Integrating Erlang with PHP

Alvaro Videla

Erlang Factory - London - June 2010

About Me

- Lead Developer at TheNetCircle.com
- Blog: http://obvioushints.blogspot.com/
- PHP Erlang Bridge Core Developer
- Twitter: @old_sound

PHP in 5 minutes

PHP in 5 minutes

- PHP: Hypertext Preprocessor

PHP in 5 minutes

- PHP: Hypertext Preprocessor
- Embedded into HTML

PHP in 5 minutes

- PHP: Hypertext Preprocessor
- Embedded into HTML
- Interpreted Language

PHP in 5 minutes

- PHP: Hypertext Preprocessor
- Embedded into HTML
- Interpreted Language
- Supports Both procedural and OOP styles

PHP in 5 minutes

- PHP: Hypertext Preprocessor
- Embedded into HTML
- Interpreted Language
- Supports Both procedural and OOP styles
- Used by Facebook, Yahoo, Wikipedia, more…

PHP in 5 minutes

- PHP: Hypertext Preprocessor
- Embedded into HTML
- Interpreted Language
- Supports Both procedural and OOP styles
- Used by Facebook, Yahoo, Wikipedia, more…
- Tons of Libraries and Open Source Projects

PHP Gotchas

PHP Gotchas

- Stateless

PHP Gotchas

- Stateless
- Requires extra tools to scale:

PHP Gotchas

- Stateless
- Requires extra tools to scale:
  - APC

PHP Gotchas

- Stateless
- Requires extra tools to scale:
  - APC
  - Memcache

PHP Gotchas

- Stateless
- Requires extra tools to scale:
  - APC
  - Memcache
  - MySQL (For every single problem)

PHP Gotchas

- Stateless
- Requires extra tools to scale:
  - APC
  - Memcache
  - MySQL (For every single problem)
- No interprocess communication

PHP Gotchas

- Stateless
- Requires extra tools to scale:
  - APC
  - Memcache
  - MySQL (For every single problem)
- No interprocess communication
- No events

Erlang To The Rescue

PHP Integration

- PHP Erlang Bridge
- C Extension
- Converts PHP into a “cnode”
- Uses Erlang Interface

What can I build?

What can I build?

- Web Admins for Erlang Systems

What can I build?

- Web Admins for Erlang Systems
- PHP Session Storage Systems

What can I build?

- Web Admins for Erlang Systems
- PHP Session Storage Systems
- Ad hoc K/V stores

