PROVE for PHP

http://www.phprove.com/

PROVE for PHPはバージョンアップ、

リグレッションテストを大幅に効率化！

自己紹介

- 大垣 靖男
  メール yohgaki@ohgaki.net
  ツイッター他のID yohgaki
  岡山県在住
- エレクトロニック・サービス・イニシアチブ有限会社
  取締役 社長
  http://www.es-i.jp/
- 岡山大学大学院非常勤講師、
  PHP技術者認定機構顧問、他

PHPとアプリのバージョンアップ

セキュリティ

PHPとアプリのバージョンアップ

バグ

PHPとアプリのバージョンアップ

バージョンアップ

セキュリティ対策

- フレームワークを利用
- セキュリティ規約を策定・順守
- 設計のレビュー・ベストプラクティス
- ソースコード検査（ホワイトボックス検査）
- ブラックボックス検査
- **セキュリティフィックスの適用**

バグ対策

- テストコードの作成
- UNITテスト
- **バグフィックスの適用**

バージョンアップ

- PHP本体のバージョンアップ
- フレームワークのバージョンアップ
- アプリケーションのバージョンアップ
- **システムはバージョンアップしていく**

バージョンアップ効率化

業務の効率化

サービス提供の
高速化・高品質化!

バージョンアップの必要性

例えば、PCIDSSでは

PCI DSS 要件

6.1 すべてのシステムコンポーネントとソフトウェアに、ベンダ提供の最新セキュリティパッチを適用する。**重要なセキュリティパッチは、リリース後 1 カ月以内にインストール**する。

注: 組織は、パッチインストールの優先順位を付けるために、リスクに基づくアプローチの適用を検討できる。たとえば、重要なインフラストラクチャ（一般に公開されているデバイス、システム、データベースなど）に重要性の低い内部デバイスよりも高い優先順位を付けることで、**優先順位の高いシステムおよびデバイスは 1 カ月以内**に対処し、重要性の低いシステムおよびデバイスは 3 カ月以内に対処するようにする。

PHP本体のバージョンアップ

- UNIT TEST!
- PHP本体のテストは
  make test で実行できる。

```
$ find . -name "*.phpt" | wc -l

11194
```

不安

というより

業務としては不十分

PHP本体のバージョンアップ

プログラマの友 - Diff!

PHP 5.3.2 → PHP 5.3.3

$ diff -ur php-5.3.2 php-5.3.3 | wc -l

166382

```
-/* $Id: zend_ini_parser.y 293154 2010-01-05 20:40:23Z sebastian $ */
+/* $Id: zend_ini_parser.y 300737 2010-06-24 22:32:42Z felipe $ */

 #define DEBUG_CFG_PARSER 0

@@ -304,7 +304,7 @@
 ;

 section_string_or_value:
-               var_string_list                 { $$ = $1; }
+               var_string_list_section         { $$ = $1; }
        |       /* empty */                     { zend_ini_init_string(&$$); }
 ;

@@ -326,6 +326,15 @@
        |       /* empty */                     { zend_ini_init_string(&$$); }
 ;

+var_string_list_section:
+               cfg_var_ref                             { $$ = $1; }
+       |       constant_literal                        { $$ = $1; }
+       |       '"' encapsed_list '"'                   { $$ = $2; }
+       |       var_string_list_section cfg_var_ref     { zend_ini_add_string(&$$, &$1, &$2); free(Z_STRVAL($2)); }
+       |       var_string_list_section constant_literal        { zend_ini_add_string(&$$, &$1, &$2); free(Z_STRVAL($2)); }
+       |       var_string_list_section '"' encapsed_list '"'  { zend_ini_add_string(&$$, &$1, &$3); free(Z_STRVAL($3)); }
+;
+
 var_string_list:
```

バージョンアップは
難しい

Webアプリの動作確認は難しい

- 同一条件の再現
- IPアドレス　(REMOTE_ADDR, X-FORWAREDE-FOR)
- HTTPヘッダ、クッキー、データベース、キャッシュ、Webサービス、メール、etc

