

GakuNin RDM 上でのデータ ガバナンスの実現に向けて

国立情報学研究所
オープンサイエンス基盤研究センター
データガバナンス機能担当
平木俊幸、横山重俊

2023年5月29日

Table of contents

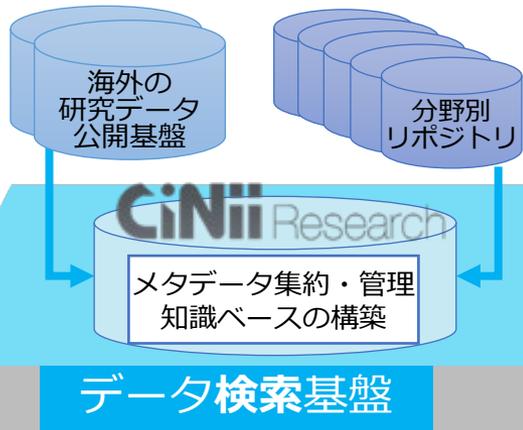
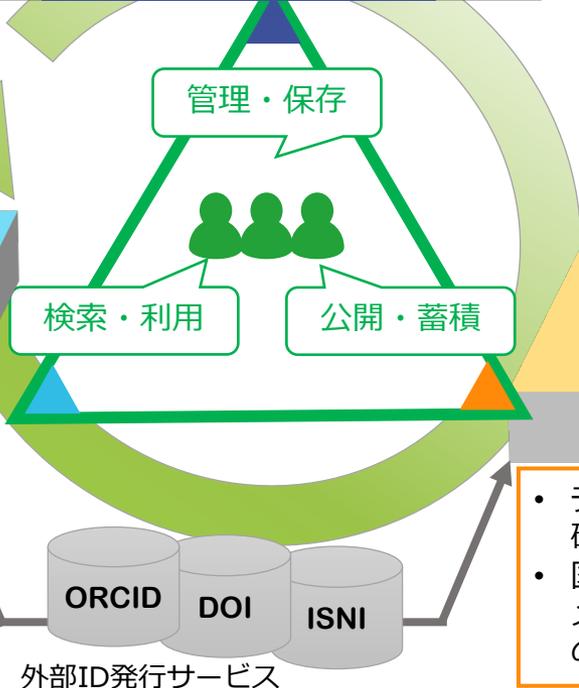
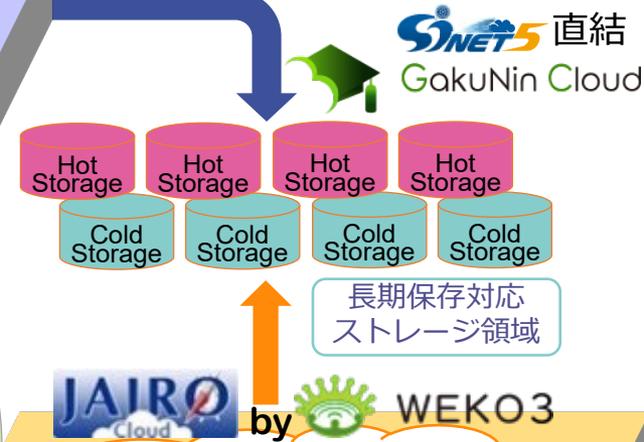
1. データガバナンス機能とは
2. データガバナンス機能のデモ
3. 今後の進め方

1. データガバナンス機能とは

研究データ基盤 NII Research Data Cloud (NII RDC) の3つの基盤

Open science と研究公正を支え、データ駆動型研究を推進する情報基盤

2017年から開発開始、
2021年から運用開始

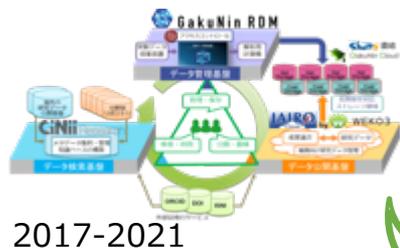


- データ収集装置や解析用計算機とも連携
- 研究遂行中の研究データなどを共同研究者間やラボ内で共有・管理
- 組織が提供するストレージに接続した利用が可能

- 機関リポジトリ+分野別リポジトリやデータリポジトリとも連携
- 研究者や機関、研究プロジェクトの情報と関連付けた知識ベースを形成
- 研究者による発見プロセスをサポート

