

Rakuのサーバーを使った実行

Running Raku using Raku server

165727F 氏名: 福田 光希 指導教員: 河野 真治

2019/10

概要

1 研究目的

Perl5の後継言語として,Rakuが開発されている。RakuはMoarVMというVM上で動作するが,起動時間がPerl5やPython,Rubyなどの主要なスクリプト言語に比べて非常に遅い。その為,本研究ではRakuプログラムの新たな実行手法の提案,またOS上でスクリプト言語を実行する最適な方法の見直し,スクリプト言語をサーバーで実行する利点と欠点の考察を行う。

2 Perl6

3 MoarVM

4 問題点

5 まとめ

中間予稿までにやったこととして,Paxosの論文を読み,ChristieにTopologyManagerという機能を実装した。

Paxosを読んだ理由は,コンセンサスアルゴリズムの調査である。実際,Paxosもビットコインで使用される候補に上がったコンセンサスアルゴリズムである。分散システムはどのようなコンセンサスアルゴリズムを用いているかで性能が変わる。例えばビットコインのコンセンサスアルゴリズムProof of Workは,計算量を多くして改ざんを起りにくくしているが無駄が多く,10分以内で解かれないように動的に条件を変更している。これは先ほどの,同時にブロックを変更するのを防ぐため,つまり信頼性を上げるためであるが,速度面で大きな課題となる。分散ファイルシステムを構成するにはスケラビリティが課題であり,ノードの数が多くなればなるほど通信時間がかかる。そのため,コンセンサスアルゴリズムとして有名なPaxosの論文を読んだ。

ChristieにTopologyManagerを実装した理由は,Christieのコードに慣れるため,そしてTopologyManager上に分散システムを実装するのが容易になるからである。TopologyManagerとは,ノードにTopologyを構成させ,ノードごと

にどこのノードにつながればいいのかを指定する機能である。Christieでは静的,動的なトポロジー管理ができる。静的ではdotファイルというものにノードごとの関係を記述する。動的ではノードの木構造を作る。

また,ブロックチェーンについては実際にブロックを実装し,簡易的ではあるがProof of Workを動かして理解を深めた。

今後の課題として,ブロックチェーンのトランザクション部分と分散環境を実装する。そして,実際に分散環境下においてブロックチェーンを動かす,データの整合性保持,追跡が行えるかを確認していく。コンセンサスアルゴリズムも調査していき,ファイルシステムに組み込めるコンセンサスアルゴリズムを探していきたい。