

Web ソフトウェア工学の新時代

Web API, マイクロサービス, クラウドコンピューティングを基礎とする新たなソフトウェア工学

青山 幹雄^{†1}

概要: 本稿では, Web ソフトウェア工学確立に向けて, Web ソフトウェアの特性とそれがもたらす新たな課題について議論する. この議論に基づき, Web ソフトウェア工学確立のために解決すべき課題について議論する.

An Invitation to the New World of Web Software Engineering

MIKIO AOYAMA^{†1}

1. はじめに: SOA の進化と深化

本稿では Web を基礎とするソフトウェア工学(以下, Web ソフトウェア工学と呼ぶ)を確立するための課題を提起する.

2000 年前後に提案された Web サービスに関する一連の技術基盤の上で, SOA(Service-Oriented Architecture)[1]が発展した(図 1). この SOA を本稿では, 文献[3]にならい, 第一世代 SOA と呼ぶ. 文献[3]でも指摘されるように, 第一世代 SOA は要求する技術要素の多さ, 複雑さのために普及しなかった. それと並行して進化してきた, REST(Representational State Transfer)[5]のもつ簡便さにより, Web サービスは REST へと転換し, そのインターフェースである Web API が急増した. 例えば, ProgrammableWeb 上で公開されている Web API の数が 2018 年 12 月時点で 2 万を超えた[10]. しかし, Web API は多くの新たな問題を提起している[4].

SOA のもう一つのアーキテクチャ進化は細粒度の独立したサービスとして実装, 提供されるマイクロサービスである[9]. しかし, マイクロサービスもインターフェース定義は Web API となり, Web API が提起する問題点を共有する.

Web の基盤として, クラウドコンピューティングが急速に発展し, 多様な Web サービスの開発, 提供が容易となり, CI (Continuous Integration), CD(Continuous Delivery)の発展と共に開発から提供までが継続的に行える DevOps へと進化している. 一方, クラウドの集中性に対して, 処理のリアルタイム性やデータ管理の処理分散などの点からエッジコンピューティング(フォグコンピューティングを含める)が登場し, アーキテクチャが再び進化している[7].

このような Web ソフトウェアの進化と深化, ならびに, そのスピードに対して, ソフトウェア開発上での新たな技術が提起されている. Web ソフトウェア工学とも呼ぶべきソフトウェア工学の新たな枠組みが求められている. これは, ソフトウェア工学そのものに対して多くの研究開発の機会を提供するものである[13].

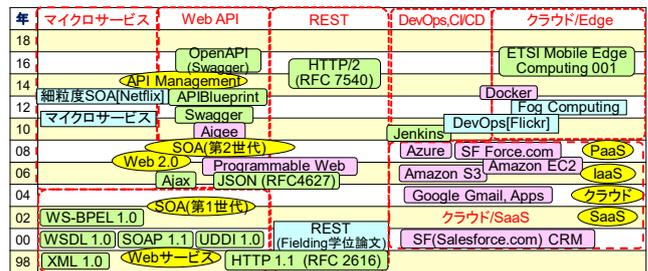


図 1 SOA の進化と深化

2. 第二世代 SOA のソフトウェア開発コンテキスト

Web APIは同一システム内で結合して利用されるAPI(以下, システム API と呼ぶ)とは本質的に異なる課題がある. 図 2 に Web API とシステム API を利用するソフトウェア開発のステークホルダと関連するシステムを示す. あわせて, 表 1 に Web API とシステム API の開発上での主要な差異を示す.

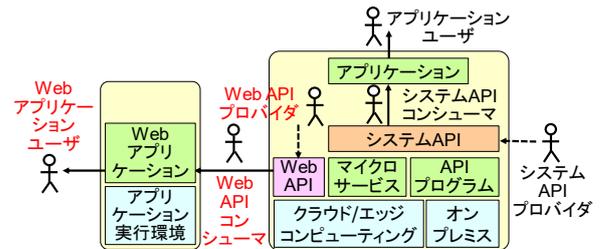


図 2 Web API とシステム API における開発コンテキスト

表 1 Web API とシステム API の開発上の主要な差異

比較項目	API	Web API
仕様とその定義言語	API 実装言語による手続き呼出しのシグネチャ	Web API 実装言語と独立なリソース表現
設計(利用)	設計時	実行時
実装	静的(Java などのプログラム言語)	動的(JavaScript, Python などのスクリプト言語)
実行	ローカル(同一環境)	リモート(未知の環境)
挙動	状態を持つ	状態を持たない

^{†1} 南山大学 理工学部 ソフトウェア工学科
 Department of Software Engineering, Nanzan University

3. 第2世代 SOA のもたらすソフトウェア工学の新たな課題

3.1 開発上の課題

Web API やマイクロサービスの利用はソフトウェア部品の再利用にあたる。第一世代 SOA ではインタフェース定義言語 WSDL が標準化されていた。しかし、第二世代 SOA では、OpenAPI などの仕様記述の提案はあるが、標準化には至っていない。また、システム API や WSDL と異なり、REST ではインタフェースは一つのリソースとして文字列で表現されることから、その仕様の形式性は弱く、型定義も明示されない。このため、多くの Web API が具体的な例示を用いてその仕様を自然言語で説明している[8]。

Web API を用いたアプリケーションの実装も JavaScript や Python などの動的スクリプト言語が多用され、インタフェースの整合性などの検証が困難となる。

