

トレンド

CPSを支えるソフトウェア生産技術の動向と東芝グループの取り組み

Toshiba Group's Approach to Software Systems Engineering for Cyber-Physical Systems

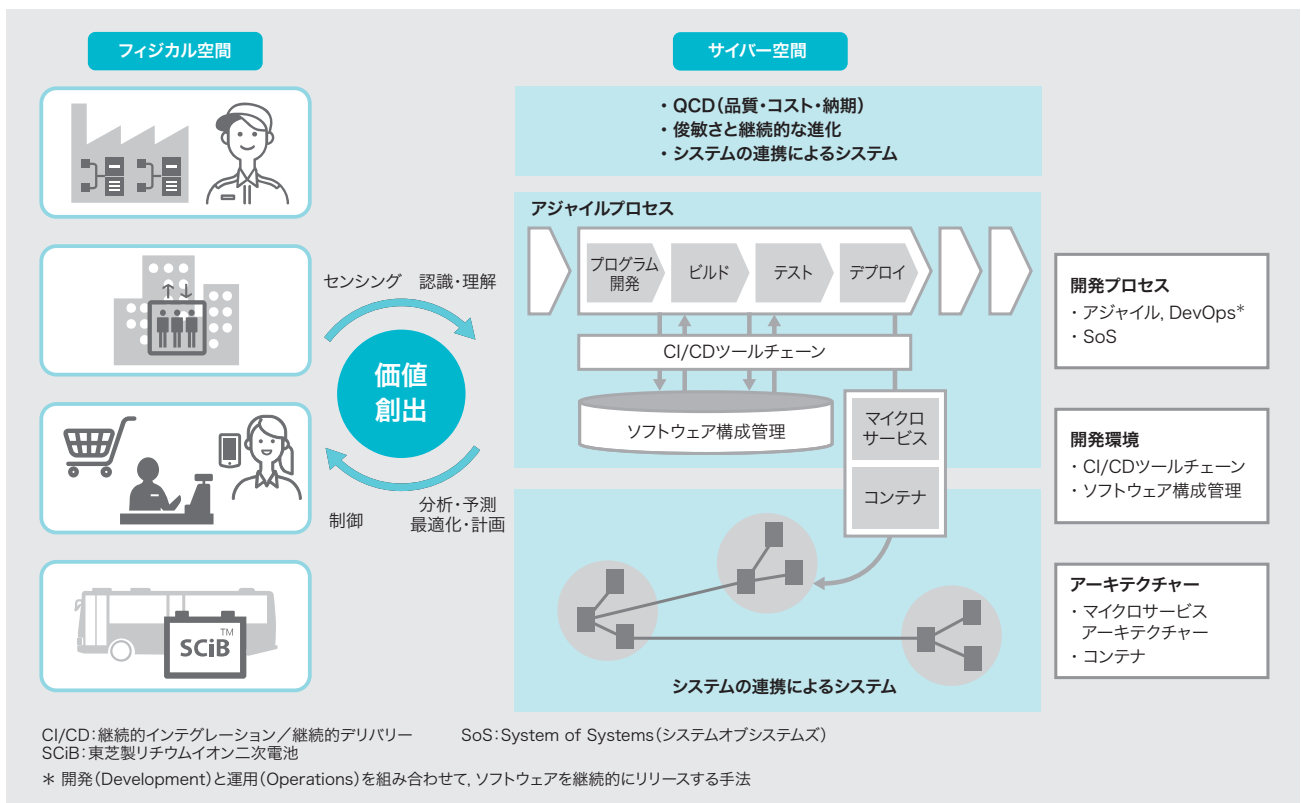
深谷 哲司 FUKAYA Tetsuji 今村 大輔 IMAMURA Daisuke

第4次産業革命の中心を成すCPS（サイバーフィジカルシステム）は、様々な機器と情報システムの融合で、高付加価値の製品・サービスを効率良く生産することを可能にする。CPSは、ソフトウェアで制御されるが、その開発には、頻繁なアップデートや、様々なシステムのAPI（Application Programming Interface）を介した連携など、従来の情報システムとは異なる開発技術が必要になる。

