

# MCJ-CloudHub の動作検証及び一般公開

齊藤 智也<sup>1)</sup>, 大江 和一<sup>2)</sup>, 西井 淳<sup>1)</sup>, 岡田 耕一<sup>1)</sup>, 爲末 隆弘<sup>1)</sup>, 王 躍<sup>1)</sup>,  
矢吹 溪悟<sup>1)</sup>, 筒井 優子<sup>2)</sup>, 丹生 智也<sup>2),3)</sup>, 竹房 あつ子<sup>2),4)</sup>

1) 山口大学

2) 国立情報学研究所

3) 国立遺伝学研究所

4) 総合研究大学院大学

info-cc@ml.cc.yamaguchi-u.ac.jp

## Operation Verifications and Public Release of MCJ-CloudHub

Tomoya Saito<sup>1)</sup>, Kazuichi Oe<sup>2)</sup>, Jun Nishii<sup>1)</sup>, Koichi Okada<sup>1)</sup>, Takahiro Tamesue<sup>1)</sup>,  
Yue Wang<sup>1)</sup>, Keigo Yabuki<sup>1)</sup>, Yuko Tsutsui<sup>2)</sup>, Tomoya Tanjo<sup>2),3)</sup>, Atsuko Takefusa<sup>2),4)</sup>

1) Yamaguchi Univ.

2) National Institute of Informatics

3) National Institute of Genetics

4) The Graduate University for Advanced Studies, SOKENDAI

### 概要

本研究では、運用組織ごとの計算機環境に応じて構築・運用可能で、かつ単一のシステムにより複数科目が同時利用可能な Web 型プログラミング演習システムの構築・運用技術の確立を目的としている。これにより、情報システム部門が手薄な教育機関等においても、プログラミング演習システムの導入・運用が可能になる。我々は、山口大学のオンプレミス環境で運用中のプログラミング演習システム（JupyterHub 及び nbgrader を独自に改修したほか、Moodle と連携可能にしたシステム）を他の組織にも導入可能とするために汎用化し、国立情報学研究所が提供している学認クラウドオンデマンド構築サービス（OCS）からの構築・運用が可能なシステムの開発を進めている。このシステムを MCJ-CloudHub（Multiple Course Jupyter-based Cloud Hub）と呼ぶ。本稿では、MCJ-CloudHub の概要、これまでの動作検証及び一般公開の状況について報告する。

## 1 はじめに

近年、人工知能（AI）及びデータサイエンス（DS）分野においてプログラミング演習及びデータ処理演習を効率的・効果的に進めるため、Web 型プログラミング演習システムの需要が高まっている [1–3]。企業や研究機関では、オープンソースの Web 型プログラム開発・実行環境である JupyterHub [4] が広く普及している。そのため、国内・海外の教育機関では JupyterHub の教育利用に関する研究・開発が進められてきた [1–3,5]。国内では、Google 社が JupyterHub をベースとして実装した Google Colaboratory の教育利用 [6] のほか、Moodle [7] 等の学習管理システムと JupyterHub を連携させたプログラミング教育環境の構築も進められている [8–10]。

山口大学では、文系学部を含む全学部にわたるデー

タサイエンス教育科目の演習環境として、Moodle, JupyterHub, 及び nbgrader を組み合わせたプログラミング演習システム（YamaguchiHub と呼ぶ）を導入・運用している [11]。JupyterHub の機能拡張である nbgrader [12] は、Web ブラウザを介して課題の新規作成、課題ファイルの配布、提出物の回収・採点、並びにフィードバック（採点結果及びコメント）の作成・配布が可能であり、多人数の演習に有用である [13,14]。YamaguchiHub では、利用者が経由してきた Moodle コースからコース情報及び参加者情報を取得することにより、アカウント登録及びホームディレクトリの作成、科目用の共有ディレクトリ等の作成及びアクセス権限の設定が自動的に処理される。各科目や利用者の環境設定が自動的に作成・適用されるため、単一のシステムを複数科目で同時に利用可能であり、システム管理者の負担も軽減される。しかし、YamaguchiHub

