# MALEXA® x AWS

中外製薬株式会社

モダリティ基盤研究部 角崎、高萩、成島

2023年6月29日

MALEXA®

# アジェンダ

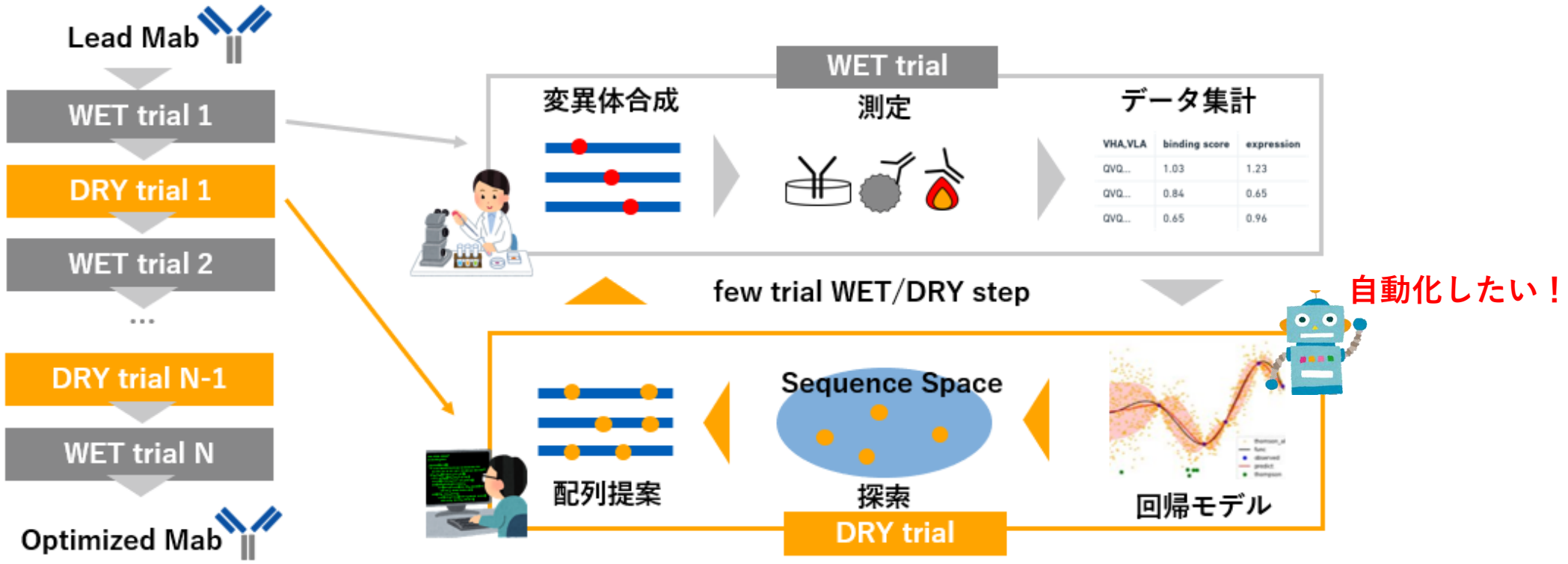
- MALEXA®とMLOps
- 実装への課題
- 選択した技術とシステムの詳細
- まとめ

MALEXA®

# MALEXA®-LO

- 中外製薬の研究所ではDRYとWETが協力して創薬を推進しています.

## ○ MALEXA®-LOのフロー図




**DRY研究員が実行していたパイプラインの部分的な自動化の検討を紹介する.**

MALEXA®

# MLOps

- MLOpsの構成要素とシステムの運用体制レベル.

## ○ ML systemの構成要素

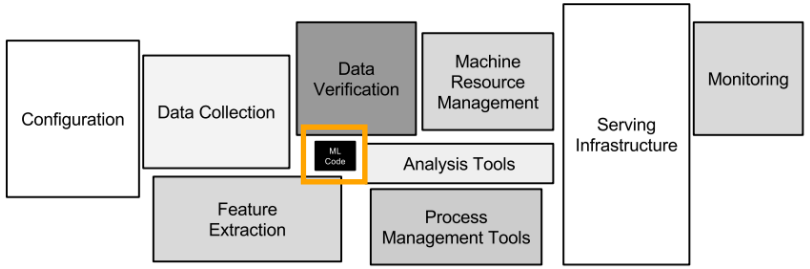



Figure 1: Only a small fraction of real-world ML systems is composed of the ML code, as shown by the small black box in the middle. The required surrounding infrastructure is vast and complex.