東芝グループは、これらの課題に対応した開発プロセス、開発環境、及びアーキテクチャーをソフトウェア生産技術として体系化し、CPSサービスの開発に適用している。

Cyber-physical systems (CPS), which form the core of the fourth industrial revolution, have made it possible to efficiently offer high-value-added products and services through the interaction of information systems and various physical components. CPS systems are controlled by software that must not only respond to frequent updates but also operate in conjunction with various types of systems via application programming interfaces (APIs). As a result, software systems engineering techniques different from those used for conventional information systems are essential for the development of CPS software.

The Toshiba Group has systematized a method for software systems engineering encompassing the development processes, development environment, and architecture of such software, and applied this method to the development of CPS services.



特集の概要図. CPSを支えるソフトウェア生産のための技術要素
 Elemental techniques for CPS software production

1. まえがき

現代は第4次産業革命の時代といわれ、目指すところは超スマート社会の実現である⁽¹⁾。超スマート社会では、生産、販売、消費といったサプライチェーンだけでなく、健康、医療、公共サービスといったあらゆる産業において様々な情報を活用することで、効率的な運用や新たなサービスの提供が行われる。

例として挙げられるのが、製造業におけるマスカスタマイズの実現や、シェアリングエコノミーによる資源・資産の効率的活用、AIやロボットの活用による人間の労働補助・代替などである。

これらを実現する仕組みがCPSである。CPSは、実世界のデータを収集・蓄積し、サイバー空間、すなわちIT（情報技術）を用いて分析・知識化を行い、そこで創出した情報・価値をフィジカル空間（実世界）にフィードバックする。

東芝グループは、インフラサービスカンパニーとして、データ活用により社会を支えるインフラの品質・効率を向上させ、更にはデータサービスにより、収集・蓄積したデータを生かした新サービスの開発と需要創出をすることで、社会に貢献することを目指している。その実現にあたって中心になるのは、やはりCPSである。

一方、ITの分野ではバイモダルIT⁽²⁾という概念が広がっている。これは、企業における情報システム及び情報システムに対する様式を二つのタイプに分け、使い分けることである。二つの様式のうち一つはモード1と呼ばれ、信頼性や安定性、業務効率を重視する様式である。もう一つの様式はモード2と呼ばれ、俊敏性、ビジネス上の価値創出を重視する様式である。従来の企業システムは、モード1として位置付けられる。これに対してデジタルトランスフォーメーション（DX）を実現する手段としてのCPSは、価値を高めることに主眼を置いており、モード2の様式を取り入れるべき分野と考えられる。

このような時代において、東芝グループは世界有数のCPSテクノロジー企業を目指すことを、事業戦略の目標として掲げている。

2. CPSを支えるソフトウェア

CPSを構成するソフトウェアが果たす重要な役割は、実世界から集められたデータの処理である。CPSプラットフォームのデータソースには、以下に示す活用例のように、IoT（Internet of Things）、IoP（Internet of People）、及びIoS（Internet of Services）の三つのドメインがある（図1）⁽³⁾。



図1. CPSプラットフォームのデータソース

CPSプラットフォームのデータソースには、IoT、IoP、及びIoSの3種類のドメインがある。

Three domains of data sources of CPS platform

- (1) スマートファクトリー 工場の様々な設備・機器のセンターデータ（IoT）と業務データ（IoP）を関連付けることによる、生産性、品質、稼働率の向上
- (2) リテール統合プラットフォーム POS（販売時点情報管理）データなどの購買データ（IoP）と物流データ（IoT）を基にした、購買・需要予測や配送最適化
- (3) 昇降機の運行・保守最適化 BIM（Building Information Modeling）情報と昇降機の運行状況（IoT）及び故障履歴などのデータ（IoS）を基にした、ライフサイクルをスルーしたサービス提供

これらのデータの蓄積・分析・知識化といった処理は、ソフトウェアによって行われるが、これらの例に共通する、従来のソフトウェア開発様式との違いは、俊敏さと継続的な進化が必要とされること、及び様々なシステムがつながり連携することで新たなシステムを構成するという点である。

