

発表概要

内部状態遷移モデルの変更により機能拡張可能な
ワークフロー実行処理系の実現に向けて丹生 智也^{2,1,a)} 佐賀 一繁¹ 竹房 あつ子¹ 大江 和一¹ 合田 憲人¹

2022年1月12日発表

ゲノム解析などの大規模データ解析では、複数の解析ツールから構築されたワークフロー（パイプライン）が用いられることが多い。近年では Make などのビルドツールや Python などのプログラミング言語の代わりに、ワークフロー記述言語と呼ばれる DSL を用いてワークフローが記述される。ワークフロー記述言語では各ステップで利用する解析ツールおよびステップ間の依存関係のみを記述し、クラウド利用やジョブスケジューラ利用などの実行環境の違いおよび失敗したステップの再実行などの実行方法の詳細は処理系であるワークフローエンジンにより吸収される。一方で、対応する実行環境などはワークフローエンジンの実装状況に依存するため、実行環境によっては効率的なデータ解析が行えない可能性がある。ワークフローエンジンの修正による対応は技術的には可能だが、その変更範囲や作業量を見積もることは容易ではなく、またメンテナンス性にも課題がある。本発表では、この問題を解決するために開発中の状態遷移に基づくワークフローエンジン ep3 について述べる。ep3 は与えられたワークフローの実行に必要な状態遷移をペトリネットとして表現し、シェルコマンド実行による状態遷移によりワークフローの実行を行う。また ep3 外部からペトリネットの構造および実行するシェルコマンドを変更することで、ep3 本体に修正を加えることなく機能拡張が可能になると期待できる。

Presentation Abstract

Towards an Extensible Workflow Engine by Modifying Its Internal
State Transition ModelTOMOYA TANJO^{2,1,a)} KAZUSHIGE SAGA¹ ATSUKO TAKEFUSA¹ KAZUICHI OE¹ KENTO AIDA¹

Presented: January 12, 2022

In large-scale data analysis such as genome analysis, researchers construct workflows (also known as pipelines), consisting of multiple data analysis applications. In recent years, workflows have been described in DSLs called workflow description languages instead of build systems such as Make or programming languages such as Python. Workflow description languages enable us to focus on the data analysis application for each step and the dependencies between steps. A workflow engine is a system to execute workflows and can encapsulate where a given workflow is executed (e.g., cloud computing resources and computing nodes in job schedulers), and how a given workflow is executed (e.g., re-execution of failed steps). However, researchers may not execute their workflows efficiently enough on their platforms because a given workflow engine does not support them. Although it is technically possible to fix the workflow engine to support them, it is hard to estimate where to fix and how long does it take. To solve this problem, this presentation describes our ongoing work of ep3, a workflow engine based on a state transition model. It represents a given workflow as Petri nets, and executes them by executing shell commands for state transitions. It enables users to modify the network structure and shell commands to be executed via external settings and thus we expect that it enables us to extend its functionality without modifying the workflow engine itself.

This is the abstract of an unrefereed presentation, and it should not preclude subsequent publication.

¹ 国立情報学研究所
National Institute of Informatics, Chiyoda, Tokyo 101-8430,
Japan

² 国立遺伝学研究所
National Institute of Genetics, Mishima, Shizuoka 411-8540,
Japan
a) tanjo@nig.ac.jp