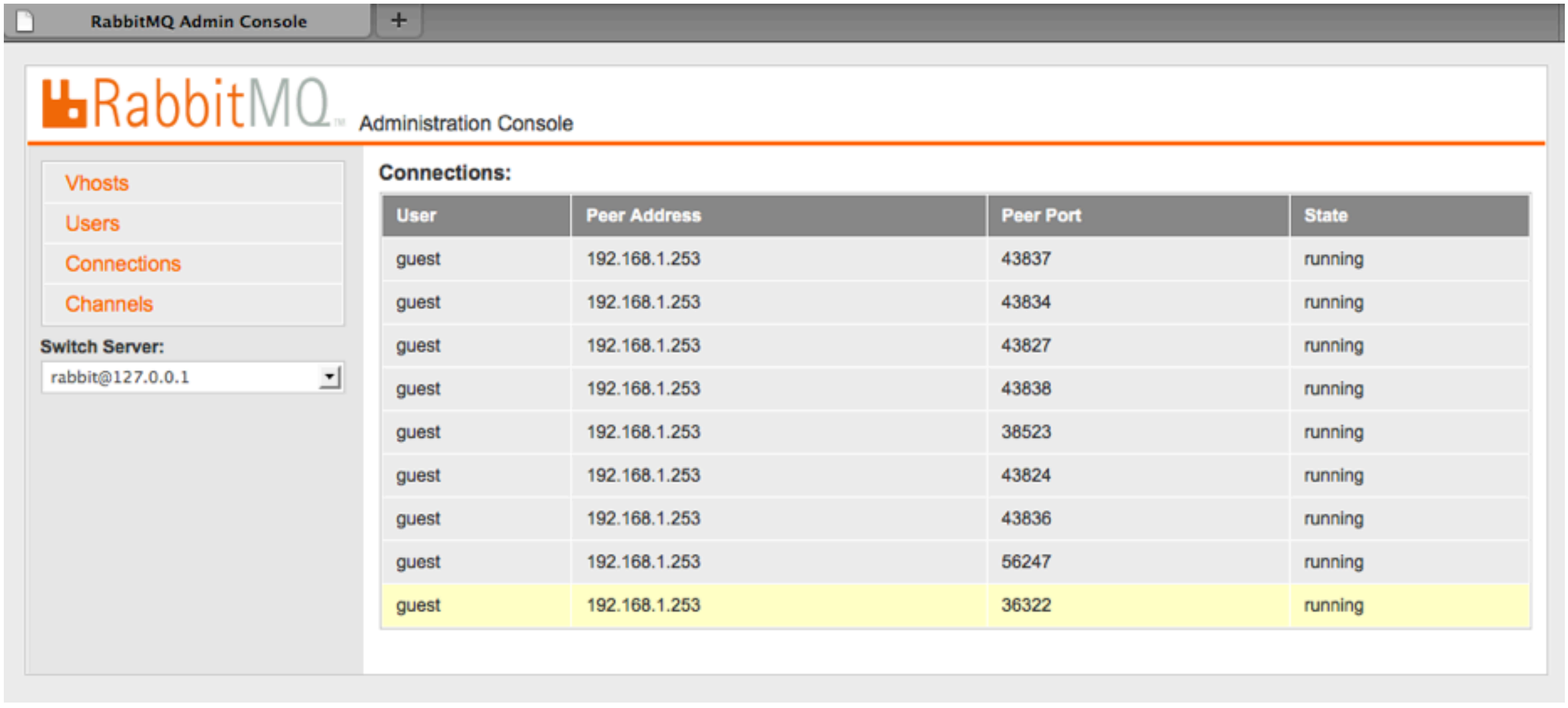
What can I build?

- Web Admins for Erlang Systems
- PHP Session Storage Systems
- Ad hoc K/V stores
- Run Map/Reduce Jobs in Erlang

What can I build?

- Web Admins for Erlang Systems
- PHP Session Storage Systems
- Ad hoc K/V stores
- Run Map/Reduce Jobs in Erlang
- Much more…

RabbitMQ Admin Console



RabbitMQ Admin Console

Connecting To Erlang

```
public function getConnection()
{
  if(!$this->link)
  {
    $this->link = peb_connect(RABBITMQ_HOST, RABBITMQ_COOKIE);
    if(!$this->link)
    {
      throw new Exception("can't connect to Erlang Node");
    }
  }

  return $this->link;
}
```

RabbitMQ Admin Console

Calling Erlang Functions

```
public function list_exchanges($vhost = "/")
{
  $x = peb_encode("[~b]", array(array($vhost)));
  return $this->call('rabbit_exchange', 'list', $x);
}

public function list_bindings($vhost = "/")
{
  $x = peb_encode("[~b]", array(array($vhost)));
  return $this->call('rabbit_exchange', 'list_bindings', $x);
}
```

RabbitMQ Admin Console

Calling Erlang Functions

```
public function call($module, $function, $args)
{
  $result = peb_rpc($module, $function, $args, $this->conn);
  return peb_decode($result);
}
```

Session Storage

Session Storage

- Why?

