

国立大学法人 筑波大学 情報学群 情報科学類

産学間連携推進室

Website: <http://www.ac-room.org/>



2023 年度

活動成果報告書

Academic Collaboration Room Journal 2022

目次

情報科学類 産学間連携推進室の概要（金子尚樹）	1
人為的ミス・いたずらの削減を主眼に置いた飲食店におけるオープンな注文システムの開発（中野あおい）	5
子音情報のみを用いた VR 向けのかな文字入力手法の提案（細島涼雅）	8
Bluetooth Low Energy ビーコンを用いたモーショントラッカーの開発（松田心杏）	11
法律文書の自動解析（金子尚樹）	15
産学間連携推進室主催の学内向けイベントについて析（金子尚樹）	35
モバイル回線を用いた管理用予備回線の構築（服部真吾）	41
遠隔で高度なプログラミング教育を可能にする環境の開発（服部真吾）	42
画像解析による地球の軌道要素の導出（服部真吾）	44
居住空間での業務用電子計算機の運用に関する研究（太田弘樹）	53
情報機器資源リユースプロジェクト活動報告（太田弘樹）	60
WIDE Project つくば NOC 運用報告（太田弘樹）	70
つくば WIDE Project 公開ミラーサービス（太田弘樹）	74
中学生向け産学官連携 ICT 教育イベント開催の総括（根本晃輔）	81

情報科学類 産学間連携推進室の概要

筑波大学 情報学群 情報科学類 産学間連携推進室

学生代表 202212117 金子尚樹 [s2212117@u.tsukuba.ac.jp]

1 概要

筑波大学 情報学群 情報科学類 産学間連携推進室は、主に情報科学類生による研究開発活動の支援を目的として学生主体の研究室であり、工学系学系 F 棟 3F230 室の割り当てを受け活動を行っている。

通常、情報科学類生は 4 年次における研究室配属まで、学内に研究拠点を持つことが無い。そこで産学間連携推進室では 3 年次までに研究活動を行いたい学生に対し学内での研究開発拠点を提供することで活動を支援し、そのような学生が互いに切磋琢磨し合える環境を作ることを目的に各種活動を行っている。また、研究開発した成果を外部にアウトプットする機会も提供し、この研究室から大学発ベンチャー起業や、優秀な技術者が排出される流れを創造することも目的に活動している。

産学間連携推進室は 2014 年度より情報学群情報科学類に移管され、同室の活動に置いて正式な支援及び予算の提供を情報科学類より受けている。これは、現在の多岐に亘る活動に加え、今後のより柔軟な活動を目指す上で大きな礎になっている。

2 目標

2023 年度において産学間連携推進室は、今までと引き続き以下の項目を目標として各種活動を行った。

- 個人または複数人でプロジェクトを立案遂行し、問題点を認識し、その問題を解決する一連の流れを経験することによって総合的な問題解決能力および開発能力の工場を図る。
- 前年度の成果をもとに研究開発活動を継続的にを行い、より長期間に渡るプロジェクトの遂行をお香事を通じて、より大規模なプロジェクトを遂行する能力の向上を図る。

3 目標に対する評価

2023 年度において産学間連携推進室の活動について、上述した目標に対する評価を述べる。

産学間連携推進室においては、様々な活動を行うにあたり、その基礎となる考え方として「問題解決型開発」を取り入れている。これは、既存の技術などについて問題を提起し、その問題を解決するためにソフトウェア等の実装を行うというものである。各活動に置いて、それぞれ問題点を提起しその問題を考察し解決するための実装を行い、問題の解決に挑んでいる。

このように、一連のプロセスに沿って開発できる統合的な開発能力を育成することに重点を置いた点については、今後長期間活動を行っていくにあたり必須であると考えら

れ、評価できると思われる。

4 組織構成について

本章においては、産学間連携推進室の組織構成について記載する。

4.1 略歴

産学間連携推進室は2007年5月頃から活動を開始した。当時は、活動拠点として旧第三エリア3B棟3B408号室を利用していた。当時の活動内容は主にメンバー個人が各々興味のある分野について個人の最良にて何かしらの活動を行うというものであり、組織としてまとまって活動を行うといったことはなく、活動拠点を個人が個人の目的に理_y王するという形態が取られていた。

また、厚生労働省が行っている特定健康診査に関するデータ入力ソフトウェアの開発をオーダーメイド創薬株式会社（現、株式会社 LOTUS）およびソフトイーサ株式会社との産学連携活動の一環として行った。このソフトウェアは現在、全国各地の診療所等で利用されている。

その後、第三エリア3B棟の改修工事に伴い活動を一時的に休止した。その期間に現在の活動拠点である第三エリア工学系学系F棟3F230号室に利用開始準備を進め、新たな体制での組織運営を行うための基盤の整備を行った。3F230号室が利用可能になった後、各種インフラ整備および運営体制の整理が行われた。

更に2011年から毎年春に研究開発成果の報告会を学内にて行った。成果報告会はプロジェクトが目指す目標や問題点、それに対する解決方法などに関して実際に行った実装の開設などを中心に行われた。また、2023年には情報科学類の主催する夏のオープンキャンパスにてもスライド発表を行い、来訪した高校生らに対し研究の紹介等を行っている。

4.2 人員構成

産学間連携推進室は、学年主体の組織である。現在では以下の学生が所属しており、互いに様々な分野で活動を行っている。

- 大学院 システム情報工学研究科 博士後期課程：1人
- 大学院 システム情報工学研究科 情報理工学位プログラム 博士前期課程：2人
- 情報メディア創生学類4年：1人
- 情報科学類3年：1人
- 物理学類3年：1人
- 情報科学類2年：2人

5 他組織との連携

5.1 WIDE Project

産学間連携推進室は2007年11月に筑波大学で初めてWIDE Projectの活動拠点となった。その一環として、筑波大学学術情報メディアセンター内に構築したNOC (Net-

work Operation Center)においてネットワーク分野の様々な研究開発活動を行っている。

5.2 ソフトイーサ株式会社

産学間連携推進室は、筑波大学発ベンチャー企業であるソフトイーサ株式会社から様々な支援を受けており、研究開発活動に必要な計算機リソース等をはじめ様々な機材の貸与を受けている。また、実験用途として IPsec ネットワークの接続性を提供するなど、相互に支援を行っている。また、PacketIX NET 公開実験にも協力したり、モバイル回線を用いた管理用予備回線の構築の方法を共同で模索したりなどしている。

5.3 輝日株式会社

産学間連携推進室のメンバーが立ち上げた筑波大学発ベンチャー企業である。ネットワーク機器の提供や技術的なアドバイスなどの支援を受けているほか、共同でネットワーク関連の研究および実証実験、教育活動などを行っている。

6 活動概要

産学間連携推進室における活動の基準は、各個人の興味にある。所属するメンバーがやりたい分野について様々な研究開発活動を自由に行うことでさらなる技術力向上を目指している。また、複数のメンバーが協力して分野をまたがった研究開発を行うことで、様々な知見を得て成果につなげることも目的としている。

具体的なプロジェクトとしては以下のものがある。

6.1 産学官連携プロジェクト

- 中学生向け産学官連携 ICT 教育イベント「インデペンデンス・サーバー・デイ」（つくば市総合教育研究所・OPEN プロジェクト）
- 「テキストベースのプログラミングに挑戦してみよう！」（つくば市総合教育研究所・ソフトイーサ株式会社）

6.2 産学連携プロジェクト

- VPN トンネルソフトウェアの活用に関する研究および公開実験（ソフトイーサ株式会社）
- 固定 IP アドレス割り当て型サービス（インターリンク株式会社・ソフトイーサ株式会社）
- モバイルデータ回線のボンディング技術（輝日株式会社）
- セキュア公衆無線 LAN に関する研究及び実証実験（輝日株式会社）
- 小・中学生向けプログラミング講座開催支援（輝日株式会社）

6.3 他組織連携プロジェクト

- WIDE プロジェクト筑波大学 NOC の運営（WIDE プロジェクト）
- 国内向けパブリックミラーサーバの運営（WIDE プロジェクト）
- 情報機器資源のリユース・リサイクルに関する支援（筑波大学システム情報系技術室）

6.4 学生活動プロジェクト

- 遠隔で高度なプログラミング教育を可能にする環境の開発
- 画像解析による地球の軌道要素の導出
- 法律文書の自動解析
- Bluetooth Low Energy ビーコンを用いたモーショントラッカーの開発
- 人為的ミス・いたずらの削減を主眼に置いた飲食店におけるオープンな注文システムの開発
- 子音情報のみを用いた VR 向けのかな文字入力手法の提案

筑波大学 産学間連携推進室 活動成果報告書

人為的ミス・いたずらの削減を主眼に置いた 飲食店におけるオープンな注文システムの開発

情報科学類 2 年 202312794 中野あおい

1 はじめに

近年の飲食店においては、ホール側・客のテーブルにタブレット端末を配置し、客がそこから注文を行うシステムが一般に採用されている。しかしそのような端末には、むやみに端末を操作してシステムの混乱を試みようといたずらのリスクや、通信の不具合により注文が届かなかったり、会計が一致しなくなってしまうりするような基本的な問題が存在する。また、端末の設定がうまくいかず不正に会計が行われる等の人為的ミスが原因のトラブルが起こる可能性がある。

そのほかにも、このようなシステムは SIer のようなシステム開発企業が開発していることがほとんどで、知見の共有等がなされにくい傾向にあり、さらに飲食店ごとのカスタマイズに高額な料金が発生する可能性があるなど、さまざまな飲食店のニーズに応じた形となっていないように思われる SIer 等営利企業の既存のクローズドなシステムは導入初期に高額な費用が必要で、さらに保守費用も必要になるため、注文システムの導入に費用をかけづらい場では採用されにくく、中小規模・個人経営の飲食店や大学の学園祭の模擬店等では、注文をシステム化することを望んでもハードルが高く手を出すことができない。

上記のような問題を解決したいと私は考える。そこで、人為的ミス・いたずらの削減を主眼に置いたオープンな飲食店向けのシステムを開発することで、これらの問題を解消し、また飲食店の注文システムといった一般的なシステムの開発市場を広げていきたい。また、オープンソースで開発することにより、他の人が手を加えることができ、知識があれば欲しい機能を自分で追加することができる。

2 実装方法

Web アプリケーションとしての実装を想定する。客席に置かれた端末から注文する形式（客席端末版と呼ぶことにする）や、客席端末を用意するのが難しい場合に向けて、客が自身のスマートフォンで QR コードを読み取って注文することができる形式（スマートフォン版と呼ぶことにする）にも対応する。

飲食店のレジで情報を集約する機器を親機と呼び、客側の端末や店舗のキッチンに置く端末などを子機と呼ぶ。今回は親機、客側の子機、キッチン側の子機の実装と各機器が共有するデータベースの実装を行う。

アプリに端末で不正な操作が行われたときに親機に通知されるシステムや、一定の時間おきに親機と子機のデータの突合せを行うシステムなどを開発するアプリに組み込む。バックエンドの言語は PHP を使用し、データベースは PostgreSQL、フロントエンドでは React を使用しようと考えている。

3 先行研究

現在主に飲食店で使われているセルフオーダーのシステムは、店内に設置された券売機式の端末で注文する方式や、客がスマホのアプリから注文する方式、客席に置かれたタブレット型の端末を用いて注文する方式、

客が QR コードを読み取って Web アプリなどから注文する方式などがある。

これらを用いることで店は人件費を削減し、更に売り上げや注文履歴の管理もしやすくなる。

現在客席端末としての使用を予定されるソフトウェアにオープンソースのものは見つからなかった。また無料で端末にインストールすることができる、POS と連携する注文アプリの中には、かつて私が実際に飲食店で使用したアプリに、アプリが途中で中断されるとデータが破損し、正常に会計が行われないものがあった。これは、誤操作や通信トラブルへの対策が十分でないと感じる点において改善の余地があると考える。

客が自身が所有するスマホで注文する形式の Web アプリケーションは、オープンソースのもの [1] が存在する。これはいたずら防止などの策を提供できていないように感じるため、改善の余地があると考える。よって、これを一部参考にしてスマートフォン版の作成、客席端末版への応用を考える。

4 進捗

これまでフロントエンド側の開発を進め、客席端末版のシステムの見た目の部分を整えた。以下の図 4.1 が今回作成したものである。ジャンルのボタンを押すことによって商品の一覧のセットが切り替わる。また、一覧の商品を選択すると図 4.2 の画面に遷移するようになっている。



図 4.1: 客席端末のフロントエンド



図 4.2: 商品ページ

5 今後の展望

今後は、親機や店舗のキッチン側の端末に表示する画面の実装や、学習を進めバックエンド、データベースの実装などを考えている。また、現在は素の HTML・CSS、JavaScript で書いているコードを React などのフレームワークを用いて記述するつもりである。

6 おわりに

本研究に取り組んだところ、セルフオーダーシステムの実装は複雑で、SIer などの企業が開発をすべて請け負う理由をよく理解できた。しかし今後もセルフオーダーシステムの需要は増えると予測され、オープンソースのソフトウェアの需要も存在すると考える。また、ソフトウェアの機能では防ぐことが難しいヒューマンエラーなどの考えられる問題へのアプローチもしていきたいと考える。

参考文献

[1] オープンソースのセルフオーダーシステム. <https://github.com/kawax/self-ordering>.

子音情報のみを用いた VR 向けのかな文字入力手法の提案

情報科学類 2 年 細島 涼雅
s2311951@u.tsukuba.ac.jp

第 1 章 はじめに

1.1 背景

近年 HMD (Head Mounted Display) の普及が進んだことで、VR (Virtual Reality) がより身近なものになっている。VR 環境下ではボイスチャットや声を用いた文章の入力がよく用いられるが、文字入力を伴うコンテンツを拡大するためには音声以外の VR 環境下での様々な文字入力機能の充実が必要である。

現在多くの HMD に標準で搭載されている文字方法はコントローラで仮想的なキーをポインティングすることにより入力を行うような直接ポインティングによるインタラクションである。主流の直接ポインティングによる入力手法では、物理的コントローラのアバタとなる VR 空間の仮想コントローラの先から Ray を飛ばし、Ray がオブジェクトと衝突した地点をポインタの位置として利用する。また仮想的なキーボードは一般的な QWERTY レイアウトか五十音のレイアウトが一般的に利用される。

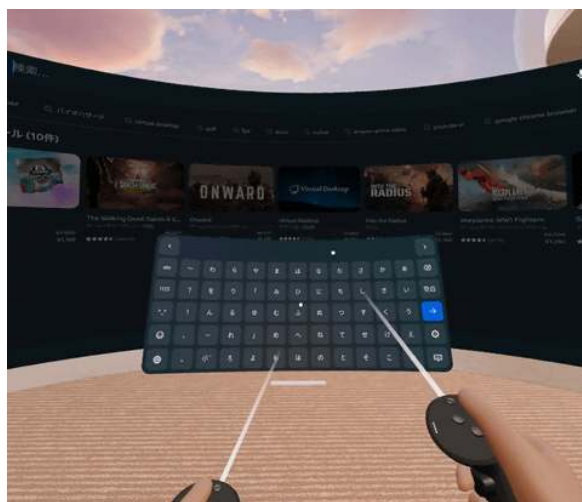


図 1: 現在主流の入力インターフェース (PC Watch 2023 年 3 月 4 日の記事より引用)

しかし、このインターフェースの課題点として「キーが多く、またキー自体が小さいため、ポインタするのが難しく入力ミスが発生しやすい」、「空中で腕を動かして操作する必要があるため疲労が大きい」といったものが挙げられる。さらに、VR 空間上では空間把握が難しいため、文字入力においては様々な支援が求められる。

1.2 研究の目的

本研究では、前節で挙げた課題を解決するような入力システムの提案、及びその開発を目指す。また、その提案手法についてユーザビリティを評価し、提案の妥当性を確認する。

また、VR コンテンツは OSS 文化が発達しており、コンテンツがオープンソフトウェアとして公開されているものも多い。開発したインタフェースをオープンソフトウェアとして公開し、多くのユーザにとって快適な文字入力インタフェースを提供することを目指す。

第 2 章 提案する手法

2.1 概要

現行手法では、以下の問題点が考えられる。

- キーが多く、キー自体が小さいため、直接ポインティングによるポインティングが難しく入力ミスが発生しやすい
- 誤入力への対応
- 疲労が大きい

これらの問題点を、本研究では以下のアプローチで解決することを目指す。「キーが多く、キー自体が小さいため、直接ポインティングによるポインティングが難しく入力ミスが発生しやすい」問題は、フリック入力を用いるキーレイアウトを使い、キー数を少なくすることでの解決を目指す。入力においては子音情報のみを入力し、その情報のみを用いて文字変換を行う。

VR 空間上の入力手法としてフリック入力手法を活用する方法は川口らや清原らにより提案・研究がなされているが、いずれも指をフリックすることが要求され、子音情報のみを利用するというアプローチは取っていなかった。[1], [2]

また、誤入力・疲労度への対応は、予測変換を用い、欠損した文字の補完を行うことにより解決を目指す。

2.2 先行研究

前述の通り、フリック入力手法のキー配列を応用した入力支援についての研究はいくつか先行研究がある。川口らや清原らにより、触感フィードバックを伴ったフリック入力においてその有用性について研究がされている。[1], [2]

田中らにより、子音情報のみを用いた携帯電話における日本語入力手法について検討がされている。[3]

子音漢字変換はかな漢字変換に比べ、実に 7 倍もの単語候補を扱う必要が出てきてしまうが、隠れマルコフモデルに PPM を融合させることにより、動的な言語モデルを提案して用いることにより、ユーザの単語選択に適合する順序で整列させることにより解決を試みている。熟練したユーザは十分に高速に文書入力が可能であることが示されている。

第 3 章 研究方法

このシステムの入力容易さ及び入力可能速度・疲労度を検証するための実験として、ユーザ実験を行う。

各ユーザには一般的な入力手法及び提案手法の両方を使用させ、複数の例文を入力させることにより評価を行う。その後、この提案手法について入力のしやすさなど 5 段階のリッカート尺度による SUS (System Usability Scale) で評価し、システムのユーザビリティを評価する。

なお、本システムは普段からフリック入力を行っている者とそうでない者・そして HMD の操作に慣れている者とそうでない者で入力速度に大きく違いが出ると考えられる。そのため、4 つのグループで分けて統計も取る。

第 4 章 進捗と今後の展望

本研究に取り組んでから日が浅いため目立った進捗はないが、主に以下のことを行った

- 既存入力手法についての調査
- 先行研究の調査
- Unity を用いた VR アプリケーションの開発

今後は、メインシステムの実装・開発とともに、ユーザ実験のための準備も進めていきたいと考えている。

参考文献

- [1]. 川口航平, 磯本俊弥, 志築文太郎, 高橋伸 :VR 向けの掌上における日本語フリック入力手法の提案, ヒューマンインタフェースシンポジウム 2019
- [2]. 清原 隆一, 沢田 篤史, 野呂 昌満 : VR 環境における文字入力支援システム, 第 29 回ソフトウェア工学の基礎ワークショップ (FOSE2022)
- [3]. 田中 久美子, 犬塚 祐介, 武市 正人 : 携帯電話における日本語入力 - 子音だけで日本語が入力できるか, 情報処理学会論文誌 Vol.43 No.10, 2002

Bluetooth Low Energyビーコンを用いた モーショントラッカーの開発

筑波大学 理工学群 物理学類 2年 松田 心杏

1 概要

1.1 研究の背景

従来の家庭用VRトラッカーは、主に光学式、慣性測定ユニット(IMU)式、カメラによるモーションキャプチャ方式に依存していた。

これらの技術は、一定の精度と体験を提供するが、高価であり、トラッカーの重量増加、頻繁な充電の必要性、カメラやトラッキングステーションの必要性など、多くのデメリットがある。このような煩雑さは、ユーザー体験を損なう要因となり、VR技術の普及を妨げていると考える。

1.2 研究の目的

Bluetooth Low Energyは、安価で量産が容易な、極めて一般的な技術である。

そのため、Bluetooth Low Energyを用いたトラッカーを作成した場合、それは安価で量産可能かつ軽量なものになると考えられる[2]。

さらに、Bluetooth 5.1で規格に追加された方向検知機能を活用することで[6]、従来の方法に比べてさらに高いトラッキング精度を実現できると考えられる。

安価で量産可能なトラッカーを開発。これが実際に量産・販売されれば、トラッカーは誰でも手の届くものになる。

一般ユーザーのVRへの参入障壁を下げることによるVR技術のさらなる浸透が、この研究の目的である。

1.3 技術的アプローチ

1.31 使用する機材の選定と評価

最初のステップとして、市場で利用可能なマイコンの中から最適なモデルを選定する。選定基準には、コスト、消費電力、信号の強度、対応するBluetoothのバージョンなどが含まれる。

1.32 プロトタイプの開発

選定したマイコンを用いて、初期プロトタイプのVRトラッカーを開発する。

このプロトタイプでは、物理的な動きを正確に追跡することができるよう設計する。

1.321 具体的手法

Bluetooth Low Energyビーコンでは電波強度を利用した端末間距離の推定が可能である。そのため、この電波強度を利用すれば、三点測位の要領で、ある端末の座標を得ることができる。すなわち、座標が既知の三点から、ある一点までの距離がわかれば、空間上のある一点の座標を特定することができる。この手法そのものは一般的であり、よく見られる。

ここでは、一般的な座標の取得だけでなく、測位精度の向上を目的としてプロトタイプを開発する。Bluetooth Low Energyビーコンに関する測位精度の向上に関しては、複数の先行研究が存在し[5]、これらを参考にしつつ開発する。

1.33 ソフトウェア開発と統合

データを解析し、VR環境に適用するためのソフトウェアを開発する。このソフトウェアは、ビーコンの位置と方向データを収集し、VRデバイスにフィードバックを提供することで、ユーザーの動きをVR空間内で再現する。

1.34 実験と評価

開発したトラッカーの性能を評価するために、複数の実験を実施する。トラッキングの精度、反応速度、消費電力、ユーザー体験のテストが含まれる。

1.4 先行研究との差別化

先行研究として、Bluetooth Low Energy 4.2を用いたトラッカー[4]が開発されている。しかし、クラウドファンディングでの資金調達に失敗し、市場に出ることはなかった。

この先行研究から、Bluetooth Low Energyを用いてトラッカーを作ること、かなり安価かつ軽量になることが期待できる[2, 4]。

本研究では、Bluetooth 5.1の方向検知機能を利用することで、先行研究を超えるトラッキング精度を実現することを目指す。

また、この先行研究ではSDKを提供するにとどまっているが、本研究ではOculusやHTC Viveなどの主要な家庭用VR機材と、より手軽に連携可能なインタフェースも含めて開発する。すなわち、非エンジニアのVRユーザーでもGUIから利用できるようにする。

2 本研究の着想に至った経緯など

本研究の着想は、現行のVRトラッカーが高価であり、その扱いが煩雑であるという問題意識から生まれた。この問題は、VR技術の普及を妨げる大きな障壁となっている。

例えば、現在最も家庭用VRのトラッカーとして一般的で入手しやすいと私が考えている、HTC社のトラッカーは1つ19000円[3]である。

全身のモーションキャプチャをするには、これを10個ほど、加えてベースステーションを購入しなければならない。

Bluetooth Low Energyの規格を調査する中で、その低消費電力の特性や広範囲にわたる互換性が、より簡易かつ経済的なトラッカー開発に貢献できる可能性があると考えた。

さらに、先行研究がBluetooth Low Energyを利用したトラッカーの開発において一定の成果を示していたことから、この方針が有望であることも確認できる。[4]

しかしながら、個人での開発には知名度の獲得や製品の市場への導入において多くの課題が伴う。具体的には、個人が開発を行った場合、製品の知名度を上げることが困難である。

他の多くの個人開発のVRトラッカーのように、予約注文を取る必要が生じればユーザーへのリーチが限られてしまう。

産学間連携推進室での研究を通じてトラッカーの開発を行うことで、企業や広範なユーザー層の注意を引きやすくなり、結果として優れたトラッカーの普及につながる可能性が高まる。

また、産学間連携推進室で研究を進めることにより、周囲の人々からアドバイスを受けることが可能になる。このような環境は、単独で開発を進めるよりも、より良い成果を生み出すことを可能にすると考えられる。

3 成果と課題点

以上の計画書を作成・提出し、実際の着手開始が2024年4月15日であったため、まだあまり成果という成果はないが、作業を進めている。

まずESP32の開発ボードを利用し、Bluetooth Low Energy規格を利用した測位を実際に実装し、素朴に電波強度を利用するだけでは大きなノイズが含まれてしまうことを認識した。

そこで現在、そのノイズの評価を適切に行い、さらに以後の開発でノイズ抑制措置の効果を確認しやすくするため、信号のプロット・ビジュアライズを行うためのソフトウェアを作成している。

参考文献

[1] SlimeVR Full-Body Tracker, “SlimeVR Full-Body Tracker | Crowd Supply”, Crowd Supply, 2021, <https://www.crowdsupply.com/slimevr/slimevr-full-body-tracker>, May 20 2024

- [2] Kang Eun Jeon. BLE Beacons for Internet of Things Applications: Survey, Challenges, and Opportunities. IEEE Internet of Things Journal. 01 January 2018, Volume: 5, Issue: 2, p811-828.
- [3] HTC Corporation. “VIVEトラッカー(3.0)|VIVE 日本”, HTC Vive 公式ホームページ. 2021.
<https://www.vive.com/jp/accessory/tracker3/>, May 20 2024
- [4] WonderWonder. “WonderSense, Sensor Node to Shape Your Ideas with No Time by WonderWonder — Kickstarter”. Kick Starter. 2020.
<https://www.kickstarter.com/projects/ww-ws/wonder-sense-ble-wearable-small-motion-tracker>, May 20 2024.
- [5] 山岸茉莉夏. BLE ビーコンの設置個数と配置方法が屋内測位精度に及ぼす影響について. 2018年度卒業研究論文要旨集. 2018,
<https://www.jc.u-aizu.ac.jp/news/management/gr/2018/16.pdf>, May 20 2024.
- [6] Bluetooth Core Specification Version 5.1, Kirkland, WA, USA, 2019.