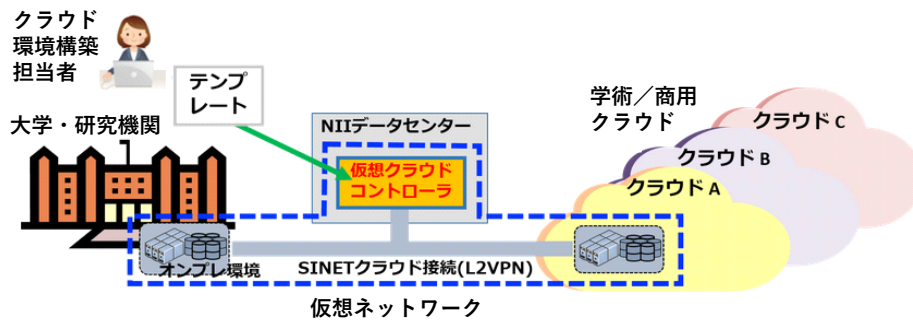


図 1: 学認クラウドオンデマンド構築サービス

は従来の JupyterHub に比べてシステムの構成要素が多く、構築手順も複雑である。加えて、山口大学の計算機環境に固有の設定値等がシステムのソースコード及び設定ファイルに直接に記述されているため、他の組織への導入が困難である。そこで我々は、YamaguchiHub の構築・運用手順を整理し、国立情報学研究所 (NII) が運営する学認クラウドオンデマンド構築サービス (OCS) [15] に対応したアプリケーション・テンプレートとして開発し、汎用化した。開発したシステムを MCJ-CloudHub (Multiple Course Jupyter-based Cloud Hub) と呼ぶ。OCS は、オンプレミス環境とクラウド環境にまたがるハイブリッド・クラウドにアプリケーションの配置・実行が可能なクラウド構築支援サービスである。OCS は複数のクラウド・プロバイダに対応しているほか、オンプレミス環境とクラウド環境の一方だけでも利用可能であるため、各組織の計算機環境及びネットワーク環境に応じた形でアプリケーションの構築・運用が可能である (図 1 参照)。2024 年 8 月より、我々は MCJ-CloudHub のアプリケーション・テンプレートを GitHub 上で一般公開している [16]。一般公開に加えて導入機関の利用者及びシステム管理者の間でユーザ・コミュニティを形成することにより、導入機関の間で環境構築、運用、授業実践に関するノウハウの共有が可能になる。

本稿では、MCJ-CloudHub の概要、動作検証の結果及び発見された問題点、並びにアプリケーション・テンプレートの一般公開について報告する。

## 2 MCJ-CloudHub

### 2.1 システム概要

図 2 に、MCJ-CloudHub の概要を示す。アプリケーション・テンプレートには JupyterHub の Manager ノード及び Worker ノードを生成する手続き、各サーバ上で稼働するアプリケーションの設定内容が含まれ

る。Manager ノードでは JupyterHub, Nginx (Web サーバ), MariaDB (DB サーバ), OpenLDAP (認証サーバ), NFS サーバ (ファイルサーバ) が稼働する。利用者の Jupyter 環境は、Docker コンテナにより実現される仮想環境として、Worker ノード上に展開される。1 台の Worker ノードには複数の Jupyter 環境が配置され得る。Manager ノードは NFS サーバを兼務し、外部ストレージを Worker ノードとの間で共有する。ストレージには科目ごとの共有ディレクトリや利用者ごとのホームディレクトリが配置される。また、図 2 では OCS のアプリケーション・テンプレートにより構築された Moodle を用いているが、運用組織の既存の Moodle とも連携可能である。

図 3 は、MCJ-CloudHub における複数科目への対応の様子を示している。MCJ-CloudHub を利用するとき、利用者 (教員・学生) は Moodle コースに設置された「外部ツール」のインスタンスを経由して JupyterHub にアクセスする。Moodle と JupyterHub の連携の仕組みとして LTI (Learning Tools Interoperability) [17, 18] を採用しているため、利用者の JupyterHub へのログインは LTI 認証 [19, 20] により自動的に完了する。ログインの完了後、利用者が経由してきた Moodle コース、及びコースにおける利用者のロール (役割) に応じて、自動的に環境設定等が行われる。まず、利用者が LDAP サーバに未登録の場合、アカウント登録及びホームディレクトリの作成が行われる。続いて、nbgrader を利用しない科目のための共有ディレクトリが作成される。共有ディレクトリのアクセス権限は、Moodle コースの「教師」は読み書き可能に、「学生」は読み込み限定に設定される。また、利用者のホームディレクトリの配下に、共有ディレクトリへのシンボリックリンクが作成される。シンボリックリンクの名称は科目によらず単一であるが、リンク先となる共有ディレクトリは、利用者が経由してきた