- データ管理基盤における簡便な操作で研究成果の公開が可能
- 図書館員やデータキュレータによる、メタデータや公開レベル統計情報などの管理機能の提供

NII RDC における 7 機能と データガバナンス機能の位置づけ



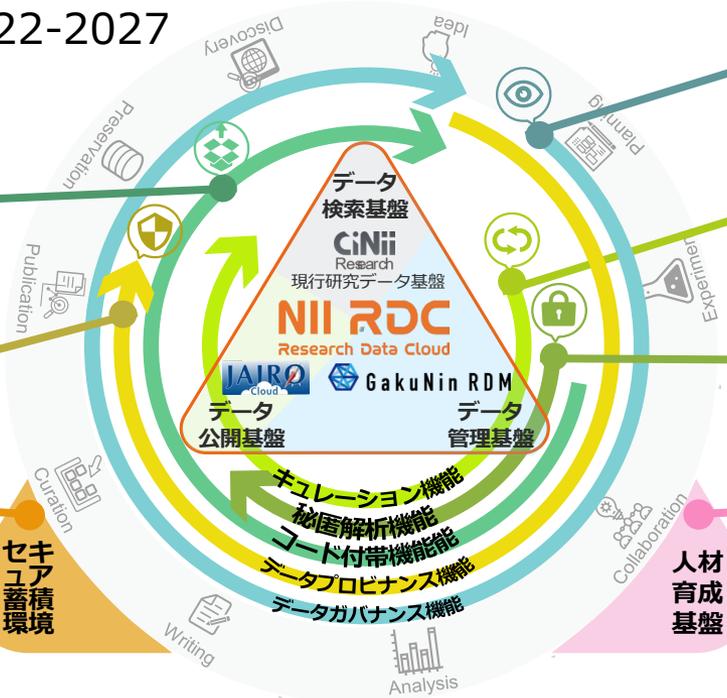
NII RDC を 7 つの側面から機能拡張

活用 **コード付帯機能**
データ・プログラム・解析環境の
パッケージ化と流通機能を提供し、
研究成果の再現性を飛躍的に向上

信頼 **データプロビナンス機能**
データの来歴情報の管理から利用
状況を把握でき、データ公開への
インセンティブモデルを提供

蓄積 **セキュア蓄積環境**
安全で強固なデータの保存・保護機
能を有する超鉄壁ストレージを提
供し、機微な情報も安心して保全

セキュア蓄積環境



データガバナンス機能 **管理**
計画に基づきデータ管理等を機械
的に支援し、DMPをプロジェクト
管理に不可欠な仕組みへと変革

キュレーション機能 **流通**
専門的なキュレーションを実践
できるエコシステムを構築し、
データ再利用の促進に寄与

秘匿解析機能 **保護**
秘密計算技術で機微な情報も安心し
て解析できる環境の提供で、新し
いデータ駆動型研究の世界を開拓

人材育成基盤 **育成**
RDMに必要なスキルを学ぶ環境
を提供し、全ての研究者を新し
い科学の実践者へと育成

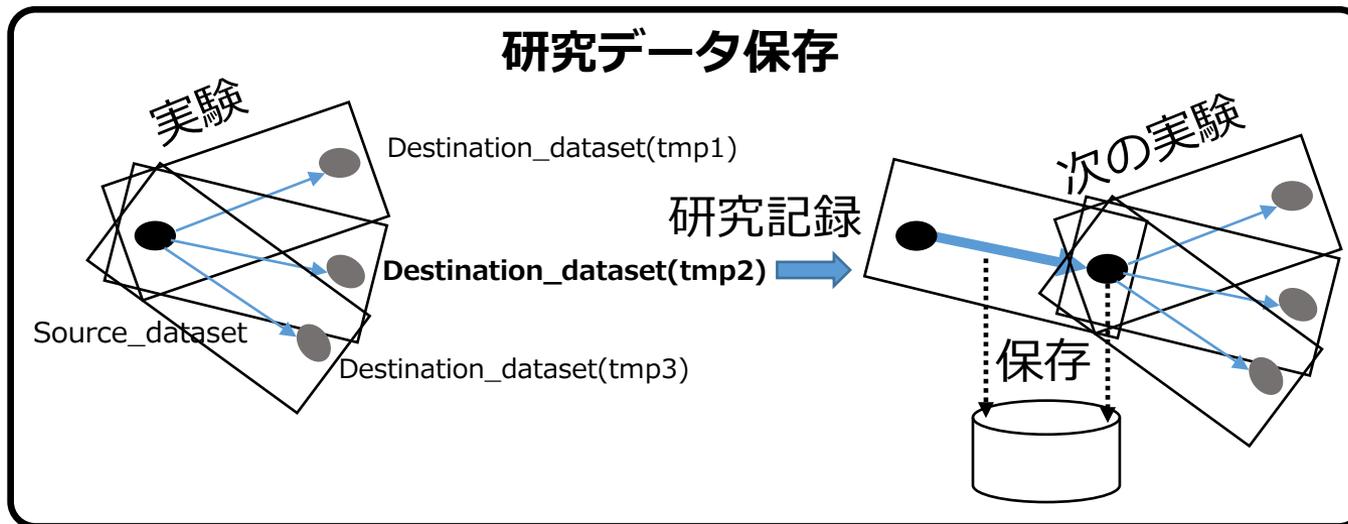
人材育成基盤

