

***YARV: Yet Another **Ruby**VM
... on Rails?***

2006 10/22

RubyConf 2006 @ Denver

SASADA Koichi

Nihon **Ruby no Kai**

The University of Tokyo

Caution! (re-review)

- I can't speak English well
 - If I say strange English, you can see the slide page
 - Or ask another Japanese. They can speak English well.
 - If you have any question, ask me with:
 - Japanese (recommended)
 - **Ruby**, C, Scheme, Java, ..., Python, Haskell,
-
- IRC (@freenode#rubyconf)
 - Easy English (less than 10 words with easy words)

Agenda

- Purpose
 - To show A Rail from YARV to **Ruby**
- Agenda
 - Self Introduction (third)
 - YARV Progress Report
 - YARV Internal Briefly
 - YARV Features
 - A Rail from YARV to Rite or **Ruby** 1.9.1
 - Conclusion

自己紹介

Self Introduction

...Japanese must use the Takahashi Method

Self Introduction

- SASADA (family) Koichi (given)
 - 笹田 → SASADA, 耕一 → Koichi
- Japanese use some patterns to represent own name in English
 - “#{given.capitalize} #{family.capitalize}”
 - Koichi Sasada
 - “#{given.capitalize} #{family.upper}”
 - Koichi SASADA
 - “#{family.upper} #{given.capitalize}”
 - SASADA Koichi (my favorite)

Self Introduction (cont.)

- A member of Nihon-**Ruby**-no-Kai
 - **Rubyist** Magazine
 - Webzine about **Ruby** in Japanese
 - Vol. 16, 2006
 - Got second anniversary
 - **RubyKaigi** 2006
 - First time **Ruby** event in Japan

RubyKaigi 2006

- Held in Japan, Tokyo
- Two keynote speakers
 - Matz
 - DHH
- 200 tickets are sold out in 3 hours

RubyKaigi 2007

- 2007 6/9 (Sat), 6/10 (Sun)
- Tokyo, Japan
- Official Language are Japanese & Ruby
- Two keynote speakers
 - Dave Thomas
 - Matz
- We'll announce about it next year.
We'll wait for your attendance

Self Introduction (cont.)

- Rubyist Magazine
 - <http://jp.rubyist.net/magazine/>



The screenshot shows the top-left portion of the Rubyist Magazine website. It features the site's logo, which consists of the Japanese characters 'るびま' (Rubyist) in a stylized font, with '日本Rubyの会' (Japanese Ruby Association) written below it. To the right of the logo is a search bar with the text '検索' (Search). Below the search bar, the issue number '0016号 (2006-09)' is displayed. A list of articles follows, including '巻頭言' (Editorial), 'Rubyの歩き方' (How to walk Ruby), 'るびまへのたより' (Message to Rubyist), 'Rubyist Hatlinks (16) essay', 'BundledLibs (9) PStore', and '他言語探訪 (9) ねでこ'.



The screenshot shows the main content area of the Rubyist Magazine website. At the top right, there are navigation links: 'るびま', 'ページ一覧' (Page List), '検索' (Search), '更新履歴' (Update History), and 'ログイン' (Login). The main title 'Rubyist Magazine' is prominently displayed in a large, white, serif font with a drop shadow. Below the title, the word 'るびま' is written in a smaller font. A horizontal line separates the header from the main content. On the right side, the update date '更新日時:2005/10/10 21:36:06' is shown. Below this, there are two sections: 'Rubyist Magazine について' (About Rubyist Magazine) and '最新号' (Latest Issue). The 'Rubyist Magazine について' section contains the text: '『Rubyist Magazine』、略して『るびま』は、日本 Ruby の会の有志による Rubyist の Rubyist による、Rubyist とそうでない人のためのウェブ雑誌です。' (『Rubyist Magazine』, abbreviated as 『るびま』, is a web magazine for Rubyist and non-Rubyist people, created by Rubyist members of the Japanese Ruby Association.)

Self Introduction (cont.)

- A member of Nihon Perl-no-Kai
 - Perl Association in Japan
 - 2006 1st Apr.
 - Perlish Magazine
 - Webzine about Perl in Japanese
 - <http://jp.rubyist.net/PerlMa/>
 - No PerlKaigi

閒話休題

Back to the YARV Presentation

DEMO

Glossary

- Rite: Code name of **Ruby** 2.0
 - a.k.a. vaporware name (vs. Parrot)
 - First day to appear is in 2001-03-28
 - vs. Parrot: 2001-04-01
 - ... Rite > Parrot ?(level of vaporware)
 - ... Rite will be fast than Parrot 😊
- YARV: Yet Another **Ruby**VM
 - First day to appear is 2004-01-01

