

Internet of Documents

木下修司*

東京都立産業技術大学院大学*

1 はじめに

東京都の「令和 3 年度 大学研究者による事業提案制度」に、筆者の提案テーマ「東京都地域防災計画の IoD 化による防災力向上」が採択され、2022 年 4 月から事業を推進している[1]. IoD とは筆者が提案する新しい概念 “Internet of Documents” の略称である. この概念は、筆者が 2018 年度総務省異能 vation プログラム「ジェネレーションアワード部門」に提案し、最終選考にノミネートされたものである[2]が、一般に公開された説明文書はない. 本稿では、この概念およびそれに関係する課題について説明し、今後を展望する.

2 文書作成にまつわる課題

2.1 文書だけでは何も動かない. しかし、文書がなければ細かい議論はできない

例えば、システム及びソフトウェア開発において現在主流となっている「アジャイル開発」(agile development)においては、その嚆矢となった「アジャイルソフトウェア開発宣言」[3]において、「包括的なドキュメントよりも動くソフトウェアを」と述べられている. この文には補足があり、「左記のことがらに価値があることを認めながらも、私たちは右記のことがらにより価値をおく」となっている.

この宣言文は、ともすれば基本設計書、詳細設計書といった文書のみでソフトウェア開発プロジェクトが進行する方式に懸念を表し、最小限動くソフトウェアを早めにリリースすることの重要性を説いている. 私の実感からも、それは事実であると思う一方、「ドキュメントは一切必要ない」といった誤った解釈をし、コミュニケーションに問題が発生するアジャイル開発プロジェクトも存在する.

すなわち、よりよいシステムを構築する議論のために、最低限のドキュメンテーションは必要だが、それがどのようなものかは明らかでないし、人間によるドキュメンテーションを支える要素技術も十分ではないと考えられる.

2.2 大規模な文書を長期間維持管理することの難しさ

筆者は以前より、大規模文書の管理に情報科学やソフトウェア工学の技術を応用する研究に取り組んでおり、特に自治体の防災業務を定めた「地域防災計画」と呼ばれる文書を事例として、神奈川県平塚市と共同研究を進めている.

地域防災計画は、災害対策基本法[4]によって日本の各自

治体に策定が義務付けられている文書であり、各種災害(地震・台風・原子力災害など)に対応するための計画を記述するものである. このような文書はどの自治体でもかなりの分量になり、例えば平塚市地域防災計画[5]においては地震対策の本編だけでも 250 ページ以上、風水害対策や資料編まで含めると 800 ページ近くなる. 東京都地域防災計画[6]であれば、地震対策(震災編)だけでも 700 ページ以上とその分量は膨大である. これらの文書の維持管理には以下のような問題が内在している.

- 修正の影響範囲の特定の困難. ある部分を修正する際に、その他どこを修正すればよいのかは目視(単純なキーワード検索による人間作業)で行うしかない.
 - 必要な情報がどこにあるのかを探すのが困難. 度重なる修正により、似たような情報でも別の章・節に散在していることがあるが、その全貌把握は難しい.
 - 修正時の手作業の膨大さ. 実際に編集する各部局に Microsoft Word 等のファイルを送付し、それを回収、マージ、競合発生時の解消など、すべて手作業
- 一方、一般のシステム及びソフトウェア開発に慣れている人間にとっては、「GitHub のようなバージョン管理システムがあればよいのでは?」「ソースコードのように型検査やユニットテストができればよいのでは?」といった発想も容易にできる. これらを含めドキュメントの維持管理を効率化する仕組みの総称が、次章で述べる Internet of Documents である.

3 Internet of Documents

3.1 ドキュメント≠データ

いわゆる IoT (Internet of Things) では、各種デバイスが通信し、相互にデータをやりとりする. これに着想を得たのが “Internet of Documents” (以下 IoD) である. 2018 年の[2]における提案文書では以下のような提案をした.

1. ドキュメントそのものと、
 2. ドキュメントの整合性(内部的整合性、他のドキュメントとの整合性)を自動検査するプログラム
- を常にペアにして、ドキュメントの不整合は自動的に通知する仕組みを構築する. (ドキュメントがインターネット・イントラネット上で IoT センサーのように通信、相互にやりとりし、変なところは自動修正したり、不整合を指摘してくれたりする仕組みを作る)

* Internet of Documents, Shuji Kinoshita, Advanced Institute of Industrial Technology

現在、ドキュメントというものは基本的には「データ」と捉えられている。例えば IEEE SEVOCAB[7]の定義では、データとは”representation of facts, concepts, or instructions in a manner suitable for communication, interpretation, or processing by humans or by automatic means”であり、何かしら人間や自動的な手段に「よって」処理されるものとされている。そうではなく、ドキュメント自体に“automatic means”を内在させることで、ドキュメント(の部分どうし)が自発的に相互の関連性や整合性をチェックするというのが、IoD の骨子である。