データガバナンス機能は研究活動全体にわたって
研究データ管理の品質の向上をサポートする。

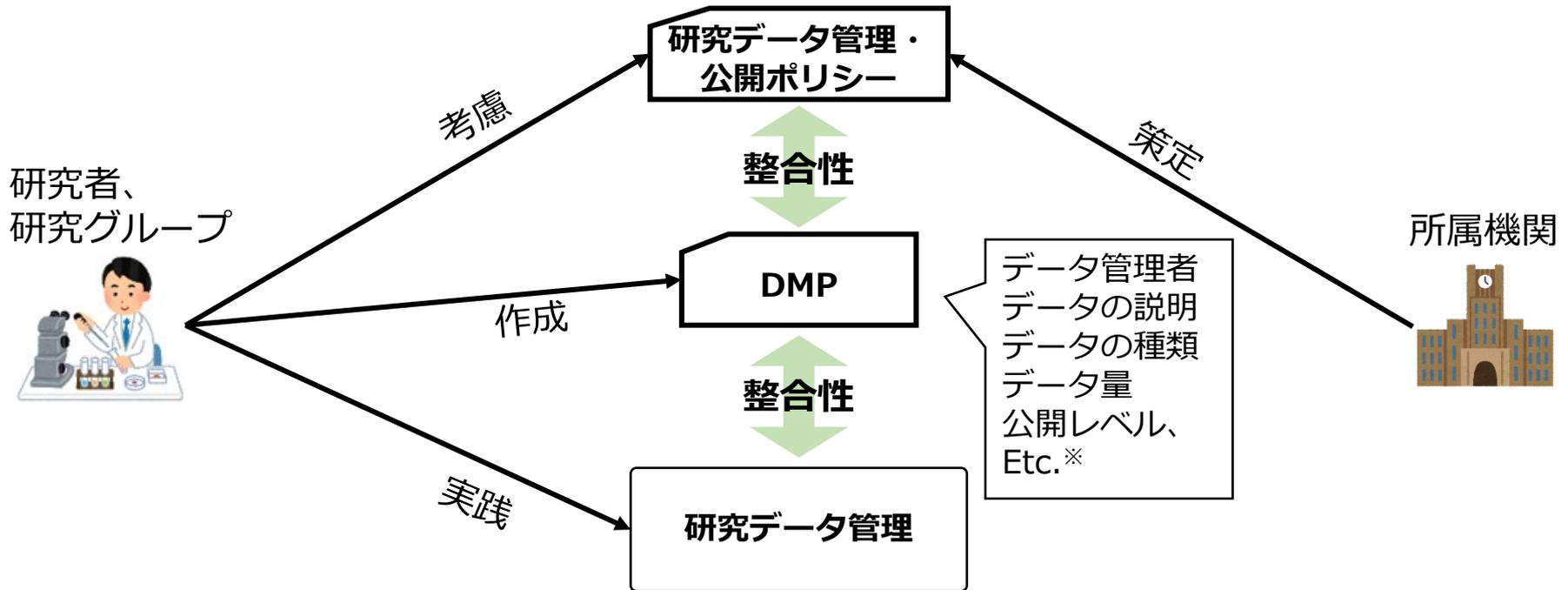
研究データ管理とは

研究データ※の取得・加工・分析・保存・公開などの行為

※研究データ：研究者が研究利用等の対象としたデータ。



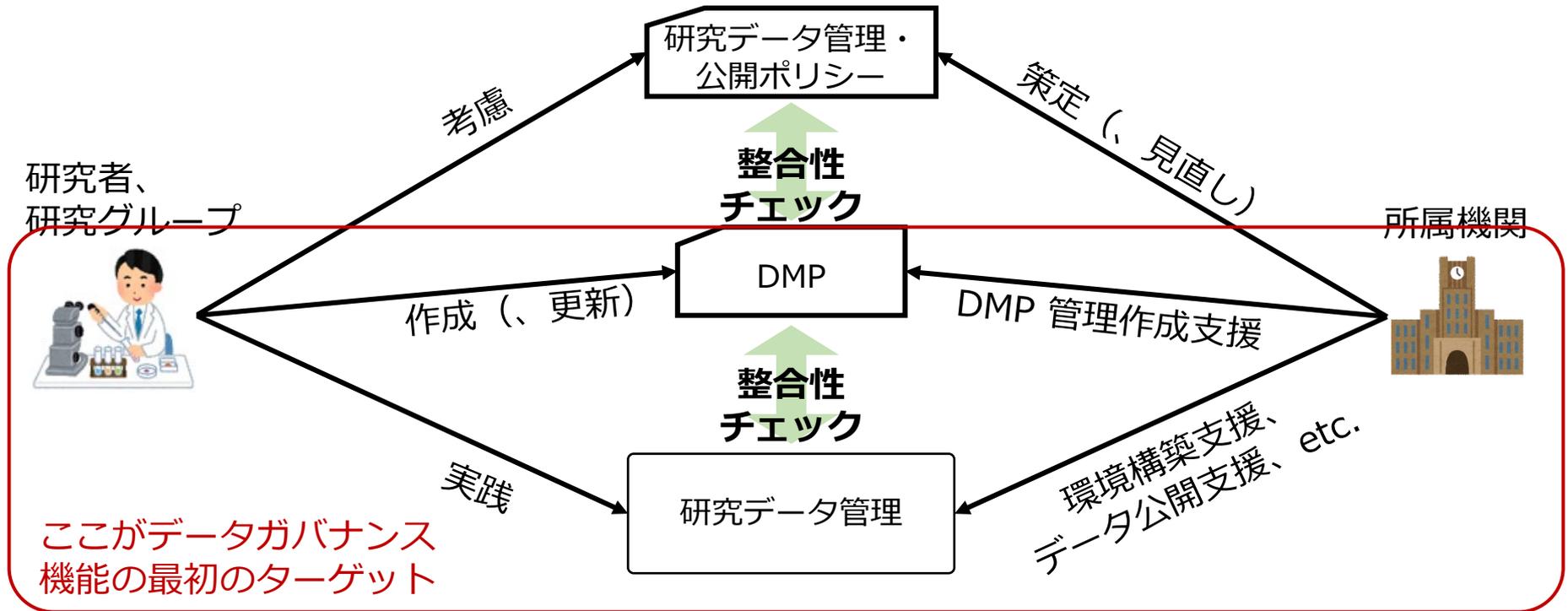
データ管理計画（DMP）とは



**研究のために収集・作成する研究データの取扱いや
整備・保存・公開についての計画を定めた文書**

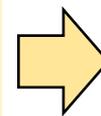
※DMPの記述内容は提出先の助成機関によって異なる。

研究データガバナンスとは



研究データ管理の品質を守る・向上するための規律に従う行為。

- ポリシーを策定し、運用する。（研究機関）
- ポリシーおよび管理計画に従う。（研究者）
- ポリシーに応じて環境を整備/利用する。（研究機関/研究者）



研究活動の促進、
組織・研究者としての
信用の向上

GakuNin RDM における データガバナンス機能の課題設定

課題

- 省庁や助成機関が要求する DMP を作成することが必要。しかしそれを研究サイクルに活かすことが難しい。
- ✓ 計画に沿った研究データ管理は研究者の裁量に...
- ✓ 研究管理部門も計画通りに研究データ管理がなされているか確認するのは手間...
- ✓ DMP が管理文書として蓄積される以外の使い道がない...