実際のデータを基に、効率や品質を改善するための仮説検証を繰り返し行い、結果をフィードバックするには、ソフトウェアの開発サイクルを短くし、かつ発生する変更を柔軟に取り込み常に進化し続ける必要がある。また、新たな価値・知見を創出するために複数のデータソースを関連付ける場合、それぞれのデータソースを管理する情報システム同士をつなげる必要がある。

今回の特集では、CPSを支えるソフトウェア生産技術として、俊敏さと継続的な進化を実現する技術、及びシステム同士をつなげる技術に焦点を当てる。また、ソフトウェア

生産技術を、開発プロセス、開発環境、及びアーキテクチャーの3分野に分けて述べる(特集の概要図)。

3. 俊敏さと継続的な進化を実現する技術

3.1 アジャイルプロセスへの取り組み

企業や公共機関で用いるソフトウェアシステムの開発では、長らく、ウォーターフォールモデルと呼ばれる開発プロセスが用いられてきた。ウォーターフォールモデルとは、開発の工程が、要求、設計、プログラム作成、テストと逐次的に進むプロセスモデルである。各工程の成果物となる仕様書や設計書などのドキュメントを明確に定義し、上流の成果物に基づいて下流の工程の作業が行われる。

ウォーターフォールモデルは、工程の全体像を俯瞰(ふかん)してスケジュールや、担当者、経費などのリソース分配の計画を立てやすいという利点があり、規模の大きな開発プロジェクトでは現在も主流の開発プロセスである。

ウォーターフォールモデルは、その名(waterfall=滝)が示すとおり、工程が上流から下流へと一方向に進むことを原則としており、プロジェクトの期間中に発生した環境変化への対応や、上流で見落としていた要求への対応がしにくいという問題がある。そこで、システム全体を幾つかの部分に分け、反復しながら段階的に機能や品質を洗練しながら統合していく、反復型開発モデルが提唱・実践されてきた。

ウォーターフォールモデル、反復型開発モデルは、いずれにせよ、製品の利用者へのリリースは全体が完成してからとなる。一方で、CPSが目指すのは、ユーザーに早く価値を届けることであり、変化に対応することであるため、ウォーターフォールモデルによる開発が適合しない場面が出てくる。特に、収集・蓄積したデータを利用者に対し可視化する機能や、分析・加工したデータを価値あるサービスとして他者に提供する機能などは、初めからベストな仕様を決めにくく、更にビジネスの変化に伴い要求される機能も刻々と変化することが予想される。

このような状況に対応するため、2001年に提唱されたのがアジャイルプロセスである⁽⁴⁾。アジャイルプロセスは、決められた計画の遵守より変化への対応に、また包括的なドキュメントより実際に動くプログラムに、より価値を置くといった考え方に則している。

更に、現在ソフトウェアは、SaaS (Software as a Service)の形態が主流になりつつある。SaaSの利点の一つに、常に最新のバージョンを利用できることがある。すなわち、サービスの魅力を維持するためには、ユーザーの要望を取り込み、ソフトウェアを継続的に改善し続けなければならないということである。このために、開発(Development)と

運用(Operations)を組み合わせたDevOpsという手法により、システム開発のライフサイクルを短縮し継続的にリリースすることが行われている⁽⁵⁾。

東芝グループでも、10年程前から一部の製品開発でアジャイルプロセスを採用しているが、より多くの製品・サービスに適用するために、手法やプロセス標準の整備をしている。また、併せてDevOpsの手法も整備し、プロジェクトで実践をしている(この特集のp.15-18参照)。

3.2 DevOpsを支援する開発環境

DevOpsの実践には、開発環境による支援も必要である。数日に一度、更には1日に何度もアップデートを行うためには、開発支援ツールを利用し、プログラムのテストから運用環境へのデプロイ(実行環境へモジュールを配置すること)に至る一連のタスクを極力自動化する必要がある。このような、ソースコードの変更に対して自動でテストをすること、及びテストを終えたソフトウェアを運用環境にリリース可能な状態にすることをCI/CD(継続的インテグレーション/継続的デリバリー)と呼び、これらのタスクの自動化を支援するツールをCI/CDツールと呼ぶ。CI/CDツールには、SaaSやOSS(オープンソースソフトウェア)などで提供される。

