

データ分析プロセス共有による研究再現例

横山 重俊^{1,3,a)} 浜元 信州¹ 長久 勝² 藤原一毅³ 政谷 好伸³ 竹房 あつ子³ 合田 憲人³

概要: データ駆動型科学研究分野で研究データの再利用を進めるためには、研究データの公開だけでなく、論文成果につながる実験結果にたどり着くまでの実験プロセス（実験手順や実験環境構築手順）を公開し共有する必要がある。これらが揃うことで、第三者である別の研究者がいつでも研究データを再利用し、研究の再現検証が可能となる。さらに、そのオリジナル研究に加えて自らの研究を派生させることがスムーズにできるようになる。本稿では、その実験プロセスをコミュニティで共有する方式に関する検討内容と今回提案する方式について述べ、その実装例および研究再現例についても報告する。

キーワード: オープンサイエンス, 研究環境再構成, DOI, Jupyter Notebook, クラウドコンピューティング

Example of Research Reproduction by Sharing Data Analysis Process

SHIGETOSHI YOKOYAMA^{1,3,a)} NOBUKUNI HAMAMOTO¹ MASARU NAGAKU² IKKI FUJIWARA³
YOSHINOBU MASATANI³ ATSUKO TAKEFUSA³ KENTO AIDA³

Abstract: In order to promote the reuse of research data in the field of data-driven scientific research, not only the publication of research data but also the experimental process (experimental procedure and experimental environment construction procedure) until reaching the experimental results leading to the paper results are disclosed. By having these, other researchers can reuse the research data at any time, and it is possible to reproduce and verify the research, and it is possible to smoothly derive his / her own research in addition to the original research. In this paper, we report the results of the examination about the method of sharing the experimental process in the community, and the implementation example and the research reproduction example of the adopted method.

Keywords: Open Science, Environment Reconstruction, DOI, Jupyter Notebook, Cloud Computing

1. はじめに

日本学術会議は今春、提言「オープンサイエンスの深化と推進に向けて」[1]として、データ駆動科学を推進し、科学を変容するデータ共有の精神が進展を目的に、データが中心的役割を果たす時代のルール作りの必要性、データプラットフォームの構築・普及の必要性、第1次試料・資料の永久保存の必要性、の三点に関する提言をまとめている。

その「データプラットフォームの構築・普及の必要性」提言の中で、「データ基盤とガバナンス」としてNII RDC(Research Data Cloud) [2]を引用し、以下のように述べている。

“NII RDCは、現在大学や研究機関が組織として導入するための試行実験が進められており、2020年度中から開始される本格運用の中で、基礎的なデータ基盤整備が具体的に実践されていくことになる。先進的な研究データ管理を実現するために基盤が実現すべき機能や役割については、欧米の先行機関の取り組みに見られるように発展途上にある。その中でも、データ解析システムや電子ラボノートといった周辺サービスとの連携機能や、研究プロジェクトにおける研究データのやりとりがさらに円滑になるツ-

¹ 群馬大学

Gunma University

² ライフマティックス株式会社

Lifematics

³ 国立情報学研究所

National Institute of Informatics

a) yoko@gunma-u.ac.jp

ルの導入など、様々な模索が続けられている。こうした研究開発による機能の高度化は NII RDC にも不可欠であり、研究機関や研究分野との密な連携を介して、ニーズに基づいた方向性を定めていく必要がある。”

本研究は、ここで述べられている「研究データのやりとりがさらに円滑になるツールの導入」の試行として、データ分析プロセス共有、すなわち実験手順や実験環境構築手順を公開し共有する取り組みの一例である。

2. 背景

オープンサイエンス推進のために、論文で使われた研究データの共有が進んでいる [3]。この際、研究データを論文と同様に長期保存するための方式と、そのための基盤の開発が進められている [4]。具体的には、図 1 の右に示すように、論文と同様に各研究データに DOI(Digital Object Identifier) を付与し、DOI による検索、長期保存の保証などを実現している [5,6]。さらには、論文との関係を DOI で表現することや論文の citation に相当する研究データ間の citation についても管理することを目指している [7,8]。