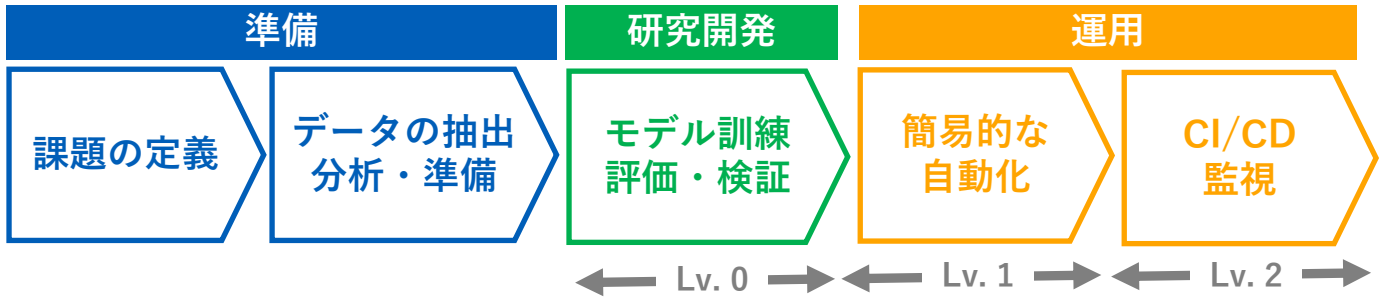
D.Sculley NIPS 2015

**ML codeはML systemの一部分でしかない.**

**データ・パラメータなどソフトウェア以外の管理が必要となる.**

**計算機の使い方も複雑になるため, インフラ構築も単純ではない.**

## ○ ML systemの達成度レベル




**MLOps Level**

**Level 0 : MLパイプラインの手動実行.**

**Level 1 : MLパイプラインの自動化・単純なシステム化**

**Level 2 : 監視・自動デプロイ等の高度な機能.**

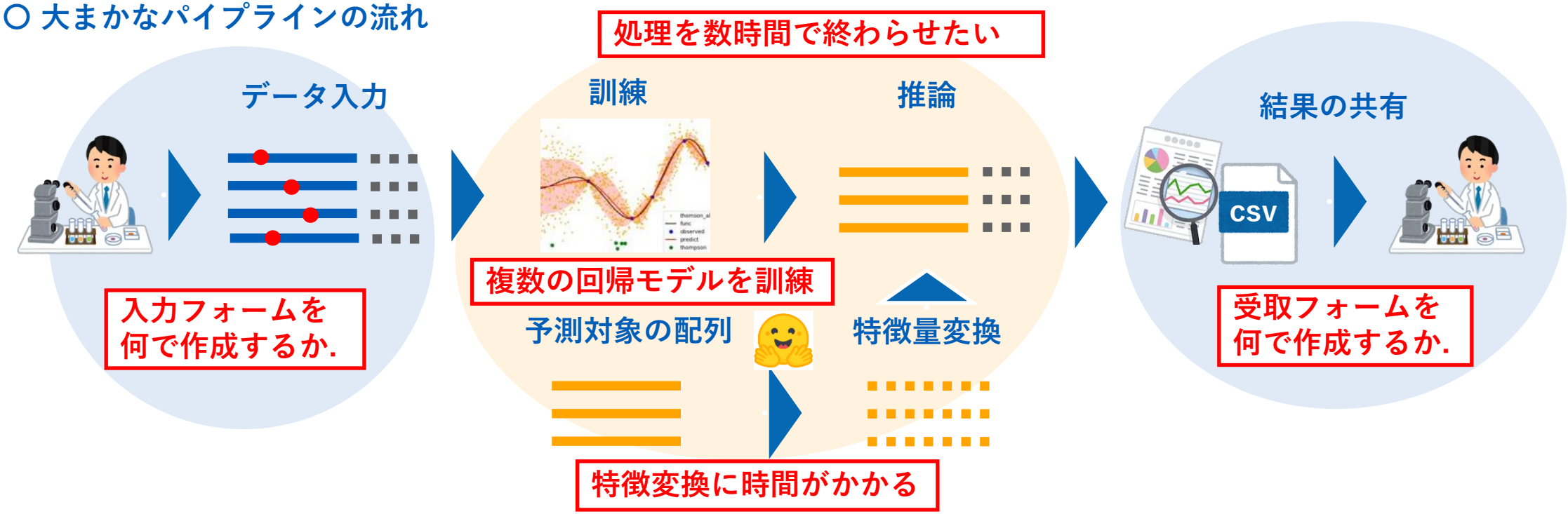
**MALEXA®のパイプラインにおいて, Level 1を達成したい**

MALEXA®

# MALEXA®-LO

- パイプラインを実装する際の懸念点.

