

Erlang 30% +
JavaScript 60% + 未知成分 10%

檜山 正幸 (HIYAMA Masayuki)

2008年12月20日 (土曜) 15:00開始

今日は、なにしよう

- 決まっているような決まってないような
 1. まえせつ -- 10分くらい
 2. Erlangのはなし -- 30分くらい
 3. ブラウザ-ブラウザ通信(デモ付き)のはなし -- 30分くらい
 4. リモート関数呼び出しとリモート・イベント配信のはなし -- 30分くらい
 5. 補足的な説明とか発展的なはなし -- 30分くらい(省略するかも)
- 単純に足し算すると130分ですが、適宜、質疑応答・ディスカッションを入れるので150分超になるでしょう。

なんで、なにを、 お話しするのか

- 今日に至る経緯

1. Erlangのミニブームに乗ってみた (2007年5月)
2. これ、面白いじゃん
3. 面白い道具で面白いことしよう (2007年秋)
4. ウタウダ、チンタラやっていた
5. そのあいだにErlangは、… あんまり盛り上がり
なかった
6. ウーンンン …
7. みんなに遊んでもらおう

遊ぶだけなの？

- いーえっ、ちゃんと勉強します！
- プログラミング言語によらない(言語中立な)、通信基盤にもよらない、関数呼び出し(function call)とイベント配信(event delivery)が主題です。

- プログラミング言語としてのErlang
 - おはなししません アームストロングの本などを読んでください。
 - PTTのスライドが幾分詳しいです。
 - が、今日は繰り返しません。
- システムとしてのErlang
 - おはなしします。
- ちなみに、アームストロング本の翻訳者の榊原(さかきはら)君は、昔(彼が学生の頃)スタッフなのでした。

- それ自体が完結したコンピュータシステム
- 仮想ハードウェア上に仮想OSが載り、アプリケーションが走る
- 仮想ハードウェア = エミュレータ
- 仮想ハードウェア + OS(シェル、基本アプリケーション含む) = ERTS
- 大量の軽量プロセスを使えることがウリ

- プロセス概念は通常のOSとそんなに変わらない、扱える量は違うが
- スレッドはない、あってはいけない！ プロセス1個にスレッド1個
- 対話的モードではシェル(EShell)が現れ、ユーザーとの対話を行う
- Webサーバーとして使うときなどは非対話的モードで実行することも可能
- むしろ、本格的な用途では非対話的モードが多そう

- 抽象機械語 = Core Erlang (形式的・操作的意味論があるらしい)
- 具体的機械語 = JAM機械語, BEAM機械語 (仕様は実装?)
- ソフトウェアによる実装 = JAMエミュレータ、BEAMエミュレータ
- Unix/Linux, Windows, VxWorksなどのホストOS上で動く
- 取り替え可能なprim(何の略?)ローダーによりOSカーネルをロード

- initが最初のプロセス
- kernelとstdlibはすべてErlangで書かれている
- ファイルシステムやネットワークはホストOS機能をそのまま利用
- だが、サーバープロセスを被せているので、やりたければどうにでもなる
- 推測するに、ハードウェアを作るつもりだったのではないか

- Erlangで実装されたWebサーバー
- 静的リソース (HTML, 画像など) の提供に関しては特に特徴はない
- 大量のコネクション / セッションをさばけるらしい
- HTML内にインラインで書かれたErlangコードを処理できる (檜山はこれ嫌い)
- 当然ながら、他のErlangアプリケーションと直接的に連携できる
- Erlangアプリケーションに対して、Webからのリクエストの窓口に使える