複数のツールを組み合わせ、連鎖的に利用することをツールチェーンと呼ぶ。東芝グループでは、OSSをはじめとした各種ツールを組み合わせ、更に独自の機能追加をしたCI/CDツールチェーンをSaaSとしてグループ内に展開しており、部門での活用が進んでいる(同p.19-22参照)。

また、現在のソフトウェアの開発は、様々なOSSを利用しアプリケーションに組み込んで行われる。OSSの利用は、オープンな技術を用いて迅速に開発を進めることができる一方で、以下のような留意点もある。

- (1) OSSごとに異なるライセンス条件 OSSには、GPL (GNU General Public License)やApacheライセンスをはじめとした様々なライセンスがあり、利用条件が異なるため、個々のOSSが商用利用可能なライセンスかを確認する必要がある。
- (2) 脆弱(ぜいじゃく)性を攻撃されるリスク ソースコードが公開されているため、そのセキュリティー脆弱性が発見されて攻撃されるリスクを負うおそれがある。そのため、少なくとも利用開始時点で既知の脆弱性の有無を確認する必要があり、その後も新たな脆弱性の発見に備える必要がある。
- (3) OSS間の依存関係の煩雑な管理 あるOSSがほかのOSSを用いていることが多く、それぞれのOSSが頻繁にアップデートされるため、バージョン間の整合性を保つための依存関係の管理が必要となる。

これらの確認作業を手作業で実施することは、多大な労力を要することとなる。そこで、ソフトウェアの構成管理と、OSSのライセンスや、脆弱性情報、依存関係のデータベースなどを統合して管理するツールが用いられるようになってきた。

東芝グループでは、これらに加え、作成したソフトウェアをグループ内で共有すべきソフトウェア資産として管理する機能とも統合し、共創ソフトウェア開発プラットフォームを開発している(同p.23-26参照)。

4. システムの連携によるシステム

4.1 システムライフサイクルプロセス

CPSは、様々なシステムがつながり連携するシステムであることを2章で述べた。このような独立したシステムが統合されたシステムを、システムオブシステムズ(SoS: System of Systems)という。そして、複数の専門分野にまたがってシステムを組み合わせるアプローチを、システムズエンジニアリングという。品質、コスト、納期といった条件を満たしつつシステムの目的を達成するために、システムズエンジニアリングで行う活動をシステムのライフサイクルプロセスモデルとして標準化したものが、ISO/IEC/IEEE 15288:2015(国際標準化機構/国際電気標準会議/電気電子技術者協会規格 15288:2015; 以下、ISO 15288と略記)である。ISO 15288は、合意プロセス、組織的プロジェクトイネープリングプロセス、技術管理プロセス、及び技術プロセスの四つのプロセス群から構成される⁶⁾。各プロセス群は、更に幾つかのサブプロセスから構成され、例えば技術プロセスは、ビジネスあるいはミッションの解析や、システム要求の明確化、アーキテクチャーの明確化などのサブプロセスから構成される。

東芝グループでは、各種標準規格を基にプロセスを標準化し、エネルギーや公共交通など、社会インフラを支える大規模システムの構築で実績を重ねてきた。そして、CPSを対象として、ISO 15288を基にシステムズエンジニアリングの観点を加えることで、従来のプロセスモデルを発展させる取り組みも行っている(この特集のp.11-14参照)。

4.2 マイクロサービスアーキテクチャー

4.1節で述べたISO 15288のプロセスにも定義されているが、ソフトウェアシステムを開発する上で、アーキテクチャーを明確化することは非常に重要である。ソフトウェアシステムにおけるアーキテクチャーとは、ソフトウェアによって構成されるシステムの外部特性、構成要素、及びそれらの関係のことである。また、システム全体の構成を対象とするシステムアーキテクチャーや、システムの要素となるアプリケーション