○ 大まかなパイプラインの流れ




・入力/受取フォームを研究員の最低限の負荷で実装する必要がある.
・計算負荷の高い処理に対して, 計算リソースをよしなにスケーリングしたい.

**上記を最低限の努力でAWS上にパイプラインを実装する.**

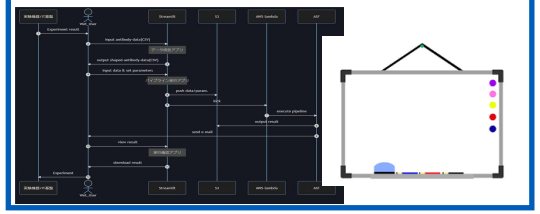
MALEXA®

# DRY系研究員の課題感

- 最低限の努力で実用に耐えるパイプライン作りを目指す際に重要な要素技術/取り組み.

## ○ 重要な要素技術

### 開発会議



・複数人での開発を行うために開催.
・機能追加や文章化による認識の共有ができたので, 開発効率Up.
・標準化によるアセット, リソースの共通化により研究生産性にも繋がる.

### GUI

Streamlit

・データ入力/受取のGUI作成に使用, Python-baseのフレームワーク.
・圧倒的に習得の難易度が低く, Pythonのモジュール群との連携も容易.

### ASF

AWS Step Functions

・パイプラインのロジック部分をASFで実装.
・任意の機能をコンポーネントとしてワークフロー形式で記述が可能.
・処理のスケーリング/並列実行を容易にやってくれる.
・AWSのサービス/インフラと密結合になる点が少しネック？

**最低限の努力でパイプラインを実装していきます.**

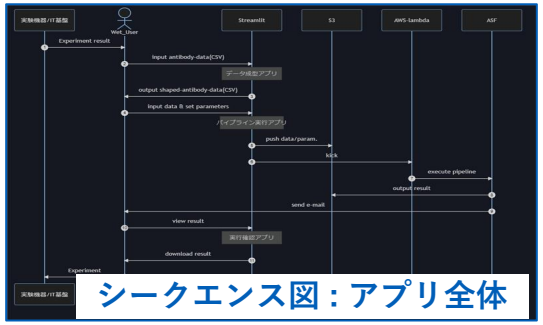
# 開発会議

- 本システムは複数人で開発しているので, 全体のイメージ共有が大事.
  - 作成物/アップデートをバージョン管理システム上で共有.
  - もくもく会の設定 ( 週１程度で集まる会議 )