法律文書の自動解析

筑波大学 情報学群 情報科学類 産学間連携推進室
情報科学類 3 年 202212117 金子尚樹 [s2212117@u.tsukuba.ac.jp]

1 はじめに

法律の世界は、六法のような基本的な法律があり、それらが埋められない隙間をより詳細な規則を定める法令が埋め、それでも埋められない隙間を判例が埋めるような構造となっている。

法的な問題の解決を行う時、問題となっている行動が法的にどう分割されるのか、そしてそれぞれがどの法令や判例に対応しどのような判定になるのかをまとめ、最終的な結論を導き出す必要がある。問題の分割や結論を出す部分は依然として人が担当しなければならないところではあるが、細かい法令や判例を読みながら分割された細かい問題に対する判断をしていく過程をソフトウェアで支援することは可能である。細かい手続きを定めた法令は実務上とても重要でありながらその条文は長く複雑であることが多く、ソフトウェアによって自動で適切な注釈を付けることができれば、法務業務の負担軽減に繋がり社会的な意義があると考えている。

本プロジェクトでは、法令・条例・判例などの法律文書に対して読み替え規定文や略称定義文、他の法令などへの参照表現について抽出することを目指している。法令を解析して構造化する研究は様々あるが、本研究では特定の項目についてのみ解析を行い、必要な情報を抽出することに主眼を置いた。これにより、多く存在する例外への対処をしやすくなり、解析の対象としない項目については無視をすることでスムーズな検証が行えるようになることを期待している。

2 前年度までの成果

前年度までの成果について報告する。

2.1 読み替え規定文の解析

2.1.1 背景

法律文書には準用と読み替えという概念が存在する。これは、一つの条文を使いまわしつつ必要最小限の修正を加えて、設置の目的などが似ている複数の規定を定義する方法である。使いまわす手法が準用と呼ばれ、準用の際に加える必要最小限の修正が読み替えである。図1のように、まず規定 A とそれを記述する条項 A が存在する。規定 A に似た新しい規定 A' ができたとき、条項 A を読み替え規定で修正しながら規定 A' の記述を行う。これが準用である。

具体例を以下に挙げる。医薬品、医療機器等の品質、有効性及び安全性の確保等に関する法律（昭和三十五年法律第百四十五号）の第七条第四項では薬局の管理について下のように定めている。

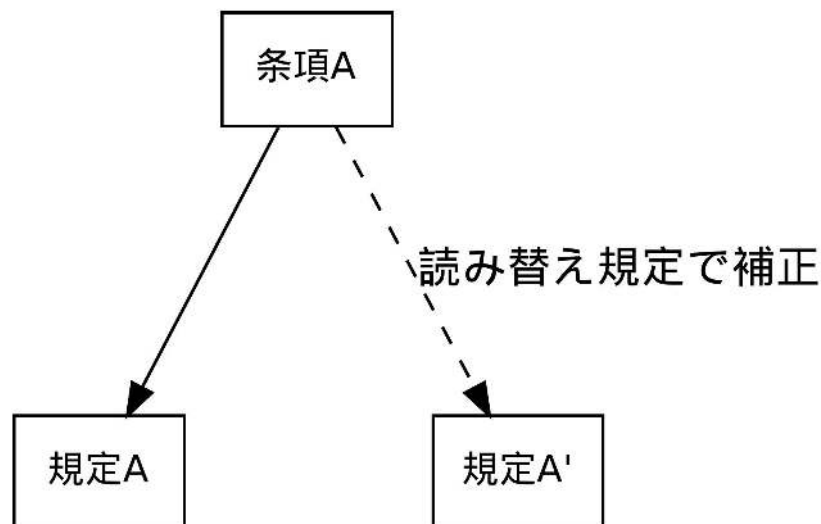


図 1：準用と読み替えの概念図

薬局の管理者（第一項の規定により薬局を実地に管理する薬局開設者を含む。次条第一項及び第三項において同じ。）は、その薬局以外の場所で業として薬局の管理その他薬事に関する実務に従事する者であつてはならない。ただし、その薬局の所在地の都道府県知事の許可を受けたときは、この限りでない。

そして同法律の第八条第一項では薬局の管理者の義務について以下のように定めている。

薬局の管理者は、保健衛生上支障を生ずるおそれがないように、その薬局に勤務する薬剤師その他の従業者を監督し、その薬局の構造設備及び医薬品その他の物品を管理し、その他その薬局の業務につき、必要な注意をしなければならない。

そして同法律の第十七条第八項では医薬品等総括製造販売責任者について以下のように定めている。

医薬品製造管理者については、第七条第四項及び第八条第一項の規定を準用する。この場合において、第七条第四項中「その薬局の所在地の都道府県知事」とあるのは、「厚生労働大臣」と読み替えるものとする。

薬局の管理について規定している第七条第四項と薬局の管理者の義務について規定している第八条第一項を医薬品製造管理者について準用することで、語句が少し違うだけの条文を再度書くコストを減らしている。この条文を読む人は「医薬品製造管理者について準用する」という文言をもとに、第七条第四項と第八条第一項内に登場する「薬局の管理者」という語を「医薬品製造管理者」に置換し、第十七条第八項の条文が意図する文章を考えて読むこととなる。ただし、薬局に許可を出すのが都道府県知事であるのに対

して、医薬品製造所に対して許可を出すのが厚生労働大臣であるという制度上の差異があり、これは非自明であるため条文中で読み替えの指示を行っている。このため、第十七条第八項が意図する条文は以下のようなものになる。

医薬品製造管理者 は、その 医薬品製造所 以外の場所で業として 医薬品製造所 の管理その他薬事に関する実務に従事する者であつてはならない。ただし、その 医薬品製造所 の所在地の 厚生労働大臣 の許可を受けたときは、この限りでない。

医薬品製造管理者 は、保健衛生上支障を生ずるおそれがないように、その 医薬品製造所 に勤務する薬剤師その他の従業者を監督し、その 医薬品製造所 の構造設備及び医薬品その他の物品を管理し、その他その 医薬品製造所 の業務につき、必要な注意をしなければならない。

元の条文が以下の物であったことを考えると、文章量の圧縮の効果があることがわかる。

医薬品製造管理者については、第七条第四項及び第八条第一項の規定を準用する。この場合において、第七条第四項中「その薬局の所在地の都道府県知事」とあるのは、「厚生労働大臣」と読み替えるものとする。

準用と読み替え規定は文章量の圧縮の他に「似ている条文をまとめ、似ている中での相違点を明確にする」という役割も存在する。これにより、過去の違う条文に関する判例をどこまで適用することができるという判断が適切にできるようになる。

このように準用と読み替え規定は便利な物であるが、読み替え規定は単語の置き換えを自然言語で指示するという特性上、

- 登場する文が不自然になる
- 置き換える単語によっては一文がとても長くなる

という難しさが存在する。保険商品の難しさについて調査を行った竹井らも保険約款などの法律文書の内容の解釈が困難になっている原因の一つに読み替え規定の多用があると指摘している [1]。

複雑な条文の具体例を挙げる。租税特別措置法（昭和三十二年法律第二十六号）の第七十条の七の四第十三項は以下のような条文である。

第七十条の七の二第二十二項から第二十六項までの規定は、認定相続承継会社について同条第二十二項に規定する評定が行われた場合における納税猶予分の相続税額の計算及び免除について準用する。この場合において、同項から同条第二十五項までの規定中「経営承継期間」とあるのは「経営相続承継期間（第七十条の七の四第一項の規定の適用を受ける経営相続承継受贈者に係る贈与者が同条第二項第五号イ又はロに掲げる日のいずれか早い日の翌日以後に死亡した場合にあつては、当該経営相続承継受贈者に係る前条第二項第六号に規定する経営贈与承継期間）」と、「第一項」とあるのは「第七十条の七の四第一項」と、「対象非上場株式等に」とあるのは「対象相続非上場株式等に」と、「認定承継会社」とあるのは「認定相続承継会社」と、「経営承継相続人等」とあるのは「経営相続承継受贈者」と、「対象非上場株式等（）」とあるのは「対象相続非上場株式等（）」と、「相続により取得をした対象非上場株式等の当該相続の時における」とあるのは「対象相続非上場株式等の」と、同条第二十三項中「を第一項」とあるのは「を同条第一項」と、「第二項第五号」とあるのは「同条第二項第四号」と読み替えるものとする。

これを読み替えられる語と読み替える語で対応が取れるように色分けをすると次のよ

うになる。

第七十条の七の二第二十二項から第二十六項までの規定は、認定相続承継会社について同条第二十二項に規定する評定が行われた場合における納税猶予分の相続税額の計算及び免除について準用する。この場合において、同項から同条第二十五項までの規定中「経営承継期間」とあるのは「経営相続承継期間（第七十条の七の四第一項の規定の適用を受ける経営相続承継受贈者に係る贈与者が同条第二項第五号イ又はロに掲げる日のいずれか早い日の翌日以後に死亡した場合にあつては、当該経営相続承継受贈者に係る前条第二項第六号に規定する経営贈与承継期間）」と、「、第一項」とあるのは「、第七十条の七の四第一項」と、「対象非上場株式等に」とあるのは「対象相続非上場株式等に」と、「認定承継会社」とあるのは「認定相続承継会社」と、「経営承継相続人等」とあるのは「経営相続承継受贈者」と、「対象非上場株式等（）」とあるのは「対象相続非上場株式等（）」と、「相続により取得をした対象非上場株式等の当該相続の時ににおける」とあるのは「対象相続非上場株式等の」と、同条第二十三項中「を第一項」とあるのは「を同条第一項」と、「第二項第五号」とあるのは「同条第二項第四号」と読み替えるものとする。

これからわかるように、この長い条文では

- 「経営承継期間」→「経営相続承継期間（第七十条の七の四第一項の規定の適用を受ける経営相続承継受贈者に係る贈与者が同条第二項第五号イ又はロに掲げる日のいずれか早い日の翌日以後に死亡した場合にあつては、当該経営相続承継受贈者に係る前条第二項第六号に規定する経営贈与承継期間）」
- 「、第一項」→「、第七十条の七の四第一項」
- 「対象非上場株式等に」→「対象相続非上場株式等に」
- 「認定承継会社」→「認定相続承継会社」
- 「経営承継相続人等」→「経営相続承継受贈者」
- 「対象非上場株式等（）」→「対象相続非上場株式等（）」
- 「相続により取得をした対象非上場株式等の当該相続の時ににおける」→「対象相続非上場株式等の」
- 「を第一項」→「を同条第一項」
- 「第二項第五号」→「同条第二項第四号」

の9個の読み替えが規定されており、とても長い文や読点から始まる読み替えが存在することがわかる。これを誤読せずに読むことはとても難しく、機械的な抽出が可能になると人への負担を大きく軽減することができる効果が望める。

法律文書の構造化を行っている先行研究は複数あるが、信岡らや木村らの提案手法の中では読み替え規定文の解析を行っておらず、今後の課題とされている [2, 3]。

2.1.2 手法及び実装と結果

読み替え規定文の書き方について磯崎は「この場合において、「A」とあるのは「B」と読み替えるものとする。」となることが基本形であると指摘している [4]。しかし、より詳細に読み替え規定文の構造を観察したところ、

1. まず (a)・(b) のパターンが0回以上繰り返される

- (a) “「～～」とあり” という表記が 0 回以上繰り返される
- (b) その後 “「～～」とあるのは「～～」と、” という表記が出現する
2. 次に “「～～」とあり” という表記が 0 回以上繰り返される
3. 最後に “「～～」とあるのは「～～」と読み替える” という表記が出現するという規則で記述されていることが分かった。このうち、

- 「～～」とあり
 - 「～～」とあるのは
- と書かれている場合の鍵括弧の中の用語は読み替えられる語であり、
- 「～～」と、
 - 「～～」と読み替える
- と書かれている場合の鍵括弧の中の用語は読み替える語である。

そこで、テキスト中に出現する鍵括弧をくくりだし、閉じ鍵括弧直後の語によって判定を変えてスタックに積むというルールベースの解析を行うこととした。

作成したソフトウェアは MIT ライセンスの元 GitHub で公開している^{*1}。

入力は e-gov 法令検索 [5] で公開されている法令 XML データとし、出力は

- 出現箇所
- 読み替えられる語
- 読み替える語

の情報が記述された JSON 形式となる。

2022 年 12 月 4 日時点で公開されていた法令データをもとに解析した。読み替え規定文 11672 個に対して提案する手法を用いて解析したところ、11428 個は抽出に成功し、残りの 244 個は失敗した。

2.1.3 考察

このように単語の出現するパターンのみで読み替え規定文のうち 97% 以上の読み替え規定文を解析することができた。今後の課題として解析に失敗した条文を調べたところ、大きく分けて

- 文中の括弧の対応が壊れているもの
- 表形式で与えているもの
- 読み替える語等の指定に鍵括弧をしていないもの

の 3 つに分けられることがわかった。今後はこれらの括弧の対応がとれていない条文や表形式で与えられている条文の自動検出と表の解析機能を実装することが目標である。また、後述する法令名の自動抽出機能と組み合わせることで、読み替え規定文適用後の条文の自動生成を行えるようになりたいと考えている。

2.1.4 情報処理学会全国大会での報告と表彰

この結果を 2023 年 3 月 2 日～4 日に電気通信大学で開催された情報処理学会第 85 回全国大会において「単語の出現規則に着目した法令中の読み替え規定文の解析手法の

^{*1} https://github.com/japanese-law-analysis/analysis_yomikae

提案」と題する研究発表で報告を行いました [6]。

その発表が評価され、学生奨励賞と大会奨励賞の二つの賞を受賞し、表彰されました^{*2}。

2.2 略称・定義規定文の解析

2.2.1 背景

法律文書では文字で書き起こすと長くなる概念が頻繁に登場する。これを毎回書いていると文章量がとても長くなってしまい、読みにくくなる。そこで、それらに対して略称を与えてこの問題を解決することがある。例えば、航空法施行規則（昭和二十七年運輸省令第五十六号）の第二百三十六条の七十一第一項第四号では以下のような条文が書かれている。

国土交通省、防衛省、警察庁、都道府県警察又は地方公共団体の消防機関その他の関係機関の使用する航空機のうち搜索、救助その他の緊急用務を行う航空機の飛行の安全を確保する必要があるものとして国土交通大臣が指定する空域（以下「緊急用務空域」という。）

この例では、「国土交通省、防衛省、警察庁、都道府県警察又は地方公共団体の消防機関その他の関係機関の使用する航空機のうち搜索、救助その他の緊急用務を行う航空機の飛行の安全を確保する必要があるものとして国土交通大臣が指定する空域」という概念に対して、「緊急用務空域」という略称を定義している。そしてこの条文の直後である同条第二項では以下のような条文が書かれており、定義された略称が使われていることがわかる（下線筆者）。

国土交通大臣は、前項第四号の規定による指定をしたときは、インターネットの利用その他の適切な方法により、その旨及び当該指定に係る 緊急用務空域 を公示しなければならない。

もしこのような略称が定義できなければ、上の条文は

国土交通大臣は、前項第四号の規定による指定をしたときは、インターネットの利用その他の適切な方法により、その旨及び当該指定に係る 国土交通省、防衛省、警察庁、都道府県警察又は地方公共団体の消防機関その他の関係機関の使用する航空機のうち搜索、救助その他の緊急用務を行う航空機の飛行の安全を確保する必要があるものとして国土交通大臣が指定する空域 を公示しなければならない。

と書かねばならず、とても読みにくくなってしまうことがわかる。

このように略称はとても便利なものであるが、略称の元の定義を知る必要がある場合も出てくる。例えば、

国土交通大臣は、前項第四号の規定による指定をしたときは、インターネットの利用その他の適切な方法により、その旨及び当該指定に係る緊急用務空域を公示しなければならない。

^{*2} <https://www.coins.tsukuba.ac.jp/2024/240327.php>

という条文を読んでいるときに「緊急用務空域は誰が指定するのか」という情報も知る必要が出ると、「緊急用空域」という語の定義に戻る必要がある。このとき、自動で定義を提示することができるのと一々語を定義する条文に戻る必要が無くなり、便利である。

略称の定義文の自動抽出を行う手法は既に中村らによって提案されている [7]。この論文では法律文コーパスを用いて解析を行う方法を提案している。しかし、この手法では登場する語がコーパスに登録されているものでないといけないため、定義された略称が登場する場合には対応できない。そのため

- 条文中に複数の定義文が登場する場合、先頭の物しか解析できない
- 括弧書き内の定義文は解析できない
- 条文の先頭から 100 字以内に書かれたものしか解析できない

という強い制約が存在している。そこで、より汎用的に略称・定義規定文の解析を行うための方法を模索することにした。

2.2.2 手法

略称・定義規定文は大きく分けて 2 つの形に分けられる。中村らの論文ではこれらをトイウ形・ライウ形と分類しており、本稿でもこの呼び名を採用する [7]。

トイウ形は「A、B 及び C（以下「〇〇〇」という。）」という形で書かれるものである。このとき、「A、B 及び C」が正式名称で「〇〇〇」が略称となる。具体例としては会社法（平成十七年法律第八十六号）第二十二条の以下のものを挙げる。

事業を譲り受けた会社（以下この章において「譲受会社」という。）が譲渡会社の商号を引き続き使用する場合には、その譲受会社も、譲渡会社の事業によって生じた債務を弁済する責任を負う。

この例では正式名称が「事業を譲り受けた会社」で、「譲受会社」が略称である。

ライウ形は「〇〇〇（A、B 及び C をいう。以下同じ。）」という形で書かれるものである。このとき、「A、B 及び C」が正式名称で「〇〇〇」が略称となる。具体例としては会社法（平成十七年法律第八十六号）第二百五十五条第二項の以下のものを挙げる。

証券発行新株予約権付社債に付された新株予約権の譲渡は、当該証券発行新株予約権付社債に係る新株予約権付社債券を交付しなければ、その効力を生じない。ただし、自己新株予約権付社債（株式会社が有する自己の新株予約権付社債をいう。以下この条及び次条において同じ。）の処分による当該自己新株予約権付社債に付された新株予約権の譲渡については、この限りでない。

この例では正式名称が「株式会社が有する自己の新株予約権付社債」で、「自己新株予約権付社債」が略称である。

トイウ形の略称とライウ形の正式名称は正規表現で簡単に抽出することができるが、トイウ形の正式名称とライウ形の略称部分の判定と抽出は自明ではない。特に正式名称部分に読点が含まれる場合や文頭から始まらない場合があり、これらの考慮が難しくなっている。例えば、青少年が安全に安心してインターネットを利用できる環境の整備等に関する法律（平成二十年法律第七十九号）第三条第一項は以下のような条文である。

青少年が安全に安心してインターネットを利用できるようにするための施策は、青少年自らが、主体的に情報通信機器を使い、インターネットにおいて流通する情報を適切に取捨選択して利用するとともに、適切にインターネットによる情報発信を行う能

力（以下「インターネットを適切に活用する能力」という。）を習得することを旨として行われなければならない。

この条文において「インターネットを適切に活用する能力」という略称に対応する正式名称は「青少年自らが、主体的に情報通信機器を使い、インターネットにおいて流通する情報を適切に取捨選択して利用するとともに、適切にインターネットによる情報発信を行う能力」である。このようなものを正規表現で抽出することは不可能である。

略称・定義規定文を含む条文の構造は「○○○△△△(〜〜)×××」のような形となっているということができる。トイウ形では「△△△」が正式名称で「〜〜」が略称に、ヲイウ形では「△△△」が略称で「〜〜」が正式名称にあたる。今解決しなければならないのは、「○○○」と「△△△」の境界がどこにあるのかを自動で判定することである。

本研究では係り受け解析を行うことでこの境界を自動で判定する手法を考案し、試した。条文のテキストを係り受け解析することで得られる、「どの単語がどの単語に係っているか」という関係の有向グラフを解析し

1. 括弧書き（「(〜〜)」の部分）の直前の単語に係る語の最長部分
2. 括弧書き（「(〜〜)」の部分）より後に書かれている語（「×××」の部分）に係っている単語が一切出現しない連続する部分

の2つの情報を得る。求める略称・正式名称に対して、1番目の情報は「この部分の中に求める分は含まれる」という部分で2番目の情報は「この部分は求める分に含まれる」という部分になっていると考えた。そこで、1番目の部分が句読点を含まない場合には1番目の部分を採用し、句読点を含む場合には2番目の部分を採用するというルールで抽出を行ってみた。これは、文の区切りが存在しない場合には考えられる中で最長な部分を採用すると確からしいだろうという予想に基づくものである。

2.2.3 ソフトウェアの実装と結果

日本語の係り受け解析部分の実装はPythonで行った。Pythonの自然言語処理ライブラリであるspaCyに基づいて作成された日本語用の自然言語処理ライブラリであるGiNZA^{*3}を使用するためである。これは事前学習モデルを導入した高性能な自然言語処理ライブラリである[8]。このライブラリを使用して条文を文節ごとにくぎり、その文節がどこに係っているのかの情報を取得し、JSON形式で吐き出すソフトウェアを作成した。これはMITライセンスの元GitHubで公開している^{*4}。

上記の係り受け解析を行うソフトウェアに対して渡す条文のテキストの生成や、係り受け解析の結果を元に略称・正式名称を取り出すソフトウェアはRustで実装した。これはMITライセンスの元GitHubで公開している^{*5}。

全法令のデータに対して解析を行ったところ、128100個の定義を抽出することができた。

生成された略称・正式名称と出現箇所の情報はJSON形式で、CC0ライセンスの元

^{*3} <https://github.com/megagonlabs/ginza>

^{*4} https://github.com/japanese-law-analysis/parse_japanese_dependency

^{*5} https://github.com/japanese-law-analysis/analysis_ryakusyou

GitHub で公開している^{*6}。

2.2.4 考察

概ね定義文を抽出することができたものの、略称・正式名称の判定が間違っている箇所が少なからずあった。例えば、上に挙げた

青少年が安全に安心してインターネットを利用できるようにするための施策は、青少年自らが、主体的に情報通信機器を使い、インターネットにおいて流通する情報を適切に取捨選択して利用するとともに、適切にインターネットによる情報発信を行う能力（以下「インターネットを適切に活用する能力」という。）を習得することを旨として行われなければならない。

という文言に対しては、略称の「インターネットを適切に活用する能力」は正しく抽出できているものの、正式名称の部分は「自らが、主体的に情報通信機器を使い、インターネットにおいて流通する情報を適切に取捨選択して利用するとともに、適切にインターネットによる情報発信を行う能力」と判定して抽出してしまっており、先頭の「青少年」という文言が抜けている。

係り受け解析の結果をもとにした解析の精度は高そうであるが、精度をあげる余地がかなりあると感じている。品詞の情報も考慮に加えたり、係り受け解析の際に使用する辞書を法令文書用のものにしたりすることで、精度を上げられるのではないかと考えている。

2.3 法令名の自動抽出

2.3.1 背景

法令や判例からは他の法令やその中の条項を参照する事がとても多い。これは、法令では他の法令の規定の補足や条件の上書きなどを行い、判例では論拠を積み上げていく過程で実際の法令の条文に当たる必要があるからである。参照先の条文の内容を自動で取得できたり、条文間の依存関係を明らかにして影響範囲を調べることができたりするなど、条項への参照を抽出することはとても有意義である。

条項についての参照表現の抽出方法については木村によって分類され、形式的な抽出が可能なが示されている [3]。法令名および法令番号の自動取得についてはハッシュ法を用いた方法が関本によって提案されている [9]。

しかし、法令名については条文中で定義された略称によって参照されることもある。例えば郵政民営化法（平成十七年法律第九十七号）の第八十三条では冒頭では以下のような記述がある。

郵便局株式会社の成立の際現に公社が郵政民営化法等の施行に伴う関係法律の整備等に関する法律（平成十七年法律第百二号。以下「整備法」という。）

「郵便局株式会社の成立の際現に公社が郵政民営化法等の施行に伴う関係法律の整備等に関する法律」という長い名前の法律に対して「整備法」という略称を定義している。実際に、そのあとの条文である第百七条第一項の口では「整備法附則第五条第一項の規定

^{*6} https://github.com/japanese-law-analysis/data_set/blob/master/law/ryakusyou.json

により」のように、「整備法」という略称が使われて条文を参照している。正確な参照を分析するためにはこの略称も含めた解析ができるようにする必要がある。今回は、これを行うための前処理として法令中で定義されている略称を自動で抽出することとした。

また、国が「略称法令名一覧」という web ページで事前に定義された略称の一覧を公開している（図2）^{*7}。ここに定義された略称は断りなく使用されることが考えられるため、これの一覧も取得する必要がある。

2.3.2 手法

略称の定義文は例にあげたように「＜法令の正式名称＞（＜法令番号＞。以下＜条件文＞において「＜略称＞」という。）」という形式で、

- 法令の正式名称
- 法令番号
- 条件文
- 略称

の4つの情報を含んで定義される。このとき、条件文とは例えば「以下次項及び第六項において」のような、略称が使われる条項を制限する文である。この文が書かれなくてもあり、その時は定義された後、同じ法令内であればいつでも使えるようになる。このうち、法令番号がわかれば法令の正式名称は自動で割り出すことができるため、抽出しなければならない情報は

- 法令番号
- 条件文
- 略称

の3つとなる。この3つの情報を含んだ文を表す

```
1 ((明治|大正|昭和|平成|令和)(一|二|三|四|五|六|七|八|九|十|〇)+年[^()、。↵
↵ あ-ん]+第(一|二|三|四|五|六|七|八|九|十|百|千|〇)+号。?(以下)?([^(↵
↵ ]+において)?、?「[^']* (法|令)」という。
```

という正規表現を作成し、これにマッチする文から上記の3つの情報を抜き出すこととした。

「略称法令名一覧」の web ページからのデータを抜き出す際には、web ページにアクセスし HTML データを解析することで情報を得ることとした。

2.3.3 ソフトウェアの実装

手慣れていることやすでに自身が作成した法令解析用のライブラリを利用することから、ソフトウェアの実装は Rust 言語で行った。作成したソフトウェアは MIT ライセンスの元 GitHub で公開している^{*8}。

「略称法令名一覧」ページのスクレイピングソフトウェアも Rust で実装した。作成

^{*7} <https://elaws.e-gov.go.jp/abb/>

^{*8} https://github.com/japanese-law-analysis/analysis_law_reference

略称法令名一覧

登録略称法令名一覧

正式法令名	法令番号	略称法令名 1	略称法令名 2	略称法令名 3	略称法令名 4
明治九年太政官布告第二百七十四号（改訂ノ布告）	明治九年太政官 改訂の布告 布告第二百七十四号				
明治八年太政官布告第五百十四号（勅令制定ノ件）	明治八年太政官 勅令制定の件 布告第五百十四号				
明治八年太政官布告第五百十四号（勅令制定ノ件）	明治八年太政官 勅令制定の件 布告第五百十四号				
明治十年太政官布告第九十三号（大勅令制定ノ件）	明治十年太政官 大勅令制定の件 布告第九十三号				
明治十九年勅令第百一十一号（本邦ノ主権保護ニ関スル法律ノ件）	明治十九年勅令 本邦ノ主権保護ニ関スル法律ノ件 第百一十一号				
明治二十二年法律 行政法	明治二十二年法律 行政法 第百一十一号				
明治二十一年法律第百一十一号（行政ノ件）	明治二十一年法律 行政ノ件 第百一十一号				
明治二十一年法律第百一十一号（行政ノ件）	明治二十一年法律 行政ノ件 第百一十一号				

図 2：略称法令名一覧という web ページ

したソフトウェアは MIT ライセンスの元 GitHub で公開している^{*9}。

2.3.4 結果

現在施工されている法令中全体に対して解析をかけたところ、3504 個の法令から合わせて 11193 の略称の定義を抽出することができた。抽出された略称一覧の JSON ファイルは CC0 ライセンスの元 GitHub で公開している^{*10}

「略称法令名一覧」の web ページで公開されているものについては、2412 個の法令について合わせて 3276 個の略称が定義されていた。抽出された略称一覧の JSON ファイルは CC0 ライセンスの元 GitHub で公開している^{*11}

2.3.5 考察

略称定義文を正規表現を使って解析することにより、合計で 1.1 万を超える略称のリストを生成することができた。この数は国が公開している略称一覧よりも 3 倍ほど多い。これにより、既に提案されている法令名抽出方法の手法や Aho-Corasick 法などの既知の効率的な文字列探索アルゴリズムを使用することで、より精密な法令名の出現箇所の解析を行うことができると期待される。

ただし、条項を含めたものについて略称を定義している場合の対応ができていないため、これについてどう略称情報を取得してデータを保持するのかを検討する必要がある。条項を含めたものについての略称とは、例えば恩給法（大正十二年法律第四十八号）の第二十四条の三第一項では「旧恩給法の特例に関する件の措置に関する法律による改正

^{*9} https://github.com/japanese-law-analysis/egov_law_abb

^{*10} https://github.com/japanese-law-analysis/data_set/blob/master/law/law_abb.json

^{*11} https://github.com/japanese-law-analysis/data_set/blob/master/law/egov_abb.json

前の旧勅令第六十八号第八条第一項（以下「改正前の旧勅令第六十八号第八条第一項」という。）に規定する」と書かれており、法令名に加え「第八条第一項」まで指定したうえで略称を定義している。

2.4 判例の自動収集およびテキストデータ化

2.4.1 背景

法律分野の文書の中では法令だけではなく判例も重要である。法令が埋められない穴を判例によって埋めていき、次の類似事例の時の判断に活かされるからである。そのため、判例の解析も法令と同様に行う必要がある。

現状、判例のテキストがオープンデータとして公開されているものが無いため、作成することとした。

2.4.2 手法

裁判所は「裁判例検索」という Web サービスを提供している^{*12}。このサービスは単語検索や裁判年月日、事件番号等で該当する裁判例を検索し、その裁判の

- 事件番号
- 法廷名
- 種別
- 判示事項
- 裁判要旨
- 判決文全文

などが記載された個別の詳細ページへのリンクを取得することができるサービスである。ここで、判決文全文は PDF ファイルであるが、それ以外は単純な HTML によって提供される。

判例の信頼性と構築の容易さから、今回はこの裁判例検索サービスを利用して判例データベースを作成することとした。

情報の抽出は主にスクレイピング技術を使用することで行った。まず、このサービスに対して取得したい期間を表す検索クエリが含まれた URL を自動で生成してアクセスし、詳細ページのリンクのリストを取得する。その後、詳細ページの HTML を解析して必要な情報を抜き出し、判決文全文の PDF ファイルをダウンロードする。最後に、その PDF ファイルからテキストを抽出することとした。

2.4.3 ソフトウェアの作成

並列並行計算が高速で安全に行うことができ、自身の手になじんでいるため、スクレイピングソフトウェアを Rust で実装した。作成したソフトウェアは MIT ライセンスの元

*12 https://www.courts.go.jp/app/hanrei_jp/search1

GitHub で公開している^{*13}。判例の一覧は JSON 形式で出力される。

判例 PDF ファイルからテキストの抽出を行う方法は 2 通り用意した。それぞれ

- poppler ライブラリ^{*14}が提供する pdftotext コマンドを用いてテキストを抽出した後に、ページ番号や行番号に相当する部分を削除する
- PDF ファイルを画像ファイルに変換した後に、本文テキストが書かれた部分のみを切り出し、tesseract^{*15}という OCR ソフトウェアによってテキストを抽出する

という方法である。作成したソフトウェアは MIT ライセンスの元 GitHub で公開している^{*16}。

2.4.4 結果

判例データの収集は公開されている中で最も古い 1926 年から実験を行った 2023 年 1 月 15 日までの期間で行った。その結果約 7 万件の判例データを収集することができた。生成された判例に関するデータは CC0 ライセンスの元 GitHub で公開している^{*17}。

PDF ファイルからテキストを抽出する手法は、pdftotext コマンドを使用する方が OCR を行うよりも精度が高かったため、pdftotext を使用する方法を採用した。生成されたテキストファイルは CC0 ライセンスの元 GitHub で公開している^{*18}。ただし、文字の抜けや行の順番のズレなどが存在することを確認しており、よりテキスト抽出の精度を上げる必要があると考えている。

2.4.5 発見した誤記・例外について

この収集および解析の過程で年月日にマイナスの値が出現している詳細ページが以下の 7 つ存在することを確認した（図3）。

- 昭和 46(あ)1051^{*19}

*13 https://github.com/japanese-law-analysis/listup_precedent

*14 <https://poppler.freedesktop.org/>

*15 <https://github.com/tesseract-ocr/tesseract>

*16 https://github.com/japanese-law-analysis/pdf2txt_precedent

*17 https://github.com/japanese-law-analysis/data_set/blob/master/precedent/list.json

*18 https://github.com/japanese-law-analysis/data_set/tree/master/precedent/text_files

*19 http://web.archive.org/web/20220606192511/https://www.courts.go.jp/app/hanrei_jp/detail2?id=51011

- 昭和 39(オ)1302 ^{*20}
- 昭和 35(オ)535 ^{*21}
- 昭和 36(オ)513 ^{*22}
- 昭和 36(オ)536 ^{*23}
- 昭和 34(オ)71 ^{*24}
- 昭和 24(レ)2055 ^{*25}

発見した翌日である 2023 年 1 月 16 日に最高裁判所に報告を行ったところその数日後に全てのページが修正された。例えば昭和 39(オ)1302 という判例では

- 修正前：https://web.archive.org/web/20230116041527/https://www.courts.go.jp/app/hanrei_jp/detail2?id=66298
- 修正後：https://web.archive.org/web/20230118092109/https://www.courts.go.jp/app/hanrei_jp/detail2?id=66298

がそれぞれ URL のページであるが、タイムスタンプを確認すると修正前が 2023 年 1 月 16 日で修正後が 2023 年 1 月 18 日となっており、報告を受けた翌日に修正が行われていることがわかる。

また、裁判例に一意に割り振られる事件番号というものが割り振られていないで空白になっている事件が 2 つ存在することを確認した。それぞれ

- 昭和 23 年 10 月 13 日判決の「市税滞納処分取消請求事件」 ^{*26 *27}

^{*20} https://web.archive.org/web/20230115065439/https://www.courts.go.jp/app/hanrei_jp/detail2?id=66298