3.2 現実の実装方針

ドキュメントとプログラムを常にペアにする、という提案ではあるが、IoTのように「現実世界で離れた場所にある機器どうしが通信する」イメージを適用すると、以下のようなアーキテクチャになると考えている。

1. クラウド上に存在するサーバ上に、ある組織が所管するデータがデータベースとして一元管理されており、そのうえで各種チェック用の共通的なプログラムが動作する。
2. 外部の文書との整合性をチェックする場合は、別のサーバが用意するAPIを利用する

例えば日本国内の法律のデータであれば[4]のようにデジタル庁所管のe-Gov法令検索があるため、これの提供するAPIを利用することで「法令改正時に、変更の必要があるかどうかを通知」といったことは技術的に可能である。

4 関連研究

4.1 Agdaによる形式アシュアランスケース

アシュアランスケースとは、システム開発におけるステークホルダの議論を記録し、セキュリティや安全性といった望ましい性質を確かに持つことを議論するための手法である。ISO/IEC/IEEE 15026-2[8]として国際標準化もされている。このアシュアランスケースも、当然論理的整合性が求められるものであり、依存型付き関数型言語であり定理証明支援系であるAgda[9]を用いてアシュアランスケースを記述し、整合性検査を自動化する研究がなされている[10]。

Agda言語の習熟は必ずしも容易ではなく、これを直接IoDの整合性検査システムとして導入するのは難しいと考えているが、プログラミング言語に用意された仕組みを用いて整合性検査を実施することは、各種整合性検査プログラムを個別に実装する必要がないという優位性がある。

4.2 オープンシステムディペンダビリティと準拠枠

IoD化された大規模文書は、IEC 62853[11]として国際標

準化されたオープンシステムディペンダビリティ(開放系総合信頼性)達成に求められている「準拠枠」(frame of reference)となる。IoD化された地域防災計画が完成することで、IEC 62853に準拠した形での防災業務の信頼性を主張することが可能になる。

5 おわりに ～超議論社会の基盤としてのIoD～

これまで紹介してきたIoDの概念は、世界が今後向かうべきである社会「超議論社会」の基盤となるものだと考えている。超議論社会(Hyper Argumentation Society)とは、筆者の造語で、「社会を維持・発展させるための様々な事柄に対して、利害関係者が、印象論や感情論に左右されることのない、厳密で精緻な議論(超議論)を行う社会」を指す。

さまざまなシステム・サービスがインターネット等で「つながる」ことによって発生する思わぬ不具合が世間でも常態化しており、常に説明責任遂行が求められるなか、厳密で精緻な議論「超議論」が行われるべきことは自明だろう。IoDという概念が、文書の整合性だけにとどまらず、広く「安心・安全なシステムやサービスの構築」「それらについての利害関係者の議論、合意形成」に役立つものとなるよう、社会実装を進めていきたい。

参考文献

- [1] 東京都, 令和3年度 大学研究者による事業提案制度, <https://www.zaimu.metro.tokyo.lg.jp/zaisei/teian/4daigaku.html>
- [2] 総務省, 2018年度「異能ジェネレーションアワード」/ノミネート発表 <https://www.inno.go.jp/result/h30/nominate.php>
- [3] K. Beck et al., Manifesto for Agile Software Development (日本語版「アジャイルソフトウェア開発宣言」平鍋健児訳), <https://agilemanifesto.org/iso/ja/manifesto.html>
- [4] 内閣府, 災害対策基本法, <https://elaws.e-gov.go.jp/document?lawid=336AC0000000223>
- [5] 平塚市, 平塚市地域防災計画, https://www.city.hiratsuka.kanagawa.jp/bosai/page-c_01661.html
- [6] 東京都, 東京都地域防災計画, <https://www.bousai.metro.tokyo.lg.jp/taisaku/torikumi/1000061/1000903/index.html>
- [7] IEEE, Software and Systems Engineering Vocabulary (SEVOCAB), https://pascal.computer.org/sev_display/index.action
- [8] ISO/IEC JTC 1/SC 7, ISO/IEC/IEEE 15026-2 Systems and software engineering — Systems and software assurance — Part 2: Assurance case, 2022.
- [9] Agda Team, The Agda wiki, <https://wiki.portal.chalmers.se/agda/pmwiki.php>
- [10] Y. Kinoshita and M. Takeyama, Assurance Case as a Proof in a Theory — towards Formulation of Rebuttals, Proceedings of the Twenty-first Safety-Critical Systems Symposium, Bristol, UK., pp. 205–230, SCSC (2013)
- [11] IEC TC56, IEC 62853:2018 Open systems dependability, International Electrotechnical Commission, 2018.