SIを必要とせずサービスを迅速に開発できる東芝IoTサービスファクトリー

Toshiba IoT Service Factory for Agile Development of Services Eliminating Need for System Integration

小寺　志保 KODERA Shiho　　中嶋　宏 NAKAJIMA Hiroshi　　杉本　信秀 SUGIMOTO Nobuhide

東芝グループは，CPS（サイバーフィジカルシステム）テクノロジーで，新たな価値の創造，社会課題の解決，持続可能な社会の実現を目指している。

CPSでは，フィジカル空間のデータをIoT（Internet of Things）技術で収集し，それらをサイバー空間のデジタル技術で分析して，フィジカル空間へフィードバックする。今回，CPSを開発・運用するための共通フレームワークである東芝IoTリファレンスアーキテクチャー（Toshiba IoT Reference Architecture，TIRAと略記）に準拠したサービスを，迅速に提供するための東芝IoTサービスファクトリー（TISF）を開発した。TISFは，CPSサービスを三つのパターンに簡略化してソフトウェア部品を最大限活用することで，従来のシステムインテグレーション（SI）を行うことなく新たなサービスを構成し，フィジカル空間の様々な変化にも迅速に対応できる。

The Toshiba Group has set the goal of becoming a company that delivers new value, contributes solutions to social issues, and helps to achieve a sustainable society through the utilization of cyber-physical system (CPS) technologies.

In CPS, large volumes of data are collected in physical space by means of Internet-of-Things (IoT) technologies, analyzed by digital technologies in cyberspace, and then fed back to physical space to promote advances and improvements. We have now developed the Toshiba IoT Service Factory (TISF) in order to swiftly provide customers with services compliant with the Toshiba IoT Reference Architecture, a common framework to promote the development and operation of CPS. TISF is a development environment that simplifies the contents of CPS services into three types of patterns and maximizes the use of software components to automatically develop new services without the need for conventional system integration (SI), making it possible to rapidly respond to various changes in physical space.

1. まえがき

CPS[(1)]の実現には，フィジカル空間の世界とサイバー空間の世界にあるシステムを融合する必要がある。従来は，各企業の業務形態に合わせてシステムを統合するSIによって，CPSのサービスを構築していた。SIは，顧客の要件にシステムがカスタマイズされており，最適なシステム構成である反面，カスタマイズによる開発期間が長いことや，システムの柔軟性・拡張性に欠けるため開発後の企業の変化やフィジカル空間の変化に柔軟に対応できないことなどの，問題がある。

そこで，東芝グループは，東芝IoTサービスファクトリー（TISF）を構築した。TISFは，リファレンスアーキテクチャーを基にCPSのサービスをパターン化し，サービスに利用するソフトウェア部品を共有・流用可能にする仕組みである。また，サービスを簡易に再構成できるユーザーインターフェース（UI）を持つ。TISFにより，サービスをSIなしで迅速に顧客に提供し，様々なフィジカル空間の変化に対応可能となる。ここでは，このサービス開発手法について述べる。

2. TIRAとサービスパターン化

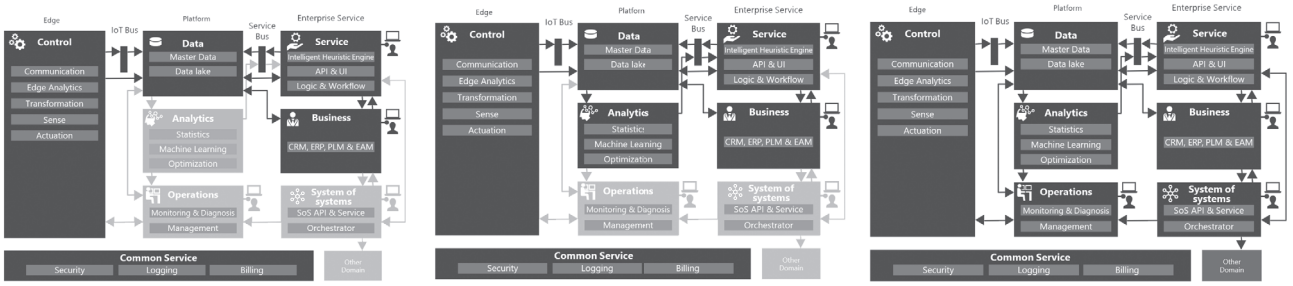
CPSのサービスは，インダストリアル用途のIoTサービスともいえる。そのサービスは，グローバルなリファレンスアーキテクチャーであるIIRA（Industrial Internet Reference Architecture）[(2)]に準拠して当社が定めたTIRA[(3)]に従って，三つのパターンに分類できる（**図1**）。