^{*21} https://web.archive.org/web/20230115072821/https://www.courts.go.jp/app/hanrei_jp/detail2?id=63595

^{*22} https://web.archive.org/web/20230115073312/https://www.courts.go.jp/app/hanrei_jp/detail2?id=63715

^{*23} https://web.archive.org/web/20230118092926/https://www.courts.go.jp/app/hanrei_jp/detail2?id=63719

^{*24} http://web.archive.org/web/20200804085713/https://www.courts.go.jp/app/hanrei_jp/detail2?id=70761

^{*25} http://web.archive.org/web/20230118094001/https://www.courts.go.jp/app/hanrei_jp/detail2?id=70761

^{*26} http://web.archive.org/web/20230118094715/https://www.courts.go.jp/app/hanrei_jp/detail3?id=22208

^{*27} http://web.archive.org/web/20230118094829/https://www.courts.go.jp/app/hanrei_jp/detail4?id=22208

原審裁判所名	仙台高等裁判所
原審事件番号	
原審裁判年月日	<u>昭和-954年3月25日</u>

図3：昭和-954年という表記（下線筆者）

原審裁判所名	仙台高等裁判所
原審事件番号	
原審裁判年月日	<u>昭和46年3月25日</u>

図4：昭和-954年という表記が修正されたあとのページ（下線筆者）

- 平成17年6月20日判決の「新株予約権発行差止仮処分決定認可決定に対する保全抗告(付更正決定)」^{*28}

である。事件番号がついていないことについて最高裁判所に問い合わせたところ、「何らかの事情によってつかないこともあり、それではないか」との回答を得た。

3 専門家らからの意見

前年度の成果報告会にて「法学分野の人からの意見が聞きたい」と述べたところ、当時の情報科学類長の伊藤誠先生より弁護士の先生や法学の研究者の方々を紹介していただき、意見を頂く機会を得ることができた。また、個人的にも建設会社の方との雑談でいくつか意見を頂くことができた。そこで頂いた意見をもとに本プロジェクトで今後行いたいことを次のように整理した。

- 関連する条例や、国会や省庁での本会議・委員会・勉強会での議事録の検索

^{*28} http://web.archive.org/web/20230118094842/https://www.courts.go.jp/app/hanrei_jp/detail4?id=9

- 法令以外にも各地域ごとの条例によって異なる制限が課されることがあり、それぞれへの対応を自動化したいというモチベーションがある
- 法令等の成立の過程における意見を踏まえて判断する必要がある
- 条文中のただし書きの把握
 - 「〇〇条ただし書き」という形で参照されることがある
- 分野ごとに存在する慣習的な略記や特殊用語への対応
- LLM 用のデータの整備
- 知的財産権系の文書の解析・生成
- 地方自治体向けの規則作成支援ツールの作成

これらを踏まえ、本年度ではまず、条例も含めて統一的に法律文書を取り扱うためのインターフェースを用意し、多種多様な法律文書に対応することにした。

4 法律文書解析用ライブラリの整備

4.1 法令標準 XML スキーマライブラリの作成

日本政府が法令のデータを公開している e-Gov 法令検索 [5] では、公開データの形式を「法令標準 XML スキーマ」^{*29}で公開している。日本政府が数多くの法令を調査した上で包括的に法令を取り扱うために整備したデータ構造であるため、法令データを統一的に扱うデータ構造としてはこれをベースとすることが適切であると考えた。

本プロジェクトで主に使用しているプログラミング言語である Rust のためのライブラリを作成することとした。機能として次のものを実装した。

- スキーマに則った XML 文字列から事前に定義した struct などのデータ構造への変換
- 事前に定義した struct などのデータ構造からスキーマに則った XML 文字列への変換
- 法令文テキストから事前に定義した struct などのデータ構造への変換

これにより、条例や規則と法令を一つのインターフェースで取り扱うことができるようになった。

作成したライブラリは Rust のパッケージマネージャに登録し^{*30}、現在累計で 1500 回ほどダウンロードされている。

4.2 支援ライブラリの作成

法令標準 XML スキーマだけではカバーすることができない解析結果を表す型や法律などのメタデータを表す型などを定義したり、解析ツールを作成する上であると便利な関数を定義したライブラリを作成した^{*31}。これにより、解析で得られた大量のデータ

^{*29} https://elaws.e-gov.go.jp/file/XMLSchemaForJapaneseLaw_v3.xsd

^{*30} https://crates.io/crates/japanese_law_xml_schema

^{*31} https://github.com/japanese-law-analysis/jplaw_tools



図 5：法令標準 XML スキーマライブラリのページ

をこのライブラリを通じて他のユーザも簡単に扱うことができるようになり、解析結果を組み合わせてソフトウェアを作りやすくなることが期待される。

また、前年度までに実装した関数の実装のアップデートも行った。2.4.4節で述べている通り、PDF からのテキストの抽出とページ番号等の除去の精度に改善の余地があった。そこで今回行番号とページ番号の除去の精度が向上するように実装の改良を施した。

5 条例データの整備

3節で述べた通り、実務を行う上で条例の及ぼす影響は大きいため、法令と同時に条例も同時に解析する必要がある。自治体の条例や規則を収集したデータベースとして「条例 Web アーカイブデータベース」というものが原田らによって整備され公開されている(図6) [10, 11]。今回はこれを利用し、公開されている例規を手元にダウンロードし、メタデータを取り出すソフトウェアを作成した^{*32}。今後は、今回作成したテキストから法令標準 XML スキーマの形式に変換するライブラリをしようし、法令と同様に解析を行い、結果をまとめることを考えている。

6 国会議事録・委員会議事録のデータ化についての調査

法令の解釈ではその法令が制定・改正されたときの議論や文脈を元にすることが多く、

^{*32} https://github.com/japanese-law-analysis/listup_jorei

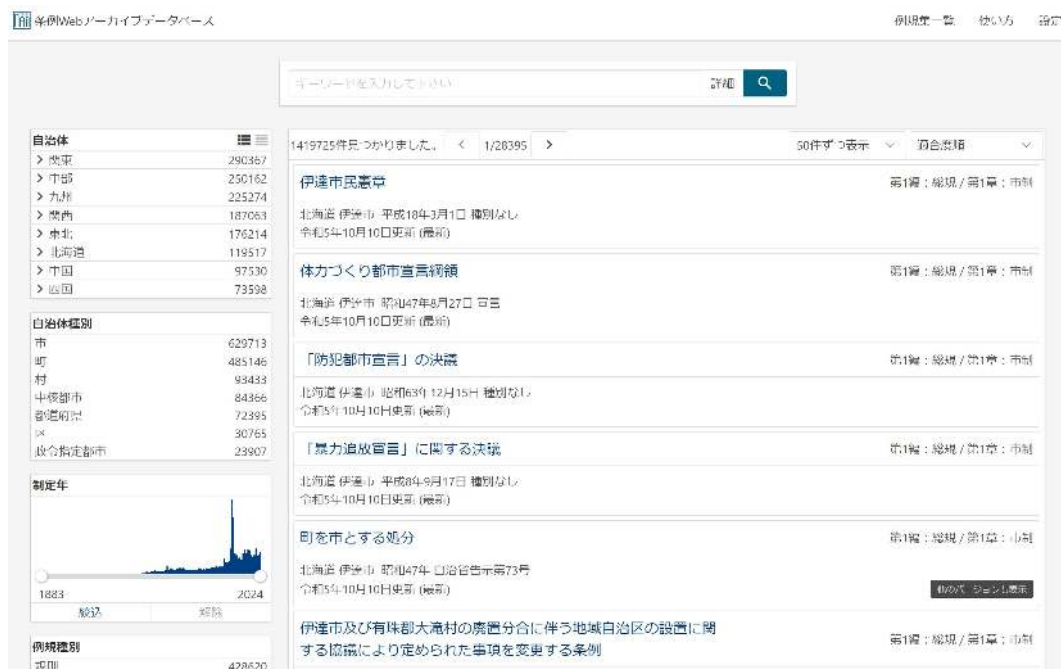


図 6：条例 Web アーカイブデータベースのページ

3節で指摘を受けている通りこれらの議事録のデータも収集することが大事となる。

国会と帝国議会の議事録はそれぞれ国立国会図書館がアーカイブを公開している（図 7-8）[12, 13] ため、これを利用してデータを収集することはできる。しかし、国会の議論は法令の成立に関連しない議論も多いため、これをどのようにスクリーニングするかが検討課題となっている。

委員会議事録も国立国会図書館の同アーカイブに保管されているため、データを収集することは同様に可能である。また、どのように必要な情報をスクリーニングするかという課題も同様に存在しているが、委員会の方がより専門的な内容を取り扱っていることが期待できるという違いもある。

7 読替規定文解析の改善

2.1.3節で述べた通り、昨年度までの解析手法では

- 文中の括弧の対応が壊れているもの
- 表形式で与えているもの
- 読み替える語等の指定に鍵括弧をしていないもの

の 3 種類には対応できていなかった。そこで本年度ではこれらへの対応を考えたこととした。鉤括弧をしていないものへの対応は今回行うことができず、カッコの対応が壊れているものについては現実的な時間での解析が行えず試行錯誤を行っている過程である。表形式で与えるものについては

- 下欄に掲げる字句と読み替える
- 下欄の字句と読み替える
- 下欄に掲げる日又は月と読み替える

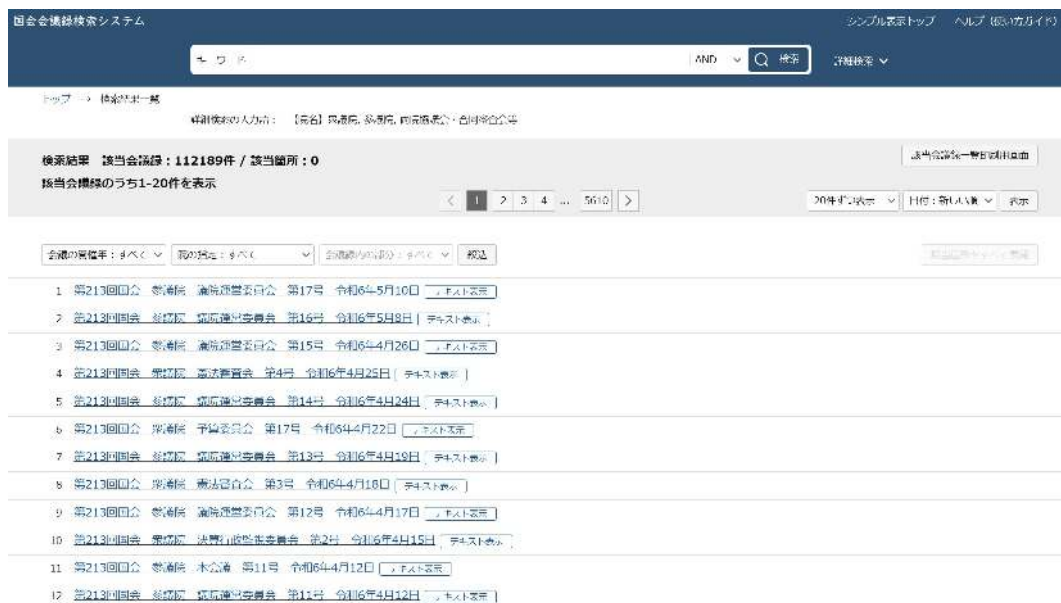


図 7：国会議事録データベースのページ

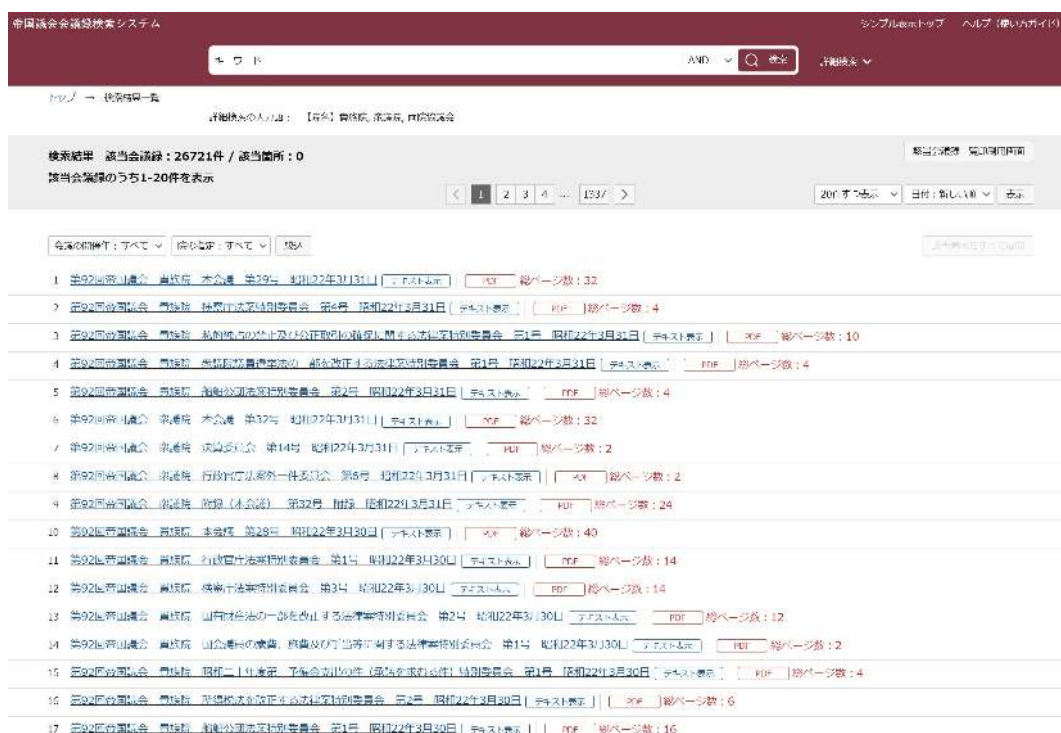


図 8：帝国議会議事録データベースのページ

という文言のいずれかが出現するときであることが調査の結果わかったため、これを元に直後に出現する表の左右を読み替えられる語と読み替える語に振り分けることで対応することができ、解析の成功率が向上した。

8 おわりに

本年度では主に法律文書解析のための統一的なインターフェースの用意に時間を要した。しかし、そのおかげで条例の解析のための準備がスムーズに行えた。また、前年度までの成果の改良なども行った。

今後は他の法令への参照の抽出を条例レベルで行ったり、判例テキストの解析や LLM 用のデータの構築などに取り組みたいと考えている。

謝辞

本プロジェクトを進めていくうえで産学間連携推進室のメンバーや山本幹雄先生、川見唯史弁護士、森田岳人弁護士、中川由賀先生（中京大学）、中山幸二先生（明治大学）、伊澤文平弁護士等の皆様よりアドバイスを頂きました。また、産学間連携推進室の担当をしてくださっている新城靖先生や弁護士や法学の研究者の方々を紹介していただいた伊藤誠先生にもお世話になりました。この場をお借りしまして、深く感謝を申し上げます。

参考文献

- [1] 竹井 直樹, 柴田 文明. 保険約款と保険商品のわかりやすさの向上について ― 最近の損害保険業界の取り組みと各損害保険会社の商品改定から考察する ―. 損害保険研究. 2010, 72 巻, 2 号, pp.109-127
- [2] 信岡俊祐, 中村誠, 島津明. 法例文の論理式への変換. 言語処理学会第 13 回年次大会発表論文集. 2007, pp.254-257
- [3] 木村祐介. 号の列挙や参照表現をもつ法令文への論理式への変換. 北陸先端科学技術大学院大学情報科学研究科情報処理学専攻修士論文. 2008.
- [4] 磯崎陽輔. 分かりやすい法律・条例の書き方. りょうせい, 2011.
- [5] 日本政府. “e-Gov 法令検索”. e-Gov ポータル. <https://elaws.e-gov.go.jp/>, 2024 年 5 月 16 日確認.
- [6] 金子尚樹. 単語の出現規則に着目した法令中の読み変え規定文の解析手法の提案. 情報処理学会第 85 回全国大会論文集 (2). 2023. pp.729-730.
- [7] 中村誠, 小川泰弘, 外山勝彦. 法令文中において括弧書きで定義されている法令用語とその語釈文の抽出. 言語処理学会第 19 回年次大会発表論文集. 2013, pp.670-673
- [8] 松田寛. GiNZA - Universal Dependencies による実用的日本語解析. 自然言語処理 Volume 27 Number 3. 2020
- [9] 関本大樹. 租税法令の自動機械処理に関する一つの試み: ハッシュ法に基づく条項名の実用的な抽出方法について. 久留米大学法学 79. 2019. 180-147.
- [10] 条例 Web 作成プロジェクト. “条例 Web アーカイブデータベース”. <https://jor-ei.slis.doshisha.ac.jp/>, 2024 年 5 月 16 日確認.
- [11] 原田 隆史, 川島 隆徳. [C31] 地方自治体が公開する例規集アーカイブの構築と横断検索システムの構築. デジタルアーカイブ学会誌. 2018, 2 巻, 2 号 pp.132-135
- [12] 国立国会図書館. “国会会議録検索システム”. <https://kokkai.ndl.go.jp/>, 2024 年 5 月 17 日確認.
- [13] 国立国会図書館. “帝国議会会議録検索システム”. <https://teikokugikai-i.ndl.go.jp/>, 2024 年 5 月 17 日確認.

産学間連携推進室主催の学内向けイベントについて

筑波大学 情報学群 情報科学類 産学間連携推進室
情報科学類 3 年 202212117 金子尚樹 [s2212117@u.tsukuba.ac.jp]

1 はじめに

2023 年から 2024 年に学内向けのイベントを複数開催したため、その開催について報告する。

2 ネットワークの構築講座

2023 年 4 月に複数の友人から「基本情報技術者試験のネットワーク分野の問題を見て概念として理解しても、実感を伴って理解できている気がしない」といった話を聞いていたため、実際に目の前にある複数のコンピュータをケーブルで繋いで設定を弄って比較実験を行える場を用意すると良いのではないかと考え、企画した。

取り扱う内容は大まかに

- 複数の PC を繋ぎ合わせて IP アドレスを振り、お互いに通信させられるようになる
- サブネットマスクで各 PC 間の通信を制御できるようになる
- スイッチを使って各 PC 間の通信を制御できるようになる

の 3 つとした。この 3 つができれば簡単なネットワークを構築できるようになると考えたからである。

機材は Ubuntu をインストールした PC を 4 つとスイッチを 1 つ用意した。

1. ネットワークの概要といわゆるインターネットの説明をスライドを用いながら講義スタイルで行った
 2. 2 つの PC を LAN ケーブルで接続して IP アドレスを振り、サブネットマスクの値の違いで通信できるかどうかが変わることを確認した
 3. 1 台の PC をルーター化し、それを挟んで 2 台から 3 台の PC 間で通信させてルーターの役割とルーティングテーブルの役割を確認した
 4. スイッチに 4 台の PC を繋いで相互に通信できることを確認し、次にスイッチのコンフィグでグルーピングの設定を行って挙動が変わることを確認した
 5. ネットワーク関連のイベントの紹介をスライドを用いながら行った
- という手順で講座を行った。また、同時開催で 100G の光ファイバモジュールの展示も行った。イベント後にアンケートを実施し、フィードバックを得た。



図 1：事前準備 (1)



図 2：講義スタイルでの解説の様子



図 3：講座の様子



図 4：講座の様子

参加人数とその所属は 2 日間で合計して表1の通りであった。情報学群の学生に多く参加してもらったほか、情報科学類に興味のある総合学域群の学生の参加も一定数あったことを確認した。

表 1：ネットワーク構築講座の参加者の所属とその人数

学類	人数
情報科学類	17
情報メディア創成学類	8
知識情報・図書館学類	2
応用理工学類	1
総合学域群 1 類	1
総合学域群 2 類	2
総合学域群 3 類	5
情報理工学位プログラム	4
合計	40

参加者のイベント参加前のネットワーク構築の経験は表2、参加後についての知識や経験は表3のとおりであった。参加者のうちネットワーク構築の経験が乏しかった人が四分の三ほどを占めていたが、講座後には逆にネットワーク機器の扱い方などがわかる人が参加者のうち四分の三を占めるようになった。講座の意義があったと評価できる。

また、この講座に対して寄せられた感想を下に紹介する。

表 2：参加前のネットワーク構築の経験

有線 LAN を使ったことがある	33
PPPoE や IPoE などの Wi-Fi の設定をしたことがある	8
ルーターの設定をしたことがある	10
複数の PC を運用したことがある	12
サーバーを設置したことがある	14
サブネットマスクや VLAN を用いてネットワークを分割したことがある	3
ネットワークの構築方法についてテキストのみで学習をしたことがある	11
無回答・該当なし	5

表 3：参加後についた知識や経験

「ネットワーク」とはどういうものかわかった	34
「インターネット」とはどういうものかわかった	33
PC 間で通信させている様子がわかった	34
複数の PC 同士でネットワークを構築する方法が分かった	33
サブネットマスクの使い方がわかった	32
スイッチやルーターの役割がわかった	31
スイッチの設定で複数のネットワークを制御する方法がわかった	27
無回答・該当なし	3

- 光ファイバを見れる機会は貴重でした。
- スライドの例えがわかりやすくて講義はスツと入ってきました。実際に手を動かすことで講義の内容を自分のものにできたような感じがあり、楽しくネットワークを学びました。スイッチやルーターの役割は説明だとあまり分かった気になれなかった。たくさんさんの PC でネットワークを構築するのに便利？ Interop Tokyo と JANOG の話、ワクワクしました。いいですね。
- ルーターの語源が分かっておもしろかった。サブネットマスクなど今まできいたことないこともしれた。
- 実際に触ってみて、ネットやテキストで知る以上の実感が得られました。ありがとうございました。
- サブネットマスクやスイッチを使ってネットワークを分割する方法をしれて、とても面白かった。
- 光ファイバーの中を初めて見た。
- 非常に面白かったです。説明されなかったナゾの物が気になってますが……
- 普段、web の開発などで適当に localhost にサーバーを立てていたが、その仕組みや、ネットワーク全体の仕組みについて理解することができてよかった。
- とても楽しかったです！マイコンとか中古機を買って自分の家でやってみます！！
- めちゃくちゃ面白かったです。
- ネットワーク IP とホスト部の区切りが分かりづらい。図がほしい。
- 自分には少し難しい内容だった。他の参加者の方々は理解しているようだったので

ごいと思った。

- 全体の雰囲気があったので、これからの勉強がやりやすくなったと思います。
- 多くの新しい知識を得られました。自分のような無知の人間のために PC の設定がどうなっているのかというのをホワイトボード等でリアルタイムで書いてくれると助かります。
- 実際にコンピューターを動かしているとき、「この PC の IP アドレスなんだっけ？」と何度か思ったので PC の側に書いておいてあると助かります。
- 全くの初心者でしたが、とても興味深い話を聴かせていただき、ありがとうございます。とても面白かったです。
- 実際にコンピューターを動かしているとき、「この PC の IP アドレスなんだっけ？」と何度か思ったので PC の側に書いておいてあると助かります。
- おもしろかったです。ただ、内容と実演内容（パソコンの小さい画面でやる奴）に対して人数が多すぎた気がします …
- 理解度としてはまだ低いところが多いが、ネットワーク全体に興味を持てた。今後もうこういった講座を開いてもらえると幸いです。
- 研究に関連しそうなので中古のネットワーク機器を買って色々しようと思っていたが、その良い足掛かりになった。実際にやっている所を見れたので、自分でもできる気がして良かった。
- 知らないことばかりで面白かった。実際に眼の前で動いているのが見えてよかった。自分でも色々用意していじってみたい。
- 仮説をもとに実際に実験をしておもしろかった。
- やっぱり実機で動いているのを見るのは楽しかった。
- とてもおもしろかった。自分がこれからしてみたいことを知れた
- 講演の熱量が高くてよかった。
- ラックに入るようなスイッチを動かしている様子が見れておもしろかった。インターネットの技術は奥が深く、学内にもたくさん技術をもった人がいることがわかった。もっと勉強をして、技術を身につけたい。

3 PC 分解大会

新入生オリエンテーションの後に coins ラウンジで産学間連携推進室にあった不要な PC を新入生に好きに分解してもらうイベントを行なった。大学に入るまでに PC を組み立てることはあっても好きに分解できる機会は通常無いため、新入生にとって貴重な体験ができ高校までとは違う自由さや面白さを感じられる場になると考え、企画した。

新入生オリエンテーションで産学間連携推進室の紹介を行なった際に告知を行ったところ、20 名ほどの新入生が参加してくれた。

内容としては

1. 一つの PC の蓋を外し、メモリや CPU などの基本的な部品について説明し、外し方を解説する
2. 軍手やドライバを渡し、好きに分解してもらう
3. 分解の過程で生まれた疑問に対しては産学間連携推進室のメンバーがその場で回答する

という順に行った。生まれた疑問の解説の際には補足資料を追加で持ってきて対応する

ことや、安全に試すための特別な対応をすることもあった。具体的なものとしては

- 水冷モジュールを分解したい
 - ビニール袋と紙皿を別で用意し、中の液体が飛散しないようにした
- HDD を分解して中を見てみたい
 - 産学間連携推進室にあるフロッピーディスクを持ってきて記録方法の解説と違いの比較を行った
- GPU や Ethernet 用の拡張モジュールの他の例が知りたい
 - 産学間連携推進室にある光ファイバー用のモジュールを持ってきて具体的に差し込む方法などを体験してもらった

などがあった。

参加した人たちがそのあとに SNS で呟いていた感想として

- 筑波大学楽しい
 - 情報科学類楽しい
 - AC 入試以外の人も目を輝かせながら分解していて良かった
- などがあり、当初の目的である
- PC をどこまでも分解する機会を提供する
 - 大学の面白さを感じられる場を提供する
- の 2 つを、どちらも達成できたと評価している。



図 5：全体の様子（2023 年度）



図 6：全体の様子（2024 年度）



図 7：分解している様子（2023 年度）



図 8：分解している様子（総合学域群の移行生のためのイベント）

4 おわりに

通常の生活で触れる機会が余りないであろう内容の体験型のイベントを 2 回実施した。どちらもイベントの目標を達成することができ、参加者に新しい知識と経験、そして楽しさを提供することができたと評価している。

想定よりも参加者数が多かったことによる機材の不足や準備不足などによる不備が確認できたため、より内容を洗練させていく必要があると感じた。需要がある内容であると確認できたため、今後は定期的に開催していきたいと考えている。また、これ以外にも体験型のイベントを増やしていきたい。

産学間連携推進室 成果報告書

モバイル回線を用いた管理用予備回線の構築

情報メディア創成学類 服部 真吾 (s2111559@u.tsukuba.ac.jp)

概要：本研究では、産学間連携推進室内に設置しているコンピューティング環境について、常用している外部接続がなんらかの理由で利用できなくなった際に、外部からアクセスを可能にする管理用予備回線として、モバイル回線を利用する方法について検討する。本研究の成果を用いることによって、従来固定回線として導入され、非常に高価であった管理用予備回線にかかる費用を軽減することが期待される。

1. はじめに

産学間連携推進室のような、ネットワークに関する実験を行う環境において、常用している外部接続が不通となる事象は、設定の投入ミスによって十分生じうる。このような場合に備えて、従来は月額固定制の固定回線が導入されていた。しかし、固定回線は費用が高額であり、また開通までにかかる時間も長い。そこで本研究では、モバイル回線を利用することにより、安価で取り回しのしやすい、管理用予備回線の構築方法について検討する。

2. 採用した機材

モバイル回線への接続方法として、スマートフォン等に用いられる sim カードを利用する場合と、IoT 機器専用の sim カードを利用する場合が考えられる。ここで、今回検討する対象は予備回線であり、常時通信を行うわけではなく、またなるべく廉価・低消費電力であることが求められる。そのため、IoT 機器専用の sim カードを利用する。IoT 機器向けの sim カードとしては、NB-IoT と LTE-M が代表的である。しかし、NB-IoT は 2020 年にドコモがサービスを終了するなど、日本国内で提供しているプロバイダが減少している。対して LTE-M は、多くのプロバイダから提供されているため、本研究では LTE-M を利用して、モバイル回線へ接続する。また、実際にログインして操作する機器については、入手性と拡張性が高いため、Raspberry Pi を採用し、sim カードを挿入する Dongle は、USB 接続で利用できるものを採用した。

3. 構築した環境

前述の機材を、産学間連携推進室に設置され、実際に運用されているコンピューティング環境の管理用ネットワークに接続した。常用している外部接続が不通となった場合は、まず Raspberry Pi に対して SSH 接続し、その後ネッ



図 1: 採用した機材 (同等品)

トワーク機器に対して接続することで、問題の解決を図ることができる。ネットワークへのアクセス方法として、Raspberry Pi を VPN サーバとし、レイヤー 2 の接続を外部から行うことも考えられるが、管理用ネットワークではある程度の通信が常時行われており、VPN サーバを利用すると、これら不必要なデータが通信量を増加させるため、今回は不採用とした。

4. 今後の展望

本研究は現時点で、外部からのアクセスに用いる機材の選定と設置を行なった。今後はこれらの機材について、最低限必要とされる電波環境はどのようなものであるのか、Dongle が不調となり、外部との通信ができなくなった場合、自動で復旧するためにはどのような手法が考えられるのか、実際の運用をどのように行えば、費用を最低限に抑えつつ目的を達成できるのか、調査を行う。

5. 謝辞

本研究は、ソフトイーサ株式会社との産学連携研究である。機材について、同社よりご提供いただいた。

産学間連携推進室 成果報告書

遠隔で高度なプログラミング教育を可能にする環境の開発

情報メディア創成学類 服部 真吾 (s2111559@u.tsukuba.ac.jp)

概要: 本研究では、遠隔でプログラミングを学習する中高生及び講師が共同で利用可能なコンピューティング環境と、その環境までのアクセス方法を検討・開発する。この環境は、産学間連携推進室内に設置されており、生徒及び講師はどこからでもアクセス可能である。また、生徒の主体性を損なわず、大規模なデータを解析できるよう、無制限に利用することができる。本環境を用いることによって、生徒は講師に積極的に質問し、探求的な学習・研究活動を、不必要な障害に直面することなく遂行できることが確かめられた。

