

# RDEデータセットテンプレートの開発を 始める前に

リリース 1.0.0 (2025.04.21)

国立研究開発法人 物質・材料研究機構

Copyright © National Institute for Materials Science. All rights Reserved.

# 目次

---

|                       |    |
|-----------------------|----|
| 1. 本書について             | 3  |
| 2. RDEとは              | 3  |
| 2.1 RDE               | 3  |
| 2.2 データセット・データ        | 4  |
| 2.3 データ構造化            | 4  |
| 3. データ登録方法            | 5  |
| 3.1 RDEアプリケーション       | 5  |
| 3.2 一括登録など            | 5  |
| 4. RDE使用の準備手順         | 7  |
| 4.1 研究チームの作成          | 7  |
| 4.2 データセットテンプレートの選択   | 7  |
| 4.3 データセットの開設         | 7  |
| 5. データセットテンプレートの開発    | 9  |
| 5.1 データセットテンプレートの構成要素 | 9  |
| 5.2 こんなとき何を編集する？      | 11 |
| 5.3 構造化処理             | 12 |
| 6. 次のステップ             | 15 |
| 7. 資料・ツール             | 16 |
| 8. 変更履歴               | 17 |

## 1. 本書について

本書は、これからRDEデータセットテンプレートの開発を始める人に、最初に読んでいただくことを目的とした資料です。

**RDE**って何？ データセット？ テンプレート？と聞いたことはあるが分からないRDEの用語が次々と出てきますが、それらの用語をざっくりと紹介しつつRDEの特徴であるデータセットテンプレートについて説明を試んでいます。本書ですべてを詳細に説明しようとはしていません。他の更に詳細な技術的資料を読む前の準備運動だと思ってください。

## 2. RDEとは

### 2.1 RDE

RDE (Research Data Express) は、物質・材料についての研究データをオンラインで迅速に登録するためにNIMSが開発したシステムです。測定データなどを登録すると自動的にデータ駆動型のマテリアル研究に適した形に構造化してクラウドに蓄積します。これによりユーザーや研究グループ内での再利用や他の研究グループとのデータの共有が容易となり、マテリアル研究開発のDX化を支援します。Figure1 にRDEの概要を示します。

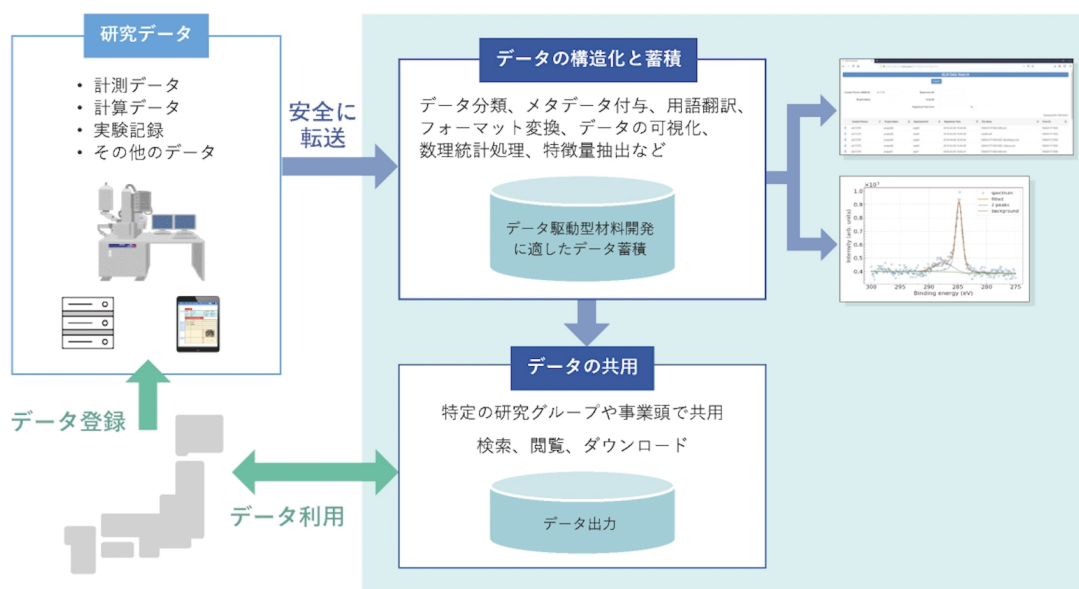


Figure1 RDEの概要

RDEの概要についてはホームページを参照してください。

## 2.2 データセット・データ

RDEは実験装置から出力されたデータファイルや計算結果を登録するためのデータベースです。そのためこれらのデータファイル(データと呼ぶことのほうが多い)を入れる箱を用意する必要があります。RDEではこの箱を **データセット** と呼んでいます。データセットの中には複数のデータを保存することができます。データセットは同じ構造 (測定データなど、メタデータ、送り状、サムネイル画像、**構造化データ**など) を持ったデータの入れ物です。一つのデータが一つの試料に相当するのが標準的な使い方です。

## 2.3 データ構造化

RDEの特徴として、**データ構造化** が挙げられます。データ構造化とは、データ分類、メタデータ付与、用語翻訳、フォーマット変換、数値データの可視化や数値統計、特徴量抽出などの処理のことで、マテリアルズ・インフォマティクス (MI) によるデータ駆動型の材料研究に適した形でデータを蓄積するために必要な処理です。RDEではデータ登録時に **構造化処理** を実行する機能が用意されています。また構造化処理はデータセットごとに異なるものを適用することができます。これを実現するためのしくみがデータセットテンプレートです。Figure2 は、データ構造化と蓄積処理フローのイメージです。