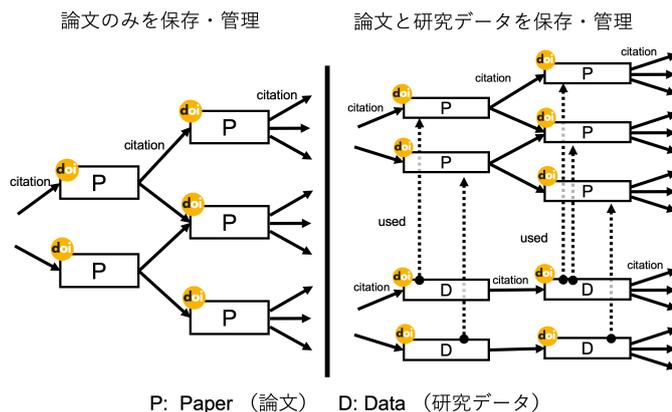


図 1 論文と研究データの長期保存・管理のために

研究データの再利用を進めるためには、研究データの公開だけでなく、研究データから論文に公表したデータ分析結果にたどり着くまでの、データ分析プロセスを公開し共有する必要がある。そのデータ分析プロセスの中には、実験手順だけでなく、その実験手順に従ってデータ分析を行える実験環境をどう構築するかについて記述した手順も必要である。

これらが揃うことで、第三者である別の研究者がいつでも研究データを再利用し、研究の再現検証が可能となり、さらにそのオリジナル研究に加えて、自らの研究を派生させることがスムーズにできるようになる。

これらの手順を保存・管理・提供する再構成基盤を新たに組み入れることで再構成に必要であった手間を大幅に削減できる可能性がある。すなわち、図 2 の右に示すように実験手順と実験環境を共有することで、派生研究者

(Researcher-D) などが論文に書かれている情報からオリジナル研究者 (Researcher-O) の用いた実験手順と実験環境を手動で再構成するという、手間のかかる作業から解放されるのである。

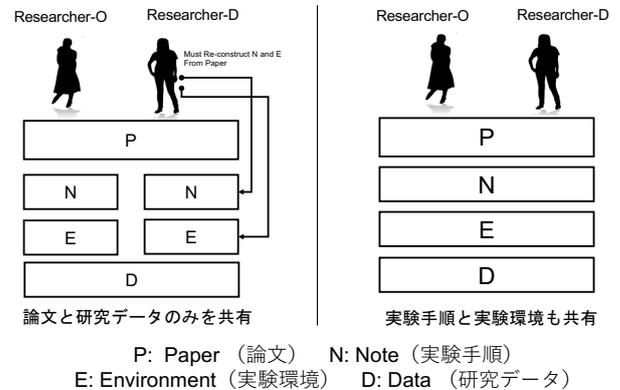


図 2 研究スタイルの比較

この際、論文 (P)、研究データ (D)、実験手順 (N) は何らかのストレージサービスに長期保存し、さらにそこに恒久的な識別子として DOI などを付与することが比較的容易である。しかしながら、実験環境はハードウェアとソフトウェアの総体であり、その上で実験が実験手順に従って研究データにアクセスしながら実行できる必要がある。

このため、実験環境の長期保存と識別子の付与体型との整合性確保を与えるには、さらに一段の工夫が必要である。そのような実験環境自体を共有する方法として大きく分けて図 3 に示す二つのアプローチがある。

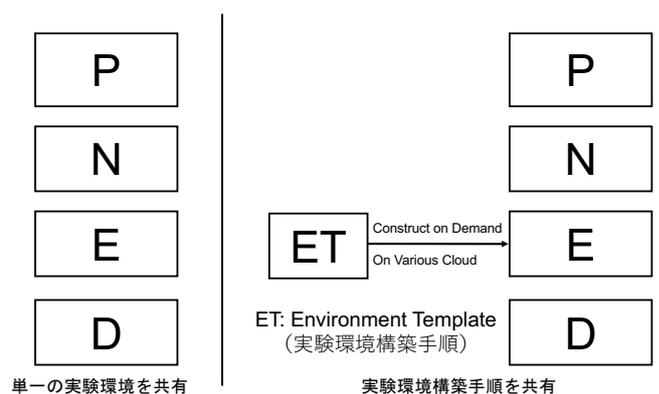


図 3 実験環境共有方法の比較

一つは図 3 の左側にある単一の実験環境を研究者間で共有し、それぞれの研究実施された実験手順がこの環境の上で常に実行可能な状況にしておくものである。このアプローチをとっているものの例が、次章の関連研究で紹介し

ている AiiDA [9] および Materials Cloud [10] である。

もう一つは図 3 の右側に示すように、実験環境をオンデマンドで構築手順書 (ET: Environment Template) に従ってクラウド基盤上に再構成するアプローチである。このアプローチをとる例は、関連研究で紹介している BinderHub [11, 12]+Zenodo [13], Chameleon Cloud データ分析環境再現サービスである [14]。

3. 関連研究

3.1 AiiDA と Materials Cloud

AiiDA は計算科学のためのフレームワークで、シミュレーションに必要なデータを保存でき、シミュレーションの自動化のためワークフローエンジンと共にワークフロー保存機能も持つ。