No.	データ名	データの説明	管理者	分類	公開レベル	秘匿理由
1	〇〇実証においてセンサより撮像したデータ及び関連データ	〇〇実証においてセンサより撮像したデータであり、道路の画像データ	〇〇研究所	委託者指定データ	レベル4 (広範な提供・利活用)	秘匿しない



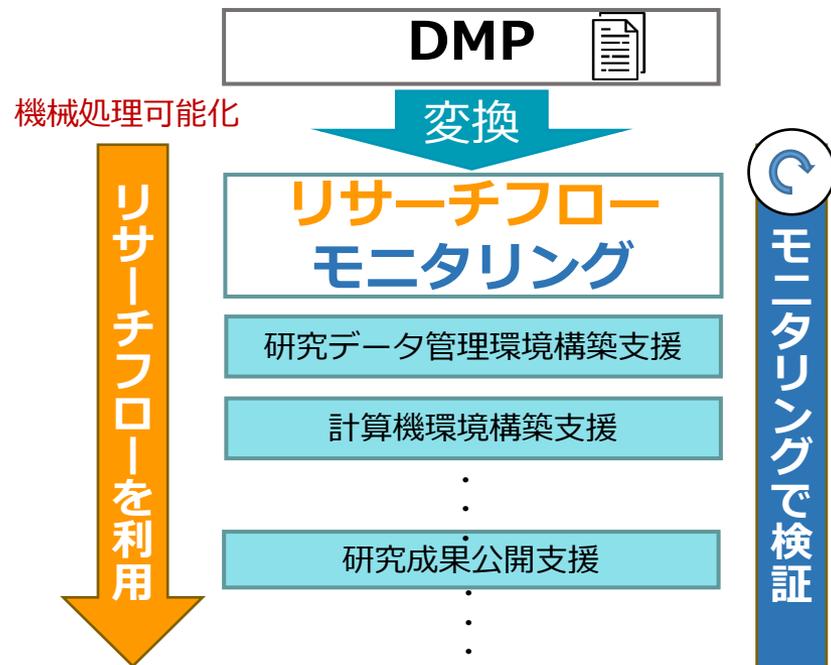
提出

それで終わり



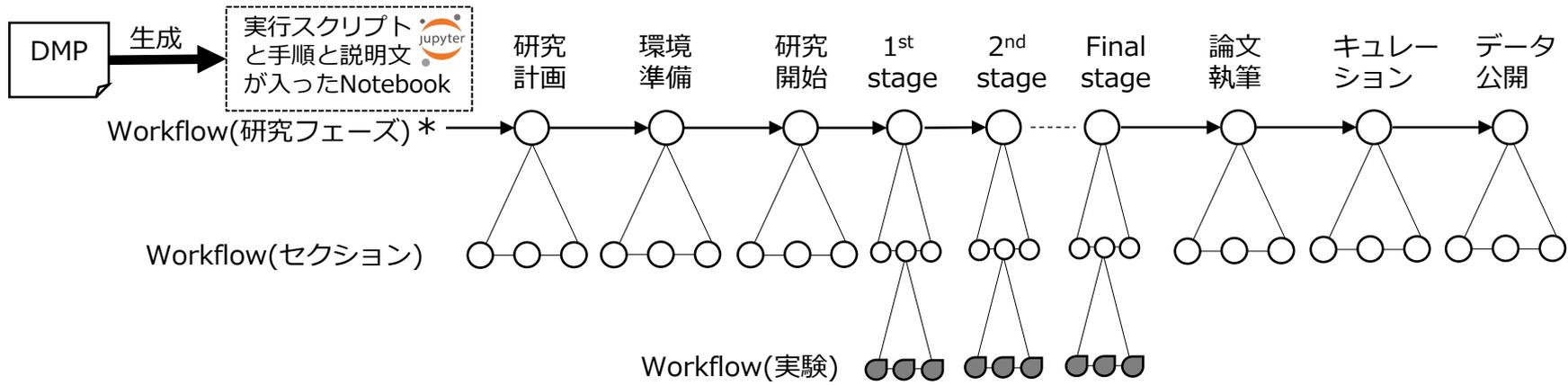
解決案

- DMP をもとに半自動で適切な研究データ管理環境を生成。
- ✓ DMP から生成されたリサーチフローで研究データ管理品質が担保される。
- ✓ DMP に沿った研究データ管理がなされていることをモニタリングにより継続的に検証し、研究データ管理品質が担保される。