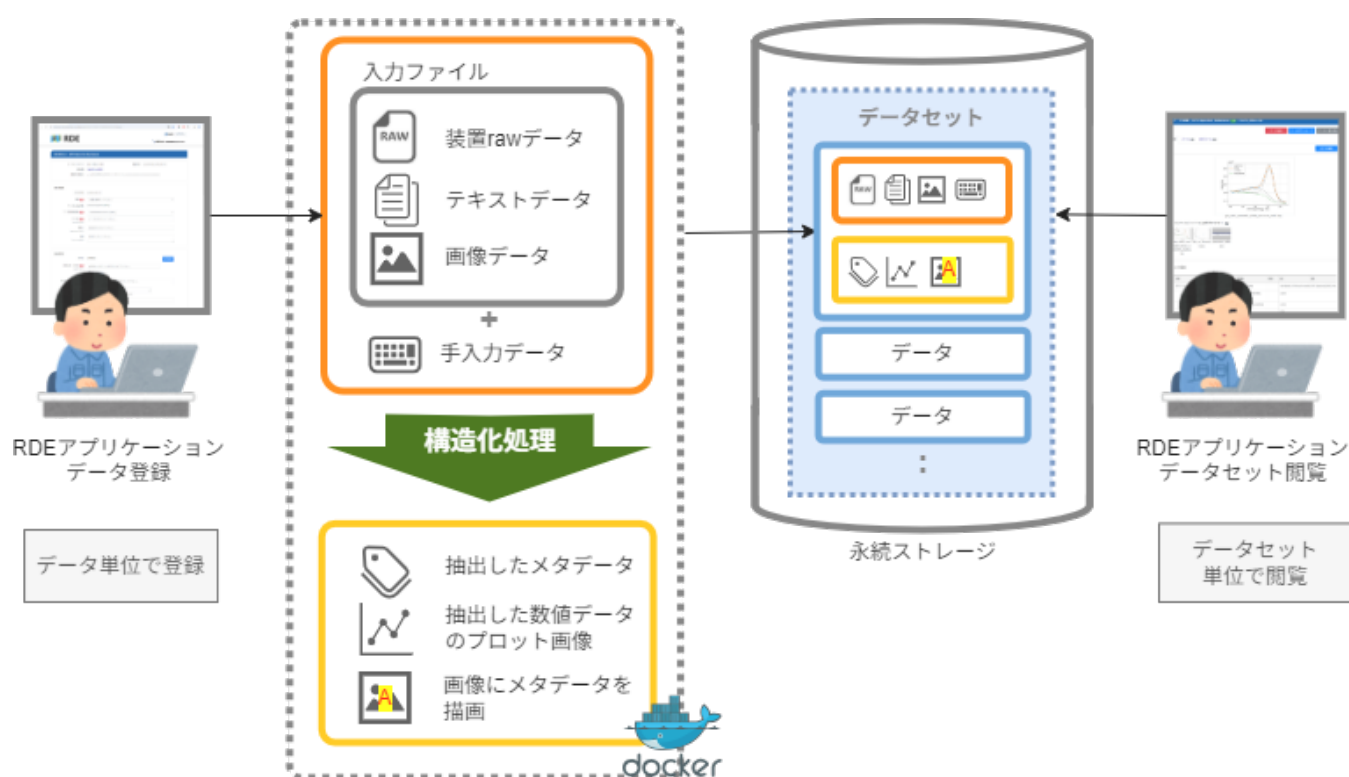


Figure2 データ構造化と蓄積の流れ

## 3. データ登録方法

### 3.1 RDEアプリケーション

RDEへのデータの登録や閲覧は、**RDEアプリケーション**で行います。データ登録には**RDEデータ登録アプリケーション**、データセット・データの閲覧などには**RDEデータセット閲覧アプリケーション**をそれぞれ利用します。いずれのRDEアプリケーションもChromeなどのWebブラウザ上で動作するWebアプリケーションです。使用にはDICEサービスでアカウントを作成(サインアップ)し、利用申請をする必要があります。

登録、閲覧画面はデータセットごとにカスタマイズすることができます。それを実現するしくみが **データセットテンプレート** です。構造化処理で抽出したメタデータやグラフなどの画像を指定できます。Figure3 に表示例を示します。