Materials Cloud はその中で、研究論文に関連するデータとデータ分析に関する情報をアーカイブするツールを提供するオープンサイエンスプラットフォームである。アーカイブに蓄積されている各研究論文に関連する情報に付与したデータ記述子の公開により、データの作成者とキュレーターは自分の作業に適切なクレジットを獲得できるなど、再現可能な研究を促進できる仕組みを持っている。

3.2 BinderHub+Zenodo

BinderHub を使用すると、Git リポジトリに Docker イメージを構築するために必要な情報を登録し、JupyterHub と接続して、研究者が実験手順を実行できる Jupyter Notebook 環境を Docker コンテナとして提供することができる [15]。この際 repo2docker [16, 17] と呼ばれるツールを利用して Git リポジトリにある情報を利用して Docker イメージを動的に構築する仕組みが利用されている。

Zenodo は永続的ストレージサービスであり、これを利用して Git リポジトリ内にある実験手順を記述した Jupyter Notebook とその Notebook が実行できる Docker イメージを構成するための構成情報を長期保存する。BinderHub と Zenodo の連携アーキテクチャを図 4 に示す。

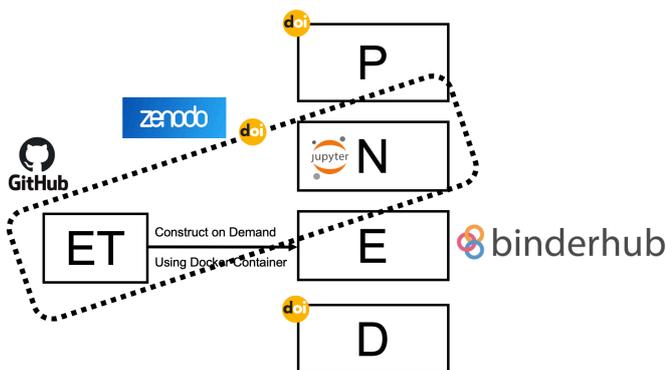


図 4 BinderHub+Zenodo のアーキテクチャ

Github などの Git リポジトリに配置した実験手順 (N) とそれを実行するために必要な環境情報を記述した情報 (ET) をそのリポジトリのブランチとして Zenodo に保存すると同時に DOI を付与する。論文 (P) からこの DOI を参照することで BinderHub によって実験環境を一つの pod (Docker コンテナ) として再現し、その中で実験を再現する方式である。

3.3 Chameleon Cloud データ分析環境再現サービス

NSF Cloud の一つである Chameleon Cloud 上の一つのサービスとして、Jupyter Notebook 環境を活用してこのクラウド上にデータ分析環境再現サービスを提供する研究がなされている。

AiiDA は All in One アプローチであり、各研究者がこの環境を使うことで、お互いの研究論文の再現性を達成できることを目指している。分野毎の要求や時間を経過することによるインタフェース仕様変更なども全て継続的にプラグイン機構を駆使することで吸収していく姿勢で運用を続けている。

なお、Materials Cloud の場合、再構築手順は研究データとともにストレージサービスである Zenodo に格納されている。しかし、Materials Cloud は再現環境自身はそれらの情報を元に再現者が自ら構築する必要があり、この再構築手順は標準化されていないため、論文毎に様々で再現者の研究再現の障害となる可能性がある。

BinderHub+Zenodo の場合、再構築手順は Github などの Git リポジトリに格納されており研究データとは別に管理され、それらの間はリンク情報で連携されている。ただ、BinderHub+Zenodo は、一つのコンテナイメージでデータ分析が実行される場合には有効であるけれど、マルチノードからなる計算機クラスタのような実行環境が必要な場合には適用できないという制約がある。

Chameleon Cloud 上のデータ分析環境再現サービスはオンデマンドで Chameleon Cloud 上にデータ分析環境再現するというアプローチで、分野毎の要求や時間を経過することによる仕様変更などは再現手順を記述する Jupyter Notebook の中で吸収する方式である。

4. 課題

データ分析プロセス共有、すなわち実験手順や実験環境構築手順を公開し共有するためには、論文や研究データと同様に実験手順や実験環境構築手順についても長期保存し、FAIR 原則 [18] に沿った管理がなされる必要がある。

論文と研究データについては、それぞれのライフサイクルが異なっておりそれぞれ独自に DOI を付与するなどして、論文間や研究データ間の関係 (citation) や論文とそこで用いられた研究データの間を DOI 間として管

理することになっている。

この方向性に合わせる形で図5に示すように、実験手順や実験環境構築手順についても DOI を付与し、論文、研究データ、実験手順や実験環境構築手順の間の関連を長期的に管理・保存することが必要である。しかしながら、その具体的な実現方式についてまだ確立したものが存在しないことが本報告で扱う課題である。

