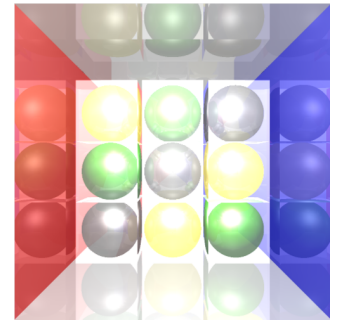


# CPU実験を騙る会

(CPUExSolverについて)

3班コンパイラ係 @\_furaga



話すこと

**CPUExSolverの宣伝**

# CPU実験は大変です

FPU

コンパイラ

コア

シミュレータ

ライブラリ  
(三角関数とか)

リンカ

アセンブラ

# CPUExSolverを使えば...

FPU

コンパイラ

コア

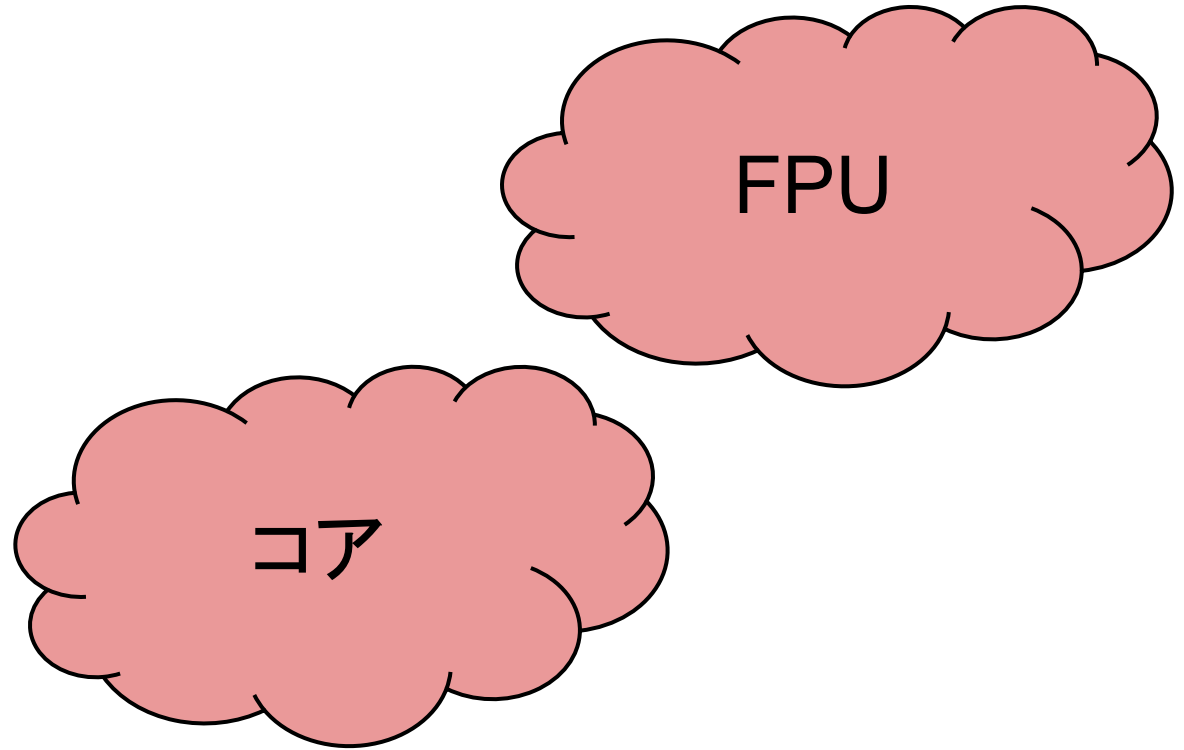
シミュレータ

ライブラリ  
(三角関数とか)