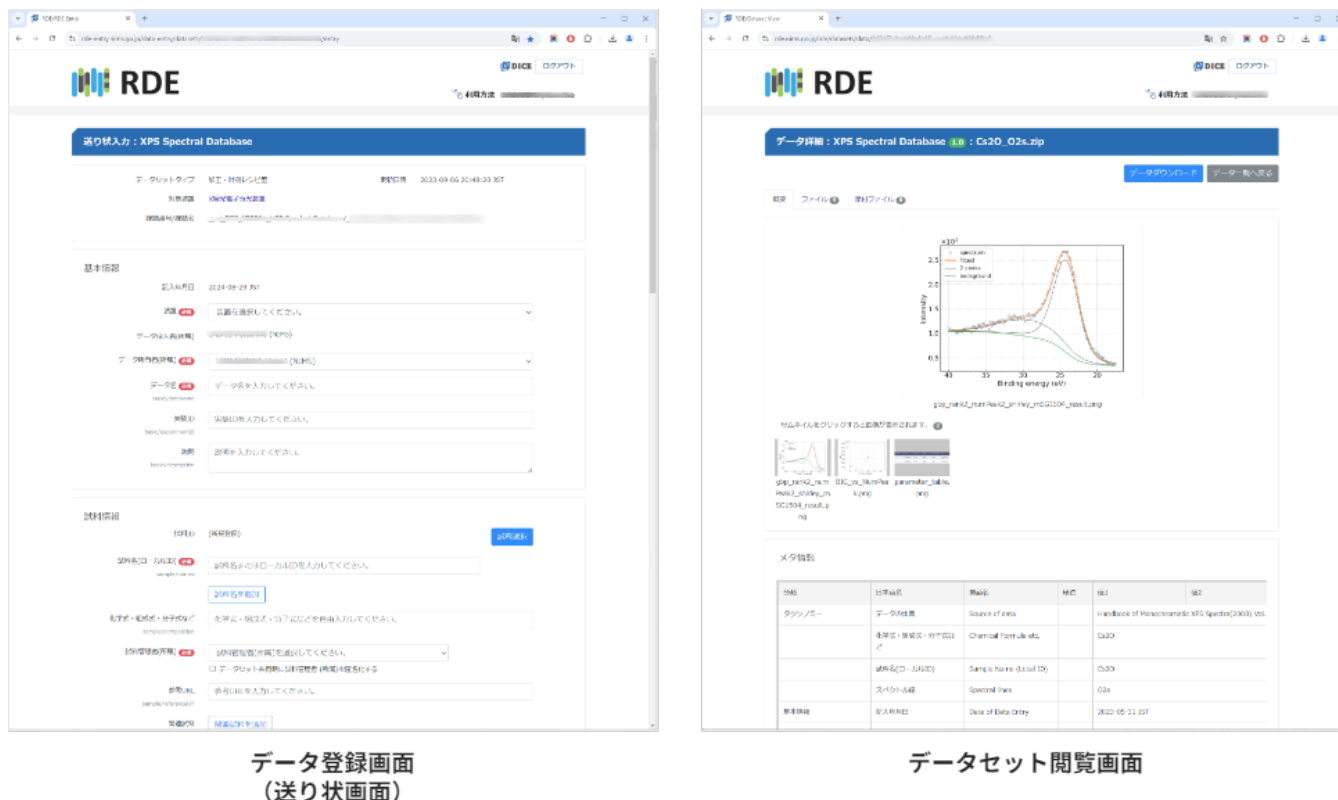


Figure3 RDEアプリケーション

RDEの概要についてはホームページを参照してください。



#### データ登録画面の名称

RDEアプリケーションのデータ登録画面は、送り状画面、入力フォームと表現する場合があります。

### 3.2 一括登録など

RDEアプリケーションでは1つの送り状につき1つのデータの登録が基本です。ただし、データセットテンプレートには、まとめて複数のデータを登録したい場合はエクセルインボイス機能を使うと1回の送り状で複数のデータを登録できる機能を持ったものが用意されています。

データセットテンプレートには、あらかじめ構造化したファイルを登録する、XRD、電子顕微鏡の測定データを登録するなどあらかじめ用意されたものもあります。RDEはデータセットテンプレートの種類を増やすことで多くの測定データなどを登録できるという拡張性があります。

### 3.2.1 エクセルインボイス

---

エクセルインボイスは、入力情報を記入した既定のエクセルファイルと複数のデータファイルをまとめてzip形式のファイルをまとめて登録することで、複数のデータとして登録するための機能です。

## 4. RDE使用の準備手順

---

RDEを使用するためには、研究チームの作成後、データセットテンプレートを選択して、データセットを開設する必要があります。いずれも申請が必要です。

### 4.1 研究チームの作成

---

データセットは研究チームの下に開設されます。研究チームはデータセットを利用するためのまとまりです。構成メンバにはそれぞれ役割を割り当てることができます。データセット管理者などの役割があります。研究チームの作成は申請が必要です。

### 4.2 データセットテンプレートの選択

---

データセットテンプレートはその名の通りデータセットの雛形です。RDEには複数のデータセットテンプレートが用意されていますので、蓄積したいデータに合ったものを選択します。要求に沿ったものがない場合は改変、または、新規作成します。データセットテンプレートでは、送り状画面項目、構造化処理、蓄積するメタデータ項目などを設定できます。次章で詳しく説明します。

#### 4.2.1 データセットテンプレートの登録

---

データセットテンプレートを用いてデータセットを開設するためには、データセットテンプレートを事前に登録する必要があります。登録するためにはデータセットテンプレートの作成が必要となります。

### 4.3 データセットの開設

---

1つのデータセットテンプレートから複数のデータセットを開設することができます。実験の単位などに応じてデータセットを開設し使い分けることができます。Figure4 にデータセットテンプレートの選択のイメージ図を示します。

