも容易に通信できる。

4.2 接続認証

エッジコントローラが送信するデータには、機器を認証するアクセストークンが付与されている。サーバシステムでアクセストークンを検証することで、通信を許可されたエッジコントローラだけが通信できる。この結果、エッジコントローラとサーバシステム間で安全に通信できる。

4.3 分散データストアの採用

エッジコントローラから送信されたデータを分散データストアに格納する。分散データストアに格納するデータフォーマットとしては、さまざまなデータ形式に対応可能な JSON (JavaScript Object Notation) 形式を採用した。この結果、エッジコントローラとサーバシステム間で計測値や画像などさまざまなデータを通信できる。

分散データストアに格納されたデータは前述のサービスインタフェース機能を介して、サービスアプリケーションが利用できる。

5 運用管理支援機能

本機能では次に示すことが要求される。

- (a) サービスアプリケーションが安定して運用できること
- (b) 運用管理が効率的に実施できること
- (c) 異常発生時、迅速に復旧作業が実施できること
- (d) 異常発生時、他のサービスアプリケーションへの影響を防止できること

これらの要求を満たすため、表 2 に示す機能を開発した。

表 2 運用管理支援機能

機能名称	機能の説明
アプリケーション監視	○さまざまなサービスアプリケーションの生存状態やログを監視し、異常発生をリアルタイムに検知する。 ○サービスアプリケーションの安定運用、迅速な復旧作業が可能になる。
リソース監視	○PaaSやIaaS上の仮想マシンのリソース状態を監視し、リソース不足などの異常をリアルタイムに検知する。 ○サービスアプリケーションの安定運用、迅速な復旧作業、異常発生時の拡散防止が可能になる。
異常検知時メール通知	○アプリケーション監視やリソース監視で検知した異常を運用管理者にメールで通知する。 ○運用管理の効率化、迅速な復旧作業が可能になる。
運用監視ポータルサイト	○サービスアプリケーションの稼動状態を表示するポータルサイトを提供する。 ○サービスアプリケーションの安定運用、運用管理の効率化が可能になる。

⑥ あとがき

汎用クラウドサービスを利用したサーバシステムについて述べた。汎用クラウドサービスを利用することにより、顧客価値の高いIoTサービスを高品質、かつ、迅速に提供する。また、サーバシステムでは、各種汎用クラウドサービス間の差異を吸収することにより、お客さまが指定するクラウドベンダを利用できる。

今後は、さらに顧客価値の高いIoTサービスを提供するために、他社が提供するサービスアプリケーションとの連携機能や、富士電機のサービスアプリケーションを他社に提供する機能などを開発し、サーバシステムの機能拡充に取り組んでいく所存である。



喜多村 卓

公共分野向けおよびIoTシステムの開発に従事。現在、富士電機株式会社パワエレシステム事業本部開発統括部システム開発センターHMI開発部グループマネージャー。



竹内 修

公共分野向けシステムの企画・開発に従事。現在、富士電機株式会社パワエレシステム事業本部エンジニアリング統括部システム技術センター社会システム部主任。



宮崎 剛

IoTシステムの研究開発に従事。現在、富士電機株式会社技術開発本部イノベーション創出センターデジタルプラットフォーム開発室組込システム研究部主査。





＊本誌に記載されている会社名および製品名は，それぞれの会社が所有する
商標または登録商標である場合があります。