1. はじめに

遠隔での実践的なプログラミング教育は、高いアクセス性と柔軟性を持っているため、地理的な制約や時間的な制約を乗り越えることができ、地域間の教育格差を解消する助けとなりうる。しかしながら、中高生を対象としてこのような教育を行うことは稀であり、実践例や研究結果が十分に蓄積されていない。また、このような教育に関する知識や経験は、企業において講師とリモートでコミュニケーションをしながら、新人研修を行う際などにも有用であると考えられる。そこで本研究では、遠隔での実践的なプログラミング教育を実施するために最適な環境、指導方法について研究する。

2. 構築した環境

遠隔かつ非同期でのプログラミング教育を可能にする環境として、Google Colaboratory[1] が挙げられる。これは、米 Google 社によって運営されている、ブラウザから Python を実行できるサービスである。このサービスでは、Google Drive と接続することによって、作成したプログラムを容易に共有することができる。しかしながら、費用がかからない範囲で行える活動には制限があり、アップロードできるデータ量、プログラムを実行できる時間を延長す

るには、追加費用が必要になる。

本研究が対象としているプログラミング教育では、天文データの分析を対象とするため、数百 GB から数 TB のデータセットを扱い、大量の CPU 時間を必要とする場合がある。この時、Google Colaboratory をはじめとした、既存の外部サービスでは、必要になる費用が膨大になる。そこで本研究では、利用するプログラミング環境を自ら構築することで、この問題を解決した。構築した環境の概要を図 1 に示す。

本環境は、グローバル IP アドレスを持つ Gateway Server と、実際にプログラムを保存し、実行する Compute Server からなる。生徒及び講師は、インターネットより Gateway Server にアクセスし、このサーバを経由して、Compute Server にアクセスする。これにより、生徒、講師の両方が、世界のどこからでも同一のリソースにアクセスすることができ、プログラムにおいてエラーが発生した場合、そのプログラムへのリンクを共有することによって、素早く修正することが可能になる。

インターネットからのアクセスにおいては、Gateway Server に SoftEtherVPN[2] をインストールして利用している。これは、将来的にグローバル IP アドレスを取得できなくなったとしても、このソフトウェアが持つ VPN Azure などの機能を利用することによって、本環境へのアクセス



図 1: 構築した環境の概要

を可能にするためである。そのほかに、SSH が利用可能である。また、専用アプリケーションをインストールすることなく本環境にアクセスできるよう、HTTP サーバの nginx[3] もインストールして利用している。

Compute Server においては、利用するプログラミング言語が Python であるため、GUI ベースの Python 実行環境である Jupyter Notebook[4] と、通常の bash の両方を利用可能にしている。まずプログラミング言語について、Python は学校の授業で扱われており、生徒は基本的な知識を持っている上、データ分析に用いるライブラリが豊富に存在しているため選択した。次に実行環境について、Jupyter Notebook はファイルの操作を GUI で行うことができ、プログラムを書いて実行するに際して、従来必要となっていた CLI に関する知識が不要であるため選択した。しかしながら、GUI のみでは、大量のファイルの効率的な操作などが困難となるため、CLI として Bash も利用可能にしている。

3. 昨年度からの変更点

昨年度までは、単一の Jupyter Notebook を起動し、これを全ユーザで共同利用していた。しかしながら、この方法では、二名以上が同一のファイルを同時に編集してしまい、実行するプログラムが期待と全く異なるものになる場合があった。関連して、データやプログラムを保存する場所を間違えてしまい、他人のデータを上書きしてしまうという問題も生じていた。また、プログラミング環境へのアクセスについて、Windows からの場合は標準の VPN クライアントアプリを、MacOS からの場合は、OpenVPN のクライアントアプリをインストールして利用していたが、普段触れることのない設定や、専用アプリケーションのインストールが必須であり、遠隔地にいる中高生が個人で設定することは困難であった。その上、VPN サーバの性能によってアクセス可能な人数が制限される、ブラウザで動作する単一の WEB ページにしかアクセスしないため、VPN サーバは過剰であるという課題もあった。

そのため本年度より、実行環境として Jupyter Hub を、インターネットからアクセスに nginx による認証つきリダイレクトを用いることで、これらの問題の解決をめざす。

まず実行環境について、Jupyter Hub とは、このソフトウェアを動かしているサーバー上に存在するユーザごとに、Jupyter Notebook を起動することのできるソフトウェアである。これにより、ユーザごとに作成されたホームディレクトリ以外にはアクセスできなくなるため、意図しないプログラムの書き換え、他人のデータの破壊を防ぐことができる。その上、ユーザが操作するソフトウェアは従来通りであるため、追加の学習コストがかからない。

次にインターネットからのアクセスについて、nginx を用いた SSL クライアント証明書による認証つきリダイレク

トを利用する。これにより、追加のソフトウェアをインストールすることなく、正当な証明書を持つクライアントのみが、インターネットから Jupyter Notebook にアクセスできる。この機能は nginx に組み込まれており、またセッションを開始する際のみ認証プロセスが実行され、それ以降は HTTP のリダイレクトのみとなるため、VPN サーバと比較して必要となる処理が少ない。

4. コミュニケーションの方法

中高生が取り組むプログラミングは、主に天文データの画像解析および統計処理であり、参考になる資料が少ない場合がある。そのため、コミュニケーションツールとして Discord を利用し、画像解析と統計処理について知識のある大学生講師への質問が可能となっている。このツールを選択したのは、基本的に料金がからず、過去のやりとりにも常にアクセスできる。チャンネルという概念があり、複数ある生徒のグループを管理しやすい。画面共有を含む通話機能が充実しているためである。生徒は質問する場合、現在操作しているプログラムファイルへの URL と、直面している問題を説明する。講師は、この質問を受けて当該ファイルを直接操作し、問題の解決を図る。

5. 本環境によって得られた成果

本環境を一年間運用し、3 名の大学生講師が助言、指導を行うことにより、高校生のチームが、人工衛星が撮影した画像をプログラミングを用いて解析し、その人工衛星の位置を特定した。加えて、白道面と黄道面のなす角の導出に成功した。この成果は第 21 回高校生・高専生科学技術チャレンジに提出され、審査委員奨励賞を受賞した。

6. 今後の展望

現在運用している環境は、単一の拠点に設置された複数台のサーバで構成されており、落雷やネットワークに係る実験などの影響を受け、アクセス不能になることが多い。そのため今後は、単一の拠点ではなく、複数の拠点に分散させ、耐障害性を高めたい。

7. 謝辞

本研究を進めるにあたりまして、名古屋大学教育学部附属中・高等学校の大羽徹教諭と、生徒の方々にご協力いただきました。この場をお借りして、御礼申し上げます。

参考文献

- [1] Google Colaboratory <https://colab.google/>
- [2] SoftEther VPN プロジェクト <https://ja.softether.org/>
- [3] nginx <https://nginx.org/>
- [4] jupyter project <https://jupyter.org/>

筑波大学 産学間連携推進室 成果報告書

画像解析による地球の軌道要素の導出

学籍番号：202111559

氏名：服部 真吾

概要

本研究では、数理モデルの構築から実データの解析に至るまでのプロセスを学習可能にする教育プログラムの構築を目的として、ひまわり 8 号の撮像した全球画像を解析することによって、地球の軌道要素を導出する手法を提案する。提案手法を用いて地軸と公転面の傾きを数値化し、作成した数理モデルをこの傾きの時間変化に対して最適化することで、実際の値に近い軌道要素が算出された。

1 はじめに

夏至と冬至で日照時間が大きく異なり、時期によって南中時刻が変化するの、地球の自転軸と公転面の成す角度が時間変化しており、かつ地球の軌道が楕円を描いているためである。これらの事実は、日常生活で実感でき、中高生レベルの物理知識で理解可能である。地球の自転軸と公転面の成す角度が時間変化していることを検証するには、年間を通じて南中高度を記録し、その変化を観察すればよく、地球の軌道が楕円であることを検証するには、年間を通じて太陽の視直径を観測し、地球と太陽が最も近づく近日点と、地球と太陽が最も遠ざかる遠日点における視直径の比を導出すれば良い。

しかしながら、太陽の視直径にもとづく導出では、観測誤差が非常に大きいことが問題であった。そこで杉原 [1] らは、観測技術の向上及びデータの積極的な公開によって、宇宙空間の静止衛星から地球を撮像した高精細なデータ

が利用できる [2] ことに注目し、地軸と公転面の成す角度の時間変化を用いて地球の軌道をモデル化することで、近日点や軌道離心率の導出を行う方法を提案している。本研究では、杉原らが提案した、気象衛星 8 号が撮像した画像をプログラミングによって解析する手法をもとに、近点年や地球の軌道離心率などを導出する。また、近年ではプログラミング教育が重視されており、学校教育の現場においてもプログラミングの積極的な活用が行われている。本研究で構築するモデル、導出するパラメータが中高生でも理解できるものであることから、本研究をもとにして、実世界の現象を記述する数理モデルの構築から、コンピュータを用いた画像データの解析に至るまでの、一連の科学的プロセスを持つ本研究に基づいた教育活動を行い、論理的思考力を持つ人材の育成を目指す。

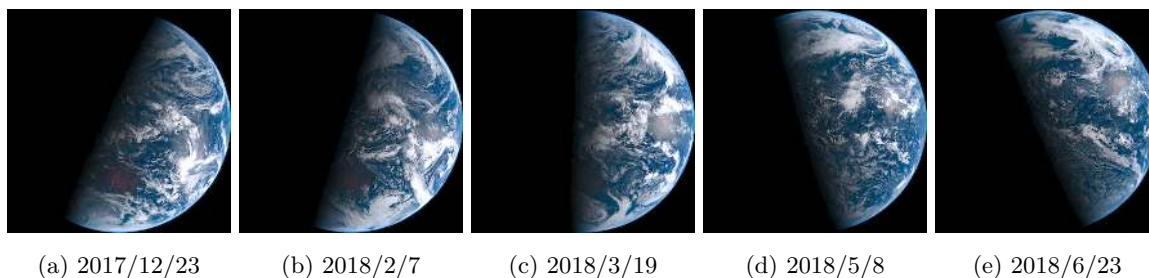


図 1: 静止気象衛星ひまわり 8 号が撮像した「半地球」の可視光画像の時間変化。

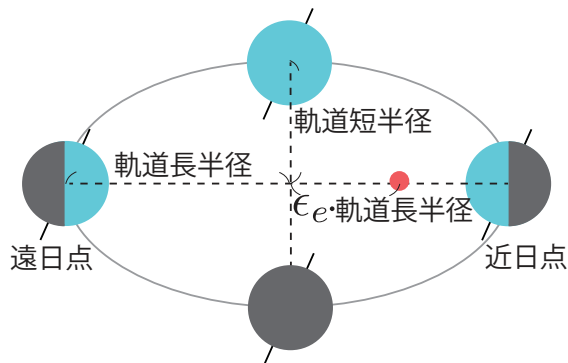


図 2: 地球の公転軌道と軌道要素の関係

2 数理モデルの作成

地球の公転面軌道と太陽の位置関係を、公転面に垂直な視点から観測すると、図 2 に示すようになる。この図において、赤い球が太陽、青い球が地球であるため、軌道離心率 e は、中心天体である太陽が、軌道長半径のどこに存在する場所を示す割合であることがわかる。

東経 140.2° 、北緯 0° で静止しているひまわり 8 号から地球を観察すると、図 1a に示すように地球のちょうど半分は太陽光が当たる状態が 1 日に 2 度ある。本研究ではこの状態を半地球と呼称し、半地球における明瞭域と暗領域の境界を明暗境界線と呼称する。

半地球の画像は地軸と画像の縦軸が平行であるため、明暗境界線と地軸の成す角は、図 1 に示すように一年を通じて時間変化する。そこで、図 3 に示す立体モデルを考える。このモデルにおいて、ひまわり 8 号が撮像した画像における明暗境界線を yz 平面に固定することで、 x 軸方向に太陽、 z 軸方向にひまわり 8 号がそれぞれ常時存在し、地軸のみが時間変化するモデルとすることができる。また、 xz 平面は地球の公転軌道面であり、 α は地軸の傾きを表す。そして明暗境界線と xy 平面に投影した地軸のなす角を、観測の基準となる日時から、撮像時

刻までの経過日数 t を用いて $\psi(t)$ と定義する。

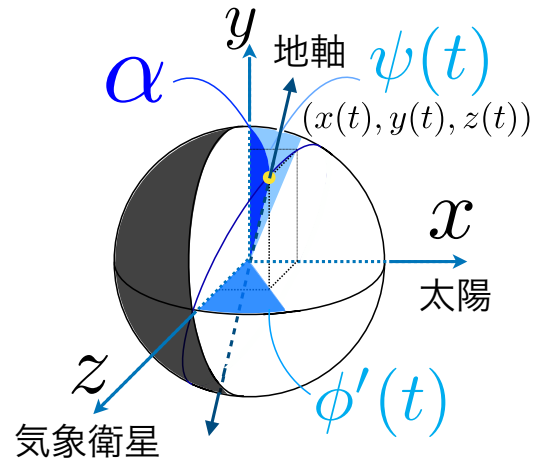


図 3: 半地球の立体図

次に、図 3 における $\phi'(t)$ を、図 4 に示すように、太陽の中心から春分点を結んだ線分と、任意の時刻の地球の中心と太陽の中心を結んだ線分のなす角と定義する。

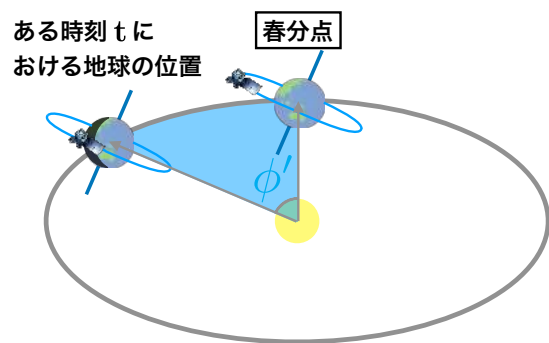


図 4: 春分点とある時刻での位置がなす角 $\phi'(t)$

数式化に際して、まず $\psi(t)$ の時間変化を表す関数を考える。気象衛星からみて半地球のとき、地球は図 5 のようになる。ここで、原点は地球の中心、 x 軸は赤道、 y 軸は地軸、明暗境界線と地球の端との交点は

$Q(-x(t), y(t))(y(t) > 0)$ である。そして、この図 5 より、

$$\tan \psi(t) = \frac{x(t)}{y(t)} \quad (1)$$

である。

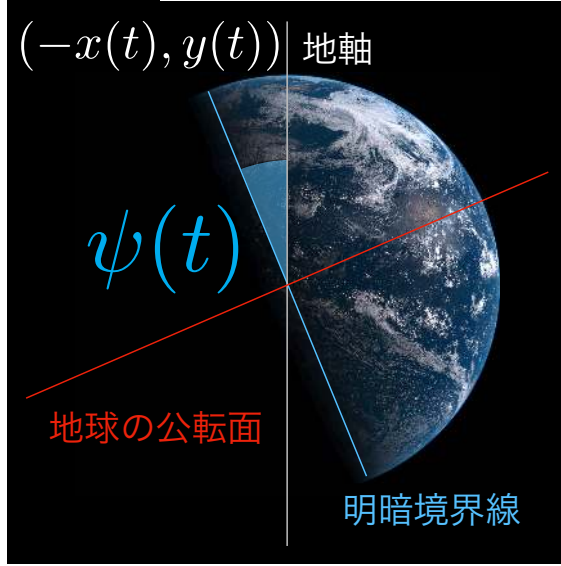


図 5: 半地球

次に、式 (1) の $x(t)$, $y(t)$ を極座標表示にする。図 3 より、

$$\begin{cases} x(t) = r \sin \alpha \sin \phi'(t) \\ y(t) = r \cos \alpha \end{cases} \quad (2)$$

である。ここで、 r は地球半径である。そして、式 (2) を式 (1) に代入すると、

$$\tan \psi(t) = \tan \alpha \sin \phi'(t) \quad (3)$$

となる。ここで、 $0 < \phi'(t) < \pi$ のとき、 $\psi(t) > 0$ であり、 $\pi < \phi'(t) < 2\pi$ のとき、

$\psi(t) < 0$ とした。また、 $\phi'(t)$ は式 (4) で表現される。

$$\begin{aligned} \phi'(t) = & \frac{2\pi}{P_e}(t - t'_0) + 2\epsilon_e \sin \frac{2\pi}{P_e}(t - t_0) \\ & - 2\epsilon_e \sin \frac{2\pi}{P_e}(t'_0 - t_0) \end{aligned} \quad (4)$$

ここで、 t'_0 は、観測の基準とする日時から、地球が公転軌道上の春分点に至るまでの日数であり、 t_0 は、観測の基準とする日時から、地球が公転軌道上の近日点に至るまでの日数である。また、 ϵ_e は地球の軌道離心率であり、 P_e は近点年である。

よって、最終的なモデルは式 5 で表現される。

$$\begin{aligned} \psi(t) = & \arctan \left[\tan \alpha \sin \left(\frac{2\pi}{P_e}(t - t'_0) \right. \right. \\ & \left. \left. + 2\epsilon_e \sin \frac{2\pi}{P_e}(t - t_0) - 2\epsilon_e \sin \frac{2\pi}{P_e}(t'_0 - t_0) \right) \right] \end{aligned} \quad (5)$$

ひまわり 8 号が観測した半地球の画像から、 $\psi(t)$ を測定する。求めたデータを、横軸を時間 t 、縦軸を $\psi(t)$ とするグラフ上にプロットする。式 (3) で与えられる曲線がこれらのデータを最もよく再現するように 5 個のパラメータ P_e , ϵ_e , t_0 , t'_0 , α を決めれば、この観測から求めた近点年 P_e と地球の軌道離心率 ϵ_e 、近日点 t_0 および地軸の傾き α が求められる。

3 解析

ひまわり 8 号が観測した半地球の画像から、 $\psi(t)$ を解析する。解析には、入門資料が多く、環境構築が比較的簡単である Python を利用し、ライブラリとして、OpenCV[6]、Numpy[5]、Emcee[7] を利用した。OpenCV は画像処理ライブラリであり、Numpy は行列計算ライブラリ

りである．また，Emcee は MCMC 法ライブラリである．本研究では，明暗境界線の決定に OpenCV と Numpy を使用し，数理モデルを測定値にフィッティングする際に Emcee を利用する．

3.1 利用したデータ

ひまわり 8 号は赤道上約 3,600km, 東経 140.7 度 (軌道保持範囲: 南北 0.1 度, 東西 0.05 度) [4] に位置する静止軌道衛星であり，可視 3 バンド (空間分解能: 0.5 km~1 km)，近赤外 3 バンド (同 1 km~2 km)，赤外 10 バンド (同 2 km) で，全球を 10 分毎の時間間隔 [3] で撮像している．

本研究では，ひまわり 8 号が撮像した可視光撮像データを 3 バンド合成し，補正した画像を，NICT サイエンスクラウドひまわりアーカイブプロジェクト [2] よりダウンロードして用いた．この画像は 11000×11000px であり，各ピクセルに R (中心波長: 0.64 μ m)，G (同 0.51 μ m)，B (同 0.46 μ m) 各チャネルの強度が 0 から 255 の整数で記録されている．

3.2 明暗境界線の決定

まず，RGB の可視光 3 バンドで構成されている半地球の画像の各ピクセルを，式 (6) を用いてグレースケール画像のピクセルに変換する．

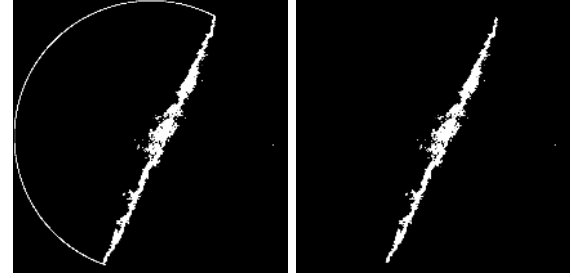
$$Y \leftarrow 0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B. \quad (6)$$

次に，式 (7) を用いて，グレースケール画像のピクセルを 1 または 0 の二値に変換する．

$$\begin{cases} N = 0 \text{ (where } Y = 0), \\ N = 1 \text{ (where } Y > 0). \end{cases} \quad (7)$$

ここで，OpenCV に実装されている，輪郭抽出関数である FindContour を用いて明暗境界線のみを検出する際，太陽に照らされている領

域の輪郭のうち，地球本来の輪郭は不要である．そのため，ひまわり 8 号から撮像された地球の中心座標がほぼ一定であることに注目して，検出された輪郭のみが描画された画像に，地球と同程度の円を描画することで，明暗境界線の輪郭のみを抽出した (図 5)．



(a) 処理前 (b) 処理後

図 6: 明暗境界線の抽出

最後に，findCountor 関数によって抽出された明暗境界線の輪郭は座標の集合であるため，Numpy に実装されている Polyfit 関数を利用して，これらの座標を一次式で近似することで図 7 に示すような明暗境界線の傾きを求めた.¹⁾

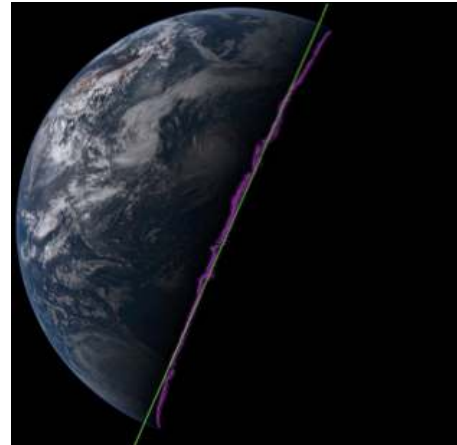


図 7: 明暗境界線の直線近似

1) 半地球の画像は，1 日で 2 回観測できるので，ある日の明暗境界線の傾きには，日出時の傾きと，日没時の傾きに -1 をかけた値の平均値を用いた．

ここで、ひまわり 8 号が撮像する画像は、図 8 に示すように、ある一点から地球を捉えたものになる。そのため、撮像された画像について、中心から離れるに従って明暗境界線に歪みが生じる。この歪みは、明暗境界線が直線である場合は無視できる。しかし、ひまわり 8 号の撮像間隔は 10 分間隔であり、完全な直線となった明暗境界線を捉えることができないため、今回は無視できない影響を及ぼす。そこで、図 9 に示すように、地球の輪郭よりもやや大きい範囲に縁を描画することで、歪んでいる領域を利用しないようにした。

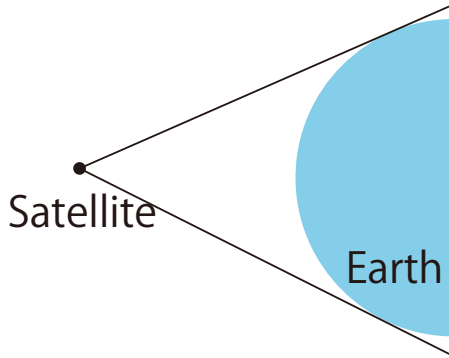


図 8: ひまわり 8 号と地球の関係

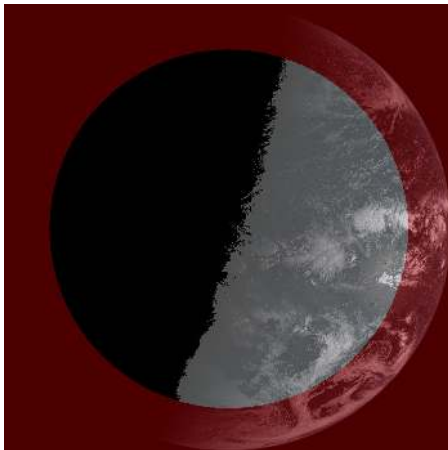


図 9: 明暗境界線のうち実際に使用した領域

また、明暗境界線はグレースケール画像を二値化することで抽出されているが、この時、雲や観測ノイズなどによって明暗境界線上に凹凸が生じる場合がある。そのため、これらノイズによる影響を軽減するために、図 10 に示すように、画像領域を縦方向に等間隔で分割し、そのうちの 1 つの領域をマスク領域として設定した。そして、残りの領域に含まれる画素を用いて、最小二乗法により直線近似を行い、明暗境界線の傾きを算出した。最後に、この処理をマスク領域を変更しながら繰り返し、得られた明暗境界線の傾きの平均値を特徴量として用いた。なお、今回利用した画像は縦横 11000 ピクセルと巨大であるため、画像を 50 分割した。

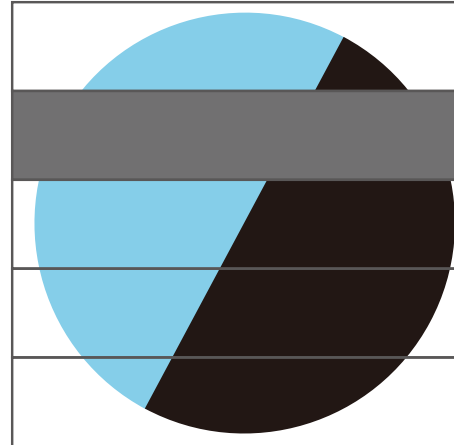


図 10: マスクの例

3.3 結果

オープンソースの MCMC ライブラリである Emcee[7] を利用して、式 (5) が得られたデータを最もよく再現するように、パラメータ P_e , ϵ_e , t_0 , t'_0 , α を推定した。このときに必要となる、観測時刻 t における観測値 $\psi(t)$ の誤差には、利用可能な観測範囲において、同じ日付の日時に観測された明暗境界線の傾きの標準偏差を用いた。

解析においては、NICT サイエンスクラウド [2] より、ひまわり 8 号が撮像した半地球の画像を、2015 年 3 月 20 日から 2022 年 12 月 13 日までの 5588 枚ダウンロードして解析した。半地球の画像から地軸と明暗境界線の傾き $\psi(t)$ を測定し、その時間変化 $\psi(t)$ を描画すると図 11 に示すとおりのになった。また、モデルフィッティングにおいては、利用可能なデータの中で、同じ日付に観測された明暗境界線の傾きの標準偏差を観測値の誤差として用いており、その分布は図 12 に示す通りとなった。

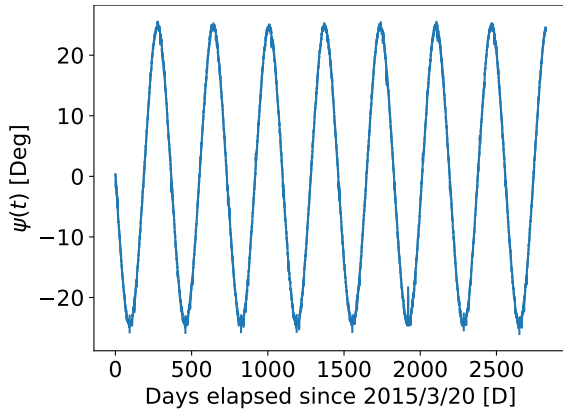


図 11: 明暗境界線の傾きの時間変化

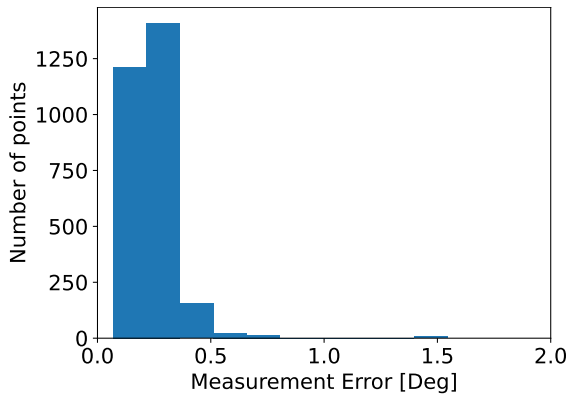


図 12: 標準偏差の分布

次に、図 11 にプロットした観測データを忠実に再現するよう、式 (5) のパラメータ t'_0 , t_0 , ϵ_e , α を最適化した。得られたパラメータは表

1 に示すとおりのになり、誤差は表 2 に示す通りになった。

地球	本研究で得られた値
近点年 P_e	365.24 日
軌道離心率 ϵ_m	-0.008
近日点 t_0	2015 年 9 月 9 日
地軸の傾き α	23.768°
春分の日	2015 年 3 月 21 日 16 時 56 分

表 1: 地球の軌道要素

地球	誤差
近点年 P_e	+0.014, -0.007
軌道離心率 ϵ_m	+0.017, -0.001
近日点 t_0	+132.60, -50.68
地軸の傾き α	+0.006, -7.073
春分の日	+62.019, -0.035

表 2: 軌道要素の誤差

4 考察

解析結果として、近点年、春分の日、地軸の傾きは実際の値とほぼ同一となり、近日点は実際とは大きく異なるものに、軌道離心率については 0 以上 1 以下であるはずが、負数となった。これは、近日点 t_0 を含む項 $2\epsilon_e \sin \frac{2\pi}{P_e(t-t_0)}$ と $2\epsilon_e \sin \frac{2\pi}{P_e(t'_0-t_0)}$ が互いに軌道離心率 ϵ_e を持っており、これらの項が残りの項 $\frac{2\pi}{P_e}(t-t'_0)$ と比較して非常に小さいことにより、フィッティングに与える影響が小さくなったためだと考えられる。また、春分の日及び近日点について、誤差が非常に大きくなっているのは、前述と同様、これらの項がモデル全体に与える影響が小さく、現在与えられているデータでは、精度を高められないためだと考えられる。

明暗境界線の傾きを導出する際に、領域を分割したことによる効果を考察する。例として、3 枚の画像について、マスクした箇所と、その状態から導出された明暗境界線のパラメータの関係を図 13 に示す。