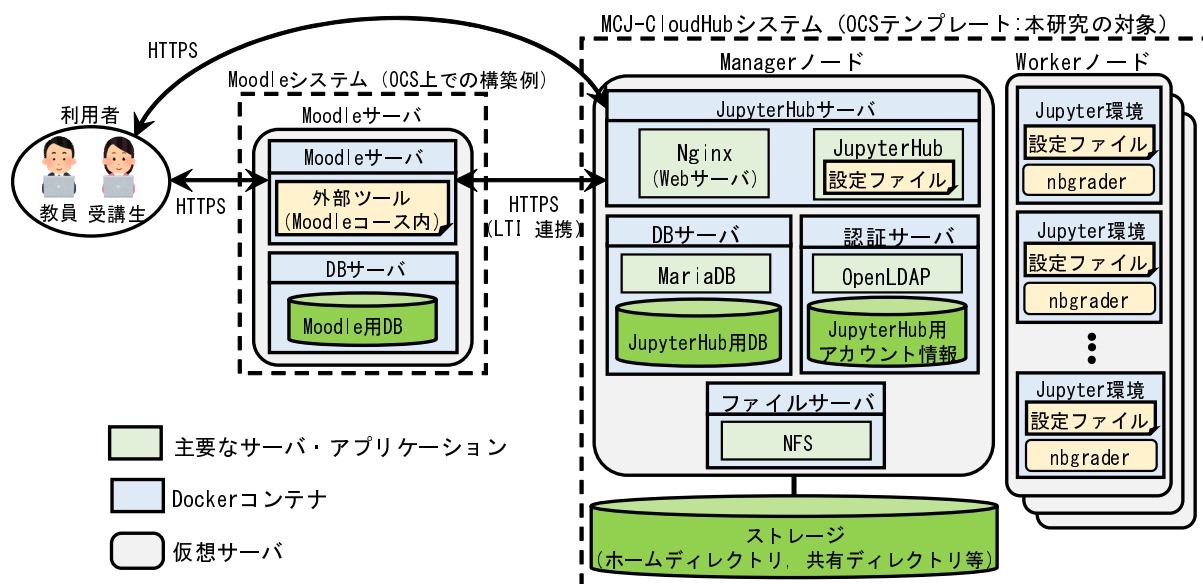


図 2: MCJ-CloudHub の概要

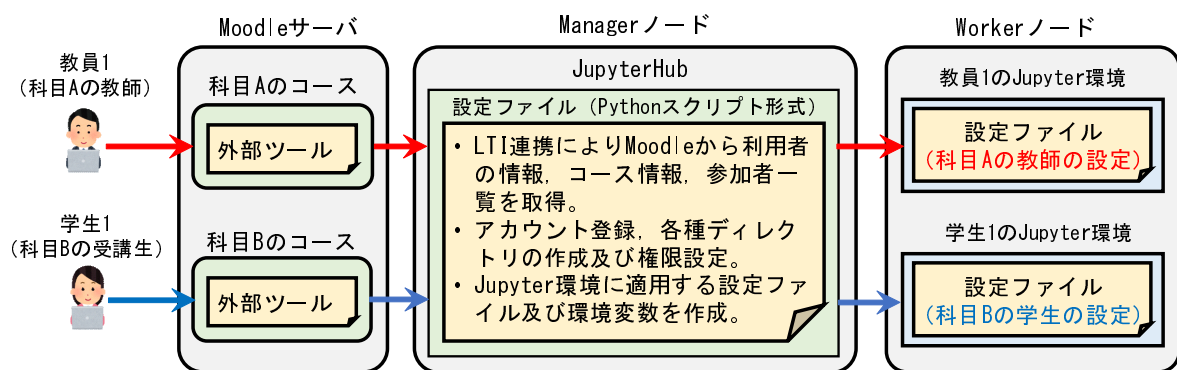


図 3: MCJ-CloudHub における複数科目への対応

Moodle コースに応じて選択される。受講生は必要なファイル等を共有ディレクトリから各自のホームディレクトリにコピーした後、講義資料の閲覧やプログラムの作成が可能である。この仕組みにより、nbgrader を利用しない場合でも課題ファイルの配布が可能である。また、利用者が経由してきた Moodle コースの情報に基づき、その Moodle コース（科目に相当）に関連する共有ディレクトリのみが表示されるため、システムの利用科目数が増加した場合も、利用者による共有ディレクトリの選択間違いが生じない。

次に、nbgrader で利用されるディレクトリ等が作成・設定される。利用者が Moodle コースの「教師」の場合は該当科目のファイル交換用ディレクトリが作成され、アクセス権限が設定されるほか、ホームディレクトリの配下にある nbgrader 用の設定ファイルに、参加者一覧が記述される。LTI 連携及び LTI 規格の NRPS (Names and Role Provisioning Services) [21] により取得した Moodle コースの名称、利用者の Moodle

コースにおける役割（教師／学生）、利用者の属性情報（ユーザ名、氏名）に加え、nbgrader で用いるファイル交換用ディレクトリのパス等の情報が Manager ノードから引き渡され、利用者の Jupyter 環境の設定ファイルに記述される。また、MCJ-CloudHub では Jupyter 環境の起動時に、利用者が経由してきた Moodle コースに対応した nbgrader コースが自動的に作成・選択済みとなる。そのため、利用者による nbgrader コースの選択間違いが生じない。

従来の JupyterHub では、科目（JupyterHub ではグループ及び nbgrader コース）に固有の環境設定を適用する場合、JupyterHub の設定ファイルに設定内容を記述しなければならない。このとき、同時に複数の科目の環境設定を有効にすることはできない。JupyterHub の設定ファイルの内容は利用者ごとの Jupyter 環境に影響を及ぼすため、システム全体が特定の科目に専用のもになってしまう。JupyterHub がある科目の環境設定により稼働している間、別の科