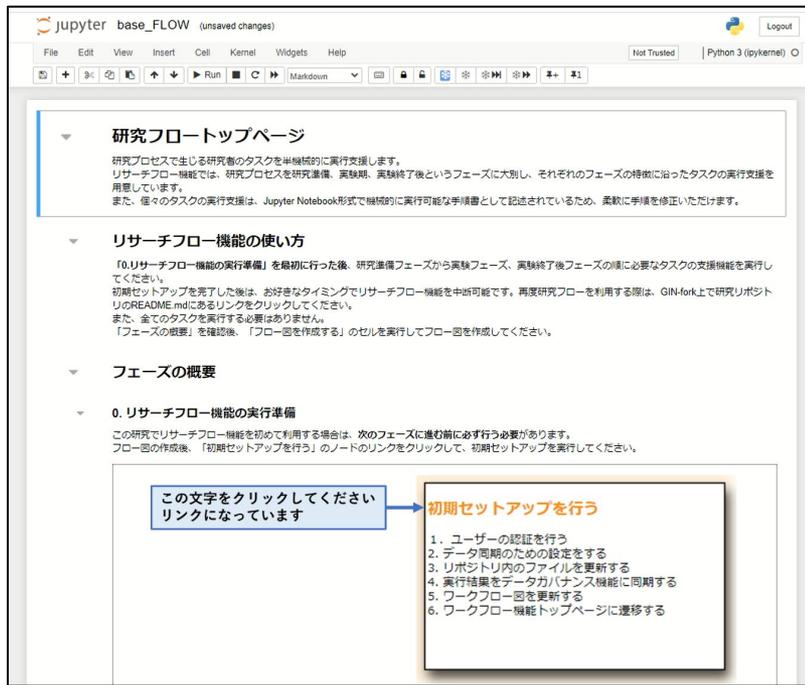


リサーチフロー

DMP によって表現された計画に基づき生成された、実行可能な手順書



リサーチフロー (Notebook) の例



モニタリング

研究データの状態が DMP 等による制約を満たしているかどうか検証する

研究者

DMPで管理側とコミュニケーション
(データガバナンス機能を使って
DMPに沿ったデータ管理を実施)

研究管理者、助成機関

DMPで研究者とコミュニケーション
(研究データ管理の実態との整合性は
データガバナンス機能が保証していると期待)

DMP

データガバナンス機能

生成

研究データ管理のための Metadata Schema とその検証ルール[※]の定義

プランの実行に必要な制約



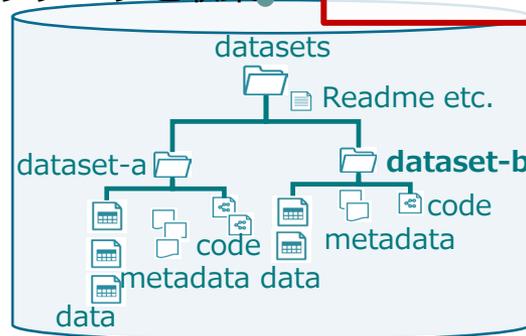
研究データの検証

メタデータのコレクション(状態)