仮に同じ条件を揃えたとして

どう確認するのか？

Webアプリのシステム構成は複雑

Chrome　Firefox　IE8　Safari

IE6　IE7　Opera

Android　iPhone　iPad　ケータイ

Proxy　Proxy　Proxy

Web Server　Web Server　Web Server

DB Server　DB Server

Webアプリの動作確認は難しい

- **マルチリンガルサイトのバージョンアップ**
  - 例えば、Drupal。観光サイトなどで日本語、英語、フランス語、スペイン語、ポルトガル語、韓国語、簡体中国語、繁体中国語をサポートするサイトのモジュールをバージョンアップした場合に正しく表示される？

Webアプリの動作確認は難しい

- **ケータイサイトのバージョンアップ**
- フレームワーク部分の修正を行った。
  サポートする全ての機種で問題なくいままで通りに表示されるか？

PROVE for PHP

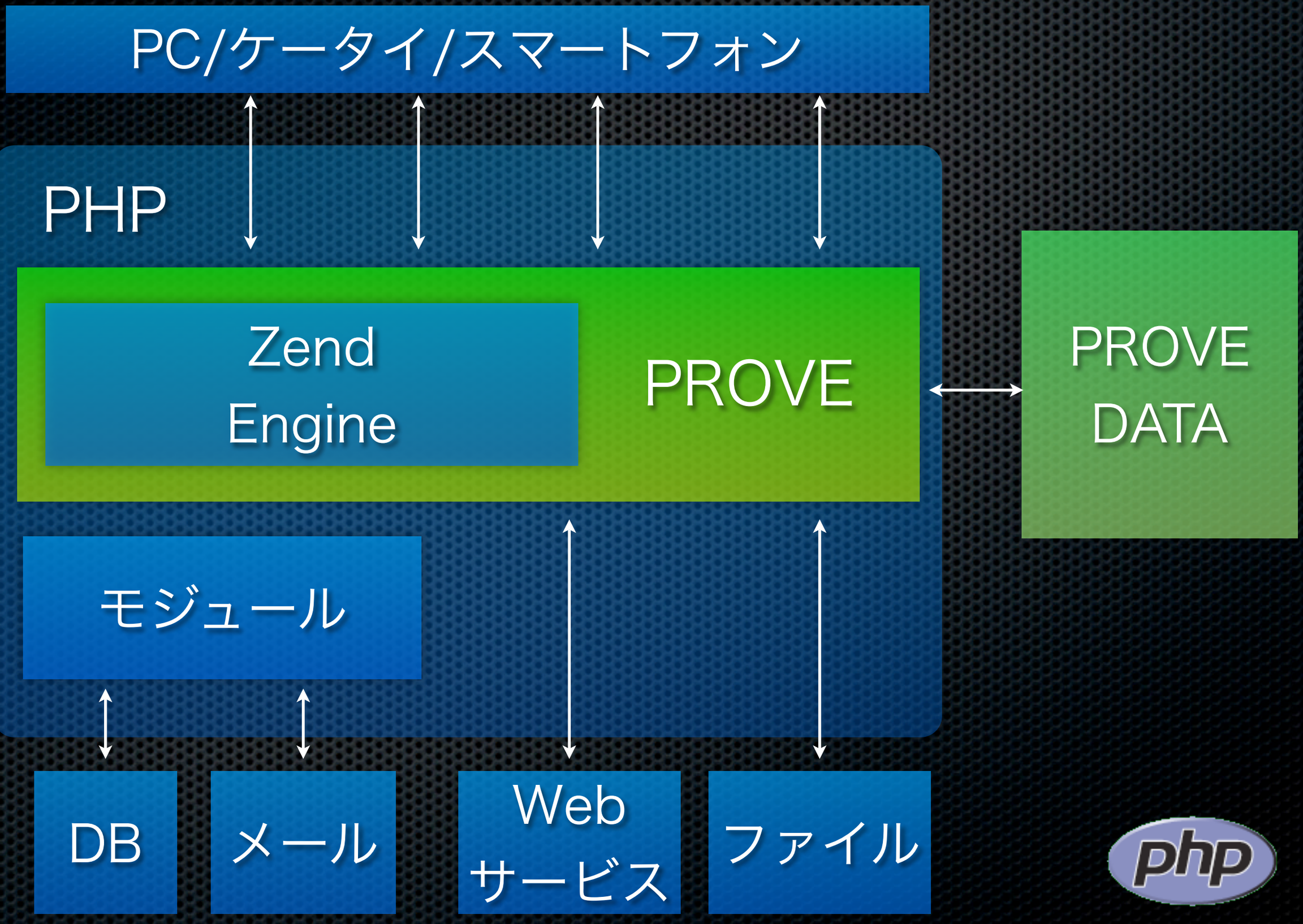
PROVE for PHPとは？

- リグレッションテストツール

PROVE for PHPとは？

- アプリケーションレベルのテストスイート
- 特にバージョンアップ時のテストに有効
- サポート予定のPHP
  PHP 4.3/5.1/5.2/5.3

PROVE for PHPのアーキテクチャ



PROVE for PHPとは？

- 本当に実働アプリケーションの
テストが可能

PROVE for PHPとは？

TRACE
モード

リクエスト
関数呼び出し
データベース
ファイル
を記録

PROVE
データベース

VALIDATION
モード

リクエスト
関数呼び出し
データベース
ファイル
を検証・**置き換え**

PROVE for PHPとは？

- Zend Engineモジュールとして動作
  - あらゆる関数呼び出しを監視
  - 必要に応じてオーバーライド
- IPアドレスも何もかも再現
  - ファイルI/O
  - データベースアクセス
- オーバーライドする関数は指定可能