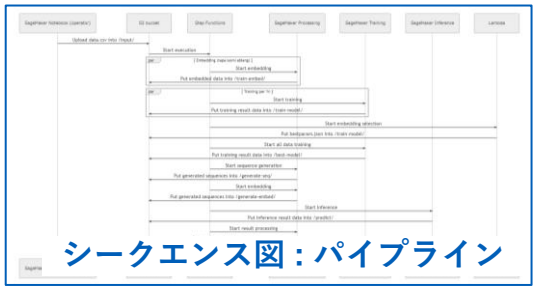
## ○ White-Boarding

↑AWSの皆様からご助言を頂き開発する図

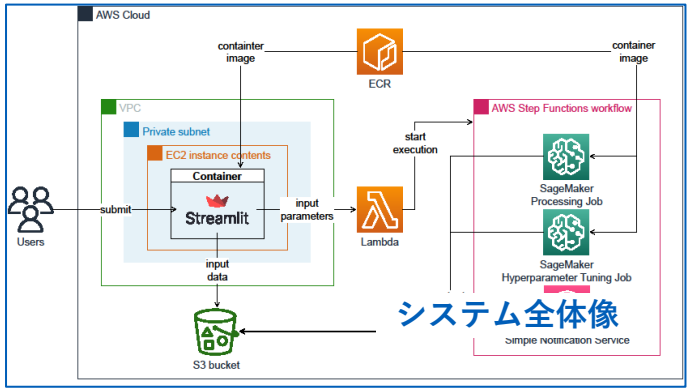
## ○ 作成物



シークエンス図 : アプリ全体



シークエンス図 : パイプライン




システム全体像

AWS リソース

- S3 bucket : `s3://[redacted]-dev-${AWS::AccountId}/`
  - inputs : `s3://[redacted]-dev-${AWS::AccountId}/inputs/`
    - antibody-data : `dataset_fmtdata_[redacted].csv`
  - dev : `s3://[redacted]-dev-${AWS::AccountId}/dev/`
    - train-embed step : `/train-embed/embeddata_{embet-method}.csv`
  - outputs : `s3://[redacted]-dev-${AWS::AccountId}/outputs/{time}/{step-name}`
- IAM roles:
  - dev:
    - `[redacted]-dev-SageMakerExecutionRole`

リソースの準備

**管理/開発/運用の面で役に立った. 複利で効く活動だと思う.**

MALEXA®

# 開発会議

- 最終的に作成するパイプラインのイメージは以下になる.
  - 実行する際のリソースの準備(中間データ/コンテナ環境)も大事.
  - 当然, 出戻りが発生するので, 共有資料をアップデートしながら開発をする.

## ○ 本システムの大まかな概要

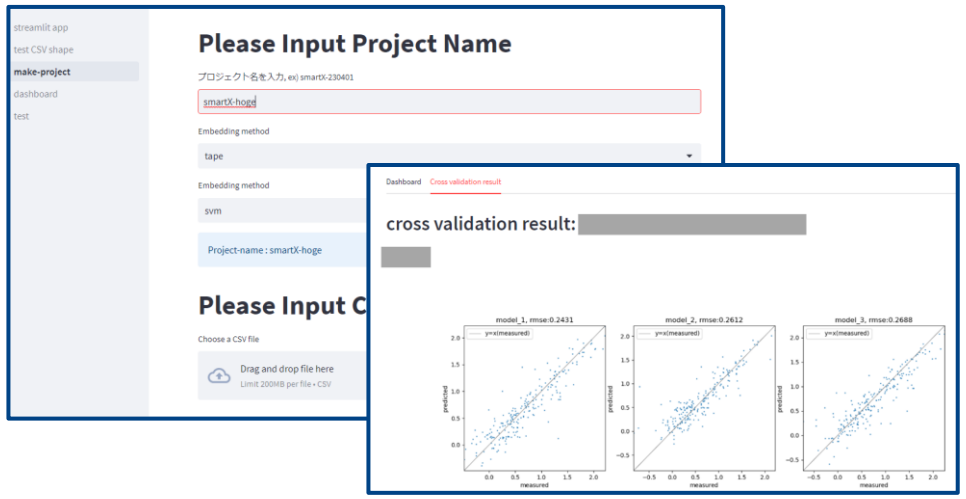

**あとは作るだけ！**

MALEXA®

# Streamlit

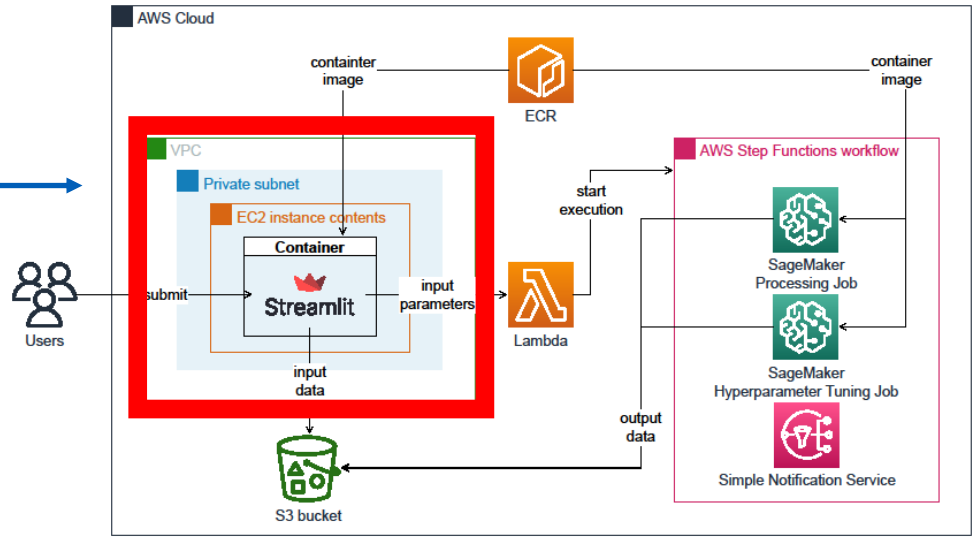
- Streamlitで「実験研究員」と「パイプライン」をつなぐGUIを作成した.
  - Pythonの知識と1日程度の時間があれば, 研究所の業務に使用できるGUIが作成可能です.
  - 敷居が低いので組織内のGUI作成の標準技術にすることが可能です.