研究データのやりとりがさらに円滑になるツールの導入のために、この方式について様々な試行が必要で今後のオープンサイエンス推進のためのエコシステム確立に向けた知見を蓄積していく必要がある。

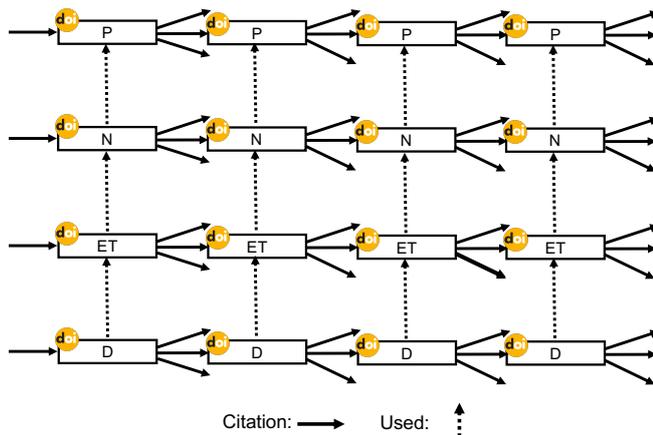


図5 実験手順や実験環境構築手順の長期保存・管理のために

関連研究で触れた Binderhub と Zenodo を組み合わせたソリューションは、この方向で実現されているソリューションであり注目すべきである。その基本的な枠組みとして図6に示すように、オリジナル研究者が自らの実験手順とその実験を実施した環境の構成情報を Git リポジトリに置き、そのコピーを Zenodo に保存すると同時に DOI を取得し、その DOI を論文や研究データと関連付けて公開することで、派生研究者は容易にその情報を発見でき、それを使ってオリジナル研究の実験を BinderHub 上で簡単に再現できる。このように実現されている研究フローは実用的であり、将来性の高いものであると考えられる。

しかしながら、実現されている実験環境としては一つのコンテナで実現できる小規模なものであり、今後益々必要性が増すであろう複数コンピュータノードなどから構成されるクラスタ状の実験環境については、現状そのスコープの外にある。

一方、そのようなクラスタ状の実験環境の再現性までを目指している Chameleon Cloud 上のデータ分析環境再現サービスについては、実験手順や実験環境構築手順についての長期保存・管理のための枠組みが不足している。また、対象とするクラウド基盤が Chameleon Cloud に限定されている。

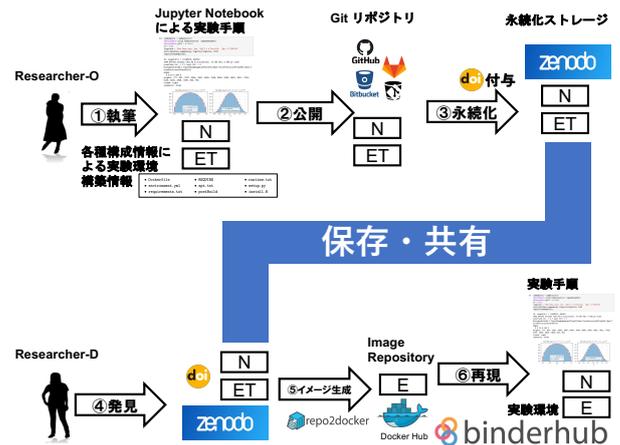


図6 BinderHub + Zenodo ソリューションの使い方

クラスタ状の実験環境の再構築までを可能とし、さらに実験手順や実験環境構築手順についての長期保存・管理のための枠組みを持ったデータ分析プロセス共有をどう実現するかが課題として残っているのである。

本報告の取り組みは、複雑な計算機環境を必要とする分析ノートも実行できる環境構築を再現者がアーカイブサービスからシームレスに実施でき、スムーズにその環境の中でデータ分析を行う環境へ移行できる仕組みを実現することである。

5. 解決策

クラスタ状の実験環境の再構築までを可能とし、さらに実験手順や実験環境構築手順についての長期保存・管理のための枠組みを持ったデータ分析プロセス共有を BinderHub と Zenodo を組み合わせたソリューションと同様に、論文、研究データ、実験手順の長期保存・管理のための枠組みを作る。ただし、BinderHub と Zenodo を組み合わせたソリューションでは実験環境構築手順については実験手順に付随する情報として Git リポジトリ内に保持し、実験環境構築については repo2docker を用いて必要に応じて docker コンテナを build することにより行っている。このため前章で述べたようにクラスタ状の実験環境の再構築ができないという制限につながっている。