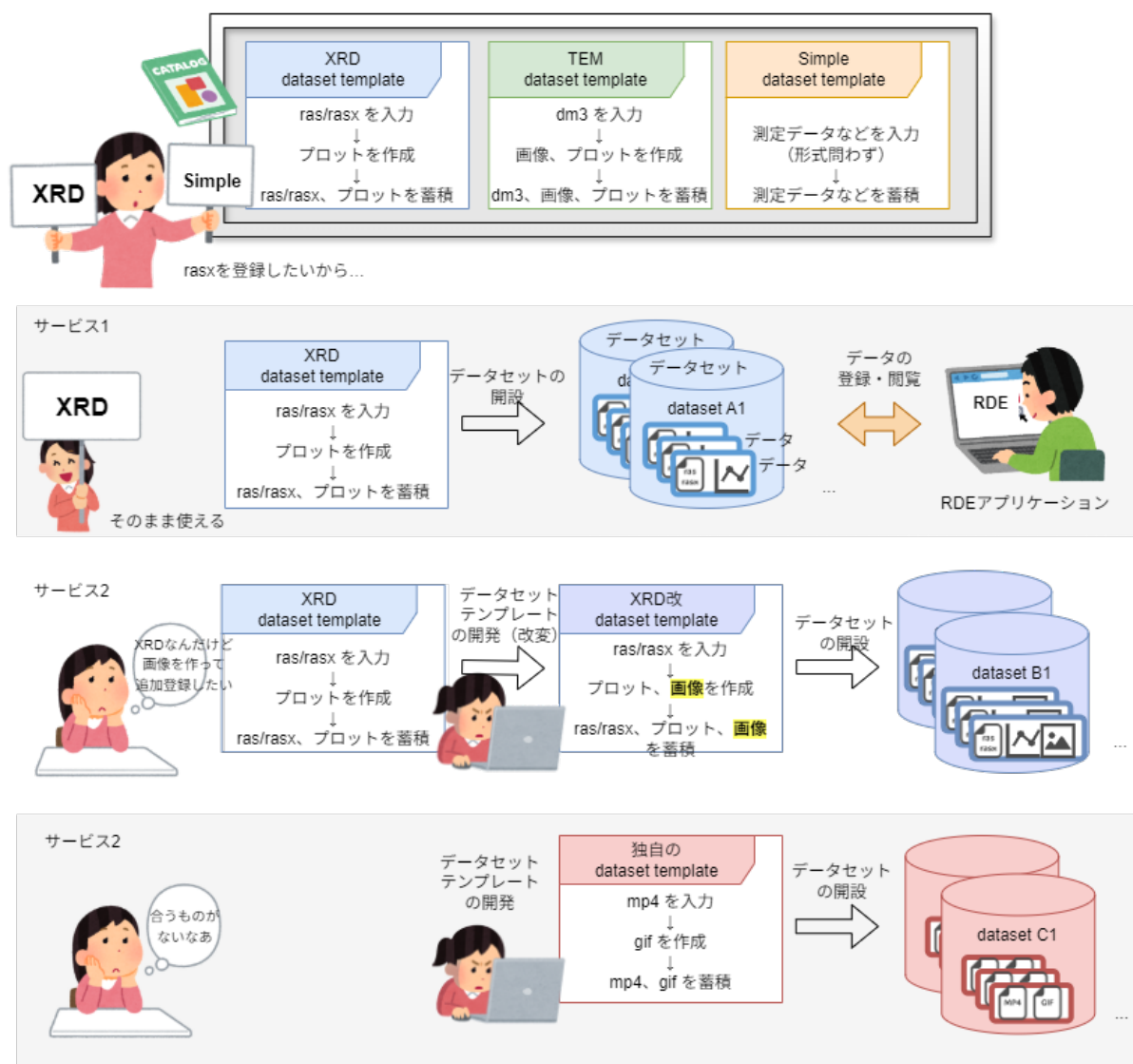


Figure4 データセットテンプレートの選択



## 5. データセットテンプレートの開発

RDEでは開発者向けの資料やツールを複数用意しています。ここでは、それらに触れる前の基礎知識の習得とイメージの共有をします。

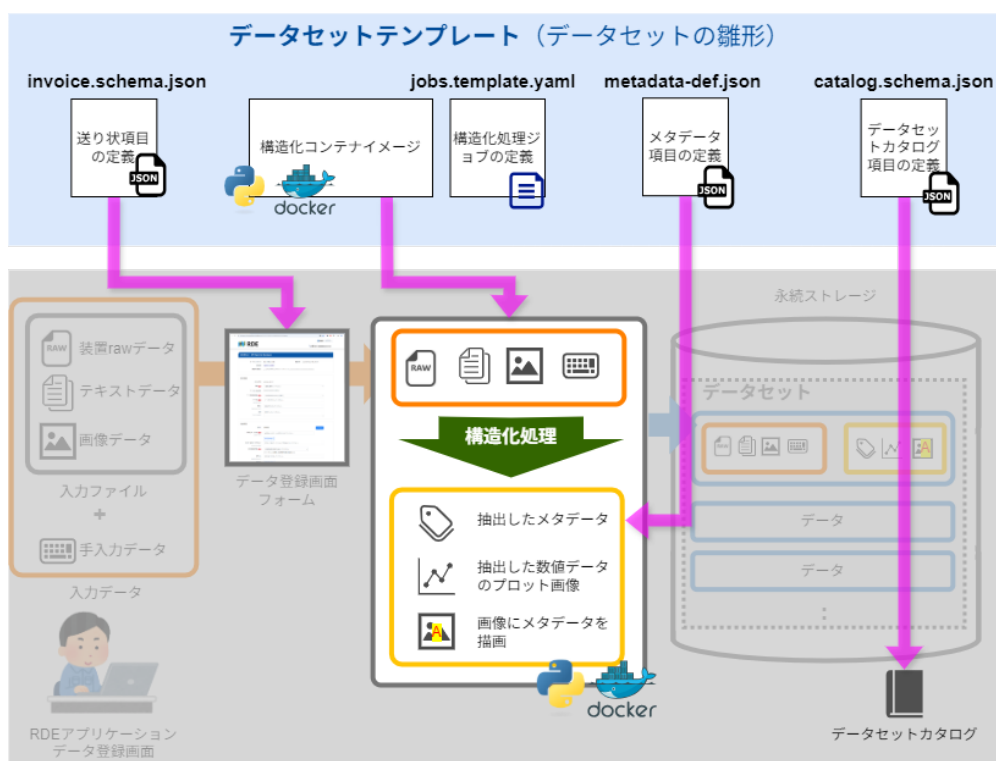
### 5.1 データセットテンプレートの構成要素

データセットテンプレートは、いくつかの定義ファイル（以下「テンプレートファイル」と呼びます）と、構造化処理用Docker Imageで構成されています。データセットテンプレート開発者はこれらのテンプレートファイルおよびDocker Imageを作成します。以下の表にデータセットテンプレートを構成する定義ファイルなどを示します。

| ファイル名前                     | 内容  |
|----------------------------|---|
| (構造化処理用 Docker Image)      | 構造化コンテナイメージ<br><code>python</code> で記述された構造化処理をDockerコンテナ化したもの            |
| invoice.schema.json        | 送り状入力画面の設定ファイル<br>送り状画面の入力項目を、表示名、表示順、必須項目などを含め指定できます。手入力項目を定義するときに利用します。 |
| metadata-def.json          | 抽出するメタデータの定義ファイル。構造化処理で抽出したメタデータの定義。                                      |
| catalog.schema.json        | データカタログ定義ファイル   |
| batch.yaml                 | データ構造化定義ファイル  |
| jobs.template.yaml         | データ構造化ジョブ定義ファイル   |
| jobs-divided.template.yaml | 分割後の個々のデータに適用するデータ構造化ジョブ定義ファイル  |
| tasksupport/*              | タスク補助ファイル<br>tasksupportディレクトリに格納された、構造化コンテナイメージが実行時に使用するファイル群            |

上記のうち、データセットテンプレート開発者が用意するものは、invoice.schema.json、metadata-def.json、catalog.schema.json、tasksupportファイル、構造化処理プログラムです。その他の定義ファイルやDockerコンテナイメージは、データセットテンプレート登録担当者が用意します。ただし、既存のテンプレートを流用する場合は一部の定義ファイルのみであったり、またはコンテナイメージの作成をお願いすることもあります。

Figure5 は、データセットとデータセットテンプレートのイメージ図です。



## 5.2 こんなとき何を編集する？

データセットテンプレートを作成するとき、まずは使えそうなものがないか調べてください。これから作りたいものに近いテンプレートを選んで変更して使うことをお勧めします。

予め用意されたデータセットテンプレートを改変して使用する場合（3章 [データセットテンプレートの選択](#)参照）、改変したい内容によって編集するファイルが異なります。主に改変するのは `invoice.schema.json`、`metadata-def.json`、構造化コンテナイメージの3つです。なお、構造化コンテナイメージを変更する場合は、DockerとPythonについてある程度の知識が必要です。以下に改変例とそれによって編集すべきテンプレートファイルを示します。