の構成を対象とするアプリケーションアーキテクチャーなど、その対象スコープ別の観点がある。

以前は、様々なアーキテクチャーが利用され、その時代の主流となるアーキテクチャーがあった。

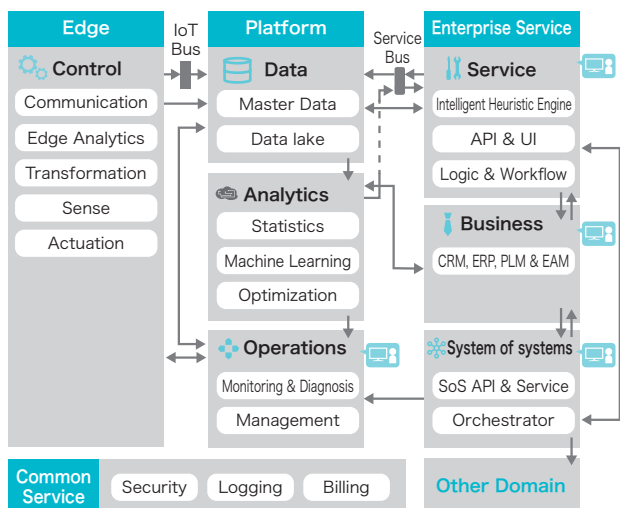
例えば、システムアーキテクチャーでは、メインフレーム(1970年代)以降、クライアントサーバーモデル(1980年から1990年代)や3層Webモデル(2000年から2010年代)といったシステムアーキテクチャーが用いられてきた。ほかにも、ネットワークを介して様々な機能を連携させる方式として、1990年代にはCORBA(Common Object Request Broker Architecture)などの分散オブジェクトモデルが、2000年代にはSOAP(Simple Object Access Protocol)というXML(Extensible Markup Language)ベースの protocols(通信の規約)を用いたSOA(Service-Oriented Architecture)というアーキテクチャーが一部のシステムで採用されてきた。これらのアーキテクチャーは、ハードウェアやネットワークの進化に応じてその処理性能を最大効率で利用するために、集中型と分散型を繰り返してきた歴史がある。

2000年以降、インターネットの普及によりクラウドコンピューティングがITの先進技術をリードするようになり、ハードウェアやネットワークのリソースの概念が大きく変わった。コンピューターの処理能力は、クラウドサービス上のリソースをほぼ無尽蔵に利用でき、また、ネットワークもブロードバンドの普及により複数拠点間で通信する場合の遅延がユーザーにとっては気にならない程になった。

その結果、現在マイクロサービスアーキテクチャーというシステムアーキテクチャーが主流となりつつある。マイクロサービスアーキテクチャーとは、以下の特徴を持つ。

- (1) システムが、比較的小さな粒度のマイクロサービスという独立した機能要素の集合体として構成されている。
- (2) マイクロサービス間は、REST(Representational State Transfer)というWebで汎用的に用いられているプロトコルを用いて通信する。
- (3) 各マイクロサービスは、その利用方法をAPIとして公開している。

このアーキテクチャーのメリットは、利用方法が汎用的かつ簡便であり、マイクロサービスをコンポーネントとして再利用し、組み合わせることが容易という点である。様々なマイクロサービスを組み合わせることで、新たな機能を柔軟に作るができる。また、このアーキテクチャーは、実装するプログラム言語や、マイクロサービスが動作するハードウェア、OS(基本ソフトウェア)といったプラットフォームに依存せずに開発できることもメリットである。このため、マイ



UI:ユーザーインターフェース
 CRM:Customer Relationship Management(顧客関係管理)
 ERP:Enterprise Resources Planning(経営資源管理)
 PLM:Product Lifecycle Management(製品ライフサイクル管理)
 EAM:Enterprise Asset Management(設備資産管理)

図2. 東芝IoTリファレンスアーキテクチャー

IIC (Industrial Internet Consortium) など世界のリファレンスアーキテクチャーのスタンダードを踏襲し、CPSに必要な要件を盛り込んでいる。