システムから
メタデータを収集

研究データのパッケージング

研究者



※分野ごとの schema と検証ルールのユニーク性は、それぞれのコミュニティで拡張する形で実現可能。

ポリシー
↓
プラン
↓
データ管理実施

2. データガバナンス機能のデモ

デモの内容

研究データ管理状態をモニタリング機能を用いて検証し、NG であればその原因を把握でき、解決する。

【想定シーン】

論文査読に向けて、論文の根拠となる研究データを査読者が再現可能な状態を作り出すための準備を進めている。

査読者へ出す前に、現時点での研究データ管理状態が以下の二つのレベルで問題ないかどうか検証する。

- ① 関連ファイルの存在性
- ② 再現性

【デモの流れ】

論文の根拠となる研究データの再現性を、モニタリング機能を用いて検証したところ、**NG**が返ってきた。

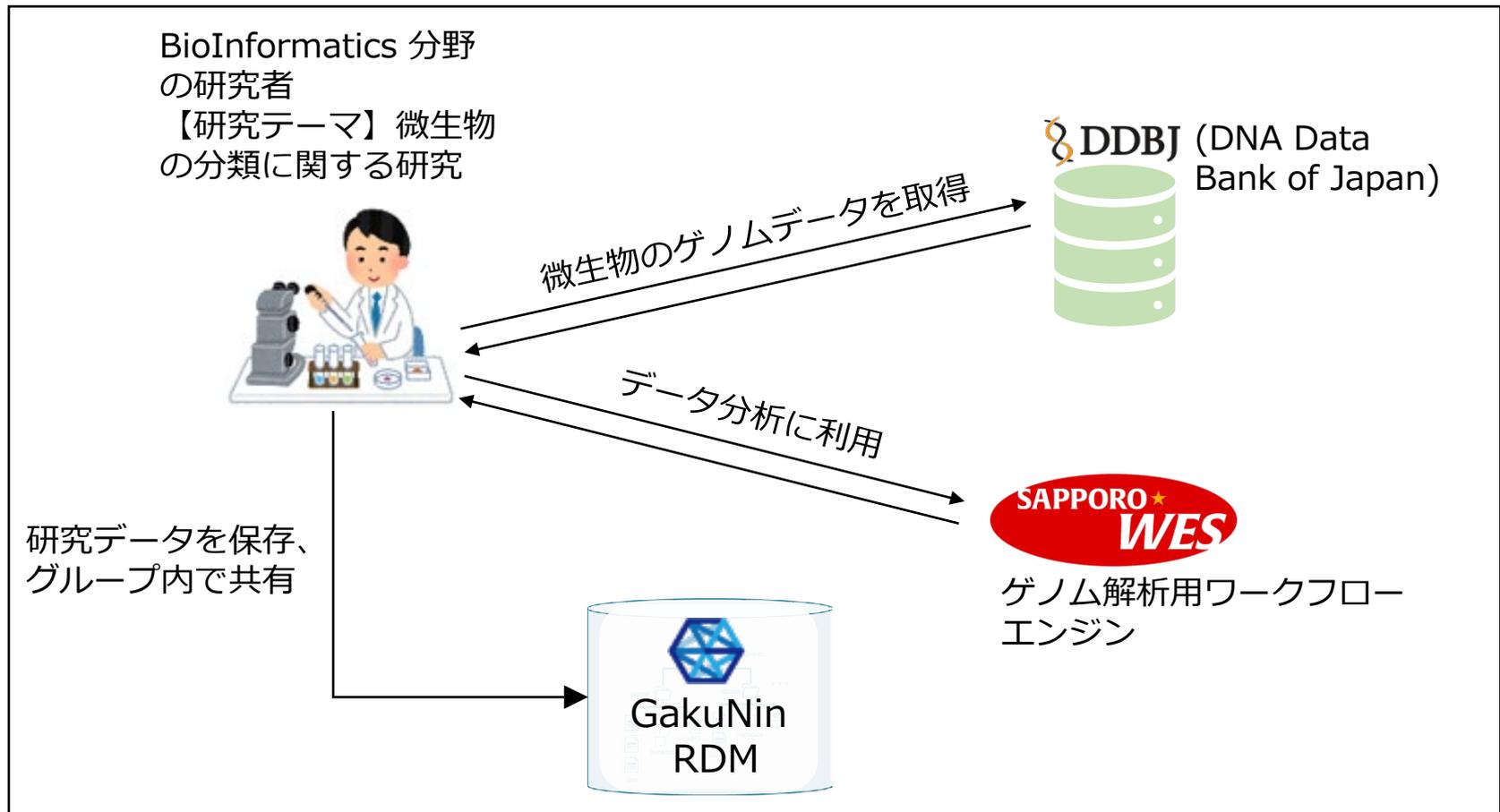
検証結果の記録にあるNGの理由が分かるメッセージを確認し、問題個所を探す。

問題個所が分かったのでそれを解消する。

モニタリング機能を用いて再現性を再検証する。

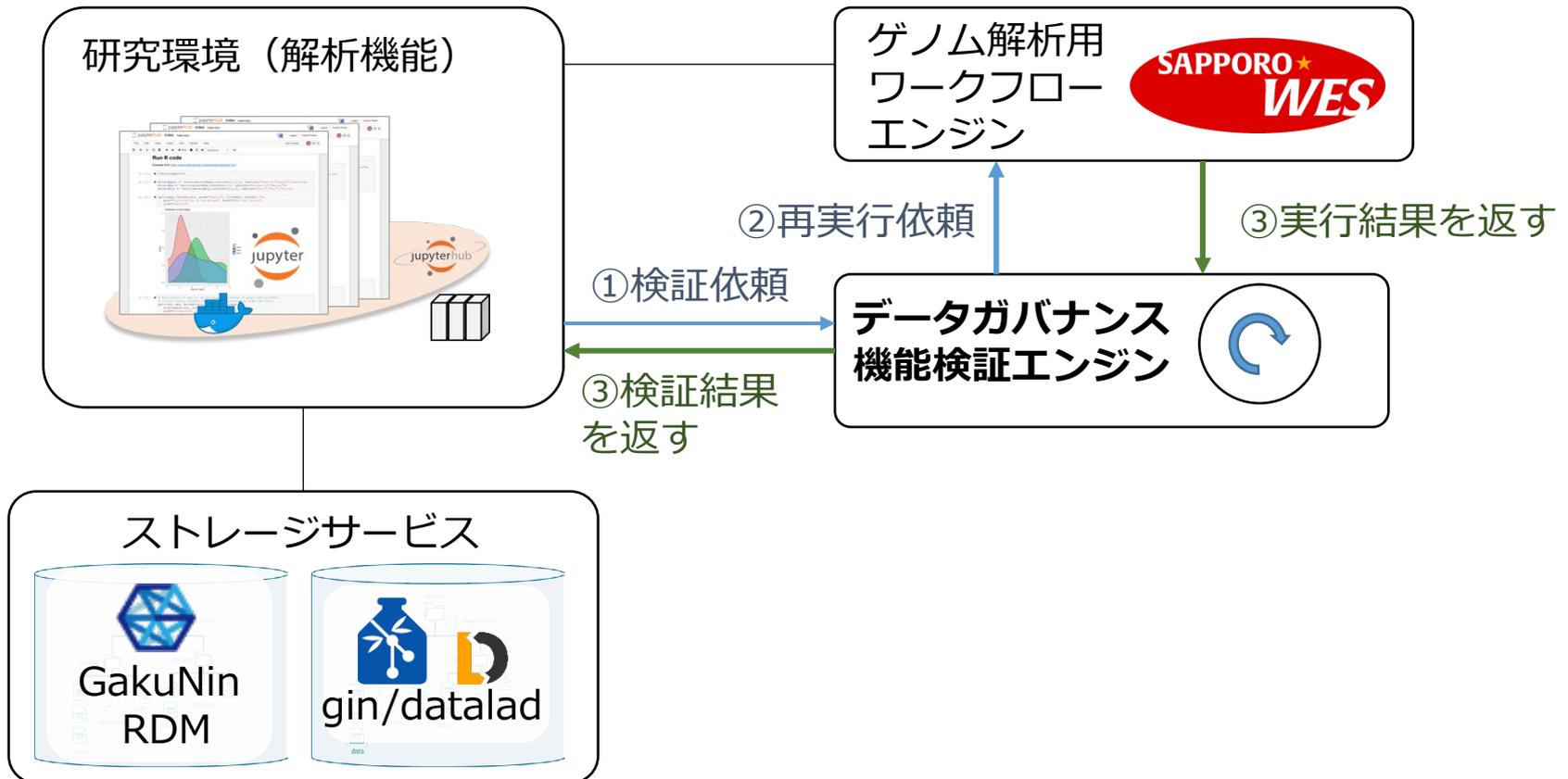
検証の結果、**再現性が確認された**。

デモで想定しているユースケース



↑この研究活動のうち、今回は分析結果に関連するファイルの存在性と分析結果の再現性を検証する。

デモのシステム構成



今回のデモでは gin/datalad を
用いて研究データを管理

デモでの DMP の一部と Metadata schema + 検証ルール

Moohshot
(MS) 型研究開
発制度におけ
るメタデータ
の一例※

リサーチフロー
の構造および
GIN リポジトリ
のサイズを決め
るメタデータ

項目名	項目の説明
資金分配機関情報	公募型の研究資金を配分した資金 配分機関（府省含む）の英語略称
プロジェクト名	プロジェクトの研究代表者が統括 する研究開発の範囲の名称
データの名称	管理対象データの特徴を示す名称 を入力
概略データ量	管理対象データの概ねのデータ容 量
ワークフロー識別子	使用するリサーチフローの種別
想定利用最大データ量	リポジトリに格納するデータの想 定利用最大値
データセット構造種別	実験データのディレクトリ構造の 種別

Metadata schema
+ 検証ルール

Moohshot 用

プロジェクト用
メタデータの検証で利用



GIN 用

リサーチフロー生成
で利用



Sapporo 用

再現性検証で利用

※詳細説明は「ムーンショット型研究開発制度におけるメタデータ説明書（第2版）」
(https://web.archive.org/web/20230309075934/https://www8.cao.go.jp/cstp/ms_metadatainstructions.pdf)を参照。

検証実施の様子（再現性検証ページ）

The screenshot shows a Jupyter Notebook interface with the following content:

- 研究フロートップページ**
研究プロセスで生じる研究者のタスクを半機械的に実行支援します。リサーチフロー機能では、研究プロセスを研究準備、実験期、実験終了後というフェーズに大別し、それぞれのフェーズの特徴に沿ったタスクの実行支援を用意しています。また、個々のタスクの実行支援は、Jupyter Notebook形式で機械的に実行可能な手順書として記述されているため、柔軟に手順を修正いただけます。