| 改変内容  | invoice<br>.schema<br>.json | metadata-def<br>.json | 構造化<br>コンテナ<br>イメージ |
|---|-----------------------------|-----------------------|---------------------|
| データ登録画面の入力項目の項目名を変更したい  | ✓                           |                       |                     |
| データ登録画面の入力項目を増やしたい  | ✓                           |                       |                     |
| データ登録画面の入力項目を増やして、メタデータとして登録したい                               | ✓                           | ✓                     |                     |
| 入力ファイルの形式を制限したい   |                             |                       | ✓                   |
| 測定データなどから作成するプロットを増やしたい                                       |                             |                       | ✓                   |
| 測定データなどから作成するプロットのタイトルをデータ登録画面で指定したい                          | ✓                           |                       | ✓                   |
| データセット閲覧画面で表示するデータごとのメイン画像を変更したい                              |                             |                       | ✓                   |
| 測定データなどから抽出するメタデータを増やしたい                                      |                             | ✓                     | ✓                   |
| メタデータの抽出方法を変更したい  |                             |                       | ✓                   |
| データ登録画面にデータ種別選択項目を追加して、入力値によってメタデータの抽出方法とプロットデータ、プロット書式を変更したい | ✓                           |                       | ✓                   |

## 5.3 構造化処理

構造化処理が行なっていることを具体的に説明します。

構造化処理では、構造化作業用ストレージに格納されたデータを使用します。構造化作業用ストレージには ジョブディレクトリ と呼ばれる定型の構造をもつディレクトリがあり、RDEアプリケーションからデータの登録を行うと、まずジョブディレクトリに入力ファイルや手入力データが格納されます。次に構造化処理を行うpythonファイルが起動され、ジョブディレクトリに格納されたデータを用いて構造化ファイルを作成します。構造化ファイルはジョブディレクトリの決められた場所に格納します。