すなわち、この実験環境構築手順部分を実験手順に直接付随する形で保存・管理するのではなく、それと独立に管理・保存し、実験手順との関係性を別途管理・保存する仕組みを提案できれば課題解決につながるはずである。

その上で、実際のデータ分析プロセス共有による研究再現の際には、BinderHub と Zenodo を組み合わせたソリューションと同様に、これらの関連性を元に研究者の簡単な指示に従って自動的に実験環境を再構築し、当該の研究データを使った実験手順が実行できるようにする必要がある。

実験環境を再構築する部分の基本的な仕組みは、筆者

表 1 研究再現方式比較

	AiiDA/Material Cloud	Chameleon Cloud データ分析環境再現サービス	BinderHub+Zenodo	OCS+LC4RI	提案方式 BinderHub+Zenodo+OCS+LC4RI
1. 実験手順保存	システム内に Workflow 形式で保存	Jupyter Notebook としてシステム内に保存	Jupyter Notebook として Zenodo 内に DOI を付与し保存	Jupyter Notebook として保存	Jupyter Notebook として Zenodo 内に DOI を付与し保存
2. 実験環境構築手順保存	単一実験環境を研究者間で共有する	実験手順と同一 Jupyter Notebook 内に保存	Docker イメージ構成情報として実験手順の付加情報として保存	LC4RI に沿って Jupyter Notebook で記述	Jupyter Notebook として Zenodo 内に DOI を付与し保存
3. 実験環境再構築	単一実験環境を研究者間で共有するため再構築不要	Jupyter Notebook 実行による再構築	Docker イメージ構成情報から docker build により構築 このため、複数ノードからなる実験環境はスコープ外	Jupyter Notebook で記述された実験環境構築手順を OCS で実行 このため、複数ノードからなる実験環境も構築可能	Jupyter Notebook で記述された実験環境構築手順を OCS で実行 このため、複数ノードからなる実験環境も構築可能
4. 実験環境再構築先	AiiDA	Chameleon Cloud	BinderHub	マルチクラウド対応	マルチクラウド対応

らの既存研究 [19] の「可搬性の高いデータ分析基盤の構築方式」で述べたように、学認クラウドオンデマンド構築サービス（以下 OCS: NII Gakunin Cloud On-demand Configuration Service）[20,21] と Literate Computing for Reproducible Infrastructure(以下 LC4RI) [22] を組み合わせることで実現できる。

OCS は、オンデマンドにクラウド環境を構築するソフトウェアサービスであり、クラウド環境構築用テンプレートを指定して起動するだけで、クラウド環境の構築・再構築が可能となる。特徴の一つは、クラウドプロバイダごとの操作方法の違いを吸収し、複数のクラウドプロバイダの同時利用や切り替えの煩雑さを軽減することである。

一方、LC4RI は、計算機システムの構築・運用の方法論であり、システムの構築・運用作業に必要な自然言語による技術情報、実行コード、さまざまな状況で起こり得る複数の実行結果を Jupyter Notebook として 1 つの文書にパッケージし、実行可能な手順書を実現している。

我々のこの OCS+LC4RI で実現する「可搬性の高いデータ分析基盤の構築方式」と「BinderHub+Zenodo」を組み合わせることで図 7 に示すように、解決策に沿ったプロトタイプを作ることでできる。この中で BinderHub 上に構成される OCS クライアントコンテナというのは Jupyter Notebook で記述された実験環境構築手順を実行するための Jupyter Notebook 環境で OCS 利用に必要な VCP SDK などを持っているものである。

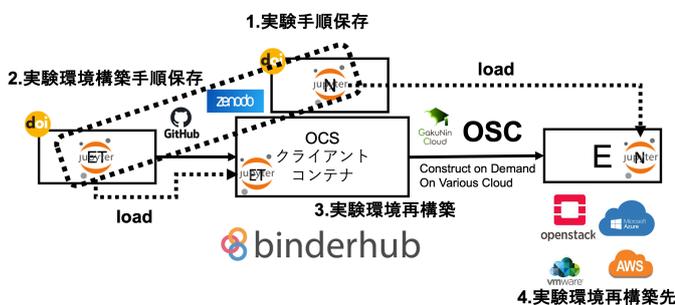


図 7 提案するソリューションの構成

ここまでの議論を整理するために、OCS 関連研究で紹介した研究再現方式との比較を表 1 にまとめて示した。この比較表で示すように提案方式では、実験手順や実験環境構