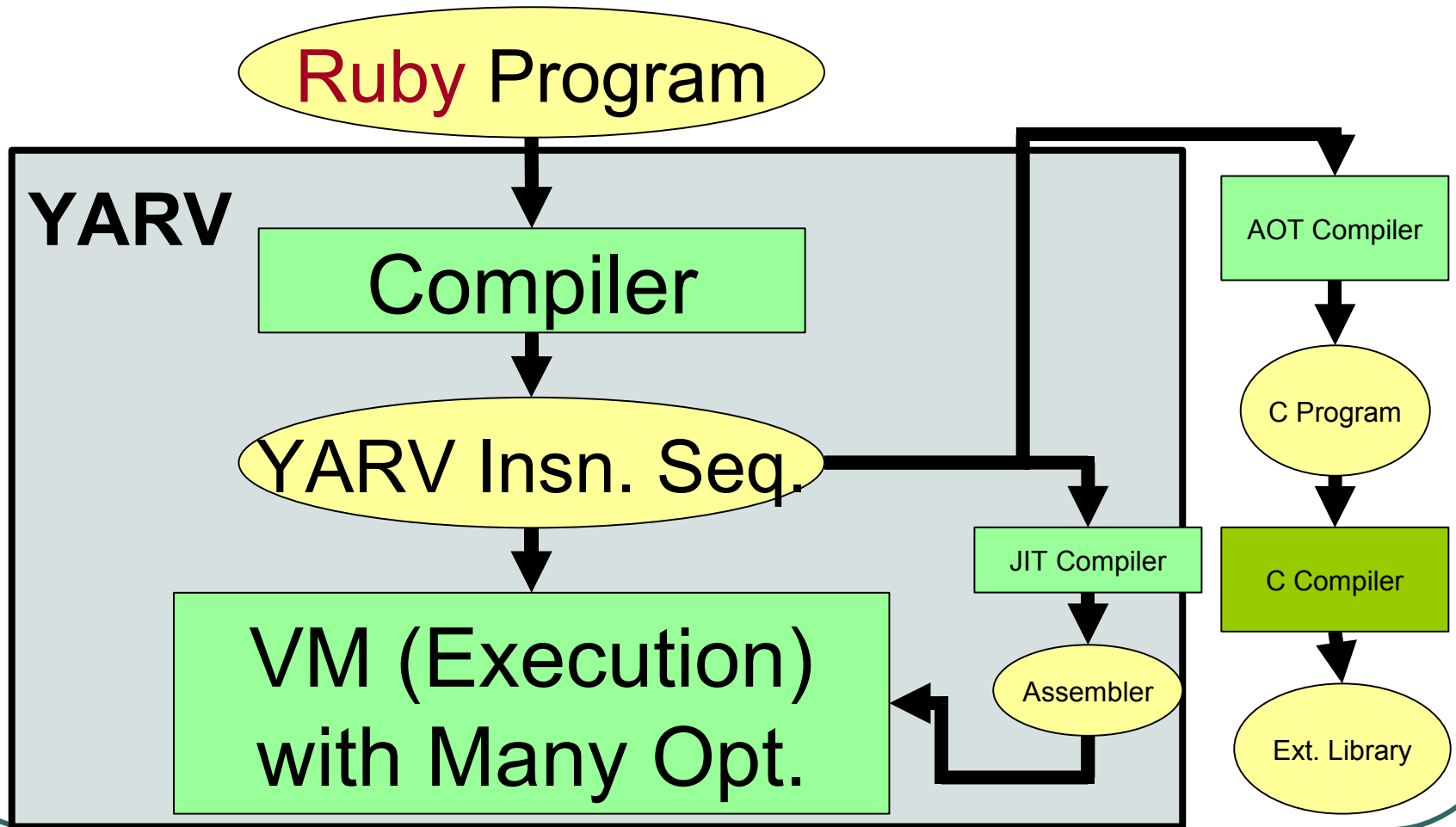
Ruby Virtual Machine

YARV: Yet Another RubyVM

- Supported by funds
 - By IPA, Japan (Finished)
- Virtual Machine
 - Simple Stack Machine
 - Specific VM Instructions (YARV Instructions)
 - Compiler
 - Interpreter (VM)
 - Many Optimization techniques to improve performance
- Open Source Software
 - <http://www.atdot.net/yarv/>



Overview of YARV



Optimizations for performance

- Compile time optimization
- Direct threaded code
- Specialized instructions
- Operands/Instructions Unifications
- Inline (Method) cache
- Static stack caching
- Profiler
- And more...

このドキュメントは、Ruby 2.0の最適化とマルチVMアーキテクチャに関する技術的考察と実践の集大成です。各スライドは、特定の技術的課題を解決するための設計思想、実装手法、およびその効果を示しています。

主要なトピックとスライド番号:

- Rubyの処理系の刷新 (22):** 今年度開始のRubyとのマージ、RubyVMのインストール、Rubyの処理系刷新の必要性。
- 設計: YARV命令セット (24):** 現時点では52命令、VM生成系による動的に生成された命令、プリティブ型がないため、数値演算命令無し。
- 設計: VM (26):** 利用する環境の仕様の仕様の基本利用、VM生成系命令の生成、スレッド管理、VM生成系命令の生成。
- 設計: VM生成系 (56):** VM生成系の実装、VM生成系の実装、VM生成系の実装。
- 設計: スレッドモデルの検討 (63):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (64):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (65):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (66):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (67):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (68):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (69):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (70):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (71):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (72):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (73):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (74):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (75):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (76):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (77):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (78):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (79):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (80):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (81):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (82):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (83):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (84):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (85):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (86):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (87):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (88):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (89):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (90):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (91):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (92):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (93):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (94):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (95):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (96):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (97):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (98):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (99):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。
- 設計: スレッドモデルの検討 (100):** スレッドモデルの検討、スレッドモデルの検討、スレッドモデルの検討。

進捗報告

YARV Progress Report

Progress Report

- YARV can build with configure/make
 - Also work on Windows
 - Doesn't work with Autoconf 2.6.0
 - ... Do you know reason?
- YARV passes most of tests
 - But “make test/test-all” claimed some errors because of detail spec. mismatch
- YARV can run most of **Ruby**
Programs/Libraries

DEMO

Using YARV

機能紹介

Introduction YARV Features

YARV Feature

YARV's MYTH

- YARV has some MYTH like

YARV is GREAT! and

**YARV will solve
ALL Problems!**

YARV Feature

YARV's MYTH (cont.)

- YARV make fast all **Ruby** Program
by **50** times

YARV Feature

YARV's MYTH (cont.)

- YARV solve problem

CHARACTER CODE

Problem!

- YARV puts down **frame WAR!**

YARV Feature

YARV's MYTH (cont.)

- YARV make you **RICH!**
- YARV make you **SLIM!**
- YARV finds my
GIRL FRIEND!

YARV Feature

YARV's MYTH (cont.)

