CRDS-FY2020-WR-04

科学技術未来戦略ワークショップ報告書

Society 5.0システムソフトウエア

開催日

第１回 2020年6月24日（水）
第２回 2020年6月29日（月）

国立研究開発法人科学技術振興機構 研究開発戦略センター
Center for Research and Development Strategy, Japan Science and Technology Agency

エグゼクティブサマリー

JST CRDS は、国の科学技術イノベーション政策に関する調査・分析・提案を中立的な立場に立って行う組織として、今後わが国が取り組むべき重要な研究領域に関する提言を行ってきている。また、その提言に向けた前段階として、広く技術分野の状況を俯瞰し、注目すべき研究動向の抽出を行ってきている。これら俯瞰や提言は、文部科学省・内閣府をはじめとする政府の各種政策立案のためのインプットとなる。

今回、システム・情報科学技術分野の俯瞰の一環として、掲題の「Society 5.0 システムソフトウエア」に注目し、この研究動向の現状・課題や今後の取り組みの方向性を把握するためにワークショップを開催した。

Society 5.0 の社会システムは、今以上に大規模かつ統合的なものとなるが、既存クラウド基盤は、多大なエネルギーを消費するアーキテクチャーを採用しており環境的・経済的な面からは大きな課題がある。また、データを国の重要な資源として活用するためには、高度なセキュリティを、後追いでなく最初から考慮した基盤が必須である。

ムーアの法則という経験則に彩られた時代の終焉が近い中、このような基盤的分野をハードのみで支えることは困難である。そのため､ハードに近い部分のソフトウエア（カーネル）部分の研究開発を推進し、①多様化するアーキテクチャーの違いを意識せず使用することが可能となる基盤技術や、②個人情報などの機微情報を用いて解析する際の高度なセキュリティ技術の構築、③ゼロカーボン化に貢献する情報システム技術等を同時実現するための､「次世代 IT 基盤ソフトウエアの開発」が急務である。

一方で、このような状況下、情報システムのオープン化、コモディティ化により、わが国の研究環境は基盤構築に関わる研究開発をする機会を喪失してしまい、研究者は基礎基盤研究より応用分野の研究に注力することが多い。また、基礎基盤研究に身をおく研究者も分散し、相互の連携も弱くなっている。しかし、上記①～③を達成するためには、基盤技術の研究者同士、また基盤技術の研究者と応用分野の研究者が連携していくことが不可欠であり、そのような連携を可能とする「大規模研究プラットフォーム」の構築が必要と考える。

以上のような背景を受け、CRDS では、システムソフトウエアに関わる技術の俯瞰と、研究動向の現状・課題や今後の取り組みの方向性を把握するためにワークショップを企画した。

今回のワークショップは、昨年度来、文部科学省情報委員会および情報処理学会において検討していたものを、幅広い研究者層を交えて議論する場として設定された。したがって、このワークショップは戦略プロポーザル作成プロセスにおける有識者インタビューとして位置づけられており、いわゆる仮説検証を行うものではない。

ワークショップは、文部科学省情報担当参事官付、情報処理学会、国立情報学研究所の共催をうけ、2020 年 6 月 24 日に研究内容に関する議論（オンライン会議）、6 月 29 日は研究成果の応用に関する議論（オンラインとオンサイトのハイブリッド形式）で実施した。全体のコーディネーターを国立情報学研究所の河原林先生と京大の岡部先生にお願いし、情報処理学会の五つの研究会（アーキテクチャー、OS、セキュリティ、データ保護、プログラミング言語）から発表者と参加者を推薦いただいた。

参加者は、ステアリングメンバー（文部科学省橋爪参事官、情報処理学会 江村会長、国立情報学研究所 喜連川所長、河原林副所長、京大 岡部教授、CRDS 木村上席フェロー）、発表者としての研究者 12 名、ディスカッサントとしての研究者 20 名、文部科学省と JST からの傍聴者 30 名で総勢約 60 名に及んだ。

60 名× 5 時間× 2 日という長大なワークショップの結果、「理論×システム基盤技術」で安心安全な Society 5.0 情報処理基盤を構築するという考え方が導出された。また、参加者に研究開発提案を募集したところ 10 件の回答があった。発表 12 件を含めて、研究動向の現状・課題、今後の方向性を把握するという当初の目的は十分達成できた有意義なワークショップだったと言える。

なお、当初は 2 日間ともオンライン会議 ZOOM で実施する予定であったが、ステアリングメンバーからの強い要請で第 2 日目をオンサイトとのハイブリッド開催とした。また、30 名の相互討議を活発化するために、第 2 日目ではブレークアウトルームを活用したグループワークを実施した。いずれもテクニカルな課題は多く残ったが、ニューノーマルの時代のワークショップにおける一つの試行となった。

目　　次

エグゼクティブサマリー

1. 開催趣旨 ····· 1

2. ワークショップ（第 1 回）研究内容に関する議論 ····· 2
2．1　挨拶 ····· 2
2．2　セキュリティ分野 Overview 松浦 幹太（東京大学） ····· 5
2．3　セキュリティ分野最先端研究
「安心・安全な Society 5.0 の実現にむけて ～プライバシー保護データ解析技術の現在～」　盛合 志帆（情報通信研究機構） ····· 9
2．4　プログラム言語分野 OVERVIEW（＋ セキュア高性能データ基盤）
田浦 健次朗（東京大学） ····· 17
2．5　（プログラム言語分野最先端研究）プログラム検証分野の動向
小林 直樹（東京大学） ····· 23
2．6　コンピュータアーキテクチャーの研究動向～セキュリティの観点から～
井上 弘士（九州大学） ····· 27
2．7　（アーキテクチャー分野最先端研究）研究紹介 Califorms
佐々木 広（東京工業大学） ····· 30
2．8　プライバシー保護分野 Overview & 最先端研究 佐久間淳（筑波大学） ··· 40
2．9　S5Kernel のための基盤ソフトウェア～システムソフトウェアの視点から～
河野 健二（慶應義塾大学） ····· 46
2.10　全体討議　コーディネーター 河原林 健一（国立情報学研究所） ····· 54
2.11　ラップアップ ····· 65

3. ワークショップ（第 2 回）研究成果の応用に関する議論 ····· 67
3．1　挨拶 ····· 67
3．2　研究成果の活用（1）次世代 IT 基盤ソフトウェアへの期待
谷崎 勝教（三井住友フィナンシャルグループ、日本総合研究所） ····· 68
3．3　研究成果の活用（2）次世代 IT 基盤ソフトウェアへの期待
高橋 克巳（NTT セキュアプラットフォーム研究所） ····· 79

３．４　研究成果の活用（3）
「データ活用セキュリティ＆プライバシーにおける基盤技術の展望」
千田 浩司（NTT セキュアプラットフォーム研究所）・・・・・・・・・・・・・・・・・・・・・ 87
３．５　研究成果の活用（4）Future of Computing　計算の未来
福田剛志（日本アイ・ビー・エム株式会社）・・・・・・・・・・・・・・・・・・・・・・・・・・・・ 98
３．６　研究開発戦略センター（CRDS）の取組み　茂木 強（CRDS）・・・・・・・・・107
３．７　全体討議　コーディネーター 河原林 健一（国立情報学研究所）・・・・・・・・・110
（１）ブレークアウトセッション・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・110
（２）各ブレークアウトセッションの報告・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・110
（３）ブレークアウトセッションの報告のまとめ・・・・・・・・・・・・・・・・・・・・・・・・・・112
（４）総合討論・・・113

４. 提言書 ・・・115

５. 付録 ・・116
付録１. ワークショップ開催概要・プログラム ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・116
付録２．参加者一覧 ・・118

1．開催趣旨

JST CRDS は、国の科学技術イノベーション政策に関する調査・分析・提案を中立的な立場に立って行う組織として、今後わが国が取り組むべき重要な研究領域に関する提言を行ってきている。また、その提言に向けた前段階として、広く技術分野の状況を俯瞰し、注目すべき研究動向の抽出を行ってきている。これら俯瞰や提言は、文部科学省・内閣府をはじめとする政府の各種政策立案のためのインプットとなる。

今回、システム・情報科学技術分野の俯瞰の一環として、掲題の「Society 5.0 システムソフトウエア」に注目し、この研究動向の現状・課題や今後の取り組みの方向性を把握するためにワークショップを開催する。

Society 5.0 の社会システムは、今以上に大規模かつ統合的なものとなるが、既存クラウド基盤は、多大なエネルギーを消費するアーキテクチャーを採用しており環境的・経済的な面から大きな課題がある。また、データを国の重要な資源として活用するためには、高度なセキュリティを、後追いでなく最初から考慮した基盤が必須である。

ムーアの法則という経験則に彩られた時代の終焉が近い中、このような基盤的分野をハードのみで支えることは困難である。そのため､ハードに近い部分のソフトウエア（カーネル）部分の研究開発を推進し、①多様化するアーキテクチャーの違いを意識せず使用することが可能となる基盤技術や、②個人情報などの機微情報を用いて解析する際の高度なセキュリティ技術の構築、③ゼロカーボン化に貢献する情報システム技術等を同時実現するための､「次世代 IT 基盤ソフトウエアの開発」が急務である。

一方で、このような状況下、情報システムのオープン化、コモディティ化により、わが国の研究環境は基盤構築に関わる研究開発をする機会を喪失してしまい、研究者は基礎基盤研究より応用分野の研究に注力することが多い。また、基礎基盤研究に身をおく研究者も分散し、相互の連携も弱くなっている。しかし、上記①～③を達成するためには、基盤技術の研究者同士、また基盤技術の研究者と応用分野の研究者が連携していくことが不可欠であり、そのような連携を可能とする「大規模研究プラットフォーム」の構築が必要と考える。

以上のような背景を受け、CRDS では、システムソフトウエアに関わる技術の俯瞰と、研究動向の現状・課題や今後の取り組みの方向性を把握するためにワークショップを開催する。なお、ワークショップは、研究内容に関する議論と研究成果の応用に関する議論とで日時を分けて実施する。

2. ワークショップ（第 1 回）研究内容に関する議論

2.1 挨拶

（1）CRDS　木村上席フェロー

今日はSociety 5.0のシステムソフトウエアというワークショップを開催する。昨年度から文部科学省、国立情報学研究所、情報処理学会とJSTで、Society 5.0を実現するための基盤という議論をしてきた。それをもう少し具体化するためのワークショップである。

基盤としては、エネルギーアウェア、セキュリティ・バイ・デザイン、IoTなどのキーワードが出ているが、今回はその中でも特にセキュリティを中心とする。

ワークショップのコーディネーターをNIIの河原林先生と、京都大学の岡部先生にお願いし、12名の発表者と、20名の研究者の方に集まっていただいた。５分野の研究者の方々から最新の研究課題を発表していただき、最後に、１時間ほどパネルディスカッションを行う。長丁場になるが、密度の濃い議論ができればと思っている。最後までよろしくお願いする。

（2）文部科学省　橋爪参事官（情報担当）

本日は、多くの先生方に参加いただいた。また、CRDS、情報処理学会の皆さま、江村会長、喜連川先生、河原林先生、岡部先生には本ワークショップを計画していただいた。

木村上席フェローからも話があったとおり、このSociety 5.0システムソフトウエアについては、文部科学省において来年度から始まる第６期科学技術基本計画に向けた議論を進めてきた中で、AIなど、どちらかと言えば実用的に近い部分が注目されているが、もっと基盤的な部分ををさらに強化して次の時代のITをつくり出していくことが重要だという議論があり、このテーマを深掘りすることになった。

新型コロナウイルス感染症は非常に大きな影響を社会に与えている。日本では感染症の拡大自体は少し収まりつつあるのかもしれないが、こうした社会の変化は今後も続いていくと思われる。情報分野に対するその期待、あるいはその役割は、ますます大きくなってきており、こういった危機を経験したことで新しいニーズ、あるいはアイデアも出てきている。

皆さま方の率直な意見、本日の議論を受けて、われわれとしても、どのような分野や課題が今後重要になっていくのか、今後の政策に活かしたいと考えている。

（3）情報処理学会　江村会長

１年ほど前に、喜連川先生から話があり、情報分野のどのあたりを強化すべきかということについて、情報処理学会とNIIのメンバーを中心に、コアのメンバーが集まって議論を始めた。それがだんだんフォーカスを絞っていく中で、今日につながっている。その背景としては、コロナの問題もある。また、テクノ地政学のような話もあり、日本の国として基幹となる部分をしっかり担保していくことが必要だろうと言われている。そこで、

Society 5.0 のシステムソフトウエアというところに議論を絞り、今日と次回、特に次回はアプリケーション側からの視点も入れて議論をする。日本の中の超一流の研究者の方たちに集まっていただいているので、しっかりしたターゲッティングをして、国としての競争力を高めることにつながる第一歩になればいいかと思う。

（4）国立情報学研究所　喜連川所長

文部科学省情報科学委員会において、第６期科学技術基本計画に向けてどういうふうに情報分野がメッセージを出すかという議論があった。そのときに、NII に対しても質問をいただいたが、われわれがものを申すというよりも、やはりこういうときこそ、学会と一緒になって、全分野が、そもそも IT として、どう動いていくのかというようなことを俯瞰する場をつくったほうがいい。そこに文部科学省にも参加してもらい、その現場感を肌で感じていただくことが重要である。そこで、情報処理学会の江村会長の賛同をいただき、ほぼ全研究会のヒアリングをかけ、それぞれの研究会の夢や未来について聞いた。その中で分かったのは、大きな IT 企業であるアマゾンとかグーグルが持っていて、日本が持っていないものというのは、やはりシステムソフトウエアである。システムソフトウエアは、成果が出るまでに時間がかかり、論文も出しにくい、こういうところこそ、むしろ光を当ててやるべきだと考えた。特に JST においては、そういうところの予算づけがかなり希薄で、戦略目標として上げられているのは分かりやすいところばかりになっている。こういうことを情報処理学会と NII でまとめて、Society 5.0 のための基盤ソフトの必要性を文科省の会議で報告し、それが委員の先生方にも受け止められて、今に至っている。

本日、参加している先生は、どちらかというと基盤ソフトウエアの部分で、今まで一番光が当たっていなかったところだと思うので、ぜひ深い議論を進めていただきたい。

（5）京都大学　岡部教授

情報処理学会の副会長は先日退任したが、本件は引き続き宿題ということで、私が担当している。

今回のこのワークショップには、急遽ご都合を合わせていただいた方も多いと思う。コーディネーターの一人として御礼申し上げる。

経緯は、江村会長、喜連川所長から説明があったように、昨年の夏に、NII を会場として、ラウンドテーブルという形で、全研究会に発表してもらった。その後、幾つかの分野について文部科学省の政策にどのように結びつけていくかということを検討してきた。

それが今回、ようやく表に出てきた。各分野のオーバービューを話される方のうちの何人かは、当時のメンバーである。そういうコンテキストであるので、ぜひ、文部科学省、JST に政策的にアピールするようにというところも意識しながら、発表をいただければと思う。

（6）国立情報学研究所　河原林教授

江村会長、喜連川所長、岡部副会長、および橋爪参事官からもあったように本企画は１年前に始まった。夏に３回ほど、NII と情報処理学会の主催による情報処理学会の全研究会の発表をいただいた。その後、９月からずっとディスカッションをしていたということを皆さんにお伝えしたい。

今ある、機械学習や、その上の応用分野の話は、確かに華やかに見えるが、GAFA の強さ､マイクロソフトの強さは基盤分野である。応用側にシフトをしている､あるいは機械学習のレイヤーとかの仕事をしている方も、基盤が重要だというのは分かると思う。私は基盤でもなければ、機械学習でもなく、基礎分野だが、基礎サイドから見ても、基盤分野というのは長い歴史がある。今日参加された方は、主にセキュリティ、暗号の方、OS、アーキテクチャー、プログラム言語の方が多い。これらの分野に共通しているのは、ACM の SIG でいうと、大体 50 年、60 年やっている方々の分野が多いということである。

私がいるセオリーも同様で、STOC（Symposium on Theory of Computing）にしても、FOCS（Symposium on Foundations of Computer Science）にしても、50 年、60 年の歴史がある。皆さんのフラッグシップ会議も多分そんな感じではないか。そのように長い間、コンピュータサイエンスをずっと支えてきた基盤分野が、最近、ファンドという意味でも、研究分野の日本内でのビジビリティという面でも、応用サイドに負けている。上のほうの華々しいところばかり見られているような嫌いがある。基盤分野の皆さんにぜひ頑張っていただいて、基盤分野がもっと日本のビジビリティを世界で発揮できるような体制ができればいいと考え、これまで手伝ってきた。なお、今回のワークショップは、CRDS の皆さまに企画からすべて準備いただいた。それに対して深く感謝したいと思う。

出席者の皆さんに招待状が届いたのは、１週間とか 10 日ぐらい前で、本当にショートノーティスだったわけだが、趣旨にご理解いただき、出席いただき、大変ありがたい。今回のワークショップを実りあるものにしていきたいと強く思う。

最後に、文科省の皆さま方には、今回も次回も全出席いただけると聞いている。昨年秋から 10 回、15 回程度のディスカッションは、ステアリングコミッティの４名と、あと何人か今日発表される方々でやってきたわけだが、そこにも文科省から橋爪参事官をはじめ出席いただいたことが何回かあった。手厚いサポートをいただいていると感謝している。今後もよろしくお願いしたい。

2.2 セキュリティ分野 Overview

松浦 幹太（東京大学）

CSEC（コンピュータセキュリティ研究会）の代表としてこの活動に参加している。図2.2-1に示したように、CSECの設立趣旨には、セキュリティなら何でもやる、ということが書かれているので、研究会が主催している年１回のコンピュータセキュリティシンポジウムの様子を見ると、少なくとも国内の動きについては、ある程度俯瞰できる状況にあろうかと思う。

■ CSEC設立趣旨書（1997年9月）から

- 「・・・本研究会は、このような新しい情報環境の利用形態において、基礎となる理論・技術、通信プロトコル、コンピュータアーキテクチャ、オペレーティングシステム、アプリケーション、応用事例、管理運用、さらに社会科学的考察まで、研究していくことを目指している。」

■ コンピュータセキュリティシンポジウム（CSS）

- 参加者＞700、論文数＞200、WS４つを含む
- 研究用の共通データセット、解析競技会などの取り組み
- 研究倫理相談窓口（JSPSサイバーセキュリティ第192委員会、日本学術会議情報学委員会安全・安心社会と情報技術分科会にも協力）

図 2.2-1　情報処理学会コンピュータセキュリティ研究会

昨年まではリアルにやっていく中で、参加者数が継続的に増え、論文も増えたため、新しいトレンドを取り込むときには普通にセッションに入れるだけではなくて、ワークショップを入れるという形で進んできている。そこで、ワークショップの様子を見ながら、トレンドのキーワードを抽出したい。さらに、今回のワークショップは全体として安全・安心が非常に重要なキーワードということなので、われわれがコミュニティとして特に重視してきた取組についてご紹介させていただく。われわれの研究では、アナリシスを本格的に進める時、データセットをどうするかという点で、研究倫理の問題が大変重要になる。研究倫理については、個々の研究者の自助努力だけに任せるのではなく、コミュニティとしての取組が必要であると考えており、さまざまな取り組みをしている。

図2.2-2はコンピュータセキュリティシンポジウム（CSS）で、現在存在しているワークショップである。ひとつはマルウェアで、これはITの基盤をどう信頼できるかというところに直接的に影響してくる。先ほど少し触れた倫理関係といった、さまざまな取組を進めていくうちに、従来は学術の世界にあまり出てこなかったような、非常に技術力の高い方々も巻き込む形で成長につながっている。

■ MWS（マルウェア対策研究人材育成ワークショップ）

- 研究実施と成果の評価に関する研究倫理がとくに繊細。

■ PWS（プライバシーワークショップ）

- 保護する立場では、不安なだけでも問題になり得る。

■ UWS（ユーザブルセキュリティワークショップ）

- セキュリティ・バイ・デザインの象徴。

■ OWS（オープンソースソフトウェアセキュリティ技術ワークショップ）

- 一般化したケルクホフスの原則（攻撃者が秘密鍵以外の全てを知っていても、なお安全であるべき）の象徴。

図 2.2-2 CSSにおけるWSから見たトレンド

次はプライバシーだが、プライバシーには非常に独特の難しさがある。これについては、最先端研究ということで、盛合さんに詳しくお話しいただけるかと思うので割愛する。

その次はユーザブルセキュリティワークショップである。今回のワークショップのテーマである、システムソフトウエアでは、安全性とパフォーマンスのトレードオフのようなことが、随所にキーワードとして出てくるかもしれない。セキュリティ関連のトレードオフに関しては、トレードオフがあるのは当たり前というところがある。その中で細かな調節をして、このあたりが最適ではないか、ということを追求するのではなくて、根本的に何かセキュリティ・バイ・デザインのようなことをすることによって、非常に大きな進歩を目指す、といった研究が比較的世界でも流れをつくっており、日本の中でも、大きな動きがある。

最後はオープンソースソフトウェアセキュリティのワークショップである。安全性を評価するときには、技術を秘密にするから安全だ、というロジックについ頼りたくなるけれども、これには非常に多くの限界があり、特に、運用上大きな限界があるので、オープンソースソフトウエアの下でセキュリティ技術をどうやっていくかということが、非常にまとまった取組として重視されつつある。

これらのキーワードは比較的分かりやすいのだが、もう少しブレークダウンして、要素技術レベルや、比較的独立したプロトコルレベルでトレンドを見ると、一般的には伝統的な技術と思われるようなところで、幾つもトレンドが出てきている。図 2.2-3 に示したように、大きな要因としては、攻撃の進化や環境、あるいは研究の目標とするところの前提が大きく変わるというところにある。例えば、暗号でいうと、量子計算機の登場によって弱体化していく暗号に関して何とかしなければならないということで、耐量子計算機暗号というものがあったり、秘密計算というのがあったりする。秘密計算については、盛合さんや NTT の千田さんに詳しくお話しいただくが、私からも詳しく述べたいと思う。あるいは、IoT への適用ということで、軽量暗号ということも非常に大きなトレンドになっている。

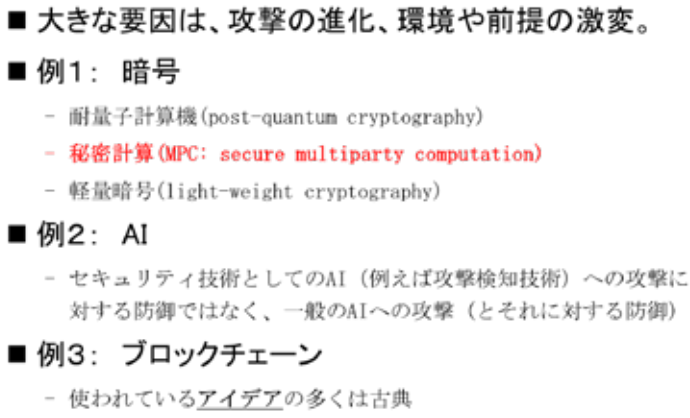


図 2.2-3　伝統的な技術にもトレンドあり

次に、セキュリティ技術で AI を使うのは当たり前なのだが、ここ５、6 年ぐらいは一般 AI への攻撃と、それに対する防御というものが非常に重要なテーマになっている。

３つ目のブロックチェーンについては、日本の活動でも頑張っているところがある。ただ、使われているほとんどのアイデアは、1980 年代まで遡るような古典的なアイデアが、再び新しい形でトレンドを作っているということがある。本日は、この中でも特に基盤ソフトウエアという観点で見て、重要な役割を果たし得る技術として、秘密計算（secure multiparty computation）を見ていきたい。

その前に、日本の学会だけでは、オーバービューとして本当に十分かと批判を受けそうなので、キーワードレベルで世界のものを並べておく。IEEE コンピュータソサエティー

が毎年示すトレンドの予測で出されたキーワードを並べて、セキュリティ関係のものに赤いハイライトをつけたものが図 2.2-4 である。おおむね３分の１程度はセキュリティと深く関わっている。

これらの特徴を、大体ざっくりとまとめると図 2.2-5 のようになる。セキュリティを何らかのモデルの下で追求していくとき、古典的には、いわゆる基本要素である、秘匿性、完全性、可用性について、できるだけしっかりと品質管理をしようという取組だったわけであるが、それにつけ加えて、アプリケーション固有の信頼関係、トラストリレーションシップというものを、しっかり管理していくというところにトレンドが出てきている。

multiparty computation について、図 2.2-6 にまとめた。一言で言うと、複数の参加者それぞれが持つプライベートな入力で関数を計算するが、どの参加者も自身の計算結果以外は知ることができないというものである。理論的可能性は非常に古くから知られていたが、応用者魂を刺激するような性能が示されてきたのは、ここ 10 年に満たないという状況である。

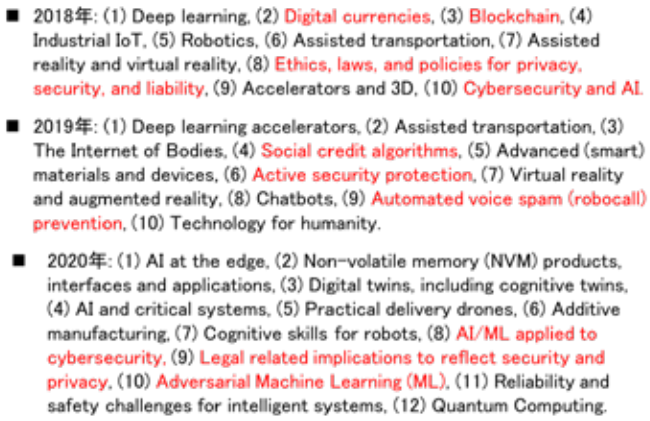


図 2.2-4　IEEE Computer Society によるトレンド予測

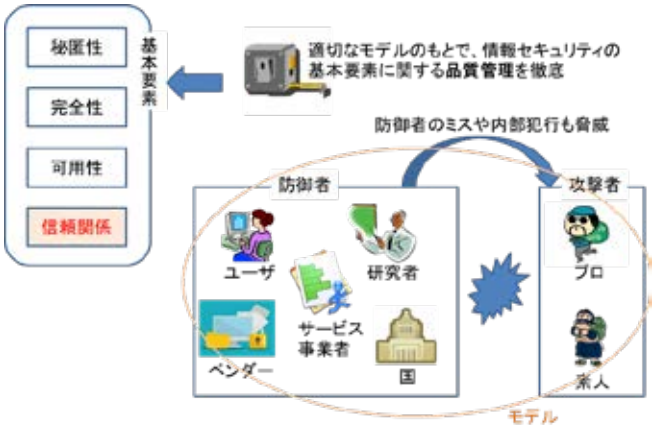


図 2.2-5　トレンドをセキュリティの基礎に立ち返って位置づけると

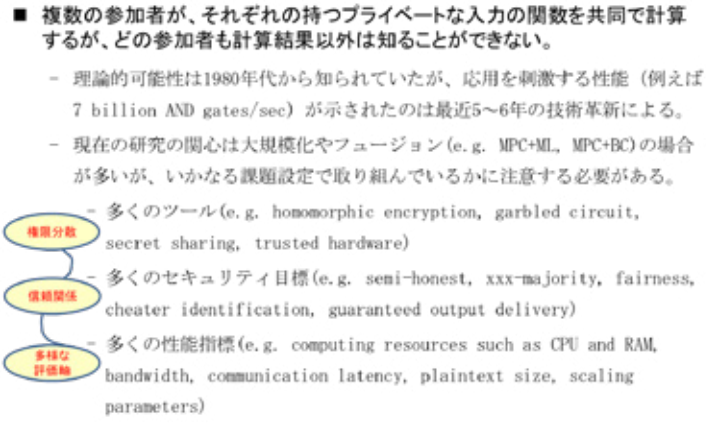


図 2.2-6　Multi-Party Computation

現在の研究の関心としては、応用を意識して、大規模化という話が出てきている。これについては、千田さんの話で詳しいことが出てくるのではないかと思う。それ以外に学術研究としては、例えばマシーンラーニングと組み合わせよう、ブロックチェーンと組み合わせようといったものがある。これは、単純に、はやり言葉を出しているということではなく、体系的な意味合いがあってトレンドになっているということになる。ただ、ツール

がたくさん出てきて、多くのセキュリティ目標があって、多くの性能指標があるという状況であり、トレンドになったせいで、逆に研究が豊富過ぎて、同じ土俵で比較できているのか、本当にそれらを同時にサーベイして、見通しのいい議論ができているのかという点については、非常に注意深く見る必要があると考えている。

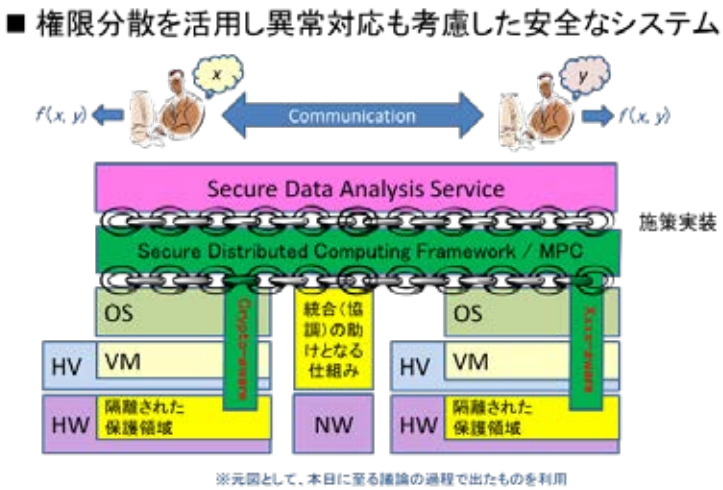


図 2.2-7　システムソフトウエアの視点

図 2.2-7 にシステムソフトウエアの視点で multiparty computation の位置づけを示した。ある関数の値を計算するときに、ある人はXだけ知っていて、ある人はYだけ知っているという形だと、当然通信が発生する。そうすると、個々のプレーヤーが、安全に実装しようとして、ハードウエアにまで踏み込んで設計・実装したとしても、コミュニケーションのところで、例えば異常対応のために非常に複雑な仕組みが必要となる。困ったことが起きたときにどうするのかという点まで踏み込んで研究開発を進めると、研究の枠組みがかなり違ってくる傾向がある。そういった意味合いもあって、先ほど少しキーワードレベルで挙げた、multiparty computation にブロックチェーンを組み合わせればさまざまな記録が残るので、計算結果を統合する上で何か困ったことが起きたときに対処できる、という付加価値が出てくる可能性がある。ここまで踏み込むと、応用によっては、ある種の施策の実装手段として非常に大きな役割を果たすことが期待できる、という状況になる。

まとめると、例えば今の MPC に見られるようなトレンドを見ても、Evolution ではなく、Revolution を目指す場合には、部品を融通するだけでなく、多くのコミュニティで連携する課題が重要になってくると予想できる。

2.3 セキュリティ分野最先端研究「安心・安全な Society 5.0 の実現にむけて ～プライバシー保護データ解析技術の現在～」

盛合 志帆（情報通信研究機構）

今日は、テーマをある程度絞った形で、安心･安全な Society 5.0 の実現に向けて、プライバシーを保護したままデータ解析を行う技術について、ご紹介する。

Society 5.0 は、サイバー空間とフィジカル空間を高度に融合させたシステムで、経済発展と社会的課題の解決を両立する、人間中心の社会として、第５期の科学技術基本計画の中でうたわれている。

Society 5.0 は PC だけでなく、IoT でモノがつながる世界を描いているが、新型コロナでテレワークや遠隔医療が身近になっているように、今はまさに ICT で社会が変革している時期ということができる。こういう時期においては、図 2.3-1 に示したように、これまで以上にデータセキュリティとプライバシーの確保が鍵になってくるのではないかと思う。

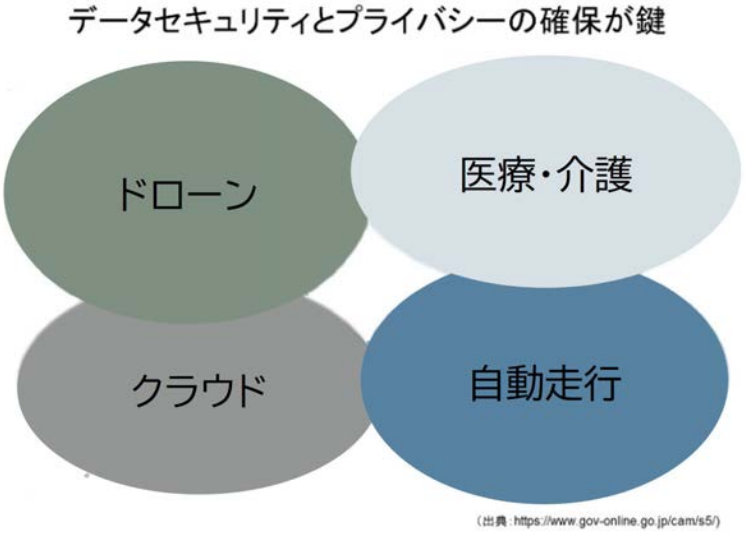


図 2.3-1　安心・安全な Society 5.0 に向けて
（図中の出典：https://www.gov-online.go.jp/cam/s5/）
2020年11月時点

また、Society 5.0 では、これまで必ずしも十分ではなかった分野横断的な連携を推進しようということもうたわれている。図 2.3-2 に示したように、IoT で多くのものと人とがつながって、さまざまな知識や情報が共有され、人工知能技術が活用されて、これまでになかった新たな価値を生み出す際に、データを守る、あるいはプライバシーを守るといった技術、つまり暗号技術と、人工知能技術の組合せのところで、難しい問題が出てくる。

図 2.3-2　暗号技術＋人工知能技術

まず、分野をまたぐ、あるいはその組織をまたぐとなると、これまで組織の中だけで使っていたデータを、外部に出すことになるので、データの漏えい対策が気になる。例えば、クラウドにデータを置くことは本当に安心できるのか、あるいは自社で蓄えたデータの中には、お客さまのプライバシー

情報が入っているけれども、それを適切に保護した上で、外に出していけるだろうか、といったことである。別の例としては、工場などでさまざまなセンサーデータを取るが、外部から来たデータは本当に信頼できるのか、といったことである。途中で改ざんされないように、暗号技術で守ってきているが、データを守りながら解析することが必要な人工知能技術で本当にうまく活用できるかということが課題になる。ここを両立することで、分野横断での利活用が促進できるのではないかと考えている。

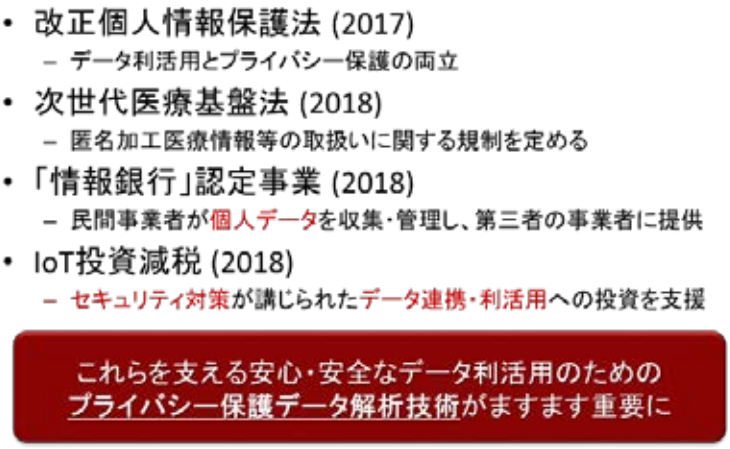


図 2.3-3　データ統合利活用を促す動き

データの統合利活用を促す動きとして、ここ数年、いろいろな法整備とか制度ができている。今日は詳しいところは割愛するが、個人情報保護法も改正されたり、医療情報に応用されたり、情報銀行や、IoT の投資減税といったこともある。これらの制度を支える安心・安全なデータ利活用のために、今日お話しするプライバシー保護データ解析技術が、ますます重要になっていると考えている。

以下、プライバシー保護データ解析、データを秘匿したまま解析を行う技術について紹介する。

プライバシー保護データ解析、あるいは秘匿計算、秘密計算といった言い方もあるが、技術的には幾つかの実現方法が知られている。今日はその中で、図 2.3-4 に書いてある準同型暗号という技術を使う方法を中心に話をする。これは暗号化したまま加算や乗算ができる、というものである。どのような計算であっても、分解すると、加算・減算グループと、乗算・除算グループで実現ができるのだが、準同型暗号は、それを暗号化したままで実行するものである。暗号化したまま加算ができるものを、加法準同型暗号、乗算ができるものを乗法準同型暗号というふうに言っている。両方できる完全準同型暗号というものが、2009 年に、当時 IBM にいた Gentry が作ったことで暗号化したままでのデータ解析に道が開けたという経緯がある。

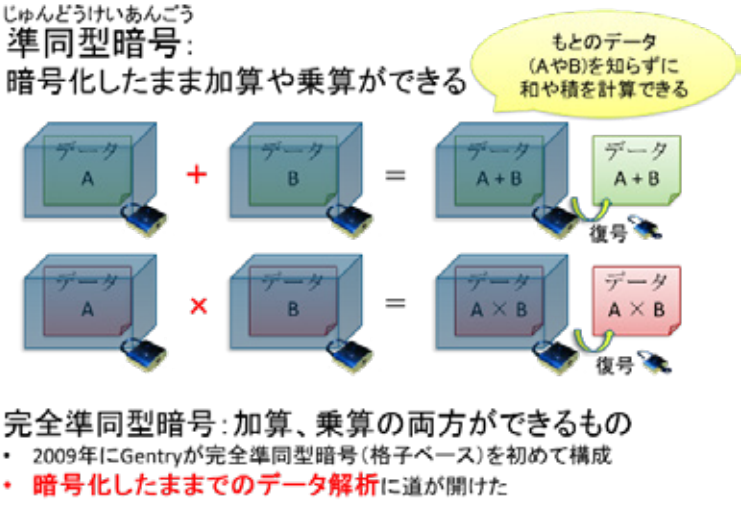


図 2.3-4　プライバシー保護データ解析

プライバシー保護データ解析には、いろいろな利用シナリオがある。図 2.3-5 に示す 3 つを例に挙げて紹介する。

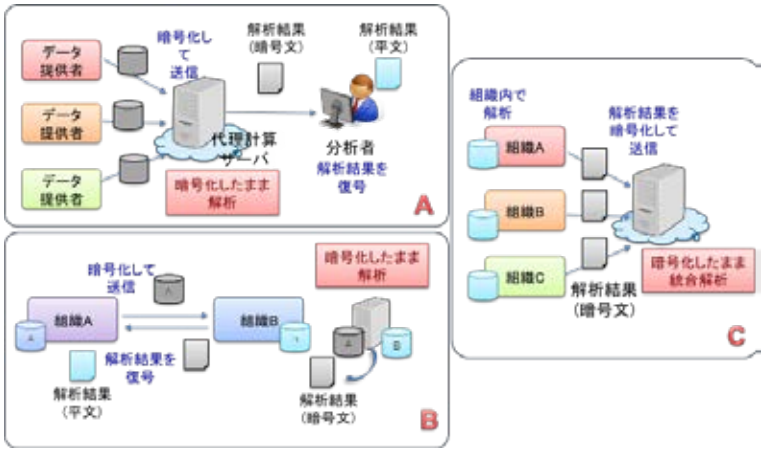


図 2.3-5　プライバシー保護データ解析利用シーン

シナリオＡはデータ提供者が３組織あり、機微な情報を暗号化してクラウド上にある代理計算サーバーに送る。灰色のものが暗号化したデータで、平文データを水色で示している。分析者は、これらのデータ提供者から集めたデータを使って、何らかの分析をするために、クラウド上にある暗号化されたデータを使い、代理計算サーバーで解析をする。その解析結果は暗号文で出てくる。これを分析者が自分の手元に持ってきて、自分しか復号できない鍵を使って、解析結果を平文で得るというシナリオになっている。

シナリオＢは、組織Ａ、Ｂがそれぞれ持っている、お互いに共有が難しいプライベートデータ、例えば医療データのようなものがあったときに、これを合わせた形で、例えば統計的な結果を得たいという場合である。組織Ａが暗号したデータを組織Ｂに送り、組織Ｂでは自分の手元にあるデータと、組織Ａから送られてきた暗号化されたデータを使って結果を得て、それを暗号化して組織Ａに渡すと、組織Ａが自分たちの組織で知りたかった解析結果を平文で得ることができるというようなものである。

シナリオＣは、データを一切クラウド上に上げないケースになっている。組織Ａの中で自分たちの持っているデータを基に分析、例えば機械学習のようなことをする。その学習結果を暗号化して、サーバーに送るということを各組織が行い、暗号化したままで統合解析を行うというシナリオである。

今日はこの３つのシナリオで、幾つか例を示す。まずＡのシナリオである。これは暗号化したデータをクラウド上で計算するというシナリオである。図 2.3-6 の例は、ビッグデータ解析で多用されているロジスティック回帰分析を、暗号化したまま計算したときに、どのぐらいの速度でできるかを調べた例である。これは、2016 年の結果なので、４年前と、少し古いデータだが、暗号化された１億件のデータを 30 分以内で複数のグループに分類できるということをシミュレーションで確認したというプレスリリースである。

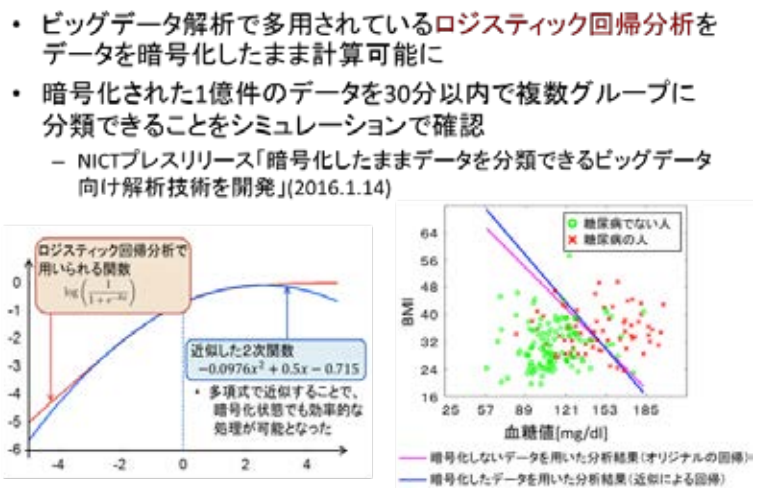


図 2.3-6　暗号化したままビッグデータ分類

右下のグラフにあるのは、身長体重の BMI と血糖値という、医療データとしては、個人情報として扱うべきデータを使って分類を行うものである。これらの情報を使い、糖尿病な

のか、そうでないのかをグループに分けるという計算を暗号化したまま行っている。

さらに、暗号化したまま分類した結果と、生のデータで分類した結果が、実は少しずれている、ということを、ピンクの線と青色の線で示している。ロジスティック回帰分析をする場合、本来は左のグラフの赤い線で示したような複雑な関数を、暗号化したまま計算しなければいけないのだが、暗号化したままでは時間がかかるので、青い線で示したような二次関数で近似している。そのため、右の図に示したようにグループに分類したときに差が出てくる。図 2.3-6 は、少し差が出るというところを示しているのだが、その後の進展したアルゴリズムを使うことで、近似がない状態で学習したり、識別したりということも、できるようになってきている。

データを暗号化したまま学習や予測をできるニューラルネットモデルというものを図 2.3-7 に示す。これは Privacy-Preserving Extreme Learning Machine という、３層のモデルを使い、近似を導入せずにワンショットで学習することで、図 2.3-6 で示したものよりも分類の精度がよくなる、という研究である。

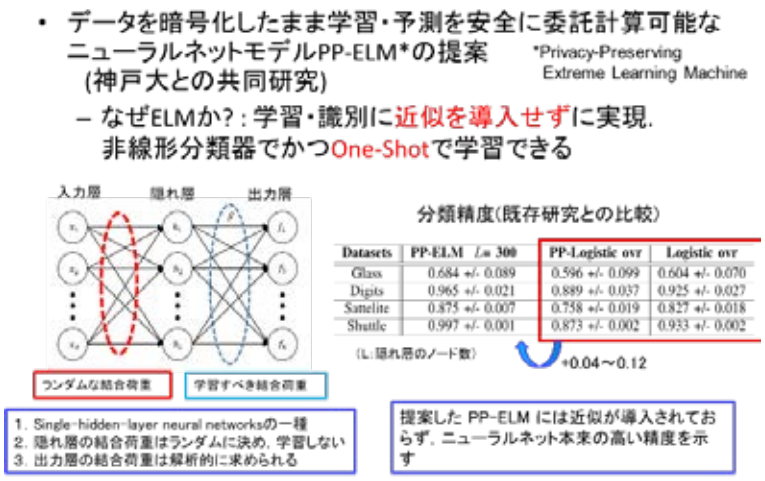


図 2.3-7　暗号化したまま近似なし学習・識別

図 2.3-8 はシナリオＢで暗号化したままデータを解析するとき、誤ったデータが混入するのを防止する技術として 2018 年にプレス発表したものである。今日の発表者の一人である、筑波大学の佐久間先生のところとの共同実験である。例えば最近非常にホットな話題として、医療データの解析がある。糖尿病に関するカルテを集めて、何らかの分析をしたいのだが、ある病院から、間違ってがんに関するカルテが出てきてしまったという想定の図である。暗号化したままデータを解析するということをやっているので、本当に正しいデータに対して分析をしているのか暗号化したままだと分からないという課題があった。暗号化したまま検索できる技術を組み合わせることで、このような課題も解決できるということを示したものである。

図 2.3-8　暗号化したままデータ解析時の誤データ混入防止

患者さんがどんな病気にかかっているかという情報と、その方の遺伝情報というのは違う機関が持っているのが通常だが、それを平文でマージすることなく、暗号化したまま統計的な関連性を解析することができたという例が図 2.3-9 に示したものである。

また、医療情報のような非常に機微な情報だけでなく、私たちの生活の中には、いろいろと暗号化したまま配慮して取り扱うべき情報がある。図2.3-10は株式会社ミイダスという、転職支援、採用支援のサービスをしている会社と一緒に実施したものである。これはある方の経歴、どんな職歴があるかとか、どんな職種を経験してきたか、年齢は幾つかといった情報を使って、その方が転職した場合にどれぐらいの年収になるかという、市場価値を予測するものである。ミイダスが既に蓄えているデータを暗号化したまま使って予測している。暗号化した経歴書を送ると、それを使って分析した結果を、自分だけが復号して見ることができるというものになっている。こうすることによって、ミイダスは、見ず知らずの方から個人情報を集めるというリスクを負うことなく、ユーザーの利便性を上げることができ、本サービスを利用してもらうことができる。