RDEの構造化処理を大胆に要約すれば、RDEの定義に従ったファイルをジョブディレクトリに作成することです。RDEのデータ登録では、ジョブディレクトリに配置されたファイルをシステムに登録します。

Figure6 は、データセットと各種ストレージの構成図です。

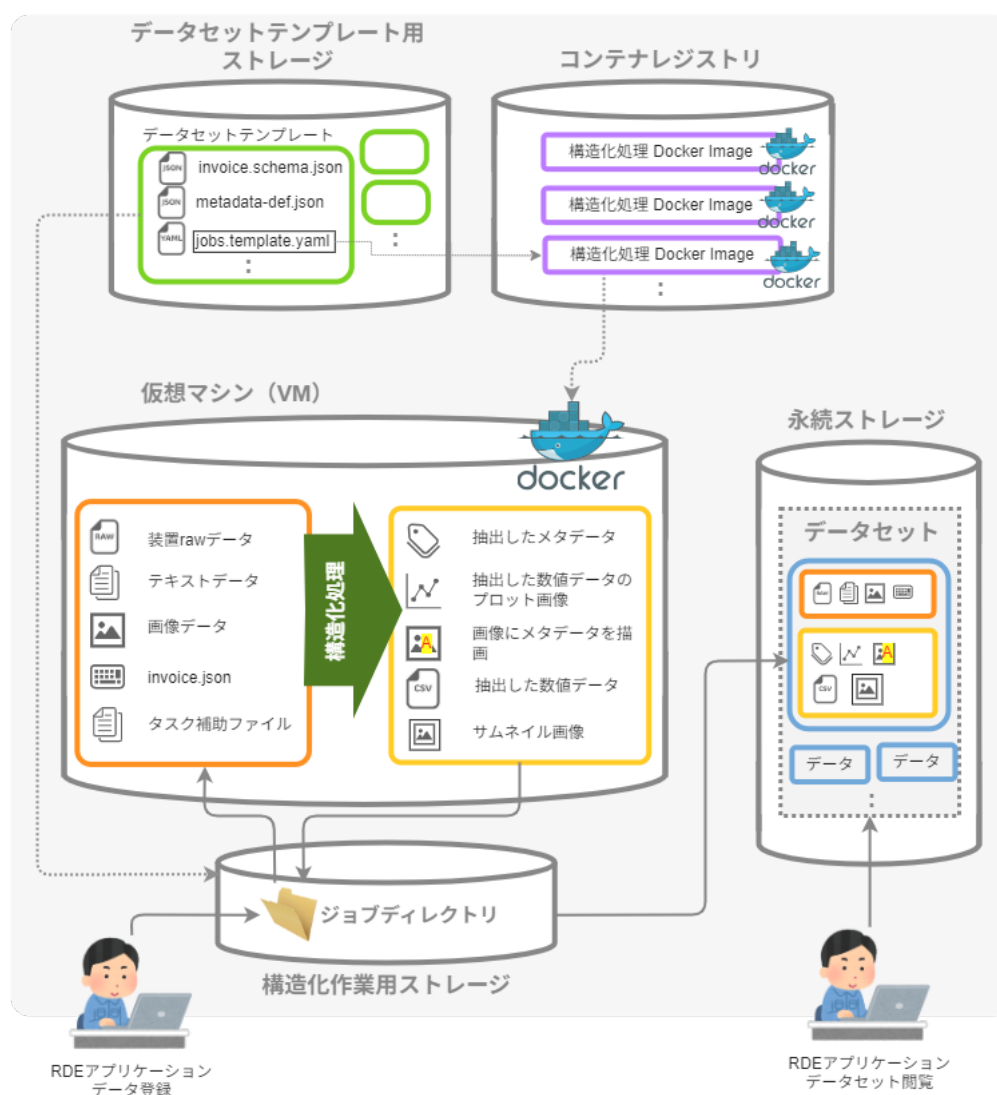


Figure6 データセットと各種ストレージの構成図

### 5.3.1 ジョブディレクトリ

ジョブディレクトリは定型の構造を持っています。以下にディレクトリ構造と格納されるファイルを示します。

|                   |                               |
|-------------------|-------------------------------|
| /                 |                               |
| ├ structured      | データ構造化用jobs.yamlファイル          |
| ├ shipyardyaml    | ジョブ設定ファイル                     |
| ├ data            |                               |
| │ └ invoice       | 送り状ファイル (invoice.json)        |
| │ └ invoice_patch | 送り状ファイル修正パッチ                  |
| │ └ inputdata     | 入力されたrawファイル                  |
| │ └ logs          | タスクで生成されるログファイル               |
| │ └ temp          | タスクで生成される中間ファイル               |
| │ └ meta          | 蓄積するメタデータファイル (metadata.json) |
| │ └ thumbnail     | サムネイル画像                       |
| │ └ main_image    | メイン画像                         |
| │ └ other_image   | メイン画像以外の画像ファイル                |
| │ └ attachment    | 添付ファイルとして入力されたファイル            |
| │ └ nonshared_raw | 非共有データ                        |
| │ └ raw           | 共有データ                         |
| └ tasksupport     | タスク補助ファイル                     |