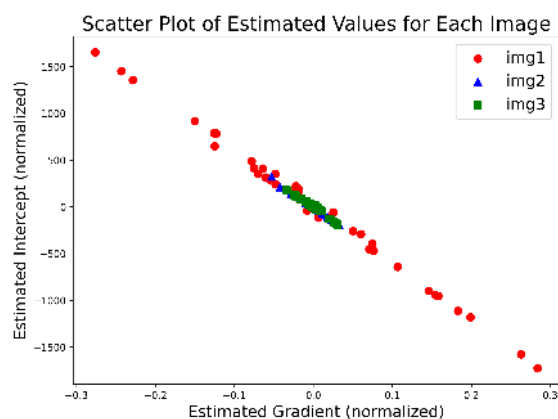


図 13: 標準偏差の分布

図より、画像によっては、マスクする箇所を変更するだけで、推定される明暗境界線のパラメータが大きく変化することがわかる。この推定されたパラメータの元になった画像 img1 と img2 をそれぞれ図 14 と図 15 に示す。

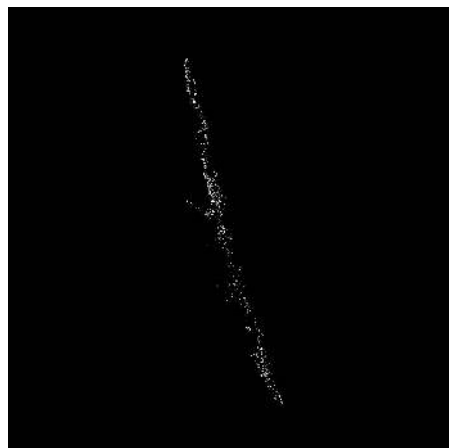


図 15: img2

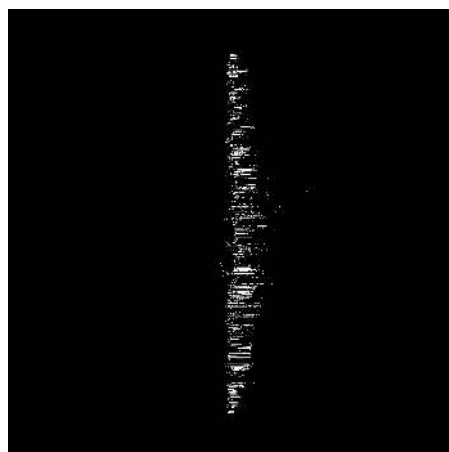


図 14: img1

図より、今回の変換方法では、明暗境界線がはっきりとせず、期待よりも太くなってしまう場合があることがわかる。このようなデータの存在によって、決定された明暗境界線のデータに微妙なずれが生じ、最終的に推定された地球

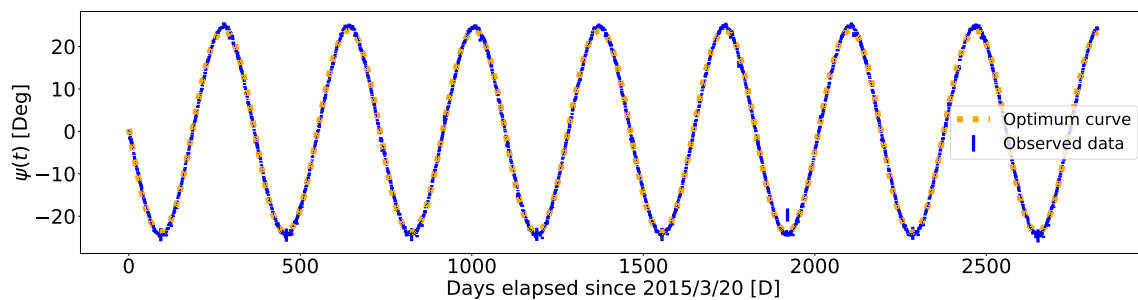


図 16: 明暗境界線の傾き $\psi(t)$ の時間変化

の軌道要素が、モデル全体に大きな影響を及ぼすものについては実際の値に近く、小さな影響を及ぼすものについては、実際の値とは異なるものになったと考えられる。

5 まとめ

本研究によって、一般に公開されているデータとプログラミングを用いて、明暗境界線の時間変化を測定し、数理モデルに当てはめることで、近点年や地軸の傾きといった軌道要素を導出できることが確かめられた。

しかしながら、モデルに与える影響が小さい項について、実際の値とは大きく異なるものが得られた。そのため今後は、これらの問題を解決するために、ひまわり 8 号の運用情報や撮影されたデータを確認し、観測機器による誤差の有無を確認した後に、より正確に明暗境界線を検出する方法を研究したい。また、フィッティングにおいて、パラメータの取りうる値を元にした制約や、与えるデータの種類を増やすことによって、近日点と軌道離心率を正確に推定できるフィッティング方法を検討したい。加えて、本研究をはじめとした様々な分析活動から構成される「数理モデルの構築から実データの解析に至るまでのプロセスを学習可能にする教育プログラム」の内容に基づいたイベントを、産学間連携推進室で開催されている「INDEPENDENCE SERVER DAY」等小中学生向けイベントの派生として、科学的活動に興味がある小中学生に対して行うのみならず、産業界においても、プログラミング教育のチュートリアルなどとして行いたい。

6 謝辞

本研究は、名古屋大学教育学部附属中・高等学校の大羽徹教諭にご協力いただきました。また、本論文で使用した画像の一部は、名古屋大学理学部の石川陽氏よりご提供いただきました。

使用したデータは、NICT サイエンスクラウド ひまわり衛星プロジェクト ひまわり衛星データアーカイブ [2] から取得しました。

参考文献

- [1] 杉原悠太, 石川陽, 山中孝太郎, “気象衛星の画像解析から求めた地球と月の軌道,” 第 15 回日本物理学会 Jr セッション, 2019, 17JPSB-24.
- [2] NICT サイエンスクラウド ひまわり衛星プロジェクト, <https://sc-web.nict.go.jp/himawari/himawari-archive.html>.
- [3] 横田寛伸, 佐々木政幸, 静止地球観測衛星「ひまわり 8 号および 9 号」の紹介, <https://www.data.jma.go.jp/mscweb/technotes/msctechrep58-3.pdf>.
- [4] 気象庁気象衛星センター, ひまわり 8 号・9 号の運用計画, <https://www.data.jma.go.jp/mscweb/ja/info/plan.html>.
- [5] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, “Array programming with NumPy,” *Nature*, vol. 585,

no. 7825, pp. 357–362, Sep. 2020. DOI: 10 . 1038 / s41586 - 020 - 2649 - 2. [Online]. Available: <https://doi.org/10.1038/s41586-020-2649-2>.

[6] G. Bradski, “The OpenCV Library,” *Dr. Dobb’s Journal of Software Tools*, 2000.

[7] D. Foreman-Mackey, D. W. Hogg, D. Lang, and J. Goodman, “Emcee: The mcmc hammer,” *Publications of the Astronomical Society of the Pacific*, vol. 125, no. 925, pp. 306–312, Mar. 2013, ISSN: 1538-3873. DOI: 10 . 1086 / 670067. [Online]. Available: <http://dx.doi.org/10.1086/670067>.

居住空間での業務用電子計算機の運用に関する研究

太田弘樹 s2220573@s.tsukuba.ac.jp

2024-05-23

本研究の目的

現在、産学間連携推進室では居住空間と隔離されない形で計算機区画が設置されており、当該区画には各員の研究に必要な無数の計算資源や通信装置が設置されている。しかし、この機材が居住空間に影響を及ぼし、室員の研究活動の妨げとなってしまうていた。この研究ではそういった問題を解決しようとするとともに、計算機室を用意できないが業務用電子計算機を利用したいという組織において発生しうる問題の列挙や対処方法の検討を行う。

業務用電子計算機の特徴

本稿で述べる業務用電子計算機は、一般の人が日常や業務において卓上で利用するような電子計算機ではなく、専用の架台に設置され、平時は近傍に人がいない場所に設置されるような計算機を指す。

これらの装置は性能や空間効率、冷却能力などを念頭において設計されており、一般向けの装置であれば考慮すべき騒音や発熱はほぼ考慮されておらず、隔離された専用の部屋で空調などを管理された状態で使用されること、つまり利用環境側で対応することが想定されている。

実際の環境における発熱や騒音

実際に産学間連携推進室に設置された機材がどの程度の騒音と発熱をしているのかを簡易的に測定した。騒音の測定は架台に設置された機材の背面 1m ほどの距離で、計測装置を胸の高さほどに保ち行った。

機材の発する騒音は約 73dB であり、騒音の目安(出典「全国環境研協議会騒音小委員会」)によると、鉄道車内と変わらないくらいの騒音であるとされる。しかし、電子計算機の発する騒音はこういった騒音に比べて高い周波数の成分を多く含む、機械的な音を発するため、体感としての騒音はより気に障るような音に感じる。

また、発熱量の測定は機材の電力消費量がそのまま熱に変換されたという想定で、電力消費量の計測を行った。

実際に計測された値は消灯状態、空調機を抜いた部屋全体の消費電力として 8.8kw であり、生活家電や各自の計算機の消費電力を考慮したとしても 7kw 以上消費している。すべてが熱に変換されるわけではないものの、

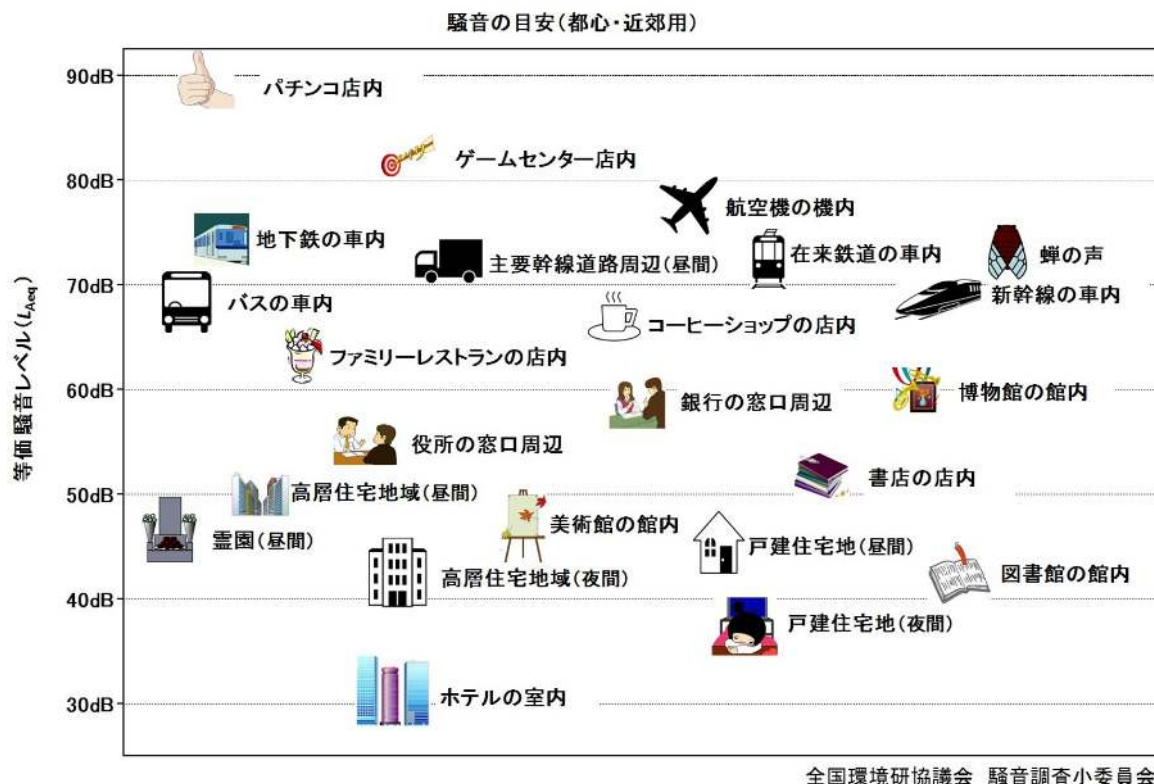


図1 騒音の目安

およそ家庭用灯油暖房機 1.5 から 2 台分程度は熱が出ている計算になる。

計算機区画と居住空間の関係

昨年度までのところでは、計算機区画と居住空間の仕切りの上部が空いており、空間的に居住空間と接続された状態になっていた。計算機区画には空調機械や自動火災報知機などが設置されておらず、居住空間に設置された物に頼っていたためであり、塞ぐわけにはいかないという事情もあった。

しかし、遮蔽が十分でないことで居住空間へは騒音が漏れ、また、区画間の換気が十分でないことで、計算機区画へ十分な冷却能力を提供するために居住空間は過度に冷却されるにも関わらず、計算機区画は対して冷えないという状況が発生していた。これによって、室内は常に肌寒く、耳栓等を着用しないと騒音により集中できない環境になっていた。この環境の悪化は研究活動に重大な負の影響を及ぼし、実際に一時期室員による研究室の利用率が低下していたと感じた。

業務用電子計算機の冷却に関する特徴

先述したように業務用電子計算機は高密度に集積された状態で運用することが多く、冷却設計もそれに対応した形となっていることが多い。具体的には、空気を全面から吸引し背面に排気する設計となっている機材が多い。この空気の流れが統一されていない状態で運用してしまうと、近隣の機材が吐き出した熱風を吸い込んでしまい、またその機材が吐き出した熱風を近隣の機材が吸い込むという悪循環が発生し、最悪の場合機材の故障などの障害が発生してしまう。

これらの装置を専門に大量に扱う業務では、こうした特性を適切に利用し空調設計などと組み合わせることにより、効率よく計算機を冷却している。

また、これらの多くの電子機器の冷却機構には、空冷機構が採用されているが、高密度搭載の機器などでは筐体高さが45mmほどであることから、冷却扇の大きさも制限されてしまい、その筐体から発せられる熱を十分に排除するためには多数の高回転で強力な冷却扇を搭載しなければならない。これら定常状態で10000回転以上の物もあり、さながら航空機の推進機関のような音を発する。

しかし、吸気温度や、機器の負荷状況によって通気量を調整する機構が存在しているため、多くの機器ではその発する熱量に応じて冷却扇の能力を制限し、冷却機構において消費される電力も削減することができるが、逆に冷却能力が不足している場合は、最大の冷却を得るために冷却扇を高速回転させる。そのため、熱気がこもってしまうと冷却扇の高速回転に伴う多大な騒音も発生することになる。

冷却能力と騒音削減の相反

私は流体の専門家ではないので背景にある理論等はわからないため、直感としてになってしまうが、騒音を発する物体を静音のためになにか箱のようなものに封じ込めようとするとき、共鳴等は考えないものとした場合箱は密閉されていることが好ましく、もし開口部があればその開口面積が小さければ小さいほど騒音の漏洩が小さくなるはずである。

しかし今回の場合箱に封じ込められるものは発熱体でもあり、これを十分に冷却したい。この場合密閉してしまうと熱が籠もり冷却ができない。そこで外気を取り入れるための開口部を設け外気を導入することとする。開口部が大きければ大きいほど、小さな力で箱内の空気を入れ替えることができるはずだ。

この2つのことを考えると、冷却と騒音削減を両立することが難しいように感じる。

産学間連携推進室で検討した対策

計算機区画の騒音は当室室員の間でも問題になり、部屋としての対策の検討を行った。

具体的な内容としては、計算機区画をの仕切りを変更し天井まで完全に密閉される仕切りに変更し、空調機や、自動火災報知機なども設置し、計算機区画を新しく部屋として分割することによって冷却と騒音の改善を図るものであった。

しかしこの計画には大きな問題があった。学生研究室にはお金がないのだ。

実際に業者に見積もりを依頼したところ、諸経費込みで 175 万円程度かかるということであった。

この額は大きな収入源のない本研究室では負担しきれる金額ではなく、また、この工事によって 175 万円を超える収入が見込まれる研究が計画されているわけでもなく、相殺等もできないという結論になり、本計画は凍結されることとなった。

また、架台を静音設計されたものに交換する案も考えられたが、こちらも一架 50 万円を超え、二架で合計 100 万円以上の出費となってしまうため、見送られた。

参考：摂津金属工業株式会社 大型中型防音ラック（静音ラック） NBSP シリーズ
38U 1200mm 578,100円

今回産学間連携推進室で試行した静音化対策

しかし、悪化した環境は耐え難きものであったため、私個人として独自の対策に乗り出すこととした。

無論、個人で支出できる金額は部屋の予算より遥かに小さく、その金額で如何に大きな効果を得るかが焦点になってくる。

そこで前述の特徴等を踏まえ、産学間連携推進室にて大きく分け 2 種類の対策を行った。一つは騒音の発生を抑えるための冷却の改善で、もう一つは発生してしまった騒音を居住空間に送り込まない改善である。

冷却の改善

前述したように近年の大規模電子計算機設置事業者などでは吸気側と排気側を空間的に分け、排気の回り込みを減らすとともに空調機械との効率良い結合を行い冷却にかかる消費電力の改善を行っている。

そこで、産学間連携推進室計算機区画に置いても、排気側と吸気側の分割を行い、冷却効率の改善を行った。

具体的には、計算機区画において架台の前後は自由空間となっていたものを近隣の店舗で購入可能な構造用合板を加工し吸気側排気側を分離し、排気の回り込みを軽減した。材料費は 1 枚 1500 円ほどの構造合板を 4 枚ほどと、板を止めるボルト等金具類 1000 円ほどを加え小計 7000 円になり、加工等して直線切断加工が 1 切断あたり 50 円を 6 回ほどで小計 300 円ほどかかった。機械加工のできない形状については追加で手加工を行った。そのため合計で 8000 円を下回る金額で空気の流れを改善することができた。

また、架台内部での回り込みについては、ビニールを貼るなどし機材間の隙間や架台内部の隙間を封鎖した。

これにより、機材吸気側温度で約 40℃ 以上常にあった状態を約 30℃ 程度に抑えることに成功した。



図2 加工作業の様子

騒音の改善

産学間連携推進室計算機区画の仕切りは上部、いわゆる欄間部分が開放された仕切りであったため、音響的な遮蔽能力はほぼ無い仕切りであった。

そのため、これも近隣の店舗で購入可能な構造用合板を加工し、一部閉鎖した。これも材料費は1枚1500円ほどの構造合板を3枚で小計4500円程度で済み、加工費も400円程度で済んだ。

参考値として、扉開状態と、扉締状態で騒音値を計測したところ、A特性において開放時62.0dBと締切時54.2dBという結果が得られたため、およそ7.8dBの削減を達成した。

まとめと今後

今回は産学間連携推進室の居住性向上のために計算機区画の遮音や空調効率改善を安価にできる方法について調査、実験を行った。

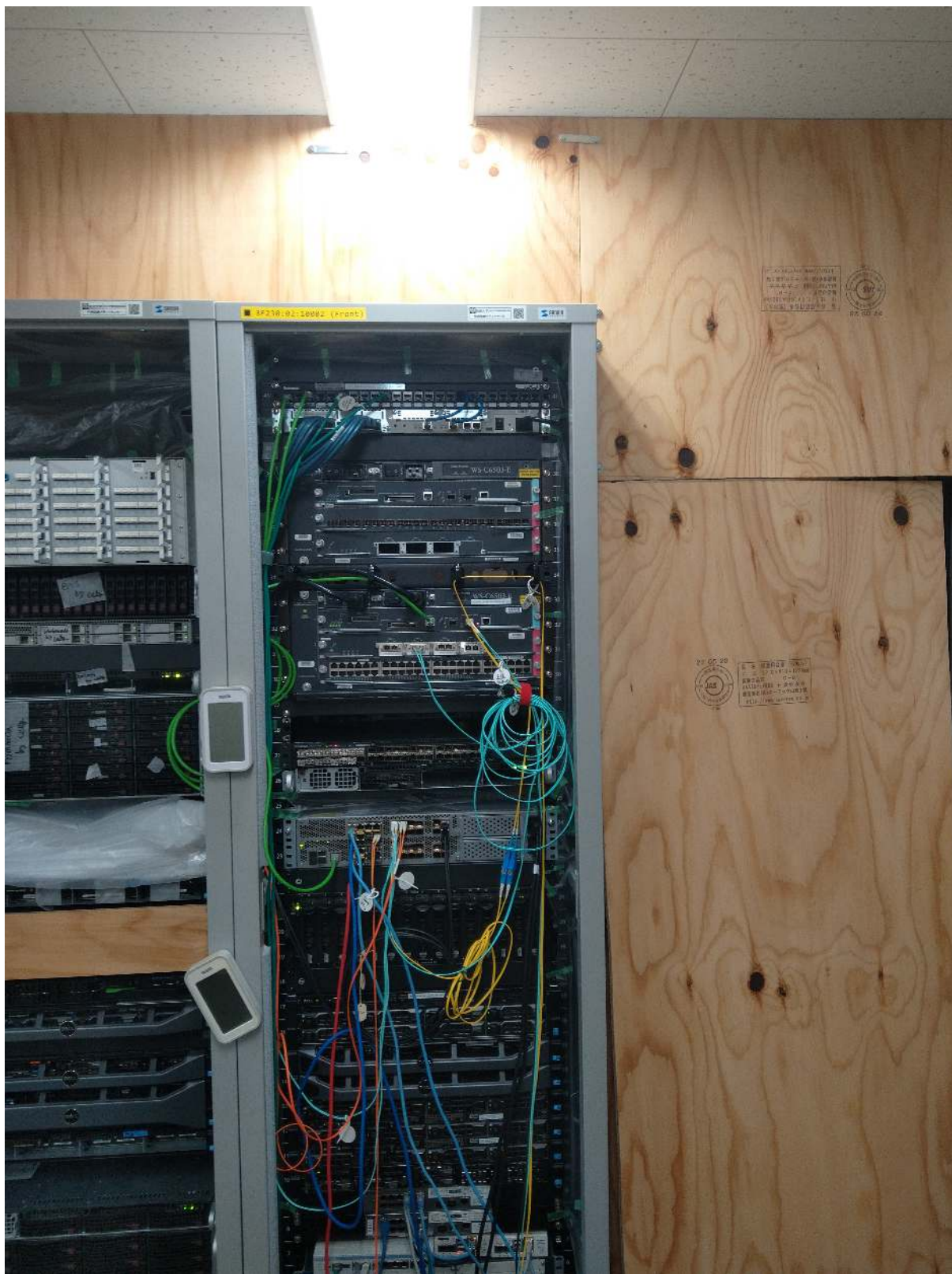


図3 施工後の架台の様子



図4 欄間

もっともらしい工事の見積もりは175万円程度であったが、今回の手法では、それより遥かに安価な2万円程度である程度の改善を行うことができた。室員からも、研究環境が改善したという意見を得ることができた。

しかし、もっともらしい工事で期待される遮音効果に比べると今回の手法は不完全であると言わざるを得ないだろう。

また、商用環境では架台内外に可燃物を設置することを禁止していることもある。そういった環境では、今回のように架台内外にふんだんに可燃物を敷き詰めるような手法は、火災発生時に延焼の原因となる可能性が高く、利用できないだろう。

しかし、可燃物仕様に制約のない環境や、個人宅での業務用計算機の習熟などの使用用途であれば、今回のような手法は適用可能であり、有効であると考ええる。

今後はより音響特性や音波の伝播特性などを調査し、より効率よく耳障りな音を遮断できるよう、調整や実験を行っていく。また、換気などについてもより効率よく静かに大容量給排気可能な手法等を調査や実験していく。

情報機器資源リユースプロジェクト 活動報告

～ 「まだ使える」機器の有効な活用を目指して ～

2023 年 6 月 3 日 筑波大学情報科学類 産学間連携推進室 活動成果報告会

情報機器資源リユースプロジェクト 第 5 代統括作業総責任者

太田 弘樹 [s2220573@u.tsukuba.ac.jp]

協力：産学間連携推進室 根本晃輔、関口亞聖、小野村卓也、服部真吾、金子尚樹

1. 序論

1-1. 本プロジェクトの背景

筑波大学 システム情報系エリアにおいては例年、毎年 2 回（6 月および 12 月）の頻度で、図 1 のような各種の大型物品¹の一斉廃棄が行われている。その中でも高度な情報機器資源であるコンピュータについては、図 2 のように、毎回の物品廃棄で約 250 台～300 台が廃棄されている。



図 1：システム情報系エリア物品一斉廃棄の様子

¹ 什器類、家電類、実験機器類、情報機器類を指す



図 2：2021 年 6 月に廃棄されたコンピュータ類

これらの廃棄されたコンピュータについては、確かに Intel Pentium4 世代や Intel Core 2 世代、あるいは第 3 世代以前の Core i シリーズの CPU を搭載したコンピュータなど、既に研究の場での現用が難しいもの²が殆どである。しかし一部には、たとえば第 4 世代以降の Intel Core i 世代のコンピュータなど、パーツを交換・増強すれば現用に耐えうる性能でありながら、単純な経年・入替のため廃棄されたものや、修理・交換可能な部品の一部のみが故障したために廃棄されている比較的新しいコンピュータも存在している³。それらは部品を適切に交換・修理したり、性能向上を図れるパーツを追加したりすることで、ある程度の期間、再度利活用することが可能になる。

このように、故障したコンピュータを積極的に買い換え、故障したコンピュータを廃棄することのできる研究室がある一方で、筑波大学における一部の研究科・専攻においては、高価な実験機材を購入するためにコンピュータに十分な予算を設けることができず、やむ

² 既に製造から 12 年以上が経過しており、近年のソフトウェアの要求性能に満たない性能の機器が多いこと、および長期の使用による修復困難な故障などが発生しているものが多いことから、現用が難しいものと考えられる

³ 中にはマザーボードの致命的な故障など重大なものも含まれているが、OS レベルの障害や一部の交換可能なパーツの故障などが発生した際に、その原因を詳細に特定することなく廃棄されているものも少なくないと推測される

を得ず Intel Core 2 Duo ～ 初代 Core i シリーズ 世代などの古いコンピュータを限られた台数で利用している研究室が存在している。また、研究室における急激な人数の増加に対し、予算が限られているなどの理由によりコンピュータの新規導入が難しいという状況や、留学生のために短期的に利用できるコンピュータが必要になるという状況にある事例もみられるほか、2020～2021 年頃には在宅での勉学・研究を行うために貸出用の PC が必要となった研究室なども存在した。

筑波大学の物品廃棄において廃棄される物品のうち、コンピュータ以外の一般の備品については、教職員が閲覧可能なシステム上で次の利用者を募集するなどのリユースの仕組みが構築され、実際に運用されている。

しかし、性能や経年の状況が外観から判断できず、限られた情報からでは詳細な状況を判断しづらいコンピュータについては、その仕組みの枠の中で取り扱うことが難しい状況にある。更に、コンピュータを廃棄・譲渡する際にはデータの厳重な消去が必要になるが、その方法については教職員間でも徹底がなされておらず、そのような徹底が不十分な状況下で安易な廃棄・譲渡を行うことは、時として情報漏洩などの重大な事故を招きうるものとなる。

1－2．本プロジェクトの目的

このような諸課題を解決するための情報機器資源のリユースの試み、および付随的な目的として情報記録媒体の確実な廃棄の補助・物品一斉廃棄の支援を行う本プロジェクトは、システム情報系技術室の要請の下、2015 年 6 月に根本（産学間連携推進室所属）らにより発足した。そして、2024 年 5 月現在に至るまで、規模を随時拡大しながら運用が続けられている。

本プロジェクトは、筑波大学システム情報系技術室の指揮・管理の下で、物品の一斉廃棄の際に廃棄されるコンピュータを適切に修理・再構成し、システム情報系技術室によって上述のようにコンピュータを必要としている研究室や各種組織などに分配することで、情報機器資源のリユースを行うことを目的としている。またリユースの過程の中に、廃棄されるべき情報記録媒体の確実な破壊・廃棄、および再利用される情報記録媒体におけるデータの確実な削除を組み込むことで、物品廃棄に際し発生しうる情報漏洩などのインシデントを未然に防ぐことも付随的な目的としている。

2. プロジェクトの概要

2-1. 活動の概略

「情報機器資源リユースプロジェクト」は、システム情報系技術室の要請・協力の下、情報科学類産学間連携推進室内に設置されているプロジェクトである。本プロジェクトにおいて、産学間連携推進室のメンバーはコンピュータの修理・再構成などの技術的部分を主に扱っているため、本稿ではその担当範囲における活動について述べる。

本プロジェクトは2015年6月（発足時）～2023年12月度の全17回の物品廃棄において、デスクトップパソコン・ディスプレイ・ノートパソコンを併せ、総計750台以上のリユースを行った。集計が完了している2021年6月度までの内訳を表1に示す。

表1：リユースの内訳。2020/06 は新型コロナウイルス感染拡大の影響により中止。

Data	Desktop	Laptop	Display	Sum
15/06	15	10	15	40
15/12	15	0	15	30
16/06	23	0	20	43
16/12	16	2	34	52
17/06	32	21	39	92
17/12	12	2	5	19
18/06	30	20	10	60
18/12	28	15	9	52
19/06	35	5	12	52
19/12	17	2	11	30
20/06	-	-	-	-
20/12	25	5	14	44
21/06	16	5	17	38

本プロジェクトは現在、2024年6月の物品一斉廃棄においても同様の作業を進めており、2024年7～8月中には再利用可能となったデスクトップパソコン・ディスプレイ・ノートパソコンをシステム情報系技術室に引き渡す予定である。

2-2. 工程と運用

本プロジェクトにおけるコンピュータの修理・再構成の工程においては、部品が欠損・故障しているコンピュータから性能の高い再利用可能な部品（CPU・メモリ・グラフィックカード等）を取り出し、状態の比較的良好なコンピュータにそれらの部品を移植することで、修理・性能向上を行う形をとっている。また、液晶ディスプレイなどの表示機器についても、動作確認を行った上で、ある程度の解像度があるものや経年していないものについては形式の取りまとめを行い、機器のリストと合わせ技術室に引き渡す形をとっている⁴。

現在の本組織の運用形態は、産学間連携推進室メンバーの 6 名および情報科学類・情報メディア創成学類・大学院生の有志学生 6 名からなる本プロジェクトのように少人数な組織であっても、十分に運用可能なものである。なお、活動の様子を記録した写真を図 3・図 4 に示す。図 3 はディスプレイ等各種機器の状態確認、図 4 は専用に構築した機器を用いてハードディスクのデータ消去を行っている様子である。



図 3：ディスプレイ等各機器の動作確認を実施している様子

⁴ なお、2017 年 12 月以降の表中の「ディスプレイ」の数値は高解像度のものや、形式の取りまとめを行っているものに限っている。高解像度でないもの、あるいは若干の経年があるものの利用が可能であるディスプレイについてもリユースを行っているが、作業の簡易化のために取りまとめを行っていないため、この表中には記載していない。