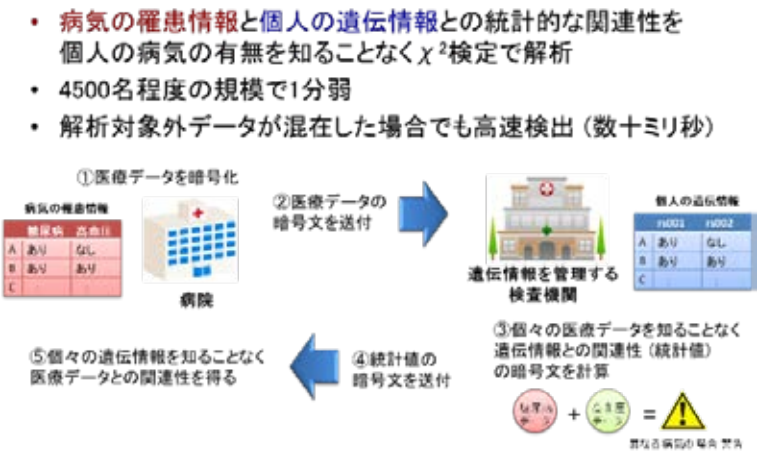


図 2.3-9　プライバシーを保護したまま医療データを解析する暗号方式を実証

図 2.3-10　データ活用事業に対する技術支援

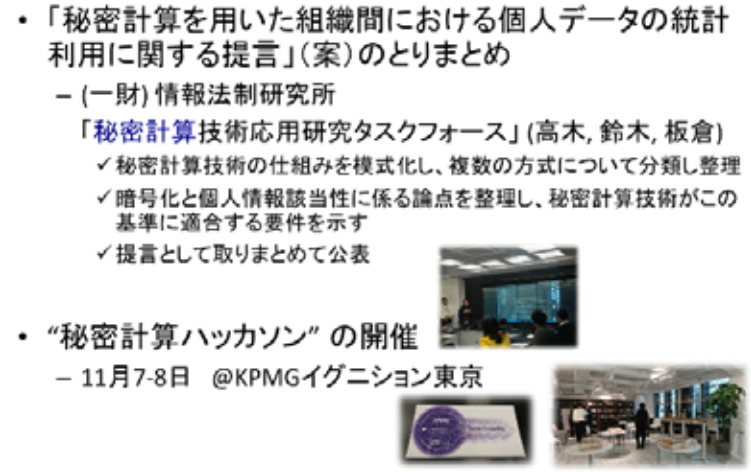


図 2.3-11　本技術の社会実装に向けた活動

こういう技術の社会実装に向けた活動ということでいろいろやっている（図2.3-11）。暗号化したままデータを解析する場合でも、日本の個人情報保護法では、暗号化しても個人情報は個人情報のままということなので、難しい壁もある。こういうところを、秘密計算を用いることによって、個人データの統計利用に関しては、どういう解釈ができるか、あるいは、どのような方向にしていったらいいか、ということを提言として取りまとめたりしている。

また、研究者コミュニティだけではなくて、もう少し草の根を広げていこうということで、この技術に興味を持たれているKPMGという監査法人と、去年の11月に、秘密計算ハッカソンというものをやったりしている。

次に暗号化したまま組織横断でディープラーニングをする技術について紹介する。図2.3-12に示したのは、金融機関と一緒に実証実験を開始した例である。これはJSTのCREST、人工知能研究領域で実施しているもので、今日参加している、産総研の花岡さんのチームが研究の代表者をしている。日本で、プライバシー保護データ解析技術をきちんと社会実装していこうということを目的としてやっている。５年計画の中の、今年が４年目になる。図の青色のプライバシー保護データ解析エンジンを、NICTと、神戸大学、エルテスという会社で開発している。特に金融機関が持っているデータを、プライバシー保護したまま解析することで、オレオレ詐欺などを含む不正取引の検知をやろうという実験である。金融機関では、銀行内のデータ、取引データ、ATMのデータなどいろんなデータを使って、マーケティングだったり、与信の管理だったり、不正取引検知だったりということをやっているのだが、銀行をまたいでのデータの利活用、統合利活用というのが難しい現状がある。われわれが研究開発した結果を使って、このような銀行のそれぞれが持っているデータを、お互いに共有することなく、何らかの課題を解決することを目指している。例えば、不正取引の検知では、各銀行に専門の部隊がいて、一生懸命人手を使ってやっている。この技術は、そういう場面で、調査コストを削減する、あるいはマイスター的な人がいないとできないといった属人化を回避していく、というあたりを狙っている。さらに、これまで人の目で見つからなかったものを、AIの技術を使うことによって見つけられないか、という期待も抱きながらやっている。

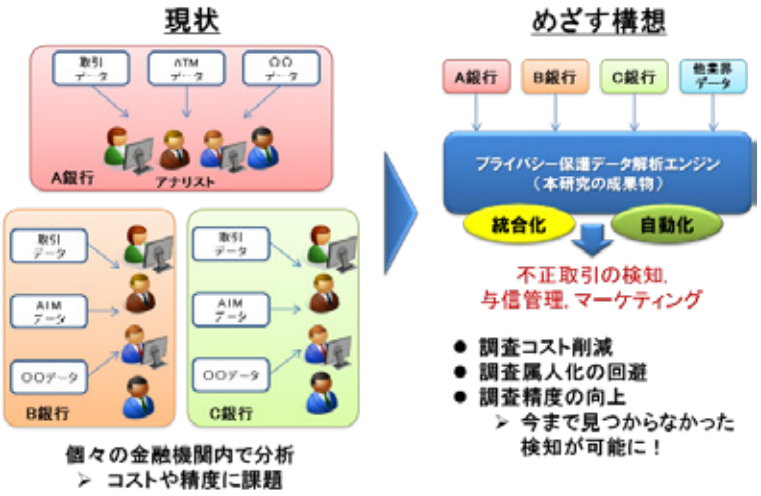


図 2.3-12　金融分野における課題と構想

図2.3-13は特殊詐欺といわれる例である。2018年のデータでは、１日１億円、年間364億円ぐらいの被害が発生している。特に昨今は、東京、埼玉、

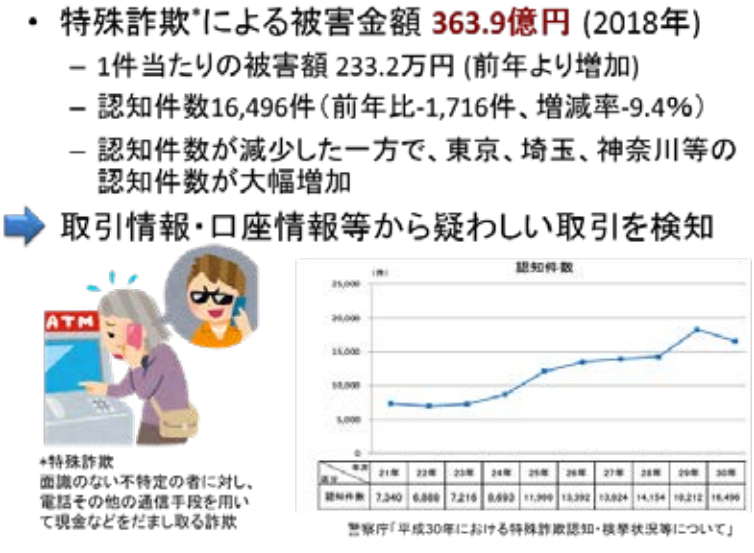


図 2.3-13　不正取引（振り込め詐欺等）検知

神奈川といったような首都圏で認知件数が増えている状況である。

われわれは銀行が持っている、何時何分に、どこのATMから幾ら下したというような取引データと、その口座情報をもらって、疑わしい取引を検知するということをやっている。

これは、ディープラーニングを使うのだが、データ量が重要で、少ないデータからだと、なかなかいいモデルをつくることができない。ひとつの銀行では学習データが不足しがちで、特殊詐欺も件数はある、被害は大きいとは言いながらも、１日何百件も起こるわけではないので、異常検知の件数として少なく、よい分析結果が得られないという課題がある。

では、銀行で集まって、連携してディープラーニングしたい、と言っても、クラウド上にお客さまのデータを預けることは難しく、できない状態が続いているという状況である。

われわれは、図2.3-14に示したように、外部にデータを開示することなく、複数の組織で連携してディープランニングを行うということを提案している。具体的には銀行の中で、閉じた形でそれぞれのデータを使って、こういうケースであれば不正であるということを学習する、ニューラルネットモデルをつくる。A銀行でも、B銀行でも、C銀行でもつくる。各銀行で学習したニューラルネットの重みの情報を暗号化してクラウド上にアップロードして、中央サーバーのほうで暗号化したままモデルを更新するということをやる。モデルの更新には足し算しか必要がないので、暗号化したまま足し算ができる加法準同型暗号を使って、モデルの更新を行っている。中央サーバーで更新された、自分以外のデータを使ってより賢くなった学習モデルを、各組織でダウンロードすることによって、精度の高い分析が可能になることを期待している。この場合、データは一切、銀行からクラウド上に上がっていないので、中央サーバーにも、第三者にもデータ漏えいの危険性がないというものになっている。

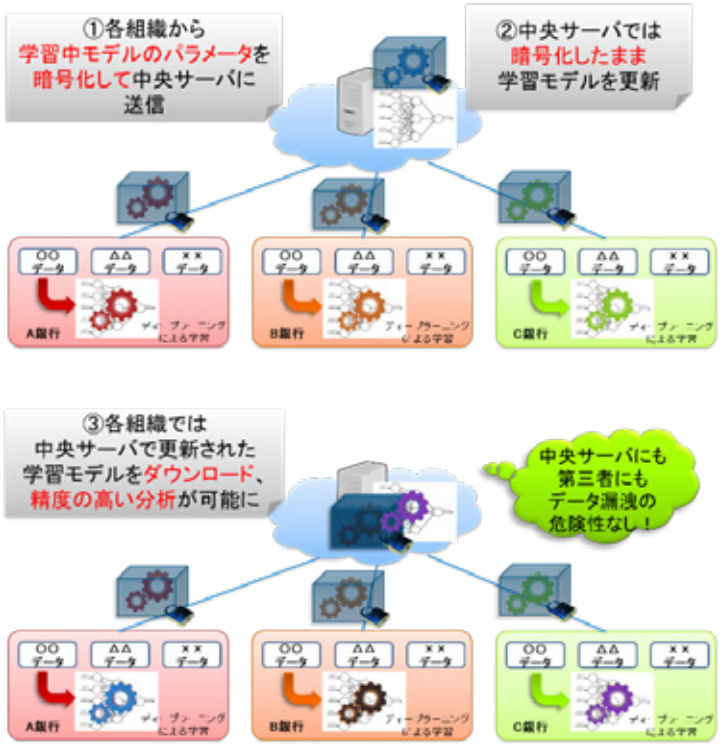


図2.3-14　外部にデータ開示することなく複数組織で連携してディープラーニングを行うには

オープンデータセットを使った実用性の検証もやっているが、最近は実際に銀行のデータをもらって、検証を行っているところである。図2.3-15に一例を挙げている。いろいろな機械学習で学習し、不正取引の検知の一例として、適合率85.5%、再現率62.1%といった結果が出ている。銀行の方からは、これだけ実用的なら実際に使ってみたいという話があり、どのように導入するかを検討している。

図 2.3-16 に不正取引検知の実証実験の状況を示している。各銀行と個別により良いモデルをつくるという実験をやっている。5 行以上の金融機関と連携し、検出精度 80% 以上で不正取引を検知するサービスをめざすという目標をプレスリリースし、参加する銀行を募集した。千葉銀行、三菱 UFJ 銀行、中国銀行、三井住友信託銀行、伊予銀行の 5 行と実証実験を開始した。これは CREST なので、来年度末までが期限である。

プライバシー保護データ解析技術で、幾つか実用的に使えるものが出てきているので、その技術を使えそうな分野とアプリケーションを特定し、その分野で相手を選んで、社会実装をさらに進めていきたいと思っている。

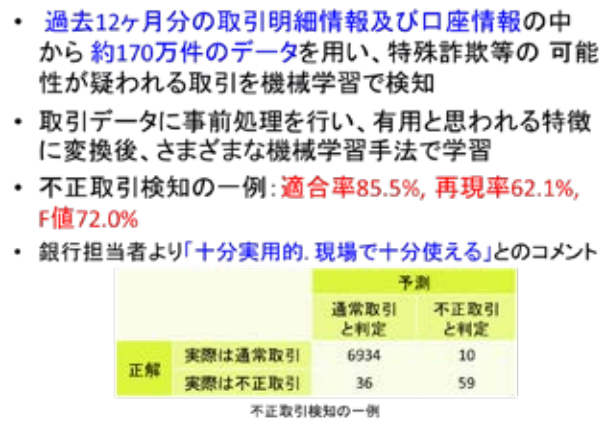


図 2.3-15　不正取引検知状況

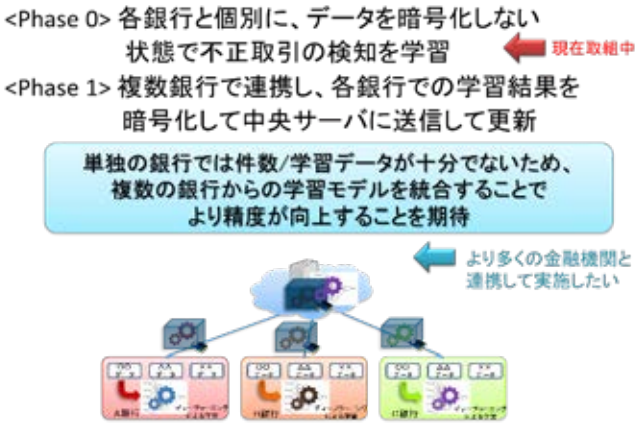


図 2.3-16　不正取引検知の実証実験

【質疑応答】　　（Q：質問、C：コメント、A：回答）

Q：図 2.3-9 の例にある、ある病気の罹患情報と遺伝情報の対応関係は、暗号化されているのか。

A：図の左側にある赤い表にある病気の罹患情報は暗号化して送る。図の右側の青い表にある個人の遺伝情報、具体的には、一塩基多型、スニップとの関連を、χ 二乗検定を暗号化した状態で計算している。したがって、遺伝情報を管理する機関では、計算内容を見ることができない。

Q：図 2.3-14 の学習結果の係数は、個人情報ではないのではないか。

A：学習係数から個人情報を計算することができないということを、論文で示しており、学習係数から、直接個人情報は出せないと考えている。

C：今日は、今できるところを話したが、暗号化したままでのデータ解析というのは、本当に難しい。

Q：性能とかも含めてか。

A：そのとおり。まだまだ課題は大きいと思っている。

2.4 プログラム言語分野 OVERVIEW（＋セキュア高性能データ基盤）

田浦 健次朗（東京大学）

プログラム言語分野とセキュアデータ基盤について紹介する。

Society 5.0 の私なりの捉え方を紹介する。データ／ＩＣＴ活用に加え、国民が等しく利益を享受していると感じ、安心できるためのデータ、システムのガバナンスであると単純に理解している。より具体的には、どこかの国のような権力による監視、少数の私企業による独占、プライバシー侵害、不透明なデータ利用などがなく、国民が安心して利益を享受できるものだ。

そのためのシステム基盤に求められる性質は、セキュリティと信頼性、透明性である。信頼性はしっかり動くとか、落ちないとかいう意味だけでなく、システムの中で何をやっているかを使う側が信用が置けるという意味も入る。それを担保するためには透明性が重要となる。よく分からないブラックボックス的なところがなければよいが、もしブラックボックス的なところがあっても、ある種の性質が外から検証できるという透明性である。

プログラミング言語の基本目標を全体として３軸で整理してみた。安全性は正しく動くこと、生産性は簡単に書けること、性能はしっかりハードウエア上で高性能に動くことを指す。この目標のために、手を変え、品を変え考えているのがプログラミング言語分野だと思う。

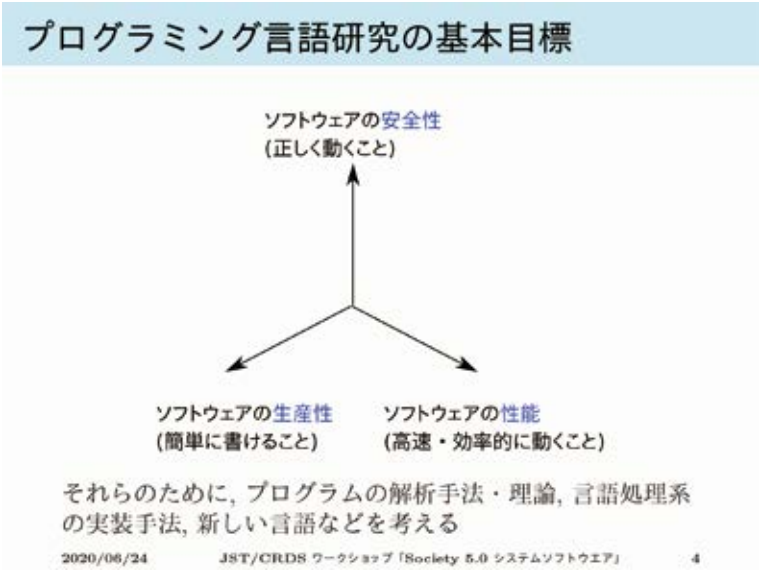


図 2.4-1　プログラミング言語研究の基本目標

安全性は大きい要素だが、ここでは全体のプログラミング言語における安全性についてのみ触れる。セキュリティは暗号やネットワークによく関連づけられる。プログラミング言語は全てのソフトウエアを書くための道具であり、しっかり仕様どおり実装されているかどうかを検証できるツールだ。世の中のセキュリティのフリーズを見ると、結局ソフトウエアがしっかり実装できていないというところに起因することが多い。プログラミング言語と安全性は非常に深く関わる。

ＰＬＤＩという言語の実装に関するトップ会議でも、セキュリティ、検証、静的解析、バグ発見などのプログラムの正しさに関するセッションが非常に多い。そのための基礎である型システムも大きなトピックになっている。

透明性の観点で、機械学習によって得られた部品の検証は大事な点となってくるだろう。与えられたデータ次第であり、中で何をやっているのかよく分からない。確率的に非常によく動くということで、部品として組み込んだソフトウエアが、システム全体に対して何

か致命的な性質を与えないことを少なくとも前もって証明しておきたい、という需要も重要になってくるだろう。

性能面ではムーア則が 2025 年、2028 年あたりに終焉するのではないかと言われている。そこで、さまざまなアーキテクチャーや、新しいデバイスを駆使した計算が必須だと言われている。その中で、今のアーキテクチャー上の区分、計算機の構成法上の工夫や、非フォンノイマン型などの話もある。

アーキテクチャー上の工夫に関して、オープンアーキテクチャーのようにハードウエアも自身で設計できる余地が増えてくるという期待が高まっている。

プログラミング言語的な観点からは、何にフォーカスしていけばいいかというのはかなり難しい。あるものに頑張り過ぎても、少し経つと何の役にも立てなくなっていることもある。多少は実用も近いと思われている量子アニーラは最適化のためのマシンだ。保険証とかでも有用な SAT ソルバーや SMT ソルバーなどにどれだけ役に立つか、立たないかをマシンへのマッピングで研究できないかという話はある。プログラミング言語として、大いに追求しがいがあるのではないかと思われる。

機械学習に関して、スパイキングニューラルネットワークが、消費電力の低さの点で、桁

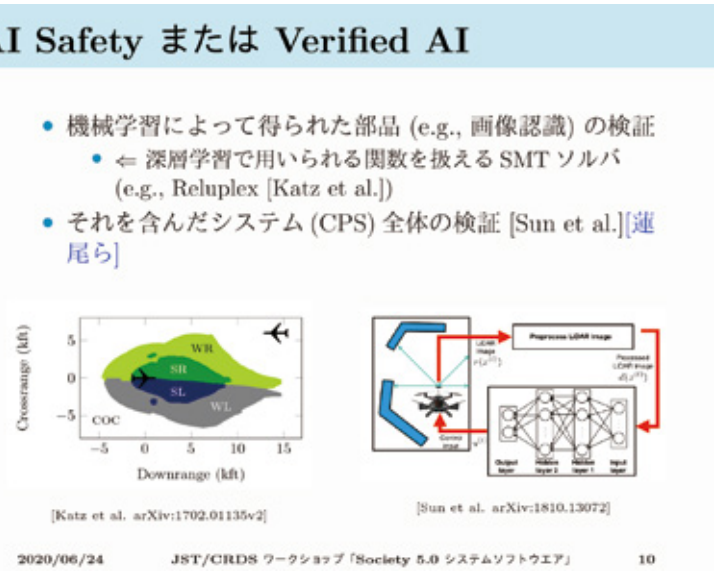


図 2.4-2　AI Safety について

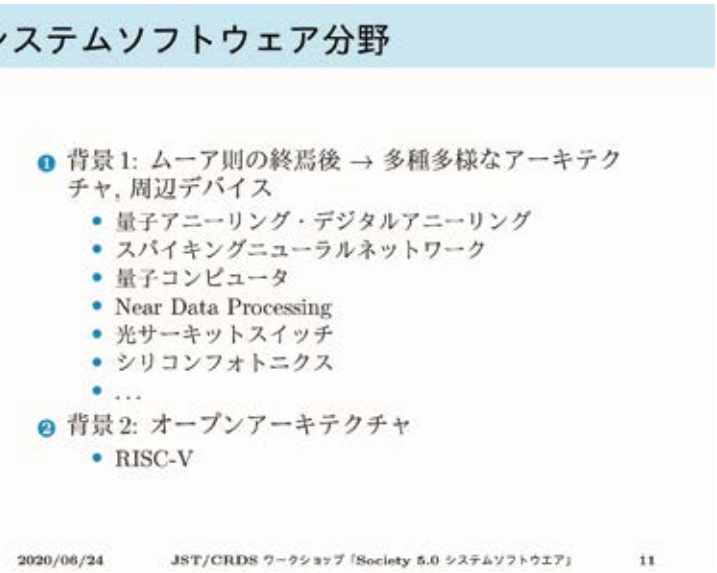


図 2.4-3　システムソフトウェア分野

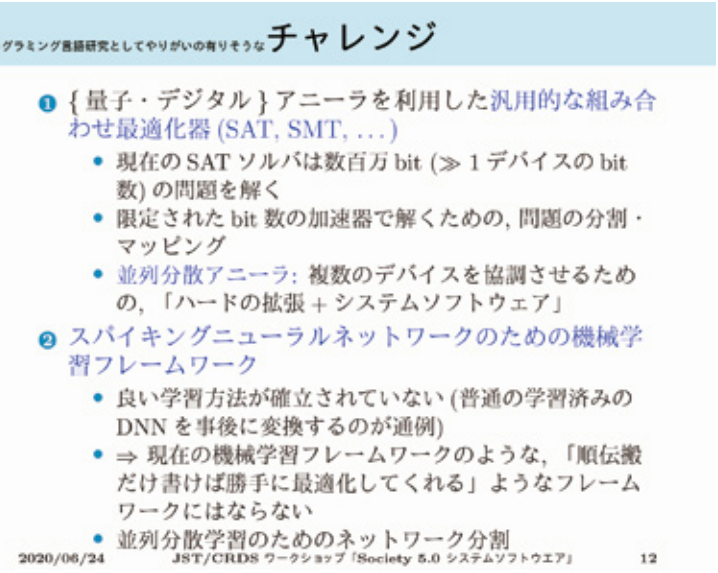


図 2.4-4　チャレンジ

違いの性能を示している。一方、学習をするためのよいフレームワークがない。機械学習が非常に誰にも使えるようになっている理由として、簡単なモデルを描くだけで、勝手に最適化してくれることがあるが、そういうフレームワークにはまだなっていない。追求するやりがいはあると思われる。

全体の関連図を示す。安全性という意味で非常に大きいセキュリティ、検証などがある。その基礎に、SAT ソルバーとか SMT ソルバーという計算量が非常にかかるものがある。それに対して、今後期待されるデバイスとして、量子アニーリング等がある。その辺りとプログラミング言語、システムソフトウエアをうまくつなぎ、領域を追求していけると面白いだろう。

ここから、セキュアデータ基盤の話題提供させていただく。

最近の政策動向で、文科省のデータ活用社会創成プラットフォームがある。データ科学やデータ活用のための基盤を提供し、産学官連携や社会実装を推し進めるための取組だ。現在、NII、産総研と、あと８大学で離陸している。ハードウエアは今年度末なので、まだ時間はかかるが、無事に立ち上がって、これからより広い連携体制を構築していこうとしている。

基盤を提供するという目的もあるが、それを使ったコミュニティ形成を目指している。学の情報系、非情報系という分野

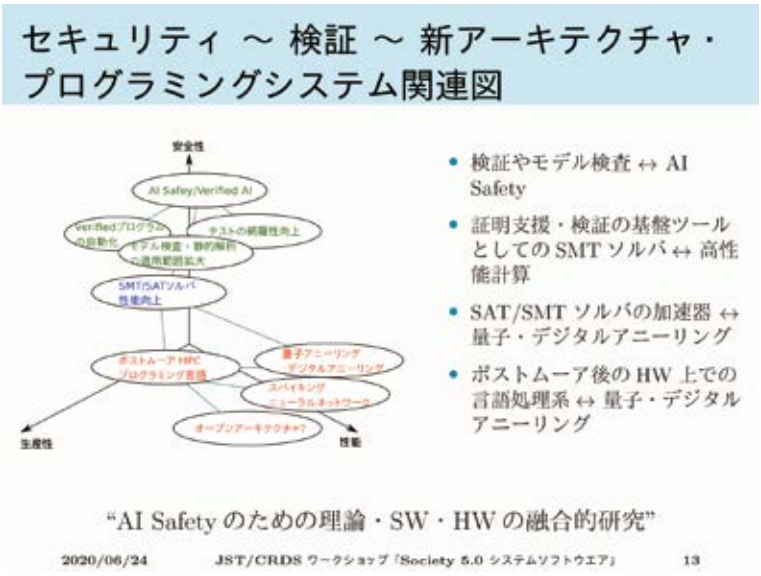


図 2.4-5　全体の関連図

図 2.4-6　データ活用社会創成プラットフォーム

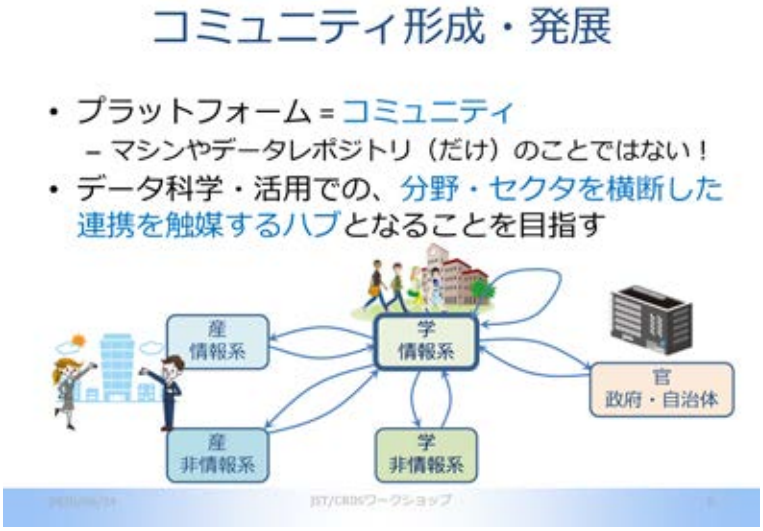


図 2.4-7　コミュニティ形成・発展

連携、産学官のセクタを結ぶ研究、社会連携、産学連携のための基盤、コミュニティをつくるハブという試みだ。

こういう試みについて、各大学の情報基盤センターでいうと、スパコンを中心として計算科学分野との結びつき、シミュレーション分野とのコミュニティ形成ということは長年やってきている。その分野を情報系も HPC だけではなく、データ科学や AI などに長けた人にも広げ、非情報系の応用研究もシミュレーションだけではなくデータ活用などまで広い分野に広げていこうという試みだ。

システムソフトウエアのシステムの部分について説明する。環境設計上の鍵だが、従来の HPC 要素のクラスタとかスパコンとは異なっている。これまでの環境を端的にいえば、単一の OS 環境に全員がログインして、バッチスケジューラで情報を走らせる環境だ。それでは済まないさまざまな要素がある。分野ごとのデータプラットフォームは地震関係、海洋関係、経済、人の移動、空間情報的などさまざまだ。そういうデータをコミュニティにサービスしたいという構想を持つ分野がたくさんある。分野データプラットフォームをホスティング、つまり連続稼働できるような環境だったり、そのために使われる非常に多様なソフトウエアを許容できたり、長年にわたってデータが蓄積・利用できたりすること、機微なデータを預かれる高いセキュリティを持った環境などを考えなければならない。

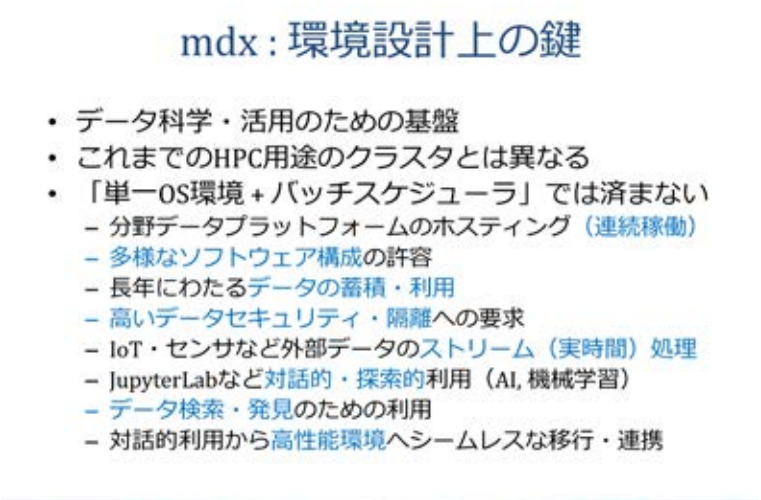


図 2.4-8　mdx：環境設計上の鍵

仮想プラットフォームで、個々のユーザーが自由に環境を構築できる。さらに NII の SINET、モバイル SINET と接続して、セキュアに IoT 環境が構築できる。高性能計算環境と連携している。mdx のシステム自身が高性能であることに加え、ほかの環境と接続できることが売りになっている。全体像を示す。端的に言うと物理的な基盤であることに加えて、個々の用途ごとに論理的にもネットワークレベルでも分離された、隔離された環境が手に入る。

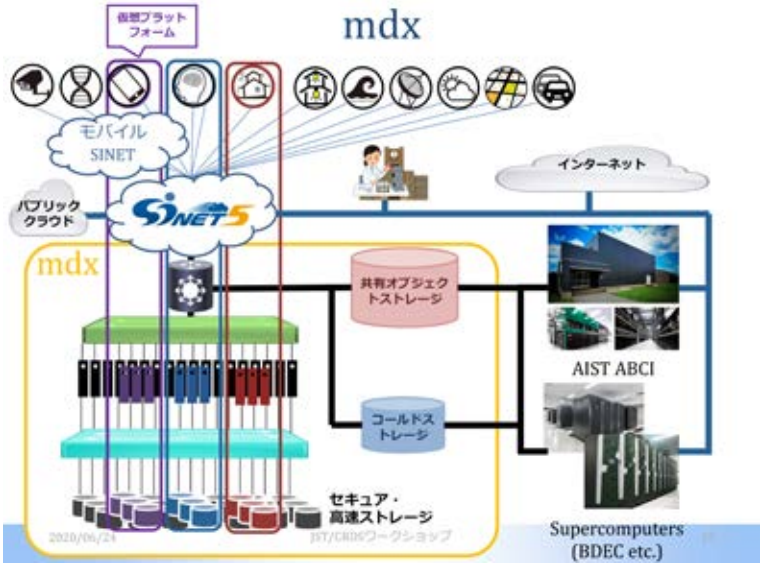


図 2.4-9　mdx の模式図

簡単に言えば、仮想化環境で疑似的に占有できる。1つの仮想プラットフォームが侵入を受けても、ほかへは影響しないという設計だ。NII の SINET やモバイル SINET と接続し、個々の仮想プラットフォーム用の VPN をつなげられる。IoT デバイスを野ざらしにしておくと、すぐにセキュリティ上問題になるが、そういった懸念を少なく実行できる。

このシステム自身の性能とあわせて、産総研の ABCI や東大のスパコンとも連携して使える。ここに研究のデータとして、いろんなものを提供していきたい。その取組としては NII も既に実施している研究のためのデータリポジトリのようなものをつくっていきたい。その1つがブログウォッチャー、いわゆる GPS データだ。人間の位置情報データを、システム上で提供する。これと感染データを合わせて、インフルエンザの感染モデリングを以前から行っていた。新型コロナではなく、インフルエンザなのだが、こういう取組がさらにできるようにしていきたい。

以上をまとめる。システム研究に対する価値として、機微データを含むビッグデータの蓄積、そのための連携ができるようなシステムにしたい。さまざまなデータの処理系や、あとは SINET の長距離広帯域を利用したシステム間連携として広域のファイルシステムとし

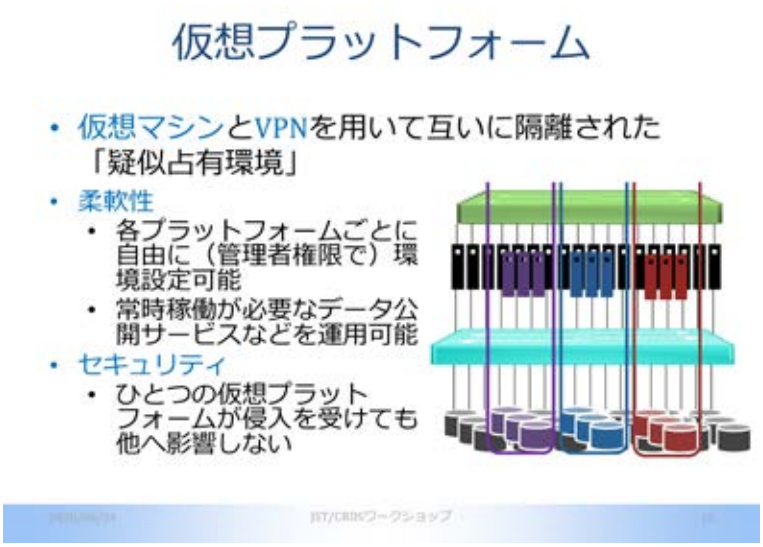


図 2.4-10　仮想プラットフォーム

図 2.4-11　ブログウォッチャー位置情報データ

図 2.4-12　GPS データと感染データ連携

た。既に NII が実施しているようなストリーム処理や環境の高速プロビジョニングもセキュア IoT として実施したい。システムを設計する上でさまざまな課題がある。既存製品にない要素を研究していきたい。

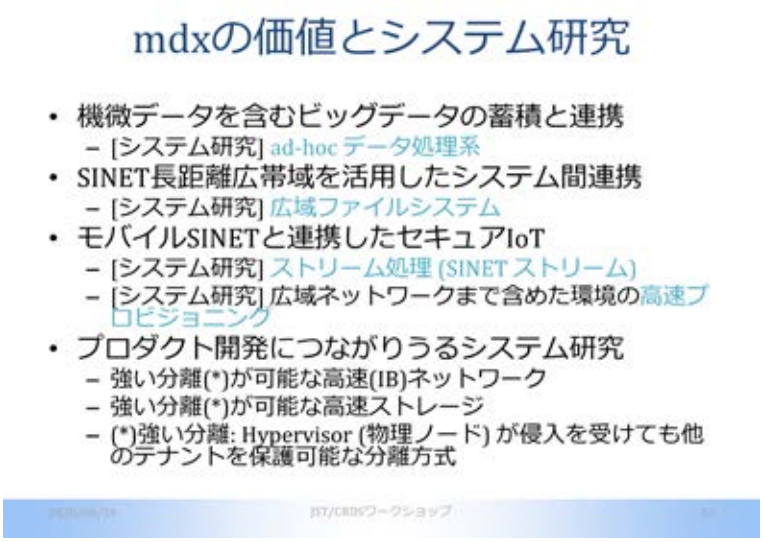


図 2.4-13　mdx の価値とシステム研究

2.5 （プログラム言語分野最先端研究）プログラム検証分野の動向

小林 直樹（東京大学）

プログラミング言語分野の中のプログラム検証というトピックにフォーカスして、私自身の研究を交えて話題提供する。簡単に自己紹介する。プログラミング言語の基礎理論と、その検証などへの応用について一貫して研究してきている。プログラミング言語には型という概念があるが、1990 年代中頃から、その型に基づいたプログラムの解析や検証をしている。セキュリティ分野については、プログラミング言語のアプローチから、ソフトウエアセキュリティについて研究してきた。例えば暗号通信プロトコルの検証や、プログラムの実行によって、機密情報が漏れないか解析する情報流解析と呼ばれるテーマなどの研究だ。2008 年頃からは、高階モデル検査にフォーカスしている。

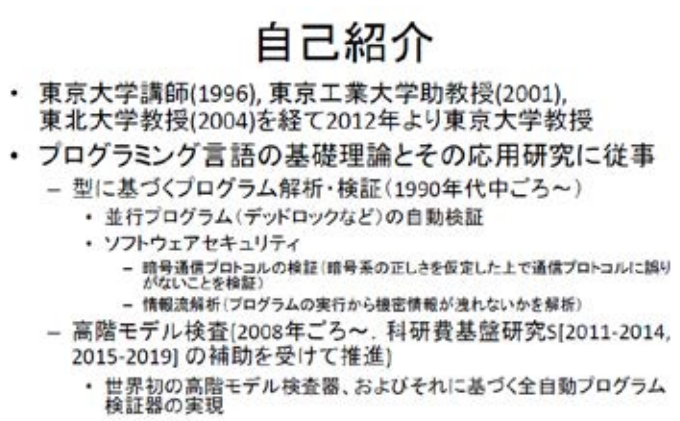


図 2.5-1　話題提供者の自己紹介

プログラム検証について、分野外の方向けに簡単に説明する。これは、プログラムが意図どおりの動作をすることを、数理科学的な手法を用いて、実行前に機械的、かつ網羅的に検証するものだ。関連する手法としてテスト手法がある。それとの大きな違いは、プログラムの正しさを、数学的に、理論的に保証できるという点だ。これについて、2002 年にビル･ゲイツが、コンピュータサイエンスの聖杯と呼んでいる。それぐらい重要な課題なのだと認識いただくとよい。

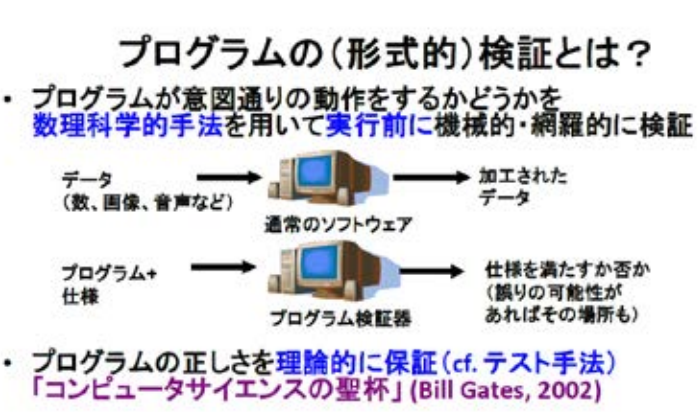


図 2.5-2　プログラムの検証について

私の研究室で作っている自動プログラム検証器のデモをお見せする。これはわれわれの自動検証器のデモ用のウエブインターフェースで、プログラムを入力して、ボタンを押すだけで検証できる。プログラムの説明は省き、何をしたいのかだけお話しする。assert 文（アサート文）という、C 言語などでもデバッグ用に挟むものがある。このアサート文が絶対フェイルしないということを検証したい。

入力が整数 n とすると、無限の値があり得る。１個１個の値についてテストしても、絶対安全だとは保証できない。そういう課題を数学的な方法を用いて検証したい。この場合、このボタンを押すと、検証が一瞬でできる。safe と出れば、このプログラムは、この

アサートコマンドでフェイルしないことを意味している。ここでわざと、ここにイコールを挟んでバグを仕込んでみると、今度は、ツールはアンセーフと出る。実際に入力は100のときに、こういう実行経路があってフェイルすると教えてくれる。これがプログラム検証器である。

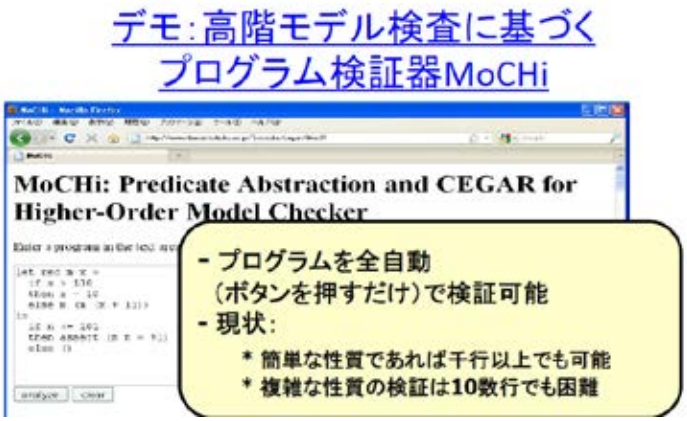


図 2.5-3　プログラム検証器 MoCHi のデモ例

お示ししたプログラム検証器の理論的な基礎になっているのが、高階モデル検査である。HORS モデル検査と HFL モデル検査という 2 種類の高階モデル検査がある。もともと代表的なシステム検証手法としてモデル検査と呼ばれるものがあり、2007 年のチューリング賞の対象になった。それを本質的に拡張したものである。普通のモデル検査は 1980 年頃に提唱されたが、高階モデル検査はずっと歴史が浅い。いずれも提唱されたのは 2005 年前後である。ただし、その時点では多重指数時間完全問題という恐ろしい計算量が既にあった。最初は理論家だけで研究されていたが、HORS モデル検査については 2009 年頃、HFL モデル検査についてもつい最近、われわれが実用的なアルゴリズムと、実際のプログラム検証への応用を見いだして、使えるようになった。

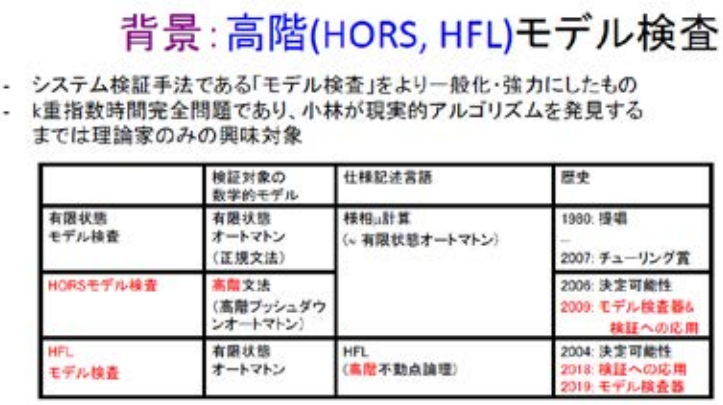


図 2.5-4　高階モデル検査

現状のプログラム検証技術がどうなっているのかご説明する。一言でプログラム検証といっても、ボタンを押すだけで検証できる全自動のものから、定理証明支援器を使って、人間が一生懸命証明を書くもの、コンピューター上に、そのソフトウエアが正しいと証明を書くものまで、いろいろある。

個々の詳細は触れないが、全体のイメージ図を示す。全自動の手法の代表的なものには、お示ししたようなモデル検査手法がある。こういう場合は自動化はできていて、人間が検証す

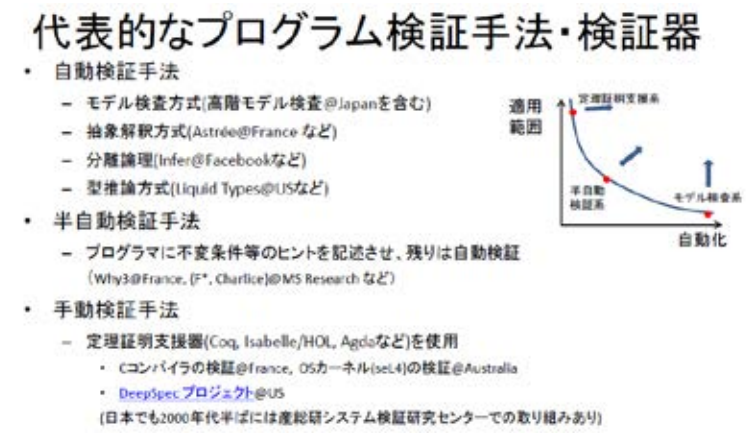


図 2.5-5　代表的なプログラム検証手法・検証器

る労力はほとんどない。しかし、全自動でやろうとすると、適用範囲や検証できるソフトウエアの規模などに、現状では限界がある状況だ。

一方、人間の手動で検証する、つまり人海戦術で一生懸命証明を書く手法は適用範囲が広い。Cコンパイラで頑張って検証したり、OSのマイクロ管理を検証したりという研究が実際にある。ただし、それにはものすごくマンパワーがかかる。

半自動の検証手法もある。全体のイメージ図の右上のほう、自動化もできて労力が少なく、大規模なソフトウエア、いろんなソフトウエアを検証できる方向が理想で、各手法とも向かっている状況だ。

世界の状況に少し触れる。例えばフェイスブックでInferという検証器をつくっている。アメリカのNSFで、この定理証明器を使ったDeep Specという大きなプロジェクトも立ち上がっている。

日本でも、2000年代半ばに産総研システム検証研究センターがあったが、残念ながら今はなくなった。

これらの検証器のインフラについて説明する。通称SMTソルバと呼ばれる自動定理証明器や、あるいは手動で人間が証明を書く定理証明支援器が一番ベースにある。その上にさまざまな手法ごとのライブラリやツールがある。その上に実際のプログラム検証器を構築するという形となる。

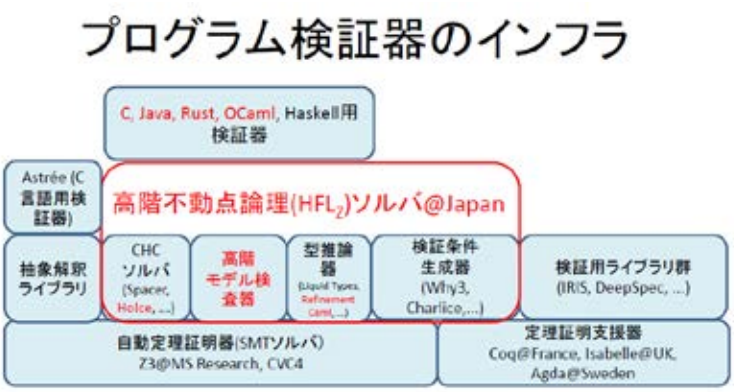


図2.5-6　プログラム検証器のインフラ

残念ながら、この自動定理証明器や定理証明支援器は、もう海外に完全に主導権を握られている。ほかの部分も正直に言って、海外勢が優勢だ。赤色の文字で示した部分あたりだけが、日本が主導権を握っているといえる状況だ。このような状況で、現在、われわれは核となるソルバをつくろうとしている。

今後の動向としてAIの普及とSociety 5.0にも触れる。

まず、AIの普及によって、われわれが研究者としても危機を感じているのが、ソフトウエアの質が根本的に変化しかねないことだ。今までは人間が書いたソフトウエアを一生懸命検証することを考えていた。しかしながら、ソフトウエアの大部分が、機械学習によって獲得されたコンポーネントが占め

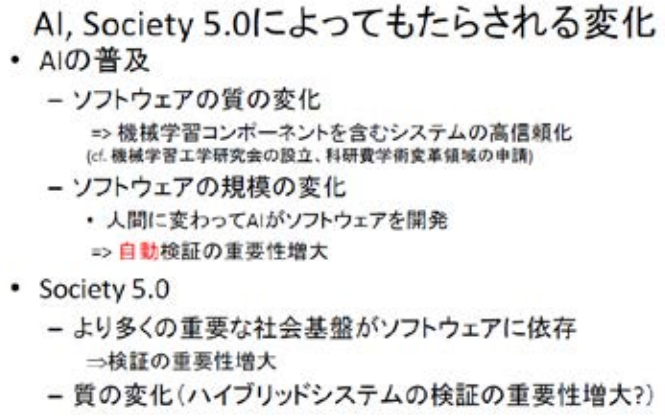


図2.5-7　AI, Society5.0による変化

るようになると、その検証の仕方は今までとまったく変わってくる。つまり、新しい取組が必要になる。実際に、海外でもさまざまな研究が既に始まっている。こういう問題意識から、国内でも機械学習工学研究会が設立された。科研費の新しい学術変革領域をつくろうという動きもある。

また、ソフトウエアの規模も変化する。今は人間がソフトウエアを書いているが、AI が生成するようになると、その規模がもうまったく変わってきてしまう。そうすると検証のアプローチとして、手動から自動まであると説明したが、手動のアプローチではもう絶対にスケールしなくなってくる。全自動、自動でやるしかないということになる。したがって、自動検証が、ますます重要になる。

