

# Assurer

サーバテスト/モニタリングフレームワーク

(株)paperboy&co.

宮下 剛輔 (mizzy)

<http://mizzy.org/>

# Assurerとは？

- ▶ 「アシュラ」と読みます
- ▶ サーバのテストや監視を行うツール兼フレームワーク
- ▶ 名前の由来
  - テストのことをQuality Assuranceと言ったりする
  - Plaggerライクなツールはみんな「er」で終わる（Archer, Observerなど）

# 開発の動機など

- ▶ サーバ構築後の動作確認を自動化したい
  - 元は単発のテストが目的
- ▶ 出力フォーマットや出力先をプラグインで柔軟に変更できるようにしたい
  - それPla

# テストツールから監視ツールへ

- ▶ テストも監視も似たようなもの？
  - やってみると結構違う
  - 特に並列処理やリトライなどは、単発テストでは必要ない ⇒ めんどくさい
  - でもやるって言った以上ひっこみがかからない

# 実行例

```
ck
miya@ubuntu$ ./assurer.pl -c examples/config.yaml [~/svk/Assurer]
Assurer::Plugin::Test::HTTP [info] Testing HTTP Test on svn.mizzy.org...
Assurer::Plugin::Test::HTTP [info] Testing HTTP Test on trac.mizzy.org...
Result of HTTP Test on svn.mizzy.org
ok 1 - HTTP status code of http://svn.mizzy.org:80/ is 200
ok 2 - Content of http://svn.mizzy.org:80/ matches 'It works!'

Result of HTTP Test on trac.mizzy.org
not ok 1 - HTTP status code of http://trac.mizzy.org:80/ is 200
not ok 2 - Content of http://trac.mizzy.org:80/ matches 'It works!'

miya@ubuntu$ [~/svk/Assurer]
```

# 設定サンプル

## global:

log:  
  level: debug  
retry: 3  
interval: 3

## hosts:

web:  
  - www0.mizzy.org  
  - www1.mizzy.org  
db:  
  - mysql0.mizzy.org  
mail:  
  - smtp.mizzy.org

## test:

- module: HTTP  
  name: HTTP Test  
  config:  
    content: It works!  
  role: web

## format:

- module: Text

## publish:

- module: Term

# global – グローバルな設定

```
global:  
  log:  
    level: debug  
  retry: 3  
  interval: 3
```

- ▶ ログレベル
- ▶ リトライ数のデフォルト値
- ▶ リトライ間隔のデフォルト値

# hosts – ホストとロール

hosts:

**web:**

- www0.mizzy.org
- www1.mizzy.org

**db:**

- mysql0.mizzy.org

**mail:**

- smtp.mizzy.org

- ▶ テスト/監視対象のホストをロール別に設定

# test – テスト実行フェーズ

test:

```
- module: HTTP
  name: HTTP Test
  config:
    content: apache
  role: web
```

- ▶ module – 使用するテストプラグイン
- ▶ name – テストの名前
- ▶ config – プラグイン用設定
- ▶ role – テスト対象ホストのロール

# format – テスト結果フォーマットフェーズ

format:

- module: Text

- ▶ module – 使用するフォーマットプラグイン
- ▶ config – プラグイン用設定

# publish – テスト結果出力フェーズ

publish:

- module: File

config:



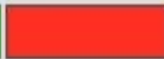












path: /file.txt

- ▶ module – 使用する出力プラグイン
- ▶ config – プラグイン用設定

# 実行例その2

## Format::HTML + Publish::File

TAP Matrix - Sun Jan 14 06:13:24 2007 GMT

Test file	Test cases						%
Apache テスト #0 FAILED							71.43%
Apache テスト #1 FAILED							75.00%
Apache テスト #2 OK							100.00%
Apache テスト #3 FAILED							0.00%
<b>TOTAL</b>	<b>4 files</b>	<b>16 test cases: 11 ok, 5 failed, 0 todo, 0 skipped and 0 unexpectedly succeeded</b>					<b>68.75%</b>

# formatフィルタ

format:

- module: Text

filter:

module: Status

status: not ok

- ▶ formatフェーズで動作
- ▶ 特定の条件にマッチするテスト結果のみ抜き出してフォーマットする

# publishフィルタ

publish:

- module: Term

filter:

module: Type

type: text/plain

- ▶ publishフェーズで動作
- ▶ 特定の条件にマッチするフォーマットのみ出力する

# フィルタの利用例

- ▶ 失敗したテストのみをIRCで通知
- ▶ テキストフォーマットとHTMLフォーマット両方を生成して、テキストフォーマットはターミナルに表示、HTMLフォーマットはファイルに保存

# シェルモード

```
$ assurer.pl --config=config.yaml --shell --role=web
```

```
shell> free
```

```
[www0.mizzy.org]
```

	total	used	free	shared	buffers	cached
Mem:	386660	351336	35324	0	50700	94516
-/+ buffers/cache:		206120	180540			
Swap:	1177460	327356	850104			

```
[www1.mizzy.org]
```

	total	used	free	shared	buffers	cached
Mem:	386660	351336	35324	0	50700	94516
-/+ buffers/cache:		206120	180540			
Swap:	1177460	327356	850104			

```
shell>
```

# テストプラグインの中身

```
package Assurer::Plugin::Test::Ping;

use strict;
use warnings;
use base qw( Assurer::Plugin::Test );
use Assurer::Test;
use Net::Ping;

sub register {
    my $self = shift;
    $self->register_tests( qw/ ping / );
}
```

# プラグインの中身(つづき)

```
sub ping {  
  my ( $self, $context, $args ) = @_;  
  
  my $conf = $self->conf;  
  
  # ... 中略  
  
  my $ping = Net::Ping->new($protocol);  
  $ping->hires(1);  
  my ($ret,$duration,$ip) = $ping->ping($host, $timeout);  
  ok( $ret, "ping ok $host, $ret, $duration, $ip");  
}
```

# Test::Moreライクなテストメソッド

- ▶ Assurer::Testからエクスポート
  - `ok( $got eq $expected, $name );`
  - `is( $got, $expected, $name );`
  - `like( $got, $expected, $name );`
  - ...

# 今後の予定

- ▶ Notifyフェーズ
- ▶ 並列処理をもっとインテリジェントに
- ▶ リモート分散並列処理
- ▶ Nagiosとの連携 (Nagiosプラグイン, NRPE)
- ▶ 他いろいろ

# 情報源など

- ▶ <http://assurer.jp/trac>
- ▶ <http://mizzy.org/>
- ▶ [#assurer@chat.freenode.net](http://chat.freenode.net)
- ▶ YAPC::Asia Tokyo 2007 でもしゃべります

ご清聴ありがとうございました！