### ○ ユーザーからの受付/受け取りフォーム例




・EC2上にStreamlitサーバーをたてる.
・Input Dataを受付時点でデータのチェック

### ○ システムの全体像



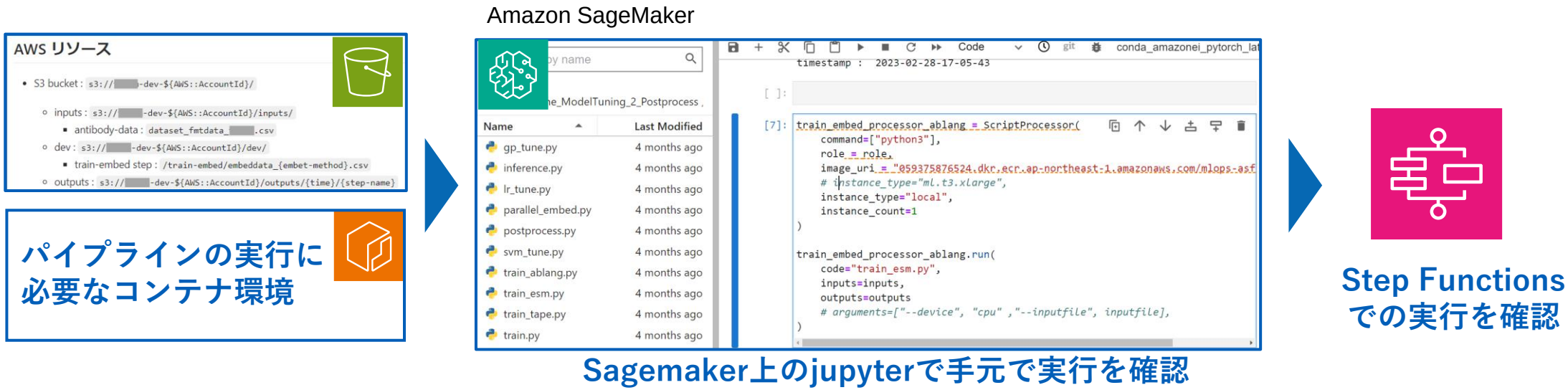

・AWSの他サービスと連携もしやすい

**最低限のGUIが内製可能になりました.**

# AWS Step Functions

- ASFは任意の機能をコンポーネントとしてワークフロー形式で記述が可能なサービスです.
  - 計算リソースのスケーリング. 各AWSのサービスとの連携が行いやすい.

- 今回の開発では, SageMakerのJupyter上 で ASF のコンポーネントの実装を行った.
  - Jupyter上で実行することで, 手元で実行の可否をすぐに判断できる.
  - 中間ファイル, コンテナなどは先に準備しておくと便利.