目の環境は利用できない。そのため、科目に固有の環境設定を施したシステムを運用する場合、複数の科目で同時利用ができないほか、時間外学習に利用できない問題が生じる。また、従来の nbgrader では、利用者はシステムに登録済みの nbgrader コースの一覧から所定のコースを選択した後に各種操作を行う。このとき、受講していない科目の nbgrader コースも表示され、選択可能である。nbgrader の利用科目数が増加した場合、利用者がコースの選択を間違える恐れがある。一方、MCJ-CloudHub では利用者が経由してきた Moodle コースに応じて、該当科目の環境設定が施された Jupyter 環境を提供する。利用者の Jupyter 環境は科目に固有のものとなるが、JupyterHub は特定の科目に依存しない状態を維持するため、複数科目による同時利用、及び授業時間外の利用が可能である。

MCJ-CloudHub では、この他にもいくつかの改善が施されている [22]。まず、JupyterHub のデータベース管理システムを SQLite から MariaDB (MySQL の互換ソフトウェア) に変更し、nbgrader のソースコードを改善した。これにより、システムの構成要素が別々の物理サーバ上に配置される分散サーバ環境においても nbgrader の機能を利用可能としている。加えて、nbgrader のファイル交換用ディレクトリのアクセス権限を改善し、他者の提出物や採点結果の閲覧・改変・削除を禁止している。

### 3 動作検証

我々はこれまでに、学術研究用のパブリック・クラウドである mdx [23]、及び AWS (Amazon Web Services) を用いて、MCJ-CloudHub の動作検証を進めた [22]。これまでの動作検証を通じて、MCJ-CloudHub の各機能は mdx 及び AWS 上で動作することが確認されたが、いずれの検証も参加者は 10~20 名であった。そこで 2024 年 7 月に、山口大学工学部知能情報工学科の演習科目の受講生を対象に、多人数による動作検証を実施した。今回の検証では mdx 上にアプリケーション・テンプレートを用いて MCJ-CloudHub を構築し、動作確認を進めた。Manager ノードの仮想コア数は 3 コア、仮想メモリ容量は 4GB とした。利用者の Jupyter 環境を展開するための Worker ノードは 3 台とし、Worker ノードごとの仮想 CPU コア数は 41 コア、仮想メモリ容量は 62GB とした。利用者の Jupyter 環境ごとの資源割り当ての最大値は、CPU コア数は 1 コア、メモリ容量は 512MB とした。教員 1 名及び受講生 89 名の計 90

名が参加し、nbgrader の各種機能について動作を確認した。Worker ノードは 3 台であるため、ノードごとに 30 名の参加者を収容している。

動作検証は以下の手順に沿って進めた。

- (1) 参加者は、配布資料に基づいて mdx 上に準備された Moodle サイトにログインし、Moodle コースに配置された「外部ツール」をクリックして JupyterHub にアクセスする。利用者の JupyterHub へのログイン及び Jupyter 環境の起動は自動的に実行される。
- (2) 受講生は課題ファイルを取得し、解答を記入し、プログラムを実行した後にファイルを提出する。課題は 2 問あり、それぞれに実行時間が数秒程度の簡単な Python スクリプトの作成を含む。
- (3) 受講生全員が課題を提出した後、教員は課題の回収及び自動採点を実行し、生成されたフィードバックの公開の操作を行う。
- (4) 受講生はフィードバックを取得・閲覧する。
- (5) フィードバックの取得・閲覧を終えた受講生は、JupyterHub 及び Moodle からログアウトする。

動作検証により、Moodle と JupyterHub の連携、アカウント登録、各種ディレクトリの作成及び権限設定、利用者及びコースの情報を反映させた Jupyter 環境の起動が正常に動作することが確認された。

nbgrader による課題の作成・配布、レポートの提出、及び提出物の回収・採点の機能のうち、動作検証では提出物の一括採点において不具合が生じた。教員が受講生からの提出物を一括採点する際、提出物のファイルを読み取りできないエラーや、nbgrader 用のデータベース (教員のホームディレクトリの配下に配置された SQLite 形式のファイル) にアクセスできないエラーにより、採点処理が途中で異常終了してしまった。これは、NFS クライアント (ここでは Worker ノード) が NFS を介してストレージを共同利用する際、ファイル・システムのデッドロックが頻繁に発生するためである。本システムでは、Manager ノードと各 Worker ノードの間のファイル共有の方式として NFSv4 を採用している。NFSv4 では、NFS サーバはクライアント ID の値を用いて NFS クライアントを識別する [24]。NFS クライアントは、クライアント用の設定ファイル内に「nfs4\_unique.id」の値が定義されている場合にはこれをクライアント ID として使用し、定義されていない場合にはホスト名を使用する。ただし、デフォルトでは「nfs4\_unique.id」の値は定義され