築手順についての長期保存・管理のための枠組みを持ち、さらにクラスタ状の実験環境の再現性までをマルチクラウドを対象に実施することができる。

図 6 の BinderHub+Zenodo のソリューションで実験環境を各種構成情報から Docker イメージにより再構成する部分を、OCS を用いて、図 8 に示すように、そのサービスのクラウド環境構築用テンプレート (ET) として提供される Jupyter Notenook で記述された実験環境構築手順書を用いて、各種クラウド基盤上に再構成する。その実験環境の中に Jupyter Notenook で記述された実験手順 (N) を持ち込むことでオリジナル研究者が行ったものと同様の実験を再現することができる。さらには実験実行時のパラメータを変更したり研究データを変更するなどして派生研究へスムーズにつなげることが可能となる。

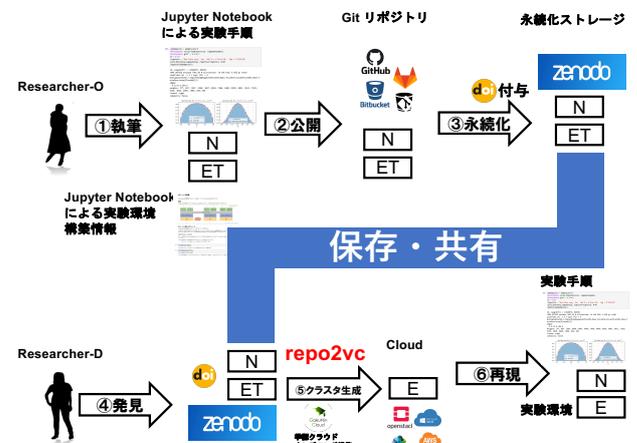


図 8 提案するソリューションの使い方

派生研究者は、Zenodo などの永続化ストレージに保存されている Jupyter Notenook で記述された実験手順 (N) を選択し、再構築のためのボタンをクリックするだけで、その実験手順に used リンクで紐づけられている Jupyter Notenook で記述された実験環境構築手順書 (ET) を呼び出し、BinderHub+Zenodo のソリューションの場合の repo2docker に相当する repo2vc を使うことで自動的に OCS 経由で必要なクラスタ (VC: Virtual Cloud) が構成され、その中に Jupyter Notenook で記述された実験手順 (N) が呼び込まれ、その手順実施のページに誘導される。これ

で実験再現を開始できる状態となる。

6. 研究再現への適用例

前章で述べた解決方針に従って具体的な研究再現を実行し、提案する解決策の妥当性を検証してみたい。まず、提案するソリューションの実現性を検証するためにそのプロトタイプを構成する。様々な実現形態が考えられるけれど、今回は実際に利用が進んできている BinderHub+Zenodo のソリューションの発展形態として位置付けることを重要視し、その拡張としての図9に示すような構成を採用した。

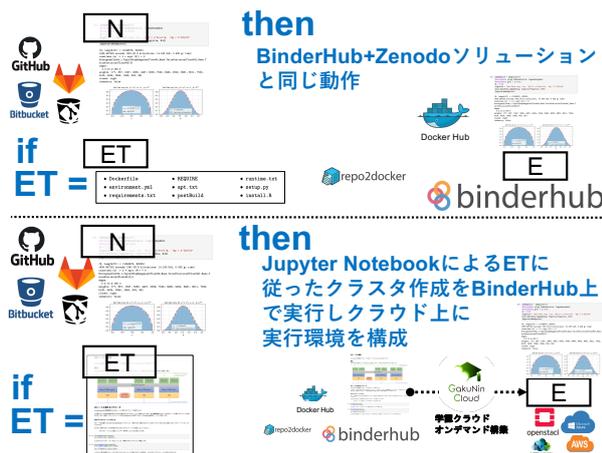


図9 repo2vc の動作

Git リポジトリ上の状況で repo2docker で扱える範囲の構成情報を実験環境として指定してある場合 (図9の上の場合) は repo2vc は現状の BinderHub+Zenodo のソリューションの動きに合わせて repo2docker として動作する。このことで今まで BinderHub+Zenodo のソリューションのために作成されている Git リポジトリはそのまま使えることに加え、研究者からの利用インタフェースも変更する必要がない。

一方、実験環境としてクラスタが必要で構築のために必要は手順が LC4RI と OCS に従った Jupyter Notebook として Git リポジトリ内に配置されている場合 (図9の下の場合) には、まずその ET である Jupyter Notebook を動作させるための環境を BinderHub+Zenodo のソリューションを使って用意する。この際利用する Docker イメージは LC4RI と OCS に従った Jupyter Notebook が動作するものでなければならぬので、その構成は Dockerfile として提供されているものとする。

このことにより研究者は実験環境構成手順の書かれた Jupyter Notebook の実行を BinderHub 上で促され、ターゲットとするクラウド基盤の指定やクラスタのサイズなど実験環境構築に必要なパラメータを設定することとなる。このパラメータの一つとして実験手順が記述された Jupyter Notebook の DOI も引き渡される。クラウド上に