Sagemaker上のjupyterで手元で実行を確認

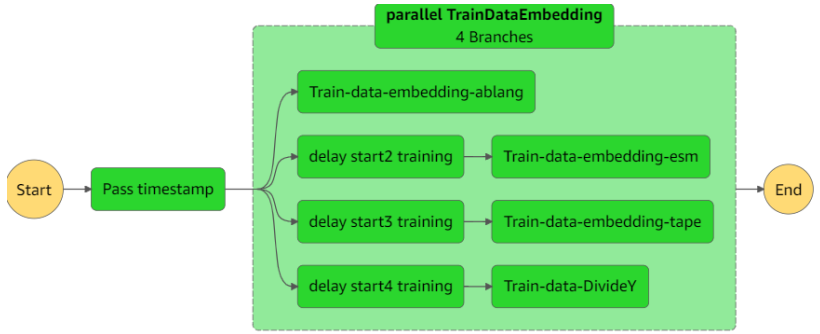
**出戻りが多くなる際に, 手元で実行できたのは便利でした.**

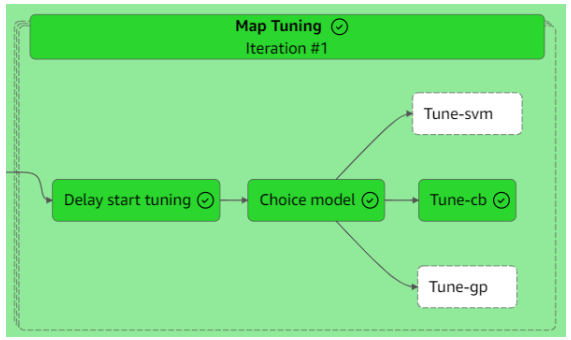
MALEXA®

# AWS Step Functions

- 処理ごとにインスタンスのスケーリングを行える.

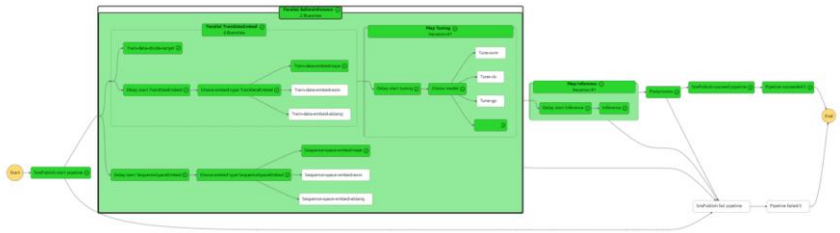
### ○ 各処理を良しなにスケーリング.




・各コンポーネントは使いまわす
・使用インスタンス数を指定可能
・計算結果はAWS上でログが残る.

### ○ インスタンス数を変えて実験 (1 instance vs 10 instances)



| | ステータス | 開始 | 終了時刻 | |
|---|---|---|---|---|
| 10 instances | | 2023年6月27日 11:14:39.374 (UTC+09:00) | 2023年6月27日 12:17:57.084 (UTC+09:00) | 63min |
| 1 instance | | 2023年6月27日 11:14:12.791 (UTC+09:00) | 2023年6月27日 18:24:35.493 (UTC+09:00) | 430min |

当然, 早くなった. ロジック上の限界はあるが, より広い範囲の配列探索も可能になる.