BAD JOKE

YARV is not Silver Bullet

YARV Features

BASIC

- Running **Ruby** Programs
 - ... That's for sure
 - ... Can't run Perl program 😊

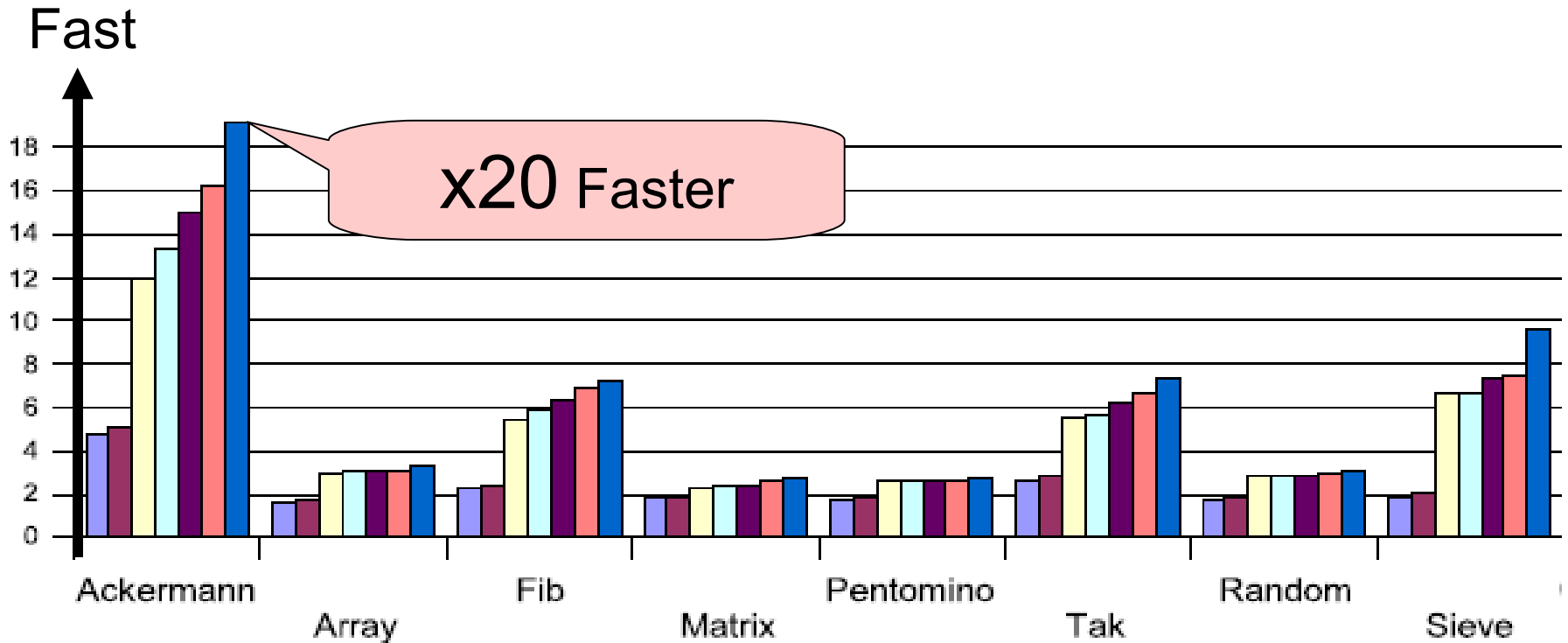
YARV Features

Performance Improvement

- **GOT IMPROVE**
 - VM Bottleneck Programs
 - Numerical Programs
 - Symbolic Processing

YARV Features

Performance Improvement (cont.)



YARV Features

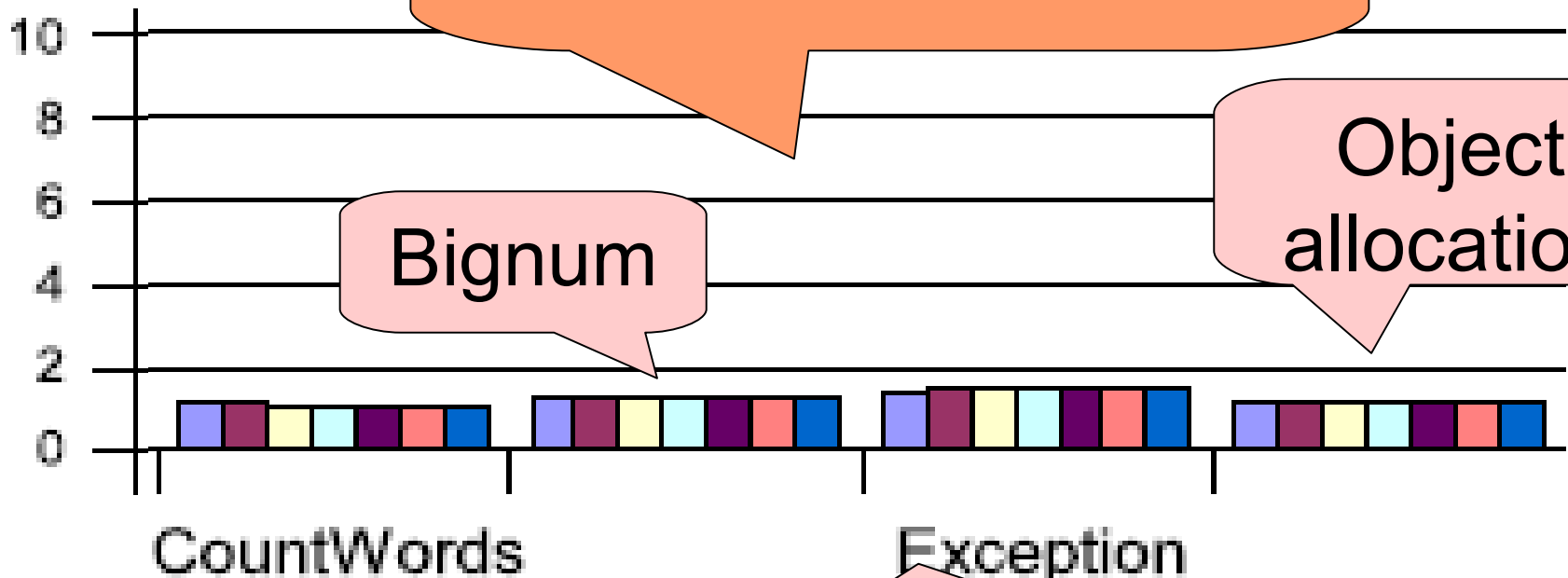
Performance Improvement (cont.)