3.2 実行上の課題

Web API ではシステム API と異なり、部品の実行環境と部品を利用したアプリケーションの実行環境が異なることが問題の原因となっている。このため、API マネジメントと呼ばれる、Web API の実行をモニタし、問題が発生した場合への対応を支援する新たなシステムの提案がある[2]。

実行時の課題として、サービス提供の継続性などの SLA に関わる課題や、プロバイダによる Web API の変化への対応などがある。これらの問題は従来からもあった。しかし、Web API では、状態を持たない疎結合性により、プロバイダ、コンシューマ共に、相手の状態に関わらず実装が変更可能であるため、問題がより深刻となる。

3.3 管理上の課題

Web API とそれを利用するシステムでは、例えば、品質特性の新たなモデルとその尺度が必要となっている。特に、Web API コンシューマ、すなわち、アプリケーション開発者にとって標準インタフェース定義の欠如や例示による説明など、API コンシューマの視点からの新たな課題が提起されている[14]。また、運用管理のために上述の API マネジメントは重要であるが、その開発技術は途上にある。

4. Web ソフトウェア工学確立への課題

Web 上での開発には様々な新たな課題があるにも関わらず、そのソフトウェア工学としての技術の整理、体系化は進展していないように思われる。

本稿では、この技術体系を Web ソフトウェア工学と呼び、その確立に必要な課題を議論する。

4.1 Web ソフトウェア工学を俯瞰できるフレームワーク

Web ソフトウェア工学では、Web API、マイクロサービス、クラウドコンピューティングなどの多様なプラットフォームを対象とし、かつ、アジャイル開発/DevOps に加え、上述の Web API を利用したシステム開発方法論の確立が必要

であるなど、ソフトウェア工学全体の見直しが必要である。そのため、これらの技術の位置づけと関係を俯瞰できるフレームワークの確立が必要である[7]。

4.2 Web ソフトウェア工学の要素技術の整理体系化

Web API とマイクロサービスなどは関係し、エコシステムを形成する[11][12]。しかし、個別に議論されることが多い。これらの技術を整理し、その開発技術の整理体系化が必要である。

4.3 Web ソフトウェア工学のマネジメントの整理体系化

Web ソフトウェア工学では、アジャイル開発、DevOps などの新たな開発形態をマネジメントできる技術の整理体系化が必要である。また、Web API はシステム API とは基本的に異なることから、従来の品質モデルは不適合である。Web API やマイクロサービスの新たな品質モデル体系化が求められる[14]。

4.4 グローバルな研究コミュニティとの連携

Web サービスに関する国際会議 ICWS と ICSOC が共に 2003 年に創設され[1]、今年で 15 年になる。この間、例えば、ICWS は 7 つの国際会議からなるマルチカンファレンスへと発展している[6]。毎年の発表論文も合計すると 500 編を超えている。Web ソフトウェア工学の流れを捉えるためには、グローバルな Web ソフトウェア工学コミュニティとの連携も必要である。

5. まとめ

Web ソフトウェア工学確立に向けて、Web ソフトウェアの特性とそれがもたらす新たな課題について議論した。多くの研究者の皆様が Web ソフトウェア工学のもたらす新たな課題に取り組み、その未開の沃野を耕して頂きたい。

参考文献

- [1] 青山 幹雄, サービス指向アーキテクチャの誕生と進化, ソフトウェアエンジニアリング最前線 2008, 近代科学社, Sep. 2008, pp. 9-16.
- [2] B. De, API Management: An Architect's Guide to Developing and Managing APIs for Your Organization, APRESS, 2017.
- [3] N. Dragoni, et al., Microservices: Yesterday, Today, and Tomorrow, Present and Ulterior Software Engineering, Springer, 2017, pp. 195-216.
- [4] T. Espinha, et al., Web API Growing Pains, J. of Systems and Software, Vol. 100, Feb. 2015, pp. 27-43.
- [5] R. Fielding, et al., Reflection on the REST Architectural Style and "Principled Design of the Modern Web Architecture", Proc. of ESEC/FSE 2017, ACM, Sep. 2017, pp. 4-14.
- [6] P. Hung, et al., Reflecting on Two Decades of Services Computing, IEEE Internet Computing, Vol. 22, No. 5, Sep./Oct. 2018, pp. 3-7.
- [7] Z. Mahmood, et al. (Eds.), Software Engineering Frameworks for the Cloud Computing Paradigm, Springer, 2013.
- [8] B. A. Myers, et al., Improving API Usability, CACM, Vol. 59, No. 6, Jun. 2016, pp. 62-69.
- [9] S. Newman, Building Microservices, O'Reilly, 2015.
- [10] Programmable Web, <https://www.programmableweb.com/>.
- [11] C. Pettey, Welcome to API Economy, Gartner, Jun. 9, 2016, <http://www.gartner.com/smarterwithgartner/welcome-to-the-api-economy>.
- [12] M. Verborgh, et al., A Web API Ecosystem through Feature-Based Reuse, IEEE Internet Computing, Vol. 22, No. 3, May/June. 2018, pp. 29-37.
- [13] E. Wittern, et al., Opportunities in Software Engineering Research for Web API Consumption, Proc. of WAPI 2017, IEEE, May 2017, pp. 7-10.
- [14] 山本 里枝子, ほか, Web API 品質モデルの提案とその定量評価, SES2018 論文集, 情報処理学会, Sep. 2018, pp. 111-120.