リンカ

アセンブラ

これだけ！



# CPUExSolverとは

@\_\_furagaがCPU実験の余興で作った

GUI上でアーキテクチャの仕様を設定し、その通りに

ソフトウェア班の製作物を **自動生成** するツール

コンパイラ

リンカ

アセンブラ

シミュレータ

ライブラリ

# しくみ

**CPUExSolver2**

**GUIでアーキテクチャ設定**

設定ファイル

ソースコードのテンプレート

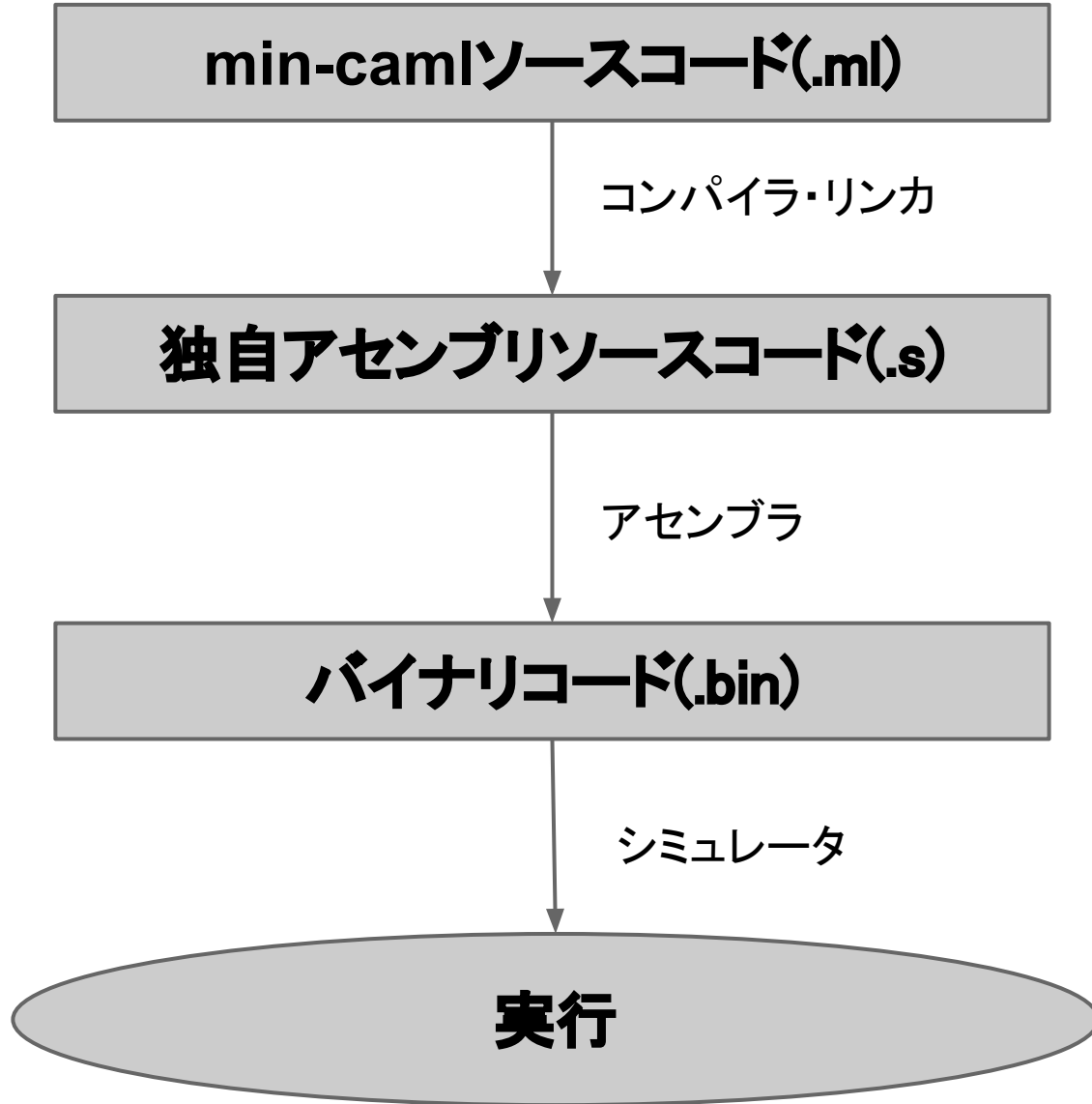
テンプレートエンジン

**アーキテクチャ**

(コンパイラ・シミュレータ・アセンブラ・リンカ・ライブラリ)

- ・ソースコード
- ・Makefile
- ・実行ファイル

# デモ





**利点**

1. コンパイラ・シミュレータ系の仕事を**全部**  
やってくれる

## 2. ライブラリ・シミュレータも生成

- min-camlはコンパイラのみ
- 例年はシミュレータ係がフルスクラッチで作成
- 例年、早くても10日くらいかかっている
- ソルバーを使えば10秒で作れる

### 3. コンパイラはそこそこ高性能

@\_\_furagaの3ヶ月にわたる苦難の成果

発行命令数 (min-rt) **14億命令** ~ 20億命令

**昨年度、最も最適化されたコンパイラ**

昨年度のコンパイラ係向け課題が実装済み！

- ・グラフ彩色によるレジスタ割り付け

でも...

これ使っていいの？

# 使っていいの？

- ・わかんない

- ・個人的にはまったく問題ないと思うけど・・・  
min-camlの亜種みたいなもの？

- ・担当のTAさんの裁量による

# もし使えなくても

- ・利用価値はあるはず
  - ・実装の参考
  - ・自作アーキテクチャが完成するまでの代用品
  - ・ライブラリくらいはそのまま使ってもいいはず
- ・CPU実験の仕様が大きく変わらない限り、無駄にはならない・・・と思う



# 実行環境

Ubuntu 12.04.1 LTS (64bit)で動作を確認

32bitだと動かないみたい

12erの学科PCでも動作を確認しました

# 使用言語など (赤文字はインストールする必要あり)

コンパイラ	OCaml
リンカ	Java
アセンブラ	C++
シミュレータ	C++
ライブラリ	min-caml, 自作アセンブリ
テンプレートエンジン	Python (Mako Templates)
アーキテクチャ設定UI	C++ (Qt Creator 2.4.1)

Java(JDK)は apt-get など

Qt Creatorは公式HPからインストール

# 導入方法

バイナリ配布が面倒だったので

ソースコードをビルドしてください

ソースコードはgithubに公開中

<https://github.com/furaga/CPUExSolver2>

ビルドには最新版のQt Creatorが必要  
詳しくは README.txt で！

# 注意

- ・コア・FPUは生成できない
- ・アーキテクチャの制約
  - 命令長: 32bitの固定長
  - R, I, J形式の命令のみサポート
- ・シミュレータが低機能
  - デバッグ機能なし
- ・GUI部分がわりとひどい(分かりにくい・使いにくい)
- ・バグは絶対ある
- ・その他常軌を逸した仕様やコードを見かけても**寛大な心で接してあげてください**

# まとめ

(多分)便利だから使ってね！  
ハードウェアのソルバー期待

**ご清聴ありがとうございました**