- リサーチフロー機能の使い方**
「0.リサーチフロー機能の実行準備」を最初に行った後、研究準備フェーズから実験フェーズ、実験終了後フェーズの順に必要なタスクの支援機能を実行してください。初期セットアップを完了した後は、お好きなタイミングでリサーチフロー機能を中断可能です。再度研究フローを利用する際は、GIN-fork上で研究リポジトリのREADME.mdにあるリンクをクリックしてください。また、全てのタスクを実行する必要はありません。「フェーズの概要」を確認後、「フロー図を作成する」のセルを実行してフロー図を作成してください。
- フェーズの概要**
- 0. リサーチフロー機能の実行準備**
この研究でリサーチフロー機能を初めて利用する場合は、次のフェーズに進む前に必ず行う必要があります。フロー図の作成後、「初期セットアップを行う」のノードのリンクをクリックして、初期セットアップを実行してください。

この文字をクリックしてください
リンクになっています

初期セットアップを行う

1. ユーザーの認証を行う
2. データ同期のための設定をする
3. リポジトリ内のファイルを更新する
4. 実行結果をデータガバナンス機能に同期する
5. ワークフロー図を更新する
6. ワークフロー機能トップページに遷移する

初期セットアップ完了後、このトップページに、研究リポジトリ配下のREADME.mdのリンクから遷移した場合は実行する必要がありません。ただし、以下のmaDMP実行環境への遷移ボタンから遷移し実行環境を再構築した場合は、再度「初期セットアップを行う」を実行してください。

検証実施の様子（検証実施）

実験の再現性を検証する

実験パッケージの
データを取得し、
再現性を検証する

```

# 再現する実験パッケージを選択
def on_click_callback(clicked_button: Button) -> None:
    global package
    package=dropdown.value
    clear_output()
    print("入力を受けつけました：", package)

dropdown = Dropdown(
    options=pipeline,
    description='実験パッケージ名:',
    disabled=False,
)

button = Button(description='入力完了')
button.on_click(on_click_callback)
display(dropdown, button)

```

入力を受けつけました： exp1

3. 実験パッケージのデータを取得する
4. 再現性に関わるメタデータを検証する
5. 検証結果を確認する
6. 研究リポジトリに同期する
このタスクの実行結果を研究リポジトリに同期します。
検証結果を同期するか破棄するかは、「5.1. 検証結果を研究リポジトリに同期するか選択する」で選択できます。
- 6.1. 検証結果を研究リポジトリに同期するか破棄するかを選択する

```

In [ ]: from WORKFLOWS.utils.tap_validation import tap_validation
tap_validation.get_validation_results()

In [ ]: import panel as pn
pn.extension()
column = pn.Column()

```

検証実施の様子（検証結果がNG）

実験の再現性を検証する

検証を実行する

検証の結果、NGとなる
(今回は再現性が検証
できなかった)