PROVE for PHPとは？

十分な性能で動作するため

実働アプリケーションから

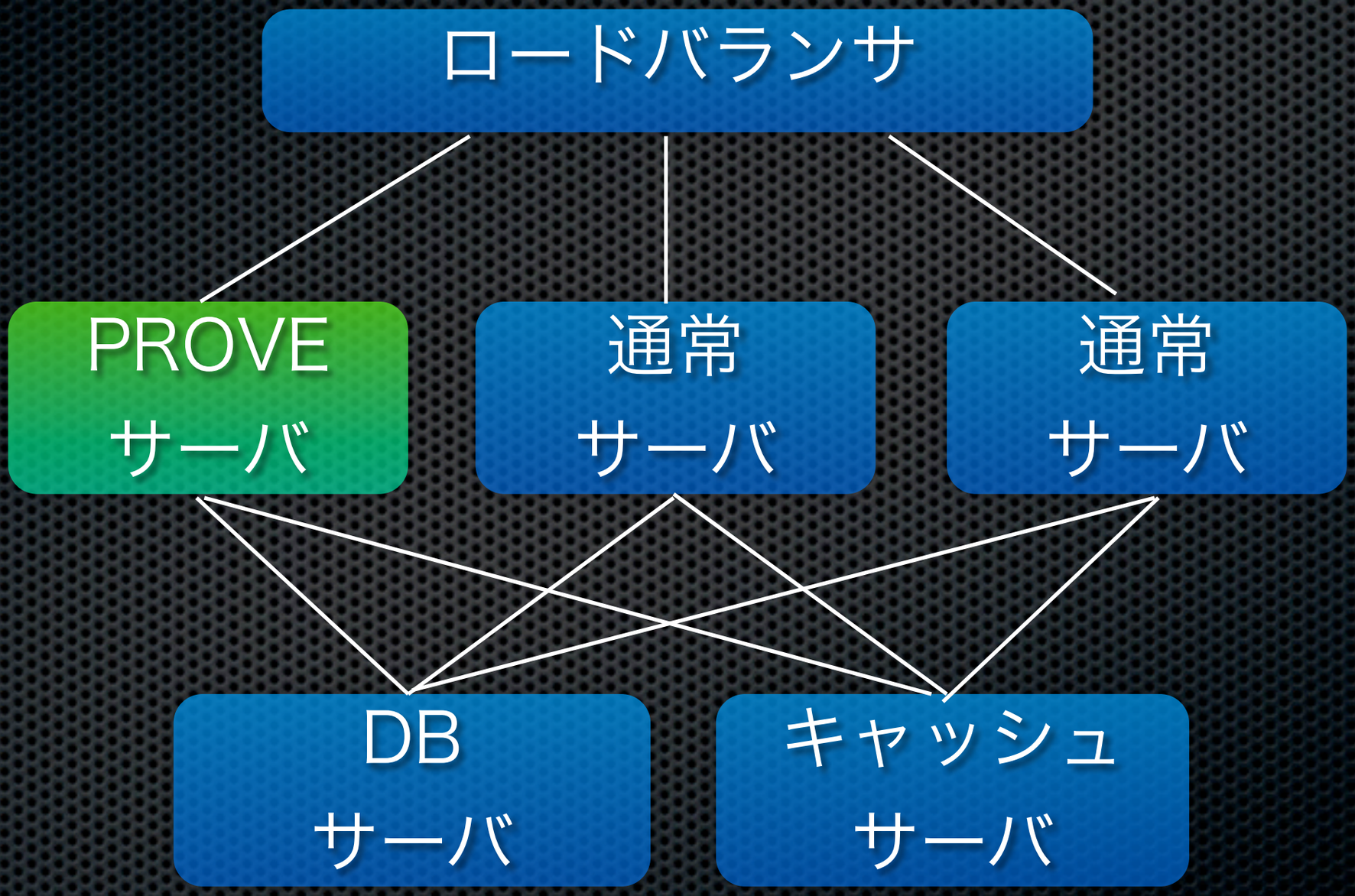
テストケースを作成可能

PROVE for PHPとは？

もちろんスケールアウトしている

大規模システムでも利用可能

PROVE for PHPの実行環境例



PROVE for PHPとは？

- ロードバランサはUltraMonkeyやバランシングをサポートしているnginxなどのオープンソース製品で十分
- UltraMonkey-L7, HAProxy, LXBなど
- クッキーパーシステンスをサポートしていると好都合

関数オーバライドとテストの信頼性

- TRACEモードでは引数と戻り値を保存
- VALIDATIONモードでは引数の検証と戻り値のオーバライド
- オーバーライド関数は利用者が指定
- このような**単機能テストはUNITテスト**
  の得意分野