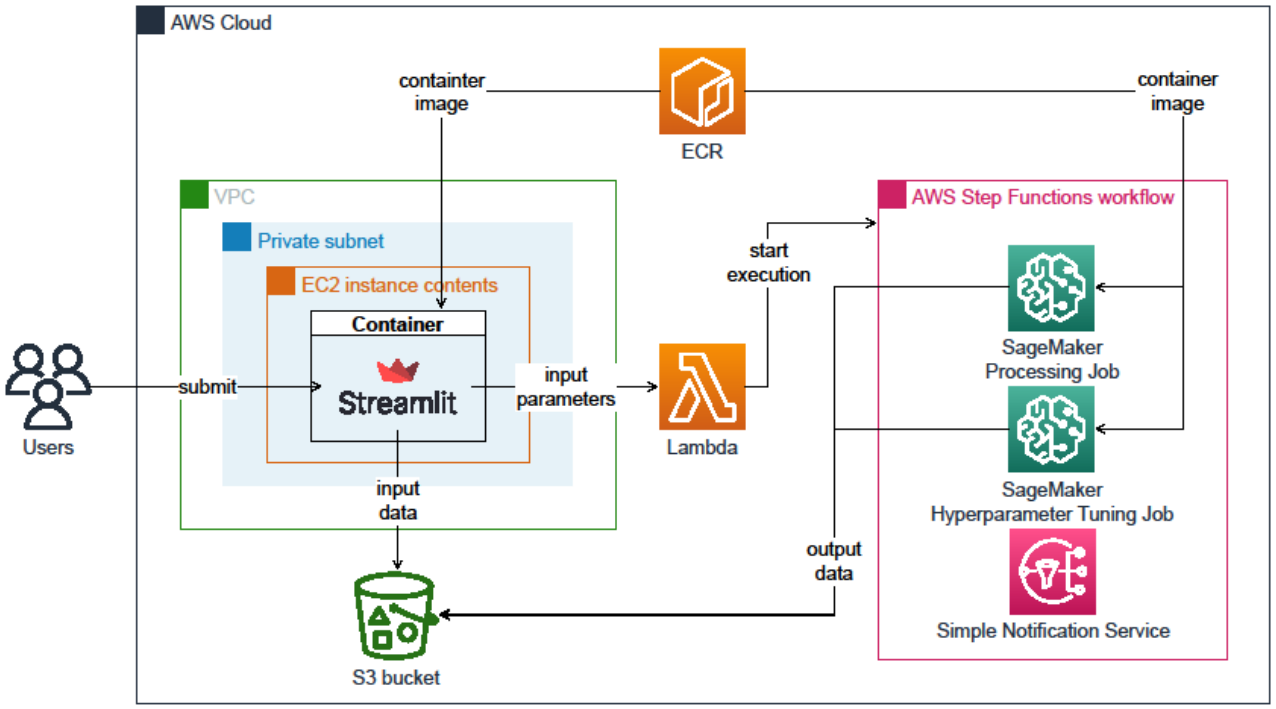
**アウトプットを迅速に研究員にフィードバックが可能となった.**

MALEXA®

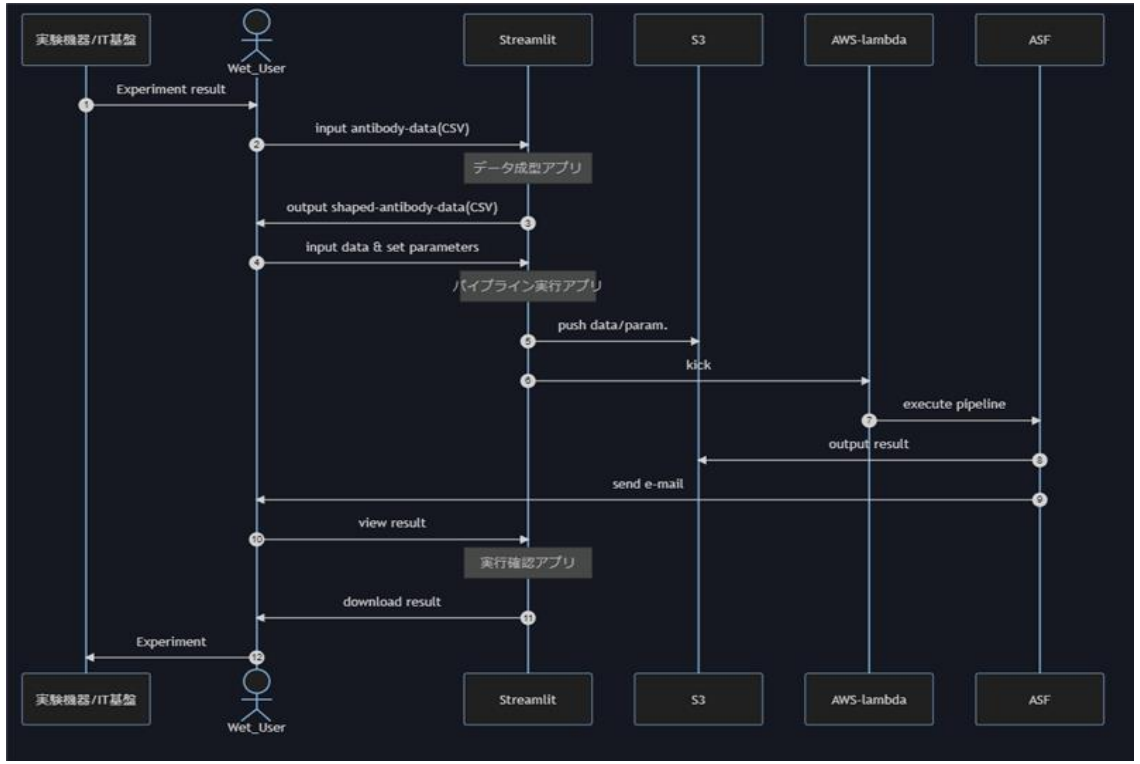
# パイプラインのまとめ

- 本システムのシステム概要について最後にまとめて説明します.

## ○ システムの構成図




## ○ システムのシークエンス図




**本システムと同じ仕組みを横展開し, 研究所のDX活用を推進していきます.**

# まとめ

- MALEXA®-LOを題材に必要最低限の努力で運用できる自動化パイプラインの体制を試作した.

- 今回のパイプライン実装で３つ重要なポイントがある.
  - Straemlit : 簡単にGUIが作成できる.
  - ASF　　: 計算リソースのスケーリングと処理のコンポーネント化.
  - 開発会議 : 全体の認識合わせ, リソースの準備で効率的な開発体制を目指す.