- DOES NOT IMPROVE
 - NOT VM Bottleneck programs
 - String/Text Processing
 - Program which allocate huge Objects
 - ...

YARV Features

Performance Improvement (cont.)

No speed-up.
VM is not bottleneck.



Bignum

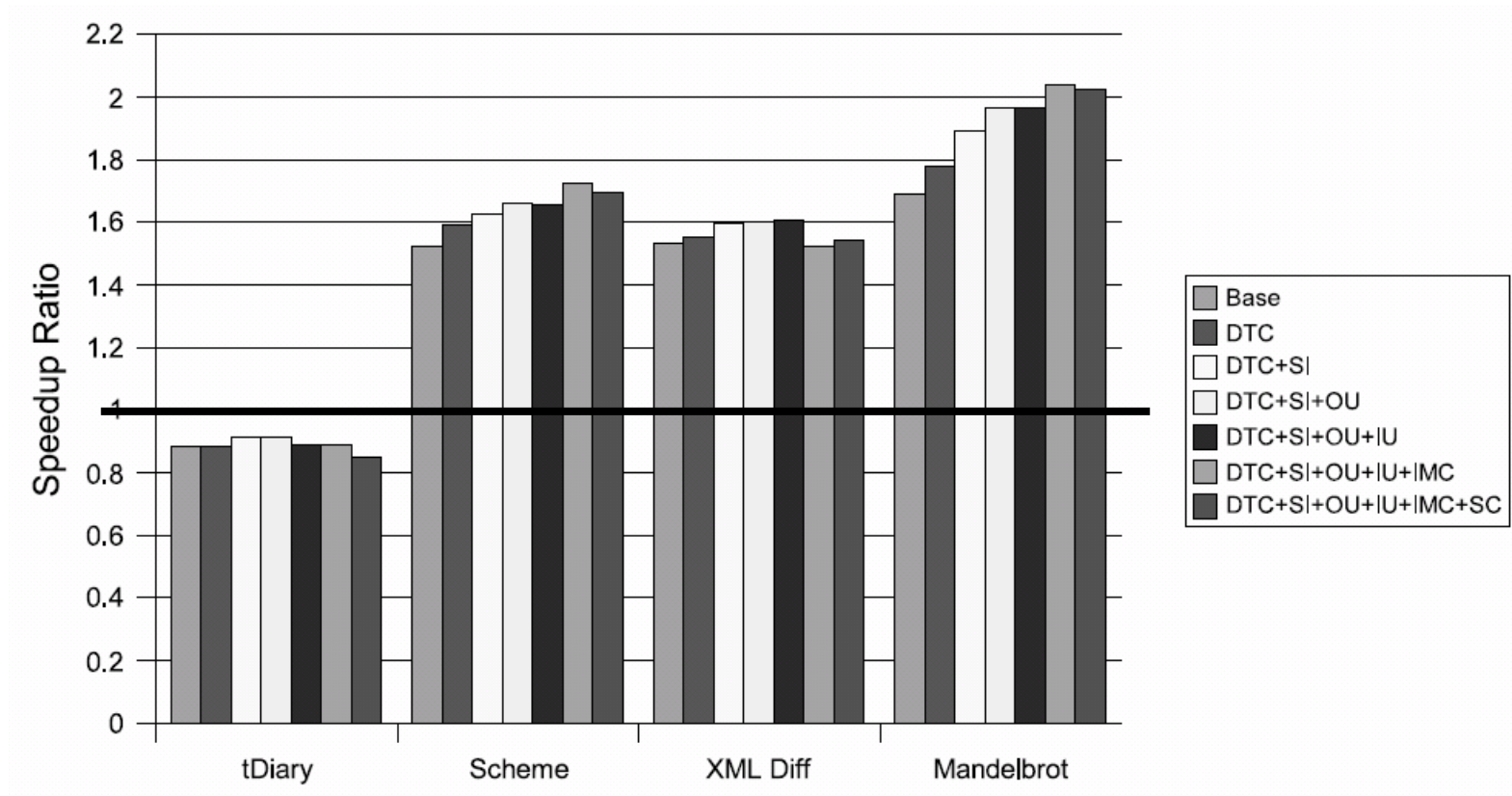
Object allocation

Regex

Build Exception object

YARV Features

Performance Improvement (cont.)