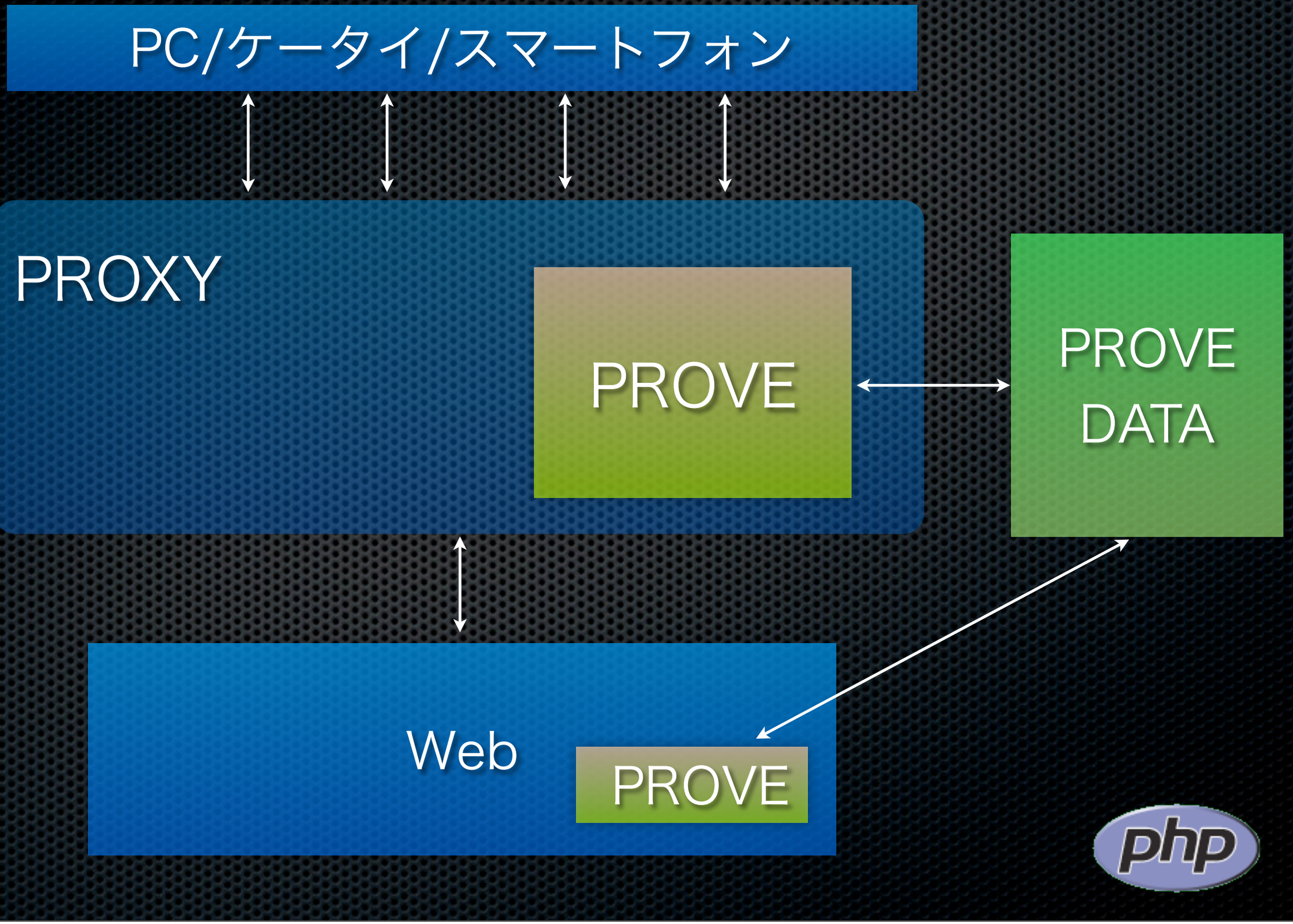
追加予定機能一覧

- バリデーションモードのWebベースGUI
- ソースコードカバレッジ分析
- オーバライド関数の追加
- PROVE for ANY (powered by PHP)
  - プロキシ型 PROVE。PROVE for PHP の簡易版
- PROVE for PHP/SQL
  - SQLクエリトレーサー

PROVE for ANY

- プロキシ型のPROVE
  - リバースプロキシとして動作
- PROVE for PHPの欠点を補完
- APCなどのZendモジュールを利用した環境でも利用可能
- システムから完全に独立させる事も可能

PROVE for ANYのアーキテクチャ



PROVE for ANY

- システムから完全に独立
- 他の言語でも利用可能
- データベース、ファイル、時間、IPアドレスなど、同じ条件となるようにするのは利用者の責任
- PHPならPROVEを通常モジュール動作&PROVEデータを利用可能

PROVE for SQL/PHP

- SQLクエリを口グするシンプルなツール
- テスト用ではなくセキュリティ対策用
- SQLクエリとPHP変数（例えば$_SESSION[‘user_id”]）を一緒にダンプ
- 監査用口グ取得がアドホックで可能

PROVE for SQL/PHPのアーキテクチャ

PC/ケータイ/スマートフォン

PHP

PostgreSQL/MySQLモジュール

PROVE

PROVE DATA

データベース
MySQL/PostgreSQL/SQLite/
etc

PROVE for SQL/PHP

- 運用サーバで常時利用が可能となるよう、軽量なモジュール
- セキュリティ用ログとして有用なログを記録可能
- システムを管理するスタッフなどの不正防止（個人情報盗み見）などにも有効