Society 5.0 について、社会にコンピューターによる制御がより広がるという認識なので、ソフトウエアの信頼性というのは重要になってくる。するとわれわれが研究しているような検証の重要性も大きくなるだろう。また、質も変わってくるだろう。単純に純粋なソフトウエアの世界だけではなく、物理的なシステムと融合したハイブリッドシステムの検証について、重要性が増すのではないか。

本日の説明をまとめる。プログラム検証は今後ますます重要になる。ただし、残念なことだが、主要なインフラや理論のほとんどが海外に握られている寂しい状況にある。これを打破するには、何か予算を投入すれば済むという話ではない。やはり核となるオリジナルの理論や技術が必要である。そこについてはわれわれが研究者として頑張らなければいけない。

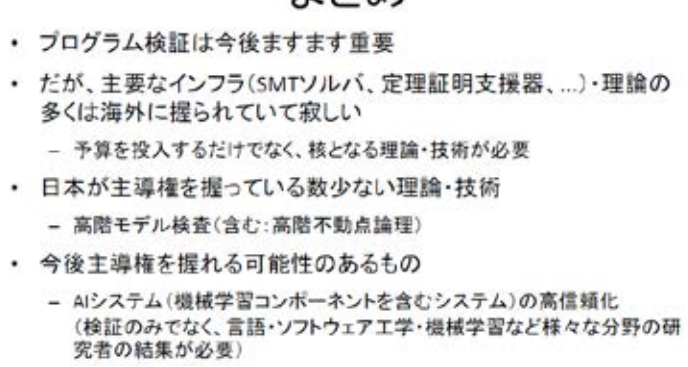


図 2.5-8　まとめ

日本が主導権を握っている数少ない理論、技術として、われわれが取り組んでいる高階モデル検査がある。今後、主導権を握れる可能性があるものとして、AI システム、機械学習コンポーネントを含んだシステムの高信頼化がある。そのためには検証のみではなく、プログラミング言語の設計とか、ソフトウエア工学分野、あるいは機械学習の人たちを巻き込み、さまざまな分野の研究者が結集するプロジェクトが必要ではないかと考えている。

2.6　コンピュータアーキテクチャーの研究動向～セキュリティの観点から～

井上 弘士（九州大学）

システム・アーキテクチャー研究会の井上と、若手の ARC の佐々木とで発表する。

ポイントが２点あり、アーキテクチャーというのは、高性能・低消費電力がよい、というところだけでなく、セキュリティも重要と考えて取り組んでいることを、皆さんに知ってほしいのが１点。

アーキテクチャー系は、海外では、セキュリティはアーキテクチャー分野で進んでいるが、日本では特に若い世代が頑張っているというのが現状で、今日のこのワークショップにも、アーキテクチャー分野の若手で、セキュリティ向けのアーキテクチャーをやっているという方々に参加してもらっている。

このオーバービューは、アーキテクチャー分野のセキュリティだけにフォーカスして、説明する。

図 2.6-1　ISCA で初の Security セッション
（図中の出典：https://www.iscaconf.org/isca2005/finalprogram.html）
2020年11月時点

コンピュータアーキテクチャーの分野で最も権威があるといわれている国際会議、ISCA（International Symposium Computer Architecture）というのがある。2005 年に初めてセキュリティというセッションができた。私はこれに参加していて、その座長が、David Patterson 先生だった。これは ISCA で最初のセキュリティのセッションであるとコメントがあった。これが来年は消えてなくなっていないことを祈るみたいなことをコメントで言っていた。そこからスタートして、セキュリティの話というのは、結構アーキテクチャーの分野でものすごく重要視されるように育ってきたという話である。セッション３のところで、セキュリティ系の、これは intrusion detection 系の専用プロセッサをつくるという話があるが、年間で、コンピュータアーキテクチャー系の中の論文で、トップピックスに選ばれている。

コンピュータアーキテクチャーの分野では、トップピックスというのが毎年、IEEE の MICRO から発表されている。これは各年にあったアーキテクチャー系のトップカンファレンスで発表した皆さんの投稿からインパクトがあったと思われる 10 ぐらいの論文が選ばれている。トップピックスというやつに選ばれるのは、採択率的には、多分 10％未満ぐらいであり、本当に各時代を代表する論文が集められるものになる。

これは 2005 年から 2019 年までどんなトピックがあったかというのを、調べた結果であり、赤字で示しているところがセキュリティ関係の話であり。2005 年に最初に、ISCA でセキュリティのセッションができて、そこでトップピックスに１本論文が選ばれているが、最近数年、2014 年以降は、かなりコンスタントにセキュリティ系の論文が、このトップピッ

クスの中に選ばれている。ここ３年は、毎年選ばれている。

今日この後に発表してもらう佐々木さんは、2019 年は残念ながらトップピックスには選ばれなかったが、次点に相当する、honorable メンションに選ばれていて、その話を今日は少し紹介していただく。セキュリティというのは、世界的にコンピュータアーキテクチャーの分野でも非常に重要視されている。

John Hennessy と David Patterson が、2017 年にチューリングアワードを取った際の、講演のタイトルで、「A New Golden Age for Computer Architecture」というのがあり、そこで１つのディレクションとして、Enhanced Security というのをうたっている。

これは MICRO という、コンピュータアーキテクチャーの世界では最も歴史のある国際会議で、一昨年、福岡で開催した際の論文投稿の状況で、青が論文投稿を示している。お化けみたいな絵が示すのがセキュリティ関係のところで、コンピュータアーキテクチャーの分野で上から５番目ぐらいに入る状況である。

なぜアーキテクチャーレベルでセキュリティなのかというのが、私は重要だと思っている。私見も入るが、基本的に守るべきデータは、回路もしくはデバイスが物理的にデジタルの信号として持っている。例え

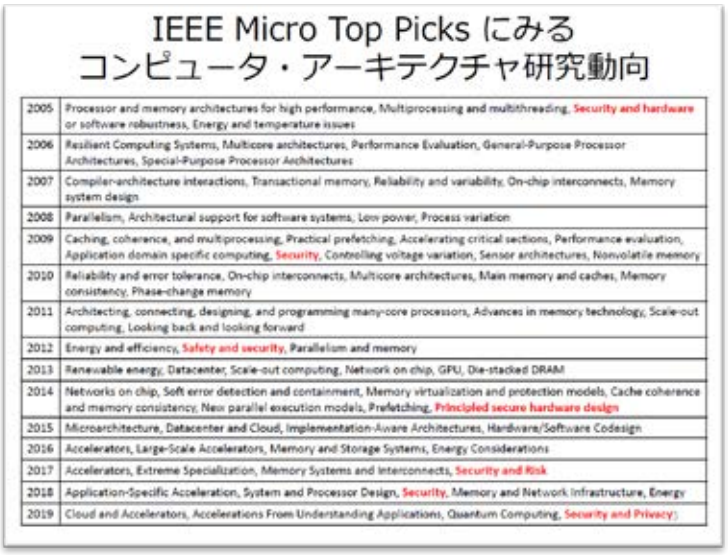


図 2.6-2　コンピュータ・アーキテクチャ研究動向

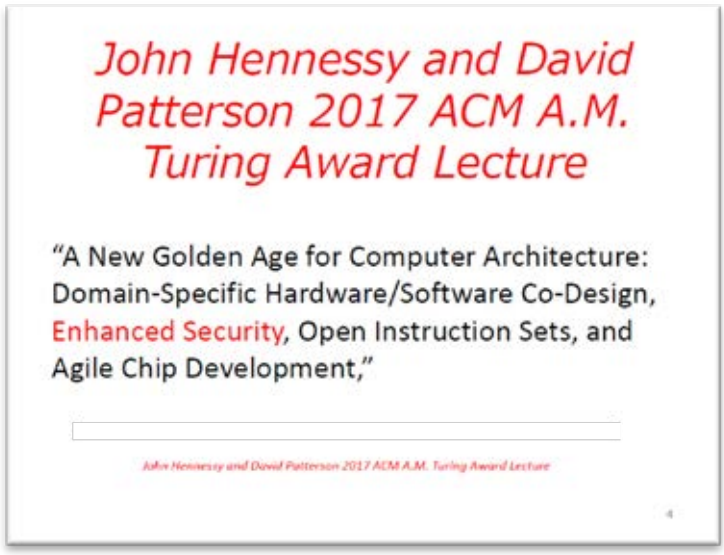


図 2.6-3　2017 Turing Award

(図中の出典：https://www.youtube.com/watch?reload=9&v=3LVeEjsn8Ts&feature=youtu.be&t=4268)
2020年11月時点

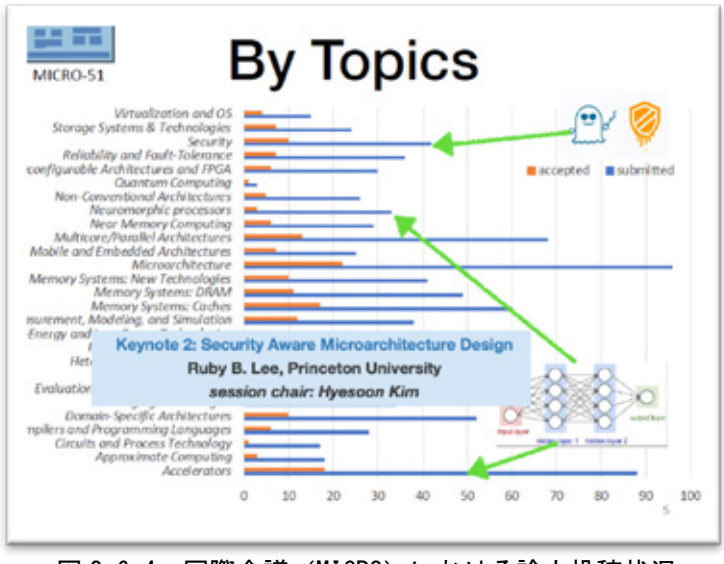


図 2.6-4　国際会議（MICRO）における論文投稿状況

ばマルウェアなんかも、結局最終的に回路もしくはデバイスで実行している。つまり、コンピュータシステムの中の物理的な世界というのが、回路、デバイスのところにある。一方で、その上のシステムソフトウエアや、プログラミングアプリケーションというのは、物理的な世界をモデル化した上で、論理的な世界で情報処理をするという構造になっている。このマイクロアーキテクチャーというのは、物理的な世界と論理的な世界をつなぐ、ちょうどインターフェースになっていて、最近 CPS という言葉がよく出るが、コンピュータシステムの中の CPS のちょうど境目を考えると、これがマイクロアーキテクチャーレベルになる。回路とデバイスレベルになると、論理値１か０というのは分かるが、その１か０が何を意味しているのかは分からない。これがマイクロアーキテクチャーレベルになると、論理的なプログラムの振る舞い等を検知することができるので、今やっている情報処理の意味を理解することができる。つまりコンピュータシステムの中で、情報処理の意味を理解できる最も物理層に近いレイヤーが、マイクロアーキテクチャーというレイヤーだと思っている。

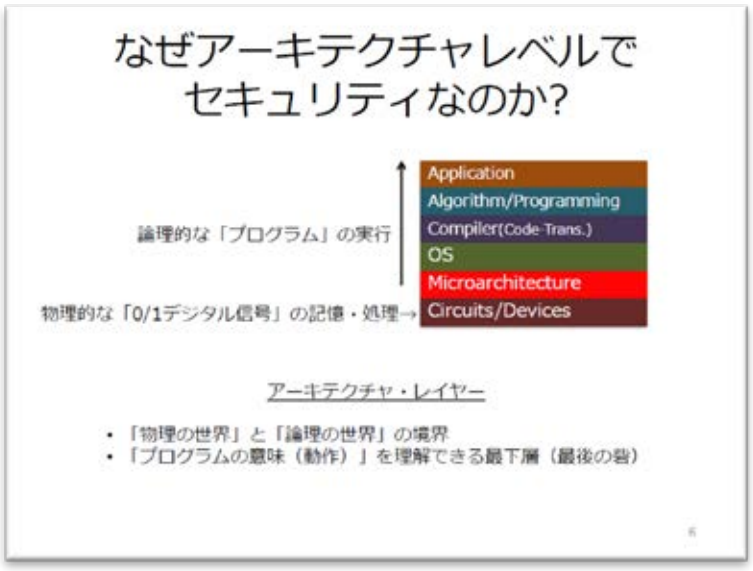


図 2.6-5　アーキテクチャレベルでのセキュリティ

従って、このレイヤーでいかにしてそのセキュリティーを担保するかというのが、結構重要になってくる。例えば基本的にこのマイクロアーキテクチャー、コンピュータアーキテクチャーといわれるところは、命令セットアーキテクチャーを決めて、ハードウエアを抽象化して、ソフトウエアレイヤーに見せるというのが大きな仕事であるが、基本的にその攻撃は、その命令セットアーキテクチャーの範囲の中で攻撃を仕掛けてくるし、プロテクションしようとするソフトウエア技術の命令セットアーキテクチャーの範囲の中で、どうプロテクションするかという、いろいろな手法を考えることになる。

ここでちょっと新しいハードウエアを追加して、セキュリティのための命令セットアーキテクチャー拡張みたいなのを考えると、よりソフトウエアレベルでのプロテクションが楽になったりとか、またちょっと違う観点からは、命令セットアーキテクチャーは変えない。つまり攻撃する側から見ても、命令セットアーキテクチャーというのは変わらないが、実はその下のレイヤーのハードウエア側でちょっとだけ仕組みを入れてあげると、がぜん、セキュリティの効果が上がるというようなことができる。

最後に、今回のこのワークショップで Society 5.0 のシステムソフトウエアというお題だと思うが、ぜひ、アーキテクチャーをやっている人から見ると、システムソフトウエアって、自分たちは違う人と思われるが、ぜひ、セキュリティアーキテクチャーをやっている若い研究者もたくさんいるので、そこの英知を活用して、Society 5.0 のためのシステムソフトウエアをつくる流れにできればなと思っている。

2.7 （アーキテクチャー分野最先端研究）研究紹介 Califorms

佐々木 広（東京工業大学）

計算機械アーキテクチャー分野の研究紹介ということで、コンピュータアーキテクチャー、セキュリティ分野で長らく大きな問題として君臨し続けているメモリ安全性問題というのに対して、アーキテクチャーの改良とシステムソフトウエアのサポートで対処する、セキュアかつ低オーバーヘッドな Califorms という技術について紹介したい。

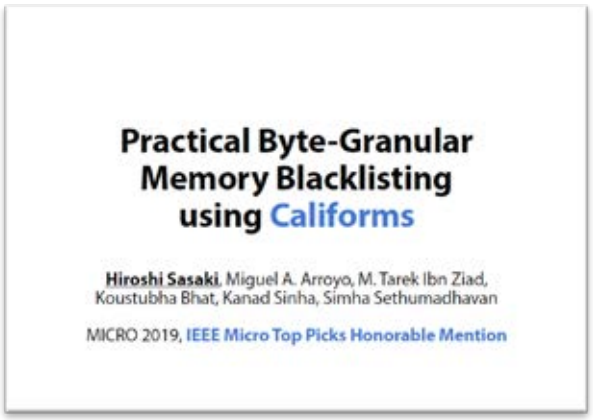


図 2.7-1

これは去年の MICRO という国際会議で発表した、「Practical Byte-Granular Memory Blacklisting using Califorms」という論文の紹介。Honorable Mention というものを受賞した（図 2.7-1）。

自己紹介：2013 年に IBM　TJ ワトソン研究所というところで訪問研究員をして、2014 年から昨年度までコロンビア大学で研究員をしていた。

まず、Practical は現実的で、Byte、粒度で、空間的に細粒度なメモリ安全性を提供する技術が、今回提案した Califorms というもの（図 2.7-2）で、今日はコンピュータセキュリティが専門でない方も多くいると思うので、今出てきたメモリ安全性というものに関して、このスライドにあるようなアウトラインで進めていく、ということで、最初にメモリ安全性について、簡単に説明する。

図 2.7-2

メモリ安全性は、Memory Safety の訳で、一般に大きく２つに分類されることが多い。日本語で、この訳が正しいかセキュリティ研究会の先生方にご指導を仰ぎたい。Spatial Memory Safety と Temporal Memory Safety という２つがあって、空間的と時間的メモリ安全性というふうにしたが、今日は時間の都合上、また提案手法のキーとなるアイデアを理解するには十分なので、上の空間的メモリ安全性にフォーカスして話をしたい。

最初に大まかに説明をすると、空間的メモリ安全だったら、空間的メモリ安全性が提供されているシステムでは、例えばバッファオーバーフロー攻撃の被害を受けない（図 2.7-3）。被害を受けないというのは、ひょっとしたら悪意のある人は、バッファオーバーフロー攻撃自体をできないかもしれないし、仕掛けることはできるが、システムがそれを検知して防ぐことができるかもしれないといった特性である。

最初に、バッファオーバーフローの説明を簡単にする。ここにまず攻撃対象のプログラム

があるとする。これは実行するとパスワードを聞かれて、正しいパスワードを入力するとシステムを破壊するというプログラムだとする。２つ変数があって、ユーザーが入力したパスワードを保存する配列の buf と、もしそのパスワードが正しかったときに、フラグを立てる is root という変数があるとして、今回コンパイルしたら、たまたまこの２つの変数が隣り合って、メモリ空間に配置されたとする。上のこの二次元で、ちょっと灰色になっているのがメモリ空間を示している（図 2.7-4）。

図 2. 7-3

ユーザーがこのプログラムを実行すると、パスワードを聞かれて、パスワード１って、これは正しくないパスワードであるが、この左のプログラムにある配列の最大のサイズを超えているパスワードを入力すると、上の配列にこの文字列が入っていって、この is root という変数を、１で上書きしちゃって、無事、システムが破壊されるとか、こういったものがバッファオーバーフロー攻撃のすごい単純化したものだと思ってください。

図 2. 7-4

紙芝居みたいなもので、空間的にメモリ安全の世界というのを紹介する（図 2.7-5）。都内に HS さんという若手の研究者がいるとして、ここに HS さんの家があると書いてあるが、こんな大きい家じゃなくて、何部屋かある物件を借りているとする。HS さんは物件を借りているので、自由にこの家に出入りすることができる（図 2.7-6）。これは何のアナロジーのつもりかというと、メモリ空間上にある自分のその変数が、該当しているメモリオブジェクトに関しては、プログラムがアクセスプログラム内からアクセスできるということである。地方に偉い KI 先生という人がいて、この人は、庭付きの大きな家に住んでいる。この人の家に勝手に HS さんが侵入しようとしても、これはできない。つまり

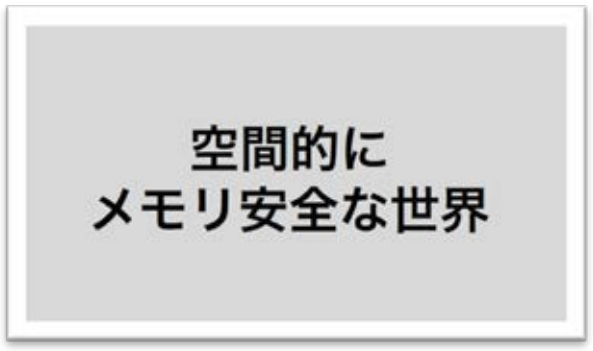


図 2. 7-5

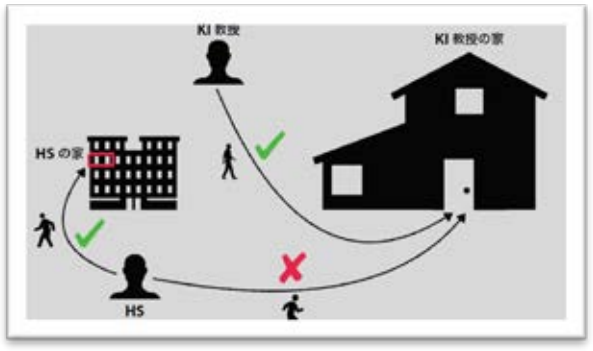


図 2. 7-6

ほかのオブジェクトをこのHSという配列の変数でアクセスすることはできない。当然、KI教授は自分の家なので、すたすたと入ることができて、これはつまりKIという変数だったら、このメモリオブジェクトを触ることができる（図2.7-7）。

細粒度とタイトルにあったが、これがどういうことかというと、1階に大きな部屋があって、これは実はKI教授の娘さんの部屋である。ここはいかにKI先生といえども、簡単にアクセスして入ることはできない。これが何かというと、先ほどKIは配列であったが、構造体だとしたら、K先生というメンバーは、この小さいKという変数にしかアクセスができなくて、娘さんの部屋にはアクセスできない。こういったことがシステムとして保証できていたら、空間的にメモリ安全であるというふうにいえるので、こういったシステムを提供したいと思う（図2.7-8、図2.7-9）。

メモリ安全性に起因するセキュリティの脆弱性をジェームス・アンダーソンさんという方が最初に1972年に書いたテクニカルレポートがある。これはもう約50年、つまり半世紀ぐらい前から知られた問題で、そこからいろんな人が研究してきたので、メモリ安全性問題は過去のものである、解決しているというのが理想であるが、これが本当かどうかというのを、僕が今日、グーグルを使って、いろいろニュースとかを検索してきたので、メモリ安全性問題の現状について紹介する（図2.7-10、図2.7-11）。

2015年ぐらいからのメモリ脆弱性のバグによる、バグをついた攻撃というのを調べてみたところ、毎年のように、先ほど田浦先生のスライドにもあったハートブリードのバグとか、Skypeのゼロデーとか、カショギ氏の暗殺に実はiPhoneにス

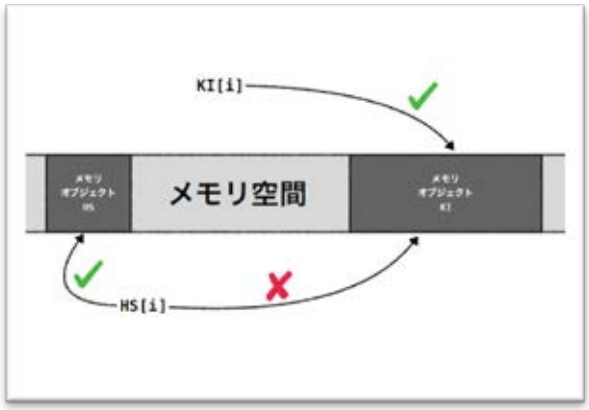


図2.7-7

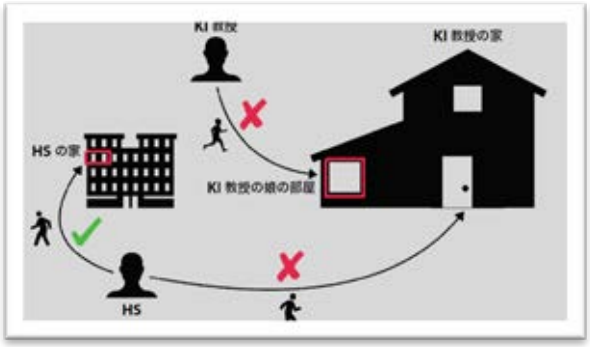


図2.7-8

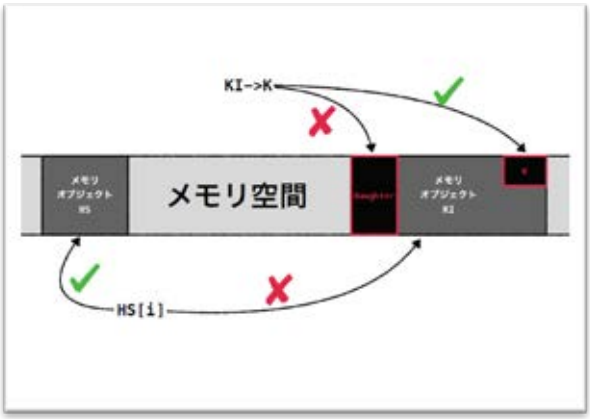


図2.7-9

図2.7-10

テルススパイウエアが埋め込まれたとか、たくさん事件の記事が見つかった。

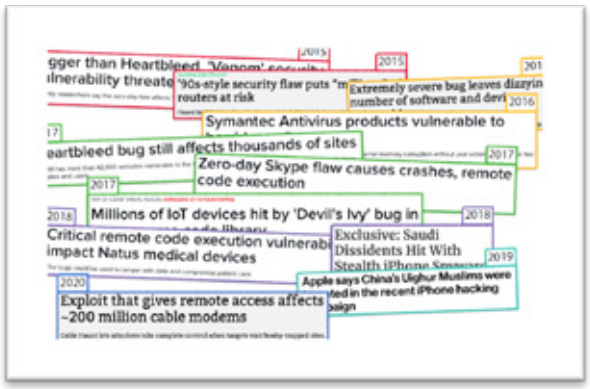


図 2. 7-11

また、テックジャイアントであるところのマイクロソフトやグーグルが出しているレポートみたいなのを見たところ、これは横軸に時間、縦軸に割合を示していて、黒い部分がメモリ安全性関連で、灰色の上の部分がその他であるが、マイクロソフトの Windows とか Office とか、セキュリティのパッチみたいのがよく来ると思うが、あのパッチの内容が修正されたバグが、メモリ安全性関連が何割ぐらいあるかというのをプロットしてみたもので、2006 年から 2018 年まで、大体 6 割から 8 割ぐらいがメモリ安全性関連だそうだ。同様に、実際に悪意のある攻撃者が、いきなり攻撃を仕掛けたという、ゼロデーのデータベースを、グーグルプロジェクトゼロというグーグルのセキュリティチームが公表していて、これも 2014 年から今年の 4 月ぐらいまでのデータがあるが、5 割から 100%ぐらい、メモリ安全性関連だそうだ。昨日、たまたま記事を見たが、クロームのバグも 7 割ぐらいがメモリ安全性関連である（図 2.7-12）。

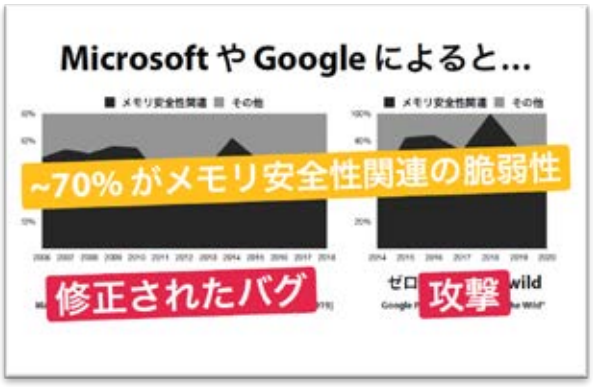


図 2. 7-12

このように、実際、約 70%ぐらいがメモリ安全性関連の脆弱性だということが、バグに関しても、攻撃に関しても言えるので、このメモリ安全性問題は過去のものであるというのは撤回して、現在も深刻であるということを、皆さんに理解していただけたかと思う。

図 2. 7-13

こういった問題に対する 1 つの答えとして、Califorms という技術を提案したので、これについて中身を、キーとなるアイデアについて説明する（図 2.7-13）。

最初に、現実的で、細粒度なメモリ安全性を提供する技術であるというふうに説明したが（図 2.7-14）、今書かれていた Califorms の特徴を大きく 2 つのクリアすべきチェックリストみたいなものにすると、1 つは細粒度のメモリ安全性を提供す

図 2. 7-14

ること、もう一つ、「現実的で」って左上にあったやつが、もう少し具体的に言うと低オーバーヘッドで、かつ実装が容易であるということになる。この２つに関して、どうやってこれをクリアしたかを説明する。

図 2.7-15

１つ目、細粒度のメモリ安全性（図 2.7-15）ということであるが、もう一度紙芝居を使うと、HS さんは、自分の家にはアクセスできるけれども、KI 教授の家に勝手に忍び込んでお金を盗んだりすることはできない。このためにどうするかというと、物騒な話ではあるが、この KI 教授の庭に、地雷を並べる。こうすると、HS さんが、KI 教授の家に行こうとすると、どかんと爆発して侵入が防がれてしまう。こういったシステムをつくる（図 2.7-16）。

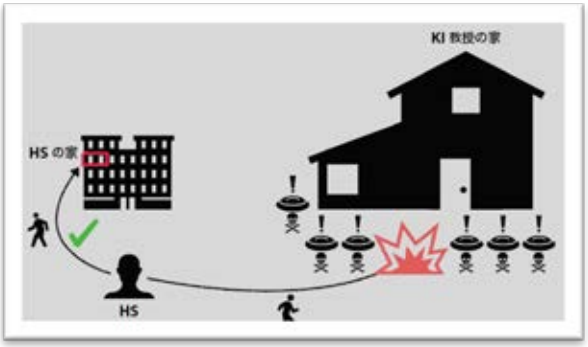


図 2.7-16

これをもうちょっとソフトウエアっぽく説明すると、先ほどのメモリ空間にオブジェクトが配置されていた図（図 2.7-17）であるが、このように細粒度なレッドゾーンというものをメモリ空間上に埋め込む。例えば HS という配列であれば、その両端をレッドゾーンでふさいで、KI という構造体は両端にレッドゾーン、中にもいろいろレッドゾーンを入れる。もう少し具体的には、例えば左にあるような構造体があったときに、われわれがつくったコンパイラのパスでトランスパイルして、各メンバーの間に、ランダムなバイト単位のレッドゾーンという char の配列を挿入する。それでこのレッドゾーンへ、もし CPU がアクセスしたら、それを検知して例外を投げる。これによって、空間的なメモリ安全性というものを提供する（図 2.7-18）。実際、この細粒度のメモリ安全性というチェックボックスに関しては、バイト粒度のレッドゾーン化（図 2.7-19）というのでクリアして、次に、この低オーバーヘッドかつ実装が容易という点について説明する（図 2.7-20）。

図 2.7-17

図 2.7-18

重要なのは、レッドゾーンへのメモリアクセスをどのように検知するかである（図 2.7-21）。例えば、この普通に正しいデータ

が入っているアドレスに、ロード命令をした場合は、システムはこれをチェックしてオーケーであるという必要があるし、レッドゾーンのアドレスに、もしロード命令がアクセスしようとしたら、もしくはアクセスしたら、それは駄目だというふうに言わなければいけない。つまり全てのメモリアクセス命令に対して、この判定をしなければいけないので、すごく高速に行う必要がある。実際、３命令から５命令に１命令ぐらいはメモリアクセス命令なので、ここを素早くしないと、性能へのオーバーヘッドが大きくなってしまうのは簡単に予想ができるかと思う。

図 2. 7-19

図 2. 7-20

理想のシステムについては、左に CPU があって、右にメモリがあったとして、縦に今度はメモリ空間を示しているが、CPU がロード命令を実行すると言ったときに、例えばアドレス、番地がレッドゾーンのバイトを含んでいるとする。そうするとアクセスしに行って、どかんとなって、例外を投げる、といったシステムが理想である（図 2.7-22）。

図 2. 7-21

これをソフトウエアで実装しようと思うと、一般的にどういった手段があるかというと、同じようにロード命令を実行しようとして、ここにいきなりアクセスしても、果たしてそれがアクセスしていいアドレスなのかどうかというのは分からない。どうするのが一般的かというと、大体、メモリ空間のどこかに領域を確保しておいて、そこにメタデータとして、どのアドレスはアクセスしていいよとかいったデータを格納しておいて、まずここにお伺いを立てて、このアドレスはアクセスできるかと聞いて、駄目だと言われることで、次にアクセスしようとしたときに、ちゃんと例外を投げるなり、何か処理をするなりということができる。つまり、このやり方だと、メモリアクセスの数が純粋に倍になるので、性能のオーバーヘッドが２倍とかになるので、プロダクションで使うものとしての性能を達成することができない。

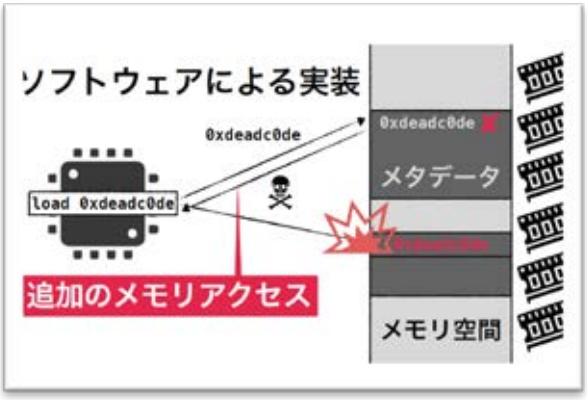


図 2. 7-22

そこでハードウエアサポートが必要になる（図 2.7-23）。ハードウエアキテクチャの研究をしている人には、この絵というのは、もうちょっと細かく見ていくと、間にキャッシュが、ハードウエアキャッシュと呼ばれる機構があって、データは実際メモリから取ってくるときに、キャッシュラインという、大体 64 バイトぐらいの大きさのサイズのデータの塊でやり取りをするわけである。キャッシュラインというものは、今、黒い箱で示しているが、中には 64 バイト分のデータがあって、ここでは 1 つの箱を 1 バイトだと取りあえず思っていただいて、そうすると、僕たちのコンパイラがレッドゾーンを挿入しているので、このキャッシュライン内に、レッドゾーン化されたバイトがある。

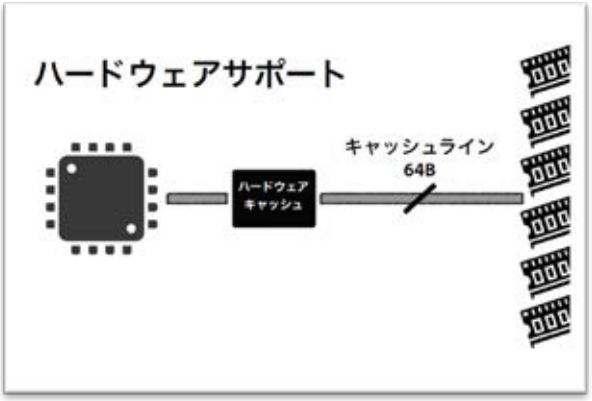


図 2.7-23

図 2.7-24

考えてみると、正規のプログラムのアクセスは、このレッドゾーンのバイトをアクセスすることは絶対にないので、つまりこのバイトのそれぞれの中に入っているデータはごみであり、何でもいい、デッドスペースである。そこで、あえてこのデッドスペースを活用して、デッドスペース内にメタデータを格納してやるというのが、僕たちが考えたアイデアである（図 2.7-24）。そのためにデータのフォーマット、キャッシュラインのフォーマットを変更することにして、それがニューキャッシュラインフォーマットということで、一部の文字を取って、Califorms という名前になった（図 2.7-25）。

図 2.7-25

左側が普通のキャッシュラインで、右側が僕たちが提案する新しいキャッシュラインフォーマットで、まず 1 ビット追加のハードウエアがあって、これはキャッシュライン内にレッドゾーンがあるかということを示している。ここではあるので、1 である。このレッドゾーンをキャッシュラインの先頭のほうに持ってきて、パッキングする。これをパケットとかもヘッダみたいにして扱って、ここにレッドゾーンの位置をエンコードする。こうすると、追加のメモリアクセスというのが不要になって、データを取りにいったときに、同時にそのアドレスがアクセスしていいかという判別ができるので、レッドゾーンをアクセスしているかどうかの判別のオーバーヘッドがほぼゼロになる。こんなことをすることによって、低オーバーヘッドかつ実装が容易ということを実現した（図 2.7-26）。

今の２つが基本的に核となる技術で、あとは全体像について簡単に述べるが、下からマイクロアーキテクチャー、アーキテクチャー、システムソフトウエアとあるときに、今のキャッシュの変更と、実際にデータのフォーマットを変更する回路を含んで、キャッシュアーキテクチャーにちょっと変更を加えた。次に、実際に誰がそのデータフォーマットを変えるかというと、CPU が指令をしなければいけないので、１命令だけ追加の命令を出す。最後にシステムソフトウエアでは、先ほどのレッドゾーンを挿入するコンパイラのパスと、あとスタック上のデータにもしレッドゾーンがあったときに、その追加の命令を発行しなければいけないので、コンパイラの変更と、あとメモリアロケータが、キープ領域にメモリを確保したときに、その両端にレッドゾーンを入れたり、キープのオブジェクト内にレッドゾーンを入れたりということをするので、そのための命令発行するロジックを追加して、あとはレッドゾーン化されているページをスワップアウトしたときに、OS がその情報を自分の管理空間にとったりとかいうことをするので、OS のサポートというのを足している（図 2.7-27）。

図 2. 7-26

図 2. 7-27

性能について触れるが、全てのメンバー変数の間にレッドゾーンを入れると、それが一番セキュリティ強度的には強くて、もう少しセキュリティ強度の落ちるポリシーを２つ用意して、３つ、評価を行って、SPEC　CPU2006 という、CPU の性能を測るベンチマークで性能を評価したところ、大体平均のオーバーヘッドが、セキュリティ強度が一番弱い設定が２％ぐらいで、全てのメンバー変数の間にレッドゾーンを入れても、オーバーヘッドが 14％ぐらいで済んだというのがまとめとなる（図 2.7-28）。

図 2. 7-28

最後にまとめをすると、このようなアウトラインで今日は話してきて、上２つはメモリ安全性について話したが、少なくとも僕たちから見えているコンピュータセキュリティというのは、まだまだ問題が山積みで、例えばここ 10 年ぐらい研究が行われているメモリ安全性すら、まだ満足な解がない。それに加えて、最近はハードウエアの実装による脆弱性というのが報告されていて、先ほど井上さんのスライドでお化けがいたが、あれがスペ

クターっていう、投機実行するプロセッサの実装に起因するバグみたいなものである。もう一つ、RowHammer という DRAM の、何か近傍のデータをたくさん読み書きすると、触っていないはずのデータのビットが反転してしまうというものがあって、やることがたくさんある。

図 2.7-29

僕たちが提案したこの Califorms というのは、アーキテクチャー手法によるセキュアかつ低オーバーヘッドな手法の一例であるが、基本はやはりこのハードウエアの改良とシステムソフトウエアが手を組んだというところだと思っていて、長らくセキュリティはソフトウエアの研究者の方々がたくさん研究しているが、どうしてもやっぱりハードウエアの力を借りたい局面というのがあると思う。今回、例えば僕たちがやった性能オーバーヘッドをどうしてもゼロにしたいというときに、ハードウエアの変更が必要だったということである。それを考えるときには、ハードウエアのアーキテクチャーの研究をしている僕たちに一日の長があると思うので、ぜひ力を合わせて、アーキテクトとシステムソフトウエアの研究者が手を取って、よりセキュアなシステムを、Society 5.0 であるが、それとか IoT とかに向けてつくっていく必要があるかなと感じている（図 2.7-29）。

【質疑応答】　（Q：質問、C：コメント、A：回答）

Q：レッドゾーンを飛び越えてほかのデータにアクセスすることはないという前提か。

A：そうである。前提はそうではないが、ほとんどの場合は、やっぱり連続アクセスで来ると思っているというのがまず１つで、飛び越えるには、当然、このレッドゾーンの大きさを予想しなければいけないわけで、そこがランダムなバイト数をコンパイラが入れているということで、飛び越えるにしても、正しい量を飛び越えないと目的の変数なりデータにアクセスできないというのが、一応これに対する回答となる。

Q：memcpy については。

A：memcpy はちょっと特別なので、memcpy するときには、レッドゾーンを１回外す。つまり幾つか、こういうどうしても避けられないオペレーションに関しては、アロウリスト、許可する関数が幾つかあって、memcpy もその１つになっている。

Q：DARPA の DeepSpec プロジェクトの中でも、CPU アーキテクチャーを拡張して、アクセス権限なども含めたメモリ安全性を保証する研究があったと思うが、どのような関係にあるか。

A：これについて把握していないので、チャットで回答する。

Q：空間オーバーヘッドについては。

A：これはどの程度、もちろんレッドゾーンを挿入するかによっていて、今手元に数字が

ないが、ハードウエアの空間をオーバーヘッドという意味では、64 バイト当たり 1 ビットである。実際にそのプログラムにどのぐらい入れるかというのは、これは完全にプログラム依存であるが、例えば先ほどメタデータをソフトウエアで、どこかの領域に格納するといったものと比較すると、随分小さいことが多い。

Q：バッファオーバーフローを防ぐＣコンパイラの研究はたくさんあると思うが、ハードウエアを改造して、どれぐらい速くなったか。

A：ソフトウエアで、僕たちが入れているぐらい細粒度に、実際バッファオーバーフローを防ぐことを実現しようとする研究というのと比較すると、例えばイントラオブジェクトオーバーフローって呼んでいるが、例えば構造体やクラスの中のオーバーフローというのを検知するためのソフトウエアは、実際にやろうとするとオーバーヘッドが何倍とかになってしまうと思う。実際に動いているシステムがあるわけではないが、その細粒度に入れるという意味では、比較できるソフトウエアがなくて、本当に簡単な、例えばスタックカナリとかだったら、オーバーヘッドはすごい小さいところで、何と比較するかによるが、僕らが提供しているぐらいのセキュリティ強度でやると言われると、14％というのは、例えば 100％とかと比べるとどうかというところである。

Q：空間メモリアタック以外にどのような難しい研究領域が残っているか。

A：メモリ安全性とは関係のない研究領域の話であると、先ほど最後に述べたが、ハードウエアの研究者としては、ハードウエアに依存する脆弱性であるスペクターと、RowHammer に関するトピックが非常にホットになっている。もちろんそれらを使って、攻撃をソフトウエア的に仕掛けることができるので、どこで止めるかということを考えると、ソフトウエアとの関連があるとは思うが、ハードウエアのアーキテクチャーの研究者としては、そこが一番ホットではある。

2.8 プライバシー保護分野 Overview & 最先端研究

佐久間淳（筑波大学）

プライバシー保護分野のオーバービューと最先端研究の両方についてお話しさせていただく。まず、最近のトップカンファレンスの傾向から、どういった研究が行われているかを私なりに解釈してまとめたものをお示しする。

Running example として、コンタクトトレースという、非常にホットな話題を取り上げる。まず、病院側が持っているような、どの病院で、どんな人が感染症に対して陽性であるか、陰性であるかというデータがある。一方で、例えば Suica の履歴や、プラットフォーマーが持っているスマホから取得した人流の情報といった人流データがある。これらデータを集約することで、感染者が今どこにいるのか、などがわかる。あるいは、陽性者の動きを見ることで、どのように感染が広がったのかわかるので、濃厚接触した未感染者にアラートを出すようなユースケースが考えられる。この際、プライバシーの問題が山ほどある。誰が陽性か陰性かといった話は、簡単には外に出せない。特に個人ＩＤがくっついている点が問題を難しくしている。誰が、いつ、どこへ行ったのかという情報も非常にセンシティブである。こういったタイプの情報を扱った統計解析を行う際、プライバシー問題の解決が大きな課題となる。

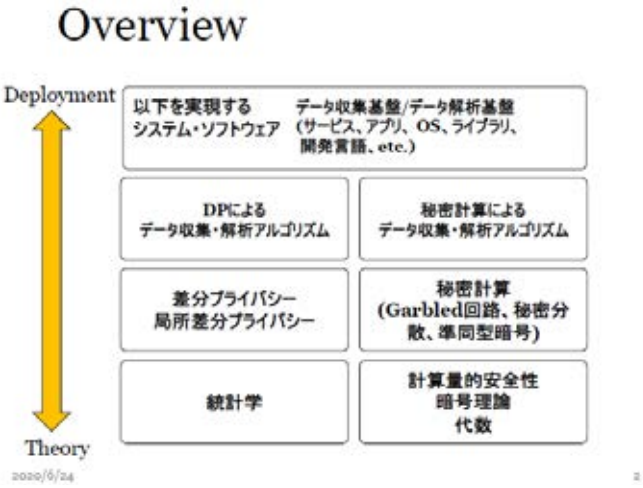


図 2.8-1 Overview

差分プライバシー技術について説明する。端的に申し上げると、大量のデータがあったときに、統計解析の結果を一般に公開する際に守られるべきプライバシー基準と考えて頂ければと思う。差分プライバシーが想定するリスクとは、外に出ていった情報から、リバースエンジニアリングのような形で元々のデータ、例えば誰が陽性であったか、といった情報が逆に推測されてしまわないかというリスクであり、それに対応しようとするのが差分プライバシー技術である。簡潔に申し上げると、実際にやることは簡単で、乱数を使って解析結

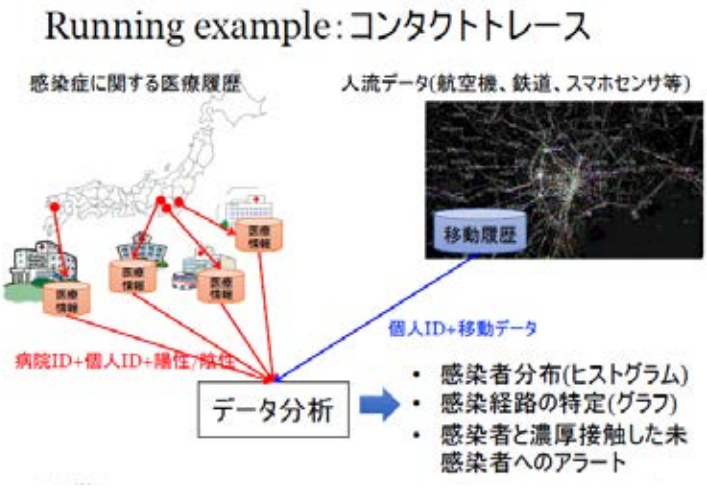


図 2.8-2 Running example: コンタクトトレース

果をランダム化し、少し汚すことで、誰が陽性だと確定的に推測できた状態から、誰が陽性なのか確率でしかわからない状態にもっていくことでプライバシーを守ろうとする技術である。

近年、大きく広がっているのが、局所差分プライバシーである。こちらはデータを収集するときに使うプライバシー基準である。先ほどは、データを集める人は trusted で、外に出ていった統計データを受け取った人は untrusted だという設定であったが、局所差分プライバシーではデータを集めている人も untrusted だという設定で、データをサーバに提供する人が、ランダム化してデータを少し汚した上で提供するというフレームワークである。そうすることで、収集者が個人のデータを推測しようとしても、一定の確信度でしかデータを推測できない。その解析結果をもらった人も、個人のデータを確実に推測することはできなくなる。

差分プライバシーへは、統計量を一般公開する。あるいはデータを公開するといった際のプライバシー保護のデファクトスタンダードになったとも言える状態だと私は思う。局所差分プライバシーは、例えばグーグルとか、アップルなどのプラットフォーマーに、個人がデータを提供するときに、プライバシー保護のために選択し得る数少ない手段であるため、重要度は高いと考えている。アップルは iOS デバイスユーザーからユーザー統計を収集する。グーグルがクロームソフトウエアを通じてユーザーの統計を収集する。マイクロソフトは Windows がユーザーのテレメトリデータ、例えばユーザーがどんな操作をして、どんなところでエラーを出した、といったデータを収集する。それらデータ収集の際に使われている。

アップルとかグーグルによって、LDP、局所差分プライバシーがどう使われているかを示す。アップルは、iOS デバイス上で、ユーザーが例えばどの Web サイトを見たか、チャットをするときにどんな文字を打ったか、などのデータをそのまま吸い上げるのではなく、デバイス内で一度ランダム化して、ランダム化されたデータをサーバに収集する。Ingestor というサーバの段階ではインターネットコネクションがあるため、デバイスの識別子などが取れているが、ここでデバイスの識別子を外した上でサーバに送る。統計情報は識別子がなくても取れるため、頻度統計などをサーバで計算する。グーグルはクロームにそういう仕組みを取り入れており、RAPPOR と呼ばれている、似たような仕組みでデータ収集をしている。

どのような分析をしているのかというと、グーグルの場合は、Bloom filter というデータコードを使い、プロセスの頻度統計や URL の頻度統計などを計算している。アップルは、チャットで使われている絵文字の頻度統計や、Web ページや動画サイトを見たときに、音声を自動再生するかどうかなどの統計を集めているようだ。