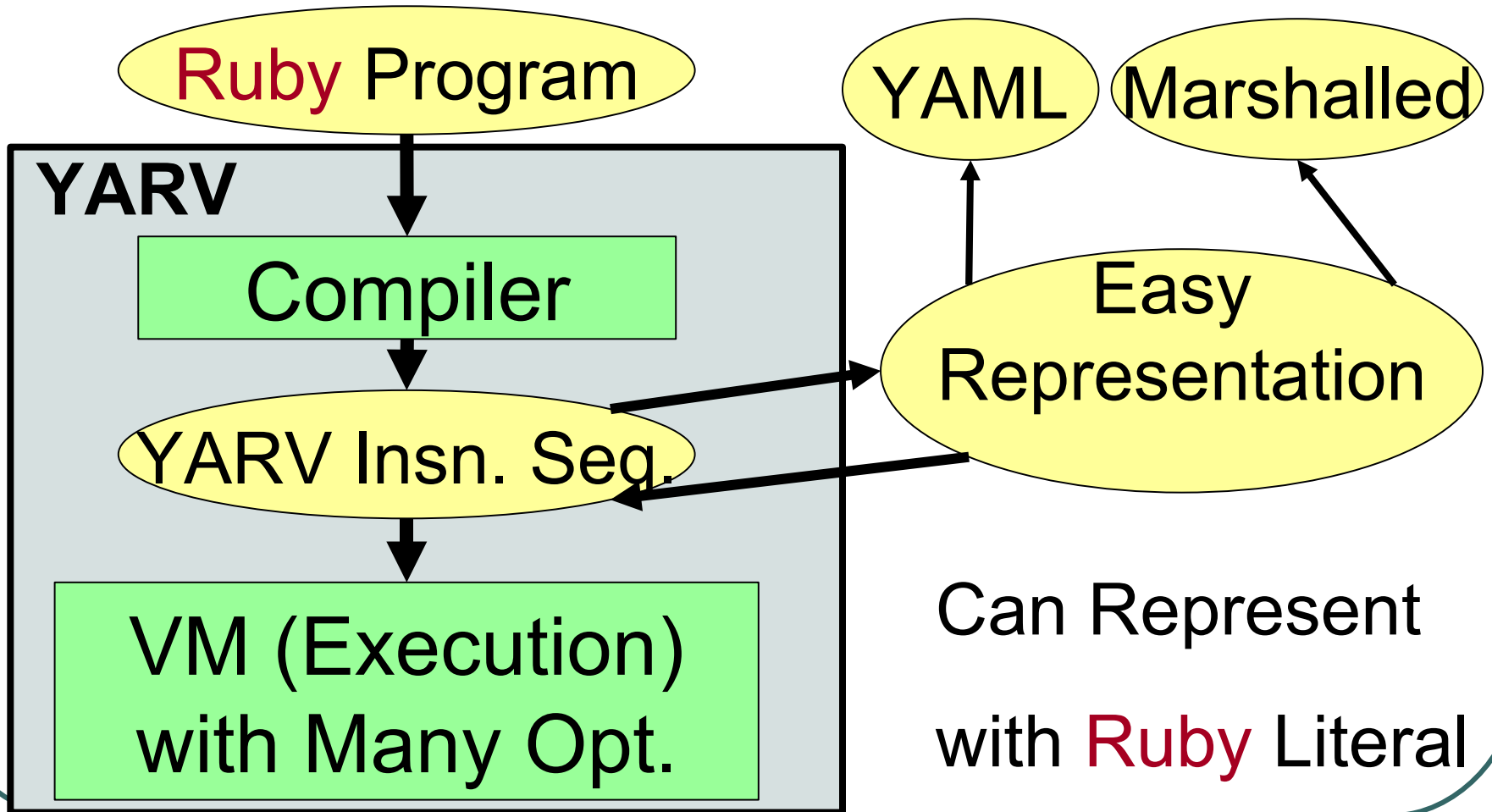
YARV Features

Instruction Sequence

- Assemble
 - You can assemble YARV iseq
- Disassemble
 - You can see disassemble results of **Ruby** Scripts
- Serialize and De-Serialize
- YARVCore::VM::InstructionSequence
 - Class name will be changed

YARV Features

Instruction Sequence (cont.)



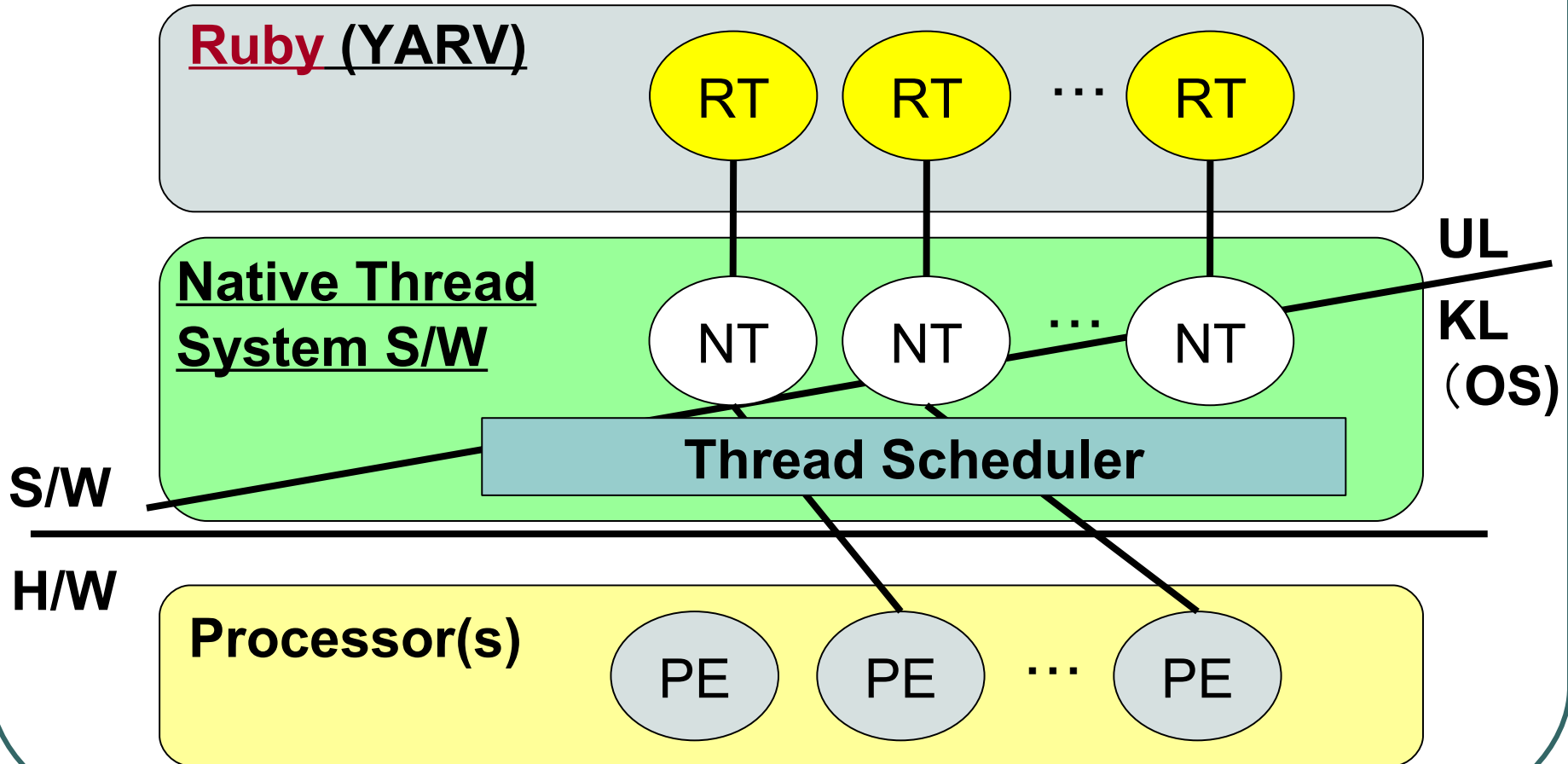
並列処理

Parallel Processing on YARV

YARV Thread System

- A **Ruby** Thread (RT) is mapped with A Native Thread (NT)
 - Supporting POSIX Thread or Windows Thread