(1) パターンA：Visualization　　取得したデータの可視化システム（例：シンプルな遠隔監視サービス）

(2) パターンB：Visualization+Analytics（AI）　　パターンAに加え，AIを用いたデータ分析も実施するシステム（例：高度解析を備えた遠隔監視サービス）

(3) パターンC：Visualization+Analytics（AI）+Action　　パターンBに加え，メンテナンスや遠隔制御も実施するシステム（例：遠隔管理サービス）

CPSのサービスは，初期のパターンAの可視化システムから，AIデータ分析も行うパターンB，ほかのシステムと連携・制御を行うパターンCと徐々に進化していく。したがって，この変化に追従できるようなアーキテクチャーとすること



(1) サービスパターンA
(Visualization)

(2) サービスパターンB
(Visualization + Analytics(AI))

(3) サービスパターンC
(Visualization + Analytics(AI)+Action)

図1. マチュリティーモデルによるCPSサービスのパターン分類

CPSサービスの機能構成をマチュリティーモデルに従って分類したもので，Visualizationの構成から順に，Analytics，Actionが加わる形で進化していく。

Classified patterns of CPS services according to maturity models

表1. CPSサービスの非機能要件のレベル分類

Non-functional requirement levels according to classified patterns of CPS services

非機能要件レベル	ベーシック	アドバンスド	ハイアドバンスド
サービスの数	～10	～100	～1,000
可用性（稼働率）	70％以上（冗長性なし）	99.5％以上	99.9％以上
性能（受信リクエスト数，レイテンシー）	～15 req/s 1 s以上（P90）	～100 req/s 100 ms以上（P90）	～1,000 req/s 100 ms以下（P90）
データ格納規模	～10 Gバイト	～1 Tバイト	16 Tバイト以上
監視	リソース監視，サービスログ	左に加えて，サービスのキープアライブ，アラート通知，データ欠損検出などデータ異常検出通知	左に加えて，異常予測アラート通知
データバックアップ	なし	あり	あり

req：受信リクエスト数　　P：パーセンタイル　　T：テラ（10^{12}）

表2. CPSサービスのセキュリティーレベル分類と代表的な対策例

Security requirements and measures according to classified patterns of CPS services

セキュリティーレベル	セキュリティー対策代表例（一部抜粋）	セキュリティー要件識別子CPSF[4]（国内）	セキュリティー要件識別子NIST CSF[5]（海外）
ベーシック	認証認可，パスワード管理	CPS.SC-4, AC-1, 8	PR.AC-1
	データ暗号化	CPS.DS-2	PR.DS-1
	通信暗号化	CPS.AC-9	PR.AC-7
	ロールベースアクセスコントロール	CPS.AC-5	PR.AC-4
	データバックアップ	CPS.IP-4	ID.BE-5
	Webアプリケーションファイアウォール	CPS.DS-6, CM-1, 3	PR.DS-4, DE.CM-1, 4, 5
アドバンスド	多要素認証	CPS.AC-6	PR.AC-4，7
	侵入検知	CPS.DS-6, 9, CM-1, 3, 4	PR.DS-4, 5, DE.CM-1, 4, 5
	証明書による認証	CPS.DS-5, 11, 13, PT-3	PR.DS-6, PR.PT-5
	OTネットワークとの分離	CPS.AC-7	PR.AC-5, PR.DS-7.PR.PT-4
	管理者アカウント管理と監査	CPS.AC-1, 5, IP-1, 2	PR.AC-1, 4, PR.IP-1
ハイアドバンスド	通信路のVPN又は専用回線	CPS.AC-9, DS-3, CM-1	PR.AC-7, PR.DS-2, DE.CM-1
	トラステッドモジュールによる暗号化（TPM）	CPS.AC-3, 6, DS-2, 4, 5, 8	PR.AC-3, 4, 7, PR.DS-1, 2, 5