局所差分プライバシーによって、どういった方向性の研究が進んできたかを説明する。最も簡単な例として、heavy hitter、要するにヒストグラムが挙げられる。例えば１人１台スマホを持っており、所在情報を広く吸い上げると、どこに何人いるのか、というヒストグラムが集められる。こういった推定に局所差分プライバシーを使った場合、どのくらいのユーザーがいれば、どのぐらいのエラーの推定ができるのか、といったことが見えてくる。推定誤差がどれだけ大きいのかをきちんと把握したうえで、より良い推定ができるかどうかを知ることが重要である。例えば、通信量が１ビットであるメカニズムにおいて、それ

が最適であると分かっているようなヒストグラムの推定を達成したと言う論文がある。非常に単純な推定問題では、既に最適であることが分かっているような、プライバシーを守りつつ、ウエートが最適であるようなアルゴリズムが分かっている。

局所差分プライバシー下で統計的推定をやる際に考えるべき問題について。どのようなタイプの統計的推定をやりたいのか、どのような対話モデルでユーザーとインタラクションしようとしているのか、などで問題が定義される。統計的推定としては、最も単純なものではヒストグラムや平均値の推定に始まり、メジアンとか線形回帰みたいな統計もあり、最も複雑になると深層学習のようなものが含まれてくる。

対話モデルとは、どのぐらいの複雑さをもってユーザーと対話をしていくのかのモデルである。最も単純なものは、非対話モデルと呼ばれるもので、ユーザーが何も考えずに１回だけ情報提供をして終わり、というモデルである。公開コインは省略する。逐次対話モデルとは、例えばユーザーが 100 人いたら、ユーザー１からまず情報をもらい、ユーザー１の情報を少し加工しユーザー２に送り、ユーザー２はそれに依存して、自分の情報を送って、というのを 100 人分やるモデルである。

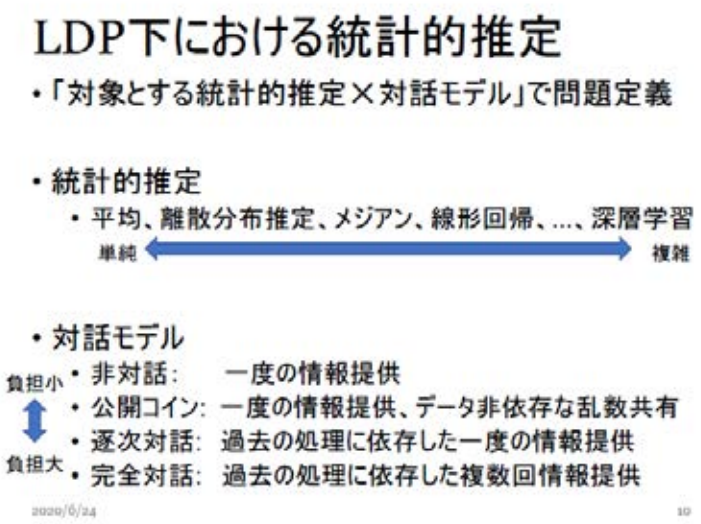


図 2.8-3 LDP 下における統計的推定

完全対話モデルとは、ユーザーと何度も何度もやり取りをするようなタイプのモデルである。非対話モデルは、ユーザーが好きな時間に自分の情報を送れば良いので負担が小さい。完全対話モデルはとても負担が大きい。対話無しモデルは非常に単純であるが、逐次的対話モデルはインタラクションが発生するため複雑である。

局所差分プライバシー下において、統計的推定の目標を考える際、小さいサンプルで小さい誤差の推定をしたい。また、ユーザーの負担の小さい対話モデル、つまり完全対話よりは逐次対話、逐次対話よりは公開コインモデルといったように、なるべく負担の小さい対話モデルで推定をしたい。効率の良い推定とは、minimax rate の意味で最適であって、ある基準がある。負担が小さい対話モデルとは、先ほど申し上げた通り序列がある。推定効率が minimax rate の意味で最適かどうかという点と、対話モデルの負担という点がトレードオフの関係にあるのかどうかは、まだよくわかっていない。例えば、非対話モデルのときの最適性と、公開コインモデルの間の最適性の間にはギャップがあるかが分かっていない。もしギャップがあるならば、非対話モデルにも公開コインモデルにも意味がある。もしギャップが無いならば、公開コインモデルを使う意味はなくなる。その辺りがまだ良く分かっていない。

推定対象の複雑さ、例えば平均値を計算するというタスクと、ディープラーニングを計

算するというタスクは、複雑さが大きく違う。しかし、推定対象の複雑さが、推定効率と対話モデルの負担のギャップにどういう影響を与えているのかも良く分かっていない。これらを解明していくことが研究としては目標となってくる。

幾つか事例をご紹介する。Duchi氏が、あるクラスの統計的推定を局所差分プライバシー下で行う場合の、効率の限界を明らかにした。これを明らかにすることで、このクラスに属するようないろいろな問題で、このぐらいのレートが最適だということが分かるようになってきた。その結果、いろいろな推定問題のLDP下での最適性の保証が可能となった。

Acharya氏は、ユーザーデータが一様かどうかを検定する際、対話なしモデルと公開コインモデルの間にギャップがあることを解明した。良い精度で推定したい場合は、ユーザーに少し負担を強いて、公開コインモデルでやったほうが良いことが分かってきた。

Joseph氏は、逐次的対話モデルと完全対話モデルの間にもギャップがあり、良い精度で推定したい場合は完全対話モデルを使う意味があることを示した。

この流れで自身の研究も紹介したい。平均値の推定というのは非常に簡単である。データがノイズで汚れていても、足し合わせると中心極限定理で誤差は小さくなる。最小値の推定では、極限にある部分も推定しなければならないためかなり難しい推定問題となる。1次元の連続データで、逐次的対話下で、局所差分プライバシーを保証しつつ、ユーザーのデータの最小値を推定するという問題について、ある種の分布に対する仮定を置いて、最適な推定誤差を達成できるような方法論の研究を行っている。もう一つは、現在投稿中の研究成果だが、個人の長期時系列データを収集する際の局所差分プライバシーについてである。時系列データは長いデータであるため、プライバシー保護を優先すると役に立たないデータになってしまう。われわれは関数データ解析という手法を使っている。関数データ解析とは、長い時系列データを、1つの関数に変換し、関数情報に対して局所差分プライバシーを利かせて、データ提供をするという方法を提案した。LDP-FDRという方法である。局所差分プライバシー下で関数データ解析を実現する初めての手法であり、推定誤差のレートが最適であるということを証明できているため、良い研究だと思う。

今後の方向性であるが、非対話モデルにおける単純な統計的推定の理論は、かなり出来上がっており、現在いろいろなサービス、システムに実装されている。そこで、より複雑な対話型モデルへ向かう。例えばRAPPORやiOSで実装されているものは、サーバとのインタラクションが一切なく、ユーザーが好きなタイミングでデータを吸い取られている状態であるが、ヒストグラムではなく難しい推定をやろうとする場合は、より複雑な対話型モデルの下でデータを提供しなければならない。この辺りについて、先ほど挙げた幾つか個別の条件においては、この対話型モデルだと、こういうレート、といったことが分かっているが、全体的な体系はまだ分かっていない。こういうものが分かってくると、恐らくグーグルやアップルなどがサービスに実装してくると思う。あるいは、IoTのようにデータを取り続けてサーバ―に提供するようなものというのは、こういった技術との関係が重要になると思う。将来的にはさらに対話型モデルを最適に選択し、プライバシーを守りつつ、ユーザーの負担が少なくAIを学習するようなことが最終的な目標になっていくのかと思う。

差分プライバシーとシステムソフトウェアについてまとめる。社会の要請は、プライバシーの保護であり、かつユーザーの負担は大きくしないことである。一方で、例えば今回のコンタクトトレースのように、幾らプライバシーが大事だと言っても、情報は収集・分析し、社会に役立てなければならない、といったタイプの要請もある。理論研究の側としては、こういう要請になるべく応えるように、最適な推定メカニズム、最適な対話モデルというものが、一体どのように特徴づけられるのかを調べる作業が必要になる。例えばこういう規模の推定をするのであれば、このぐらいの時間内で終わらせねばならないため、プラットフォームとしてはこういうスペックでこういう計算ができないといけない、といった要求を出すことが必要である。データ収集や解析プラットフォームの側からすると、現状、こういうネットワークがあり、こういうデバイスがあり、こういうプロセスがあり、このぐらいのスピードでデータを収集でき、このぐらいのスピードで計算ができるので、その範囲内で要請されているプライバシーを守りつつ推定できる方法を考えてほしいと理論側に要求することになる。それらがインタラクションしながら、理論側は限られたスペックで推定をする。プラットフォームの側としては、理論側に調整を求めていくような感じになるのかと思う。

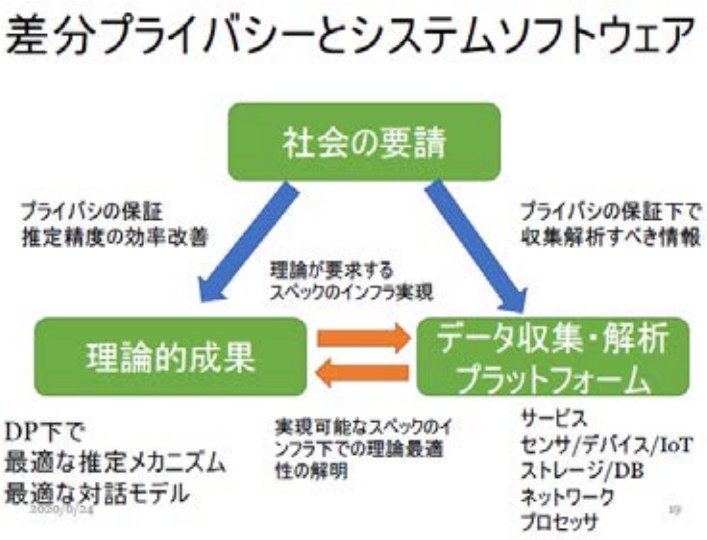


図 2.8-4 差分プライバシーとシステムソフトウエア

【質疑応答】　（Q：質問、C：コメント、A：回答）

Q：秘密計算で、準同型ではないお話がもしあればお願いしたい。

A：秘密計算は、大きく分けて 3 種類の計算がある。Garbled circuit は、10 万 gates を 1 秒ぐらいで計算できる。ただこれは 17 年の結果なので古く、その後進展があっただろうと思う。Secret sharing は、暗号技術を使わず、データを乱数に分割して秘密を守ったまま計算するタイプの技術である。暗号を使わないため非常に早く、18 年の結果であるが 10 億個の AND gates が 1 秒で評価できる。準同型暗号は、それより何桁も遅く、われわれの研究グループで 10 万乗算が 1 秒ぐらいというのが、喜連川先生の CREST でやった結果である。

Q：河原林先生からのチャット質問を読み上げる。「heavy hitter の話は、STOC なので、理論的話だと思うが、実装などされて既に応用されているんでしょうか」というご質問について。

A：本日ご紹介した Bassily さんの結果が、実装自体は別に難しくないと思うが、アップ

ルやグーグルのところで動いているのはSTOCやコックスなどで提案されたものとは少し違う。ただ、中で働いておられる方を多少存じ上げているが、STOCなどで論文を発表されているような方なので、近いものがあると思う。この分野は理論と実装の間のギャップというのは、ある意味小さい。理論の成果がダイレクトに跳ね返ってくるような、応用できるようなタイプの領域である。ただ、理論とギャップがあるとすると、例えば1億台のiPhoneから定期的に情報を送らせるといった場合に、例えばユーザーの帯域を使わないといけないことについて、理論としては関係がないがサービスをやる上で特有の事情はあるようだ。理論でやったことを、思いどおりに、そのまま直ちに実装できるかというと、実際はそうでもない。お答えになっているか分からないが、そういう状況である。

Q：heavy hitterの話はよく聞くのだが、佐久間さんのおっしゃった通りで、やはりSTOC、フォックスと実装の本当に近い分野だなと思っている。となると、日本は相当頑張らないと追いつかないなという、絶望的な気分になりながら聞いていた。日本から貢献できるような、入っていけそうな分野が何かあるか。

A：隙間として入り込める。アメリカでは、大学からFACEBOOKに行き、1-2年ほどサービス開発をやり、また大学に戻っている人を見かける。

Q：STOCやフォックスの人々はそこまでやっていない。グーグルにコミットしている人は結構いるが、100％そっちに使うという人はあまりいない。

A：あるいはクロスアポイントメントで、グーグル、メリーランド大学とかいう人はいる。

Q：あるいはグーグルに1-2年間、サバティカルで張りつくパターンは結構耳にする。

A：あれはどうしているのかよく分からない。すごいお金を積まれて、1-2年だけ行って大学に戻るのか。それとも面白そうだから行くのか。そこまで突っ込んで聞けないため知らないが、とにかくアカデミアとインダストリーの間で、気軽に移動し、成果が出たらアカデミアに戻って、といったことをやっているため、そもそも体制が違うのかなと思う。もちろん実力不足もあると思う。

Q：サバティカルで1-2年くらいで、グーグルに行き。居心地がよかったらそのまま残る。あるいは帰ってくるといったことをその時に決めると言う人が結構いるようで、理論は恐らくそんな感じなのか。やはり理論のトップの人が、いわゆるSTOC、フォックスばりばりの人々がこの分野にコミットしているのを見ると、危機感を感じる。恐らく、日本ではできないと思い、やはり人材を育成しないとまずい。

Q：その辺の話は、パネルのテーマでお願いしたい。最後に、縫田先生からご質問が来ている。「秘密計算については、プロトコルだけではなくて、Intel SGXのようなハードウエア的な保護機能を併用して効率を向上させる研究があるが、DP/LDPはどうか」といったご質問である。

A：Secret sharingなどでは、ランダムシードがセキュリティにダイレクトに利いてくるため、Intel SGXを使うモチベーションはあると思うが、DPの場合はあまりセキュリティ上に大きな影響がなく、あまり研究されていないと理解している。

2.9 S5Kernel のための基盤ソフトウェア～システムソフトウェアの視点から～

河野 健二（慶應義塾大学）

システムソフトウエア、OS から見た S5Kernel の研究課題に関して、最先端研究の動向と今後の方向性を説明する。まず、Society 5.0 の基本的アーキテクチャーを図 2.9-1 に示す。今後の ICT システム・サービスはこのような構成で構築されていくものと考える。クラウドのような巨大なコンピューティング環境を、単一の組織体で占有して使用することは不可能なため、1 つのクラウドをマルチ･テナント環境で利用することになる。さらに、単一のクラウドベンダにロックインすることは現実的でないため、マルチ･クラウド環境で多種多様なサービスが構築されることになると考えている。特に、Society 5.0 での利活用を想定した場合、さまざまなサービスを提供する事業者がお互いに連携することで、さらに新しいサービスをつくっていく形態となるため、マルチ・テナント環境かつマルチ・クラウド環境が将来システムの基本になることは自明と考える。

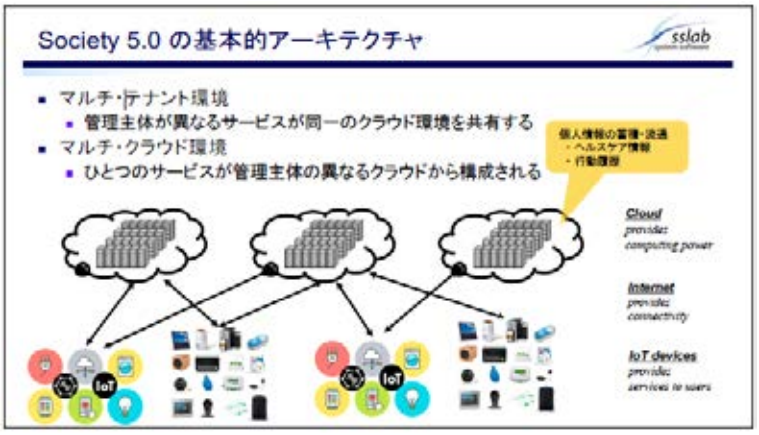


図 2.9-1 Society5.0 の基本的アーキテクチャ

このような将来システムの顕著な特徴の一つは、システムの管理主体が階層化している点である。ハードウエア、OS、アプリケーションの全てを単一事業者が保守・運用・管理している場合には、責任の所在が明確である一方で、現在のクラウド環境は管理主体が多階層化しているため責任の所在を明確化できない事が課題となる。ハードウエア上で、VMM に代表されるハイパーバイザーが動作し、ハイパーバイザー上でゲスト OS が稼働し、ゲスト OS 上で、クラウドプラットフォームやライブラリが動作し、これを利用してアプリケーションが実行される環境となっている。このため、Society 5.0 の利用者から見えるのはアプリケーション提供者のソフトウエア群であるが、そのソフトウエアが動作するクラウド環境はクラウド事業者が管理･運用するため、管理主体が多層化した環境下で一貫した安全性を提供する仕組みが重要となる。さらに、ハイパーバイザー、OS、クラウドプラットフォーム自体が、巨大なソフトウエアになっており、クラウド事業者はこの巨大

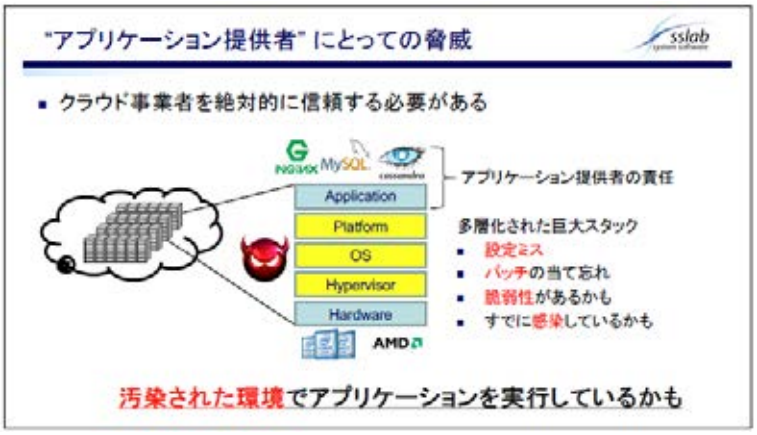


図 2.9-2 アプリケーション提供者にとっての脅威

なソフトウェア・スタックを保守・管理・運用することが困難な状況になっている。この結果として、ソフトウエアの設定ミス、パッチの当て忘れ、脆弱性リスクへの不十分な対応が存在しており、すでに汚染された環境で、アプリケーションを実行している危険性があり得るということになる（図 2.9-2）。

これは、アプリケーション提供者から見ると、非常に迷惑な状態になっている。つまり、自分たちのアプリケーションの脆弱性やセキュリティホールが自己責任である一方で、自分たちが利用しているクラウド事業者の問題（セキュリティ脆弱性）により、例えばユーザーから集めたプライバシーデータが漏えい、あるいは改ざんされてしまうという危険性が存在している事が問題となっている。では、どの程度クラウド事業者が安全性を保証しているかというと、通常の対策は行うものの、何らかの安全性を保証している訳ではないのが実態である。某クラウド事業者の Customer Agreement を見ても、最近のセキュリティ報告書をみても、内部犯行事例まで含めて“信頼できるもの”には程遠い状況であることが分かる。

このような背景のもと、アプリケーション提供者に求められていることは、基盤ソフトウエアが汚染していたとしても、データの秘匿性や完全性を保証できるようにアプリケーションを開発すること、になる。しかし、基盤ソフトウエアは、最高の権限（Privilege）を保有し、やりたいことは何でもできるように開発してきたため、基盤ソフトウエアが乗っ取られた場合には、全てのコードデータを閲覧し、改ざんできることになるため、既存技術では、汚染された基盤ソフトウエア上に、データの秘匿性や完全性を保証可能なアプリケーションを開発することは不可能であり、深刻な課題となっている（図 2.9-3）。

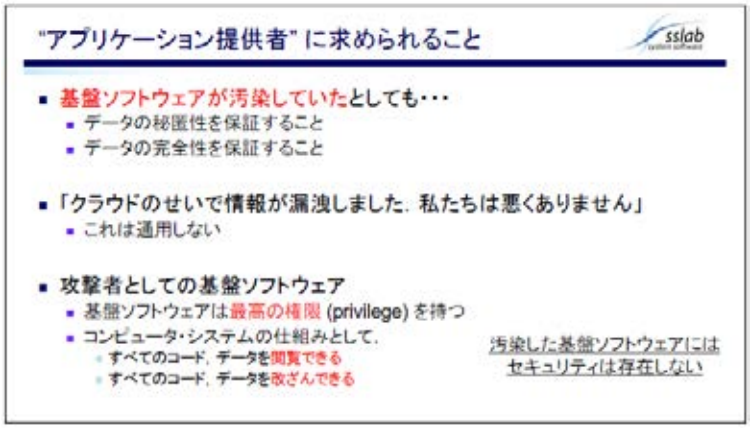


図 2.9-3 アプリケーション提供者に求められること

本課題の解決に向けた研究開発アプローチの１つとして注目されている、Trusted Execution Environment（TEE）について、最新動向と今後の発展研究の方向性を説明する（図 2.9-4）。TEE というと非常に狭い意味で使う場合も多いが、まず、広義での Trusted Execution Environment の考え方を紹介する。これは、悪意のある特権ソフトウエア、OS、

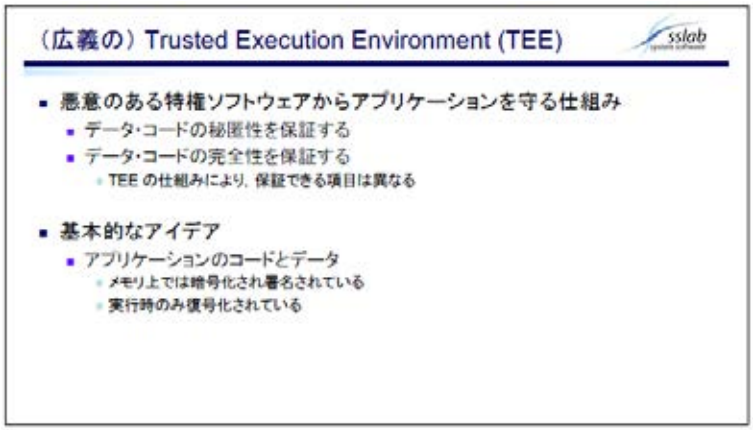


図 2.9-4 広義の Trusted Execution Environment

ハイパーバイザーからアプリケーションを守るような仕組みである、という定義である。つまり、どの部分をどんなレベルまで処理するかは複数のアプローチがあるが、OS、ハイパーバイザーが乗っ取られていても、アプリケーションが持っているデータとかコードの秘匿性とか完全性を保証するというもので、近年注目されている技術である。TEE では、アプリケーションのコードとデータは、メモリ上では全部暗号化されて完全性も保証したいときは署名されており、実行するときだけ CPU の中で復号化され実行され、特権ソフトウエアからは暗号化されたデータ・コードしか見えないような状況を保証する、事が基本アイデアとなっている。

TEE の考え方は、10 年以上前から指摘されているものであり、オープン・パブリックなクラウド環境では、特権ソフトウエアに悪意があった（つまり感染した）としても、アプリケーションのデータを守るという根源的なものである（図 2.9-5）。最初は、ソフトウェア技術のみで実現しようと研究開発がスタートしたが、近年ではハードウエアサポートが必須として研究開発が進められている。ソフトウェア技術のみでの実現を試みた研究事例では、2008 年の Overshadow や 2011 年の Cloudvisor が著名である。アイデアは、アプリケーションに対して Shim と呼ばれる一種のライブラリを導入し、アプリケーションが直接 OS と相互動作しないようにした上で、さらにハイパーバイザーを改良して、アプリケーションが利用するメモリの暗号化・復号化を行うことで、OS からアプリケーションのメモリを隠蔽するものである。しかしソフトウェアのみでの実現方式のため、実装も非常に複雑で、オーバーヘッドも軽くはなかった点が欠点であった。

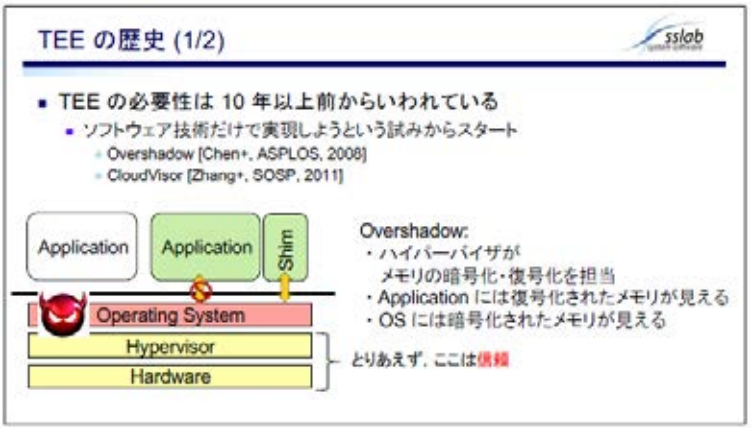


図 2.9-5 TEE の歴史

このため、2015 年あたりから、ハードウエアサポートされた狭義での TEE として、Intel SGX、AMD SEV/SME、ARM TrustZone、RISC-V Keystone が開発された。Intel SGX は図 2.9-6 に示すとおり、アプリケーションを暗号化したまま実行するための CPU 拡張であり、OS 等の特権ソフトウェアからも暗号化されたメモリしか参照できない仕組みになっている。具体的には、エンク

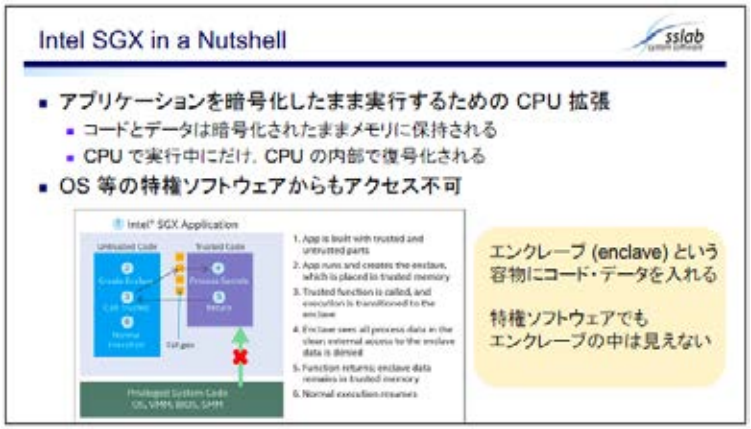


図 2.9-6 狭義の TEE（Intel SGX）

レーブと呼ばれる領域にコードやデータを格納しておけば、特権ソフトウェアであってもこれを参照できないという仕組みになっている。

現在、TEE を利活用した研究の方向性は図 2.9-7 が示すとおり、（1）TEE を素直に使って、これまでは開発できなかった高安全な新しいアプリケーションを構築する、と（2）既存のアプリケーションを安全に実行可能とする TEE を用いたプラットフォーム（安全な実行環境）を構築する、（3）TEE の制限を緩和するソフトウェア技術の探求、（4）TEE 向けアプリケーションの開発支援、がある。今回は、（1）（2）の研究事例を紹介する。特に（2）は、アプリケーションを TEE 環境に丸ごと入れれば課題はないような印象があるかもしれないが、技術的にはそれほど単純には解決できない課題があり、これを説明する。

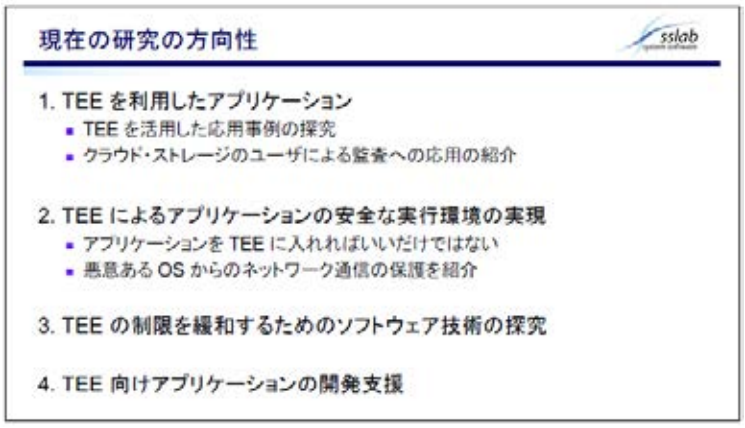


図 2.9-7 TEE 研究の現在の方向性

（1）の研究事例として、クラウド・ストレージのユーザーによる監査アプリケーションを紹介する。近年発生し問題となっている Gmail でのアカウントリセットや Dropbox でのユーザファイル消失を監査するためのアプリケーションである。このような事態が発生することも問題であるが、利用者視点では、事故発生がクラウド事業者によってリアルタイムに報告・公開されていないという事も大きな問題であり、利用者自身がデータの保全性を守らないといけないという課題意識に基づく研究事例である。事例として紹介する LibSEAL（図 2.9-8）は、例えば Dropbox に保存したデータが、クラウド事業者側のバグ・障害で改ざんされていないかを監査する Auditting ツールである。利用者が操作したファイル追加の情報と、Dropbox システム側からの応答通知をつきあわせてデータ改ざんの発生を検知する仕組みを Intel SGX の TEE を利用して実現している。監査機能の実現には、監査データ自身の改ざん防止が鍵となるが、TEE を活用することで効率的に実現できる事例である。

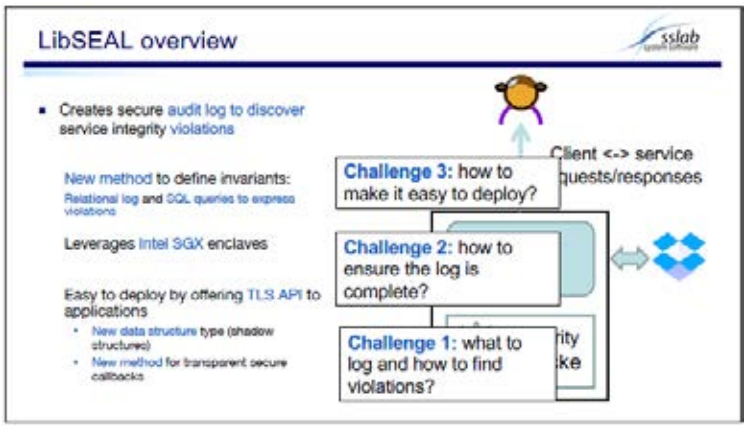


図 2.9-8 LibSEAL の概要

次に（2）の研究事例を紹介する。TEE によるアプリケーションのセキュア実行の実現方式はおおきく 3 種類に分類されると考えている。1 つ目は図 2.9-9 に示すとおり、アプリケーションを素直に TEE のエンクレーブの中で実行させる方式である。本方式は TEE

に入れ込むコード量を小さくできるため、脆弱性発生リスクを低く抑え、TCB を小さくできる利点がある。一方で本方式は、TEE 環境と OS・ライブラリが libc レベルでつながるため、API の幅が広くサイバー攻撃の対象となるリスクを高めてしまう事が欠点となる。このため、使用するライブラリを注意深く選択するような新しいアプリケーション実装には適用可能であるが、既存アプリケーションに適用した場合には、大きな攻撃ベクタが残ることになる。

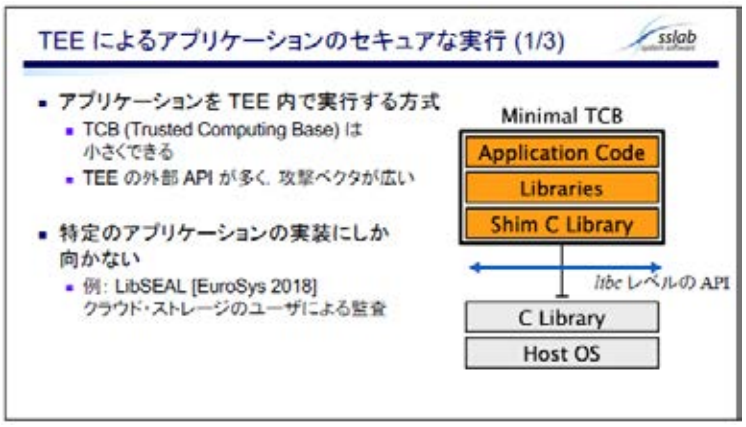


図 2.9-9 TEE によるアプリケーションのセキュア実行①

TEE によるアプリケーションのセキュア実行の 2 つ目の実現方式は、図 2.9-10 に示すとおり、OS、ライブラリ、アプリケーション全てを TEE 上で実行させる方式であり、2014 年の Haven が著名である。本方式の場合、TEE 内の OS において全てのシステムコードを実装できないため、ホスト OS と API により相互動作を行う必要があるが、Haven の場合は 22 個と API を非常に限定的にできることが特徴である。このため外部から API 経由でサイバー攻撃を行うことはやりにくくなる、一方で、TCB が非常に大きくなってしまうのがある種の欠点である。

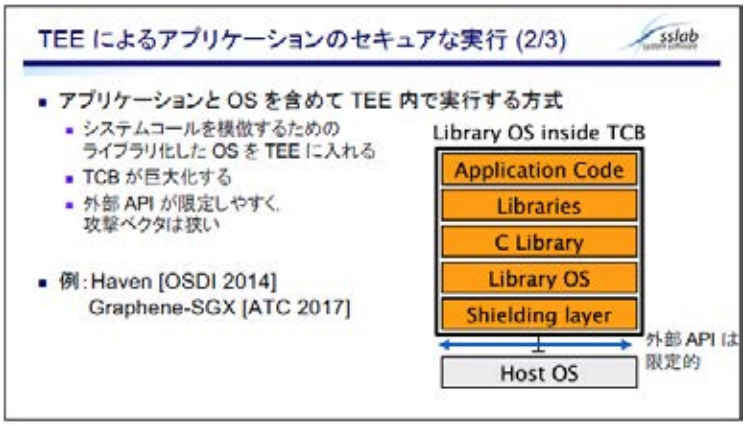


図 2.9-10 TEE によるアプリケーションのセキュア実行②

TEE によるアプリケーションのセキュア実行の３つ目（最後）の実現方式は、図 2.9-11 に示すとおり、１番目と２番目の中間的なソリューションで、システムコールレベルの API を用いて、TEE とホスト OS を相互動作させる方式である。つまり１番目の C ライブラリと２番目の OS と比較して、TCB サイズや外部 API 数が中間的

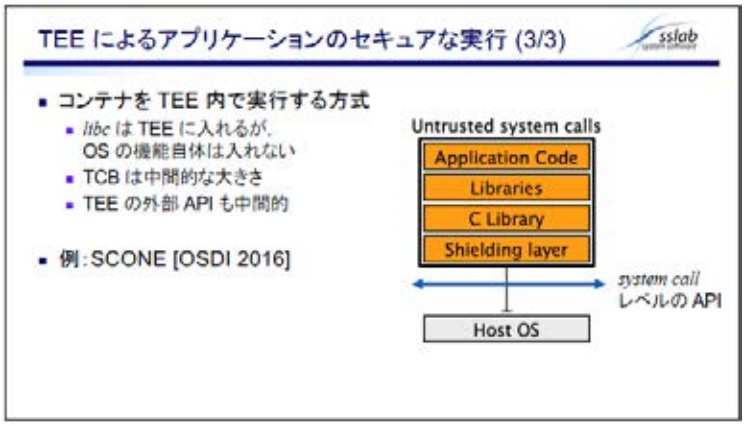


図 2.9-11 TEE によるアプリケーションのセキュア実行③

なレイヤーを設定するものである。2016 年の SCONE が著名であり、現在製品化を進めていると聞いている。

ここで、自身の研究成果の一部を紹介する。図 2.9-12 に示すように、TEE を用いてネットワーク・スタックを安全に実行させる研究を進めている。例えば、TCP/IP は OS で実装されているため、OS が乗っ取られると、通信メッセージを傍受すること、改ざんすること、虚偽メッセージを外部に発信すること等が可能となり、深刻な問題である。これを解決するために、ネットワーク・スタックを OS が隔離し、TEE 内部で実行することで、通信の秘匿性・完全性を保証することを提案している。本方式は、OS が汚染しても、TEE 内のアプリケーションとネットワークは正常に動作することを保証することが可能である。さらには、NIC（Network Interface Card）を隔離することで、乗っ取られた OS が NIC に悪意のあるアクセスから保護可能である。

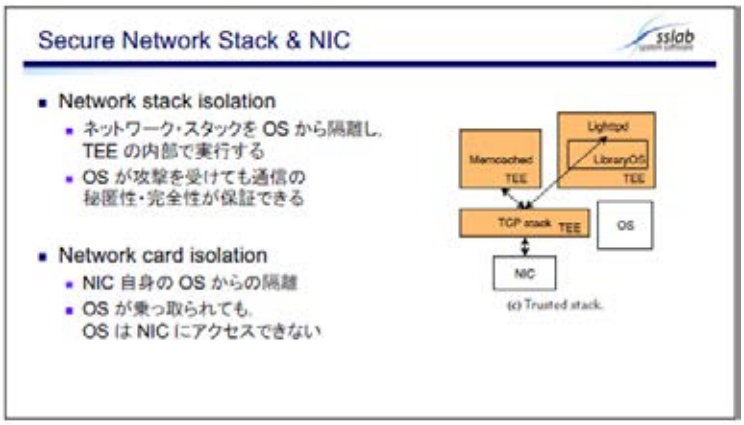


図 2.9-12 TEE によるネットワーク・スタック

この NIC を TEE を用いて OS から隔離する手法の、システム構成例を図 2.9-13 に示す。本構成では、少し賢い NIC として、TEE で保護された暗号化ルーチンが動作する環境を前提としている。このような環境下で、NIC とホストのネットワーク・スタックの間で、OS を介在することなく鍵交換を行い、NIC とネットワーク・スタック間で暗号化通信を行う手法を提案し、Mellanox BlueField を用いた試作を行っている。当然、OS がアプリケーションを装って NIC と通信できないような仕組みの実装も行っている。

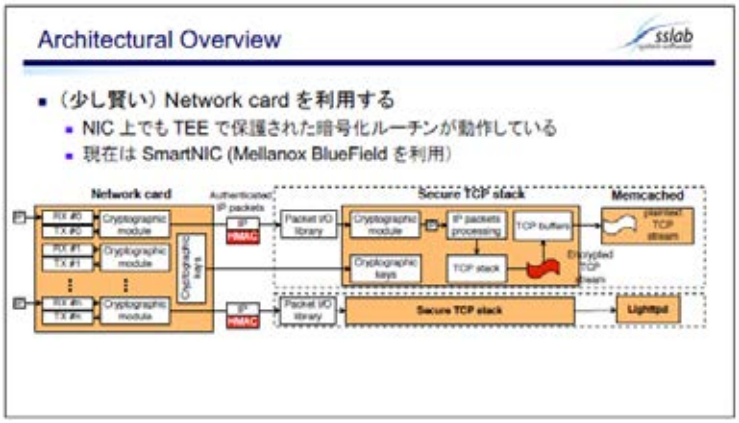


図 2.9-13 TEE を用いた SmartNIC の構成例

次に、今後の方向性について考えを述べる。これまでの特権ソフトウェア（OS）は、図 2.9-14 の左側に示すとおり中央集権的なモデルであり、全てのハードウエア・ソフトェアが制御下におかれていた。このため、OS が乗っ取られてしまうと、データの秘匿性・完全性を保証できなくなり、アルゴリズムや処理内容も全て盗難可能となる課題を抱えていた。一方で、近年になって、ヘテロジニアス・コンピューティングという概念で、システム自身がシングル CPU から、CPU を搭載する周辺デバイス群になるマルチ CPU へとコン

ピューティング環境が変革している。先の中央集権的モデルの根本課題の解決策として、このようなヘテロ環境のパワーと安全・安心のために利活用する、つまり単一のハードウエアを管理する特権ソフトウェアが複数存在し、これらの相互監視的なモデルにより総合的に安全性を保証する新しいシステム・ソフトウェアモデルを提言する。ヘテロなスマートデバイス毎に、うまく特権ソフトウエアの権限を分割委譲することで、どこか1か所乗っ取られても、攻撃被害を局所化し、アプリケーションレベルのサービス提供を継続させることを可能にできると考えている。

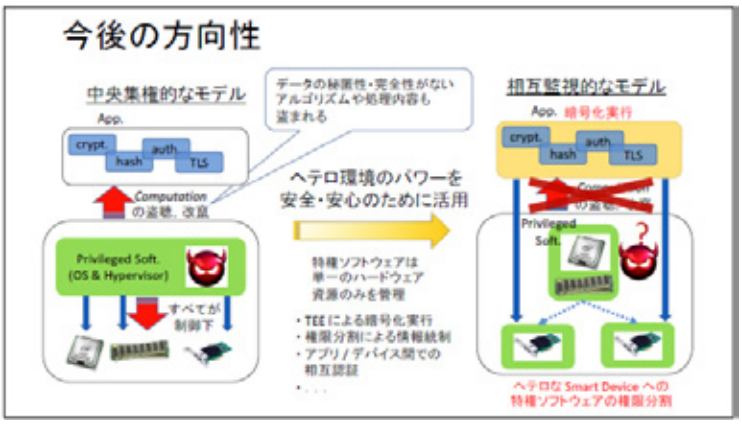


図 2.9-14 今後のシステムソフト研究の方向性

これまでは、ハードウエアは信頼できるという仮定をおき、システムソフトウエア技術の動向と今後の方向を説明してきた。しかし、ハードウエアさえも信頼できない時代であることを、念のため、図 2.9-15 により説明しておく。ハードウエアの脆弱性として著名なのは、Meltdown、Spectre、Rawhammer である。Rawhammer では、DRAM の性質を悪用し、メモリ内容の強制書換え、ページテーブルの不正書換えが可能になっており、さらには、Intel SGX でさえ信頼できない（ex. 暗号化キーを盗聴可能）との指摘も存在する。

図 2.9-15 ハードウエアが信頼できない事例

個人的には、ハードウエアにおいても、RISC-V のようなオープンソース化とコミュニティによるセキュリティ機能強化が重要になっていると考えている。

システムソフトウエア領域は、図 2.9-16 に示すとおり、管理主体が階層化され、脅威モデルが大きく転換している時代

図 2.9-16 まとめ

であることを説明してきた。これまで、OS は絶対信頼できる前提で構築されたものが、そうではなくなり、さらには、ハードウエアも信頼できない状況である。このような、非常に難しい状況下で、システム全体の安全性をどのように保証するかが、システムソフトウエアの“今”直面している問題である、という結論で説明を終了する。

【質疑応答】　（Q：質問、C：コメント、A：回答）

Q: ご発表のなかにあった少し賢い NIC（Network Interface Card）とは、DPDK が搭載された NIC と理解して良いか？

A：ご指摘のとおり、最先端の NIC は高価ですが、CPU が搭載され簡易な OS が動作し、DPDK で通信処理を行っている。このため、NIC で暗号化等のインテリジェントな処理が可能という意味で、少し賢いという説明をした。

Q: ハードウエアが信頼できない時代に、どのようにして SECURE な計算を実現するのか？また、業界はどんな方向に向かおうとしているのか？

A：正直には、私がその答えを聞きたくて、今回話をした。OS 業界では、原則として、ハードウエアを信頼するという前提で研究開発を進めてきたが、その前提が成り立たない時代になった。SECURE な計算を実現する方向性は、複数ベンダのハードウエアから構成される計算機環境に同じ計算処理を行わせてビザンチン合意のような判定処理を行うアプローチと、Risk-V に代表されるオープンソースハードウエア環境下における SECURE 計算を担保するアプローチがあると考えてる。特に後者は、ソースコードが公開されているためセキュリティ対処も行い易いという利点がある。現状では、SECURE 計算を担保するために、OS 技術者がインテル SGX や GPU のリバースエンジニアリングを行っており、ある種無駄な労力をかけていることが大きな問題と思っている。

2.10　全体討議

コーディネーター 河原林 健一（国立情報学研究所）

河原林：全体討議に移る前に、今日は日本を代表する研究者の方々にもご参集いただいているので、参加者の皆さま全員に一言ずつお願いしたい。

橋爪：今日は、セキュリティやプライバシーを中心としたさまざまな分野の先生方に、最新動向についての情報や御意見を教えていただいた。今後、パネルディスカッション等々で議論を深めていくと思うので、頂いたご意見を参考に、日本として強みを持って育てていくべき領域を検討させていただきたい。さらに、議論の中では、今後の方向性も示していければよいと考えている。

江村：個別の領域のお話は、皆さん、大変分かりやすくお話しいただいて、よかったなと思う。全体を通して、セキュリティとその裏側の脆弱性みたいな話があり、ある程度理解が深まったと思う。皆さんがいろんな図を描かれている中で、そのセキュアさを保っていくみたいなことを、全体としてどうやってやっていくのというあたりが１つの論点かなと思う。さらに、途中でも議論があったが、その中で日本の強みはどこに本当にあるのというあたり、もし強みがなかった場合に、グローバルな中で、どういう立ち位置でやっていくのかみたいなところの全体像が、描ければいいかなという感想を持った。

喜連川：江村さんが言われたことが全てみたいなもので、ここまではそれぞれの人が言いっ放しになっている。それで、物理的な開催の場合は、この後、もうちょっと詳細な議論を、それぞれ興味のある人たちが寄り集まりながら、話が進められるのが、このサイバー空間なので、ちょっとそこがこなれない。上から目線で、がさっと全体をまとめるというよりは、個別の研究者同士が、自分よりもちょっと広めのところの、今日お話になられたり、興味を持っている方同士とで、どういうふうに自分の世界を広げるかというのをお進めいただくというようなことが、次にマストになるのではないかなと思う。

裏事情としては、ここまでよく来たなというようなところでもあるので、多くの先生方ご参加いただいていると思う。けど、日本の中で比較的同じような領域の人が、ここまで発表して、議論できるような場にこぎ着けたというのは、大した一歩じゃないかなと、私どもとしては、個人的には感動しているような、そんな感じである。

岡部：今日、初めてこのアクティビティに参加された方は、ようやく全体像が少し見えてきたぐらいじゃないかと思うが、江村会長からもあったように、それぞれ、要素技術で、皆さん、すごい最先端の研究をされている。これを今、S5Kernel というブランドで、１つのアクティビティとして見せるようにしたい。どのようにすると、さまざまな要素技術が１つのアクティビティとして、それぞれシナジーを持って外に、特にアプリケーションをつくりたい方々に対して、分かりやすく、使いやすいものにできるか。そういう観点でこの後議論していただければと思う。

木村：私は研究の現場から離れて 10 年以上になるが、今日の話を聞いていて、セキュリティというのがいろんな分野に入って、こういうコンピューターのシステムの中のいろんな分野に入ってきていることを再認識した。しかし、少し個別の研究が深掘

りされている感じがするので、もう少しそこを幾つかのバーティカルを仮定して、何か考えてみると、有効的な研究テーマが出てくるのかなという感想を持った。そして、江村さんがおっしゃっていたように、日本の強みとか、世界の中での立ち位置をどうするかを意識しながら考えてみると良いと思った。

さらに、今日の話は技術的にはいわゆるコンピュータシステムに目が行っている感じがしていて、Society 5.0 というコンテキストを考えるならば、IoT とか、今までわれわれがあまり目にしてこなかったところが意外とネックになりそうな気がする。IoT のような小さい装置は、長く使われるし、更新はそんなに頻繁じゃないし、素人さんが使うとかいうのがあって、そういう背景も鑑みて、センターシステムを考えていくべきではと思った。

井上：１個の大きな旗を、みんなで考えなきゃいけないのだろうと思うが、いきなり全体をまとめようとすると大変なので、例えば僕らのアーキテクチャーとかだと、河野さんのところの OS と連携しやすいみたいな話があるので、ちょっとずつ、幾つかのまとまり、小さなものを少しずつ大きくして、最終的に大きな１個の旗にするというやり方もあるかなという気もしている。

佐々木：日本の強みということで何か出すんだったら、各研究領域が近いということが挙げられるかと思っていて、もう少し各分野が幾つかくっついて何か１つのプロジェクトをやるみたいな提案を増やすのは、日本の強みという点ではやってみてもいいのかと思う。