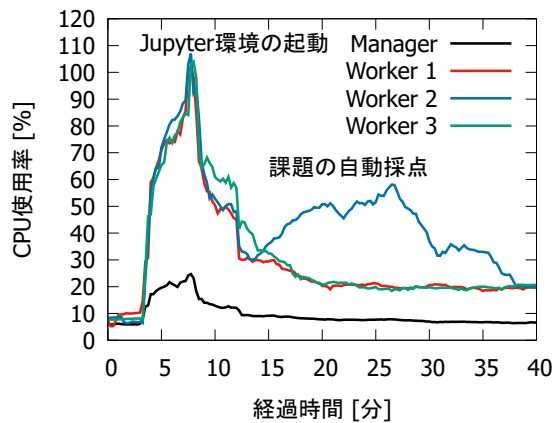


図 4: 経過時間に対する CPU 使用率の変動

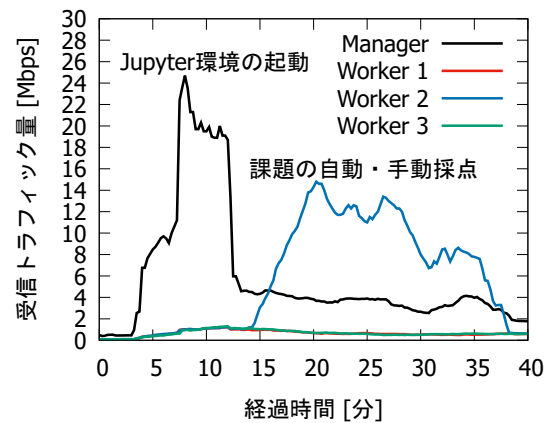


図 6: 経過時間に対する受信トラフィック量の変動

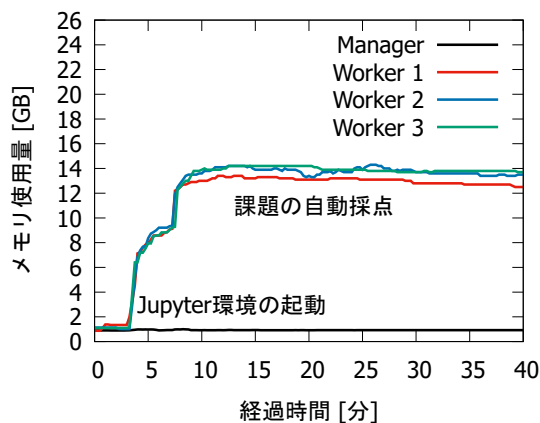


図 5: 経過時間に対するメモリ使用量の変動

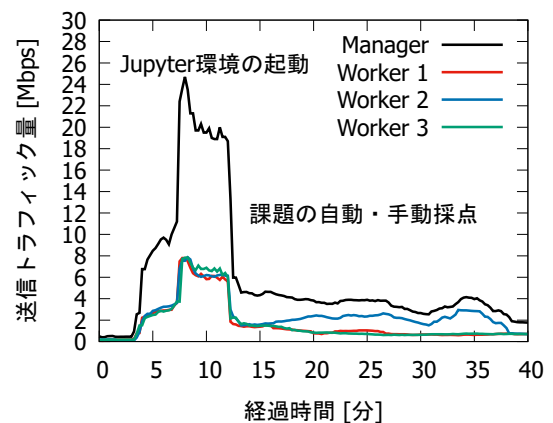


図 7: 経過時間に対する送信トラフィック量の変動

ていない。また、mdx の API を用いて仮想サーバを動的に生成する場合、仮想サーバのオペレーティング・システムに設定されるホスト名はすべて同一のものになるため、NFS サーバは NFS クライアントを適切に識別できない。これにより、ファイルのロックと解放に異常が生じ、NFS の共有ディレクトリがアクセス可能な状態とアクセス不可能な状態を短時間で繰り返していた。受講生による課題ファイルの取得やレポートの提出の処理においてはエラーは発生しなかったが、一括採点の処理の過程で提出物やデータベースにアクセスができなかった場合には異常終了が発生した。

図 4 及び 5 はそれぞれ、動作検証の開始時点からの経過時間に対する各仮想サーバ（Manager ノード及び Worker ノード）の CPU 利用率及びメモリ使用量の変化を示している。動作検証の開始から 3 分程度経過した頃より、受講生の Jupyter 環境の起動に伴い、Worker ノードの CPU 使用率及びメモリ使用量が増加している。今回の動作検証では、受講生が JupyterHub にアクセスするタイミングを合わせていないため、Jupyter 環境が起動される時間帯に多少の

ばらつきが生じている。これに伴い、CPU 使用率及びメモリ使用量は数分間にわたって段々と増加している。なお、今回の動作検証に使用した Jupyter 環境の Docker イメージのサイズは 2GB 程度で、利用者ごとの Jupyter 環境の起動時間は 5~10 秒程度であった。