Overview of YARV Thread system

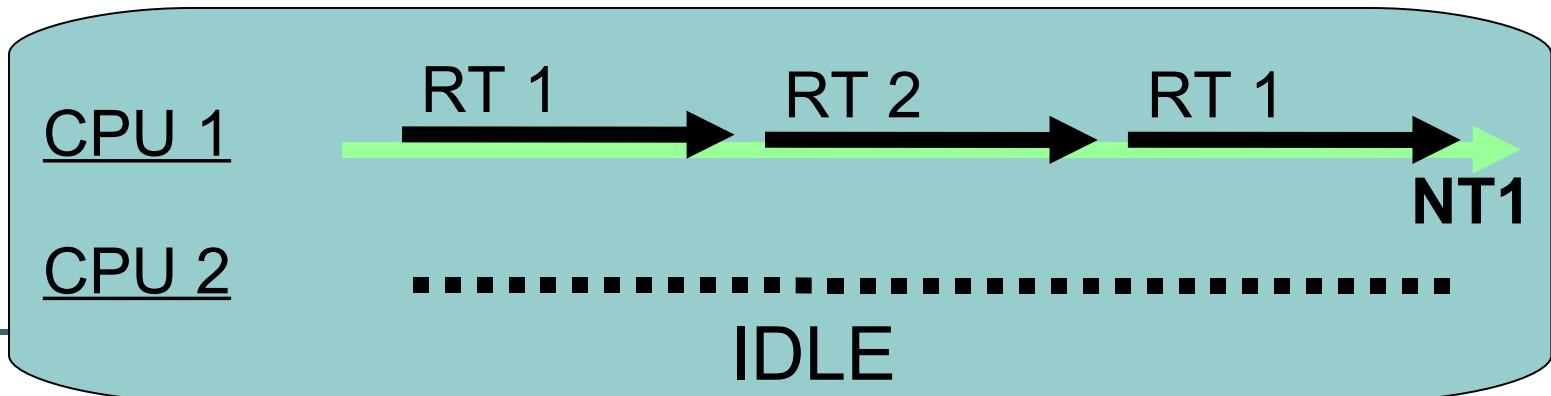


PE: Processor Element, UL: User Level, KL: Kernel Level

Current **Ruby** Thread Model (Thread model 1)

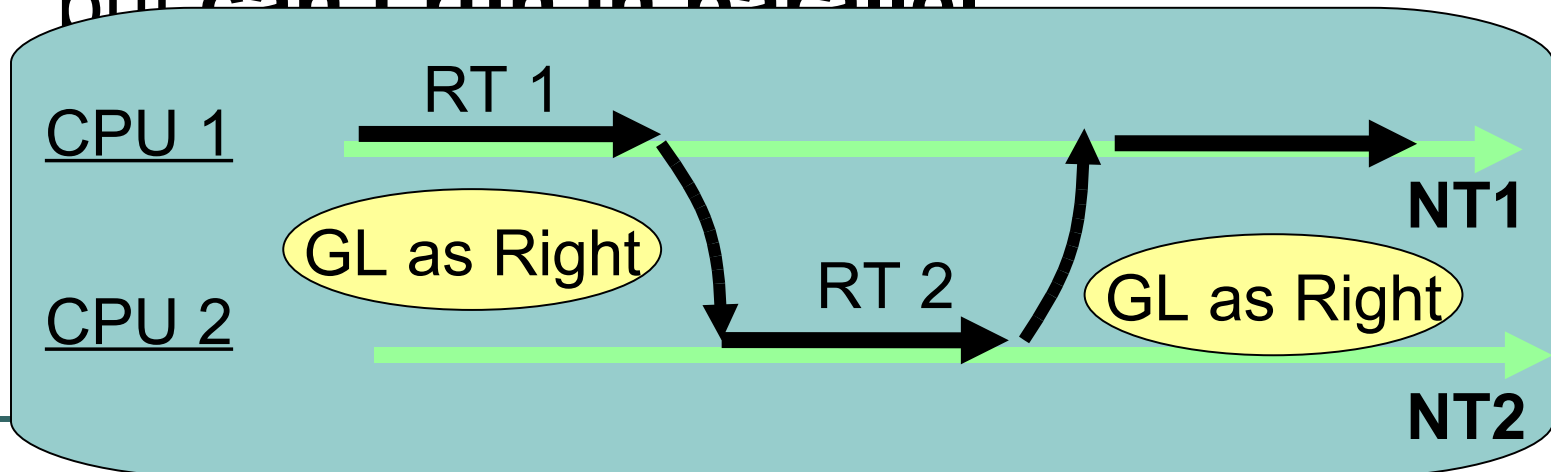
- No Parallel Execution
- No need to synchronize each thread in C code → **So much C Libraries need to add synchronization**

