

発表概要

世代別 Mostly-Copying GC の Ruby VM への実装に向けて

永原 治^{†1} 鷓川 始陽^{†2} 岩崎 英哉^{†2}

近年, Web アプリケーションをはじめとする Ruby による大規模アプリケーションの開発が行われるようになってきている. 大規模なアプリケーションの特徴として, 大量のデータを扱い長時間動作し続けるという特徴があげられる. 既存の Ruby 処理系は保守的マークスイープ方式でごみ集め (GC) をしているが, このような特徴を持つアプリケーションでは, 世代別コピー GC で GC の時間を短縮できることが知られている. しかし, Ruby 処理系には保守的 GC を前提に作られている部分があり, 移動できないオブジェクトがあるため, 通常のコピー GC は実現できない. そこで, 移動できないオブジェクトが存在しても, コピー GC を実現できる Mostly-Copying GC を用いることにした. 本研究では, 世代別 Mostly-Copying GC の実装による GC 時間の短縮を目標として, Ruby1.9 の VM に Mostly-Copying GC の実装を行い, Ruby on Rails 組み込みの Web サーバなどを対象に移動できないオブジェクトの数やその寿命など世代別 GC の設計に必要なデータ収集を行った. その結果, 移動できないオブジェクトの割合は少なくないが, それらは長寿命であるという傾向が分かった. 本発表では, これらのデータを基に世代別 Mostly-Copying GC の実装方針と設計について述べる.

Toward an Implementation of Generational Mostly-Copying GC on Ruby VM

OSAMU NAGAHARA,^{†1} TOMOHARU UGAWA^{†2}
and HIDEYA IWASAKI^{†2}

Recently, large-scale applications, such as web applications, have come to be developed in Ruby. A large-scale application tends to treat a large amount of data and to keep working for a long time. While the Ruby VM has the conservative mark-sweep GC, it is known that the time spent on GC in such an application can be reduced by using the generational copying GC. However, we

cannot implement the copying GC on the Ruby VM, because part of the VM depends on the conservative GC, and some objects cannot be moved. Therefore we decided to use the mostly-copying GC, which collects garbage by copying only movable objects. In this research, we aim to reduce the time spent on GC by implementing the mostly-copying GC on the VM of Ruby 1.9. We implemented the mostly-copying GC on the VM and examined the behavior of objects in the heap, such as the ratio of immovable objects and their lifetime, using the VM and a web server program of Ruby on Rails. As a result, we found immovable objects tend to be long-lived though the ratio of them is not small. In this presentation, we present the design of generational mostly-copying GC based on the observation.

(平成 21 年 10 月 29 日発表)

†1 電気通信大学大学院電気通信学研究科情報工学専攻

Department of Computer Science, Graduate School of Electro-Communications, The University of Electro-Communications

†2 電気通信大学情報工学科

Department of Computer Science, The University of Electro-Communications