受講生は Jupyter 環境の起動後すぐに課題ファイルを取得し、簡単な課題についてプログラムを作成・実行し、すぐに課題ファイルを提出している。そのため、各 Worker ノードの CPU 使用率は動作検証開始から 8 分程度経過した頃にピークに達した後、15 分経過後まで緩やかに減少している。Worker ノードのオペレーティング・システムには Ubuntu を採用している。Linux 系 OS では、OS がいったん確保したメモリは、割り当て先のプロセスが終了しても OS が確保し続け、他のプロセスに利用する。そのため、利用者の Jupyter 環境の起動が終了しても、ノードごとのメモリ使用量は減少していない。動作検証の開始から 14 分程度経過した頃より、教員による課題の採点のため、Worker ノード 2 の CPU 使用率が増加している。教員は最初、すべての提出物の一括採点を実行したが、

前述の不具合により異常終了する事態が続いたため、その後は個々の提出物に対して手動で採点処理を実行した。この操作の過程でも何度か異常終了が発生し、すべての提出物の採点には 25 分を超える時間を要した。図 4 及び 5 では、Worker ノード 2 の CPU 使用率及びメモリ使用量は採点処理に伴って緩やかに上昇している。不具合が発生せず、すべての提出物の採点が正常に終了した場合には、Worker ノード 2 の CPU 使用率及びメモリ使用量はより急激に増加し、短時間で減少傾向に転じるものと考えられる。

図 6 及び 7 はそれぞれ、動作検証の開始時点からの経過時間に対する各仮想サーバの受信トラフィック量及び送信トラフィック量の変化を示している。動作検証の開始から 3 分程度経過した頃より、受講生の Jupyter 環境の起動に伴い、Manager ノードの受信トラフィック量及び送信トラフィック量が増加している。Manager ノードの受信トラフィックと送信トラフィックが似た傾向を示し、各 Worker ノードの送信トラフィック量の総和とほぼ同量のトラフィック量であるため、Manager ノードは Worker ノードから受信したデータを利用者に中継していることが分かる。今回の動作検証で受講生が取り組んだ課題は、実行結果として少量のテキスト及び画像が利用者の画面に表示されるものである。大きなデータのストレージへの書き込みを伴わない課題であったため、Worker ノードから Manager ノードに送信されたデータのほぼすべてが、利用者の Web ブラウザに転送されている。動作検証の開始から 14 分程度経過した頃より、教員による提出物の採点処理に伴い、Worker ノード 2 の受信トラフィック量が増加している。しかしながら、Manager ノードの送信トラフィック量は増加していない。提出物の採点処理では、ストレージに保存されている課題ファイルの内容の読み込みや、Manager ノードで稼働するデータベースからのデータの読み込みが頻繁に発生する。これらのデータは Manager ノードから教員の Jupyter 環境に送信されるため、Manager ノードの送信トラフィック量と Worker ノード 2 の受信トラフィック量は同様に増加しなければならない。トラフィック量の相違が生じた原因が前述した NFS 関連の不具合であるか否かは、現時点では不明である。

## 4 一般公開

我々は、2024 年 8 月下旬に GitHub 上で MCJ-CloudHub の初期バージョンの一般公開を開始した [16]。初期バージョンでは、3 節で示した NFS 関連

の不具合については解消されていない。本システムを mdx 上に展開して利用する場合、システム管理者は Worker ノードの仮想サーバを生成した後、仮想サーバのオペレーティング・システム上のホスト名を適宜設定しなければならない。今後公開するバージョンでは、アプリケーション・テンプレートをを用いてシステムを構築する際に、Worker ノードごとに別々のホスト名を設定可能にする予定である。

一般公開の開始後、2024 年 8 月下旬に MCJ-CloudHub の利用に関するセミナーを実施した [25, 26]。セミナーの参加者は MCJ-CloudHub の概要について説明を受けた後、nbgrader の機能を利用した課題の作成・配布、提出物の回収・採点に関する演習に取り組んだ。参加者は以下の手順に基づき、教員と学生の双方の立場の操作を体験した。

- (1) 教員用のアカウントで Moodle 及び JupyterHub にログインし、課題の作成及び公開開始の操作を行う。
- (2) 学生用のアカウントで Moodle 及び JupyterHub にログインし、課題の取得、解答の作成、課題の提出を行う。
- (3) 教員側の画面において課題を回収し、一括採点及び手動採点を行う。
- (4) 学生側の画面においてフィードバック（採点結果及びコメント）を取得し、閲覧する。

nbgrader の機能を用いた演習の流れを体験した後、講師及び参加者の間で MCJ-CloudHub の導入・運用、実際の授業で利用するプログラミング言語の処理系やパッケージ等を含む Jupyter 環境の準備、及び多人数に対するプログラミング演習の進め方についての議論が交わされた。MCJ-CloudHub に限らず、JupyterHub やそれをベースにする演習システムを多数の科目に利用する場合、利用科目数の増加に伴い、すべての科目の要件を満たす Jupyter 環境の Docker イメージの作成が困難になる。また、多人数による演習科目の運営に際し、受講生のモチベーションの維持、理解度の確認、受講生ごとに個別の課題を配布する方法について議論が交わされた。プログラミング教育に代表される関連分野の研究・開発の促進に加え、システムの構築・運用、教育実践に関するノウハウの情報共有を通じてシステム管理者及び授業担当教員の負担を軽減するためにも、MCJ-CloudHub の導入機関のシステム管理者及び授業担当教員によるユーザ・コミュニティの形成及び活発化が必要である。