OT：Operational Technology　　VPN：Virtual Private Network
TPM：Trusted Platform Module

が重要である。

非機能要件についても，**表1**に示すように，サービスの規模も踏まえて3段階のレベルごとに項目を分類した。非機能要件は項目数が多く，組み合わせにより複雑化しやすいが，三つのレベルに分類することで簡略化を実現した。

セキュリティーについては，国内向けにCPSF（サイバー・フィジカル・セキュリティ対策フレームワーク）[4]，海外向けにNIST（米国国立標準技術研究所）CSF（Cybersecurity Framework）[5]のセキュリティー要件を基に，対策例として，レベルごとに必要な項目を定めた。代表的な対策例一覧を**表2**に示す。

これらのパターン化によって，実現したいサービスと規模が分かれば，必要な機能と構成，設定すべきセキュリティー項目，非機能要件を比較的容易に決定でき，サービスごとに品質が異なるおそれがなくなる。更に，パターンAからパターンCをそれぞれ部品化して共有することで，開発期間を短縮し，素早いサービスの提供を実現できる。

3. TISFによるサービス構築

サービスパターンから実際のサービス構築を行うまでの流れを説明する。

図2に，実際にTISFで構築するサービスを示す。TISFでは，サービスをツリー構造で表現する。ツリー構造は，サービスの各機能を構築するレシピと，機能を実行するコンポーネントであるコンテナ（ソフトウェアを，必要なライブラリーや設定などとパッケージ化したもの）で構成される。サービ

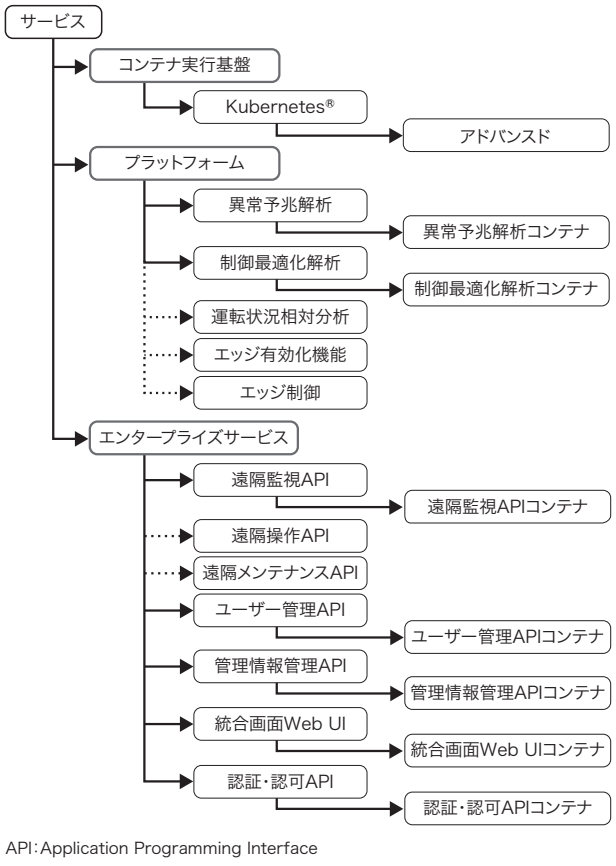
ス構築時には，ツリー構造の上から下に向かって順番に参照し，各機能を構築していく。通常，ツリー構造の最上位にはコンテナを実行できる基盤（例えばKubernetes®）を置き，その下に構築したいサービスの機能を構成するレシピやコンポーネントを追加していく。図2では，図1のパターンBを構築する際に選択する項目を実線の矢印で示している。コンテナ実行基盤のレシピには表2の非機能要件・セキュリティーのレシピをひも付ける（図2ではアドバンスド）。サービスをカスタマイズする場合は，必要なレシピやコンポーネントを入れ替えればよい。