- 本パイプラインで良くなる(予定の)こと.
  - 実験のタイムラインに間に合う形で実行結果を取得できる.
  - DRY/WET研究員のリソースが削減することが可能になる.

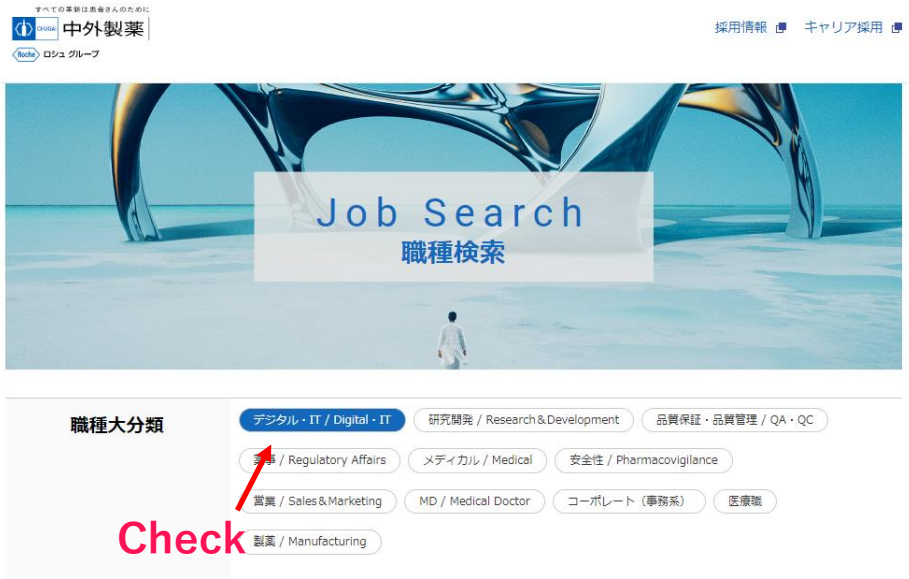
- 計算リソースのスケーリングをよしなにやってくれる.
  - 各コンポーネントを再利用することで, 計算機実験のサイクルを早くでき, 普段の研究開発にも活かせる.
  - AWSのサービスを迷いなく連携することができる.

- 今後の動き
  - スモールスケールのプロジェクトで, 実践適用する.
  - 今回の反省点を活かして, よりよい開発体制を目指す.

さいごに

# 研究本部のデジタル系人財募集状況

- 2023年6月現在, 8職種の募集がございます. (https://www.chugai-pharm.co.jp/recruit/career/index.html)




**研究本部 / Research Division** +

**データサイエンティスト（AIを活用した化学創薬技術開発の研究者）/ Data scientist (Specialist in the development of AI-based chemical drug discovery technologies.)**
中外ライフサイエンスパーク横浜 Chugai Life S…

**創薬研究ラボオートメーションスペシャリスト / Laboratory Automation Specialist for Drug Discovery Research**
中外ライフサイエンスパーク横浜 Chugai Life S…

**ゲノミクス・プロテオミクスのデータ解析スペシャリスト / Data Scientist in Genomics and Proteomics Research**
中外ライフサイエンスパーク横浜 Chugai Life S…

**医薬品研究のデータサイエンティスト / Data Scientist in Pharmaceutical Research**
中外ライフサイエンスパーク横浜 Chugai Life S…

**医薬品分子設計の機械学習研究者 / Machine learning researcher in drug discovery**
中外ライフサイエンスパーク横浜 Chugai Life S…

**医薬品研究のデータエンジニア / Data Engineer in Pharmaceutical Research**
中外ライフサイエンスパーク横浜 Chugai Life S…

**代謝分析研究員 / Metabolite analysis researcher**
中外ライフサイエンスパーク横浜 Chugai Life S…

**データサイエンティスト（創薬のためのタンパク質科学研究におけるデジタル化担当）/ Data Scientist for Digitization in Protein Science Research Field of Drug Discovery**
中外ライフサイエンスパーク横浜 Chugai Life S…

Thank you

# Acknowledgement


平山裕之

PyZAP

長島慶宜

PyZAP

角崎太郎

MALEXA

PyZAP

高萩航太郎

MALEXA

成島大智

MALEXA

# 創造で、想像を超える。