構築した実験環境でこの実験手順に従って実験が進められる。この際、この BinderHub からアクセスできる場所に OCS を利用するために必要な機能である OCS が提供する VCC (Virtual Cloud Controller) が存在するとする。この VCC から各種クラウド基盤に VCP SDK を利用して実験環境を再現するのである。

具体的な研究再現例としては実験数学分野の佐藤予想の再現実験を扱う [23]。この再現実験は実験対象とする曲線を並列に多数対象とするため、図10に示すように、クラウド内に並行実行するための計算ノードを確保し、そのパラメータ指定が可能な Jupyter Notebook を実験手順として Git リポジトリ内に保存しておく。パラメータの指定範囲は外部の ethercalc [24] のような部分に蓄積して各計算ノードから参照し未実施のパラメータを獲得してそのパラメータを指定し papermill [25] により Jupyter Notebook を起動することで並列計算を実施する。計算結果自体は外部のストレージに蓄積する構成となっている。

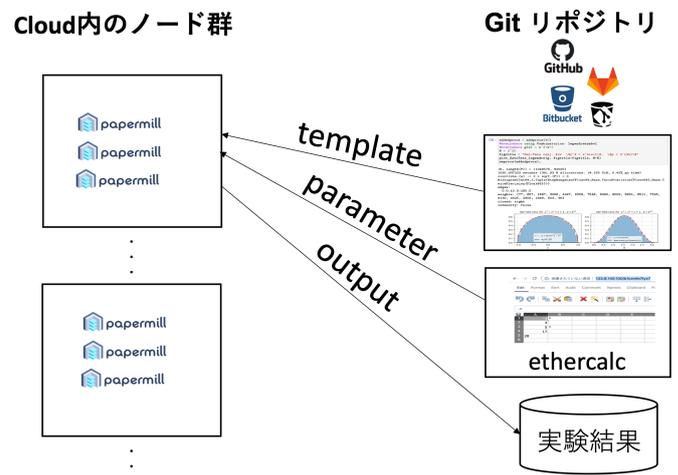


図10 佐藤予想の再現実験環境

この実験環境を再現するために以下の手順を記述した Jupyter Notebook を作成する。

- クラウド基盤内に必要な数のノードを確保
- 各ノードに Docker 実行環境インストール
- 各ノードに papermill および Jupyter with Julia kernel などの環境の入った Docker イメージを配備

この Jupyter Notebook を Git リポジトリに配置し、Zenodo を用いて DOI 付与と長期保存を行う。

DOI 10.5281/zenodo.3977447

元々の佐藤予想の再現実験のための Jupyter Notebook も同様に Git リポジトリに配置し、そちらには付加情報として前者の実験環境構築手順 Jupyter Notebook の DOI を配置する。この配置が used リンクに相当する。この後、実

手順 Jupyter Notebook についても、同様に Zenodo を用いて DOI 付与と長期保存を行う。

DOI 10.5281/zenodo.3977810

これらの準備の元、この実験手順 Jupyter Notebook の DOI をオリジナル論文の参照情報などから発見した研究者は、そこに付随した実験再現ボタンを押下することで、図 8 の下のプロセスに沿った実験再現が可能となる。

7. 今後の取り組み

今回紹介したプロトタイプシステムは、まだモックアップでしかなく、実際の OCS クライアントコンテナ用の Dockerfile は公開されていないため別の Dockerfile で代用した。また、BinderHub(mybinder.org) 内に立ち上げた OCS クライアントコンテナからクラウド基盤を制御するためには、国立情報学研究所が運用する VCC にそこから到達する必要がある。このため、これらの制約が無くならないと、今の構成で実際に利用者を受け入れるためのサービスを構成することはできない。

前者の制約は、近い将来解決できる見通しがあるけれど、後者についてはセキュリティ的な観点も含めてさらに議論をする必要がある。別のソリューションとしては BinderHub のインスタンスを各組織が利用する VCC と同一のプライベートな空間にデプロイし、それを利用することも考えられる。

これらの検討も含め、実際にいくつかのプロトタイプを構築し、利用経験を蓄積することとすることで、他機関の利用可能なサービスとしての展開を検討したい [26]。

この方式は OCS を活用することでそのターゲットとするクラウド基盤を幅広く選択できるというメリットがある。具体的な想定ターゲットとしては、各種パブリッククラウドやプライベートクラウドはもとより、大規模なオープンデータの蓄積が期待されている各種学術クラウドもターゲットとして有力であり、それらを使った具体的な研究再現の実績を蓄積したいと考えている。