## 5 まとめ

本稿では、運用組織ごとの計算機環境に応じて構築・運用可能で、かつ単一のシステムにより複数科目が同時利用可能な Web 型プログラミング演習システムの構築・運用技術の確立について述べた。山口大学で運用中のプログラミング演習システム (YamaguchiHub) では、大学の計算機環境、及びネットワーク環境に依存する箇所が多く、他の組織への導入が困難である。そこで我々は、YamaguchiHub を OCS に対応したアプリケーション・テンプレートとして開発し、汎用化した。開発したシステムを MCJ-CloudHub と呼ぶ。ハイブリッド・クラウドへの対応により、組織ごとの計算機環境に応じた実装・運用が容易になる。また、アプリケーション・テンプレートの一般公開により、情報システム部門が手薄な教育機関においても、希望する計算機環境の上にプログラミング演習システムの構築・運用が可能となる。MCJ-CloudHub の導入機関の利用者、及びシステム管理者によるユーザ・コミュニティを形成し、導入機関の間での環境構築・運用、授業実践に関するノウハウの共有、関連分野における研究・開発の活性化を促すことも目指している。

我々はこれまでに、MCJ-CloudHub を OCS に対応したアプリケーション・テンプレートとして実装し、mdx 及び AWS を用いた動作検証を行った。いずれのクラウド環境においても、MCJ-CloudHub は科目に応じた利用環境を自動的に生成・適用し、利用者は nbgrader の機能を用いた課題の作成・配布・提出・採点等の各種操作が可能であることが示された。しかし、mdx を用いた多人数による演習においては、NFS によるファイル共有に関する不具合により、教員による課題の採点処理が異常終了する事態が生じた。初期バージョンのアプリケーション・テンプレートは現在、GitHub 上で公開中である [16]。現在公開中のテンプレートでは NFS 関連の不具合に未対応だが、今後、対応済みのアプリケーション・テンプレートを公開予定である。一般公開の開始後、MCJ-CloudHub の利用に関するハンズオン・セミナーを実施した。

今後の課題として、GakuNin [27] に準拠した Shibboleth サーバを利用者の属性情報の供給元として選択可能にすることが挙げられる。現在、利用者の属性情報の供給元は Moodle コースのみであるが、教育機関の方針や都合により、MCJ-CloudHub の運用に必要な属性情報が Moodle のユーザ・アカウントに含まれていないケースも考えられる。LTI 規格の NRPS と

同様、GakuNin 準拠の Shibboleth サーバには特定の運用組織に依存しない共通の属性項目が定義されているため、複数の教育機関に MCJ-CloudHub を導入するために有用である。また、アプリケーション・テンプレートの改善、受講生の課題進捗状況の収集・可視化機能等の支援機能の研究・開発、並びにセミナー等を通じた普及活動に取り組む予定である。

## 謝辞

この研究は 2024 年度国立情報学研究所公募型共同研究 (24S0802) 及び情報・システム研究機構「戦略的研究プロジェクト」(2024-SRP-06) の助成を受けています。

本研究にご協力いただいた株式会社アスケイドの那須野淳様、羽鳥文子様、増山隆様、神田千暁様、鳥井貴史様、寺山流晟様に深く感謝いたします。

MCJ-CloudHub の動作検証にご協力いただいた山口大学大学院創成科学研究科の佐村俊和 准教授、藤田悠介 准教授、楊鯤昊 講師、小林泰良 助教に深く感謝いたします。

## 参考文献

- [1] J.W. Johnson and K.H. Jin, "Jupyter notebooks in education, Journal of Computing Sciences in Colleges", Vol.35, No.8, pp.268-269, 2020.
- [2] A. Al-Gahmi, Y. Zhang and H. Valle, "Jupyter in the Classroom: An Experience Report", Proceedings of the 53rd ACM Technical Symposium on Computer Science Education, Vol.1, No.1, pp.425-431, 2022.
- [3] H. Go, S.-W. Kim and Y. Lee, "Design and Implementation of a Programming Automatic Assessment System in Jupyter Notebook", International Journal on Advanced Science, Engineering & Information Technology, Vol.13, No.3, pp.1080-1086, 2023.
- [4] Project Jupyter, "JupyterHub", <https://jupyter.org/hub>, 2024 年 10 月 15 日確認.
- [5] 田浦健次郎, 「Jupyterhub と nbgrader で授業・試験・レポートを DX してみた件」, 第 39 回大学等におけるオンライン教育とデジタル変革に関するサイバーシンポジウム, 2021.
- [6] 山口健二, 「Google Colaboratory による自宅学習教材の作成と遠隔授業方法の検討」, 高等教