図4：ハードディスク処理作業の様子

3. リユースの事例

本プロジェクトの活動実績を示す上での参考として、実際にこれまでのプロジェクトの進行の中で行ってきたコンピュータの修理・再構成の例の一部を示す。

3-1. ノートパソコンの液晶画面の入れ替え

搬出された同型のノートパソコン 2 台のうち、1 台は液晶ディスプレイ部分が破損していたものの筐体をはじめとする他部分には損傷がなく、もう 1 台はタッチパッド・DVD ドライブ部をはじめとする筐体部に大きな歪みがあったものの、液晶ディスプレイ部分には損傷がない状態であった。

そこで、後者のノートパソコンから液晶ディスプレイ部を取り外し、前者のノートパソコンに移植することで液晶ディスプレイの交換作業を行い、無事に利用できる状態に回復させた。なお、後者のパソコンに搭載されていた取り外し可能な他のパーツのうち、メモリとハードディスクについては取り外しを行い、他のノートパソコンへの移植を行った。

3－2．エアダスターや無水エタノールなどを用いた分解清掃

搬出されたデスクトップパソコン・ノートパソコンのいずれについても、基板上・ファン内部などの各部位に蓄積した埃などについては、エアダスターや無水エタノールを用いた清掃を行っている。また、ノートパソコンについては消毒用エタノールを用いた清掃を行い、次の利用者が快適に利用できるような環境を整えている。

特に、デスクトップ Mac 環境などの特殊な環境の動作を維持するためのパーツは絶対数が少なく、供給の少ない部品を清浄しながら利用する必要がある。中には煙草の脂と思われるものが付着しているものもあったが、この分解清掃を行うことで外見・動作とも問題のない状況に仕上げる事が出来た。

3－3．パーツ世代・電力量などを加味した各種パーツ等の増設や高性能化

搬出されたデスクトップパソコンについては、例えば CPU をより上位のものにアップグレードできるものや、更に大容量のメモリを搭載できるものなどが存在する。それらのパソコンについては、アップグレード可能なパーツがあれば、電力量や想定用途などのさまざまな要素を加味した上で、再利用の際に支障が出ない範囲でのアップグレードを施している。

4．今後の展望と課題

本プロジェクトにおける今後の活動の展望、および現状の課題に関して以下の通り記す。

4－1．継続的な活動に向けて

本プロジェクトは 2024 年度の物品廃棄においてもコンピュータの修理・再構成を実施しており、今後も活動を継続していく予定である。特に 2022 年度の活動においては、特殊工具を含めた資材の大幅な拡充等を行ったことにより、前年度に引き続き作業効率や作業速度を向上させること、および作業手順のマニュアル化を進めることで作業の属人化・負担の集中を防ぎ、今後も継続して活動ができるプロジェクト運営を行うこと、の 2 点を重点におく形で活動を進めることができた。本プロジェクトは創設より 5 代に亘り代替わりを続けながら活動を継続しているが、大学組織に特有の「メンバーの入れ替わり」があっても持続可能な、なおかつ人々に必要とされる組織を構築していく上では、本年度の成果には非常に大きな意義があるといえる。

4－2．他エリアへの拡大に向けて

筑波大学においては、システム情報エリアに限らず、さまざまな学域・エリアにおいて物品の一斉廃棄と、それに伴うコンピュータ類の搬出が行われている。それらのコンピュータ類にも、システム情報系物品廃棄において搬出されているコンピュータ類と同様、まだ十分に利用可能な製品や、修理および再構成を行えば数年間は利用可能な製品が含まれており、これらをシステム情報系の物品廃棄における本プロジェクトと同様の形でリユースすることができれば、当該エリアにおける情報機器資源の効率的な運用や、比較的高性能なコンピュータを活用した研究の推進に役立つことが考えられる。またもちろん、情報技術に明るくない学域・エリアにおいては、本プロジェクトの副次的な効果である HDD/SSD などの記録媒体の安全な処理を推進していくことで、より安全に情報記録媒体の廃棄を進めていくこともできるだろう。

エリアを跨いだ活動には現状下で困難が多いと思われるが、このような活動をより広範な形で推進していくことにより、大学内における経年した旧型機器の入れ替えの推進による研究活動の効率化や、昨今の大学・各種研究機関において非常に大きな問題となっている予算の逼迫の解決、廃棄機材からの情報漏洩への対策に対する一つのアプローチとなるのではないかと思われる。

4－3．知見の一般共有

本プロジェクトの副次的な効果として、コンピュータの組み立てやトラブルシューティング、およびそれらに関連する知見が豊富に得られることが挙げられる。例えば、以下ののような知見を得ることができる。

- PC パーツの世代の見分け方や、同一世代内での性能の素早い見分け方⁵
- PC ケース内の整線や、カスタマイズの行いやすいパーツの配置方法
- CPU やメモリのトレイなど、余剰パーツを収納するための特殊な部材の入手先⁶

このような一般的な知見について、プロジェクト内部のメンバーだけでなく、Web サイトなどを活用して外部にも積極的に公開していくことは、本プロジェクトそのものの広

⁵ 中には外見上非常に見分けづらいパーツが存在するが、型式などが記載されている部分を探せばすぐに世代を特定することのできるパーツ（メモリなど）も存在する。

⁶ たとえば、CPU トレイは国内に取扱いがなく、需要がニッチなため ebay などによる海外通販を利用するしかない。しかし、その海外通販もあまり知られていない。

報・発信の観点からも望ましいことである。このような情報の共有は 2023 年度までにあまり力を入れて実施できなかったことでもあるため、2024 年の活動においては特に積極的に推進されるべきことであると考えられる。

それだけでなく、例えば、コンピュータの組み立てを行ったことがないが、それに興味のある小～中学生を対象に、コンピュータのパーツの役割を説明しながら実際に組み立てていく講座を開設するなど、ワークショップ形式で情報発信を実施していくことも選択肢の 1 つとなるだろう⁷。

4－4．廃棄機材の他プロジェクトにおける活用

現在試験的に実施している事柄として、情報記録媒体について安全な処理を施した状態の廃棄機材を「インデペンデンス・サーバー・デイ」における「サーバ解体コーナー」、産学間連携推進室の主催する情報科学類生向け PC 分解講座・ネットワーク講座等に活用する取り組みが挙げられる。このような取り組みは、物理的にコンピュータという機器の内部に触れる機会が減ってきている近年の生徒ら・学生らにとって、またとない学習機会となるであろう。

4－5．今後の活動に向けての課題

現在、この活動は特に PC ハードウェアに関する知識を有する学生によって成り立っている。しかし、昨今のノート PC の高性能化やタブレット型 PC の出現などから、デスクトップ PC を組み立てたことのある学生の割合は年々減少しつつあるように思われる。しかし、研究の場で用いられるコンピュータは、依然としてデスクトップ PC が殆どであろう。

そのような状況下においても滞りなくこの活動を継続させるために、デスクトップ PC の組み立てなどについて興味のある学生へのアプローチを中心に、持続可能なプロジェクト運営に向けての方策の構築に、今後十分に時間をかけて取り組まなければならないと考えられる。2023 年度は幸いにも 4 名の新入生・2 名の学生を本プロジェクトに迎え入れることができたため、 신입生に対する技術指導を中心として、今後の活動の持続可能性を高められるような取り組みを進めていくことが求められるものと思われる。

⁷ 「夏休み自由研究お助け隊」や「インデペンデンス・サーバー・デイ」などとの協調実施も視野に入りうる。

5. 謝辞

本プロジェクトを進行するにあたり、システム情報系技術室 雨谷 恵 様をはじめとする、筑波大学内のさまざまな方々、プロジェクトメンバーの皆様、プロジェクト OB/OG の皆様のご支援・ご協力を頂いております。この場をお借りしまして、深く御礼を申し上げます。

WIDE Project つくば NOC 運用報告

太田弘樹 s2220573@s.tsukuba.ac.jp

菊田一真 s2420597@u.tsukuba.ac.jp

服部真吾 s2111559@u.tsukuba.ac.jp

2024-05-23

つくば NOC とは

つくば NOC は筑波大学術情報メディアセンター支援の元、情報科学類産学間連携推進室内サーバー室にて運用している NOC (Network Operation Center、ネットワークを構成する機材等を収容する施設) である。

本 NOC では通常の大学ネットワークから独立した別系統のネットワークを独自に運用しており、大学ネットワークでの運用や実施が難しいプロジェクトや実験 (不特定多数との間で大量のトラフィックを発生させるもの、大学外の一般に向けたサービスの運営、ハニーポットを用いたものなど) を実施している。また、この過程においてネットワーク運用に関する知見や技術の習得を目的としている。特につくば WIDE Project はこのつくば NOC を使用した主要な研究活動である。

つくば WIDE に関して

このつくば WIDE は 2007 年 11 月から産学間連携推進室により運用されている WIDE Project の拠点の 1 つである。WIDE Project は複数の大学など教育機関や企業の研究所などにより構成されたもので、大規模ネットワークの運用・研究のほか、安定した高品質なネットワークを一般の WIDE BB 利用者や研究者等に提供することを目的としている。つくば WIDE では、この WIDE Project の一部として、つくば地域において WIDE バックボーン (東京大手町) と接続された NOC を運用している。2014 年には CTF シリーズを開催することで新たに情報セキュリティ分野での活動が行われた。つくば WIDE では現在、以下のようなプロジェクトを行っている。

パブリックミラーサービスの提供

主に日本国内向けに Linux OS (CentOS, Debian, fedora など) や各種ソフトウェア (Apache, GCC, Vim など) の配信を行うミラーサーバーである。

30TB 超のストレージレイを構成し、イメージなどの提供を行っている。おおむね 1 日 150 万~200 万件のリクエストがある。

つくば WIDE において、このプロジェクトは最も大きなトラフィックの発生を伴うプロジェクトである。

IPv6 公開トンネル / 公開 NAT64 実験

VPN (Virtual Private Network) 技術を用いることで誰でも簡単かつ手軽に IPv6 接続環境を得られるようにするトンネリング型サービスを無料で提供している。50~100 セッション程度が確立されている。本プロジェクトはソフトイーサ株式会社と相互に協力し共同で実施している。

また、つくば WIDE では、DNS64 および NAT64 により IPv4 アドレスでの公開されているホストへの接続も提供している。この NAT64 のサービスはプレフィックスを WIDE ネットワーク内で広告しているため、IPv6 アドレスだけを割り当てたクライアントも IPv4 で提供される Web サイトなどに接続できるようになっている。この NAT64 は公開トンネルプロジェクトの利用者にも提供されるため、手軽に IPv6 Only のネットワークを体験することができるようになっている。

学内向け中規模ネットワークの運用

産学間連携推進室のほか、学術情報メディアセンター、情報科学類 WORD 編集部、新城研究室にまたがる学内中規模ネットワークを運用している。このネットワークは学内ネットワークとは独立した実験環境が必要な場合や NOC ではない各組織の部屋内でサーバーを運用したい場合などに使用している。また、このネットワークの運用を通してネットワーク運用に関する技術や新しい知見の習得を目指している。

構成

対外線接続

つくば NOC と WIDE バックボーンとの間はレイヤ 2 で接続している。接続は筑波大学学術情報メディアセンター内のアクセススイッチと WIDE プロジェクトバックボーンとの間では SINET の L2 接続サービスを、学術情報メディアセンター内と産学間連携推進室内を結ぶ通信は学内ネットワークの光ファイバーをお借りして接続している。

なお、学内ネットワークの借用と SINET の L2 接続サービスの利用は学術情報メディアセンターの多大な支援により実現しているものである。

経路

アドレスレンジ

IP アドレスは以下の範囲のアドレスを使用している。

- IPv4 アドレス: 203.178.132.64/26
- IPv6 アドレス: 2001:200:0:7C00::/56, 2001:200:1C8::/48

経路交換

上記の対外線に関する記載の通り、筑波大学及び SINET とは直接的にルーティングを行える箇所は存在せず、経路交換は行っていない。経路交換は WIDE バックボーンとの間でのみ行っている。ルーティングプロトコルは IPv4 接続においては OSPF を、IPv6 接続においては OSPFv3 を用いている。

設備更新

パブリックミラーサービスの設備更新

昨年度のところでパブリックミラーサービスの機材更新を行った。これはサービス提供開始から運用してきた機材の老朽化から機材トラブル（特にストレージに関するもの）によるサービス提供停止が目立っていたためであり、更新を行い現在は安定稼働している。なお、本サービス及び更新の詳細は別途案内する。

物理経路変更

昨年度までのところでは、学術情報メディアセンターと産学間連携推進室間の接続は学内ネットワークの VLAN 接続を利用して接続していたが、この回線の制約として通信速度が 1Gbps までしか出ない状況であった。そのため昨年度行ったアップリンク 10Gbps 化プロジェクトの一環として、学術情報メディアセンターの協力を得て、学内光ファイバーをお借りし、学術情報メディアセンターと産学間連携推進室間を直接接続することができたため、10Gbps またはそれ以上の速度に対応可能な接続となった。

アクセススイッチの更新

産学間連携推進室内の LAN を含む、つくば NOC の運用に用いているスイッチ類について、一部機材の更新を実施した。このアクセススイッチの更新に伴い、WIDE Project とのアップリンクは 10Gbps に強化された。また、パブリックミラーなどの対外サービスも、10Gbps で利用できるようになったためこれらのサービスの利用が更に快適になった。

つくば NOC の今後

WIDE Project や SINET との直接接続

これはまだ検討段階ではあるが、つくば WIDE として AS を運用したりし、筑波大学を経由した WIDE Project との接続ではなく、一組織として独自に接続する検討をしている。

現在の接続が筑波大学経由である以上、筑波大学自体の運営や各研究機関どうしをつなぐ通信が重要であり、我々の行う実験等でこれらを圧迫してしまうことはこちらとしても避けたいと考えている。

そこで筑波大学とは物理的にも完全に独立したネットワークで WIDE Project と接続することでこういった

懸念を払拭することができると考えている。また、筑波大学全体の帯域幅に影響されることなく通信を行うことができるため、より高速なネットワーク (100Gbps クラス) の運用に関する知見や技術の習得が行えると考えている。

筑波大学学内との直接通信

筑波大学学内での研究活動でもつくば NOC のパブリックミラーサービスを利用しているという声を時々聞く。しかし、現状、つくば NOC で行っているその他のサービスも含めて、通信先は学内にあるにも関わらず、筑波大学学内からの通信はつくばから出て、大手町のデータセンターを一度経由し、つくばへ戻ってきている。これは双方の他所行きの通信を圧迫するだけでなく、遅延の増加も招き非常に非効率である。そこで筑波大学と経路交換を行い、学内で完結可能な通信は学内で完結させたいと考えている。これにより、当パブリックミラーで提供しているソフトウェアを利用している研究ではより高速に環境構築やソフトウェア更新などができるようになり、通信速度に研究活動が律速される機会が減少し、研究支援につながると考えている。

つくば WIDE Project 公開ミラーサービス

太田弘樹 s2220573@s.tsukuba.ac.jp

2024-05-24

つくば WIDE Project 公開ミラーサービスについて

つくば WIDE Project 公開ミラーサービス (以下ミラーサービス) は現在産学間連携推進室に場所をお借りし運用している WIDE Project つくば NOC によって広く一般に提供しているサービスの一つである。このミラーサービスでは主に日本国内向けに Linux ディストリビューション (CentOS, Ubuntu, Raspberry Pi OS など) やソフトウェア (GCC, Vim など) の配信を行っている。1 日あたり、おおむね 150 万~200 万件のリクエストがあり、WIDE Project つくば NOC において、このミラーサービスは大きなトラフィックの発生を伴うプロジェクトであり、様々なオープンソースプロジェクトの支援を行うとともに、つくば NOC に定期的に通信が発生する環境を提供している。

機材構成

- MB/Chassis: Dell PowerEdge R530
- CPU: Intel® Xeon® CPU E5-2630 v3 @ 2.40GHz
 - Thread(s) per core: 2
 - Core(s) per socket: 8
 - Socket(s): 2
- RAM: 384GB
- Storage
 - RAID: Broadcom / LSI MegaRAID SAS-3 3108
 - DISK: 6TB HDD x8 RAID6 約 33TB
 - * 内約 27TB 使用中 (82%)

機材更新後の運用状況

昨年度のところで、機材の故障に伴う大規模なサービス中断が発生したため、機材を更新した。更新直後は設定作業などで短時間の中断はあったものの、現在は安定稼働しており、大学の法定点検に伴う停電時を除き、ほぼ停止することなく安定稼働している。

ミラーサービスの技術

ミラーサービスの同期は多くの場合一般的なファイル同期用のアプリケーションで行われており、同期の都度、すべてのファイルの更新状態を確認し、変更のあったファイルを同期している。しかし、ミラーサービスの同期の特徴として、大半のファイルは変更がなく、ごく一部のファイルのみ変更追加されるという点がある。そこで、今後はミラーサービス同期に特化したファイル同期用のアプリケーションの拡張などを研究していきたいと考えている。

不審なアクセス

本ミラーサービスの利用状況や動作状況の確認のために、アクセスログを収集し人間の目視によって定期的に確認を行っているが、一部のホストから多重にアクセスが行われている事象を時々確認できる。この内、新たにミラーサーバーをセットアップしようと本ミラーサービスからコピーしているような動作や、実際のアプリケーションの更新などに伴うアクセスはこのミラーサービスの運用目的と合致しているため問題ないが、一部、スピードテストの利用方法や、定常的なトラフィックの生成、または本サービスに過剰に負荷をかけるために繰り返しアクセスされている事があるようである。

そういったアクセスは、本サービスの主な目的であるソフトウェアの配布に影響を及ぼしかねないことから、気付き次第アクセス拒否を行っているが、IP アドレスを共有している環境なども考慮し、できる限り本来の利用に影響の出ないように行っている。

アクセス傾向

先で述べた不審なアクセス傾向に付いてだが、これはサーバー上のある1つのファイルに対して繰り返し取得を行うものである。当該ファイルは `http://ftp.tsukuba.wide.ad.jp/Linux/centos/7.9.2009/isos/x86_64/CentOS-7-x86_64-DVD-1511.iso` で公開されているものになるが、これは CentOS 7 の最終バージョンをインストールするためのディスクイメージであり、約 9.5GB の大きさがある。本サービスでは多種多様、さまざまなファイルをホストしているが、なぜ、このファイルのみへ大量のアクセスが行われるのかははっきりとした理由はわからないものの、ある程度の仮説は立てることができる。

ファイルサイズ

まず、ファイルの大きさである。先述の通り様々なファイルをホストしているが、ファイルの大半はアプリケーション・パッケージであり、1 ファイルあたり数キロバイトから大きくても 100MB 程度であり、これらのファイルを用いて大きなトラフィックを生成しようとする、多数のリクエストを送信する必要がある。しかし、OS のインストールイメージはその OS の動作に必要な多数のパッケージや、よく利用されるパッケージを一つのディスクイメージに収録してある。そのため、サイズが大きくより少ないリクエストで、多くのトラフィックを生成できるためであろうと考える。

公式ミラー

また、公式ミラーであることも要因の一つとして考えられる。ミラーサーバーは同期元がファイルを公開していれば、自らの構築したサーバーにコピーして公開状態にすれば、原則として自由に立てることができる。しかしそのままでは、世界中の人はその構築したサーバーがあることを知ることはできませんから多くの場合自分しかアクセスする事はありません。この状態を公開ミラーと区別してプライベートミラーと呼ぶ。では、どのようにして公開ミラーになり、世界中の人から利用してもらい、ひいてはそのソフトウェアプロジェクトを支援するかというと、そのプロジェクトに連絡し、公式サイトなどに構築したミラーサービスを掲示してもらうのである。この掲示されている状態を公開ミラーと呼び、そのプロジェクトを利用する人などが、掲示をもとにアクセスしてくるのである。さて、ではなぜ公式ミラーであることが原因と言えるのであろうか。これは、アクセスしているファイルのアドレスに注目する。本サービスでは異なったファイルパスで全く同じプロジェクトのファイルを公開しているものがあるが、アクセス対象になっている CentOS も該当しており、<http://ftp.tsukuba.wide.ad.jp/Linux/centos/> と <http://ftp.tsukuba.wide.ad.jp/centos/> の2つのアドレスで公開されている（/Linuxが含まれているかどうか異なる）。クローラなどで横断的にアクセスし、大きなファイルを探索していとすれば、どちらのアドレスへアクセスされても良いと考えるが、不審なアクセスにおいては、決まって前者の、/Linux の含まれたアドレスへアクセスされる。この2つのアドレスの違いは、前者のみが、公式ミラーとして登録されており、後者はいわゆるプライベートミラーの状態であるから、この違いから、不審者は公式ミラーを調べ、大きなファイルへの多重アクセスを行っているものと考えられる。

アクセス元

本ミラーサービスへは、日本国内からのアクセスが主ではあるものの、様々な国からのアクセスがある。しかし、このような不審なアクセスは確認できている範囲では中国からのみである。アクセス元は単一の IP アドレスではなく、多くの場合複数の似たような IP アドレスからである。この使用されるアドレスレンジにも少し特徴があるように見える。不審なアクセス元を whois で確認すると、登録されているアドレスレンジ全体からまんべんなくアクセスされるわけではなく、そのレンジの中の一部のレンジからアクセスされているようである。

転送量

本サービスではログの項目として、そのリクエストに対して応答したデータ量も記録を行っているが、これら不審なアクセスへの応答量は多くが数百メガバイトであり、ファイル全体、約 9.5GB には大きく届かないものである。HTTP レスポンスステータスコードも 200 であり、206 などの部分取得ではないことも確認できる。無論、転送に失敗し中断してしまい、再度取得を試みている挙動であることは否定できないが、もしインストーイメージを利用したいという目的であれば、複数回失敗すれば、別のミラーからの取得を試みるのが当然であり、このように多重にリクエストを繰り返す可能性は低いと言える。

具体例

この事案は 2024 年 4 月 16 日 22 時 24 分 53 秒を起点とする一連のアクセスについて述べる。IP アドレスは通信の秘密を守るため、この具体例では匿名化し記述する。

確認できる最もはじめのリクエストは、---.---.79.192 からのリクエスト 2 件であり、前述の CentOS インストーラーを取得するものである。このときのユーザーエージェント文字列は k6/0.50.0 (<https://k6.io/>) であり、これは負荷試験用のツールであることが確認できた。無論、負荷試験ツールの動作は本ミラーサービスへの事前通告無しで行われており、明示的な許可等もしていない。この 2 つのリクエストの転送量も 463MB と、ファイルの全体を取得し切る前に通信が打ち切られている。はじめの 2 リクエストのあと、約 30 分後の 23 時 06 分 32 秒に今度は curl/7.29.0 でのアクセスが始まる。この curl によるアクセスは、1 リクエストあたり 100-300MB ほどで推移していく。多いときは秒間 10 リクエストほどで、定常的には分間 10 リクエストほどである。また、アクセス元 IP アドレスは ---.---.79.96, ---.---.79.128, ---.---.79.192 の 3 つの IP アドレスから行われた。

その後、curl と同じ IP アドレス群から、k6 を利用したと思われるアクセスが追加される。これらも、リクエストあたり約 200-600MB の転送量しかなく、イメージ全体の転送はできていない。また、k6 でも最大秒間 10 リクエストほど、定常的に分間 200 リクエスト近く要求されている。

開始の翌日一日での集計を行ったところ、113562 リクエストを処理していた。つまり、一日でおよそ 45TB 近くこれらのホストへ転送していた計算になる。

また、4 月 17 日 16 時 33 分 59 秒からは ---.---.79.174 のホストも k6 を使用したリクエストに加わり、同 16 時 39 分 06 秒には ---.---.79.253 も加わった。

2024-04-22 からは ---.---.80.120 や、---.---.80.25 などリクエストに加わり、2024-04-24 には一日で 337064 リクエストが、これらのホストから送信されていた。一転送あたり 400MB と仮定すると、一日で 134TB 以上転送されていたこととなる。

翌 25 日には私が不審なアクセスに気づき、---.---.79.0/24 と ---.---.80.0/24 からのアクセス拒否を設定した。しかし、これ以降も直ちにリクエストが止むことはなく、26 日にも 365633 件のリクエストが行われた。

79.192	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	282098	-	16/0.50.0 (https://k6.io/)
79.128	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	202453	-	car/12.29.0
79.192	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	379109	-	16/0.50.0 (https://k6.io/)
79.96	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	461645	-	16/0.50.0 (https://k6.io/)
79.192	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	458749	-	16/0.50.0 (https://k6.io/)
79.128	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	495898	-	16/0.50.0 (https://k6.io/)
79.192	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	211141	-	16/0.50.0 (https://k6.io/)
79.96	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	382005	-	16/0.50.0 (https://k6.io/)
79.128	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	338585	-	16/0.50.0 (https://k6.io/)
79.192	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	350149	-	16/0.50.0 (https://k6.io/)
79.96	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	212539	-	16/0.50.0 (https://k6.io/)
79.128	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	461545	-	16/0.50.0 (https://k6.io/)
79.128	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	495853	-	16/0.50.0 (https://k6.io/)
79.128	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	468835	-	16/0.50.0 (https://k6.io/)
79.128	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	342309	-	16/0.50.0 (https://k6.io/)
79.192	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	377681	-	16/0.50.0 (https://k6.io/)
79.192	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	374785	-	16/0.50.0 (https://k6.io/)
79.128	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	382005	-	16/0.50.0 (https://k6.io/)
79.96	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	384221	-	16/0.50.0 (https://k6.io/)
79.128	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	381733	-	16/0.50.0 (https://k6.io/)
79.192	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	202453	-	16/0.50.0 (https://k6.io/)
79.192	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	288333	-	16/0.50.0 (https://k6.io/)
79.96	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	383453	-	16/0.50.0 (https://k6.io/)
79.96	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	341481	-	16/0.50.0 (https://k6.io/)
79.96	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	458853	-	16/0.50.0 (https://k6.io/)
79.96	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	398485	-	16/0.50.0 (https://k6.io/)
79.128	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	288021	-	16/0.50.0 (https://k6.io/)
79.128	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	296578	-	16/0.50.0 (https://k6.io/)
79.192	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	547077	-	16/0.50.0 (https://k6.io/)
79.128	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	277749	-	16/0.50.0 (https://k6.io/)
79.96	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	212539	-	16/0.50.0 (https://k6.io/)
79.192	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	377681	-	16/0.50.0 (https://k6.io/)
79.96	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	534045	-	16/0.50.0 (https://k6.io/)
79.96	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	218381	-	16/0.50.0 (https://k6.io/)
79.192	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	380357	-	16/0.50.0 (https://k6.io/)
79.192	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	528253	-	16/0.50.0 (https://k6.io/)
79.192	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	211141	-	16/0.50.0 (https://k6.io/)
79.96	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	200737	-	16/0.50.0 (https://k6.io/)
79.128	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	463098	-	16/0.50.0 (https://k6.io/)
79.128	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	332773	-	16/0.50.0 (https://k6.io/)
79.128	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	463098	-	16/0.50.0 (https://k6.io/)
79.96	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	342899	-	16/0.50.0 (https://k6.io/)
79.96	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	377681	-	16/0.50.0 (https://k6.io/)
79.192	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	268091	-	16/0.50.0 (https://k6.io/)
79.128	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	211141	-	16/0.50.0 (https://k6.io/)
79.96	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	292229	-	16/0.50.0 (https://k6.io/)
79.192	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	384901	-	16/0.50.0 (https://k6.io/)
79.96	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	338837	-	16/0.50.0 (https://k6.io/)
79.128	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	451509	-	16/0.50.0 (https://k6.io/)
79.128	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	387833	-	16/0.50.0 (https://k6.io/)
79.128	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	332773	-	16/0.50.0 (https://k6.io/)
79.128	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	267618	-	16/0.50.0 (https://k6.io/)
79.96	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	218829	-	16/0.50.0 (https://k6.io/)
79.128	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	377681	-	16/0.50.0 (https://k6.io/)
79.96	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	267618	-	16/0.50.0 (https://k6.io/)
79.192	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	240073	-	16/0.50.0 (https://k6.io/)
79.192	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	300317	-	16/0.50.0 (https://k6.io/)
79.128	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	442821	-	16/0.50.0 (https://k6.io/)
79.96	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	202453	-	16/0.50.0 (https://k6.io/)
79.128	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	280781	-	16/0.50.0 (https://k6.io/)
79.192	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	265268	-	16/0.50.0 (https://k6.io/)
79.128	-	[16/Apr/2024:23:54:02 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	377681	-	16/0.50.0 (https://k6.io/)
79.96	-	[16/Apr/2024:23:54:03 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	461437	-	16/0.50.0 (https://k6.io/)
79.128	-	[16/Apr/2024:23:54:03 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	473701	-	16/0.50.0 (https://k6.io/)
79.96	-	[16/Apr/2024:23:54:03 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	376213	-	16/0.50.0 (https://k6.io/)
79.128	-	[16/Apr/2024:23:54:03 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	202453	-	16/0.50.0 (https://k6.io/)
79.128	-	[16/Apr/2024:23:54:03 +0900]	GET	/Linux/centos/7.9.2009/iso/x86_64/DvdOS-7-x86_64-Everything-2009.iso	HTTP/1.1	200	428518	-	16/0.50.0 (https://k6.io/)

図1

その後しばらくは大量のアクセスに対して 403 Forbidden を返し続けていたが、5 月 17 日に試験的に HTTP レベルではなく IP レベルで拒否を行ったため、同時接続 500 セッションくらいで遷移していたものが、不審なアクセスが拒否され、一時的に 50 セッションくらいまで減ったものの、拒否解除後はアクセスが戻り、途絶えなかった。

同日 12 時過ぎから、アクセスパターンの変化がグラフから見受けられるので、なにかに気づき再試行等を行ったように見える。

5 月 17 日 18 時 42 分 12 秒を最後に k6 でのアクセスが止むが、同 45 分 59 秒から Go-http-client/1.1 でのアクセスが観測された。また、このあたりから IPv6 での同様なアクセスパターンが少数確認されたが、この IPv6 アドレスに関しては拒否を実施しなかった。時折、k6 でのアクセスが混じる事もあったが、大半は Go によるアクセスだった。