#### 構造化処理での役割

一部のディレクトリは、構造化処理前にRDEシステムによって作成され、RDEアプリケーションから投入されたデータ、さらに、データセットテンプレートに登録されているタスク補助ファイルが格納されています。構造化処理ではそれらのファイルを読み込み、可読化、可視化、メタデータの抽出などを行い、作成したファイルを決められたディレクトリに格納します。さらに、一部のファイルはファイル名も決められています。Figure7に構造化処理のジョブディレクトリの例を示します。図中の赤字のディレクトリとファイルは、構造化処理によって作成されたものです。

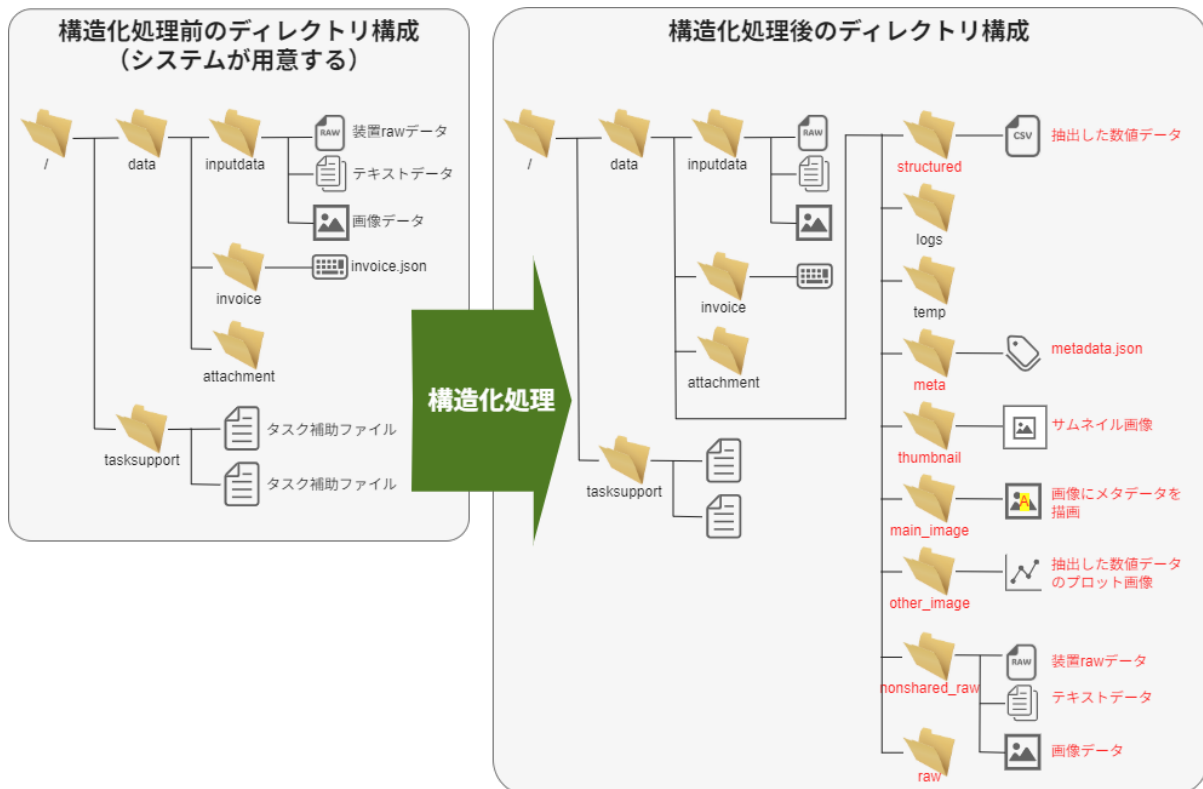


Figure7 構造化処理のジョブディレクトリ

## 5.3.2 ジョブディレクトリとRDEアプリケーションでの表示

特定のジョブディレクトリに格納したファイルは、RDEアプリケーションでのFigure8 のように表示されます。

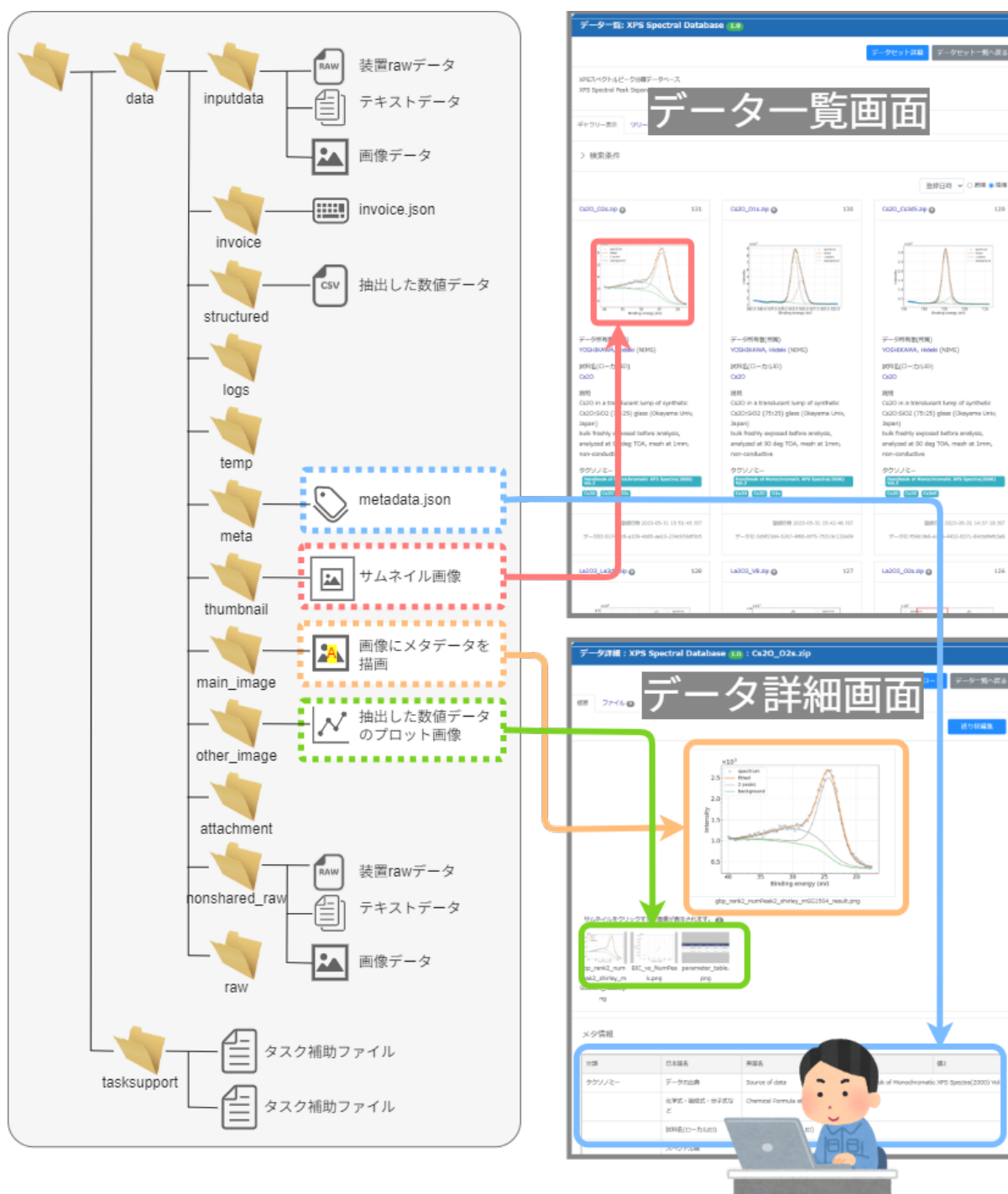


Figure8 ジョブディレクトリとRDEアプリケーションでの表示

## 6. 次のステップ

---

理解するには手を動かしてみるのが一番。そのような人は以下の資料で実際にデータセットテンプレートを作成し動かしてみてください。ローカルPCでお試しできます。

### **RDE構造化処理プログラム開発 手順書**

RDEToolKit(invoiceモード)を利用したシンプルなRDEデータ構造化処理プログラムハンズオン

データセットテンプレートの定義ファイルについてもっと詳しく知りたくなったら、以下の資料を参考にしてください。

### **RDEデータ構造化とデータセットテンプレート解説**

## 7. 資料・ツール

データセットテンプレートの開発に役立つ資料やツールを用意しています。

準備中、公開準備中のものが多くてごめんなさい。順次公開予定です。

| 名前                       | リンク   | 内容   |
|--------------------------|---|--|
| RDEポータルサイト               | <a href="#">URL</a>                           | DICEのRDE利用者向けサイトです。利用者マニュアル、利用手続きに関する説明や申し込みフォームが用意されています。   |