TISFでは，各機能の自動構築をレシピとコンポーネントで行う。レシピは，Infrastructure as Code（IaC）と呼ばれるコードを記述して，ソフトウェア的にクラウドサービスの機能，設定を構成する方法を用いる。代表的なIaCには，複数のクラウドサービス環境に対応したTerraform®があるが，スクリプトやほかの言語も使用できる。コンポーネント（コンテナ）は，構成情報が記述されたレシピ（例えばKubernetes®のManifestファイル）の下に置く。これは，コンポーネントがレシピの構成情報に従い，指定したクラウドサービス環境に自動的に配置されることを示す。

図3にTISFの機能構成を示す。TISFは，図2に示したサービス構成ツリーを編集・作成する機能（サービス編集機能），サービスを構成するレシピ，コンポーネントを管理する機能（部品管理機能），レシピを登録するソースコードリポジトリー，コンポーネントを登録するコンテナレジストリー，レシピやコンポーネントとその関連を定義するためのデータベース，サービス構成ツリーを参照しながらクラウドサービス環境に対してレシピを順に実行してサービスを構築する機能（構築機能）などで構成されている。

まず，開発者はIaCスクリプトなどをソースコードリポジトリー，コンテナをコンテナレジストリーに登録する。次に，サービス編集機能を用いてレシピやコンポーネントを登録する。登録したレシピやコンポーネントをそれぞれ組み合わせ，各機能のツリー構造を作成する。作成したレシピとコンポーネントのツリー構造は部品管理機能で管理し，データベースに格納され，利用者間で共有される。

一方，サービス提供者は，サービス編集機能を用いて開発者が登録したツリー構造を組み合わせ，新たなサービスのツリー構造を作成する。作成したツリー構造は構築機能を用いてクラウドサービス環境に構築される。構築機能は，