PROVE の カバレッジ分析

- APD, Xdebugのカバレッジ分析は実行したコード（行番号）のみが分かる。
- PROVEのVALIDATIONモードではパフォーマンスは求められない。コードがある行と実際に実行した行でカバレッジを計算する。

PROVE の カバレッジ分析

APD/Xdebug

```
 1  <html>
 2  <head></head>
 3  <body>
 4  <?php
 5  if ($_GET[‘a’]) {
 6      echo ‘a’;
 7  } else {
 8      echo ‘b’;
 9  }
10  ?>
11  </body></html>
```

PROVE

```
 1  <html>
 2  <head></head>
 3  <body>
 4  <?php
 5  if ($_GET[‘a’]) {
 6      echo ‘a’;
 7  } else {
 8      echo ‘b’;
 9  }
10  ?>
11  </body></html>
```

PROVE の カバレッジ分析

- 正確なカバレッジが分れば、よりバージョンアップに自信が持てる
- カバレッジ分析により、バージョンアップした環境で以前の状態と同じ動作をすることが確認できる

これでもまだPHP4を使いたい方

- **SRA OSS** の
  「**PHP4 セキュリティ保守サービス**」
  の利用をお勧めします。
- パッチ数は30以上、うちいくつかはクリティカルで**リモートからの攻撃が可能**
- RHELでも未修正

サポート

PHP4 セキュリティ保守サービス

本サービスは Web サーバ用途にオープンソースのスクリプト言語 PHP を利用されている Web サイト向けに、PHP バージョン 4 に対する保守サービスを提供するものです。

» 本サービスで提供しているパッチ情報（例）は こちら

サービス仕様書を改定いたしました。

短期契約も承ります！
契約終了後も（以降のリリースは得られませんが）、その時点までのパッチセットはご利用いただけます。

1 セット分の契約でご利用できる範囲が広がりました！
1 担当者が窓口であれば、自社あるいは同一のエンドユーザ様むけに提供されている複数システムに適用できます。

PHP4 のメンテナンスが終了

PHP コミュニティより、PHP 4.x の開発停止とメンテナンス終了が発表されています。 2007 年 12 月 31 日にて、PHP 4.x のすべての開発を終了し、その後は、大きなセキュリティホールがあればケースバイケースでリリースを行います。 そして、2008 年 8 月に完全にメンテナンスを終了となりました。 PHP 5.x への移行が推奨されています。

セキュリティパッチがリリースされない？

現状すでに PHP5 で修正されているいくつかのセキュリティホールが PHP4 については修正が発表されていません。 これらはクラッカーによる攻撃のターゲットになる恐れがあります。

PHP4 未修正の問題
- PostgreSQL エスケープ関数の問題
- 変数リファレンスカウントの問題

他...

PROVE for PHPのライセンス

- 商用ライセンス
  - ビジネスモデルの策定はこれから
  - サブスクリプション型
  - 個人ユースに無償で提供
  - レガシーバージョン、再販売用のライセンスは別途

PROVE for PHPのお問い合わせ

- エレクトロニック・サービス・イニシアチブ有限会社
- info@es-i.jp
- http://www.es-i.jp/

PROVEの画面とデータ

IPv6 Support	enabled
Registered PHP Streams	php, file, data, http, ftp
Registered Stream Socket Transports	tcp, udp, unix, udg
Registered Stream Filters	string.rot13, string.toupper, string.tolower, string.strip_tags, co consumed

This program makes use of the Zend Scripting Language Engine:
Zend Engine v2.2.0, Copyright (c) 1998-2010 Zend Technologies
with PROVE for PHP v0.0.1-
dev, Electronic Service Initiative, Ltd All Rights Reserved, by Yasuo Ohgaki