| RDEデータセットテンプレートの開発を始める前に | 準備中   | 本書。RDEデータセットテンプレートの開発を始める前に読んでおいていただきたい用語などの説明資料です。  |
| RDE構造化処理プログラム開発 手順書      | 準備中   | RDEToolKit(invoiceモード)を利用したシンプルなRDEデータ構造化処理プログラムハンズオン。RDE構造化プログラムの作成を実際にプログラムを作成しながら習得出来ます。                          |
| RDEデータ構造化とデータセットテンプレート解説 | 準備中   | データセットテンプレートの定義ファイルの詳細情報をまとめた資料です。   |
| RDEToolKit               | <a href="#">ドキュメント</a><br><a href="#">pip</a> | RDE構造化プログラムのワークフローを作成するための基本的なPythonパッケージです。RDEToolKitの各種モジュールを使うことで、RDEへの研究・実験データの登録処理を簡単に構築できます。                   |
| (準備中)メタ情報カスタマイズ手順書       | 準備中   | データセットテンプレート改変におけるメタ情報のカスタマイズ方法について書かれています。  |
| (準備中)エクセルインボイス説明資料       | 準備中   | データ一括登録機能であるエクセルインボイスの機能や使い方を知ることができます。  |
| RDE_preview_tool         | 準備中   | 構造化処理結果（dataディレクトリ）をプレビューするためのHTMLファイルを生成するツールです。RDEシステムにデータセットテンプレートを登録する前に、RDEデータセット閲覧アプリでどのように表示されるかを確認することができます。 |
| RDE_template_tools       |   | エクセル様式ファイルからテンプレートファイルを生成するpythonプログラムと、VSCodeで表示確認するためのExtensionのセットです。   |



## 8. 変更履歴

---

| バージョン | 日付         | 変更者        | 変更内容 |
|-------|------------|------------|------|
| 1.0.0 | 2025.04.31 | H.Tosaka   | 初版   |
| 0.9.0 | 2025.04.18 | K.Sasajima | 初版   |