Session Storage

- Why?
- To ease session data distribution

Session Storage

- Why?
- To ease session data distribution
- Be able to add logic to sessions

Session Storage

- Why?
- To ease session data distribution
- Be able to add logic to sessions
- Several storage backends to choose from

Session Storage: Bitcask

* http://blog.basho.com/2010/04/27/hello,-bitcask/

Session Storage: Bitcask

- Developed by Basho as a Riak backend

* http://blog.basho.com/2010/04/27/hello,-bitcask/

Session Storage: Bitcask

- Developed by Basho as a Riak backend
- Minimalistic API

* http://blog.basho.com/2010/04/27/hello,-bitcask/

Session Storage: Bitcask

- Developed by Basho as a Riak backend
- Minimalistic API
- Stores data on disk

* http://blog.basho.com/2010/04/27/hello,-bitcask/

Session Storage: Bitcask

- Developed by Basho as a Riak backend
- Minimalistic API
- Stores data on disk
- Easy to Backup and Restore

* http://blog.basho.com/2010/04/27/hello,-bitcask/

Session Storage: Bitcask

Custom PHP Session Handler

```
<?php
require_once('./PHPCaskSessionHandler.class.php');

$sh = new PHPCaskSessionHandler('phpcask@localhost', 'erlang_cookie');

session_set_save_handler(
    array($sh,"open"),
    array($sh,"close"),
    array($sh,"read"),
    array($sh,"write"),
    array($sh,"destroy"),
    array($sh,"gc")
);

session_start();
```

http://github.com/videlalvaro/phpcask

Session Storage: Bitcask

Custom PHP Session Handler

```
public function close()
{
  $this->phpcask->close();
}

public function read($session_id)
{
  $data = $this->phpcask->get($session_id);
  return $data[0] == 'not_found' ? '' : $data[0];
}

public function write($session_id, $session_data)
{
  $result = $this->phpcask->put($session_id, $session_data);
  return $result[0] == 'ok';
}
```

Session Storage: Bitcask

Bitcask PHP Client

```
<?php

class PHPCask
{
  public function put($key, $value)
  {
    $x = peb_encode("[~b, ~b]", array(array($key, $value)));
    $result = peb_rpc("phpcask", "put", $x, $this->link);
    return peb_decode($result);
  }

  public function get($key)
  {
    $x = peb_encode("[~b]", array(array($key)));
    $result = peb_rpc("phpcask", "get", $x, $this->link);
    return peb_decode($result);
  }
}
```

Session Storage: Bitcask

Erlang Bitcask API Wrapper

```
handle_call({get, Key}, _From, #state{ref=Bitcask}=State) ->
  Reply =
  case bitcask:get(Bitcask, Key) of
    {ok, Value} -> binary_to_term(Value);
    _ -> not_found
  end,
  {reply, Reply, State};

handle_call({put, Key, Value}, _From, #state{ref=Bitcask}=State) ->
  Reply = bitcask:put(Bitcask, Key, term_to_binary(Value)),
  {reply, Reply, State};

handle_call({delete, Key}, _From, #state{ref=Bitcask}=State) ->
  Reply = bitcask:delete(Bitcask, Key),
  {reply, Reply, State};
```

Ad Hoc K/V Store

Ad Hoc K/V Store

- Using ETS Tables

Ad Hoc K/V Store

- Using ETS Tables
- Adapt the API to suit your needs

Ad Hoc K/V Store

- Using ETS Tables
- Adapt the API to suit your needs
- Easiest way to build a web counter

Installing the extension

http://code.google.com/p/mypeb/

Installing the extension

- grab the code

http://code.google.com/p/mypeb/

Installing the extension

- grab the code
- phpize

http://code.google.com/p/mypeb/

Installing the extension

- grab the code
- phpize
- ./configure

http://code.google.com/p/mypeb/

Installing the extension

- grab the code
- phpize
- ./configure
- make

http://code.google.com/p/mypeb/

Installing the extension

- grab the code
- phpize
- ./configure
- make
- make install

http://code.google.com/p/mypeb/

Hello Erlang

```
% erl -sname hello -set_cookie demo
-module(hello).
-export([start/0, hello/0]).

start() ->
  MyPid = spawn(hello, hello, []),
  register(hello, MyPid).

hello() ->
  receive
    {Pid, Msg} ->
      Pid ! {self(), welcome},
      io:format("Hello ~s.~n", [Msg]),
      hello();
    quit ->
      ok;
    X ->
      io:fwrite( "Got ~p.~n", [ X ] ),
      hello()
    end.
```