vs. Java



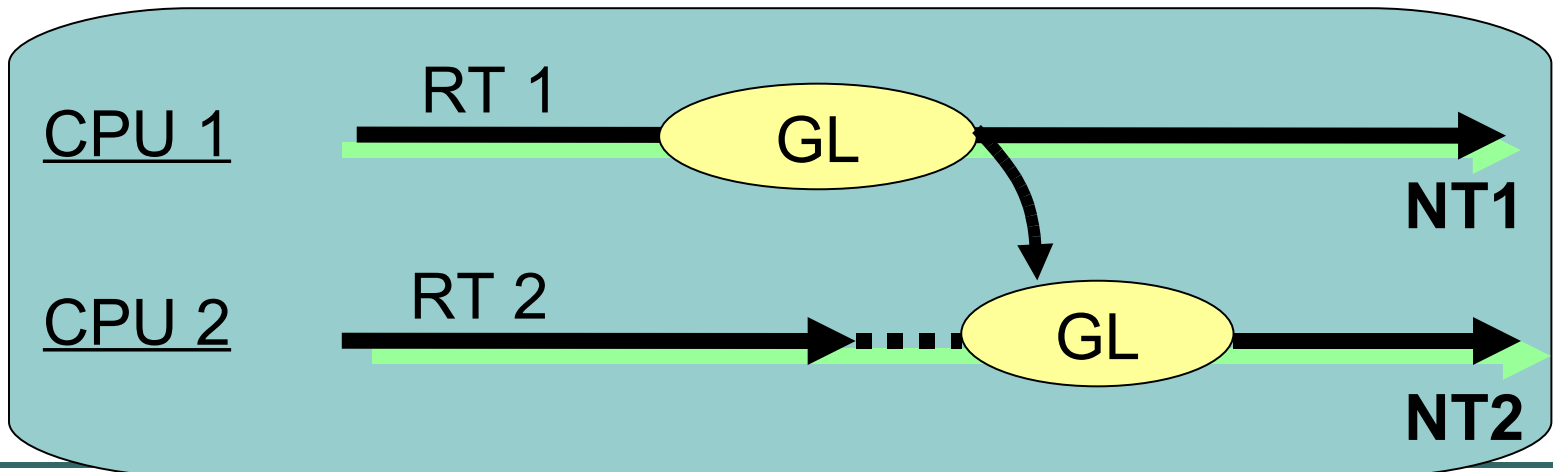
Thread model like Python (Thread model 2)

- Using Native thread for each **Ruby** Thread
- Make a Giant Lock (GL)
- Only a thread which has GL can run
- **No need to add sync. codes,**
but can't run in parallel



Parallel Execution with GL (Thread model 3)

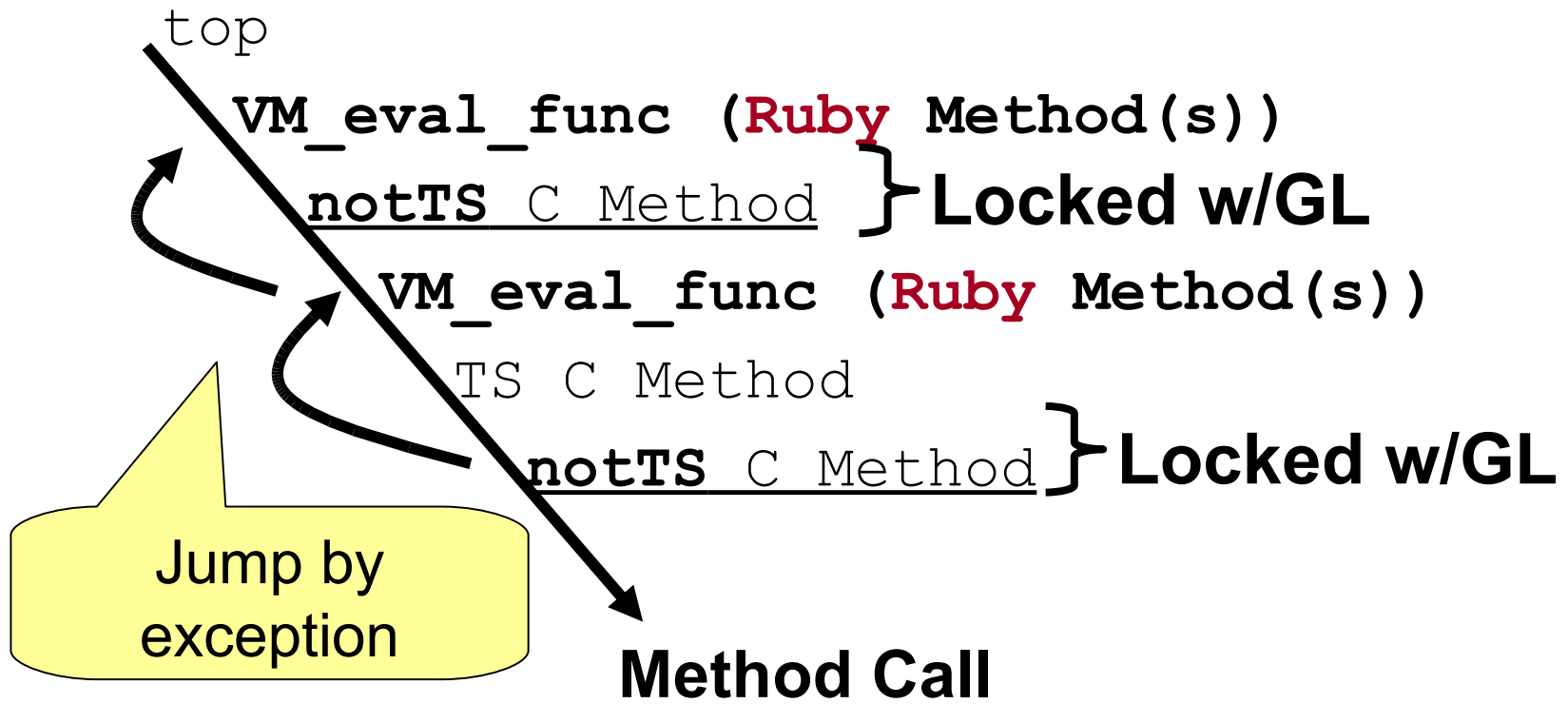
- **Ruby** Threads in Parallel
- When invoking non-thread safe code, VM gets GL