20 時 8 分 28 秒からは同 IP レンジで Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.0.0 Safari/537.36 からのアクセスを確認し、20 時 15 分 5 秒を最後に Go からのアクセスは途絶えた。

最終的に 23 時 35 分 56 秒を最後に、これらの IP アドレス群からのリクエストは途絶えた。

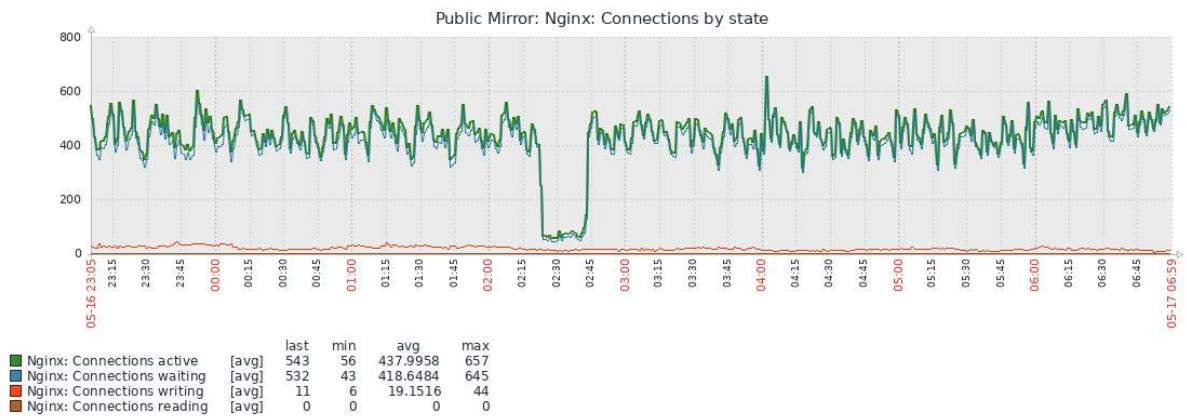


図2

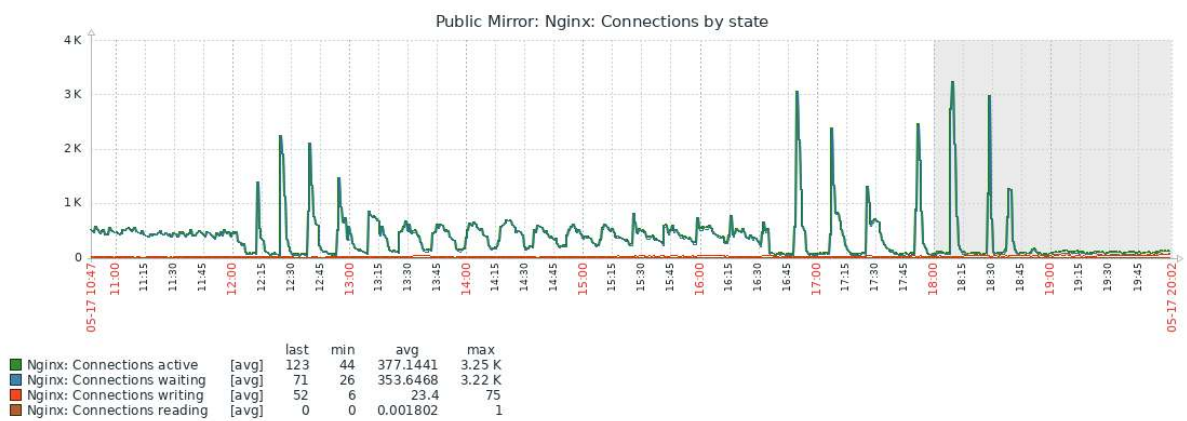


図3

最終的な IP アドレスとリクエスト数の一覧をいかに示す。

表1: IP アドレス一覧

リクエスト数	IP アドレス
813435	---.---.79.69
1167856	---.---.79.96
1310817	---.---.79.128
986129	---.---.79.174
1310823	---.---.79.192
1003040	---.---.79.253
963350	---.---.80.25
939887	---.---.80.61
200806	---.---.80.88
849402	---.---.80.120

691613	---.---.80.128
690457	---.---.80.160

ブロックの是非

様々な地域などから別け隔てなく利用してもらおうという考えや、現状 IPv4 アドレスが枯渇し、一つの IP アドレスを複数人で共有して使用している可能性があることから、できる限りブロックは避けたいと考えている。しかしながらその一方で、一部の利用者による偏った利用で他の利用者が不利益を被ることも避けたいと考えている。

そのため現在は .iso ファイルのみ、更に IP アドレスレンジを /24 またはより適切な範囲で拒否を行っており、定期的に拒否を解除したりしている。

一般的なファイル共有サイトなどでは人間からのアクセスのみを処理すればよいため、CAPTCHA などの技術を用いて機械的なアクセスを拒否することができるが、ソフトウェアパッケージなどは原理的に機械的にアクセスされるため、こういった技術を使うことはできない。

こういった中で本来の目的外の利用をうまく拒否できるような方法を今後も研究していく。

他組織との調査・協力

不審なアクセスの発生する条件の仮定や、アクセスパターンに基づくと、他の組織の運営するミラーサーバーにも同様なアクセスが行われている可能性は高い。実際に、不正使用されている IP アドレスを調査しているサービスである AbuseIPDB で今回の IP アドレスについて確認すると、我々ではない誰かが同様の事象を報告していることが確認できた。報告者が匿名である以上同様のサービスを運営するものかどうかはわからないが、他のサーバーへも不審なアクセスが行われたことは確からしいので、今後はこういったミラーサービス特有の不正利用などを、他のミラーサービスと協力して調査していきたいと考えている。

今後について

近年、データ配布の方法として CDN などの利用も増え、また、非営利プロジェクトへの理解のある CDN プロバイダなどもあり、これらのミラーサービスの存在を感じることも少なくなっているようである。しかし、依然として本サービスのようなミラーを使ったソフトウェアの配布方法も多く、利用の減少は見られない。現在約 27TB のデータをホストしている本ミラーサーバであるが、これからもサービスの安定提供に努めていくだけでなく、ソフトウェアの配布技術に関連する研究なども行っていきたい。

中学生向け産学官連携 ICT 教育イベント開催の総括

～ サーバ構築実習イベント「インデペンデンス・サーバー・デイ」の継続的な開催 ～

2024 年 5 月 26 日 筑波大学情報科学類 産学間連携推進室 活動成果報告会

ICT 教育イベントプロジェクト代表 根本 晃輔、 副代表 服部 真吾

協力：産学間連携推進室 佐藤大哲、関口亞聖、太田弘樹、金子尚樹

概要

産学間連携推進室においては、2016 年度より茨城県つくば市総合教育研究所¹、および筑波大学発ベンチャー企業数社²等と協力し、茨城県つくば市の中学生を対象とした実践的な産学官連携 ICT 教育イベント「インデペンデンス・サーバー・デイ」を運営している。本イベントは、各受講生に固有の物理サーバ（Raspberry Pi に代表されるシングルボードコンピュータ 1 台）とグローバル固定 IPv4 アドレス 1 つを配布し、受講生にネットワーク・サーバの動作原理、構築や運用の方法、運用上重要なセキュリティ対策等を学んでもらうもので、受講生は本イベントの開催期間中だけでなく、イベントの後も割り当てられたサーバを継続して自由に運用することができる。本イベントの目的は、高度情報化社会に顕著な「クラウド化」の傾向の中で、特に情報技術・IT インフラ技術に強い興味を持つ生徒らに対し、自主学習の難しい「物理的なサーバの動作や運用」の学習機会を提供し、更に高度な情報技術の学習に向けての橋渡しを狙うことにある。本報告においては、2023 年に開催した「インデペンデンス・サーバー・デイ 8」の開催報告、2016 年以降の継続的な開催の中での形式の変遷や考察、また 2024 年 8 月 21 日・22 日に開催予定の「インデペンデンス・サーバー・デイ 9」に向けての取り組みについて述べる。

1. 序論

2020 年度の小学校教育における「プログラミング教育」の必修化を契機に、近年、若年層へのプログラミング教育に対する注目・関心が急速に高まりつつある。

¹ つくば市教育局内の組織で、教育における情報技術の導入・運用を専門的に扱う組織。物理的な施設はつくば市大形 1333-1（旧大形小学校）に所在する。

² 2024 年 5 月時点での協力組織は、ソフトイーサ株式会社（代表取締役 登 大遊 氏・筑波大学情報科学類 2003 年度 AC 入学）、輝日株式会社（代表取締役 佐藤 大哲 氏・筑波大学情報科学類 2016 年度入学）、太平電算（代表 根本 晃輔・筑波大学情報科学類 2013 年度入学）である。本イベントの開催に当たり、各組織にはネットワーク資源・各種機材・運営学生への謝金等の協力を頂いている。

特に 2010 年代後半以降においては、官・民の別を問わず、多様な難易度の若年層向けプログラミング・情報技術教育イベントが開催されている。「セキュリティ・キャンプ」の知名度が向上し、応募者が年々増加傾向にあること、小中高生を対象とする「未踏ジュニア」が 2016 年にスタートしたことなどは特に記憶に新しい。

1-1. 高度な技術の学習機会の必要性

さて、官民の主体を問わず開催されている「プログラミング」に関連するイベントについて、簡易なビジュアルプログラミング環境である Scratch³や、ビジュアルプログラミングを介して物理的なパーツを動作させオリジナルのロボットなどを設計することのできる LEGO MINDSTORMS⁴、小中学生に高い人気を誇るサンドボックスゲーム環境を基盤として簡易なプログラミングに習熟できる Minecraft などの環境を用いて簡易的なプログラミングを学ぶようなものについては、さまざまな企業・団体により、日本各地で同様のものが開催されている。公立小中学校における「プログラミング教育」の運用、または部活動における「プログラミング」の取り組みの拡大もあり、小中学生の児童・生徒らにとって、簡易なプログラミングを学習するための土壌が成立しつつある。

しかし、これらの「世間一般における取り組み」のほとんどは、あくまで特定のプラットフォームの範囲内で Scratch / Python などの環境に閉じた・ごく簡易なプログラミングのみを取り扱うものであったり、運営主体が異なっても特定の教材・教具を用いたワンパターンなものばかりであったり、果ては「プログラミング未経験者可」として講師の求人がなされているものがあつたりと、イベント・講座内で技術の学習が閉じており、情報技術全体の学習への発展性のないものと言えるだろう。それらより高度な内容に当たる「テキストベースのプログラミング」や「サーバ・ネットワークの構築実習」「オリジナルのサービスの設計や構築・保守」を取り扱う、若年層を対象とし、なおかつ気軽に参加できるようなイベントは、2024 年 5 月現在においても、ほとんど存在しない⁵と言える。

³ 一般にはキャラクター（画像）の動きをプログラミングするものであるが、発展させてゲームを制作したり、動画作品のようなものを制作したりすることが可能である。

⁴ 2000 年代初頭から同名のシリーズが展開されているが、現在販売されているものは当時より高性能・高機能なシステムとなっている。また、一部の教育機関が実際に教育の場に取り入れているなど、活用事例も豊富になってきている。

⁵ 既存の「社会人向けプログラミングスクール」が小中学生向けに門戸を開くケースもあるが、24 回の授業で 70-80 万円近い高額な代金が発生すること、また講師の技術力に疑義があること、該当のスクールについて IT 技術者からの悪評が目立つことなどから、現実的な選択肢ではないといえる。

また、学校教育などの場における「プログラミング」の内容をほとんど理解してしまっている児童・生徒らにとっては、学校や各種のイベントが取り扱う内容より高度なことに取り組んでみようととしても、それを牽引してくれる、あるいは困った時に相談できる「IT 関連の正しい知識・技術を持った大人」を周囲に見つけることが難しく、結果として高度な知識・技術に触れる機会を失っているという現状が挙げられる。

学校・地域の特性如何では、学校に掲示されたポスター等から高度な技術イベントの存在を知るようなことさえも難しいケースも多い。国・関係機関が運営するようなイベントのポスターでさえ、学校には渡るが実際に掲示すらされない、というケースも散見される。このような背景から学校の支援・環境づくりに期待ができない場合、保護者に IT 関連の知識があり、各家庭において教育ができることが最善であるが、IT 技術に習熟した保護者の全体数はさほど多くはないと推測される。

中学生・高校生といった若年層がサーバ構築やネットワークに関する専門的な技術を学ぶための教育イベントの例としては、サーバ・ネットワーク等が強く関連する「情報セキュリティ」の高度な学習を主題とし、毎年 8 月に独立行政法人情報処理推進機構（IPA）によって開催されている「セキュリティ・キャンプ⁶」が存在する。しかし、セキュリティ・キャンプは全国の 22 歳以下の学生・生徒を対象としたものであり、年齢に基づく対象層は大学生・一部の大学院生までと幅広いことから、参加者全体に占める中学生の割合は非常に低い。また、参加に当たっては応募課題の回答内容による書類審査の過程も存在するため、中学生・高校生にとってはやや敷居の高い環境である。

なお、もちろん「セキュリティ・キャンプ」に積極的に参加している中学生・高校生は存在するものの、その絶対数は少ない。例えば 2018 年の開催回においては全体の参加者 85 名中、高等専門学校の参加者が 7 名、高等学校の参加者が 4 名、中学校の参加者が 1 名、中等教育学校の参加者が 1 名であった。

「セキュリティ・キャンプ」は、情報セキュリティの第一線に立つ技術者がそれぞれの専門分野の講義を行う、極めて高度な内容を扱うイベントとして知られている。情報技術に興味・関心がある中学生・高校生にとっては、先に述べたように、いきなり「セキュリティ・キャンプ」等の高度なイベントに応募することのハードルは相応に高いものとなるであろう。また、応募課題に対して回答するための知識・技術を身に着けるための場が身近

⁶ 情報処理推進機構「セキュリティ・キャンプ」>「セキュリティ・キャンプ事業紹介」

<https://www.ipa.go.jp/jinzai/security-camp/about.html>

にある中学生・高校生は決して多くない。最初から「セキュリティ・キャンプ」に参加できるような知識・技術を持ち合わせているような生徒でなくとも、将来的にこのような先進的なイベントへの参加を見据えた上で、受講生が自らの知識・技術を高めていけるような場が求められていると言える。

このように、児童・生徒らにとって実践的な学習機会が不足している、より詳細に言えば「学校教育における情報技術の学習に満足していないが、セキュリティ・キャンプのような高度なイベントに参加するための知識・技術を有しているレベルには至っていない」児童・生徒らが「より高度な学習機会にアクセスするための、学校教育より高度で実践的な知識・技術を得る」ための場がないという状況は、将来にわたって各種の技術を維持・活用するための高度な IT 人材の育成という観点から、大きな損失を招きうるのではないかと考えられる。

1－2．クラウド化の傾向、物理的なネットワーク技術の学習機会の減少

近年の情報化社会に顕著な「クラウド化」の傾向は、情報技術を専門としない一般的なユーザにも、容易かつ安価に大規模なコンピュータ・ネットワーク・サーバを取り扱うことを可能にした。2000 年代初頭から存在するような「レンタルサーバ⁷」だけでなく、AWS や Google Cloud Platform などの「クラウドコンピューティングサービス⁸」は急速に知名度を獲得し、商用の大規模なサービスに用いられている例も多く存在する。

確かにこれらのサービスを用いることで、何らかの Web サイト、あるいはアプリケーションを公開したいユーザにおいては、本来は大変複雑な「ネットワーク・サーバ」に関する操作を最低限とし、Web サイト、あるいはアプリケーションの開発そのものにリソースを集中させることができるだろう。しかし視点を変えれば、これらの技術を用いることで、ユーザはネットワーク・サーバ環境そのもののハードウェア・ソフトウェア的な動作について詳細に理解することなく、環境を利用するために必要な表層的な知識・操作方法のみを習得すれば良い状況が生まれてしまうと言えよう。したがって、これらの技術の発展は、それぞれの物理的な環境を動作させるための複雑な実装にユーザが直接触れる機会を減少させるものとなりうるのである。レンタルサーバ、クラウドコンピューティングサ

⁷ 一例として、さくらインターネット株式会社による「さくらのレンタルサーバ」は、年額 1,500 円ほどから利用できるなど、低価格で利用しやすいことで有名である。商業用途に利用できるエディションも提供されている。

⁸ これらのクラウドコンピューティングサービスは、利用したい時に利用したいだけの資源を都度課金にて利用できるという手軽さから、個人用途から商業用途まで幅広く用いられている。

ービスに関連する諸技術の発展以降、情報技術に強い興味や関心を持つ子供らにとって、物理的なサーバの動作や、インターネットとの直接の繋がりを知覚する機会は次第に失われつつある。

1－3. ICT 教育イベント「インデペンデンス・サーバー・デイ」の運営

このような背景を踏まえ、産学官連携推進室 ICT 教育プロジェクトでは、2016 年以降、グローバル IP アドレスと物理サーバを用いたサーバ構築・運用技術を学ぶ産学官連携 ICT 教育イベント「インデペンデンス・サーバー・デイ」を毎年継続的に運営している。本イベントは、つくば市総合教育研究所における通年の ICT 教育イベント「キッズプロジェクトプログラミング講座」の一環として実施されているものであり、「インデペンデンス・サーバー・デイ」は小中学生の夏休み期間に開催されるイベント「つくばプログラミングフェスタ」内の 1 講座として開催されているものである。

本イベントにおいては、将来の優れた IT 人材となりうる子供らに、サーバの基本的な構築方法や動作原理、セキュリティなど各種の先進的な問題について学習する環境を提供することを目標としている。

以降の章においては、本イベントの通算 8 回目の開催に当たり、2023 年 8 月に開催した「インデペンデンス・サーバー・デイ 8」に関する報告、2016 年以降の各回イベントにおける実施形式・講習内容の変遷や考察、また 2024 年 8 月に開催が予定されている「インデペンデンス・サーバー・デイ 9」に向けての取り組みについて述べる。

2. 「インデペンデンス・サーバー・デイ 8」の開催報告

2－1. 2023 年度開催「インデペンデンス・サーバー・デイ 8」の概要

産学官連携 ICT 教育イベント「インデペンデンス・サーバー・デイ 8」は、2023 年 8 月 23 日・24 日の 2 日間、両日とも 9:30 - 16:00 につくば市総合教育研究所において「つくばプログラミングフェスタ 2023」の 1 講座として開催され、つくば市の中学生 11 名が参加した。なお定員 11 名に対し、応募者数は 50 名ほどであった。

2018 年以降、本イベントは事前のポスター・パンフレットなどでの広報に注力している。特に 2022-2023 年度においてはつくば市の全中学生の人数の 1% に当たる 50-60 名ほどの応募を記録したことから、本活動が広く周知されている様子を窺うことができ、広

報によって特に大きな成果を得られたと言える。例年見られる状況ではあるが、申し込み開始当日に申し込みを行う受講希望者、(自己申告によるが)既に自宅でサーバを構築・運用した経験のある受講希望者が応募者に多くみられるなど、極めて意欲的な受講生が多かったように思われる。

なお、「インDEPENDENS・サーバー・デイ 8」の開催告知に利用したリーフレットは、次頁掲載の通りである。本リーフレットは、産学間連携推進室所属の学生のデザインによって、また協力組織の資金負担によって製作され、つくば市内の全ての公立中学校の生徒 6,500 名ほどに配布された。なお、同様のデザインであるポスターも、つくば市内の全ての公立中学校に掲示された。



INDEPENDENCE SERVERS DAY 8
- REVIVAL -
インDEPENDENS・サーバー・デイ・8
～つくば市中学生対象 サーバー・固定IPアドレスふれあいイベント～

2023.08.23^(水)/24^(木) 09:30～16:00
両日参加必須

1人1つ固定グローバルIPアドレスとサーバーを配布

自分専用の固定グローバルIPアドレスとサーバーを持ち、インターネットに専用線でこれらを直結し、世界に向かって自由にサービスを公開したいと思っただけではありませんか。このイベントでは、参加者の皆様に1台ずつ、サーバー(Raspberry Pi)を配布し、筑波大学のコンピュータ系の学生による講習のもと、自由にサーバーを構築していただきます。また、固定グローバルIPアドレスを1人1台配布し、イベント終了後も、各自のサーバーを設置しておける電源およびインターネット専用接続環境をつくば市総合教育研究所内に用意します。「レンタルサーバー」や「クラウド」、「NATの内側」では味わえない、真のサーバーおよびインターネット直結環境を用いて、高度でセキュアなサーバー構築の体験をしましょう。

講座内容 ● サイバーセキュリティ専門家による安全のための講義 ● インターネットサーバー構築に関する講義および実習 ● 専用設備内に設置されたサーバーの継続運用 講師：筑波大学の情報科学類および大学院情報理工学学位プログラム等関係者数名	開催情報 - 対象：つくば市に在学している中学生 - 定員：10名(超過時は抽選となります) - 料金：無料 - 場所：つくば市総合教育研究所 3F(予定) ※ ご自分のPCをお持ちの方は是非ご持参ください。
---	--

詳しくは <https://tsukuba.open.ad.jp> お申込み <https://forms.office.com/r/hxxRzxNNAj>

※ つくば市総合教育研究所 ※ 筑波大学 情報学群 情報科学類 ※ 筑波大学 OPEN プロジェクト ソフトイース株式会社 輝日株式会社 太電電

図1：「インDEPENDENS・サーバー・デイ・8」のリーフレット

2-2. 「インDEPENDENCE・サーバー・デイ 7」の主催・後援・協力組織

本イベントの主催組織は、つくば市総合教育研究所である。後援組織である筑波大学情報科学類には、講座において利用する機材の貸与や費用の支援を受けているほか、産学間連携推進室の学生が本イベントの直接的な運営に関与している。協力組織である筑波大学 OPEN プロジェクトおよびソフトイーサ株式会社、輝日株式会社には、同様に機材の貸与や費用の援助、グローバル IP アドレスなどの提供を受けている。

本イベントの実施に当たっては、ソフトイーサ株式会社によってつくば市総合教育研究所の実習室内に光ファイバ線（フレッツ光）の引き込みが行われ、筑波大学 OPEN プロジェクトのネットワークを利用できる状態とすることで、現在に至るまで「インDEPENDENCE・サーバー・デイ」において受講生が構築したサーバを運用するために活用されている。

また、産学間連携推進室内の「情報機器資源リユースプロジェクト」を中継し、筑波大学システム情報系技術室より、受講生が遠隔操作の実習・調べ物などで一時的に利用するための、情報機器資源リユースプロジェクトによってリユースされた Windows ノート PC 8 台（図 2 にその一部を示す）、ディスプレイ 13 台の貸与を受けている。



図 2：情報機器資源リユースプロジェクト・技術室経由で貸与を受けた Windows ノート PC

更に、イベント当日に受講生の実習の補佐をするスタッフとして、開催当初より筑波大学情報科学類の学類生有志、情報メディア創成学類の学類生有志、情報理工学位プログラム（旧・コンピュータサイエンス専攻）大学院生有志の協力を受けている。

2－3．対象とする受講生について

本イベントの対象は、つくば市内に在学する中学生で、2023 年 8 月 23 日および 8 月 24 日 両日 9:30～16:00 の時間帯の全てを通して対面形式のイベントに参加できる、以下のスキルと意欲を持つ生徒とした。

- パソコン操作（キーボード、インターネット検索）ができること。
- 「TCP/IP」、「ポート番号」などの用語の意味・概念が少しわかること。
- インターネットサーバー構築に興味があること。

2－4．講習内容について

本イベントでは、参加者のそれぞれに小型の SBC である Raspberry Pi 3B を 1 台ずつ配布し、Raspberry Pi 上の Raspberry Pi OS (debian ベース、旧称 Raspbian) で動作する Apache Web サーバを用いた基本的なサーバ構築手法と、HTML および CSS を用いた簡易な Web サイトの作成方法に関する講義・実習を行った。講習の様子を収めた写真を、次頁の図 3 に示す。

また下記に、講座において取り扱った内容の概略を箇条書きで示す。

- サイバーセキュリティ専門家による、公開サーバのセキュリティの講義・諸注意
- コンピュータの動作原理および OS の基本的な概念、および Linux の基礎についての講義と、ターミナルを用いたコマンドによるディレクトリ・ファイル操作の実習
- グローバル IP アドレスとホスト名の設定を実習形式で行ったのち、ping、traceroute、nslookup などのネットワークコマンドの操作方法に関する実習
- Tinker Board への Apache のインストールと基本的な設定、およびサーバ上への HTML コンテンツの配置
- HTML と CSS の記述による Web サイト制作の実習
- 実際にサーバをラックに設置したのち、サーバへの SSH・SCP による遠隔接続の実習、および遠隔操作方法に関する説明
- 自由な質疑応答コーナー「なぜなにコーナー」



図 3：「インDEPENDENS・サーバー・デイ・7」開催の様子

なお、当日の講習においては、それぞれについてオリジナルのスライドを作成し学習用の教材とした。「コンピュータの動作原理および OS の基本的な概念、および Linux の基礎についての講義と、ターミナルを用いたコマンドによるディレクトリ・ファイル操作の実習」は服部氏が、「グローバル IP アドレスとホスト名の設定を実習形式で行ったのち、ping、traceroute、nslookup などのネットワークコマンドの操作方法に関する実習」は太田氏が、「HTML と CSS の記述による Web サイト制作の実習」は金子氏が、それぞれ教材作成・講義の実施を担当した。また、それぞれの分野の専門書を受講生らに配布し、事後の学習に役立ててもらえるように配慮した。

イベントで受講生らが利用した Raspberry Pi はイベント以後もつくば市総合教育研究所内に設置し、受講生が遠隔ログインを行って引き続き実習を行えるようにした（図 4 にその写真を示す）。また、単純なサーバの構築手法を解説するだけでなく、近年の中学生の関心の強い話題を用いてサーバ・ネットワークについて受講生により強い関心を持ってもらうため、図 5 のようなスライドを用いて「なぜなにコーナー」と称する自由な質疑応答コーナーを設け、参加者の中学生から寄せられたさまざまな疑問・質問に講師陣が回答した。

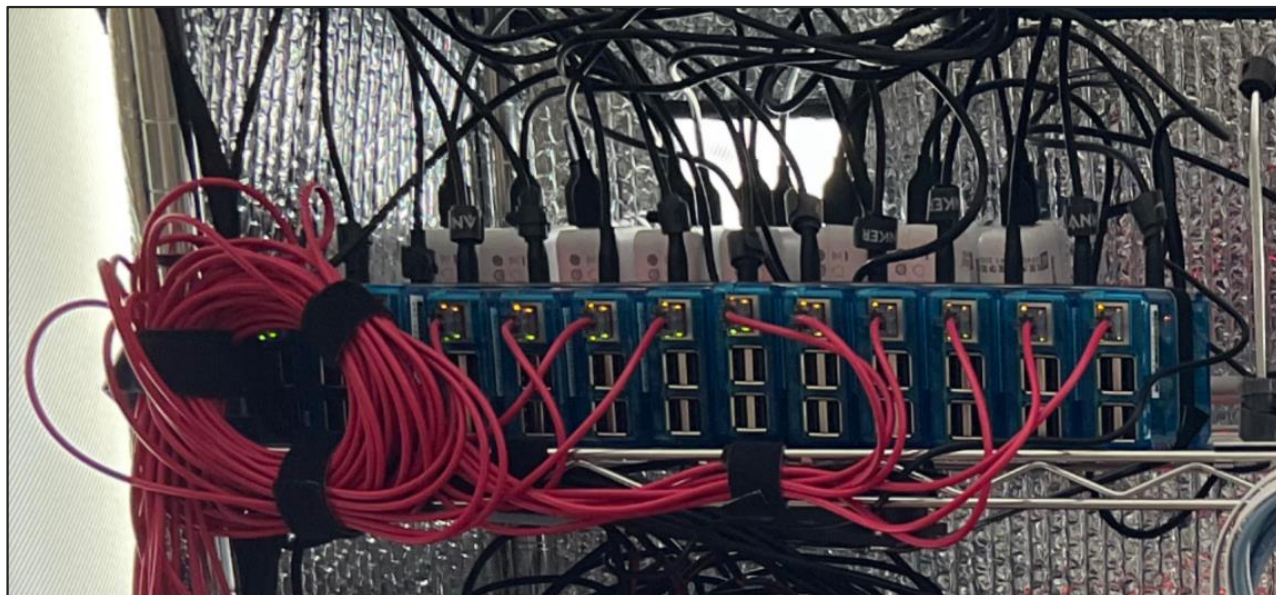
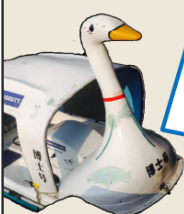


図4：「インデペンデンス・サーバー・デイ・8」において受講生が設置したサーバ

Q9.