参考文献

- [1] 日本学術会議オープンサイエンスの深化と推進に関する検討委員会：オープンサイエンスの深化と推進に向けて、<http://www.scj.go.jp/ja/info/kohyo/pdf/kohyo-24-t291-1.pdf> (accessed on 07-07-2020).
- [2] 喜連川優：オープンサイエンスと研究データ基盤、学術の動向、Vol. 24, No. 10, pp. 86–89 (2019).
- [3] Hasselbring, W., Carr, L., Hettrick, S., Packer, H. and Tiropanis, T.: From FAIR research data toward FAIR and open research software, *it - Information Technology*, Vol. 62, No. 1, pp. 39–47 (online), DOI: 10.1515/itit-2019-0040 (2020).
- [4] 山地一禎：日本の研究データ基盤の構築、<https://rcos.nii.ac.jp/document/> (accessed on 07-07-2020).

- [5] 武田英明, 村山泰啓, 中島律子: 研究データへの DOI 登録実験, 情報管理, Vol. 58, No. 10, pp. 763–770 (オンライン), DOI: 10.1241/johokanri.58.763 (2016).
- [6] ジャパンリンクセンター運営委員会: 研究データへの DOI 登録ガイドライン (2015).
- [7] 大向一輝: オープンサイエンスと研究データ共有, 心理学評論, Vol. 61, No. 1, pp. 13–21 (2018).
- [8] DataCite: DataCite, <https://www.datacite.org/> (accessed on 07-07-2020).
- [9] AiiDA: AiiDA, <http://www.aaida.net/> (accessed on 07-07-2020).
- [10] AiiDA: Material Cloud, <https://materialscloud.org/> (accessed on 07-07-2020).
- [11] Jupyter: BinderHub, <https://mybinder.org/>, <https://github.com/jupyterhub/binderhub> (accessed on 07-07-2020).
- [12] Jupyter-Project, Bussonnier, M. et al.: Binder 2.0 - Reproducible, interactive, sharable environments for science at scale, pp. 113–120 (online), DOI: 10.25080/Majora-4af1f417-011 (2018).
- [13] CERN: zenodo, <https://zenodo.org/> (accessed on 07-07-2020).
- [14] Anderson, J. and Keahey, K.: A Case for Integrating Experimental Containers with Notebooks, *11th IEEE International Conference on Cloud Computing (Cloud-Com 2020)* (2019).
- [15] Docker: Docker, <https://www.docker.com/> (accessed on 07-07-2020).
- [16] Jupyter: repo2docker, <https://github.com/jupyter/repo2docker> (accessed on 07-07-2020).
- [17] Forde, J., Head, T., Holdgraf, C., Panda, Y., Nalvarete, G., Ragan-Kelley, B. and Sundell, E.: Reproducible research environments with repo2docker (2018).
- [18] FORCE11: FAIR 原則, <https://www.force11.org/fairprinciples> (accessed on 07-07-2020).
- [19] 横山重俊, 浜元信州, 長久 勝, 藤原一毅, 政谷好伸, 竹房あつ子, 合田憲人: データ駆動科学研究の再現性を支えるデータ分析環境再構成方式, 第 48 回インターネットと運用技術研究会 2020-IOT-48(13), 情報処理学会, pp. 1–8 (2020).
- [20] 竹房あつ子, 佐賀一繁, 丹生智也, 横山重俊, 合田憲人: 学認クラウドオンデマンド構築サービスの推進, AXIES 2018 年度年次大会 (2018).
- [21] 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> (accessed on 07-07-2020).
- [22] 長久 勝, 政谷好伸, 谷沢智史, 中川晋吾, 合田憲人: Notebook を介した作業ノウハウの継承・移転を分析するための基盤, 第 44 回インターネットと運用技術研究会 2019-IOT-44(16), 情報処理学会, pp. 1–6 (2019).
- [23] 黒木 玄: 佐藤・Tate 予想のプロット, <https://nbviewer.jupyter.org/gist/genkuroki/fdcc26c337d6c22f85087b4f76587979> (accessed on 07-07-2020) (2018).
- [24] Ethercalc: Ethercalc, <https://ethercalc.net/> (accessed on 07-07-2020).
- [25] Jupyter: Papermill, <https://papermill.readthedocs.io/> (accessed on 07-07-2020).
- [26] 藤原一毅: RDM と連携するデータ解析サービス, <https://www.nii.ac.jp/openforum/upload/4f8009421d57f2ae55d72996c9c047f0e9b3e446.pdf>, https://www.youtube.com/watch?v=Jf3E1k_525g (accessed on 07-07-2020) (2020).