河野：佐々木先生のやつは、終わりのほうだけすこし聞けて、そういうことをやっているのなら、うまくくっつけそうだなと、同じ感想である。いきなりくっつけようとしても、多分、発散するだけなので、お互い分かるところからやるというのがいいと思う。

松浦：私は最初にセキュリティの概観させていただいて、一番重要なのは、われわれが狭い意味のコミュニティとしては離れていた者同士で、本格的に連携する課題が大事だというふうに申し上げた。言い換えると、あるソリューションを出すために連携しましょうというよりは、まず課題をつくるところから連携するというのが大事なんじゃないかなと感じた。

盛合：セキュリティやプライバシーに関して、いろんな幅広い分野の方がお話されていて、非常に感銘を受けたのと、いろんな接点を感じたところであった。日本として、やはり強くしていくために、大学で強みを持っている分野、企業でその強みを生かせる分野のおのおのを生かしながら、連携チームの組み方を考えると良いのかなというのを感じた次第である。

佐久間：研究内容自体が強い、弱いという話と、日本の大学とか研究所と企業との関係っていうのが、どういうふうに連携しているのかという、体制的な部分の２つが絡み合った問題のような気がしていて、それ自体を一研究者がどうかということはちょっと難しいと思う。しかし、体制自体をすぐに変えるのは難しいのかもしれないんですけど、やり方を議論するだけでもいろいろ変えていけることができるんじゃないのかなと、思ったところである。

田浦：全体をどうがっちゃんこするかが課題だという話があると思うが、自分の感想は、関連はすごくあって、何かデータを収集するプラットフォームを想定すると、差分プライバシーとか、秘密計算とか、データを集めるときのプライバシーを侵害しないための基礎的な技術があって、プラットフォームとして、河野先生が紹介したクラウドでのTEE上で、クラウドで動いているソフトウエアがちゃんと検証され、確認できるという絵が、それなりにいいのかなとは思っている。IoTに関しては、あちこちに散らばったデバイスなので、検証や暗号の重要性がより増してくるので、全体のまとまり感はあると思う。

小林：分野横断という必要性は私も強く認識していて、そのきっかけとして、私が最後に申し上げたAIでソフトの質とかが変わるので、それにどう対処すべきかということを、いろんな分野の人が集まって、何か一緒にやるといいのかなと思っている。日本の強みはというのは、本当に大丈夫なのかなというのがあって、基盤ソフトを開発している企業がない、グーグルとかマイクロソフトとかに圧倒されてしまっている、大学の研究も理論寄りでこじんまりとしたプロトタイプをつくっておしまいになりがち等が何とかならないかとずっと思っている。

高橋：私は、暗号とかプライバシー保護というコンテキストの話を来週する予定である。それで、今日の議論の中にもあったが、それらは、理論的にすごく基本的な仕組み、ソフトウエアに、幾つかはぐさっと歴史的に入っているものだと思う。やはり研究をして、実証していくというふうにやると、どうしても研究が小粒になってしまうと企業の立場でも感じている。その悩みとかを、うまく、ちょっと難しいが、来週、表現するように準備したいなと思う。

福田：今日のお話で、インテグレーションが課題というふうにおっしゃっていたが、それぞれ研究されている方たちの役割を考えたときに、インテグレーションって誰の役割なのだろうというのを思って、アカデミアで研究されている方は、そもそもそういうことをやられて、興味の対象になっていないんじゃないか。じゃ、誰がインテグレーションするのだろう。企業の立場でインテグレーションするときに、あまり国の境界とかっていうのは、そんなに意識することはなくて、これ、誰が本当にインテグレーションを進めていくのかなというのを考えるのは難しいかなと思った。一方で、イノベーションというのは、分野の境界領域で起こるということは強く感じていて、さまざまな分野の人たちが集まって、いろいろ話をして、新しいことが生まれるのではないかというのが、非常に面白い試みだなと思った。最後に1つ、何か日本の強みを出そうと思ったときに、これからゲームチェンジが起こるようなエリアで、新しいプラットフォーマを日本から出していくということが重要なのだろうなと思った。

吉川：日本が強いところはどこかということを考えることは、ある意味、逆を考えるのも大事かなと思う。つまり、ここはもう手を出さないほうがいいというのはどこなのかというのを、みんなで共有するとかいうことも、あるいは実は同じ裏側から見ると同じことかなと思う。例えば個人的には、あるものが使えないとしたら、それを前提としてチームをどうつくるべきかということを議論すべきと思っている。

小野：やっぱり課題とか問題をみんなで発見して、そこに1つ旗を立てて、各分野が交流

することが非常に重要だなというのを、強く感じた。さまざまな分野のオーバービューとか最先端研究というのを聞かせていただいた。けれども、やっぱり知らないこともあるし、これらの先生方と定期的に何か交流が持てて、議論ができていくと、今後ますます自分の研究も面白くなると思うし、日本の分野としても強くなっていくのではないかなと思った。

塩谷：最後の河野先生の発表は聞いたが､河野先生の発表や田浦先生の資料などだけでも、一緒にできることは幾らでもあるなというふうには感じている。

品川：政治的な話でいうと、今日はスパコンの「富岳」の話が、世界１位になったという話があったと思う。非常に勇気づけられて、日本もやればできるのではないかというところがあって、ハードは強い。そのハードを展開していけば、世界制覇ができるのではないかということをツイッターでつぶやいたら、ソフトウエアをつくる組織は完全に壊滅状態とかいうリプライが来たりしている。しかし、実は日本でもそういうシステムソフトウエアが好きな人とか、物すごいできる人が結構散らばっているのだろうなという感じは受けていて、うまくそれを集めてくる仕組みであるとか、旗を立てて、みんなで集まっていくような仕組みを、うまく（ハードやセキュリティとの連携を含めて）ソフトウエアのほうでもつくってあげれば、結構日本も力はまだまだ残っているのではないかなというのは、個人的には思っている。

光来：まずは近い分野であると、どう連携していいかというのが見えやすいなと思ったので、そこから始めればいいかなというのは感じた。けれども、ただ、それはある意味､結果がある程度の成功で収まるのかなという気もしていて､やっぱり大きな成功に結びつけるためには、完全に異分野の方たちと連携していくというのが面白いかなというふうには感じた。ただ、もちろん失敗するリスクも非常に高いので、１％成功すればいいかなと、そういう感じになってしまうと思う。けれども、狙っていくんだとすると、そこになっていくのかなと感じている。

須崎：NEDO プロジェクトで、RISC-V のプログラムをつくっている。これをつくっていて、サイドチャネル攻撃等がかなり深刻な一方で、対策技術も進んでいるので、ぜひともいろいろ協力して開発できないかと考えている。

穐山：私は競争力という意味では、企業の方たちと協力というか、総合的にその企業のほうが得意な部分があると思うので、うまく活用・連携できると良いと感じた。

花岡：大きいシステム全体のセキュリティを可視化して、今どこがどのように守られているのかということが分かるようになってくると、じゃ、この技術はここ、この技術はここ、これを組み合わせたらできるねということができるような気がする。私自身どうやったらいいか分からないが、そんなようなものを、このコミュニティでつくっていけたらなというふうに、勝手ながら思っている。

縫田：テクニカルな点で気になることとしては、IoT とか、そういうところとの関連というのを狙っていこうとしたときに、今日のお話って、どちらかというと、ハードウエアとかというのも含めて、割と電子的な世界の話が多かったと思うが、電子的なハードウエアとかネットワークよりもさらに外側にある物理的な世界に詳しい人にも手伝っていただけると、もっと有益なんじゃないかなという感想を個人的に抱い

た。

矢内：セキュリティ関連では、検証や解析結果を IEEE 等のさまざまなプラットフォームで共有を行っている。このようなセキュリティ情報共有プラットフォームを学術界からつくっていく力が、現状は相当弱いと思っており、それを乗り越えていけるような連携があるといいかなと思っている。また、セキュリティは目に見えない分、かなり分かりにくい世界だなと思っていて、多分、プラットフォーマとしての力を押し上げるのだったら、分かりやすさ、まさに可視化とかが１個そうだと思うが、というのを考えていくのが、１個、取っかかりになるのかなとは思っている。

川本：ソフトウエアの検証技術はセキュリティ全般に適用できると思う。今日少なめだったテーマとしては、サイドチャネル攻撃の問題がある。例えば、暗号化したまま計算する場合であっても、通信量やタイミングや電力から、セキュリティやプライバシーにかかわる情報が漏洩してしまうことがある。また、キャッシュメモリへのアクセスのタイミングから暗号の秘密鍵が漏洩する事例についての研究もある。そういったサイドチャネル情報漏洩の問題に対しても、情報流解析（information flow analysis）といったソフトウエアの検証技術が応用できる。また、暗号アルゴリズムの正しさや安全性が数学的に証明されていたとしても、それを実装したソフトウエアがアルゴリズムを正しく実装できているとは限らず､安全であるとも限らない。暗号の実装の正しさや安全性の確認に対して、ソフトウエアのテスト技術や検証技術を応用していくことは重要なテーマである。

曹　：システムソフトウエアとして、これからの可能性のある方向性の一つは、さまざまなプライバシー、セキュリティ技術の融合によって、社会問題を解決することになれるかなと思う。例を挙げると、グーグルとアップルでつくっている局所的な差分プライバシは、プライバシーのほうが強いが､ユーティリティのほうに問題がある。つまり局所的な差分プライバシーに達するために、かなりのノイズを加える必要がある。最近それを解決するために、最近、シャッフリンという TEE を利用している差分プライバシー･モデルがかなり注目を集めており、差分プライバシー、暗号、PCTE 等の技術融合が進んでいる。

五十嵐：これだけの人たちを１つのところに集めて、何かいろいろ話をしていくと、自然発生的にコラボレーションというのはできるかなとは思っている。それを１つの旗で緩く結合しているのを、１つのアクティビティだとして見せられれば、何かいいのかなとは思った。また、プロトタイプからさらにみんなを驚かせるようなツールとかをつくるときにマンパワーが足りないなというのをひしひしと感じている。そこら辺で、例えば企業の方と連携できれば良いと思う。

住井：こういう機会を設けていただいて議論するだけでも、情報系という意味での大きな分野の活性化につながるのではないかと感じた。今日は、プログラムとかシステムとか、ハードウエアの話が主だったと思うんですけれども、プログラミング言語理論、広い意味で、計算システムの記述の理論なので、もっと広く、IoT 等の話もあったが、個人的には例えば社会のプロセスとか、社会のシステムの記述や検証等もプログラミング言語理論の範疇に入り得ると思っているので、そういう意味でも、こういう広い分野の先生方の集まりがあると、日本の情報学の研究の活性化にも大変

つながるのではないかなと思った。

姜 ：人工知能の応用分野として、GPS データ、軌跡データを使って人の動きの予測の研究をしている。軌跡データといえば、やはりプライバシーの問題が浮かんでくると思う。なので、本日、佐久間先生と盛合先生の差分プライバシーの研究に大変興味を持っている。やはり複数の先生がおっしゃったとおり、分野の横断が必ず必要だと思う。ので、差分プライバシーとか、既存のプライバシーの技術を使って、その軌跡データのプライバシー問題を解決したいと思う。また、日本の強みといえば、やはり基礎研究だと思う。ビッグデータと AI は、論文数から見ると、中国とアメリカにビハインドしている。しかし、その一方で、ユビキタスコンピューティングの研究分野では日本の存在感が大きい。なので、今回の Society 5.0、システムソフトウエア、基礎研究としてのプロジェクトの中でも、IoT とかビッグデータの話もどんどん入れていただきたいと思う。

井上（拓）：コンピュータサイエンスって､今まで水平分業して､それぞれのレイヤの中で､みんなが個々に頑張ることで、全体として進化してきたという部分があるというふうに感じている。さらにレイヤを融合していくということで、個々のレイヤの中でつくった面白い種をつないでいく話と、各レイヤーの中で新しいのをつくって連携する話という話をどうやってバランスをとっていくのかが面白いと考えていた。

竹房：田浦先生にご紹介いただいた SINETStream を開発している。そのライブラリというのは、IoT のアプリケーションを書く人たちに対して、専門分野の人があまりそのコンピュータサイエンスの専門知識がなくても簡単に書けるようなライブラリをつくるということを目的として、ソフトウエア開発をしている。ライブラリの中では、例えば暗号化するとか、IoT アプリを構築する上で必要な機能とかもいろいろ入れていくということを考えていて、さらにはバックエンドのクラウドなり、ネットワークなりのシステム側に関しても、自由にいろいろな技術が組み込めるような設計をして、今つくっているところである。いろいろ協力できそうなところがありそうだなというふうに感じた。引き続きよろしくお願いしたい。

宮地：今日は貴重なお話をたくさん聞けて良かったなと思っている。特に日本の国が強くなっていくというような研究をしていくときには、分野連携って非常に重要と思っている。私はセキュリティが専門だが、コンピュータセキュリティ、ネットワークセキュリティ、アルゴリズム的なセキュリティが、ジョイントしてやっていけると、もう少し大きな勝負ができるのかなというのをすごく感じた。

河原林：これで多分全員の方にお話しいただいたと思う。このような研究者が集まれる場というのが必要なのは、これは皆さんの一致した意見かと思う。一方で、特に今回のワークショップの１つの目的として、来年度の戦略目標にこの分野全部含めたものを立てたいというのがある。ですから、やっぱり何か旗を立てなきゃいけない。しかし、どういう旗にするかという議論を、今から 30 分でやろうというのは無理なので、それは実は次回の 29 日に、皆さんにぜひご意見をいただきたいというところである。あと 30 分程度、フリーディスカッションということで、ご意見がある人は、どんどん言っていただければと思う。

喜連川：それはいつまでにやらなきゃいけないかというのをまず。橋爪参事官から、大体

こんなタイムスケールというところから始めたほうがいいんじゃないかと思うが。

河原林：自分が聞いたのは、7月20日ぐらいまでに、何かポンチ絵を出さなきゃいけないという話は聞いたがそれは正しいのか？

橋爪：戦略的創造研究推進事業については、去年の実例では、各課からの提案を夏頃に集め、その後、省内で議論を深めていくという形であった。今年は、少しそれよりも少し前倒しというような動きもあるので、それを考えていくと、先ほど河原林先生がおっしゃったようなスケジュール感となるのではと思う。ただ、戦略目標の広がりという面では、技術の特定、絞り込みをし過ぎるというのも適切ではない部分もあるかもしれないので、先ほど言われたように、このSociety 5.0を支えるソフトウエアシステムについて、幾つかの方向性、軸というのを出していければ、非常にありがたい。

また、これはJSTともコラボして深めていくものであり、JSTの視点からも何か必要なところがあれば補足していただきたい。

茂木：参事官がおっしゃった話は、われわれも同様に考えていて、今日のお話をずっと聞きながら、Society 5.0のシステムソフトウエアとしては、ちょっとレイヤが下というか、例えばプライバシーセキュリティの話だけで、Society 5.0がちゃんとできるのだというストーリー立てにはしにくいなと考えている。一方で、ヘテロな環境をうまく扱うとか、IoTを追加しようとすると、今度は広がり過ぎてしまう。そのバランスをどうするかが、まさにこの後の議論になってくると思っている。さらには、これをやると、Society 5.0の一部でも役に立つよねというふうなシナリオまで育てていかないとと考えている。

橋爪：具体的にどういう技術があって、どういうところが課題なのか、ここをまず掘り下げた上でやっていく必要があると思っているので、今日いただいたご意見は、われわれとしては非常にありがたいと思っている。

木村：こういう基礎的な、基盤の技術はやらなきゃいけなくて、ただ、それをやるためには、何か傘が要るのかなと。九大の井上先生は旗とおっしゃったけども、何かそういうのを言いながら、基本的なところはきちんと押さえていって、分野連携をやっていくと、シナジー効果が出てきて、いいのが、すごく日本らしいものが出てくるのではないかなというふうに思っている。もう1点、Society 5.0というと、日本がまだ強いと言われているのはデバイスとかIoT関連やリアルワールドに近いところであり、これをサイバーというか、コンピュータシステムにうまく取り込んでいくことを考慮すると、面白い出口が見えてくるのではないかなと感じている。

河原林： 多分、CRDSとしては、ポンチ絵1枚というものは、ある段階で恐らく一月以内に出さなきゃいけないので、何かしらの旗を立てる、または傘をつくることが必要かと思う。その点に関して、次回はもうちょっと、今日はNTTの高橋様からのお話があった。けれども、企業サイドも含めた視点も入れていかなきゃいけないこともあるかと思う。この点についてコメントがあれば、江村会長よろしくお願いする。

江村：Society 5.0の話がJSTからあったが、ちょっと安易な議論をされている感じがあって、上のレイヤの話をすればいいという話ではなく、コロナの話にもあったように、

もう１回、次の社会をちゃんとつくっていかなきゃいけないという構造になっていると思っているんですね。次の社会をつくっていく上での基盤という意味での情報システムをちゃんとつくろうというのが、私たちがこの１年間、議論してきたことなので、これをちゃんと理解して、JST がやっていただかないと、と感じた。

茂木：今回のシステムソフトウエアを研究開発して、何かつくり上げることが、どう上のレイヤにつながっていくのだというところをきっちり埋めていかないと、文科省の中でのそういう議論に対抗できないと思っている。これは、そのシナリオをつくる JST であるとか、文科省の方と協力してやるという仕事だと感じている。

江村：それが、旗を立てるということだと思う。ちょっと話はそれるが、経団連の中西会長なんかも、Society 5.0 って、この COVID を見たときに、もう１回見直さなきゃいけないということをおっしゃっている。だから、そういうことに対して、研究開発ではこういうところをちゃんとやっていくので、次の中でリードしていけるポジションの可能性があるのだよねということを､ちゃんと打ち出していく｡そのストーリが、多分文科省の方や、内閣府とか、そういう人たちに響くような形に創っていかなきゃいけないので、これはぜひ一緒にやらせていただきたいし、そこら辺のところは、何で今回セキュリティとかに重点を置いた議論をしているかというのも、実はつながっていることなので、次回も含めてだが、そのストーリをいろいろ議論させていただくのが重要と思っている。

木村：江村会長のお話に少しコメントをつけ加えさせてください。われわれは、実は別のストリームで、コロナを経験して、じゃ、Society 5.0 をどう見直すという宿題をもらっていて、それと同時並行的に、今回のこのワークショップもやらせてもらっている。ご指摘のように、このセキュリティというのと Society 5.0 が、今回のコロナを経験してきて、われわれが次どうするべきかというところは、かなりつながっているような感覚を持っている。つまり、コロナというか、ユーザー視点から見たときの Society 5.0、市民から見たときの Society 5.0 というのと、個々でやっているような情報システムを含めた基盤技術としての Society 5.0 というのをうまく融合するというか、つなげるのを考えてみたい。ご指摘は、そのとおりであり、そこをうまくつなげていきたいというふうに思っている。

喜連川：今、経団連の話が出ている。けれど、一番最新の 5.0 を中西さんがどう捉えているかというのは、この間、先々週、NII のオープンハウスで、中西さんに来ていただいて話をしていて、40 分程度のレコーディングも公開しているので、それを聞いていただければ、一番、今どういう問題意識を持っているかというのがお分かりいただけると思う。やはり、アッパー系をやろうとしても、結局、砂上の楼閣になってしまう。そこの問題意識が相当大きいような気がする。確かに日本が弱いところがあるのは、それは言い出したら、みんないろんなところで弱いが、弱いから、じゃ、上だけやるのですかというと、結局そのプラットフォームのところが、全部取られたままの体力で、日本がこれから持ちこたえられるのだろうか、が根源的な課題だと思っているのが、こういう会をオーガナイズした一番のモチベーションである。

どなたかが、「富岳」が１等賞だから、それと同じようにやればいいとおっしゃって、まさにそれと同じことを、私は科学官のときに、たまたま「京」のスパコンを

つくるときだったので、「京」のソフトウエアというものを、大学も一緒につくるべきだということを強く主張していて、そういう流れに持っていけないかなと思った。しかし、結果的にはハードウエアも丸ごと富士通がつくる、ソフトウエアも丸ごと富士通がつくる、何一つ介入することができない。今度、「富岳」にするときには、まずオープンにしてくださいと。今回は少なくとも全部オープンにしましょうというので、スパコンの推進委員会は、かなり大きな方向転換をした。

今回、こういう分野の方々が集まって、何か一緒にものをつくるような、そういう場をつくれると、日本が強くなるのではないかなという気持ちになっている。なぜかと言うと、やっぱりITのシステムそのものが、1つのキーアイデアだけでつくれるような時代というのは、もはや終わってしまって、完璧にコンパウンドなワールドになったような気がする。フェイスブックが何か唯一のキーアイデアを持って、それをつくっているかというと、全然そんなことはなくて、そこそこ普通の技術を上手に寄せ集めるということがコンピタスパワーになっていると。だとすると、このハードウエアというものを何かつくりながら、みんなでいろんなソフトウエアをごちごち載っけていくような場というものをつくることが、次の日本のこういうシステムソフトウエアの研究にとって、非常に重要な役割を果たしていくんじゃないのかなという気がしている。

情報処理学会、40研究会ぐらい全部ヒアリングしたわけだが、そのときに、例えば介護のアプリとか、いろんなアプリが並んでいて、それはそれ単体で予算が取ってこられる。一番予算が取れないのが、こういう基盤層で、基盤層のプロポーザルを出すと、一体これは何に使うのですかと。アメリカに勝てるのかとか言われた瞬間に、ちまちまっとした話にしかならない。だからずっとここが冷や飯を食っていたと。そこを何とか根源的に打破したいと思っている。そのときの切り札は、今日もどなたかがおっしゃっていたと思うが、セオリーと、インプリメンテーションが極めて、タイトバインドするという技術領域が、僕は面白いところじゃないかなと思っている。さらには、学と学、あるいは一定程度の産を招きながらやるという、ある種のやり方を模索するということも非常に重要なんじゃないかなという気がしていて、その辺をぜひJSTさんのほうでお酌み取りいただきながら、そして文科省と情報参事官の方々と連携しながら、こういう分野をぜひ育てていっていただきたいなと。応用から攻めるというのは、もう腐るほどやっている。ですから、そっち側は重要じゃないとは言わない。重要じゃないとは言わないんですけれども、今、日本が弱いところをどうやるかということが、非常に大きなこのプロジェクトのターゲットだということ、非常に重要なポイントなので、共有できればなと思う。

木村：「京」の開発責任者をやった手前、少し言い訳させていただくと、私が「京」を始めたときに、そのアーキテクチャー、命令セットを設計するときに、いろんな大学の先生方、国研の専門家の方々にいろいろヒアリングをさせていただいた。けど、なかなか具体的なこういうものがあるといいというのは、幾つか、あまりもらえなかったということがあって、それで、時間的な制約もあって、じゃ、やりますかねということになってしまった。ああいう結果になってしまったことには、ご指摘のとおりで、すごく反省している面がある。今回、こういうふうに学と官が共創的に

シナジーを出していけるという環境ができそうだということで、私はすごく、今回のこの活動に対しては期待もしているし、JST としては責任も感じていると思っている。

茂木：全然異論はなくて、私自身が３年くらい前から、喜連川先生がシステムソフトウエアは絶滅危惧種だというふうな話を言われたときから、セキュリティであるとか、ソフトウエア基盤であるとか、実は文科省の戦略目標の提案をし続けてきた。ずっと敗れてきた経緯があって、そのときに何を言われたかというのが、さっき言われたようなことだったので、そこをしっかり埋めないと、今日の議論があったような、こういったコミュニティができつつある。あるいは旗に集まろうとしている、そういうものをうまく束ねるためには、僕らがそこをきっちり埋めてやらないといけないと強く思っている。

喜連川：今回は、参事官が随分、力添えというか、６期のドラフトの中に、随分こういうことも書き込んでいただいている。ですから、今までの戦い方とはちょっと違う持込み方が、私はできるのではないかなというふうに希望している。これは、アメリカだと当たり前だが、何かトピックスペシフィックな人が集まる場所というのをオーガナイズするメカニズムが日本の中にないのではと思う。情報爆発したとき、実は 3.11 のときに、情報爆発の最終シンポジウムがあったが、皆さんから言われたのは、研究代表者が集まって、腹を割って議論できる場というのは極めて貴重だと。それぞれの先生に渡しているお金は大したことない。１人数百万円しか渡していない。けれど、すごくシナジー感がある。あれがなくなってしまって、どうするのかというようなことを随分言われたが、JSPS があの枠組みを捨ててしまったので、あれ以上できなかった。

今度、ぜひやっていただきたいのは、やっぱり一定程度の予算的なフレームワークを、JST さんがおつくりいただいて、何か基盤みたいのは、また NII もお手伝い、ファンディングエージェンシーができにくいことは、大学共同機関としての NII がやるというような、何か新しい枠をつくりながら、こういう研究のコミュニティをどうやって、ちゃんと絶滅危惧種じゃないようにしていくかというのを本当に真剣に考えていきたいと思う。年がら年中、GAFA に負ける、負ける、としょうもないことをお経のように唱えていたって、しゃあない。私は、今回のこのチームづくりというものがうまくいくと、太い幹となるようなコミュニティがそこに出来てきてほしいなと強く思っている。今日、皆さんが来てよかったなとおっしゃっていただけたというのは、本当に心の底からうれしいなという気がしているので、ぜひ JST さんも強いご支援をいただけるとありがたいと思う。よろしくお願いする。

河原林：ほぼ時間だが、最後に喜連川先生に締めていただいたという感じになった。次回は産業側から、ちょっとアッパーサイドから基盤への期待ということを込めて、講演いただきたいと思う。また、今日参加いただいた。ほとんどの先生方、皆さま方から、先ほど喜連川先生からあったように、エンドースメントをいただいた。こういうような場をつくるということの重要性、それから緩くてもいいから繋がって良いものを、分野を乗り越えるようなものを創りたいというようなご意見をたくさん頂いたので、次回も参加いただいて、皆さまのほうから、こういうものをやりたい

とかという提案というものがあったら、次回のパネルミーティングは時間もあるので、提案いただければと思っている。

最後になったが、最後までお付き合いいただいた。文科省関係者、特に橋爪参事官、皆さま方、どうもありがとうございました。JST 関係者もどうもありがとうございました。

2.11 ラップアップ

（1）文部科学省　橋爪参事官

河原林先生、喜連川先生、江村先生、それからご参加の皆さま方、長時間、お忙しい中で時間を割いて将来のために話し合っていただき、本当に感謝申し上げる。こうした場でさまざまな課題や研究内容を共有できることについても、大変ありがたい。ともすれば、最近、選択と集中、トップダウンというような言葉がはやりがちであるが、やはり科学というものは、研究者一人ずつの力で基礎からしっかりと育てていくことが大事であり、その上で、どこに注力をしていくかというところを初めて考えられるのではないかと思う。

先ほど絶滅危惧種という話もあったが、そんなことないと強く期待している。一方、例えばアプリケーションをつくっていくときのマンパワーが足りないとか、多くの方がソフトウエアに関する知見を持っておられるという声もいただいている。JST とそういった課題も共有をして、すぐ答えが出るかどうかは分からないが、どういう新しいやり方でやっていけるかぜひ皆さま方の声をお聞きしながら、取り組んでまいりたいと思う。今後ともお付き合いいただけるようお願いする。

（2）NII　喜連川所長

皆さんは橋爪参事官がおっしゃった言葉を当たり前のように感じておられるかもしれないが、こんなミーティングはこの何十年間の間ではまずなかった。これだけ研究者と文部科学省がタイトな関係になって、こういうところまで、長い間参加いただいて、皆さんをエンカレッジするような態度は過去にはなかった。なぜ今これができているのかというのをよく考えると、去年の夏に、情報処理学会と NII が協力しながら、40 個の研究会を全部スキャンしたからである。それだけわれわれが頑張って、きちんと研究の新しい方向を見定めようという姿をずっと橋爪参事官はご覧になっていた。皆勤だった。そういうアティチュードをわれわれが見せている中で、文部科学省さんとしても、いやいや、これは捨てておけないという形になっている。

旗を立てるということは、JST か誰かがやってくれると思っているかもしれないが、そうではないということを考えてもらいたい。「情報爆発」というのはつまらない名前で、単なる四文字熟語だと思われているかもしれないが、あれはその前の特定研究の中で、次の特定研究として何をするか考えなさいというのが、安西先生からの宿題であった。それに対して、いろいろなことを考え、最後に先生方の前で発表させていただく。そういう過程で、これで行こうということを有川先生たちが決めた。

今はもう皆さんの時代になっているので、旗をデザインするのは自分だと思っていただき、それを JST に提案してください。JST はそれに対して檄を飛ばしていただきたい。

（3）NII　河原林副所長

次回は６月 29 日になる。少し長めの１時間半程度のパネルの時間を取っている。今日は皆さまの思いをお話しいただいたが、次回はどういうものをやりたいかというアイデア

を出していただくような場にしたいと考えている。

今日は、長い間、お付き合いいただき、ありがとうございました。

3. ワークショップ（第２回）研究成果の応用に関する議論

3.1 挨拶

（1） CRDS　木村上席フェロー

前回は主にアカデミアのほうから話をしていただいたが、今日は産業界の方々に講演をいただき、議論をしたいと思う。その後、前回と同様に、皆さんでディスカッションをするので、どうか活発な、建設的な議論をお願いする。

（2） NII　河原林副所長

前回はアカデミアの参加者の方からエンドースメントをいただいた。今回はエンドースメントだけではなく、そのときの議論にあったように、どうやって傘を作るか、どうやって旗を立てるかというところでのディスカッションをしたいと考えている。

3.2 研究成果の活用（1）次世代 IT 基盤ソフトウェアへの期待

谷崎 勝教（三井住友フィナンシャルグループ、日本総合研究所）

三井住友フィナンシャルグループのグループ CDIO と、日本総合研究所の社長を兼務している。「CDIO」はチーフデジタルイノベーションオフィサーの略称で、SMBC グループ全体のデジタルイノベーションを統括する立場である。

今日は、銀行を支えてきたシステムを概観し、SMBC グループの新中期経営計画を紹介し、次に SMBC グループから見た課題認識に基づき、デジタルで攻めへ転じるための「IT モダナイゼーション」、デジタライゼーションを支える「サイバーセキュリティ」、データ活用とプライバシー保護を両立する「データ・テクノロジー」の３点について話をして、最後に皆さま方に期待することを述べる。

金融とテクノロジーが融合してきたという話をよく聞くが、50 年前から既に融合していたと個人的には思っている。IT が導入される前の銀行は、支店ごとにお客さまの情報を全て紙で管理していた。自分の口座というものは、開設した支店でしか入出金ができず、支店には手作業で作業する職員が大勢いた。

金融とテクノロジーが結びついたのは前回の東京オリンピックの頃と言われている。1964 年の東京オリンピックで、IBM の大型計算機でつくられた情報システムを使い、競技結果の速報がリアルタイムで生成され、提供された。1965 年には、当時の三井銀行が日本で初めてオンラインシステムを IBM から導入した。

そもそも、お金というのは基本的に数字で表現できるので、データとして計算機で扱いやすい。さらに、いろいろな支店で同じような処理をするので、そういった意味でも IT の導入によって得られる効果は非常に大きかったと考えられる。コンピューターに記録されるようになると、１つの支店にしかなかった情報が、全ての支店で共有できるようになっていった。

現在の銀行は図 3.2-1 のような状況になっている。カウンターはあるけれども、後ろで仕事をしている職員はいないという形である。

行員が事務をするスペースというものを、ほとんどなくしてしまっているので、支店に来ていただいたお客さまがゆったりと過ごせるスペースを確保できている。

図 3.2-1 現在の銀行

支店の改革を積極的に進めてきた背景は、インターネットバンキングの導入や、スマホ・モバイルの普及で IT によりお客さまはどんどん便利になり、支店に行かなくてもあらゆることができるようになってきたということにある。

IT が導入されると、お客さまは支店に行かなくて済むし、銀行は支店の事務を圧倒的に

減らすことができるということで、双方にメリットがある。お客さまは便利に使え、銀行はコストが減らせるということで、双方にとっての効率化のためにITが使われてきたということである。

10年間にわたってオンラインの仕組みをどんどん拡大し続けた結果、銀行システムは国内外にわたってさまざまな業務を支える巨大なシステムになっている。基本的な定例メンテナンスを除くと、24時間365日、大規模なプログラムが安定的に稼働し続けている、といった意味で、銀行はもはやIT装置産業になっていると言ってよいと思う。

どのぐらいの規模感で銀行のシステムが動いているのか、ということを図3.2-2に示している。SMBCの国内にあるシステムの規模だけで言うと、プログラムの行数にどれほどの意味があるのか今ではよく分からないが、大体3億行というふうに言われている。スペースシャトルシステムの規模が0.5億行と言われていることを考えると、スペースシャトルの大体6倍ぐらいのプログラムが毎日障害を起こすことなく動いているということになる。

図3.2-2 銀行システムの規模

SMBCグループは、この4月から新しい中期経営計画を始めた。その中でどんなことを考えながら業務をやっていこうとしているのかを示したのが図3.2-3である。

図3.2-3 新中期経営計画の基本方針

「最高の信頼を通じて、お客さま・社会とともに発展するグローバルソリューションプロバイダー」という新しいビジョンを掲げている。そのうえで、具体的に3つの方向性として、情報産業化、プラットフォーマー、ソリューションプロバイダーを示している。情報産業化というのはデータを経営資源として十分に生かして、それをいかにお客さまサービスにつなげていくかということである。プラットフォーマーというのは金融機能をベースに新たな付加価値をどのように提供していくかということである。3つ目として、お客さまの課題を解決し、信頼される存在になりたいということで、ソリューションプロバイダーを挙げている。

いずれの方向性も、デジタルとかITといったテクノロジーを使わない限りはなし得ないと位置づけている。そこで、図3.2-4には、われわれが考えるデジタル戦略を定義している。デジタライゼーションといっても、各社各様にいろいろなことを定義していて、きちんとした定義はないというふうに思うので、自分たちはこういうふうに考えているとい

うことを示している。

大きく「攻めのデジタル化」と「守りのデジタル化」に分けている。通常私が講演するときは、聴衆の方たちに耳触りのいい「デジタルイノベーション」や「デジタルトランスフォーメーション」といったキーワードで、最近こんな新しいことをやっている、お客さまとの関係で新しい価値創出の仕事をしている、新しいビジネスはこんなことをやっている、そういう話をよくする。それは、上の「攻めのデジタル化」の領域だが、本日は下段の「守りのデジタル化」について説明する。

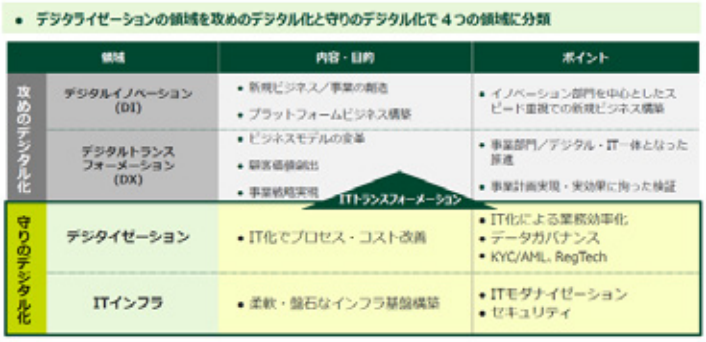


図 3.2-4 デジタライゼーションの定義

今まで手でやったり、FAX でやったりしていた手作業を全てデジタルの情報に変えていくということを、われわれのところでは「デジタイゼーション」という言い方をしている。そして、重要な大きな柱が「IT インフラ」である。IT システムを最新の技術でモダナイゼーションするということがない限りは、守りのデジタルから攻めのデジタルへの本質的な動きはなかなかできないのではないかと思っている。

われわれの金融機関で攻めのデジタルと守りのデジタルのシステム経費をどのぐらいの比率で使っているかというと、攻めのところで使うのは３割以下で２割強ぐらいと理解していただければよいかと思う。７割以上は守りのところで、大きな IT インフラを支えていくといったところにお金が回っている。これから、攻めのデジタルに経営資源、お金の配分先を変えるためには、いかにしてこの守りのデジタル化のコストを削減するか、ということが大きなテーマだと思っている。

次に SMBC グループにおける課題認識から見た問題提起をお話しさせていただく。

「守りのデジタル化」ということに関しては、図 3.2-5 に示した３つの手段が大切なものだと思っている。システム数がどんどん増大化していく、肥大化しながら複雑化していくという課題に対して、いかに IT モダナイゼーションを起こしていくか。それから、われわれ金融機関に対しては、ものすごい勢いでサイバー攻撃が来ているので、これに対してどのようにセキュリティを高めていくか。それから、今後のデータ活用とプライバシー保護の両立に向けたデータ・テクノロジーをどうするのか、これら３つが大きな課題としてあるのではないかと思っている。

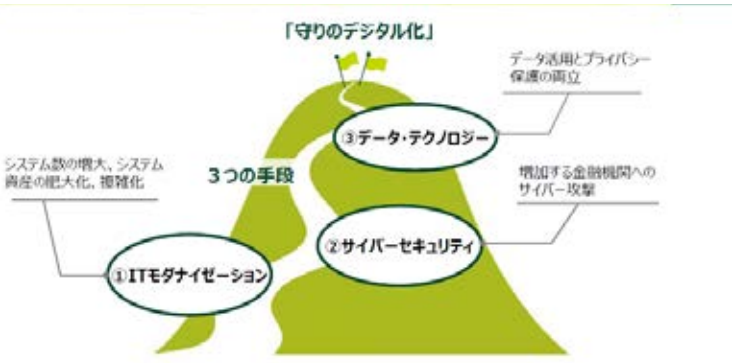


図 3.2-5「守りのデジタル化」の概観

最初の IT モダナイゼーションについては、複雑化・ブラックボックス化した既存のシ

ステムの温存が続くと 2025 年には壁にぶち当たるという話がある（図 3.2-6）。毎年 12 兆円の経済損失に至るというレポートが発表されている。金融業界にとっては、2025 年だからという問題ではなく、IT システムの刷新というのは常に大きな課題だと思っている。

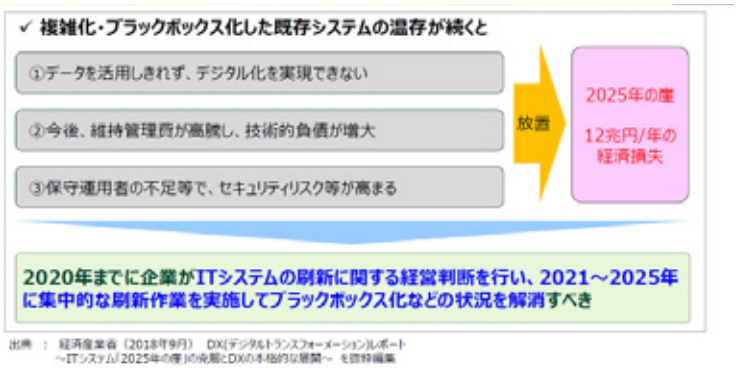


図 3.2-6 2025 年の崖の克服

レガシー資産の実態として２つあると思っている。図 3.2-7 の下側に書いてある「IT インフラ」は最新のテクノロジーで刷新しやすい資産だが、上の「お客さまサービスを実現する業務アプリケーション」のインフラの部分は、コアコンピタンスとして継承していかなければならない資産だと考えている。

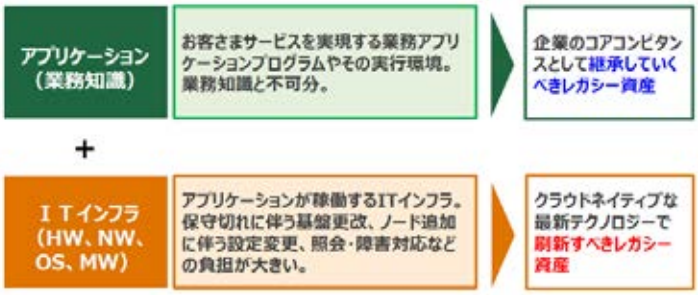


図 3.2-7 ２つのレガシー

そして、コアコンピタンスとして継承しなければならない業務アプリケーションの継承においては、大きな課題が存在していると思う。４次オンラインということで、われわれは４回ほど業務アプリケーションの更改を続けているが、時間がたってシステム同士が有機的に結びついていくにつれて、技術負債というものが蓄積されていく。複雑化し、スパゲッティ化することでシステムの保守性が低下し、多くの担当者は日々の保守作業に忙殺されて、新たな技術にチャレンジすることがなかなかできない状況に追い込まれている。われわれ金融機関のように IT システムを古くから導入している企業ほど同じ課題、悩みを抱えていると思う。そこで、図 3.2-8 に示した２０年に１回行われる「式年遷宮」のように、１０年に１回ぐらい金融機関の基幹系システムを抜本的につくり替えるということで、技術的負債の解消や、スキル移転を次世代にもたらすといったことをやらなければならない状況だと思っている。これについては、非常に大きなコストとリスクが伴っている。しかし、金融システムに求められる持続可能な安定性と信頼性を維持し続けるために、このリスクをコントロールしながら膨大な守りの投資を投じてい

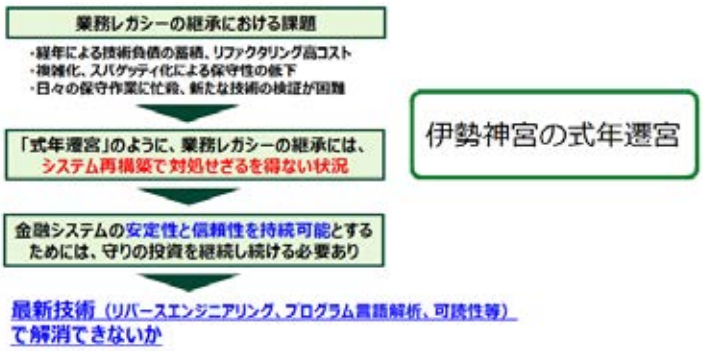


図 3.2-8 業務レガシーの継承（式年遷宮）

るということだと思う。

最近のスマホといったものに慣れている人は、ワンクリックで新しいシステムができるように思っているかもしれないが、最新のテクノロジーで業務レガシーの継承も簡単にできないだろうかということを大きな課題認識として持っている。

刷新できるレガシーアセットのイメージを図 3.2-9 に示す。従来は、ハードウエアからOS、ミドルウエア、アプリケーションのレイヤーまで全て自前で個別に金融機関としてつくっていくことが当たり前という時代だったが、保守切れに伴って基盤更改をするとか、定期的な設定変更とか、保守・障害対応といった維持管理と言われる費用が増大していくことになる。新しくつくるシステムや、リスクの小さいシステムはクラウドにつくることができるが、古くからあるシステムはコンテナ化などで仮想化のレイヤーを上に上げていくことで、刷新のコストを下げるというようなことをやっている。さらに、仮想化のレイヤーを上げることで継承するレガシー資産である業務アプリケーションの開発に注力できるのではないかという将来感を持ちながらやっている。

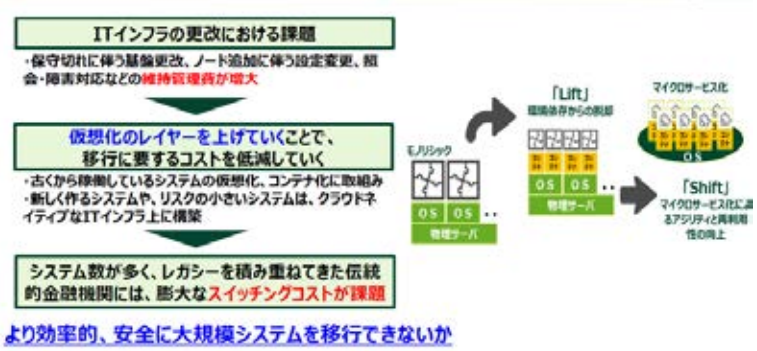


図 3.2-9 刷新すべきレガシー遺産

レガシーを積み重ねてきた伝統的な金融機関としては、既存のシステムからの更改をするためのスイッチングコストとスイッチングリスクの増大というのが課題になっており、少々システムは止まっても構わないという社会環境の中であれば、いろいろな実験ができるが、実際は止めるわけにはいかないので、スイッチングコストとリスクの削減に資するテクノロジーに大きな期待をしている。

続いて２つ目の手段は、デジタライゼーションを支える「サイバーセキュリティ」ということで、私たちの課題を共有していただきたいと思う。

金融機関を狙ういわゆるサイバー攻撃というのが増加している。世界各国でいろいろな形で金融機関は狙われている（図 3.2-10）。

図 3.2-10 金融機関を狙うサイバー攻撃

接続先のシステムからいろいろな個人情報が漏洩するということはよくあることだが、特に日本で問題になっているのは、われわれ金融機関そのものが狙われて、そこからお金が盗まれるわけではなく、巧妙な手口でわれわれの金融機関に口座をお持ちのお客さまがだまされる事例というのが非常に増えてきていることだと思う。

具体的な数字を図 3.2-11 に示す。不正アクセスの件数は右肩上がりで上がっており、標的型攻撃件数もすごい勢いで伸びてきている。また、DDoS 攻撃の数も大変な勢いである。

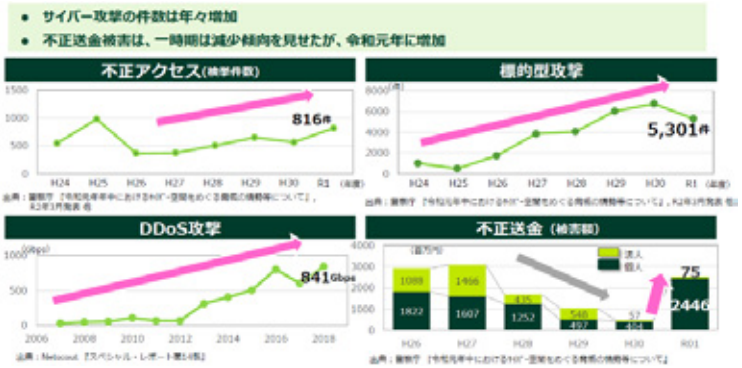


図 3.2-11 サイバー攻撃による被害の状況

このようにサイバー攻撃の数が大変な勢いで伸びてきている一方、不正送金の額もいったんは収束傾向にあったが、去年あたりからまた急激に増加している傾向にある。

不正送金というのは、お客さまがだまされてしまって、図 3.2-12 に示した手口のように、偽のショートメッセージが送られてきて、メッセージに書かれたアドレスをクリックすることによってお客さまの口座が何らかの形で攻撃者に乗っ取られ、金融機関のルールに基づいた形で送金されてしまうというようなことである。こういった不正送金が、頻繁に起きている状況である。

図 3.2-12 フィッシングの手口

図 3.2-13 は、フィッシングへの対応から得られた知見と課題認識と問題提起を整理したものである。テクノロジーの観点では、ID・パスワードに頼っている限りフィッシング脅威はなくならず、認証は必ず突破されるということである。今後、パスワードレスを確実にしていくために生体認証や行動認証といった、さまざまなテクノロジーが入ってくると思われる。また、端末やブラウザの設定に対する不正や改ざんを自動的に検出して自律的に修復する技術といったものも求められていくのではないかと思っている。

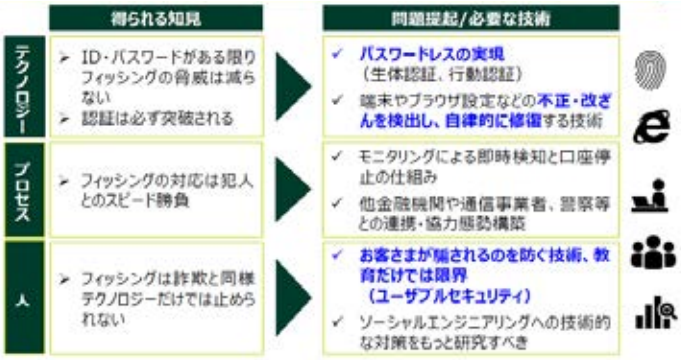


図 3.2-13 フィッシング対応から得られる知見と問題提起

次に、フィッシングの対応というのは犯人とのスピード勝負だと考えている。われわれがスマホを使ったモバイルバンキングの利便性を上げてスピーディーに処理ができるような仕組みを入れると、そこに対する攻撃者の興味が湧いて、何とかしてそれを突破しよう