Toshiba IoT Reference Architecture

クロスサービスの提供者は、最適なプログラム言語やプラットフォームを選択できる。

一方で、多くのデータが関連し合い、かつデータが参照されるだけでなく頻繁に更新される企業向けシステムでは、分散するマイクロサービスとデータの境界を一致させることが難しいという問題もある。東芝グループでは、マイクロサービスアーキテクチャーをインダストリアルIoTサービスであるTOSHIBA SPINEXでいち早く実践して実績を積んできた。

また、TOSHIBA SPINEXも準拠している東芝IoTリファレンスアーキテクチャー (Toshiba IoT Reference Architecture) (図2) は、機能間の連携をAPIで行うことを規約としており、リファレンスアーキテクチャーを具体化した東芝IoTリファレンス実装 (Toshiba IoT Reference Implementation) では、APIをマイクロサービスにより実現している。この特集では、最新のアーキテクチャー (同p.7-10参照) や、CPS領域におけるマイクロサービスアーキテクチャーの実践事例 (同p.27-34参照) について説明している。

5. マネージドサービスによる運用の最適化

ここまで、CPSをサービスとして提供するために必要なソフトウェアの開発技術について述べてきた。3章でDevOpsについて述べたように、運用と開発を短いサイクルで繰り返す

ためには、運用のしやすさに配慮しなければならない。複数の種類のサービスを同じ手順でまとめて運用できれば、個々のサービスの運用コストを下げるができる。システムの運用まで含めて、システムの機能をサービスとして提供することをマネージドサービスと呼ぶが、マネージドサービス化するには、開発段階から運用の自動化・定型化を想定することが必要となる。

東芝グループでは、運用を自動化・定型化するためのガイドライン(この特集のp.35-38参照)を整備しており、前述の東芝IoTリファレンス実装の開発などで活用している。

6. 今後の展望

CPSをサービスとして提供するためにソフトウェアの占める役割は今後ますます大きくなり、その開発に用いる技術も更に早い進化を遂げられると思われる。

安全・安心な品質を保ちつつ、迅速かつ柔軟に価値あるサービスを提供し、東芝グループが優れたCPSテクノロジー企業として社会に貢献するために、今後も最新のソフトウェアの生産技術を実践していく。

文献

- (1) 内閣府政策統括官, “新たな産業変化への対応”. 日本経済2016-2017 一好循環の拡大に向けた展望一. 内閣府, 2017, p.72-75. <https://www5.cao.go.jp/keizai3/2016/0117nk/pdf/n16_2_1.pdf>, (参照 2020-06-10).
- (2) Gartner, "Bimodal". Gartner Glossary. <<http://www.gartner.com/it-glossary/bimodal>>, (accessed 2020-06-10).
- (3) 東芝デジタルソリューションズ, “「東芝IoTリファレンスアーキテクチャー」と「TOSHIBA SPINEX」”. DiGiTAL T-SOUL. <<https://www.toshiba-sol.co.jp/articles/tsoul/32/001.htm>>, (参照 2020-06-10).
- (4) Beck, K. et al. “アジャイルソフトウェア開発宣言”. Agilemanifesto. <<https://agilemanifesto.org/iso/ja/manifesto.html>>, (accessed 2020-06-10).
- (5) Bass, L. et al. DevOps: A Software Architect's Perspective. Addison-Wesley Professional, 2015, 352p.
- (6) ソフトウェア高信頼化推進委員会 システムズエンジニアリング推進WG. “システムライフサイクルプロセス説明(案)”. 第2回システムズエンジニアリング推進WG. 横浜, 2017-10, 情報処理推進機構, 2017, 37p. <<https://www.ipa.go.jp/files/000062132.pdf>>, (参照 2020-06-10).



深谷 哲司 FUKAYA Tetsuji
 技術企画部 ソフトウェア技術センター
 情報処理学会会員
 Corporate Software Engineering & Technology Center



今村 大輔 IMAMURA Daisuke
 東芝デジタルソリューションズ(株)
 ソフトウェアシステム技術開発センター
 Toshiba Digital Solutions Corp.