

発表概要

Ruby 3に向けた新しい並行実行モデルの提案

笹田 耕一^{1,a)} 松本 行弘^{1,b)}

2016年10月26日発表

マルチスレッドプログラミングは、共有するデータに対して、複数のスレッドから同時に読み込み、書き込みを行うとデータレースなどの問題を発生させるため、排他制御などの同期を適切に行わねばならず、難しい。この問題は、スレッドが変更可能なデータを簡単に共有できるため生じるので、そのような共有を制限する方法が必要である。プログラミング言語 Ruby の次期メジャーバージョンである Ruby 3 では、マルチスレッドプログラミングの持つ難しさを緩和するため、“Guild” という Ruby 向けの新しい並行実行モデルを検討している。インタプリタは1つ以上の Guild を持つ。変更可能なすべてのオブジェクトはある1つの Guild に所属し、Guild 間では、変更可能なオブジェクトを共有しない。ただし、不変オブジェクトや、クラスやモジュールは共有する。Guild は1つ以上のスレッドを持ち、異なる Guild は並列に実行する。Guild 間の通信はチャンネルを通して行い、変更可能なオブジェクトを Guild 間で通信するにはコピーと移籍という2つの方式を提供する。移籍は、所属する Guild を変更する操作であり、単なるコピーより軽量となる。性能などの観点から、変更可能なオブジェクトを共有しなければならない場合、特別なデータ構造を提供することができる。本提案は、既存のアイデアの「いいとこどり」を狙ったものである。本発表では、この並行実行モデルの目論見について議論する。

A Proposal for a New Concurrent Execution Model in Ruby 3

KOICHI SASADA^{1,a)} YUKIHIRO MATSUMOTO^{1,b)}

Presented: October 26, 2016

Multi-thread programming is difficult because data which shared by multiple threads requires appropriate synchronization by mutual exclusion or other techniques to avoid issues such as data race and so on. This problem is because sharing mutable data between threads easily, so that we need a way to restrict such sharing. Ruby 3 is the next major release of programming language Ruby. Ruby 3 aims to introduce new concurrent execution model for Ruby named “Guild” to avoid multi-thread programming difficulties. An interpreter has at least one Guild. All of mutable objects belong to one guild and multiple guilds can not share mutable objects. But guilds can share immutable objects and class/module objects. Guild has at least one thread and threads belong to different guild can run in parallel (simultaneously). Inter guild communication is given by channel. There are two way to communicate mutable objects each other, one is copy and another is transfer membership (shortly, move). Transferring membership means changing belonging guild to another guild. It is faster than simple copying. If we need to share mutable objects because of performance, we can offer special data structure. Our proposal is to cherry picking existing good ideas. In this presentation, we will discuss about our concurrent execution model design.

¹ Heroku, Inc.
Heroku, Inc., Chiyoda, Tokyo 100-7012, Japan

^{a)} ko1@heroku.com

^{b)} matz@heroku.com