とするので、言ってみれば攻撃者のトレーニングをし続けている感じだと思う。こちらサイドが防御のレベルを上げると、さらにそれを上回るテクノロジーを使ってくる感じで、いたちごっこになっている。

最後に、フィッシングというのは詐欺と同様なので、テクノロジーだけでは止められない領域が結構ある。お客さまがだまされるのを何とかして防ぐという、「ユーザブルセキュリティ」というキーワードを示しているが、そういった技術の開発というものも求められていると思う。

参考として、図 3.2-14 に示したものは、われわれのグループ会社で、アイルランドの生体認証の技術を持つ会社と一緒につくったポラリファイというサービスである。現在このサービスをお客さまに展開しているところである。

図 3.2-14 ポラリファイ

最後に３つ目の手段である、データ活用とプライバシー保護を両立する「データ・テクノロジー」についてお話したい。

データ・テクノロジーに関しては図 3.2-15 に示すような、３つの論点がある。

図 3.2-15 データ・テクノロジーの論点

１つ目は、データガバナンス。クラウドに展開した膨大なシステムには大量で多様なデータが存在する。そのデータを効率よく管理するための技術としてデータカタログ、メタデータ管理、といったものにわれわれは期待している。

次に、データ流通促進。金融機関においても、クラウドの活用、API 連携などを通して、いろいろな企業や組織間のデータ連携が拡大している。データを確実に守りながら活用していくための技術としての高機能暗号、これが今後どのように進展していくかに対しては大きな期待がある。

３つ目は、データ分析の民主化。民主化という言い方をしているが、データ利活用の負担を軽減していきたいということである。データ分析基盤の整備を進めているが、高度な専門スキルを要求されるデータ分析業務をツール化したり、自動化したりして、企業内に蓄積されるビッグデータの活用を通じた課題解決を加速するといったことを考えていきたい。

図 3.2-16 はデータガバナンスの内容を示したものである。データの中身や定義を正確に把握しながら、継続的に安全・安心に利活用するためのデータガバナンスの構築が不可欠だと思われる。そのためには、データカタログだとかメタデータ管理といった最新技術で課題解決できるのではないかと考えている。トップレベルの研究者にとっては簡単なのかもしれないが、末端ユーザーが利用できるようにしてもらえないかと思っている。

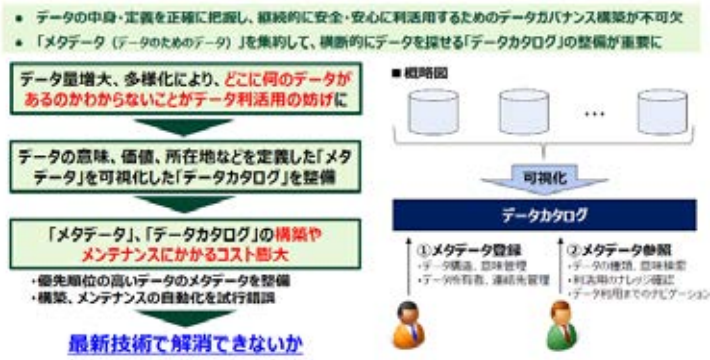


図 3.2-16 データガバナンス

図 3.2-17 に示しているのは高機能暗号への期待である。金融機関でも、ここ数年間で、クラウドの利用が飛躍的に進んでいる。弊社のグループだけでも 300 に近いほどのシステムがクラウドに移行している状況である。クラウドを利用したり、オープン API を利用したりすることによって、外部へデータを出す場面が増えてきている。さまざまな企業間、あるいは組織間で、データ連携をやっているので、これから安心・安全なデータ流通の実現はとても重要なテーマである。

図 3.2-17 データ流通促進

図 3.2-18 はデータ分析をいかにして簡単にできるようにするか、を示したものである。データ分析に関しては、さまざまなことをやっているのだが、手作業で行っており、かなりの時間がかかる。特徴量の作成や予測モデルの作成にも、高度なデータサイエンスが必要だというふうに従来から言われているが人材不足である。そこで、できる限りデータサイエンスのスキルレベルが低い人でもできる仕組みを作るために、Trifacta だとか dotData だとかいったデータ分析ツールを使ってデータ分析の精度を向上させ、作業期間を減らす努力をしているところである。

図 3.2-18 データ分析の民主化

最後に、図 3.2-19 に示す提言と期待でまとめたい。

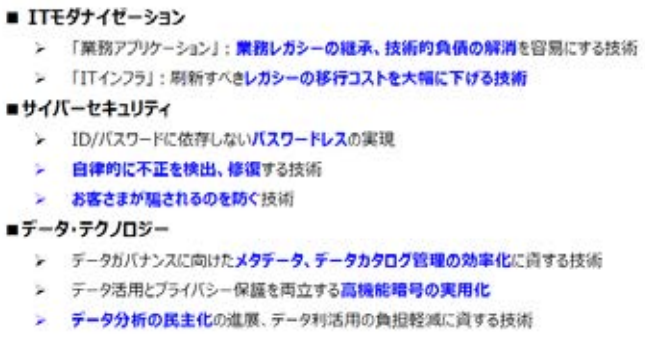


図 3.2-19 提言と期待

これまで話をしてきた課題認識の中で期待をしているのは、IT モダナイゼーションとサイバーセキュリティとデータ・テクノロジーである。業務アプリケーションにおいて、技術的負債の解消を簡単にできるような技術、IT インフラの移行リスクを少なくして、コストを大幅に下げる技術、こういった技術が出てくると、経営資源を新しいことに振り向けられるようになる。

サイバーセキュリティについては本当にいたちごっこだというふうに思うが、これからセキュリティ分野への投資はますます必要不可欠なものになってくる。セキュリティの投資を惜しむわけにはいかないので、一個一個のシステムをつくるときにはセキュリティ・バイ・デザインという形で、最初からセキュリティを理解した上で設計をしなければならないと思う。

データ分析については、これからの新たな競争力の源泉だというふうに思っているので、データ・テクノロジーの実用化を強くお願いしたい。

Society 5.0 に向けてわれわれも一生懸命頑張っていくので、研究者の皆さまのお役に立てるようなことは、これからもしっかりとやっていきたいと思っている。

【質疑応答】　（Q：質問、C：コメント、A：回答）

Q：３億行のコードは大体どれぐらいのペースで書き換わっていくものか。

A：３億行を全部取り替えるという作業は、やっていない。基幹システムと言われる勘定系のところはほとんど書き換えることなく動かしているというのが実態である。比率で言うと、書き換えているのは毎年、１割から２割ぐらいだと思う。その１割から２割をずっとやっていって３億行が全部換わっているかというと、そうではなく、５割から６割に近いところは書き換えられることなく、そのまま使い続けているといったイメージだと思う。

Q：私はソフトウエアの検証について研究をしているので、そういう大規模なソフトウエアというのは研究対象として非常に魅力に感じるのだが、例えば研究用にアカデミアにコードの一部なりを提供してもらうといった可能だろうか。もちろん、守秘義務は守るとして。

A：今までそういうことを考えたことがないので、すぐお答えしにくいが、過去にそのような研究を大学としたことがあるので、可能だと思う。

Q：１対１であれば比較的やりやすいと思うが、複数の大学のグループが共同して、いろいろなアプローチからコード解析をするということも可能だろうか。

A：可能だと思う。基幹システムを変更すると、膨大なテストをこなさなければならない、という現実がある。テストをしなくて済むような技術があれば、テストをやっているリソースを新しい領域に振り向けられるのではないかと感じている。

C：研究者にとっても非常に魅力的なターゲットに思えるので、機会があればお願いしたい。

C：開発の高度化といったあたりは、共同研究をできると、非常にありがたい。

Q：本来そういったことは協力しているITベンダーが提案すべきだと思うが、それがなされていないということは、もはや日本のITベンダーは役に立たないということか。

A：図3.2-5で説明した攻めの領域などで新しいことをやろうとすると、日本のITベンダーに頼んでも新しいアイデアがなかなか出てこないという感じがしている。一方、守りの領域である、従来のITインフラのところはあまりにもレガシーとしての特性を持っているので、急にアメリカの会社に見せても、やる気が起こらないと思うし、費用もかかるので、日本のITベンダーが頼りになる。

Q：頼りにはなるが、役に立っていないのではないか。

A： ITインフラを維持しなければならない面があるので、新しい領域にリソースを振り向けるためには、維持が早く、安く、簡単にできる技術が必要である。しかし、なかなかうまくいっていない。

Q：従来のベンダーに継続して依頼した結果こうなっているので、役に立たないのではないか。

A：更改とか新しいことをするたびに、既存のベンダーありきではなく、ゼロベースで検討を始めて、アメリカの会社にも声をかけて、どちらがいいのかという比較検討はするが、最終的には期間と費用の点で、日本の企業のほうが安く早くできてしまうということだと思う。

Q：大きなシステムというのは原則バージョンアップしてはいけないということは昔から言われていたことであり、バージョンフリーズをいかに上手にやるかというのが大規模ITにとっては一番肝だというのが常識になっているのではないかと思う。ベンダーに言われるまま、バージョンを上げてきた結果が現状ではないのか、という印象だが、そういうことを感じる局面はなかったか。

A：一番大きな基幹システムのところはなるべくバージョンアップしないで最低10年、できれば20年間ぐらい使うつもりでやってきているのだが、その周辺に、さまざまなシステムを追加すると、ミドルウエアやOSレベルでの保守切れに常に遭遇する。特に、オープン系のシステムで起きることで、対応せざるを得なかったというのがある意味事実だと思っている。

C：先ほど質問があったように、学者から見ても非常に深みのある技術課題だと思う。これができるかできないかでITベンダーの能力はほぼ全て決まるようなテーマだと思うのだが、大学はそういう現実の悩みを企業と一緒になって理解するという機会がかなり少ない。また、現状のインセンティブメカニズムの中では、こういう仕事にはあ

まり光が当たらないので、今までやってこなかったというのもある。
しかし､この話題にエキサイトメントをかなり感じるグループの人々が､今日はリモート会議なので見えないけれども、谷崎さんの前に 50 人ぐらい座っている。そういう、相当シュールな種族がいるので、ぜひこの問題をもう少し継続的に議論させていただけるとありがたい。
図 3.2-6 守りのデジタル化の概観として、①、②、③とあったが、②と③はグローバルに考えれば、そこしかないということは誰でも分かっている。①の問題は、アカデミアでなかなか訴求できない問題である。今日、そこを最初に言ってくれたことが大変印象的だったと感じている。

Q：ここで議論している先生方は、次のテクノロジーの開発をしようということで議論されている。SMBC が新しいものを入れていくときのプロセスと、アカデミアの先生方が研究されているものをどのタイミングでどうコミュニケーションして入れていくか、というのは１つのチャレンジかと思うが、そのあたりの進め方で、今まではやっていないが、こういう取組をしたらいいのではないか、といったことはないか。

A：日本総合研究所の中に先端技術ラボという、実業よりも少し研究に近いところの組織を２年ほど前につくった。そこを中心にアカデミアの方たちとも関係を深めていきたいと思っている。先ほど指摘いただいた、レガシーのところをどうするかといったことや、新しい技術の適用の仕方といったことも含めて、一緒に共同で開発など、いろいろと協力していきたいと思っている。

3.3 研究成果の活用（2）次世代 IT 基盤ソフトウェアへの期待

高橋 克巳（NTT セキュアプラットフォーム研究所）

ＮＴＴセキュアプラットフォーム研究所はＮＴＴグループの中央研究所の位置づけにあたる。200 名ぐらいの研究所で、半分がサイバーセキュリティで、残り半分がデータ・テクノロジーのセキュリティを研究している。私はデータ・テクノロジーの中の高機能暗号に取り組んでいるが、本日は企業がシステムにセキュリティで期待することについて話題提供する。

システムといっても具体的に運用されているシステムというより、漠然とクラウドのようなものを想定して説明する。セキュリティも、情報セキュリティとプライバシー保護を含んで説明する。

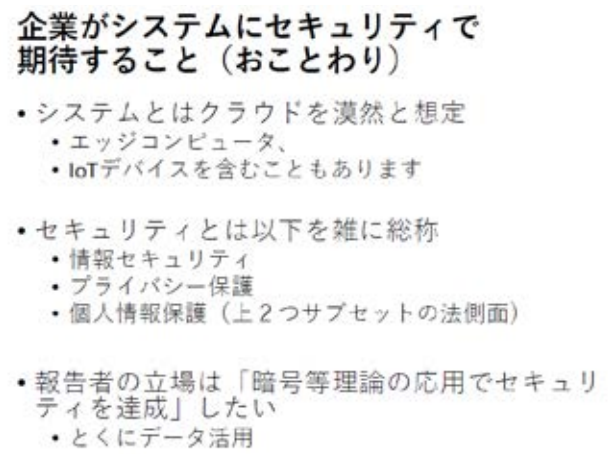


図 3.3-1　企業の期待 1

企業が期待する話として、まず、１つの結論を示す。「プロバイダー・オペレーター」の立場と「ユーザー」の立場で分けているが、セキュリティホールがないこと、データ漏洩しないことなどがある。

運用的な視点だと、新しい事業にチャレンジできることなども期待にある。出典は、ソースもあるが、乱暴に言えば「自説」である。

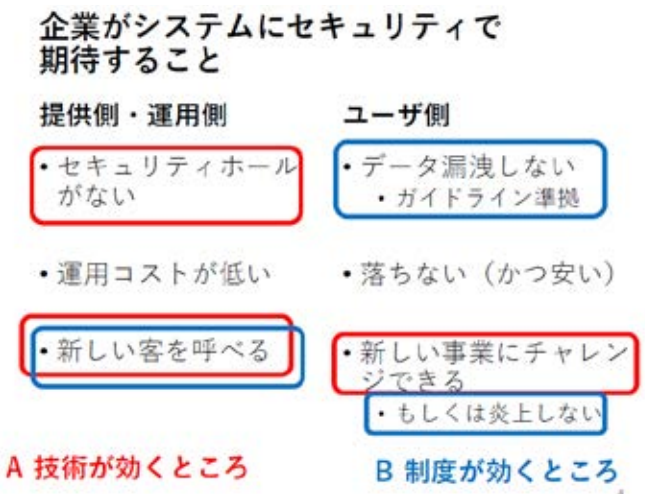


図 3.3-2　企業の期待 2

赤色の枠線が比較的に技術が効くところ、青色の枠線が比較的に制度が効くところとなる。例えば、「データが漏洩しない」「ガイドライン準拠」では、意識が高いユーザーだと本当にデータが漏洩しないということを旗印に考える。一方、意識の低いユーザーでは、ガイドラインに準拠していればいいやというような対応になってしまう。こういった構造がある。概要だが、技術に関して、要素技術としては高機能暗号などがあるが割愛する。

参考だが、セキュリティをデータ中心でどう見ているかを表にした。暗号コミュニティだと左上の機密性、可用性、完全性などがよく意識されている。一方、コードやシステムの側からは、本当にそれが信頼性があって、ずっとしっかり動くのかといったいう観点に広がる。プライバシー保護はさまざまな定義があるが、節度を守ることなのかなと考えて

いる。「データ・ミニマイゼーション」という言葉があるが、目的達成に対して最小であることがプライバシー保護の１つの考え方と捉えている。

技術が効くところとして、まず、セキュリティホールがないこと。これはプログラム検証やハードウエアセキュリティが間違いなく重要だ。ただし、それが現状のシステムにない場合、結局は導入時の検査と、運用時のモニタリングにならざるを得ないと認識している。

新しい価値へのチャレンジについて、暗号のコンテキストでどういう貢献があるか。PKIやブロックチェーンは偉大な成果なのだが、データ活用の手前までの話で、データ活用そのものではない。

そのため、秘密計算と匿名化が代表的だと言われている。ただし、これが基盤システムになり得るのかというと、いろいろな意味でアプリケーション止まりなのが現実だと思われる。パフォーマンスの問題や、匿名化は非常に局所的でもあり、保留されてしまうということがある。別の観点として、使いこなすのが結構難しい。結局のところ、Google や Apple が自分たちでつくって自分たちで使うから使いこなせているのであり、ディファレンシャル・プライバシーが入ったデータを普通の人が使いこなすのはかなり難儀なことだ。

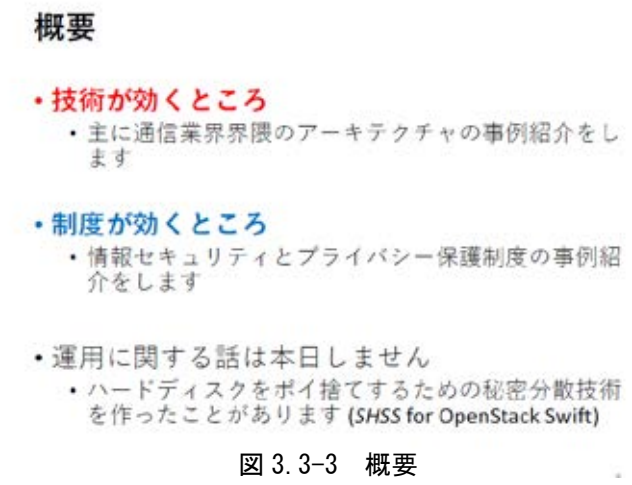
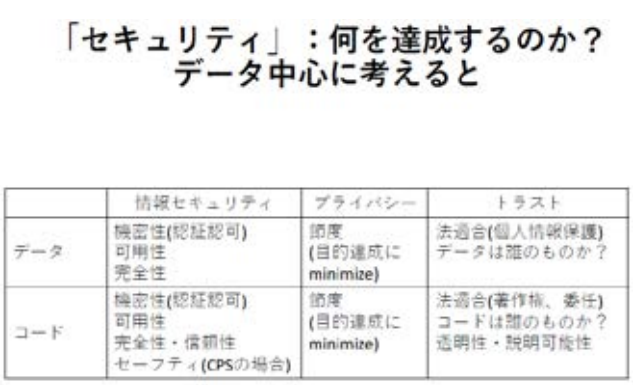


図 3.3-3　概要



	情報セキュリティ	プライバシー	トラスト
データ	機密性(認証認可) 可用性 完全性	節度 (目的達成に minimize)	法適合(個人情報保護) データは誰のものか？
コード	機密性(認証認可) 可用性 完全性・信頼性 セーフティ(CPSの場合)	節度 (目的達成に minimize)	法適合(著作権、委任) コードは誰のものか？ 透明性・説明可能性

図 3.3-4　何を達成するのか？

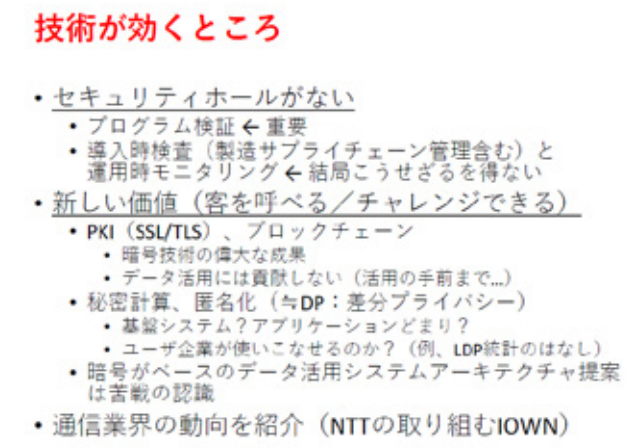


図 3.3-5　技術が効くところ

そこで、暗号がベースのデータ活用アーキテクチャーの提案を常に考えているが、正直に言って、これはなかなかうまくいっていない。

通信業界の動向として、IOWN 構想を NTT が１年前に発表した。端的に言えば、光を中心としたすごいネットワークを 10 年後につくるぞ、という構想だ。高速、大容量、高品質、低遅延、低消費電力の問題などを掲げ、端末やネットワークだけでなく情報処理基盤

まで提供したいと考えている。

３つの主要技術の特徴がある。まず、デバイスまで含めたオール光ネットワーク。次にコグニティブ・ファウンデーションは、通信も含めた IT の話だ。狭い意味で言えば、オーケストレータなどを考えている。そして、アプリケーション的だが、デジタルツインという言葉があちらこちらで最近また言われている。NTT もデジタルツインコンピューティングということを掲げている。これはオール光などととは独立して定義される問題と考えている。

図 3.3-6　IOWN 構想について
（図中の出典：https://www.rd.ntt/iown/ より）
2020年11月時点

つい先般、IOWN GLOBAL FORUM という形でロードマップを発表した。ロードマップを発表した時点で CFP を出したとことが、われわれとしては非常に変わった点だ。皆さんと一緒に解いていく形で、「four technical directions」に沿った提案を募集している。NTT が何か訳の分からぬことをいうだけではなく、皆さまのお力を本当にお借りして実現したいと考えている。

もう一つの特徴として、NTT はどうしても大きなアーキテクチャーというよく分からない図を描いてしまうのだが、それを反省した。ユースケースの図から入ろうという点が特徴になっている。

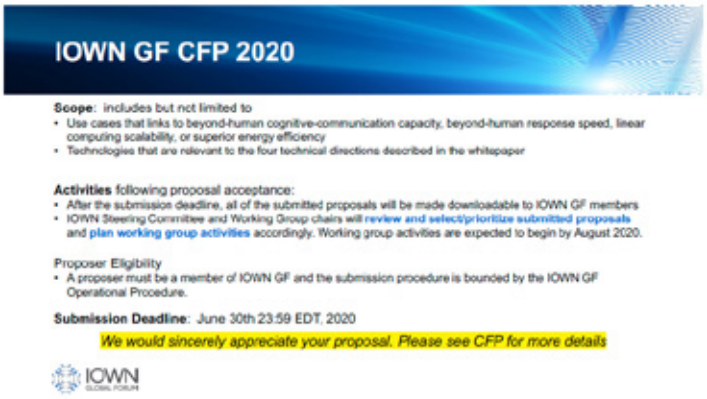


図 3.3-7　IOWN の提案募集

左側の何かを検知するシステムから、右側に行くにつれて検知したものからフィードバックがあるという図だ。右から２番目の XR は AR などで、一番右は AI による、予測も含めた、ユースケースを考えている。

一番大事だと考えているのは、データセントリックコミュニケーションハブという考え方だ。これをつくっていきたい。これはアブソーブモデルでデータ処理をしていく、あとはデータ共有を新しいアーキテ

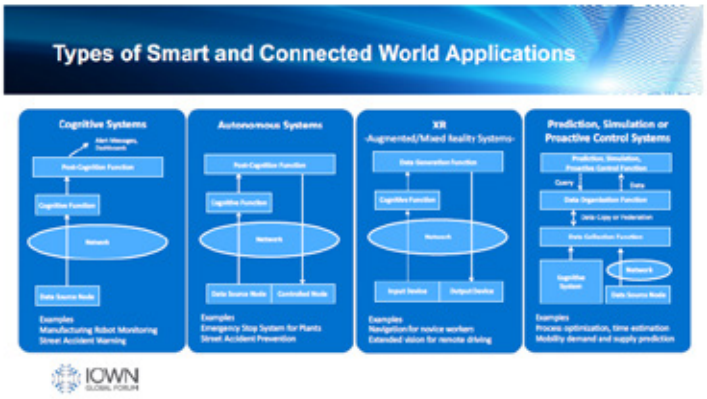


図 3.3-8　ユースケースによる分類

クチャーの中で進めていきたいという絵になる。

続いて、制度的な面について説明する。

まず、データ漏洩しないことが絶対重要だ。しかし、ユーザー企業側からは具体的なイメージがどうしてもできない。その結果として、ユーザー企業としてはガイドラインにしっかり準拠している、あるいは準拠させるという対応が支配的になってしまう。これは、あまり楽しい話ではないのだが、こういったことがどうしても起こっている。

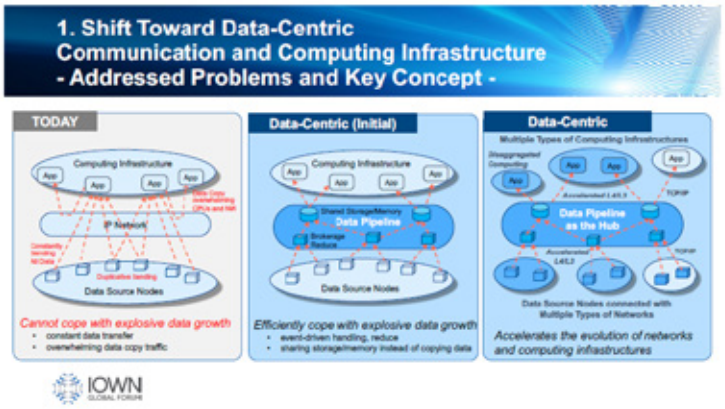


図 3.3-9　将来構想の概念図

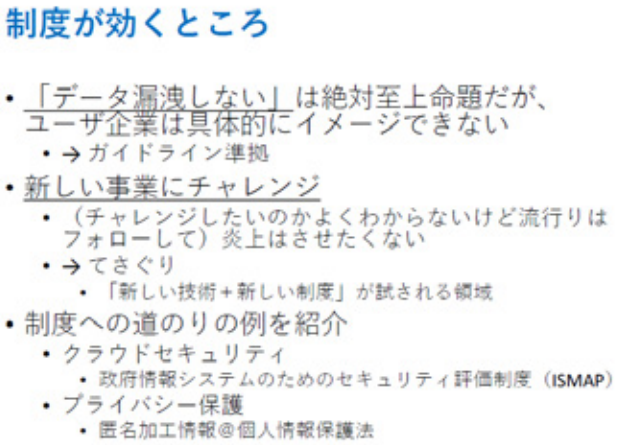


図 3.3-10　制度が効くところ

新しい事業へのチャレンジというコンテキストではどうか。先ほどの Google、Apple が local differential privacy を導入した話で、使いこなせている彼らだけはすごく良さそうである。しかしながら、他の多くの会社はセキュリティのコンテキストで、炎上させたくないといった姿勢にだけなってしまうのが残念だが現実だ。しかし、手をこまねいていてもよくない。われわれが新しい技術と新しい制度を提案し、その提案が試されていくべき領域だと考えている。言い換えると、今のいろいろな制度にきゅうきゅうとしているだけで、チャレンジしない言い訳をしていてもしようがない。精神論になってしまうが、新しい技術の可能性も含めて頑張って提案していきましょうと。

クラウドセキュリティの事例として、最近 ISMAP という評価制度ができた。これは政府の情報システムのセキュリティ基準の評価制度だ。民間とは関係ない面はある。

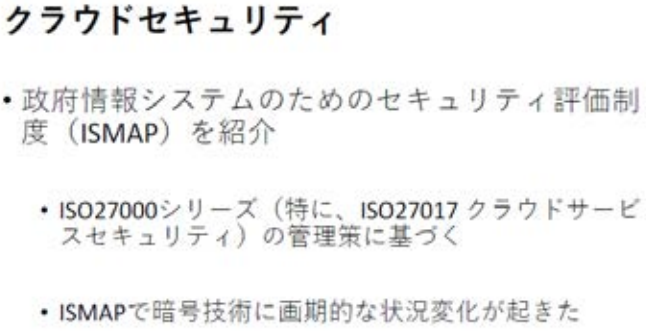


図 3.3-11　クラウドセキュリティ

内容は ISO27000 シリーズなので、それ自体は特に目新しいことはない。ただし、この扱い方についての配慮でよいところがある。強調したいのは、暗号技術に画期的な変化が起きたことである。

多くのセキュリティ要求基準があってよく分からないし非効率な問題がある。ISMAPの基本的なモチベーションとして、これを解決したい。セキュリティ要求基準を統一し、それを政府の調達基準にして評価することで解決する方向性だ。具体的にどうやって統一するかが課題である。

１番はいわゆるJIS Q 27000シリーズで、特にクラウドに関する27017を中心につくったもの。２番は日本国政府の統一基準と呼ばれているガイドライン等。３番はNISTのSP800シリーズで、幾つかあるが、53が代表的だ。１、２、３の和集合を整合の取れた形でISMAPと定義した。

ISMAPでは、暗号鍵を消去すれば暗号化データを消去できたこととすると、日本では初めて認められた。データ消去と簡単に言っても、ご存じの方も多いが、なかなか消せないという実情がある。データが暗号化されていて、その暗号鍵を正しく消去すれば、データを消去したこととすると基準化した。それに伴って、暗号化・鍵管理機能は最初から提供してくださいという原則がISMAPの管理基準として定められた。これまでは物理的に破壊するか、消磁の箱に入れるか、あるいは何回も上書きするという方法のみがNISCの基準と、日本のガイドラインで認められていた。それに論理的消去も加えられ、暗

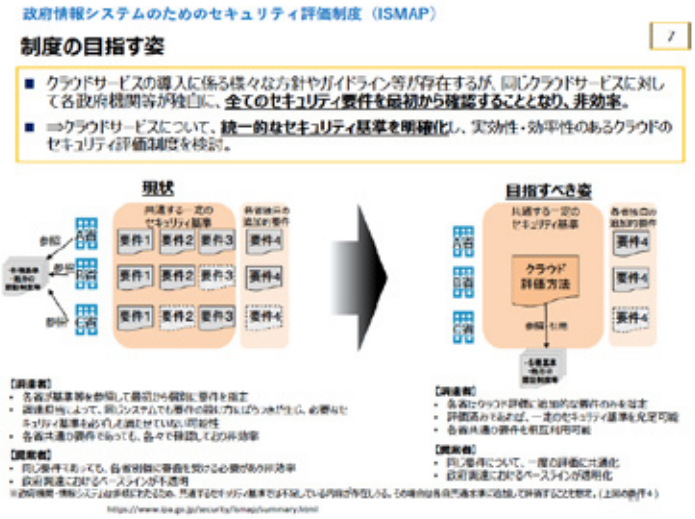


図 3.3-12　制度の目指す姿
（図中の出典：https://www.ipa.go.jp/security/ismap/summary.html）
2020年11月時点

図 3.3-13　ISMAP管理基準の構成Ⅰ
（図中の出典：https://www.ipa.go.jp/security/ismap/summary.html より）
2020年11月時点

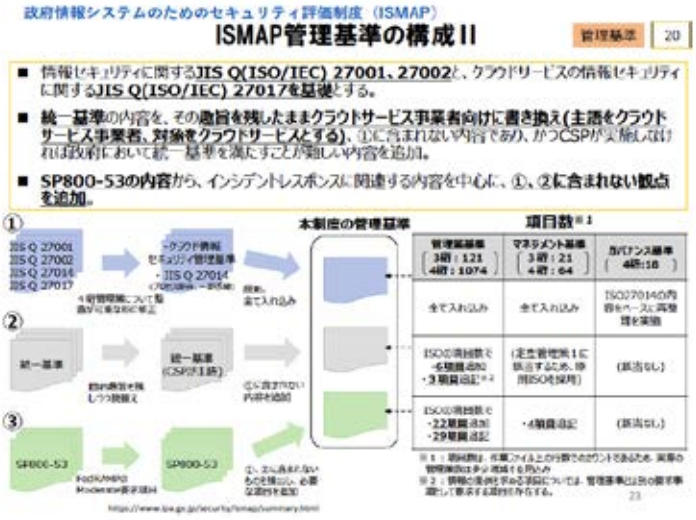


図 3.3-14　ISMAP管理基準の構成Ⅱ
（図中の出典：https://www.ipa.go.jp/security/ismap/summary.html）
2020年11月時点

号業界は非常に大喜びの話ができた。

これはすごく小さい話だけれども、暗号が情報基盤システムに入るための本質的な機能としての位置づけがこれまでずっと定まっていなかった。ISMAP の運用が始まったことは喜ばしいことで、日本国政府にクラウドを提供する主体も対応を準備しているところだと想像する。

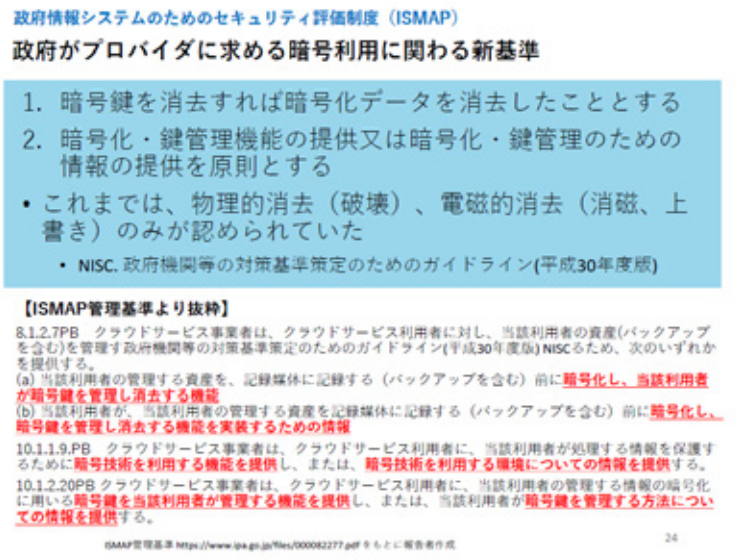


図 3.3-15　暗号利用の新基準
（ISMAP 管理基準 https://www.ipa.go.jp/files/000082277.pdf をもとに報告書作成）
2020年11月時点

プライバシー保護技術は DP（Differential Privacy）と秘密計算に尽きるという意見に私も賛同する。これら２つは扱いにいろいろな制約があるので、まだ本当のシステム基盤ソフトウエアにはなっていない。

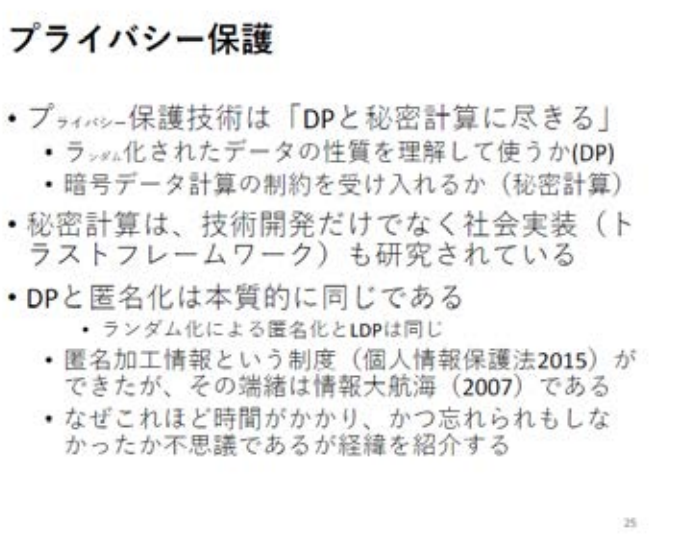


図 3.3-16　プライバシー保護

DP の制約は、ランダム化されたデータの性質を理解して使えるかという話に戻る。Google、Apple なら自分で作っているからノイズが乗ったデータがどういう素性なのか分かる。しかし、ノイズの乗ったデータを提供するのは、提供先に嫌われるというところがやはり根強くある。一方、暗号化データ、高機能暗号はパフォーマンスを含めた制約がある。どちらを受け入れるかという話だ。もちろん、両方を相互補完的に使えるのだが、それぞれの制約は引き続き残る。

秘密計算の技術開発については、誰が誰を信じ、誰を信じないかという整理をしっかりしないと、何か暗号で計算できて嬉しいという次元の話になってしまう。そのため、社会実装、トラストフレームワークも研究している。

DP と匿名化の詳細は省くが、ランダム化による匿名化と local differential privacy は本質的には同じものと考える。そういった意味で DP もしくは匿名化が大事な要素だと説明できる。匿名加工情報という制度は約３年前にできたときに話題になった。その端緒は情報大航海プロジェクトだった。2007 年から 10 年もかかったが、忘れられずに、世の中で必要だとされて今に至った経緯を紹介したい。

k-匿名化は L.Sweeney 先生がもともと提唱された。情報大航海プロジェクトの中でプラ

イバシー保護技術の検討が一部検討された。それを NTT が受託して、k- 匿名化と併せてランダム化が重要だと報告した。ランダム化は LDP の話と全く同じだ。その後、プロジェクトの中で日立が中心となって k- 匿名化の匿名化基盤を開発した。インパクトがあったが直接的には盛り上がらなかった。2012 年に総務省がパーソナルデータ研究会を立ち上げ、私は構成員として、匿名化を紹介した。この研究会から個人情報保護法での匿名化の検討が始まり、内閣官房で検討され、2015 年に個人情報保護法ができて、匿名加工情報が制度になった。NTT では匿名化の研究はしていたが、システムはつくっていなかった。こういった形になったので、ソフトウエアをつくって、クラウドで実装する形になった。ここから得られた教訓は、コンピュータサイエンスとすればつまらない話かもしれないが、時間がすごいかかってもこつこつやることで何か変わるということだ。

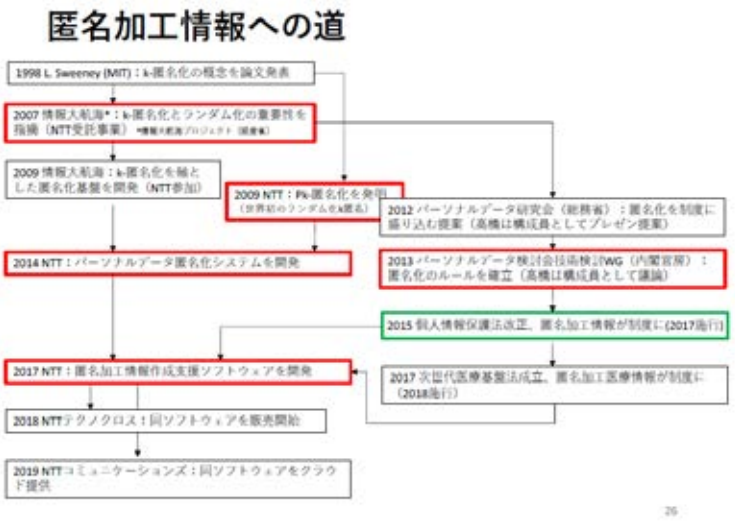


図 3.3-17　匿名加工情報への道

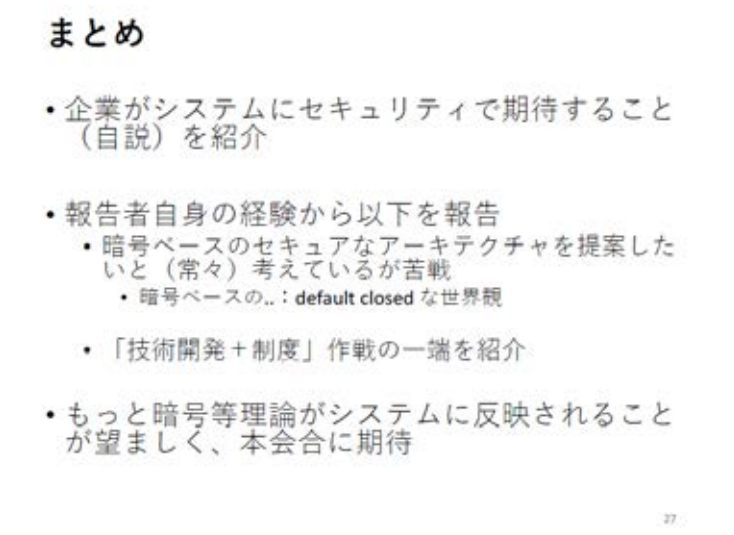


図 3.3-18　まとめ

以上をまとめる。まずは自説を期待することして述べた。暗号ベースのセキュアなアーキテクチャーは端的に言えば、default closed な世界観だ。すごいインターネットではないが、default closed での情報処理は、セキュリティとしてよいと考えている。正直に言えば苦戦しているのが事実だ。、「技術開発＋制度」の作戦もそれなりに大事で、手応えもあった。こういった暗号等理論がシステムの安全性を守ることにもっとストレートに反映されることが大事だ。

【質疑応答】　（Q：質問、C：コメント、A：回答）

Q: 高機能暗号や秘密計算の話の中で触れられていたパフォーマンスの制約について質問したい。まだそんなに簡単にはいかないという、パフォーマンスの制約はどのくらいのものと思えばよいか。

A：秘密計算では良い数字が記憶になく、出てこない。しかし、統計計算に限れば瞬時にできるものが幾つも出てきている。ある意味では、統計計算は役に立つと言っていい。ただし、スケールのさせ方が違う。本当に大規模データ処理に役に立つかについて、まだいろいろ議論がある。

パフォーマンスは暗号方式によってもだいぶ違う。秘密分散を基にしたものが一番速い。準同型暗号のほうがまだ二桁ぐらい遅い。特に準同型暗号の研究だとパフォーマンスが出せず、個別のチューニング、個別のアプリケーションづくりのようにどうしてもなってしまうため、一般的なシステムにしようと意識されている。

3.4 研究成果の活用（3）
データ活用セキュリティ＆プライバシーにおける基盤技術の展望

千田 浩司（NTT セキュアプラットフォーム研究所）

2000 年 NTT 研究所 入所、暗号技術、プライバシー保護技術の研究開発を行ってきた。現在は、セキュアプラットフォーム研究所のデータ保護技術グループのグループリーダ。

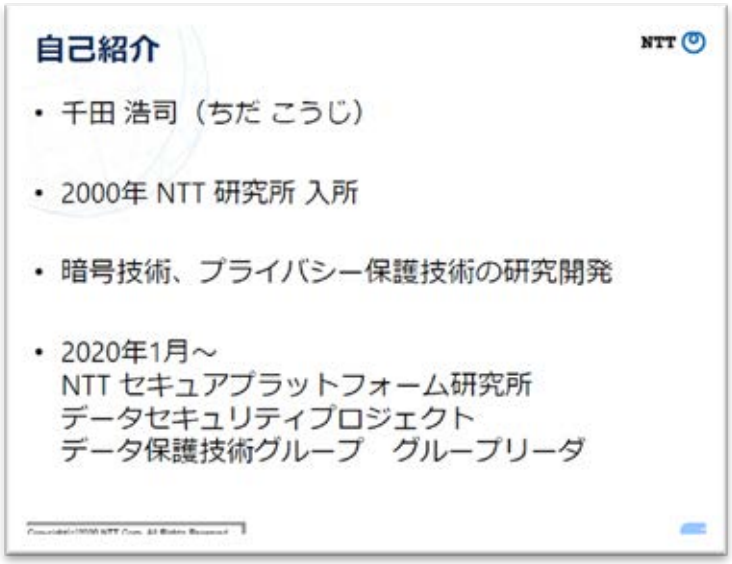


図 3.4-1　自己紹介

入社当時は、電子投票の研究テーマ。インターネットを使って、厳格な選挙に用いる技術を目標に行ってきた電子投票の要件は、匿名性・投票の秘密、1 人 1 票まで、結果の正しさを満たすことであるが、匿名性を確保してどうやって一人一票にするか、投票は秘密にされるのに、どうやって結果が正しいことを証明するのかという矛盾をはらんだ要件になっている。これに対して、投票者がインターネットの通信路を通じて投票して、その先に「安全な箱」というのがあって、覗き見はできないし、一人一票という入力制御があって、改竄できないような機能がある万能な箱があれば電子投票は実現できるが、実際にこのような安全な箱はないということに気づき、暗号技術を検討してきた。

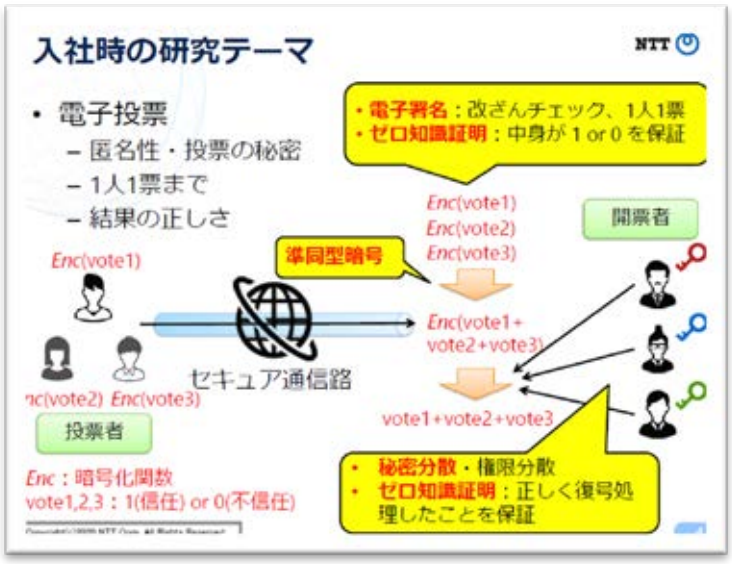


図 3.4-2　入社時のテーマ

暗号技術等を駆使して、安全な箱に見立ててソリューションを実現していく。図の左下にある Enc が暗号化関数、vote1,2,3 が投票者の投票内容とする。簡単のために 1 か 0 が入っていて、1 だと信任とする。暗号文に投票者各自の署名を打って送る。そうすると図の右上に暗号文が送られてくる。まず電子署名により、改竄が行われていないか、一人一票が確認できる。この時点では暗号化されているので投票内容も秘密にされる。さらに投票内容は 1 か 0 であること、10 とか 100 とか水増しされていないことを証明する「ゼロ知識証明」という技術もあって、完全性を強化できる。さらにその

先に、先週、今日も話があったが、準同型暗号というのを使うと暗号化したまま足し算ができる。その足し算された暗号を復号すると何票入ったかということだけが分かる。さらに暗号を解読できるような強力な人がいると、図の上の vote1,2,3 が解読できるということがあるので、ここで「秘密分散」や「権限分散」という技術を使って、何人かが集まらないと復号できないという機能や、分散された鍵を持つ人たちが正しく処理したことを示すゼロ知識証明を使うことで安全な箱を見立てたような解決ができる、という技術で、20年前から研究してきた。

データ活用：現在はデータ活用のセキュリティやプライバシー保護について研究開発をしている。この図は IoT の普及や計算基盤、AI の実用化などの背景から、データ活用への期待は大きいが、その上で、セキュリティやプライバシー保護が問題になってくる。

図 3.4-3　データ活用
（図中の出典：https://labevent.ecl.ntt.co.jp/forum2018a/info/smartworld.html）
2020年11月時点

データ活用のモデル化：データ活用でセキュリティやプライバシー保護の問題を明確にするために、モデル化ということを行っている。図の左上にあるデータ提供主体自身の個人や組織のデータ、あるいは IoT 機器から送られてくるデータが、提供主体のところでまとめられる。例えば、収集主体から二次利用主体に、収集したデータを他社に提供するというパターンであったり、あるいは、図の右側に活用主体とあるが、データを使って分析結果がほしいような人は、処理の分析基盤を使ってデータを受け取る、というフローがある。さらに最近の機械学習、深層学習といった用途では、図の右下のコード提供主体というところで考案した学習モデルをこの処理基盤に入力して、活用主体が利用して結果を得る。このモデルにおいては、コード提供主体自身も収集主体のデータを使って学習モデルを形成していくというケースも考えられる。こういったモデルを一つ考えたうえで、どこにセキュリティの問題があるか、プライバシー保護の問題があるか、明確にしな