API：Application Programming Interface

図2. サービスのツリー構造

TISFは，サービスの各機能を構築するレシピと機能を実行するコンテナで構成されたツリー構造で，サービスを表現する。

Tree structure of service constructed by TISF

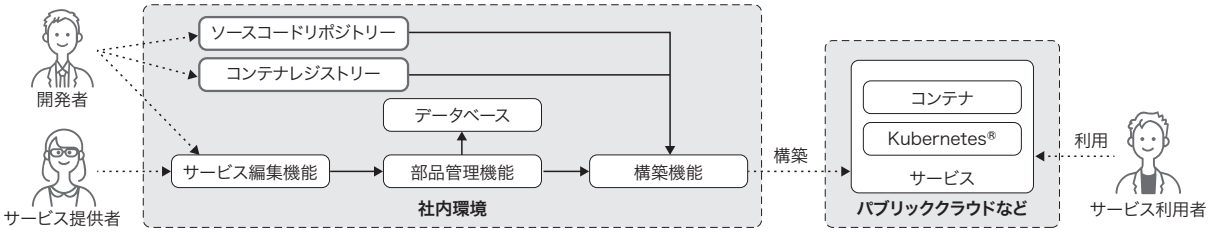


図3. TISFの構成

開発者が登録した部品をサービス提供者が編集し，サービスを構築する。

Architecture of TISF

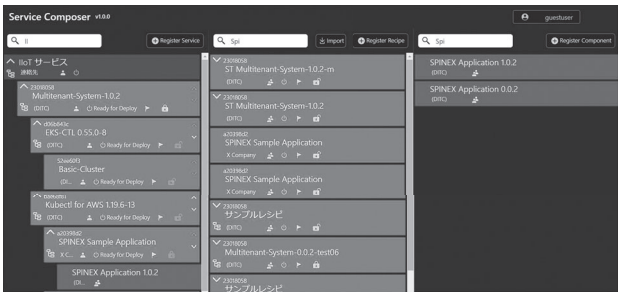


図4. サービス編集機能の操作画面

各部品がカード形式で表示されており，ドラッグアンドドロップでそれらを組み合わせ，サービスを構築する。

User interface display of TISF for editing of service functions

レシピやコンポーネントの情報を基に，ソースコードリポジトリーとコンテナレジストリーから該当する部品を読み込み，サービスに必要な機能を順に構築する。

TISFを利用する利点として，主に3点が挙げられる。1点目は，サービス提供者に高度なスキルがなくとも必要なセキュリティーや機能を持つサービスを構成できる点である。サービス提供者は，**図4**に示すサービス編集機能の操作画面を使用してサービスを構築する。各部品を組み合わせたパターンをドラッグアンドドロップしてコピーや編集を行い，簡単に新しいサービスを作成できる。また，サービスパターンのアップデートも操作画面上でドラッグアンドドロップにより行うことが可能である。

2点目は，オンプレミスや任意のクラウドサービス環境に自動的にサービスを構築できる点である。サービス構築には同じレシピのIaCを用いることで，同じ品質のサービスを数分から数時間で自動構築できる。

3点目は，CI/CD（継続的インテグレーション／継続的デリバリー）開発環境と連携可能なAPI（Application Programming Interface）を提供している点である。部品管理機能が持つREST（Representational State Transfer）APIにより，サービスの更新を自動化できる。

このように，TISFによって，サービス提供者は新たなサービスの構築や更新を素早く実現でき，高品質なサービスを迅速に提供できる。パターンA（図1）の遠隔監視サービスを適用した例では，二つの異なるサービスの構築にTISFを利用し，開発期間を従来の約2週間から3日に削減できた。

4. あとがき

東芝グループは，CPSサービスを顧客に迅速かつ高品質で提供するため，サービスをパターン化してその機能を部品化し，部品の構成を自由に構成できるUIで，SIを必要としない新しいサービスを提供できるTISFを開発した。TISFは，部品を各サービスと流用することで，開発・構築時間を大幅に削減し，セキュリティーや品質を損なわずに迅速に顧客にCPSサービスを提供できる。また，その提供したCPSは，DX（デジタルトランスフォーメーション）によるシステム変化にも柔軟に対応できるアーキテクチャーである。

今後は，TISFの仕組みをオープン化して部品の開発者を増やすとともに，部品の充実と様々なシステムの連携ができるSystem of systemsのエコシステムを実現していく。

文　献

(1) 山本　宏．東芝Cyber戦略2019 ～世界有数のCPSカンパニーを目指して～．東芝，2019，35p．<https://www.toshiba.co.jp/about/ir/jp/pr/pdf/tpr20191128_2.pdf>，（参照 2022-04-20）．

(2) Lin S-W. et al., eds. The Industrial Internet of Things Volume G1: Reference Architecture Version 1.9. Industrial Internet Consortium, 2019, 58p. <https://www.iiconsortium.org/pdf/IIRA-v1.9.pdf>, (accessed 2022-04-20).

(3) 東芝．CPSを実現する東芝IoTリファレンスアーキテクチャー．<https://www.global.toshiba/jp/cps/corporate/architecture.html>，（参照 2022-4-20）．

(4) 経済産業省．サイバー・フィジカル・セキュリティ対策フレームワーク Society5.0 における新たなサプライチェーン（バリュークリエイションプロセス）の信頼性の確保に向けて．Version 1.0，2019，261p．<https://www.meti.go.jp/press/2019/04/20190418002/20190418002-2.pdf>，（参照 2022-04-20）．

(5) National Institute of Standards and Technology. Framework for Improving Critical Infrastructure Cybersecurity. Version 1.1, 2018, 54p. <https://nvlpubs.nist.gov/nistpubs/CSWP/NIST.CSWP.04162018.pdf>, (accessed 2022-04-20).

小寺　志保　KODERA Shiho
デジタルイノベーションテクノロジーセンター 技術戦略部
Technology Strategy Dept.

中嶋　　宏　NAKAJIMA Hiroshi
デジタルイノベーションテクノロジーセンター 技術戦略部
Technology Strategy Dept.

杉本　信秀　SUGIMOTO Nobuhide
デジタルイノベーションテクノロジーセンター 技術戦略部
Technology Strategy Dept.