情報技術に初めて興味を持ったのはいつですか？そのきっかけは何ですか？





中学一年生の終わり頃でした。**Microsoft Officeを消し飛ばしてしまい**代わりに入れた **L^AT_EX** が面白く、様々な拡張機能を作る過程で沼に沈んでいきました。[金子]

小学4年生の終わり頃に、父親のお下がりのノートPCをもらったところからこの道に入りました。**中学3年の春にITパスポートを取った頃は**まだ情報学の道に進むとは思っていなかったんですがね…。[梅津]

小学校後半で**家にFTTHが引かれてから**でしょうか。仮想化技術を使ってLinuxを触り始めた記憶があります。[太田]

33

[このスライドは 2023 年 8 月 23 日 18:00 時点の情報を元にしています / 各 FAQ の内容は講師個人の感想や経験に基づくもので、再現性は保証されません]

図5：「なぜなにコーナー」における質疑応答の例

また、本イベントはサーバの継続的な運用の実習の場としても活用できるようにするため、運用上の問題や技術的な質問などに回答するための連絡の場である Slack を設けるなど、受講生らの興味・関心を長期的に満たすことのできるような運営を行っている。図6に、Slackにおいて実施された受講生とのやり取りの例を示す。



図 6：Slack における受講生との継続的なやり取り

2-5. イベント実施を踏まえて

本イベントにおいては、各回の受講生に対して「講座の難易度」や「扱った問題の難易度」「講座への感想」などの項目によるアンケート調査を実施している。「インデペンデンス・サーバー・デイ・8」においても、本イベントは中学校におけるコンピュータ（技術家庭科）の授業を易しいもの・物足りないものと感じる受講生らにとって、学校の授業では扱わないような幅広い事項を 2 日で扱うというややペースの速い・ごく実践的な実習⁹であり、ある程度高い難易度でありながらも、楽しみながら実践的な内容を理解してもらえるイベントとなったと考えられる。

⁹ 筑波大学情報科学類において開設されている授業である「コンピュータリテラシ」を基準として考えれば、大学生が数週間かけて学習するような内容を 2 日でこなす内容となっていると言えるだろう。

3. 「インデペンデンス・サーバー・デイ」の各回実施における変遷

本イベントは、2016 年 8 月からの計 8 回の実施の中で、その講義内容・実施形態について応募者層の変化、受講生（あるいは応募者）の反応、学習機材の調達市況等に基づき、さまざまな変化が生じている。まずは「インデペンデンス・サーバー・デイ」の初回の開催の概要について示した後、2022 年度の開催に至るまでの変化の概要について、下記の通り示す。

3－1. 2016 年：「インデペンデンス・サーバー・デイ」初回の実施

2016 年 8 月 23 日に開催された「インデペンデンス・サーバー・デイ」には、定員 15 名に対し 5 名の受講生が参加した。当時は現在のようなリーフレットによる広報をほとんど行っていなかったこと、また他に類を見ないイベントであったことから、定員を大きく割り込んでの開催となった。

当該回におけるコンピュータやネットワークの概念に関する講義、サーバ構築の実習については、標準で利用するエディタが 2016 年の実施回のみ vi であったこと、また 2016 年当時は講座全体が 1 日での開催であったことを除き、2022 年とほとんど変わらない内容の構成にて行った。サーバ上に展開するコンテンツについては、運営側が用意した PHP による Twitter Bot を展開する、といったものであり、その難易度に比して当日の講義・実習に十分な時間を用意できなかったことから、受講生の知識・技術のレベルを実習ベースで引き上げることへの課題が残る実施となった。

なお、当初は受講生が興味を持ってもらえると推定される題材として選定した「Twitter Bot」は、参加者内での Twitter の利用率が極めて低かったこともあり、もとより受講生の興味・関心を集められるものでは無かったとも考えられる。

3－2. 2017 年：「インデペンデンス・サーバー・デイ」2 回目の実施

2017 年 8 月 2 日に開催された「インデペンデンス・サーバー・デイ 2017」には、定員 10 名に対し 10 名の受講生が参加した。当該年より本イベントはつくば市総合教育研究所の「キッズプロジェクトプログラミング講座」に組み込まれ、同講座が夏休みに開催するイベントである「つくばプログラミングフェスタ」において、プログラミング・情報技術を扱ういくつかの他の講座と同日の開催となった。なお、現在では日程の都合上、「つくばプログラミングフェスタ」の他の講座と本イベントは別日での開催となっている。

前年度の開催からの変更点は、大きく分けて 2 点である。

そのうちの 1 点目は、講習中に受講生が利用する（講習において利用を推奨する）エディタを vi から nano に変更したことである。vi は有用なテキストエディタであり、講師・スタッフの多くが操作に習熟しているが、vi に初めて触れる受講生にとって、これを短時間で使いこなせるようになることは難しいものであった。一方で nano は vi より操作を行いやすく、また実習環境・他の多くの Linux ディストリビューション環境においても同様に操作を行うことができるため、受講生にとっては格段にテキストの記述が行いやすくなったと言えるだろう。これにより受講生のコンソール環境におけるテキスト編集の簡易性が増し、各種の設定ファイルの編集等をよりスムーズに進めることが可能となった。

2 点目は、サーバ上に展開するコンテンツを HTML および CSS から構成される Web ページとし、運営側で用意したサンプルファイルを受講生が各自のサーバ上に展開し、それを用いて基本的な HTML のタグや CSS による装飾を学ぶという形をとったことである。しかし、2017 年度に運営側で用意した Web ページのサンプルは一般的な企業のコーポレートサイトに近い、やや複雑な構成のものであったため、そのサンプルを受講生が自在に改変したり、受講生が自身のコンテンツを設置したりする例はほとんど見られなかった。

このように、実習に用いるソフトウェア、および実習を通して取り扱う題材については、前年の実施回より簡易かつ基本的な技術を用いたものとなったものの、運営側で用意したサンプルが複雑なものであったため、受講生にとってはそのサンプルを活用した学習が難しかったものと考えられる。この反省は、2018 年度の開催に大きく活かされることとなった。

3-3. 2018 年：「インデペンデンス・サーバー・デイ」3 回目の実施

2018 年 8 月 1 日に開催された「インデペンデンス・サーバー・デイ 3」には、当初の定員 10 名に対し多数の応募があり、抽選の結果として 11 名の受講生が参加した。なお、当該年も本イベントはつくば市総合教育研究所の「つくばプログラミングフェスタ」内において、他の講座と同日での開催となった。本イベントは他の多彩な、なおかつ集客力が大きいと思われる講座と排他的な申込形態をとるイベントであったが、本イベントのみの応募者数は定員の 4 倍程度であり、前年に比して格段の盛況が見られた。

前年度の開催からの変更点は、大きく分けて 3 点である。

そのうちの 1 点目は、受講生用サーバとして利用する機材を Raspberry Pi 3B から Raspberry

Pi 3B+ に変更したことである。この変更は、2018 年 6 月に日本国内で Raspberry Pi 3B+ が発売され、受講生向けに最新の、より良い環境を提供したいというモチベーションが生じたことによる。これにより、確かに受講生が利用するサーバの処理性能はある程度向上したものの、実際に受講生が実習を行う上で体感できるような変化、あるいは Raspberry Pi 3B+ でなければ行えなかったような実習・取り組みの内容の変化は生じなかった。一方で、Raspberry Pi 3B+ は Raspberry Pi 3 よりも発熱量が多く、サーバーラック内に密集させたことによる過熱、また過熱による動作の停止が見られた。このことから、Raspberry Pi 3B+ を利用したのは当該年度のみであり、Raspberry Pi 4 が発売されて以降の 2021 年に至るまで、引き続き Raspberry Pi 3B が利用されることとなった。

2 点目は、HTML および CSS のサンプルとなる Web ページを非常に簡易なものとし、HTML のタグ構造を読み取りやすく、また編集しやすくしたことである。これにより、受講生がサンプルページを改変する形での学習に取り組みやすい環境を整えることができたと言える。しかし、2018 年時点ではイベント開催は 1 日のみであり、イベント当日の実習時間を十分に確保できなかったことから、HTML および CSS を用いた Web ページの実習については、その多くが受講生の自主学習の範疇に組み入れられることとなった。この反省を活かし、2019 年度以降に導入される 2 日間開催の形態が検討された。

3 点目は、受講生がイベント終了後も本イベントのスタッフに気軽に技術的な相談を行えるよう、更なる発展的な学びの場としての環境を設けるべく、従前より用意していたメールでの運営への連絡・質問に加え、グループチャットシステムである Slack を用いてスタッフ・受講生の交流の場を開設したことである。しかし、受講生らの家庭の方針などからグループチャットシステムに加入するということへの敷居が高かったこと、Slack というツールの非技術者層への認知度が低かったこともあり、結果的に Slack グループに加入した受講生は参加者 11 名中 3 名であった。

3－4．2019 年：「インデペンデンス・サーバー・デイ」4 回目の実施

2019 年 8 月 22 日・23 日に開催された「インデペンデンス・サーバー・デイ 4」は、前年度に講座の時間が不足したことを受け、2019 年 8 月 9 日に先んじて開催された「つくばプログラミングフェスタ」とは独立した 2 日間の連続的な日程にて実施する運びとなった。開催日程が異なることから「つくばプログラミングフェスタ」の他講座に申し込んだ受講生でも参加できる形態とし、講座当初の定員 10 名に対し 3 倍程度の申込があり、抽選の結果として 12 名の受講生が参加した。

前年度の開催からの変更点は、大きく分けて 2 点である。

1 つ目は前述の通り開催日程を 2 日間とし、受講生が HTML や CSS を用いて各自のコンテンツを制作する時間を 2 日目に集中して確保したことである。当日の運営スタッフが受講生の学習を適度に支援したことにより、受講生 12 名のうち 11 名がサンプルとなる Web ページを改変し、あるいは独自の Web ページを制作することによって、サーバ上に独自のコンテンツを展開することができた。

2 つ目は、受講生が自主学習において利用するために配布している市販の技術書の難易度を落とし、より平易なものとしたことである。具体的には、従来は HTML と CSS を用いた Web ページの「デザインの技法」について扱う書籍を用いていたところ、当該年度においては HTML のタグの説明や CSS の指定の説明などの基礎的な内容を中心に扱う書籍に変更したものである。これにより、先に述べた HTML / CSS に関連する事項の実習時間の確保と併せ、受講生が独自のコンテンツを制作・構築することを大きく支援することができたと言える。

3-5. 2020 年：「インデペンデンス・サーバー・デイ」5 回目の実施

2020 年 8 月 17 日に開催された「インデペンデンス・サーバー・デイ・5」は、同時期に流行した新型コロナウイルス感染拡大の影響を受け、従前の対面形式での開催とは異なり、Zoom によるオンラインでの開催となった。また、新型コロナウイルス感染拡大の影響により中学生の夏休み期間が短縮されたことから、前年度実施された 2 日間での開催は行われず、2016-2018 年と同様に 1 日開催の形式となった。中学生にとっては学業、私事など多彩なイベントが短い夏休みに集中したこともあってか、2020 年度の申し込み数は前年より少ない 20 名弱であったが、定員 5 名に対し倍率ベースで 3-4 倍程度の申し込みがあり、最終的に 6 名が参加したことから、抽選の倍率そのものは前年度と大きく変わらない状況であった。

2020 年度の「インデペンデンス・サーバー・デイ・5」においては、オンラインでの開催であったことから、Raspberry Pi の実機に触れてのネットワークの設定など、本講座の根幹を成す内容の一部を講師・スタッフが事前に行うこととなった。また 1 日開催であったことから、講座内で扱うことのできる内容も前年度より絞られることとなり、学習効果は前年の「インデペンデンス・サーバー・デイ」と比較し大きく低下してしまったのではないかとと思われる。

1 日開催という状況自体は 2018 年までの実施形態に戻った、と考えればそれまでであるが、2018 年までのイベントにおいて発生していた時間の不足は、本年においても発生することになってしまった。

3－6．2021 年：「インデペンデンス・サーバー・デイ」6 回目の実施

2021 年 7 月 29 日・30 日に開催された「インデペンデンス・サーバー・デイ 6」は、前年度の反省を生かし、新型コロナウイルス感染症対策を十分に実施した上で、Raspberry Pi を利用した物理的なサーバのセットアップ過程のみ、全受講生を半数ずつに分けての対面開催を行い、HTML や CSS による Web コンテンツの制作過程については全受講生が同時にオンラインで受講するというハイブリッド形式を採用して開催された。2019 年同様の「完全な 2 日間開催」とは至らなかったが、本講座における「物理的なサーバのセットアップ」という重要な部分を、対面で、受講生自らがサーバに触れる形で行うことができたことは 大変に喜ばしいことである。

なお、講習の内容は概ね 2019 年に準じる形となった。2019 年に引き続き「なぜなにコーナー」という自由な質疑応答コーナーを導入したことにより、受講生からの技術的な興味・関心に基づく積極的な質問が見られるようになった。また、短時間ではあるが「サーバ分解コーナー」という、実際にラックマウントサーバの中身を眺めることのできるコーナーを設けたのも 2021 年からの変更点である。

3－7．2022 年：「インデペンデンス・サーバー・デイ」7 回目の実施

2022 年 8 月 22 日・23 日に開催された「インデペンデンス・サーバー・デイ・7」は、2019 年以來の完全対面形式、および 2 日間全日での開催となった。ただし新型コロナウイルス感染拡大の影響から、受講生数は 2019 年同様とはならず、募集定員は 8 名に絞られた。なお、応募倍率は 7 倍近くと過去最高を記録し、実際の参加者は 2 名の欠席により 6 名となった。

2022 年においては、半導体の価格高騰・急激な円安の進行等により、Raspberry Pi を含めたシングルボードコンピュータの調達が著しく困難となった。次頁図 7 に、2022 年 7 月 22 日時点で Amazon.co.jp において記録した Raspberry Pi 3B の販売価格を、図 8 に同日時点での Raspberry Pi 3B の価格推移グラフを掲載する。

そのため、2022 年の「インデペンデンス・サーバー・デイ・7」の開催に当たっては、Raspberry Pi の代替となるシングルボードコンピュータを選定するべく多様なシングルボードコンピュータ製品の比較・検討（使い勝手、性能、2022 年当時の調達可能性の検証等）を行い、結果として 2022 年度のみ ASUS Tinker Board S R 2.0 を採用することとなった。



Raspberry Pi Raspberry Pi3
Model B Element 14

★★★★☆ ~ 7

¥19,980

2022/7/29 金曜日~2022/8/1 月曜日
にお届け
配送料無料
残り20点 ご注文はお早めに



Raspberry Pi 3 MODEL B RS UK製
正規代理店品 ラズベリーパイ 開発ボード

★★★★☆ ~ 29

¥19,880

2022/8/10 水曜日~2022/8/17 水曜日
にお届け
配送料無料



Raspberry Pi 3 Model b+ ラズベリー
パイ3b+ 32GB microSDカード/5V/3A スイッチ付き電源/HDMI
ケーブル/MicroSDHCカード...

★★★★☆ ~ 86

¥33,980

¥2,000 OFF クーポンあり

prime 無料翌日配達
明日中7/23中にお届け

図7：Raspberry Pi 3 ボードの価格（2022年7月22日に amazon.co.jp にて記録）



図8：Raspberry Pi 3B の価格推移（2022年7月22日に記録）

なお、2022年の「インデペンデンス・サーバー・デイ・7」は2019年の開催とほぼ同様の開催様式であったことから、2021年に追加された「サーバ解体コーナー」を除いては、2019年の開催当時より大きな変更点は生じていない。ただし、技術面においては ASUS Tinker Board S R 2.0 について、従前採用していた Raspberry Pi と概ね同様の扱いが出来るようにするため、Tinker Board S R 2.0 に導入する OS、また投入する設定等について、講師・スタッフを交えてさまざまな議論がなされた。

3－8．2023 年：「インデペンデンス・サーバー・デイ」8 回目の実施

2023 年 8 月 23 日・24 日に開催された「インデペンデンス・サーバー・デイ・8」は、2019 年以来の完全対面形式、および 2 日間全日、また受講生数を 2019 年段階の形式に戻しての開催となった。応募倍率は 5 倍近くと高い値を記録し、実際の参加者は 11 名となった。

2023 年においては、協力企業が Raspberry Pi 3B の安価・多数の調達に成功したことを受けて、利用するシングルボードコンピュータを再び Raspberry Pi 3B に戻す形がとられた。

講習の内容は概ね 2019 年に準じ、2019 年以降に導入された「なぜなにコーナー」、2021 年以降に導入された「サーバ解体実習コーナー（図 9）」等も維持することで、最も大規模に・充実した内容の講習を行うことができたものと思われる。



また、実際に IT 企業における技術者育成に関わっている IT 技術者の監修を受けたことにより、講座全体の品質を向上させることができたほか、実践的な情報技術学習の場の構築に強い興味・関心を有する技術者に実際に「インデペンデンス・サーバー・デイ」のサポートスタッフとして参加して頂くことにより、この取り組みを広報し、また実際の企業での実践的な情報技術学習の取り組みに活用してもらえるような足掛かりとすることができた。

3-8. 「インデペンデンス・サーバー・デイ」全体の変遷を通して

ここまで述べたように、「インデペンデンス・サーバー・デイ」は、2016 年の初回の実施から 2023 年まで 8 回の実施の間において、年ごとに前年度の反省を活かし、講義・実習内容を変化させている。全体の変遷の中での受講生を対象としたアンケート結果、および技術的な質問・サーバの利用状況の調査の結果やそれらの変化を照合することにより、講義・実習内容の変化が受講生にいくつかの変化をもたらしていることが読み取れる。その内容について、次の通り示す。

まずは 2019 年度の開催において、開催日程を 2 日間としたことにより、それまでの期間においては低かった「サーバ上への独自のコンテンツの展開」までを行うことのできている受講生、ひいてはイベント期間後にサーバを独自に活用している受講生の割合が急激に増加したことが挙げられる。2018 年までの実施回においては、7~8 割程度の受講生がサンプルファイルをサーバ上に展開したのみで学習を終えてしまい、また受講生に配布した書籍の難易度も高いものであったことから、受講生が独自にコンテンツを制作するための土壌が整っていなかった可能性が高いものと考えられる。

また、2016 年の初回実施時には PHP を利用した Twitter Bot を制作する、という中学生にとって難解な題材を取り扱っていたところ、2017 年から HTML および CSS を利用した Web ページの制作に題材を変更し、また 2019 年においては受講生に配布した書籍の難易度を従来より引き下げ、より基礎的な書籍を配布するように変更したことにより、受講生が新しいコンテンツを制作するためのイメージ作りや、そのコンテンツを制作するための技術習得に向かっていきやすい環境を整えることができたことが結果として表出しているように思われる。

次に、2018 年度以降に設けた受講生・スタッフ共用の Slack グループチャット、また 2019 年度以降に設けた「なぜなにコーナー」において、いずれも前年より多くの質問が寄せられていることが挙げられる。このようにコミュニティを設けることにより、受講生らにとっては情報技術に関する多様な知識・技術の習得の場を獲得でき、講座の内容をベースとした自主学習を行う際だけでなく、自主的に多様な技術を学習していく際のサポートを受ける環境を整備することができるのである。なお、2017 年までの間ではメールでの質問サポートのみを実施したものの、メールベースでの技術的サポートは受講生にとって難しかったようであり、実際に寄せられた質問の件数は 2016 年および 2017 年の 2 年間で 1 件のみであった。このことから、受講生のメールによる質問サポートを利用するモチベーションは相応に低かったことが読み取れる。

また、2019 年以降の開催においては技術的な質問の場として「どのような内容でも質問ができる」という趣旨の質疑応答コーナーを設けたことにより、メールベース・Slack ベースでの質問件数の倍近くの質問が寄せられたことも変化の 1 つに挙げられる。この質疑応答コーナーは、イベントの 1 日目のアンケート内に「何らかの質問を記入した場合は翌日の講習にて解説を行う」という趣旨の自由記述欄を設け、2 日目のイベントの最後にスライドを用いたプレゼンテーション形式にて質疑応答を行うものである。導入当初の 2019 年は 12 名の受講生から 8 件の質問が、最新の 2023 年は 7 名の受講生から 13 件の質問が寄せられ、その質疑応答を通して受講生に「講座の内容に限らない」情報技術に関する知識・技術を身に付けてもらう契機とすることができたように思われる。また、過去の世代の受講生からの質問や「何らかの資格に合格した」「Web サイトを作成した」などの報告も定期的にみられるなど、受講生が盛んに Slack を活用しようとしている様子もみられる。

4. 2024 年度の「インデペンデンス・サーバー・デイ」

2024 年度の「インデペンデンス・サーバー・デイ」は、2024 年 8 月 21 日 および 8 月 22 日に開催することが確定している。

成果発表会当日：2024 年 5 月 26 日時点においては、つくば市内の中学校に配布するリーフレットの作成途中であるため、その詳細が完全に確定している状態ではないが、2024 年の開催規模については、2023 年と同様のもの（定員 10 名・応募状況により 12 名まで受け入れ・2 日間開催）となる予定である。本イベントはつくば市をはじめ、関係組織からの期待も大きく、応募者数も極めて高い水準を維持していることから、2020 年から 2023 年までの「インデペンデンス・サーバー・デイ」の開催によって得られた知見：たとえば「なぜなにコーナー」の効率的な運用、スタッフ・受講生間の Slack（に限らないが）コミュニティでの関係性の醸成、サーバ・ハードディスクなどの解体コーナーの運用等を引き続き運用・発展させることを目標としていきたいと考えている。また、講座の本筋はもちろんのこと、講座から派生したさまざまな「興味を惹く」取り組みにも目を向けてもらえるよう、引き続き取り組みを進めていく所存である。

5. 総括と今後の課題

「インデペンデンス・サーバー・デイ」は、2024 年度以降も継続的に開催し、2016 年度～2023 年度と同様、産学間連携推進室のメンバーが講習資料作成や当日の運営に関与する予定である。「インデペンデンス・サーバー・デイ」の総括、および今後の本イベントの継続開催や、他種の発展的な ICT 教育イベントを行う上での課題について、次の通り記す。

5－1. 「インDEPENDENCE・サーバー・デイ」の総括

「インDEPENDENCE・サーバー・デイ」は、過去 8 年に亘る開催実績、およびその中で積み上げられてきた多様なノウハウの下、昨今の情報化社会において、地方自治体と地元の大学・ベンチャー企業が関与するという産学官連携の理想的な形の下、高い技術・知識を持つ意欲的な中学生に対し、より高度かつ実践的な技術・知識を学べる有意義な「情報技術者教育」の場を提供することができたと考えられる。受講者の満足度も高かっただけでなく、受講希望者ベースでは最初期に実施した「インDEPENDENCE・サーバー・デイ」の 12 倍ほどの規模となり、ポスター・リーフレットを用いた広報、また過去の受講者からの口コミ等も含め、大きな成功を収めたと言える。

一方で、現在の本イベントは 2 日の中にさまざまな要素を詰め込む形の開催であることから、内容の取捨選択や講義速度の調節について、緻密な計画が必要となる。本イベントは先に述べた通り、今年度以降も継続して開催する予定であるが、引き続き講座で取り扱う情報量の精査・検討を行い、「受講生の理解」を第一に据えた適切な進捗で実習を進められるような改善を行っていく予定である。

また、本イベントの受講希望者数、および受講生へのアンケートにおいて、前年度に引き続き、「つくば市の中学生」の実践的な IT 技術を取り扱うイベントへの参加の意欲が非常に高いことが分かった。アンケートの回答内容や、イベントの Web サイトへのインターネットコミュニティ上の反応などから、サーバ構築・運用について取り扱う本イベントの定期的な開催、また地域や年齢層などの規模を拡大しての開催だけでなく、以前産学間連携推進室が運営していた中学生向けイベントである「テキストベースのプログラミングに挑戦してみよう！」に代わり得るような一般的なプログラミング言語を用いたプログラミングの実習・アプリケーションの制作や、OS やネットワークの詳細な事項について掘り下げて取り扱うような多種の実践的なイベントを開催することが、将来活躍するであろう高度な IT 人材の育成に強く求められているのではないかと考えられる。

5－2. 他種の講座の開設について

「インDEPENDENCE・サーバー・デイ」、またかつて開催していた「テキストベースのプログラミングに挑戦してみよう！」の両講座における受講生向けのアンケートを通じ、情報技術に高い関心を持つ中学生から、引き続き多様な実践的技術を取り扱う講座の開催を望む声が挙がっていることが分かっている。

また、著者は中学生向けに筑波大学が実施している「夏休み自由研究お助け隊」内の講座「Arduino を使ってプログラムしてみよう」におけるメイン講師・教材作成担当を 8 年間務めているほか、「つくば SKIP アカデミー」において同系統の講座である Arduino プログラミング講座を運営した経験もあること、また 2017 年頃より一般企業における技術研修を業務として行っていることなどから、高度な意欲を持つ中学生向けの ICT 教育に関する知見が豊富に蓄積されている状態にある。

2024 年度は「インデペンデンス・サーバー・デイ」の継続開催を軸におきつつ、つくば市総合教育研究所を始めとする関係組織と連携しながら、より幅広い参加者を募ることのできる他種の講座の開催・「インデペンデンス・サーバー・デイ」の内容の洗練・マテリアルの整備という 2 つの目標を設定し、本活動をより発展的なものとしていくことを検討している。

新規に開設する講座の具体的な候補としては、下記のようなものが挙げられる。

● 「情報機器資源リユースプロジェクト」によって回収された廃棄 PC を活用し、PC の分解や組み立ての実習を行い、併せて PC を構成するさまざまな部品について解説する短時間・単発の講座を開催する。短時間であることから気軽に参加できることが想定されるほか、低予算で繰り返し実施でき、親子での参加等も可能であるため、実施までのハードルが極めて低い。また、現在産学間連携推進室が学内向け（新入生向け）に行っているイベントの知見の応用が可能である。

● 「openfab 創房」において設置されている機材を用いた、デジタルなものづくりを体験できる講座を開催する。システム情報系技術室の支援を得られる状況にあるほか、「夏休み自由研究お助け隊」での内容に近い実習形式となるため、こちらも実施までのハードルが低く、なおかつ一般的な中学生の興味を惹ける内容ともなっている。

● 簡易に取り扱えるマイコンボードである Arduino を活用し、ブレッドボードを用いた簡易な電子工作と組み合わせ、触って遊べるおもちゃ・ゲームを制作する。ある程度限られた予算・受講期間であっても、早押しゲームや記憶ゲームなど、受講生の記憶に残るような楽しい作品を制作することが可能である。性質上、先述の「夏休み自由研究お助け隊」と近い内容となることが想定される¹⁰。

¹⁰ 「夏休み自由研究お助け隊」では 2.5 時間の枠内で、Arduino を用いた LED の点灯から、スイッチと LED、ブザーを用いた記憶ゲームの制作までを取り扱っている。なお、一部に筑波大学情報科学類開設科目である「情報科学基礎実験」の内容と共通している部分があり、この点でも中学生にとっては応用的な内容を扱っていると捉えることが可能である。

● Java 言語や C# 言語などを活用し、Scratch で実装するようなキャラクターの描画を伴うプログラムを記述する。Scratch を既に体験したことのある生徒に受講してもらうことにより、Scratch など習得したプログラミングの手法が、実際の開発に用いられるプログラミング言語でも活用できることを学習してもらうことを目標とする。また、場合によっては、産学間連携推進室関係者が開発したオープンソースのゲームエンジン Altseed を活用し、簡易なゲームに拡張することも検討する。

● Raspberry Pi の GPIO ピンを活用し、各種のセンサなどの部品と連携したデバイスを制作する。Raspberry Pi の性質上、インデペンデンス・サーバー・デイで取り扱っているような単体でのサーバの構築が可能であるため、例えば「アクセスすると部屋の温度が実測値で表示されるサーバ」など、複数の要素を組み合わせた成果物を仕上げる事が可能である。ただし、内容の都合上 1-2 日での実施が難しいことが想定されるため、この内容での開催に当たっては開催日程やその対象についてよく検討する必要があるものと思われる。

5-3. 講座開催に伴う知見の共有について

「インデペンデンス・サーバー・デイ」の開催を通して得られた知見については、産学間連携推進室のイベント運営者有志により「プログラミング教育講座の舞台裏」という自費出版書籍に取りまとめを行い、定期的に日本最大の同人誌即売会「コミックマーケット」にて刊行している。その様子を図 9 に示す。



図 9:「プログラミング教育講座の舞台裏」の頒布の様子

「プログラミング教育講座の舞台裏」は、2019 年 8 月に創刊し、2024 年 5 月現在時点では Vol.05 まで、および「教育者のための情報技術」というスピンオフ書籍が出版されている。Vol.01 から Vol.05、およびスピンオフ書籍を含めた現在までの累計発行部数は、現時点で 700 部弱に上る。「コミックマーケット」会場ではコンピュータ技術者、教育者、また情報技術教育に携わる方々によって多く購入されているようである。

6. 学会発表等に関する情報

2018 年春時点までの「インデペンデンス・サーバー・デイ」に関する総括は、2018 年 3 月 13 日 - 2018 年 3 月 15 日に早稲田大学にて開催された 情報処理学会 第 80 回全国大会にて、学生セッション 6ZF-06「サーバ構築と運用の実習を通じた中学生対象の実践的な ICT 教育イベントの試み」として口頭発表を実施した。本イベントの性質上、学会のセッションにおいてはかなり異色の口頭発表となったものの、このようなイベントの開催を高く評価する声も多く、有意義な発表であったと考えられる。

2020 年春時点までの「インデペンデンス・サーバー・デイ」に関する総括は、2020 年 3 月 14 日にオンラインにて開催された 情報処理学会「コンピュータと教育研究会」第 154 回研究発表会にて、学生セッション[3]「サーバ構築・長期運用の実習を主題とする中学生向け ICT 教育イベントの継続的实践とその成果」として口頭発表を実施し、学生奨励賞を受賞した。情報技術教育に関する研究会において、本イベントの概況、およびその成果を発表できたことは、社会的にも大きな意義のあることであると思われる。

2020 年夏以降の「インデペンデンス・サーバー・デイ」に関しては、2020 年～2022 年までの新型コロナウイルス感染拡大の影響下におけるイベント開催形態の変遷、また 2023 年度以降における本イベントの発展・拡大等も踏まえた内容として、今後の学会発表に向けた準備を行っている。

7. 謝辞

「インデペンデンス・サーバー・デイ」の開催に当たっては、関係各所の多くの皆様のご協力・ご支援を頂きました。「インデペンデンス・サーバー・デイ」を継続して主催して頂いております つくば市総合教育研究所 の皆様、ご後援を頂いた 筑波大学情報科学類 関係教職員の皆様、ご協力を頂いた 筑波大学 OPEN プロジェクトのご担当者の皆様、ソフトイーサ株式会社の皆様、輝日株式会社の皆様に、心より御礼を申し上げます。

また、いずれの講座にも当日のスタッフとしてご協力いただいた、産学間連携推進室のメンバーの皆様、筑波大学大学院・筑波大学の有志の皆様に厚く感謝を申し上げます。

2023 年度 筑波大学 情報学群情報科学類 産学間連携推進室 活動成果報告書

発行日： 2024 年 5 月 26 日 初版発行
発行者： 金子 尚樹（情報科学類 産学間連携推進室 学生代表）
編集者： 金子 尚樹、根本 晃輔
制作者： 情報科学類 産学間連携推進室 所属学生一同