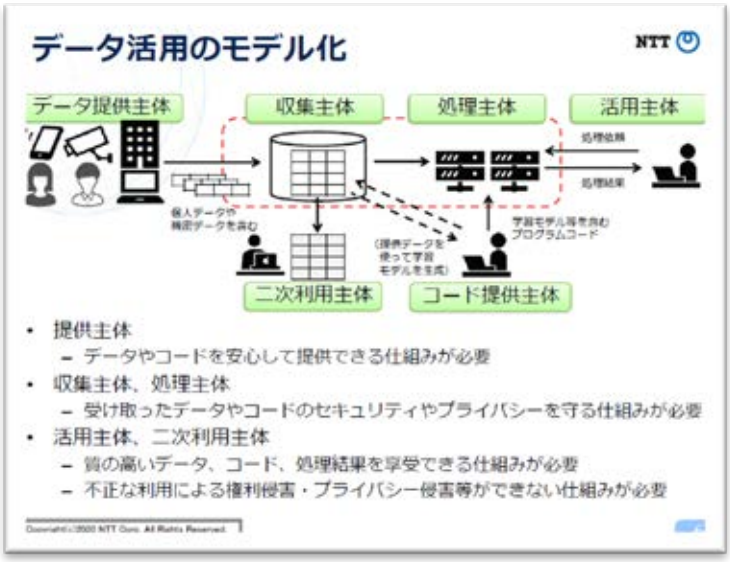


図 3.4-4　データ活用のモデル化

がら進める、というアプローチを検討している。簡単に言うと、データ提供主体がデータやコードを安心して提供できる仕組みが必要である。収集主体や処理主体は、同じ組織や違う組織の場合もあるが、受け取ったデータやコードのセキュリティや プライバシーを守る仕組みが必要である。活用主体、二次利用主体と定義しているところでは、質の高いデータ、コード、処理結果を享受できる仕組みが必要。また、活用主体、二次利用主体が悪意を持って不正な利用をすることで、データ提供主体等の権利侵害、プライバシー侵害が起きないことが重要になってくる。

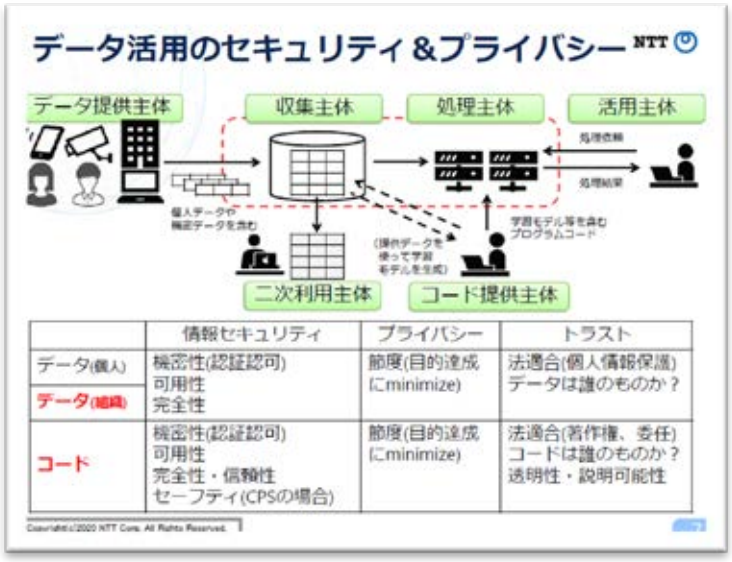


図 3.4-5　データ活用のセキュリティ＆プライバシー

データ活用のセキュリティ & プライバシー：具体的な検討すべき技術に落とすため、先ほどの高橋の表を再掲している。データ、コードの縦軸と、情報セキュリティ、プライバシー、トラストという横軸がある。個人のデータのセキュリティやプライバシーは重要であると前から取り組んできているところであるが、今回は組織のデータのセキュリティやプライバシーという観点で話をする。コードのセキュリティという観点では先週、コードの完全性、信頼性の観点で話をいただいたが、コードの機密性や知的財産の保護といった観点に資する技術についても話をしたい。

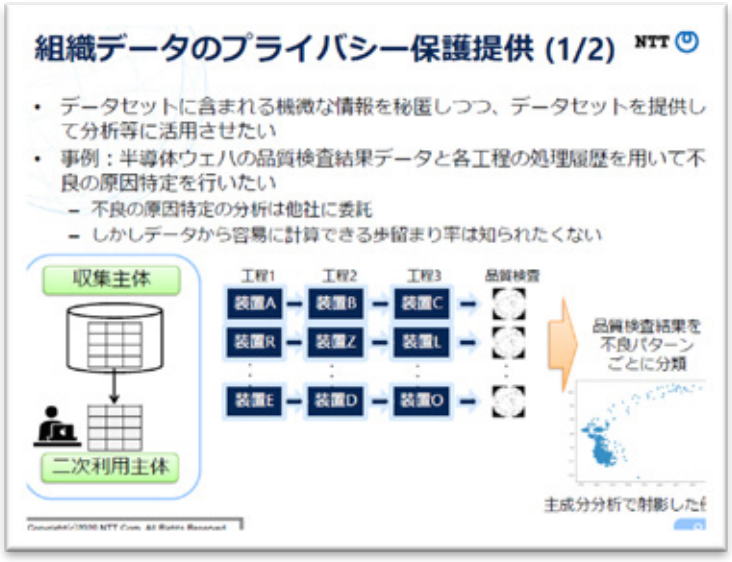


図 3.4-6　組織データのプライバシー保護提供（1）

組織データのプライバシー保護提供：図の左下は、先ほどのモデルの一部を抜き出してきたもの。収集したデータを加工して安全に二次利用に提供する。これはいままで匿名化や差分プライバシーという技術の適用シーンと同じと思うが、一つ違いがあるのが、組織のデータを取得するという観点である。データセットに含まれる機微な情報を秘匿しつつ、データセットを提供して分析等に活用させたい。何が違うかというと、具体例として半導体ウエハの事例で、半導体ウエハの品質検査結果データと各工程の処理履歴を用いて、AI を使って、不良の原因特定を行いたい、ということを考えている。問題となるのが、まず実

際にデータ分析をするのが、他社に委託するというシーンを想定している。ところが、そのままデータを渡してしまうと、そのデータには企業の機密情報であるような歩留まり情報を含んでいて、これは非常に機微な情報で知られたくない。データは提供したいがそこに含まれる機微情報は知られたくないという相反する要件が出てくる。実際に図の右下のイメージ図で示しているが、処理工程や品質検査というような元データに対して、やりたいことは不良の原因特定で、例えば、品質検査結果をクラスタリングして測定するということをしたいが、歩留まり情報は知られたくないということである。

ひとつ目は先ほどの再掲、二つ目は SCIS2020 で、東芝と一緒に共同検討を行っており、半導体ウエハの品質検査結果と各工程の処理履歴を疑似データとして生成する。ただし、やりたいことは歩留まり率は知られたくないという要件から、元データに対して、疑似データから得られる歩留まり率というのは、元データから得られるものと十分異なる。ノイズを入れることをするが、分散が大きくなるようにして隠す。かつ、データを使ったクラスタリング（k means, BIRCH 等）の結果は元のデータから得られるものは、疑似データで得られるものと近い。こういう２要件を満たす疑似データ化に取り組んでいる。こういう具体例でも両立は難しい。データから得られる機密情報を隠したままデータ提供するというニーズはかなりあると考えている。こういう問題を一般化すると図の右下に示すような要件になると思っている。元データ D に対して、疑似データ化した D' としたとき、D から得られる秘匿したい値 D^*について、疑似データ D'， 差像 F が得られても、元の D^*の推定はできないという安全性の要件と、元のデータからいろいろ分析したいというときに、疑似データからも同じように分析したときに、同じような結果がしっかり得られるという有用性が両立する技術の研究に取り組んでいる。具体例で取り組

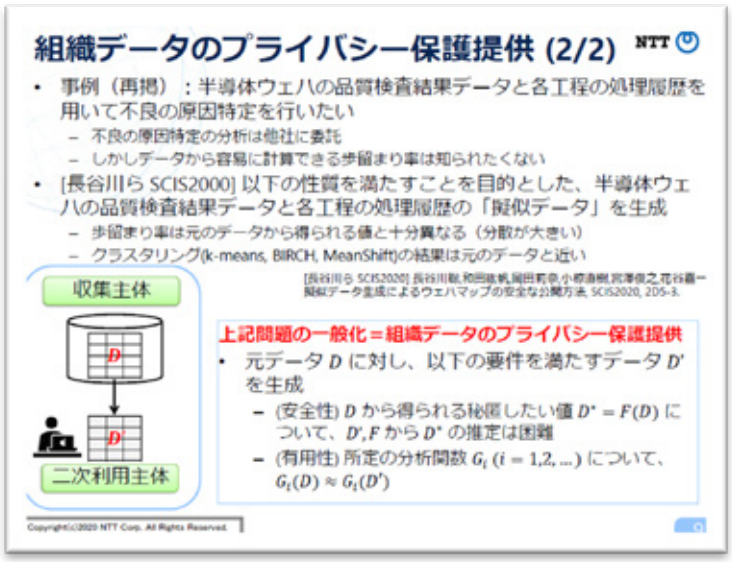


図 3.4-7　組織データのプライバシー保護提供（2）

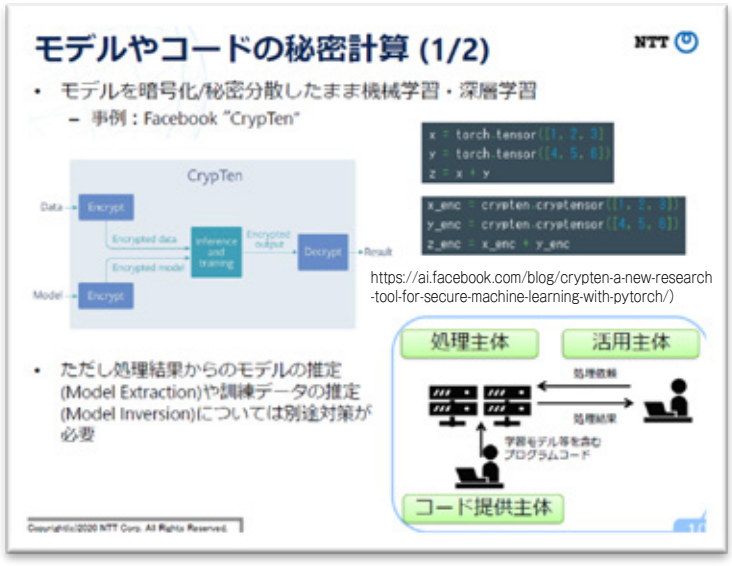


図 3.4-8　モデルやコードの秘密計算（1）

(図中の出典：https://ai.facebook.com/blog/crypten-a-new-research-tool-for-secure-machine-learning-with-pytorch/)
2020年11月時点

んでいるが、理論的にどこまで達成できるのか、どういうクラスまでできるのか、まだ分かっていない状況なので、今後基礎研究として取り組みたい。

モデルやコードの秘密計算:学習モデルの秘密計算にフォーカスする。例えば、Facebook は CrypTen と呼ばれる秘密計算ライブラリを公開しているが、これは学習モデルを暗号化、秘密分散したまま学習できるということを謳っている。図の右上に示すように、テンソルの値をｘ、ｙに入れて、それを足し算してｚに入れる、というのを Pytorch で書かれた構造のものであるが、これを CrypTen で書くと、それぞれの処理が暗号化されるようになり、そういうコードを開発者が書いて、これをコンパイルすることでモデルを秘匿したまま処理できる。ただし、テンソルのベクトルの要素とか計算した結果は隠せるが、どういった関数を処理するとかどういった演算をするとかいうところまでは隠していないので、完全に隠しているわけではない。図の左下のように処理結果から漏れる秘密情報という観点では、そこまでの考慮はなされておらず、最近では処理結果から学習モデルの推定を行う Model Extraction という攻撃や訓練データを推定する Model Inversion という攻撃があるので、こういった技術の組み合わせを行う必要があると思う。図右下のモデルは、コード提供主体が処理主体にコードを提供するときに、秘密情報や知的財産情報を秘匿しながら活用させるというモデルになる。

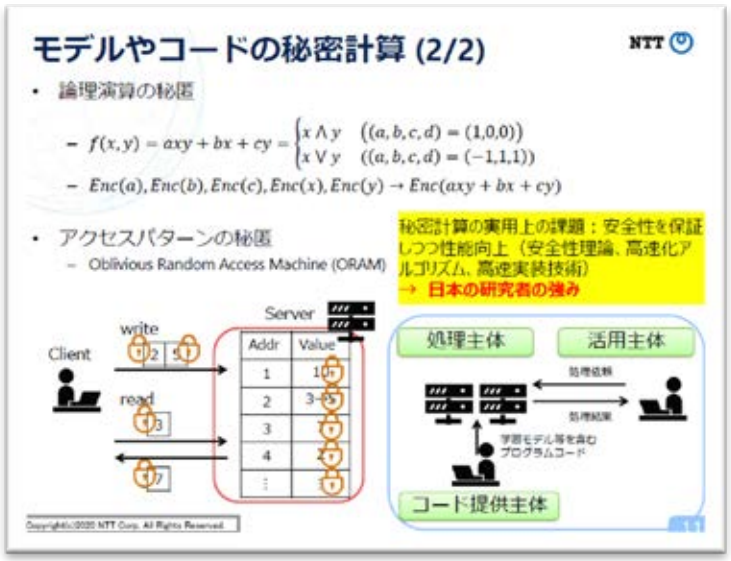


図 3.4-9　モデルやコードの秘密計算（2）

さきほど、学習モデルのパラメータは隠せるというのを Facebook の例でお伝えしたが、実際どういうロジックで処理したかは隠せないという話をした。こちらは基礎研究と思うが、論理演算の秘匿というものが古くから知られているので紹介する。論理演算の秘匿と書いてあるのは and ゲートか or ゲートで処理したかわからない例で、古くから概念は知られていた。ｘ、ｙに各ビットが入力されて多項式 axy+bx+cy という計算において、係数ａ、ｂ、ｃ、が１，０，０であれば and ゲートの処理、－１、１、１であれば or ゲートの処理になる。これは係数も取得して、入力も取得する。a、b、c、x、y 全部暗号化しておいて準同型暗号で積和を求めて暗号を元に戻さずに計算するということで and ゲートか or ゲートがわからないが、どちらかで計算したことが実現可能になる。さらに論理演算の場合、今一つのビット演算の話をしたが、実際には論理回路で考えると、さらにビット演算の結果を次のゲートの入力にしてというワイヤーが入り組んで処理されるわけで、ワイヤーの処理までは秘匿できていない。これも秘匿できるようにするのがアクセスパターンの秘匿で、Oblivious Random Access Machine（ORAM）と呼ばれる技術である。左下の図のように、クライアントとサーバーがいたときに、クライアントがどの番地にどのデータを書き込むか、あるいは、どの番地のデータを読み込むか、一切サーバー側に見せずに処

理できるという技術である。こちらはまだ研究レベルと認識しているが、こういったことも実現できるので、論理演算の秘匿と組み合わせて、回路の秘匿をしたまま計算ができることが理論上可能になる。こういった技術を組み合わせてわれわれの方ではインテグレートする、高速化するという研究をこれまでしてきているが、秘密計算の実用化は性能の問題がある。ただ性能をあげればよいものではなく、安全性を満たしつつ性能を上げるという条件を満たす必要がある。これは安全性理論、高速化アルゴリズム、高速実装技術が組み合わさって初めて実現可能になる技術で、国内の研究者の強みだと思っている。われわれも安全性の理論のところで大学の先生方と連携して日本の強みを生かしたい。

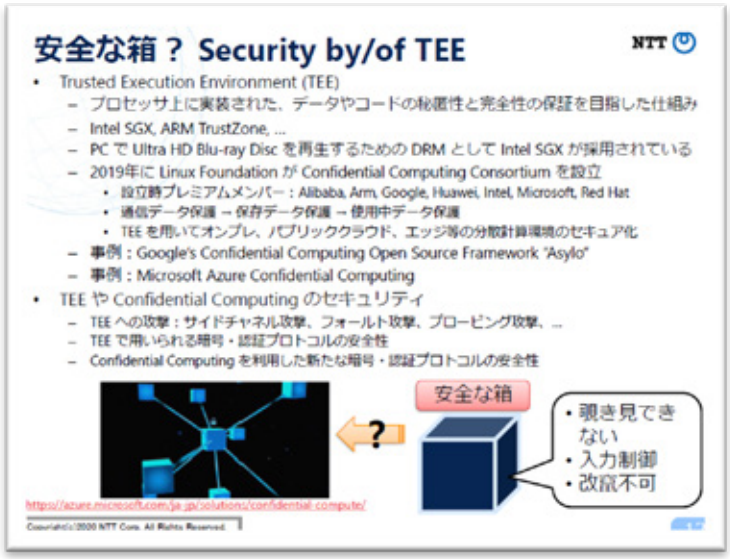


図 3.4-10　安全な箱？ Security by/of TEE
（図中の出典：https://azure.microsoft.com/ja-jp/solutions/confidential-compute/）
2020年11月時点

安全な箱？ Security by/of TEE：電子投票に戻るが、安全な箱を実現するというアプローチとして、Trusted Execution Environment（TEE）があると考えている。TEE を使ってセキュリティを守る、TEE のセキュリティの両面がある。まず、TEE を使ったセキュリティの観点では、Intel SGX, ARM TrustZone といったところは代表的であるが、実際、Ultra HD Blu ray Disc を再生するための DRM として Intel SGX が 採用されているという実用化の動きも出ている。2019 年に Linux Foundation が Confidential Computing Consortium を設立して通信データ保護、保存データ保護から使用中データ保護というコンセプトで、TEE を用いてオンプレ、パブリッククラウド、エッジ等の分散計算環境をセキュアに実行するというものを目指したプロジェクトが動き出している。Microsoft の Azure では Confidential Computing ができるソフトも開発されている。下のイメージ図では安全な箱、電子投票で実現したかったような箱がネットワークでつながって、安全な箱と安全なネットワークでしか秘密情報、プライバシー情報をやりとりせず、というものを目指したものである。一方で、TEE や Confidential Computing のセキュリティという観点では、安全な箱へのさまざまな攻撃が考え得る。サイドチャネル攻撃、フォールト攻撃、プロービング攻撃などのさまざまな攻撃があり、攻撃事例が研究報告されている。また TEE で暗号・認証プロトコルが動いていて、脆弱性がないかという安全性の観点もある。さらに Confidential Computing というネットワークが加わった時の暗号･認証プロトコルの安全性といった観点もある。こういったところをクリアしていくのが今後の課題ではないかと考えている。

耐量子暗号技術：量子コンピューターでも解読が困難な暗号技術・アルゴリズムで、暗号化技術ということで、共通鍵暗号や、公開鍵暗号をイメージされがちであるが、電子署名、ハッシュ関数、MAC 、鍵交換、セキュア通信など、さまざまな暗号技術の総称と捉え

ている。用途としては、従来の暗号技術同様、通信路の秘匿、保存 データのセキュリティ、データの完全性・真正性保証、アクセス制御（認証・認可）等に利用できると思っている。量子暗号や量子鍵配送は、通信路を秘匿する手段として研究が進んでいる。研究が進んだ背景としては欧米では耐量子暗号技術の標準化や試験利用の動きが活発化してきているが、まず 90 年代に公開鍵暗号や電子署名に対する、量子コンピューターを用いた攻撃アルゴリズムが発見され、当時は量子コンピューターの実現がまだ先であろうと思われていたが、近年では Google や IBM が出してきたところで、危機感が出てきている。実際には RSA 暗号等を解読できる量子コンピューターはあと 10 年はかかるだろうと言われているが、15 ～ 20 年先といったところでは、どうなるかわからないというところもある。そういったリスクを考えると、通信やデータ保存において、長期に渡って秘匿性の保持や真正性保証が必要なデータの分野・アプリケーションから普及が進むと考えられる。ナショナルセキュリティとか医療分野、金融分野から耐量子暗号技術は普及すると考えられている。

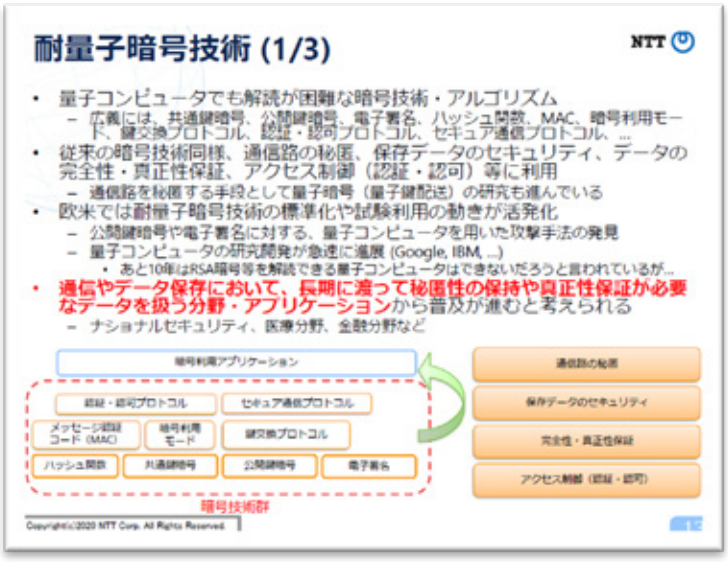


図 3. 4-11　耐量子暗号技術（1）

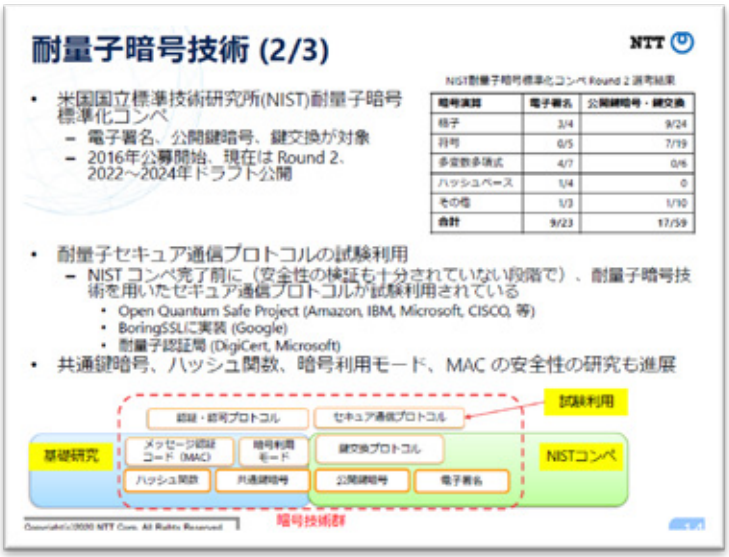


図 3. 4-12　耐量子暗号技術（2）

耐量子暗号が欧米では検討が進んでいるが、いま、NIST の方で標準化コンペが、電子署名、公開鍵暗号、鍵交換を対象に行われているところ。いまラウンド２で、もうすぐ３に進む。３～４年後にはドラフト公開と言われている。耐量子暗号は非常に難解な暗号演算、格子暗号や符号、エースの方法、いろんな方法が残っているという状況である。一方でセキュア通信プロトコルの耐量子化というのは米国で試験利用がなされている。まだ NIST コンペも完了しておらず、どの方式が安全かも十分検証されていないが、データは今も不正に取られていて 20 年後に解読されては困るというデータに関してはいまから対策を打たなければならないのでここは先に進んでいると思われる。実際にいろいろなプロジェクトが出てきているところ。共通鍵暗号はただちに量子コン

ピューターに解読されるというものではなく、大丈夫ではないかと言われているところだが、安全性の研究も進展しているところ。イメージとしては下の図の右の NIST コンペで行われている領域、左側の基礎研究の領域、そして先行して試験利用されているセキュア通信プロトコル、といったフェーズになる。

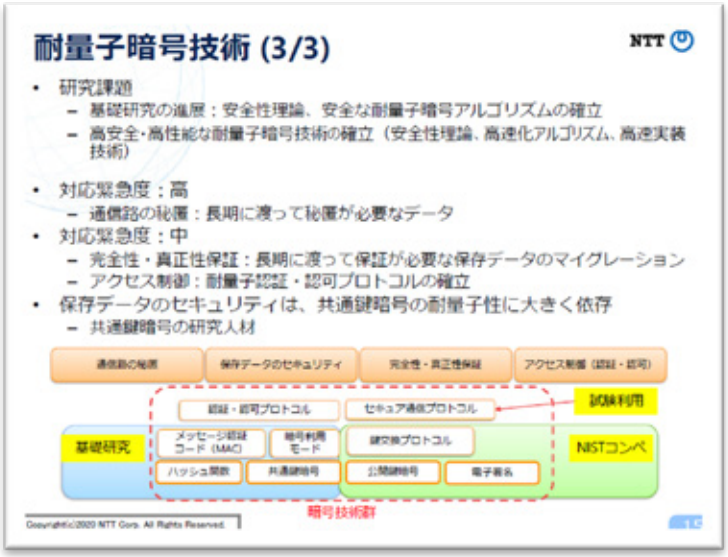


図 3.4-13　耐量子暗号技術（3）

われわれが耐量子暗号の研究課題としてとらえているのは、まずは基礎研究の進展である。量子暗号のコンペで残っているものがアタックされたとなると、一気に使えなくなるので、そこは安全性理論はしっかり確立されないとならない。安全なアルゴリズムを確立する。高安全・高性能な耐量子暗号技術の確立というステップを踏んでやるべきと考えていて、特に、通信データの取得というところについては今対策がなされているところなので緊急度は高いと考えている。そして電子署名のような長期保存しているデータは、量子コンピューターの導入によって一気に真正性が保証されなくなるという問題が出てくるので、そういう台頭の前にしっかりマイグレーション等も完了させる必要があるというところで、そちらも対応が必要と考えている。また保存データのセキュリティに関しては、基本は共通鍵暗号で暗号化されているものが多いと思うが、実際の安全性の基礎研究がなされているが、本当に安全かどうか、安全でなかったときにどう対策すればよいか、しっかり理論的なところから研究できる人材が必要と考えていて、現在は共通鍵暗号の研究人材はまだ多くないという認識であり、このあたりが課題になってくると考えている。

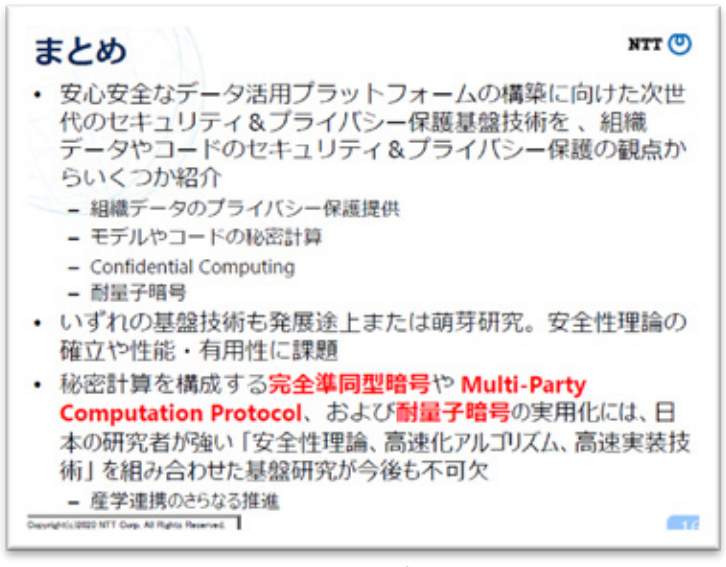


図 3.4-14　まとめ

まとめ：組織データやコードのセキュリティという観点で４つの技術を紹介した。秘密計算を構成する完全準同型暗号や Multi Party Computation Protocol、および耐量子暗号といった技術は「安全性、理論、高速化アルゴリズム、高速実装技術」を組み合わせた基盤研究が必要と考えていて、産学

連携のさらなる推進が必要と考えている。

【質疑応答】　（Q：質問、C：コメント、A：回答）

Q：クラスタリングができれば、不良の原因特定ができるという前提か。

A：シンプルな分析手法からということで、k-means 法のクラスタリングにおいて疑似データでも元のデータとあまり変わらないということを試験評価している段階。原因を完全に特定できるというものではない。

Q：スライドでは一般的なことが言えているような整理だったような気がするが、直感的には不良の解析は生データに近いものを相当みないとダメだというイメージでいるが。

A：そのあたりは精度と秘匿性のトレードオフの関係があるので、どれだけ元のデータに強いノイズをいれても結果が変わらないかということを実験的に消化している。まだ確立された技術ができているわけではない。まだまだ研究段階である。

Q：「秘密計算」の実用上の課題とは、理論的に安全性を保証するということか。それとも理論のみならず、実用上の課題を加味しながら安全性を担保するということか。

A：こちらの安全性保証は基礎的、理論的な側面の話になる。例えば、準同型暗号のアルゴリズムをいじって高速に処理できるというときに、トレードオフとしてセキュリティのレベルが下がる場合なども出てくる。セキュリティレベルを一定以上に保ちつつ、どれだけ高速なアルゴリズムができるか、あるいは、高速実装に関するセキュリティパラメータがあり、どのパラメータが最適かということを探索している。かなり基礎的な安全性理論の話になってくる。

Q：高橋さんも含めての質問になるかもしれない。半導体のデータは興味深い。あれを個人情報のプライバシーの対象として挙げておられるのは何のことかよく分からないが。

A：個人データではなく、組織のデータの機密保持、プライバシーという言い方をしている。

Q：個人情報保護空間でなく、対象となる法律が不競法になるので、違う論理になるのではないかと思うが、あえてなんでこれを出すのか。どうしてこの例でこの議論をしているのかがよく分からない。関連があるのか。

A：プライバシー保護というところ？　組織のデータの機密を守りながらそのデータは重要な分析ができるという意味合いで書いた。

Q：高橋さん、これは対象とする法律が全然違うので、今後議論するときに、コンフュージングな事例で議論すると何を議論しているか分からなくなると思って今の指摘をしたのはいいとして、全体感からお伺いしたいが、匿名化する機関に対して、データを渡す。データを渡すときは非匿名情報、というのが今の基盤法だと思うが、そのときは何ひとつ、もらった側は丸ごとみている。その代わり、国家の認証をかけると。そのときにこういうプライバシープレザービングでどうこうするっていう話は出てきて

ないような気がする、一方で今回の濃厚接触のアプリの場合は、アップルはそもそも個人情報を取らない。つまり収集のところを抑え込んでるという話になっていて、問題の設定をどういう風に整理するかということは結構重要だと思うが。世の中的にはどうなっているのでしょうか。前者ならば委託でやれば法律的にも守られているのでそれでよいのではないか。

A：基盤法というのは次世代医療基盤法というやつで、先生は理解しているが、念のため説明すると、次世代医療基盤法で認定された事業者は病院から生データをもらうことができる。そのあと匿名化して二次流通させることが認められている。それに関するご質問で、まず次世代医療基盤法で認定された事業者は莫大な生データに対する権限というか見る能力をあたえられているというのは客観的な事実である。正直、秘密計算みたいなことを考えていた人間にとってはあればポコッと穴を開けられてしまった感じで正直がっかりしたところではある。医療の特性を鑑みてあのような制度が作られたと聞いている。その考えに確信があるとか支持するわけではない。そこに平文で集めてもいいという考え方は、次世代医療基盤法からさらに広がっていくとは個人的にはとても思えないし、法律の専門家も他に広がることはないと見ている。ココアのアプリケーションの話だとご指摘のとおり、個人情報はとらないということで、引き換えに位置情報が使えなくなるとか、いろいろなことがアプリケーションの能力としてなくなったので、アップルグーグルの COCOA の枠組みはそれなりに個人情報が保護されているというのは正しいと思うので、個人情報をとらなくて何かできるという情報空間の判断というのを了解していく必要がある。これから複雑なことを皆やりたいはずなので、今回のものは限界があると思う。まとめると現在の規制の話でいうと同意をとってというところに戻ってきてしまう。

Q：一回整理をしてみた方がよいのではないか。個々の病院が生のデータを持っていて、そこからほんわかと秘匿計算をしながら情報を使いながらモデルを作ることはわが国はやらなかった。なぜそうなったかわからないと言っても、補助法の法律家が関わったのは厳然とした事実なので、それを抜きにしてわが国で通るわけがない。それでは議論が成り立たないので、どういうシステムを目指すのかという前提を整理した方がよいのではないか。理論的フレームワークとして突き詰めたいという気持ちは分からなくはないが、一方でわが国が基盤法という大胆な法律を作って、高精度な解析ができる、強烈なパニッシュメントを与える、ということで、一つのソリューションを与えている､台湾がやったことと同じようなもの。だとすると何も悪いものではなくて、今回グーグルとアップルで、グーグルは明らかに根こそぎデータを持っている。アップルは絶対に持たない。フィロソフィーみたいな問題で、一回、どこを原点にしながら議論するのかというのは、論文がこういうスタイルでたくさん書かれているからというのでやるのも一つの手だが、もう一回原点に戻ってもいいのかなと個人的には気にしているというのが一点。もう一つ、さっきの問題を聞きたかった。澤田さんは IOWN を興したいということでこれは NTT からみると昔の FTTH と同じぐらい大きな話で、すごく野心的というか、非常にフューチャリスティックで、あの時 FTTH をやったから日本がいまもっているようなものなので、それ自身はすばらしいことだと思う。だけど後半の話とどうつながるのかがよく分からなくて、光にしたから、その

部品をファーウエイから買ってもいい、という話がいったいどういうロジックで成り立つのかがよく分からない。後半のいくつかのアプリの図から入るというのは、ああいうチャラチャラした図はいまでも描けるような気がしていて、基盤となる技術というところで、光にするからプロテクトできるというシナリオは、今日のご説明を聞いているとよく分からなかった。

A：まずデバイスからちゃんと作ろうという決意でいる。それがロードマップ的にはかなり後になっていてギャップができている。デバイスの方でちゃんとしたものを作っていくということはいまホワイトペーパーの中にきちんと書かれていないので、ご指摘はもっともだと思う。

Q：そこはNTTとしてキャリアなので、他のところはオペレーションしているわけではないから、谷崎さんのところのほうが背負っているものが大きい。NTTは自分のところは守っているという言い方だったが、これも最近はブレークされてくる中で、どうやって光デバイスを守るのかという話がないと、ちょっと話が通じないなと思うので、そこを丁寧にまとめるというか、ここに日本の50人の先生方が集まっているのは、次に日本がどうやって固めるのかという根幹となる技術を攻める場所を議論しているつもりだと理解しているが、JSTの支援をいただきながら、そこをもう少しお話しいただける機会があるとよい。ちなみに大航海ではプライバシーを端でなく、相当真ん中に置いているので、あの発言はなされないようにしてください。

A：はい。

3.5 研究成果の活用（4）Future of Computing　計算の未来

福田剛志（日本アイ・ビー・エム株式会社）

本日は「研究成果の活用」というお題を頂いたので、まずは本日言いたいことを 1 ページにまとめてみた。本質的に重要なことは、ゲームチェンジを起こすこと、あるいは他者が起こしたゲームチェンジをつかまえるということと認識している。ここでゲームチェンジとは、既存のルールが壊れて、無くなり、新しいルールに基づいてゲームが始まるということである。IT の世界に限らず、研究の成果がゲームチェンジを起こし、新しい企業や産業が生まれたりすることが起こってきたと思う。IT の世界で最近起きたゲームチェンジが、ニューラルネットワークやディープラーニングの有効性の発見だと思う。一時は忌み嫌われていた「AI」という言葉も、社会にあっという間に浸透した。結果として多くの企業が収益を上げている。ディープラーニングの有効性が発見されたのが 2011 年頃だったと思うが、当時の NVIDIA の株価は 9 $ しかなかった。GPU をユーザプログラムから簡単に利用できるようなライブラリを提供したことにより、NVIDIA が事実上唯一のプラットフォームの計算基盤となり、現在の株価は 380 ドル近くになっており、10 年間で 40 倍近くになっている。ディープラーニングのブームは NVIDIA が起こしたものではないが、彼らはゲームチェンジの波に乗って大きく発展した。

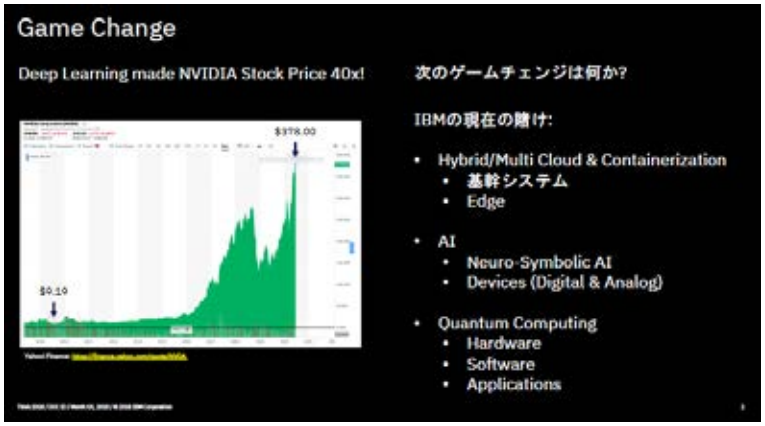


図 3.5-1 Game Change

次のゲームチェンジは何かが前もって分かれば簡単であるが、分からないからこそ多くの人々がいろいろな仮説に基づいて研究したり投資したりしている。現在、IBM がどのような賭けに出ているかをお話しする。1 点目は、Hybrid Cloud、Multi Cloud、あるいはコンテナ化と言われているエリアである。現在、IT システムのクラウド化が非常に急速に進んでいる。今や銀行のシステムのようなハイセキュリティ、ミッションクリティカルなワークロードもクラウド化してしまうという動きがある。今後、クラウドがコモディティ化していくことを考えると、クラウドのさらに上のレイヤーが主戦場になると考えられる。企業が使うような基幹システムや情報システム、あるいはネットワークのエッジにおいても起こってくると考えている。2 点目は、AI の進歩の中から生まれてくると思う。後ほど具体的にお話しする。3 点目は量子コンピューターで、ハードウエア、ソフトウエア、アプリケーションであるが、後ほど具体的にお話しする。これらはあくまで私が恣意的に選んだもので、他にも IBM ではセキュリティやプライバシーや耐量子暗号、完全準同型暗号などの研究も盛んに行っているが、私が得意とする分野ではないため割愛する。

本日は計算の未来、future of Computing というストーリーラインでお話をする。計算の

未来は bits と neurons と qubits から成ると考えている。bits とは従来のコンピューターのことである。ムーアの法則が 1965 年頃に発表され、それ以降、40 年から 50 年にわたって続いた結果、bits の値段が大幅に安くなり、現在のデジタルワールドが出来上がった。ムーアの法則に従って一気に物事が進んできた中で、最もハイエンドな事例が昨年発表された IBM のメインフレームである。皆さんあまりなじみがないかもしれないが、z15 というものである。１日でウェブトランザクションだと１兆回実行できる。安全性、可用性が Seven 9（99.99999%）の可用性を持っているというシステムである。

これと真逆の方向性として、砂つぶサイズのコンピューター、小さいコンピューターも開発している。「塩つぶ」や「塩つぶよりも小さい」と呼んだりもするが、塩は半導体の敵であるため「砂つぶ」と呼ばせていただきたい。「砂つぶ」の中に、CPU、メモリ、バッテリー、無線通信など普通のコンピューターが持っている全ての機能が集まっている。このような小さなコンピューターにセンサーをつけ、中に暗号キーを入れ、タンパープルーフにする。基本的には配線ができないため、いろいろな商品に埋め込むことで、クリプトアンカーとして実際の世界とデジタル世界をセキュアに結びつけることができる。このように小さくすることには意味があるが、元の値段は大きさにおよそ比例するため、コンピューターも小さく作れば安くなり、どこにでも張りつけることができるようになると考えている。例えば、ブロックチェーンと組み合わせることで、本物と偽物を区別できるような世界を作りたいと考えている。

図 3.5-2 砂つぶサイズのコンピューター
（図中の出典：https://www.youtube.com/watch?v=6yqfYER2YxA）
2020年11月時点

次は neurons についてお話しする。neurons は AI のことである。スペインの生物学者のサンティアゴ・ラモン・イ・カハールという人は、脳がニューラルネットワークであるということを発見した人である。1906 年頃、ノーベル賞を受賞している。

IBM の最近の成果をご紹介する。Debater という人間と討論することのできるコンピューターを開発した。「Jeopardy!」というクイズショーでチャンピオンに勝つというチャレンジをしたのが 2011 年であり、翌年から開発を始めたものである。ディベートというのは、４分、４分、２分というスピーチを２人の対

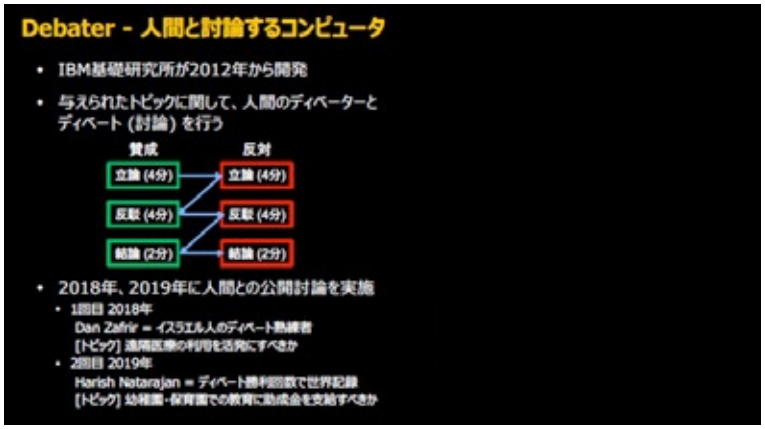


図 3.5-3 Debater 人間と討論するコンピューター

戦者が交互に行って、あるトピック、与えられたトピックに対する賛成、反対の立場で２人が討論をしまして、その優劣を競うという競技である。一昨年と昨年に公開デモを行った。ニューラルネットワークと、それからシンボリックな AI、少し前の世代の AI を組み合わせて動作するニューロシンボリックな AI の事例である。

ニューラルネットワークの AI でできる仕事はある程度限られていることがだんだんわかってきた。従来の普通のコンピューターのプログラムで解きにくい問題がディープランニングで解けるということが分かったが、逆にディープラーニングのシステムでは解きにくい問題もたくさんある。このニューロシンボリックなシステムは専用マシンだが、逆に汎用的に作る、性能良く作るような理論や、それに基づくソフトウエアが出来たらゲームチェンジになると考えている。

先日の ISC で今年前半のトップ 500 が発表され、「富岳」が世界一になった。その直前まで世界一だったのが、オークリッジナショナルラボにいる IBM の「Summit」である。このマシンは 9,000 個の CPU と 2 万 7,000 個の GPU から成るマシンであり、149 ペタ FLOPS である。消費電力が 15 MWと、およそ 7,000 世帯分の消費電力である。「富岳」の性能を調べたところ、415 ペタ FLOPS で 28.3 MWである。ざっくりと「富岳」と「Summit」を比較すると、演算能力で３倍弱、消費電力で２倍弱となるため、消費電力当たりの性能で 1.5 倍くらいである。ここで言いたいことは、消費電力の悪さである。28 MWというのは、１つの町の消費電力全体ぐらいである。

少し古いスライドであるが、「Jeopardy!」のチャレンジのときに IBM が使ったコンピューターのスペックと、それから人間の脳の比較を行った。そのときのマシンの消費電力が定格で 200 ｋWだった。これに対し、人間の脳は 20 W程度しか使わないと言われている。１万倍の差がある。本日のテーマは「システムソフトウエア」であるが、ソフトウエアではこの差は埋まらないと思う。

われわれは、アーキテクチャーからデバイスやチップ、材料に至るまで全ての技術を見直し、電力消費の削減に努めている。2018 年に発表した RaPiD AI Accelerator は、二次元のシストリックアレイになっており、精度を色々と調節できる。16bit のフローティングポイントを使った場合、1.5TFLOPS の計算能力がある。Google は TPU があり、先日、Green500 の１番となった Preferred Networks の MN-3 も専用の CMOS チップを作るということで、同じような流れだと思う。True North は、時期的には少し前になるが、人間の脳と同じような Spike である。短い電圧のパルスを使って信号を伝達するものである。2016 年に発表された True North のチップは、１チップ当たり 100 万個のニューロンがあり、シナプスが２億 5,000 万あり、普通の動画からのリアルタイムのオブジェクト認識をさせると

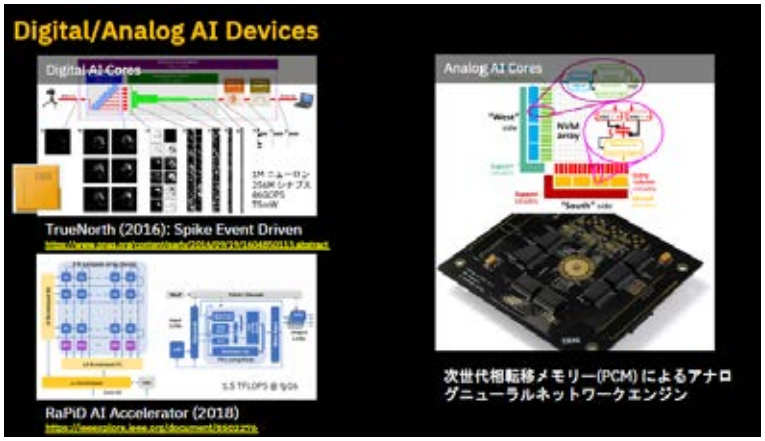


図 3.5-4 Digital/Analog AI Devices
(図中の出典：https://www.research.ibm.com/artificial-intelligence/ai-hardware-center/)
2020年11月時点

いうタスクを行いながら電力消費を測定して、75 mWであった。75 Wじゃなくて 75 mW である。クロックで駆動されていないので、イベントドリブンで動くため、非常に低い消費電力が実現できる。Phase-change memory というのを使ったアナログニューラルネットワークのチップもある。簡単に言うと、オームの法則を使って掛け算をして、それからキルヒホッフの法則を使って足し算を行う。ニューラルネットワークのセキュア演算をその物理法則を使ってやってしまうものである。Phase-change memory に抵抗の逆数のコンダクタンスを記憶させて使う。これでもって、現在の GPU の消費電力効率を 3 桁ぐらい良くすることを目指している。

GPU のケースでは、NVIDIA が CUDA といったライブラリを提供し、TensorFlow や PyTorch などの AI フレームワークがサポートすることで事実上のデファクトスタンダードとなった。河原林先生から、私の研究所がプログラム言語において、あるいは実績があるということでこのワークショップに呼んで頂いたと伺った。私自身が参加していたわけではないが、そのチームは現在、AI デバイスを有効に使うための基盤ソフトウエアを作成する研究を行っている。

IBM が作っている量子コンピューターのプロトタイプについてご説明する。量子コンピューターの素子は 15mK で動作する超伝導素子になっている。絶対零度から 0.015℃だけ高い温度で動作する。量子コンピューターは古典コンピューターから制御される。古典コンピューターから出てくるデジタルシグナルが量子コンピューターに送られると、内部でマイクロウエーブのアナログシグナルに代わり、冷凍機を通じ、半導体のチップに届けられる。その中で反応というか、物理現象が起こり、その結果もマイクロウエーブで読みだされ、冷凍機を通って、量子コンピューターから出るときにはデジタルシグナルとなり、古典コンピューターにつながり、動作する。

2016 年から IBM では、動作する量子コンピューターをネットワーク、クラウドに接続し、一般に公開している。皆さんも使うことができるので、御興味がある方は「IBM Q Experience」で検索いただきたい。現在、14 台の量子コンピューターがクラウドに接続されて稼働している。ちなみに、量子コンピューターの消費電力は、フェムトワットレベルです。量子の計算は基本的にエラスティックであるため、電力を基本的には消費しない。ただ、希釈冷凍機を動かすのに数ｋＷぐらいかかる。これを動かすには当然ソフトウエアが非常に重要だと思っている。

IBM では、Qiskit（キスキット）というオープンソースのプロジェクトで、量子コンピューターの基本ソフトウエアの研究を進めている。一番下のレイヤーに、IBM のＱの実機や Qiskit Aer という state vector simulator、Third Party の実機やシミュレーターを持ってくることも出来る。このレイヤーは、その上のレイヤーにある Qiskit Terra で quantum circuits を翻訳し、実機で実行できるような形式に変換し、実機あるいはシミュレーターで実行される。量子コンピューターの実機は非常に貴重な資源であるため、シミュレーターで実験することは非常に重要である。Qiskit Aer という state vector simulator は、並列計算と GPU を活用することで、世界で最高速の量子コンピューターのシミュレーターとなっている。ＮＩＳＱと呼ばれるノイズを低減したり、どういうノイズがあるのかを理解したりするためのライブラリが Qiskit Ignis である。Qiskit Aqua は、Quantum アルゴリズムを色々とまとめたライブラリである。化学や金融や最適化などのハイレベルな