5. 検証結果を確認する

```
[7]: from WORKFLOWS.utils.tmp_validation import tmp_validation
tmp_validation.get_validation_results()
ERROR:root:再現性が検証されませんでした。確認後、「確認を完了する」ボタンをクリックして次にお進みください。
「確認を完了する」ボタンがクリックされていない場合は、検証結果を含んだこのノートブックがリポジトリに同期されます。
[
  {
    "entityId": "#sapporo-run",
    "props": "sapporo.SapporoRun:state",
    "reason": "The status of the workflow execution MUST be COMPLETE; got EXECUTOR_ERROR instead. Please check the log of the
workflow execution using GET http://dg02.dg.rcos.nii.ac.jp:3000/runs/3dde09f3-a4b9-4ed8-ab0f-770c8775f73d ."
  }
]
==== http://dg02.dg.rcos.nii.ac.jp:3000/runs/3dde09f3-a4b9-4ed8-ab0f-770c8775f73d より取得した実行エラーは以下です。====
Traceback (most recent call last):
  File "<string>", line 1, in <module>
  File "/usr/local/lib/python3.8/site-packages/sapporo/ro_create.py", line 152, in generate_ro_create
    add_workflow_attachment(create, run_dir, run_request, yevis_metadata)
  File "/usr/local/lib/python3.8/site-packages/sapporo/ro_create.py", line 567, in add_workflow_attachment
    if "script" in magic.from_file(source):
  File "/usr/local/lib/python3.8/site-packages/magic/_init__.py", line 179, in from_file
    return m.from_file(filename)
  File "/usr/local/lib/python3.8/site-packages/magic/_init__.py", line 112, in from_file
    with _real_open(filename):
FileNotFoundError: [Errno 2] No such file or directory: '/home/ivis/nii-dg/sapporo_example/run/3d/3dde09f3-a4b9-4ed8-ab0f-770c8775f73d/exe/ERROR34597_2-small.fq.gz'
/app/sapporo/run.sh: line 150: {}: command not found
ool/pathmapper.py", line 70, in _init_
    self.setup(dedup(referenced_files), basedir)
  File "/usr/local/lib/python3.10/site-packages/cwltool/pathmapper.py", line 173, in setup
    self.visit(
  File "/usr/local/lib/python3.10/site-packages/cwltool/pathmapper.py", line 133, in visit
    with SourceLine(
  File "schema_salad/sourceline.py", line 256, in _exit_
schema_salad.exceptions.ValidationException: workflow_params.json:1:14: [Errno 2] No such file or directory:
/home/ivis/nii-dg/sapporo_example/run/3d/3dde09f3-a4b9-4ed8-ab0f-770c8775f73d/exe/ERROR34597_1-small.fq.gz'
ERROR [step trimming] Cannot make job: workflow_params.json:1:14: [Errno 2] No such file or directory:
/home/ivis/nii-dg/sapporo_example/run/3d/3dde09f3-a4b9-4ed8-ab0f-770c8775f73d/exe/ERROR34597_1-small.fq.gz'
INFO [workflow ] completed permanentFail
WARNING Final process status is permanentFail
```

6. 研究リポジトリに同期する

このタスクの実行結果を研究リポジトリに同期します。

ファイルが見つからない
というエラーが出ている

検証実施の様子（問題点を解決して再検証）

実験の再現性を検証する

再検証する

検証の結果、OKが出る
(再現性が検証できた)

base_validate_repeatability_duar... demo/demo202305 - GIN-Fork x General x NII-DG/DG-demo202305-exp2 x +

_validate_repeatability_during_exp Last Checkpoint: 1分前 (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)

5. 検証結果を確認する

[7]: from WORKFLOWS.util.tap_validation import tap_validation
tap_validation.get_validation_results()
再現性が検証されました。次にお進みください。

6. 研究リポジトリに同期する

このタスクの実行結果を研究リポジトリに同期します。
検証結果を同期するか破棄するかは、「5.1. 検証結果を研究リポジトリに同期するか選択する」で選択できます。

6.1. 検証結果を研究リポジトリに同期するか破棄するかを選択する

In []: import panel as pn
pn.extension()
column = pn.Column()
def save_selection_result(event):
 global need_sync
 done_button.button_type = "success"
 done_button.name = "選択完了しました。次の処理にお進みください。"
 need_sync = True if select.value == 1 else False
select = pn.widgets.Select(name='検証結果を同期するか破棄するかを選択した後、完了ボタンをクリックしてください。', options=['同期', '破棄'])
done_button = pn.widgets.Button(name='選択を完了する', button_type='primary')
done_button.on_click(save_selection_result)
column.append(select)
column.append(done_button)
column

6.2. 研究リポジトリに同期する

研究リポジトリにこのタスクの実行結果を同期します。
「5.1. 検証結果を研究リポジトリに同期するか破棄するかを選択する」で同期せずに破棄するを選択した場合は、検証結果は同期されずこのファイルの実行結果のみが同期されます。

7. 研究フロートップページに遷移する

3. 今後の進め方

取り組み方針

1. 検証機能の拡充

- DMP との連携の強化
- FAIR 原則^[1]に基づくデータ管理状況の検証

2. リサーチフローのブラッシュアップ

- データガバナンス機能の機能評価試験版サービス
利用者からのフィードバックへの対応など

3. 他基盤との連携実施

4. 研究機関側に必要な機能の検討

- 研究データ管理・公開ポリシーへの対応等

[1] <https://force11.org/info/the-fair-data-principles/>

データガバナンス機能の機能評価試験版サービスの利用案内

GakuNin RDM におけるデータガバナンス機能の機能評価試験版サービスの提供を予定しております。

先行ユーザーからのフィードバックを受けてデータガバナンス機能の改善を実施し、実証実験レベルへのブラッシュアップを計画しております。

データガバナンス機能の機能評価試験版サービスの提供準備が完了次第、GakuNin RDM のサポートポータル (<https://support.rdm.nii.ac.jp/>) の「お知らせ」にてアナウンスいたします。

- 提供予定期間 : 2023/6/19～2024/3/31
- 問い合わせ先 : データガバナンス機能サポート
dg_support(at)nii.ac.jp
- 問い合わせ時 : 氏名、所属、連絡用メールアドレス、
に必要な情報 利用希望者リスト、参加希望理由