GL: Giant Lock

Giant Lock Management

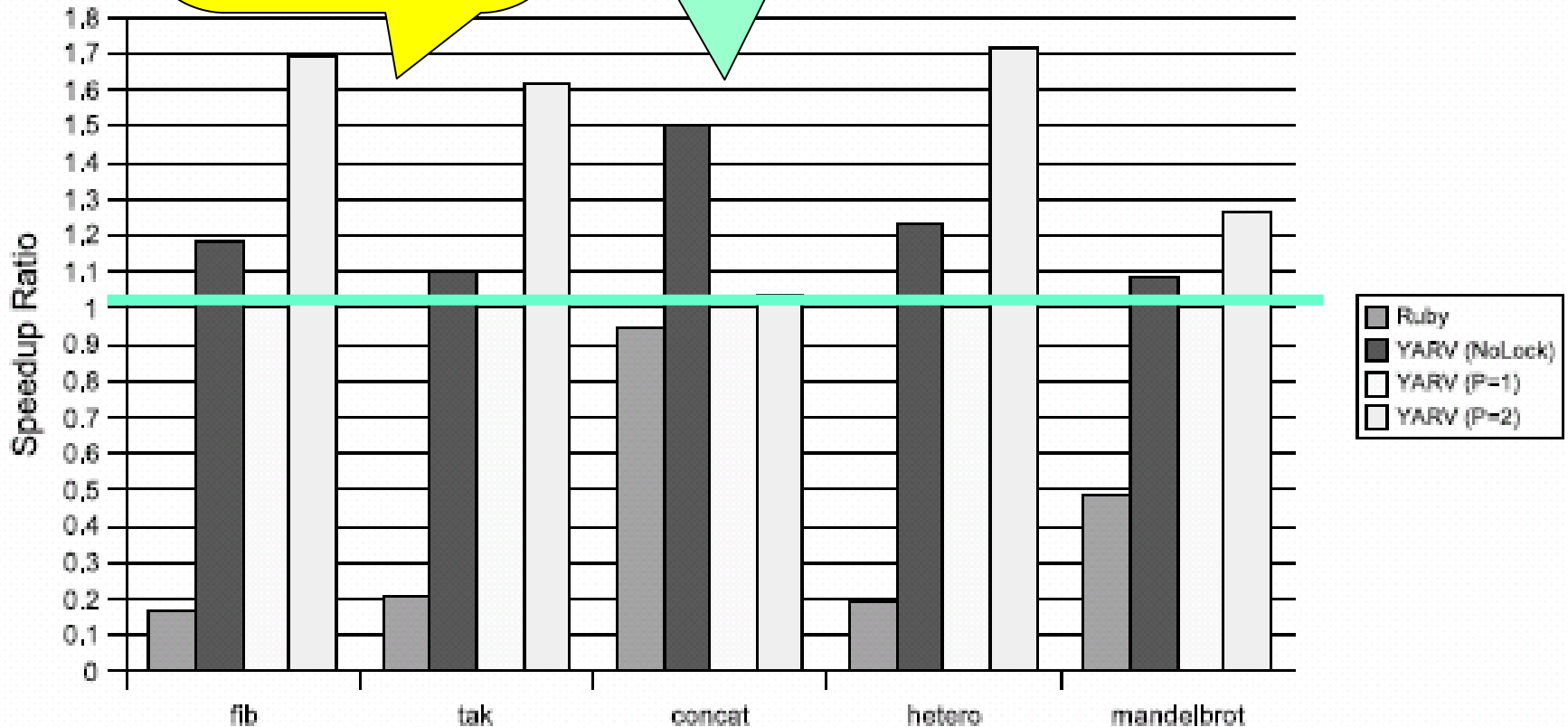
Machine level Call Graph:



Parallel YARV Evaluation

Good Scalability

GL Conflicts
Bad Scalability



YARV Thread Specification (limitation)

- Mutex class will be built-in
- Thread.critical will be vanished
 - Not be obsolete, but be unsupported

公式処理系 への道

A Rail to Official **Ruby** Interpreter

The Rail from YARV to Rite

- “**Matz said**” that he will release **Ruby** 1.9.1 on 2007 Christmas
- **Ruby** 1.9.1 merged with YARV
- Merged with official and YARV in 2007/Spring? Summer?

The Rail from YARV to Rite Specification

- Thread model will be model 2
 - No parallel execution
 - Easy to write C Extension Libraries
 - No synchronization code

開発体制

YARV Development System

YARV Development

- Developers
 - SASADA Koichi ... only one?
 - Matz has an account to access YARV repository
 - He is busy to play with Current **Ruby** and **Ruby** Spec.
- Test codes are very welcome!

YARV Development (cont.)

- Mailing List as BTS
 - Register Bug entry with ML post
 - “Subject: [BUG]> foobar” or
 - “Subject: open> foobar”
 - You can see registered bug entries at <http://www.atdot.net/~ko1/yarvbugs/>

YARV Development (cont.)

- IRC
 - Freenode#yarv is official English channel but I'm always away...
- Autobuild/Autotest
 - <http://i.loveruby.net/autobuild/yarv/>
 - By AOKI san
- You can find YARV Information on <http://www.atdot.net/yarv/>

まとめ

Conclusion

Conclusion

- **Ruby** will be Akiba-kei language
- YARV can build with “**make**” and can install with “**make install**”
- YARV can run most of **Ruby** Apps.
- YARV will be merged with 1.9.1 (2007 Christmas, **if you trust Matz**)
- YARV needs more developers or testers!

Conclusion: Future Work

- YARV lacks some big stuffs
 - Hook interface (set_trace_func) or debugging, profiling interface
 - Cache up Current **Ruby** 1.9
- JIT/AOT Compiler
 - AOT compiler is already implemented (but incomplete)

Additional Project

- **High-Performance Ruby**
 - One of my research
 - Try to realize High-Performance Computation in **Ruby**
 - GOAL: Easy to write HPC code
- vs. x10, Fortless, Chapel, ...
- I want to show it next year with **Ruby 1.9.1**

Finish

Thank you for your attention!

ご静聴ありがとうございました

ございました。

SASADA Koichi

ko1@atdot.net