問題を、Qiskit Aqua が提供しているアルゴリズムと組み合わせることで、比較的高水準の記述で描くことを目指したライブラリである。例えば、最適化のアルゴリズムであれば、量子コンピューターやイジングマシン、例えば古典的なソルバー、CPLEX のようなソルバーを使い、それをバックエンドとして切り替えながら実行するようなことが、この Qiskit Aqua やこの Qiskit の仕組みを使うとできる。

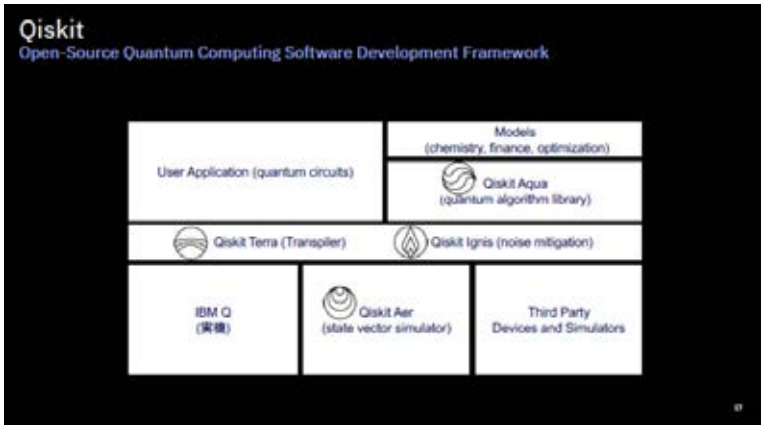


図 3.5-5 Qiskit
（図中の出典：https://qiskit.org）
2020年11月時点

皆さんは、量子コンピューターはまだまだ先のこととお考えかもしれないが、どれぐらいのスピードで量子コンピューターの性能が向上しているのかをプロットしてお示しする。2017 年から毎年量子コンピューターの性能を Quantum volume という指標で測っている。ノイズがあるため、ノイズの大きさや qubits 数など、いろいろなことを考慮し、量子ビジット間の接続関係などを考慮した上で公平に測れるように考えられた指標である。これが過去 4 年間で毎年倍々に増えている。ムーアの法則が発表された 1965 年のとき、まさに 4 つしかデータポイントはなかったのだが、これと同じような状況にあると言える。ムーアの法則は 40 年、50 年続いたと言われているが、この量子コンピューターのエクスポネンシャルが 40 年とは言わず、今後 10 年ぐらい続くと、今よりも 1,000 倍の処理能力となる。今の普通のコンピューターでは解くことができない問題を量子コンピューターが確実に解くことができるという時代がやってくる。これがこの通り進むかどうかというのは、これからのお楽しみである。

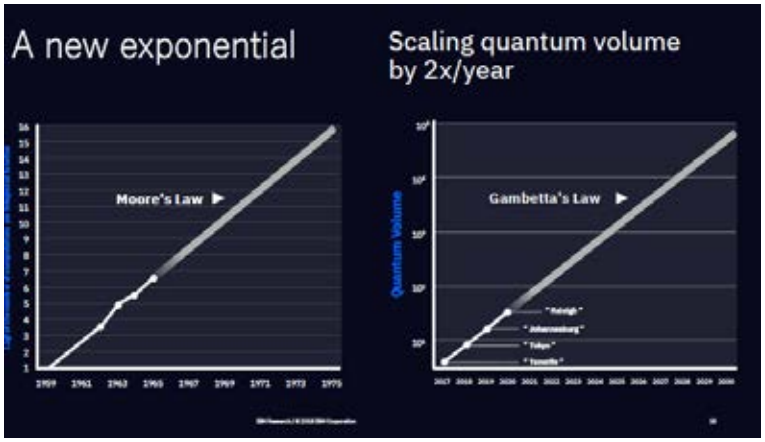


図 3.5-6 A new exponential

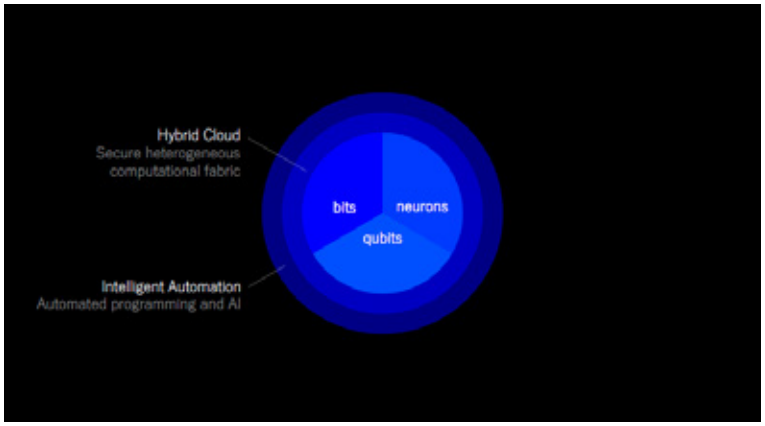


図 3.5-7　まとめ

ここまでをまとめると、bits と neuron と qubits を組み合

わせた計算の未来は非常に複雑である。全く違う種類の計算を組み合わせることとなる。ユーザーはどのようにこの計算を使っていくかを考える。複雑さが増したこの計算のファブリックをくるむ形として、クラウドのさらに進化した、さまざまなクラウドが混じり合っている世界として「Hybrid Cloud」といったものを考えており、そこがヘテロジニアスな computational にラップしていくと考えている。ソフトウエアの役割として考えられるのは、例えばユーザーのワークロード、こんなことがしたいといったものがあったら、それを bits のシステム、neurons のシステム、qubits のシステムに分割して、サブタスクに分割して、それを実行して、さらに結果を総合して返すというようなことをやる必要があるというふうに考えられる。さらに、「Hybrid Cloud」の運用管理を考えたときに、目覚ましく発展する AI によって自動化され、AIOps を発展した形になっていくと思う。オペレーションだけでなく、現在は人力にほとんど頼っているソフトウエア開発も可能なところから自動化が進んでくると考えられる。既に IBM の製品である Watson「AutoAI」は、自動的にニューラルネットワークのモデルのデザインまでコンピューターが自動に行うことができる。

最後に、近未来の絵をお示しする。左側がコンシューマ系のシステム、右側がエンタープライズ、企業向けのシステムを何となく念頭に置いている。水が循環するように、データがこの全体をぐるぐる循環する。データは本当に循環するわけではないが、巡っている。注目いただきたい点は、エッジデバイスなどにも AI が搭載され、インテリジェントなことを何かやっていると考えている。さらに各所にある雲の絵のようなものが、コンテナ化された、Docker で動くようなコンピュータリソースである。エッジ含め、コンテナ化が進み、エンタープライズのシステムと基本的には同じような仕組みでシームレスに運用管理ができるようになる世界を想定している。エッジが全てではないにしろ、コンテナ化されていくということは、ソフトウエアのエコシステムに対して非常に大きな影響があると考えている。従来、こういったシステムは非常にヘテロジニアスであったため、ばらばらに開発されてきたが、それが共通化され自由化され、民主化されることが起こることを意味しており、大きなゲームチェンジが起こる可能性があると思っている。

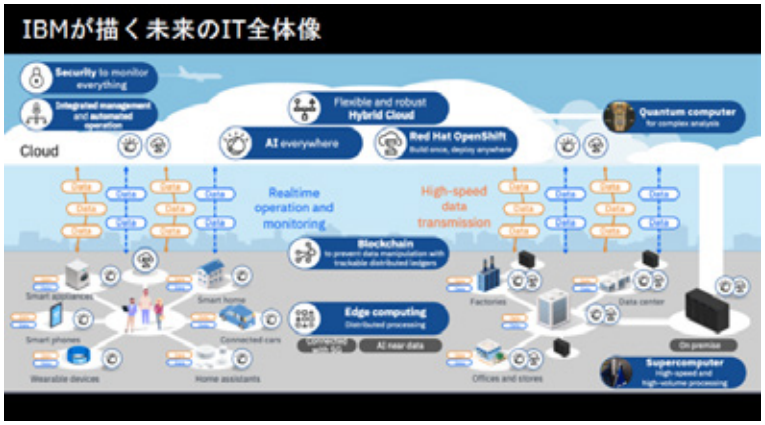


図 3.5-8 IBM が描く未来の IT 全体像

以上、IBM の活動を幾つか御紹介する中で、本日のワークショップと関係あると思ったことについて私見を述べさせていただいた。

【質疑応答】　（Q：質問、C：コメント、A：回答）

Q：チャットで質問が２つほど来ている。まず、河原林先生からの Hybrid/Multi Cloud & containerization について。

A：現在のチャレンジである。新しいインフラストラクチャーになったが、これから新しく作るものを Hybrid Cloud、Multi Cloud で動かすことは、ある意味スクラッチから作ればよいので、企業にとっての難しさというのはあまりない。ただ、新しいシステムが動くだけでなく、既存の古くなってきたシステムを Multi Cloud、Hybrid Cloud で動かし、コンテナの上でマイクロサービス化していくことは、アーキテクチャーから見直して作り直さねばならず、非常に大きなプロジェクトとなる。いかに AI の力を使って自動的に進めていくか。全て自動は無理だとしても、ツールとしてどうやってサポートしていくか。トラディショナルなソフトエンジニアリングの課題ではあるが、現代の環境の中で起きてきていることの１つである。これがうまくできると、いわゆる過去のしがらみにとらわれない IT システムの変革というのが比較的容易になるため、最初はかなり苦しいが、そういったツールなどをうまく使いながらモダナイズしていくということが求められていると思う。

Q：続いて井上先生の御質問について。

A：AI チップのソフトウエア基盤で肝となる、重要なことは何か、というご質問。既にディープニューラルネットワークが流行り出してから 7-8 年が経っている。この間にたくさんのフレームワークが生まれ、たくさん死んでいった。ある程度コミュニティというか、ソフトウエアを開発する人たちが出来上がってきた。その人たちをいかに新しいプラットフォームに引き込んでいくか、新しい AI チップに引き込んでいくかということになる。いろいろな戦略があり得るが、ユーザレベルから見ると、今ある上位のフレームワークである Tensorflow や PyTorch、その上の Keras などがそのまま使えることが望ましいのだと思う。しかし、新しいチップであるからこそ、何か新しいことを入れなければいけないため、それらのバランスである。基本的には既存の資産が有効活用できるようなソフトウエア基盤を作ろうと努力している。同時に、AI チップのアドバンテージが出るような、そういったソフトウェアスタックを作っていく必要がある。もう１つの質問の、neuron と qubits はどのような相乗効果が期待されているのか、あるいはヘテロで独立したものなのか、という点について。量子コンピューターもいろいろな研究の中で１つの固まりとして、AI に対する応用が考えられている qubits 空間に特徴ベクトルを射影し、その中で例えば分離を行い、また元に戻すようなことをすると分類精度が上がるような研究もあり、あるいは早くなるような研究もあり、そういった量子コンピューターを使った AI の応用が考えられている。今ある AI チップのアクセラレーションのほかに、量子コンピューターを使ったアクセラレーションや、速さだけでなく分類能力などでの貢献も期待されている。今後、neuron のシステムと qubits のシステムの組合せは生まれてくる可能性が十分あると思う。

Q：もう１点質問したい。True North に限ったとき、Spiking Neural Network と qubits というのはどういう関係になるのか。

A：良い質問だと思う。qubits は波である。量子の波である。それから、Spiking もある意味波が脳の中に伝わっていくところをモデル化していると言える。波を使ったコンピューティングだということで統一理論が考えられるのではないかと研究しておられる方がいる。もちろん、テンソル積の空間と、普通のベクトル積の空間で違うのだが、統一的に見て、学習するということはどういうことか、といったところにつながる可能性はあるかもしれない。

Q：量子コンピューターの話は、IBM さんも慶應と一緒になって広めるようなお話とか、日本では Q-LEAP が走るとか、そういったピーキーなものは放っておいても施策が立つ。今回、福田さんにお願いをさせていただきたいと思ったのは、IBM の基礎体力に対しての期待感がある。少し前にチューリング賞で、ヘネシーとパターソンが取ったが、われわれから見ると IBM の 801 のほうが RISC という観点では圧倒的に速く、RAID も IBM のほうが基幹特許をずっと前に取っている。最初に仰ったように、IBM は何か知らないけれどもゲームチェンジを見張っているというような感じではなく、昔は自分で考えておられた。われわれはＩＢＭを尊敬しながら見てきたのが正直なところである。例えば 99.99999％ぐらいの可用性を実現するといったときに、多分ハードだけではなく、非常に深いソフトウエアの検証を行ったり、バグが絶対出ないような工夫とか、そういうのをやっておられるのではないかと思い、そのようなシステムソフトウエアから、プログラミング言語から基盤的な IT 領域に対しての IBM の現状は何なのか、深いところをお伺いしてみたい。

A：難しいご質問である。メインフレームは非常に長い歴史があり、古く作られたソフトウエアの蓄積があり、今の信頼性が得られている。よくこの信頼性を作り込んだなというところを、私も同じ会社にいて感心する。ここにはとても新しいテクノロジーがたくさん入っている。例えば、耐量子暗号のようなものや、エスティカルエグゼキューションの仕組みなど、新しいのがどんどんメインフレームの中に入りつつ、リラックスマシンとして動くような、新しいものがたくさんありながら動いている。コンテナ化が流行っているが、そういう意味では IBM は 40 年前にある意味それを実現していた。秘密、秘訣のようなものが何なのかは分からない。

Q：そういった基礎的な研究をお続けになっておられるのか。レッドハットを買うようなところで生き延びているのか。IBM の外から見ると、ロメッティの時代は死んでいる気がする。もう少し上手にやれたのではと思う。そういう底流にある技術を育ててきたのではないかと感じられるが、そうでもないのか。

A：アルビンド・クリシュナという research 出身者がロメッティの後に CEO になったが、彼自身は Research 出身である。現在は Dario Gil という者であるが、その前の Research のディレクターが現在の CEO のアルビンド・クリシュナです。アルビンド・クリシュナは、研究者に好きにさせていた面があったが、新しいディレクターになって少し理詰めで、10 年先の研究をするのか、５年先の研究をするのか、３年先の研究をするのかということを割とポートフォリオをきれいに組んで研究していこう、という動きになっている。逆に言うと、従来は研究者の良識に任されており、すごく長い研究をしている、長期の足の長い研究をしている者もいれば、来年出る製品のた

めの、今これが必要だという機能を作っているような人間もおり、オートノマスというか、自律的に行われていたのをかなりインテンシブに、インテンショナルにそこのポートフォリオを組み直そうということをやっている。ロメッティの時代だったからショートタームにやってきたとか、そういうことは全然なく、自発的、自律的であり、これからは時間軸に対しても、それから大きな賭けも、小さな賭けもポートフォリオを組んでやっていくというような形で研究を進めている。

3.6 研究開発戦略センター（CRDS）の取組み

茂木 強（CRDS）

CRDS の取組みについて報告する。研究開発を例えば文部科学省の戦略目標に結びつけるためには、説得のストーリーを作成しないといけない。今回はその舞台裏をご紹介する。

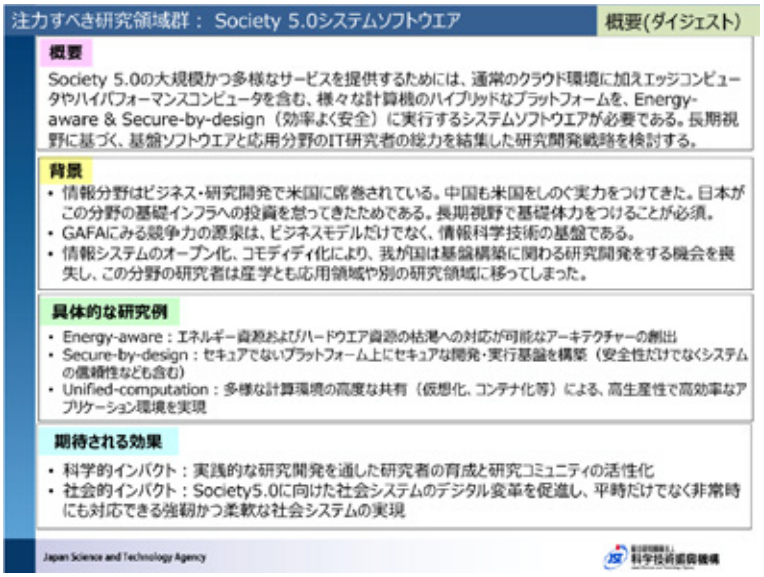


図 3.6-1 注力すべき研究領域群

提案のダイジェストを示す（図 3.6-1）。タイトルは「Society 5.0 システムソフトウエア」である。まず、概要として、Society 5.0 の大規模かつ多様なサービスを提供するためには、通常のクラウド環境に加えエッジコンピュータやハイパフォーマンスコンピュータを含む、さまざまな計算機のハイブリッドなプラットフォームを効率よく安全に実行するシステムソフトウエアが必要であり、そのために必要な研究開発戦略を検討すると書いた。

ついで、アメリカ、中国が席巻していて、日本はシステムの研究力が弱っている。それは、そこに投資してこなかったからである。したがって、それをやることが今必要であると背景で述べている。

具体的な研究例には３つ挙げている。１つ目エネルギー資源およびハードウエア資源の枯渇への対応が可能なアーキテクチャーの創出である。2 つ目は、今回のワークショップのテーマであるところのセキュアでないプラットフォーム上にセキュアな開発・実行基盤を構築、そして 3 番目に、多様な計算環境の高度な共有（仮想化、コンテナ化等）による、高生産性で高効率なアプリケーション環境の実現を挙げている。

その結果、期待される効果としては、科学的インパクトとして、研究者が養成されて研究コミュニティが活性化する。また、社会的インパクトしては、Society 5.0 に向けた社会システムのデジタル変革を促進し、平時だけでなく非常時にも対応できるような安心かつ安全で強靱な社会システムを実現する。大きくはこういうストーリーで説明しようと考えている。

次のスライド（図 3.6-2）では、システムソフトウエアの位置付けを示している。左にある Society 5.0 プラットフォームの図は内閣府の公開資料より借りてきた絵だが、ここではサービスシステム群の下に AI 等の基盤技術とデータベースが書き込まれている。しかし、実際には、それらが動くためには、OS にあたるシステムソフトウエアやハードウエアが必要である。右の図ではそれらを明示的に書き込んでいて、赤枠で囲った「Society 5.0 システムソフトウエア」が必要だと訴求している。

前回のワークショップの冒頭に江村会長と喜連川先生からお話があったように、文部科学省の情報委員会に向けて「情報学の次世代検討会」として三回にわたり、情報処理学会の各研究会の先生方を交えて、情報学のビッグサイエンス化、OSやミドルウエアといった基盤ソフトウエアの重要性を議論してきた（図 3.6-3）。今回のワークショップはそれを踏まえた取り組みという位置づけである。

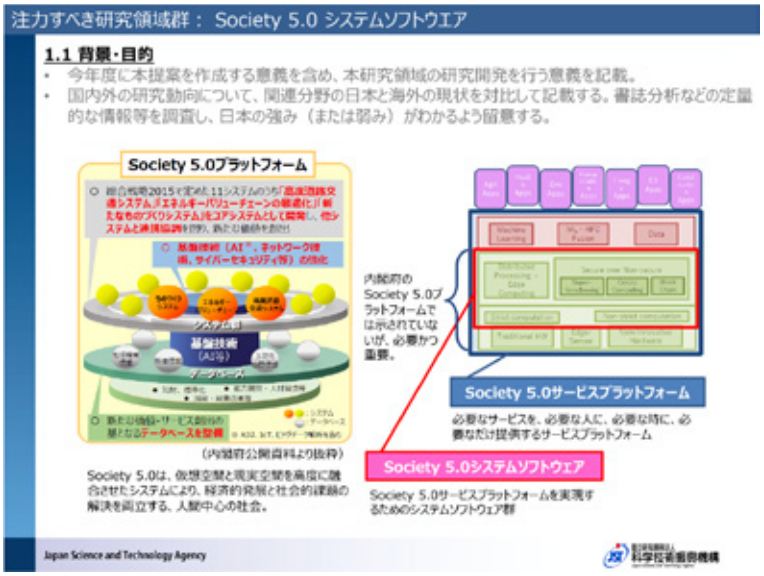


図 3.6-2 背景と目的

アーキテクチャーに関連した戦略目標として、H30 年度「革新的コンピューティング（Society 5.0 を支える革新的コンピューティング技術の創出）や H31 年度「次世代 IoT（次世代 IoT の戦略的活用を支える基盤技術）」があり、それぞれ、今回の Society システムソフトウエアのハードウエアやエッジコンピューティングに関わる部分を支える研究開発である。今回はこれらの言わば集大成版になるものと考えている。

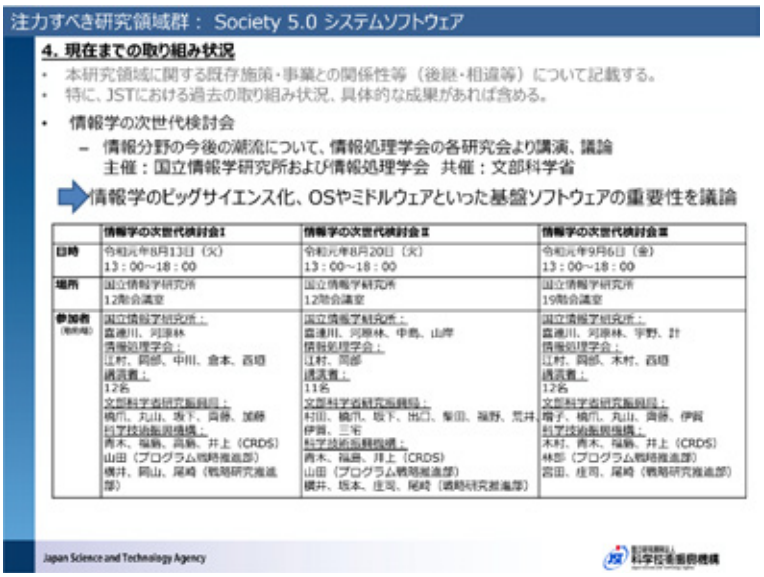


図 3.6-3 現在までの取り組み状況

次のスライドでは論文や特許の世界動向を示す（図 3.6-4）。ここでは、1980 年から、OS、言語、仮想マシンというキーワードで検索した結果を調べた。米国（青）が圧倒的に強いが、ここ 10 年来中国（赤）が猛追している状況である。以下、ドイツ、英国、フランス、日本、この第三位グループである。また、近年インドが新たなプレーヤーとして頭角を表してきた。

また、著者の所属機関では、トップ 20 までは米国 13、中国 3、日本は IBM と東大が入っている。圏外には、NTT、阪大、東工大、京大が入るという状況である。

最後になるが、今回のワークショップの位置付けと狙いをお伝えしたい。通常、戦略目標をターゲットにした活動は、例えば、私ども CRDS がテーマを選定したものを、有識者にインタビューしたり、ワークショップを開催したりして、戦略プロポーザルに仕上げて、文部科学省の担当課に提案して、その後、省内の策定プロセスに引き継いでゆく。その後、戦略目標になって、CREST やさきがけの公募段階で、皆さまに参加の機会が与えられる。

今回、徹底的に違うのは、最初の段階からご参加いただいている。皆さまがつくる戦略目標という形で、こういった場を設けて、研究者の皆さまの考え、知見、見解、思いを集約するというアプローチを取っている。これは私が知る限り初めての試みで、橋爪参事官が非常に強力に協力、推進していただけたおかげで実現できている。

研究者の皆さまへのお願いは、まず、研究コミュニティ内から一歩出て、隣とか、ちょっと離れた人々との議論を活発にしていただきたい。今日、このあとの時間でその第１弾をやっていただくことになる。次に、さらに広い分野の研究者を巻き込んで、大きなムーブメントにしていきたいのでご協力ください。今回のワークショップをそのきっかけにしたい。

「まずは」と書いてあるのは、手始めに今の思いを提案書につぎ込んでくださいという意味である。７月 13 日が締切りになっているが、ぜひ熱い思いのうちに書いていただきたい。多くの方からの提案をお待ちしている。

図 3.6-4 論文特許の世界動向

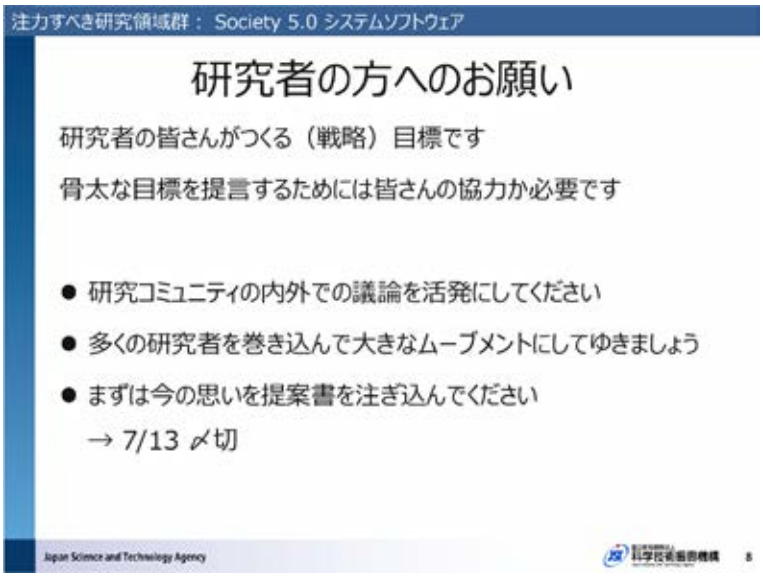


図 3.6-5 研究者の方へのお願い

3.7　全体討議

コーディネーター 河原林 健一（国立情報学研究所）

（1）ブレークアウトセッション

4 つのブレークアウトセッションに分かれて討議を行った。（詳細は割愛）

（2）各ブレークアウトセッションの報告

○岡部

全体的な話としては、一般の人に分かってもらうためには分かりやすいゴールが必要だ。「富岳」だと、例えば LINPACK で 1 位になるというような、分かりやすい数値目標があるが、われわれの S5Kernel は一体どういう評価尺度を出せばいいかという話がでた。

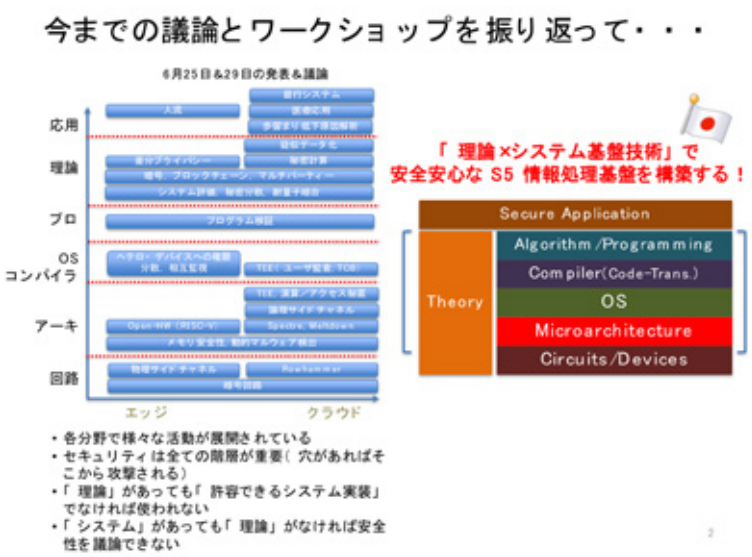
その中で、例えば今日の最初の講演にあったようなレガシーのシステムにも対応できるというゴールもあるという話も出た。

もう一つは、この S5Kernel がどこまでカバーするのかということで、例えば、下はハードウエアの耐タンパー性を突き詰めていくと、今考えているアーキテクチャーのレベルよりもっと下のデバイスのところまでいかないといけないし、あるいはセキュリティといってももうちょっと応用寄りの人工知能などにも関わっていかないといけないという話も出た。一応の整理としては、今考えているところの界面を S5Kernel として、もちろん、それより上の人とか下の人たちが、このプロジェクトが立ち上がったときに、専門性を持ってコミットすることについては大歓迎だけれども、どこかで線を引くとしたら、今のあたりだという話になった。

○井上（弘）

各分野でさまざまなセキュリティに対する活動が展開されているが、セキュリティは穴があったら駄目なので、各階層でちゃんと考えるのが重要である。ただ、これには理論が大事だが、理論があっても、コストや実行時間などがきちんと許容できるような実装になっていないと使われない。逆に、どんなにきちんとシステムを実装できたとしても、理論がないとどれだけ安全かという議論もできないので、理論も重要になってくる。

結局のところ、再認識したのは「理論×システム実装技術」によって安全･安心な情報処理基盤を作るということである。日本は理論は強いと考えており、そことアーキテクチャー研究者が組むことが重要であろ



う。例えば秘密計算を効率よくやるのにはアーキテクチャーとコンパイラと理論が必要になる。Open-TEE では、信頼できるプロセッサとして RISC-V ベースにすると、アーキテクチャーから OS、コンパイラ、プログラミングまでの連携が必要になる。権限分割 OS のところではアーキテクチャーと OS の連携が重要になる。アーキテクチャー分野における脆弱性検証、あるいはセキュリティ検証が非常にホットになっており、これをやろうとすると、当然理論も必要だし、プログラミングやハードウエアの設計レベルできちんとセキュリティを検証するということもやらないといけない。Dynamic Information Flow Tracking、つまり外から入ってきた入力が怪しいときに、それをきちんとトラッキングできるような技術においては、理論からいろいろ階層をまたがないといけない。

こういうのを見ながら議論した結果、大きく出たのは２つであった。

１つは、やはり理論が重要であり、理論と組めるというところが１つの大きな話であり、強みになるという意見があった。

もう１つは、セキュリティというのは１つの大きな旗になるということである。ただ、セキュリティにしても、プライバシーを守りつつ分析するというようなバランスがある。性能と安全性、エネルギーと安全とかだけではなく、いろいろなトレードオフがあるので、そういったことも考えながらセキュリティを旗にいろいろなバランスを考えるというのもよさそうである。

もう１つの大事な話は、企業の方から実際にベンチマークというのを出してもらえると、いろいろな問題設定がしやすいので、レイヤーをまたいでやるのであれば企業にも参加してもらいたいということである。ただし、企業がシステムをつくるときは、お客さまと一緒につくるので、そこは結構ハードルが高いかもしれないという話も出た。

○河野

S5Kernel がどういうふうに世の中の役に立つのかということを分かりやすく言わないといけない。銀行などの企業から見ると、サービスをつくるときにかなりのコストが実はセキュリティのためにかかっている。しかし、セキュリティは一般のユーザーから見ると、守られて当たり前なので、経営コストはかけにくい。そうすると、S5Kernel のような基盤がしっかり用意されるのであれば、企業から見るとセキュリティにかけていたコストを減らすことができる。それが国としての戦闘力になるという話があった。見えないところではあるが、そこをしっかり基盤として支えることで、一般の方から見えるアプリケーショ

ンが質の高いものになっていく。

また、そのためのイネーブリングなテクノロジーとしてシステムソフトウエアや暗号などの理論というのはビルディングブロックとしてお互いに重要なのだということに気づいた。

ここに集まった研究者にはさまざまな要素技術があり、企業の要求にこたえられると思う。

○松浦

２つの話題について議論した。基盤技術を主役にすると、非常に筋のいい研究が出てきたときに、他の技術との時間軸、スケールが異なる場合が得てしてある。どの研究分野でも起こりえるが、相対的に基盤寄りにその傾向が強いように思う。今回の活動の結果としてファンディングがあり、プロジェクトがあるとすると、そこをどうそろえていくのか？参画する先生方が安心して存分に腕を振るえるようにそろえる必要があるだろう。

それからもう一点は、セキュリティ分野だと攻撃の研究として革新的なものが出てくると、それがゲームチェンジを起こすという場合が往々にしてある。その研究自体はゲームチェンジ後に世を席巻する技術の枠組みの中で自分がその始祖であるとアピールできるものにはならないが、そのゲームチェンジを起こす上では絶対的に必要であって、かつ、それができるプレーヤーが実際その後のコンストラクションするときでも相対的に非常に強いというものがある。だから、そこを育てないといけない。本当にエポックメーキングなものを育てていくのは難しい。狭い意味の暗号や情報セキュリティだけだと分かりやすいが、今回のようなシステムソフトウエアとして見たときに、後で日の目を見ないかもしれないが、押さえておかなければいけないもの、創造的破壊の中心になるような重要なピースとなるようなものを入れた旗にするのが大事だという意見があった。

（3）ブレークアウトセッションの報告のまとめ

○河原林

今回の話題の１つは､「理論」プラス「システムアーキテクチャー」というのがいいキーワードになりそうだということである｡理論寄りの人もいれば､システムアーキテクチャー寄りの方もいて､その組合せで仕事をするというのは両方にとって非常によいと思われる。例えば暗号にしても、本当の基礎の基礎というのは、ある意味ユダヤ人が強過ぎて手が出せないところではあるが、理論とシステムとかとなると少し話は変わってくる。そこらへんが狙い目になるのではないだろうか。

次に、企業からのエンドースメントあるいは企業が持っているシステムの問題点というのは、アカデミアにとっては非常によい問題提起である。いろいろな企業とのコラボをこの分野ではやっていかなければならない。

システムソフトウエアと基盤というのが、暗号などの研究者から見ると重要なプレーヤーであるという認識はあるかと思う。井上先生からは逆の立場のことも指摘され、システムから見ても理論というのは非常に重要であるとのことだ。井上先生の図は、もう少し肉づけをすれば、具体的な研究テーマとしてはよくなると思う。さらに、企業の話なども

含めていくとだんだんまとまっていくように思う。

あと松浦先生から時間スケールという話があったが、この分野というのは機械学習とかAIとは全然違う時間スケールで動いているので、一緒にするのはナンセンスだと思う。機械学習では、論文を年間10本、20本書く人は多くいるが、この分野では数年に1本というようなところなので、それはここでは共通認識としている。文科省の皆さま含めてぜひ共通認識としていきたい。

（4）総合討論

○茂木

最近CRDSに若山先生が着任され、数学の力というのをすごく強調され、われわれも少しずつ勉強している。数学のコミュニティとうまく連携できないか？

○河原林

それは時間をかけてやらなければいけないが、大きな戦力になると思う。しかし、数学のコミュニティが入ったからといって、ユダヤ人コミュニティといい勝負になるかというと、そうはならないが、先ほどのキーワード、「理論」プラス「システム」ということで言うと、理論に役立つというのは間違いない。

○江村

九州大学で産業数学というのをやっている。だから、そういう切り口の視点は結構大事である。数学界の中でも社会問題を解いていこうというように考えている方も結構いるので、時間はかかるかもしれないけれども、コミュニティとして新しいものをつくっていくというのはこの機会に少し検討いただくとよい。

○喜連川

東大にもある。数学との連携はむつかしいが、時代としては連携に進みつつある。JSTにも数学に関連するプログラムがあるはずだ。

○茂木

数学に関連するプログラムは幾つかある。数年に1つくらいはプロジェクトが立っている。

○喜連川

そういうところとも何かしらコラボレートしながらやるというのは非常にいいと思う。

○河原林

暗号などはCREST・さきがけには結構たくさんあった。ただ、純粋数学とはなかなか連携が難しい。

○縫田

私は今回、暗号の研究者枠で呼んでいただいていると思うが、自分はもともと数学畑で、JSTの数学方面のプログラムにもかかわっている。そういう経験から言うと、数学の人を巻き込むというのは割と見込みがあると個人的には思っている。ただ、数学の分野全体を巻き込むという話ではなく、数学の分野によっては難しいという話があったが、分野というよりは人依存だという印象がある。数学の中にもいろいろ分野があるが、それぞれの中で本当に狭い意味での数学にしか興味がないという人もいれば、ほかの世界にも興味を持つ人もいる。

数学を巻き込む、協力するという見込みはかなりあると思う。

○喜連川

今回コロナで明らかになったのは、科学技術の研究をしている人が役に立たなければならないが、それには結構リスクがあるということである。そのリスクを軽減するには、皆が語らいながら、一緒になりながら問題を解決しなければいけない。そういう風土を数学や数理モデルをやっておられる人たちと現場が作っていかなければならない。感覚的に言うと、災害が来て疫病が来ると、次に来るのは戦争だ。この戦争は物理戦争よりも、むしろサイバー戦争のほうが中核になる。理論とシステムがしっかりタッグを組むということが求められている。

○佐々木

今回の検討の大きな目的は、日本の基盤システムを強化することであるが、そこに日本のいわゆる技術層として低いレイヤーの人材をたくさん巻き込んでいきたい。結局、研究者が増えないと研究力は上がっていかない。アメリカに６、７年いたが、アメリカの大学院生の多くは企業で経験がある人が世界中からアプライしてくる。日本でも社会人ドクターを増やして、現場を知っている人がアカデミアに戻り、この領域の先生方と一緒に研究して何か作り、交流もして基礎力も高めていくというようなスキームを作ることができればよい。

○河原林

ありがとうございました。これで総合討論を終了する。

4. 提言書

今回のワークショップは、発表者以外も研究者の参加が各回とも 20 名と多く、時間の制約もあり、全員からご発表いただくことはできない。それでは、せっかく第一線の研究者にお集まりいただく機会を設けたのにもったいないということで、提案書という形で参加者に募ったところ、以下の方々から回答があった。

●提案書（A4 用紙 1 ～ 2 枚）

Society 5.0 システムソフトウエアとして取り組むべき研究テーマのタイトル、研究内容、社会的インパクトなど

#	分野	タイトル	提案者
1	アーキテクチャー	全体像とコンピュータ･アーキテクチャー	井上 弘士（九州大学）、 佐々木 広（東京工業大学）、 入江 英嗣（東京大学）、 塩谷 亮太（東京大学）、 小野 貴継（九州大学）
2	OS	IoT からクラウドまで総合的なセキュリティを実現するモダンなセキュア OS	品川 高廣（東京大学）
3	OS	信頼できる隔離実行 ---TEE から CertifiedTEE へ ---	須崎 有康（産業技術総合研究所）
4	OS	メモリ省電力化とセキュリティを両立する基盤技術の研究	穐山 空道（東京大学）
5	セキュリティ	システムソフトウェアレベルのトラスト基盤技術	松浦 幹太（東京大学）
6	セキュリティ	数理とソフトウェア科学を融合した機械学習のセキュリティ研究	矢内 直人（大阪大学）
7	セキュリティ	SECURITY PLATFORM FORWARD TO POSTQUANTUM CYPHER, TEE AND IOT	宮地充子（大阪大学）
8	プライバシー	データ管理に関する S5Kernel 提案書	吉川 正俊（京都大学）
9	プライバシー	異分野連携による Society5.0 セキュリティ検証技術の研究	川本 裕輔（産業技術総合研究所）
10	言語	広義の「プログラミング言語」にもとづく Society5.0 時代のシステム検証研究	住井 英二郎（東北大学）、 小林 直樹（東京大学）、 田浦 健次朗（東京大学）、 五十嵐 淳（京都大学）

5．付録

付録1．ワークショップ開催概要・プログラム

ワークショップ（第1回）

【テーマ】研究内容に関する議論

【日　時】2020年6月24日（水）15:00 ～ 20:20（休憩1時間あり）

【場　所】ZOOM会議

【プログラム】（発表タイトル未定、敬称略）

15:00	挨拶と趣旨説明	JST/CRDS 木村 康則
	文部科学省研究振興局	橋爪 淳
	情報処理学会、日本電気	江村 克己
	国立情報学研究所	喜連川 優
	国立情報学研究所	河原林 健一
	京都大学	岡部 寿男
15:20	研究分野発表（1）セキュリティ分野 OVERVIEW	松浦 幹太
	セキュリティ分野最先端研究	盛合 志帆
15:55	研究分野発表（2）プログラム言語分野 OVERVIEW	田浦 健次朗
	プログラム言語分野最先端研究	小林 直樹
16:30	研究分野発表（3）アーキテクチャー分野 OVERVIEW	井上 弘士
	アーキテクチャー分野最先端研究	佐々木 広
17:05	研究分野発表（4）プライバシー分野 OVERVIEW& 最先端研究	佐久間 淳
17:40	休憩（1時間）	
18:40	研究分野発表（5）OS分野　OVERVIEW ＆最先端研究	河野 健二
19:15	パネルディスカッション	
20:15	ラップアップ	
20:20	閉会	

ワークショップ（第２回）

【テーマ】研究成果の応用に関する議論

【日　時】2020 年 6 月 29 日（月）13:00 ～ 17:40

【場　所】ZOOM 会議＋オンサイト（調整中）

【プログラム】（発表タイトル未定、敬称略）

13:00	挨拶と趣旨説明	JST/CRDS 木村 康則
13:10	研究成果の活用（1）三井住友フィナンシャルグループ	谷崎 勝教
13:45	研究成果の活用（2）NTT セキュアプラットフォーム研究所	高橋 克巳
14:20	研究成果の活用（3）NTT セキュアプラットフォーム研究所	千田 浩司
14:55	研究成果の活用（4）日本アイ・ビー・エム東京基礎研究所	福田 剛志
15:30	休憩	
15:45	CRDS の取組み	CRDS 茂木 強
16:00	パネルディスカッション	
17:30	ラップアップ	文部科学省、CRDS
17:40	閉会	

付録２. 参加者一覧

【ステアリングメンバー】

- 喜連川 優　　国立情報学研究所 所長
- 江村 克己　　情報処理学会 会長、日本電気 NEC フェロー
- 河原林 健一　　国立情報学研究所 教授 / 副所長（コーディネーター）
- 岡部 寿男　　京都大学 教授 / センター長（コーディネーター）
- 橋爪 淳　　文部科学省研究振興局 参事官（情報担当）
- 木村 康則　　科学技術振興機構 上席フェロー

【第 1 回ワークショップの発表者】

- 井上弘士　　九州大学 教授
- 佐々木 広　　東京工業大学 准教授
- 河野 健二　　慶應義塾大学 教授
- 松浦 幹太　　東京大学 教授
- 盛合 志帆　　情報通信研究機構 企画戦略室長 兼 上席研究員
- 佐久間 淳　　筑波大学 教授
- 田浦 健次朗　　東京大学 教授 / センター長
- 小林 直樹　　東京大学 教授

【第２回ワークショップの発表者】

- 谷崎 勝教　　日本総合研究所 代表取締役社長、三井住友フィナンシャルグループ CDIO
- 高橋 克巳　　NTT セキュアプラットフォーム研究所 主席研究員
- 千田 浩司　　NTT セキュアプラットフォーム研究所 主幹研究員
- 福田 剛志　　日本アイ・ビー・エム 東京基礎研究所 所長

【ディスカッサント（コメンテーター)】

- 吉川 正俊　　京都大学 教授
- 小野 貴継　　九州大学 准教授
- 塩谷 亮太　　東京大学 准教授
- 品川 高廣　　東京大学 准教授
- 光来 健一　　九州工業大学 教授
- 須崎 有康　　産業技術総合研究所 主任研究員
- 穐山 空道　　東京大学 助教
- 花岡 悟一郎　　産業技術総合研究所 研究チーム長

● 縫田 光司　東京大学 准教授
● 矢内 直人　大阪大学 助教
● 川本 裕輔　産業技術総合研究所 主任研究員
● 曹 洋　京都大学 特定助教
● 宮地 充子　大阪大学 教授
● 五十嵐 淳　京都大学 教授
● 住井 英二郎　東北大学 教授
● 姜 仁河　東京大学 助教
● 鈴村 豊太郎　IBM Thomas Jr. Watson Research Center, Research Scientist
● 井上 拓　日本アイ・ビー・エム 東京基礎研究所 研究員

【文部科学省】

● 橋爪淳　参事官（情報担当）
● 出口夏子　参事官（情報担当）付 参事官補佐
● 齊藤修啓　参事官（情報担当）付 情報科学技術推進官
● 犬塚恵美　参事官（情報担当）付 情報処理推進係
● 福野達也　計算科学技術推進室 係長
● 太田洋一郎　計算科学技術推進室 専門職
● 宇賀神昌利　計算科学技術推進室 調査員

【科学技術振興機構（JST）】

● 木村 康則　CRDS システム・情報科学技術ユニット 上席フェロー
● 茂木 強　CRDS システム・情報科学技術ユニット フェロー
● 若山 正人　CRDS システム・情報科学技術ユニット 上席フェロー
● 青木 孝　CRDS システム・情報科学技術ユニット フェロー
● 高島 洋典　CRDS システム・情報科学技術ユニット フェロー
● 辻 真博　CRDS ライフサイエンス・臨床医学ユニット フェロー
● 松村 郷史　CRDS 環境・エネルギーユニット フェロー
● 村川 克二　CRDS 科学技術イノベーション政策ユニット フェロー
● 桐葉 佳明　戦略研究推進部 ICT グループ 主任調査員

■作成メンバー■

総括責任者：	木村　康則	上席フェロー	（システム・情報科学技術ユニット）
リ ー ダ ー：	茂木　強	フェロー	（システム・情報科学技術ユニット）
メ ン バ ー：	若山　正人	上席フェロー	（システム・情報科学技術ユニット）
	青木　孝	フェロー	（システム・情報科学技術ユニット）
	高島　洋典	フェロー	（システム・情報科学技術ユニット）
	辻　真博	フェロー	（ライフサイエンス・臨床医学ユニット）
	松村　郷史	フェロー	（環境・エネルギーユニット）
	村川　克二	フェロー	（科学技術イノベーション政策ユニット）
	桐葉　佳明	主任調査員	（戦略研究推進部）
	吉田　健司	副調査役	（未来創造研究開発推進部）

※お問い合せ等は下記までお願いします。

CRDS-FY2020-WR-04

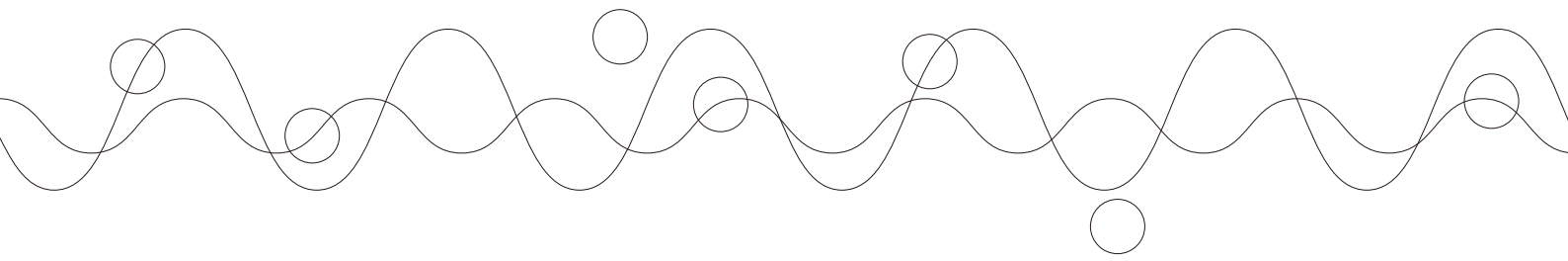
科学技術未来戦略ワークショップ報告書

Society 5.0 システムソフトウエア

令和 2 年 11 月　November 2020
ISBN 978-4-88890-693-7

国立研究開発法人科学技術振興機構　研究開発戦略センター
システム・情報科学技術ユニット
Systems and Information Science and Technology Unit,
Center for Research and Development Strategy Japan Science and Technology Agency

〒102-0076　東京都千代田区五番町 7　K's 五番町
電　話　03-5214-7481
E-mail　crds@jst.go.jp
https://www.jst.go.jp/
©2020 JST/CRDS

許可無く複写／複製をすることを禁じます。
引用を行う際は、必ず出典を記述願います。
No part of this publication may be reproduced, copied, transmitted or translated without written permission.
Application should be sent to crds@jst.go.jp. Any quotations must be appropriately acknowledged.

ISBN 978-4-88890-693-7