- 育と学生支援：お茶の水女子大学紀要，Vol.10，pp.59-63，2019.
- [7] Moodle HQ，"Moodle: Open-source learning platform"，<https://moodle.org/>，2024 年 10 月 15 日確認.
- [8] 石坂徹，桑田喜隆，合田憲人，政谷好伸，横重俊，浜元信州，「Moodle と Jupyter Notebook の連携によるプログラミング教育環境の構築」，日本モデル協会全国大会（2019）発表論文集，pp.32-37，2019.
- [9] 井関文一，浦野真典，「LTI カスタムパラメータによる Moodle - JupyterHub 連携に関する研究」，日本モデル協会全国大会（2022）発表論文集，pp.20-25，2022.
- [10] 池田裕希，長尾和彦，田房友典，Moodle と JupyterHub を用いたプログラミング環境の構築，情報処理学会第 84 回全国大会講演論文集，Vol.2022，No.1，pp.205-206，2022.
- [11] 齊藤智也，王躍，西井淳，河野綾華，ルールマルク，「Moodle と JupyterHub を用いた Web 型プログラミング教育環境の改善および運用」，情報処理学会研究報告（CLE），Vol.2023-CLE-39，No.8，pp.1-8，2023.
- [12] Jupyter Development Team，"nbgrader"，<https://nbgrader.readthedocs.io/en/stable/>，2024 年 10 月 15 日確認.
- [13] N. Klever，Jupyter Notebook，"JupyterHub and Nbgrader，University of Applied Sciences Augsburg"，[https://klever.hs-augsburg.de/nb/OWL/LAB\\\_2020\\\_02\\\_Klever\\\_Jupyter\\\_Notebook\\\_JupyterHub\\\_Nbgrader.pdf](https://klever.hs-augsburg.de/nb/OWL/LAB\_2020\_02\_Klever\_Jupyter\_Notebook\_JupyterHub\_Nbgrader.pdf)，2024 年 10 月 15 日確認.
- [14] R.D. Smet，S. Thielemans，J. Lemeire，A. Braeken and K. Steenhaut，"Educational software-as-a-service based on JupyterHub and nbgrader running on Kubernetes"，2022 IEEE 9th International Conference on e-Learning in Industrial Electronics (ICELIE)，2022.
- [15] 国立情報学研究所，「学認クラウドオンデマンド構築サービス」，<https://cloud.gakunin.jp/ocs/>，2024 年 10 月 15 日確認.
- [16] 山口大学，国立情報学研究所，「MCJ-CloudHub」，<https://github.com/nii-gakunin-cloud/mcj-cloudhub>，2024 年 10 月 15 日確認.
- [17] 1EdTech Consortium Inc.，"Learning Tool Interoperability Core Specification 1.3"，<https://www.imsglobal.org/spec/lti/v1p3>，2024 年 10 月 15 日確認.
- [18] 村上幸生，喜多敏博，江川良裕，中野裕司，「LMS との連携を実現するための LTI 準拠学習支援ツールの実装例と実装方法」，情報処理学会研究報告（CLE），Vol.2012-CLE-8，No.5，pp.1-6，2012.
- [19] 1EdTech Consortium Inc.，"Learning Tools Interoperability as a SSO Mechanism"，<https://www.imsglobal.org/learning-tools-interoperability-\sso-mechanism>，2024 年 10 月 15 日確認.
- [20] Project Jupyter Contributors，"LTI Authenticator for JupyterHub"，<https://ltiauthenticator.readthedocs.io/>，2024 年 10 月 15 日確認.
- [21] 1EdTech Consortium Inc.，"Learning Tools Interoperability Name and Role Provisioning Services Version 2.0"，<https://www.imsglobal.org/spec/lti-nrps/v2p0>，2024 年 10 月 15 日確認.
- [22] 齊藤智也，大江和一，西井淳，岡田耕一，爲末隆弘，王躍，筒井優子，丹生智也，竹房あつ子，「複数科目で同時利用可能な Web 型プログラミング演習システムの構築・運用方法の汎用化」，学術情報処理研究，Vol.28，No.1，2024 年.
- [23] 東京大学情報基盤センター，「mdx - データ活用社会創成プラットフォーム協働事業体」，<https://mdx.jp/>，2024 年 10 月 15 日確認.
- [24] The kernel development community，"NFS Client - The Linux Kernel documentation"，<https://www.kernel.org/doc/html/v5.8/admin-guide/nfs/nfs-client.html>，2024 年 10 月 15 日確認.
- [25] 大江和一，「オンデマンド構築サービスを使った講義・演習環境アプリケーション概要」，学認クラウドオンデマンド構築サービスを利用した講義・演習環境体験セミナー，2024.
- [26] 齊藤智也，「講義・演習環境アプリケーション ハンズオン: nbgrader を用いた演習課題の配布・回収・採点」，学認クラウドオンデマンド構築サービスを利用した講義・演習環境体験セミナー，2024.
- [27] 国立情報学研究所，「学術認証フェデレーション」，<https://www.gakunin.jp/>，2024 年 10 月 15 日確認.