Hello Erlang

```
<?php

$link = peb_connect('hello@127.0.0.1', 'demo');

if(!$link) die ("Can't Connect\n");

$msg = peb_vencode('{~p, ~s}', array(
                                      array($link, $argv[1])
                                     ));

peb_send_byname('hello', $msg, $link);

$message = peb_receive($link);
$rs = peb_vdecode($message);

print_r($rs);

peb_close($link);
?>
```

Connecting to erlang

- peb_connect($host, $cookie);

Sending messages

- peb_send_by_name($node, $msg, $link);
- peb_send_by_pid($node, $msg, $link);

Encoding messages

- peb_vencode($format, $data);
- peb_encode($format, $data);

Encoding Examples

```
<?php
//encode a atom:
$msg = peb_encode('~a', array(
                                'hello'
                                ));
//encode a double
$msg = peb_encode('~d', array(
                                3.1415926
                                ));
//encode a list with two element
$msg = peb_encode('[~a,~d]', array(
                                 array( 'hello', 3.1415926)
                                ));
//encode a list with two element:
$msg = peb_encode('{~a,~i}', array(
                                 array( 'hello', 1234 )
                                ));
?>
```

Formatting Characters

Formatting Characters

- [] List

Formatting Characters

- [] List
- {} Tuple

Formatting Characters

- [] List
- {} Tuple
- ~a Atom

Formatting Characters

- [] List
- {} Tuple
- ~a Atom
- ~s String

Formatting Characters

- [] List
- {} Tuple
- ~a Atom
- ~s String
- ~b Binary

Formatting Characters

- [] List
- {} Tuple
- ~a Atom
- ~s String
- ~b Binary
- ~i Integer

Formatting Characters

- [] List
- {} Tuple
- ~a Atom
- ~s String
- ~b Binary
- ~i Integer
- ~d Double

Formatting Characters

- [] List
- {} Tuple
- ~a Atom
- ~s String
- ~b Binary
- ~i Integer
- ~d Double
- ~p Pid

Receiving Messages

- peb_receive($link);

Decoding Messages

- peb_vdecode($msg);
- peb_decode($msg);

Decoding Messages

Decoding Messages

- String, Atom, Binary -> String

Decoding Messages

- String, Atom, Binary -> String
- Tuple, List -> Array

Decoding Messages

- String, Atom, Binary -> String
- Tuple, List -> Array
- Pid -> Resource

Decoding Messages

- String, Atom, Binary -> String
- Tuple, List -> Array
- Pid -> Resource
- Integer -> Integer

Decoding Messages

- String, Atom, Binary -> String
- Tuple, List -> Array
- Pid -> Resource
- Integer -> Integer
- Float -> Float

RPC Calls

- peb_rpc($node, $module, $function, $msg);
- peb_rpc_to($node, $module, $function, $msg);

TODO

- Simpler API for encoding/decoding
- Cleaner extension installation for several platform
- Better Testing/Benchmarks

Nice To Have…

- PHP for the Frontend
  - Templating
  - MVC
- Erlang APIs for Backend
  - DB access
  - Cache
  - Session Storage
  - more…

Resources

- PHP Erlang Bridge: http://code.google.com/p/mypeb/
- Erlang Website: http://www.erlang.org/
- Online Book: http://learnyousomeerlang.com/
- Community Site http://trapexit.org/
- Conferences: http://www.erlang-factory.com/

Questions?

Thanks!

Alvaro Videla

http://twitter.com/old_sound

TheNetCircle.com