PHP Credits

prove

PROVE for PHP	enabled

Directive	Local Value	Master Value
prove.override_functions	time microtime date	time microtime date
prove.prove_mode	1	1
prove.trace_dir	/tmp/prove-trace/	/tmp/prove-trace/
prove.validate_dir	/tmp/prove-validate/	/tmp/prove-validate/

```
/prove-0012852981*
 /tmp/prove-trace/prove-0012852981298129.148183-1403
 /tmp/prove-trace/prove-0012852981298131.772474-1404
/tmp/prove-trace/prove-0012852981298135.149234-1405
/tmp/prove-trace/prove-0012852981298139.183980-1406
```

```php
<?php
$id ='001285298139.183980-1406';
$prove_data=array(
 'request_info'=>array(
  'request_method'=>"",
  'query_string'=>"",
  'post_data'=>"",
  'raw_post_data'=>"",
  'raw_post_data'=>"",
  'cookie_data'=>"",
  'content_length'=>0,
  'post_data_length'=>0,
  'raw_post_data_length'=>0,
  'path_translated'=>"ttt.php",
  'request_uri'=>"",
  'auth_user'=>"",
  'auth_password'=>"",
  'auth_digest'=>""
 ),
 '_POST'=>array (
),
```

```
$prove_output = 'YXJyYXkoMjgpIHsKICBbIk1BTlBBVEgiXT0+CiAgc3RyaW5n
yZS9tYW46L3Vzci9YMTFSNi9tYW4iCiAgWyJURVJNX1BST0dSQU0iXT0+CiAgc3Ry
gInh0ZXJtLWNvbG9yIgogIFsiU0hFTEwiXT0+CiAgc3RyaW5nKDkpICIvYmluL2Jh
rODRBeFowRWxTbGFBMXVnOVExVlUrKytUSS8tVG1wLS8iCiAgWyJBcHBsZV9QdWJT
VQTMvUmVuZGVyIgogIFsiVEVSTV9QUk9HUkFNX1ZFUlNJT04iXT0+CiAgc3RyaW5n
pYi9xdDMiCiAgWyJPTERQV0QiXT0+CiAgc3RyaW5nKDE0KSAiL1VzZXJzL3lvaGdh
fTU9ERSJdPT4KICBzdHJpbmcoOCkgInVuaXgyMDAzIgogIFsiU1NIX0FVVEhfU09D
gIFsiX19DRl9VU0VSX1RFWFRfRU5DT0RJTkciXT0+CiAgc3RyaW5nKDEwKSAiMHgx
9PgogIHN0cmluZygxNzYpICIvb3B0L2xvY2FsL2Jpbjovb3B0L2xvY2FsL3NiaW46
uOi91c3IvbG9jYWwvc2JpbjovdXNyL2JpbjovdXNyL3NiaW46L2Jpbjovc2Jpbjov
bIkJMT0NLU0laRSJdPT4KICBzdHJpbmcoMSkgImsiCiAgWyJQV0QiXT0+CiAgc3Ry
KICBzdHJpbmcoNSkgImVtYWNzIgogIFsiTEFORyJdPT4KICBzdHJpbmcoMTEpICJc
TSExWTCJdPT4KICBzdHJpbmcoMSkgIjEiCiAgWyJIT01FIl09PgogIHN0cmluZygx
hY3MgKyVkJyIKICBbIkxPR05BTUUiXT0+CiAgc3RyaW5nKDcpICJ5b2hnYWtpIgog
gc3RyaW5nKDI2KSAiL3RtcC9sYXVuY2gteXBncEx5L29yZy540jAiCiAgWyJHSVRf
fIl09PgogIHN0cmluZyg1KSAiLi9waHAiCn0K';
```

```
$fcall_trace[]=array (
   'function' => 'main',
   'filename' => '/Users/yohgaki/php-5.2.14/ttt.php',
   'linenum' => 0,
);
$fcall_trace[]=array (
   'function' => 'foo',
   'filename' => '/Users/yohgaki/php-5.2.14/ttt.php',
   'linenum' => 13,
);
$fcall_trace[]=array (
   'function' => 'bar',
   'filename' => '/Users/yohgaki/php-5.2.14/ttt.php',
   'linenum' => 9,
);
$fcall_trace[]=array (
   'function' => 'debug_backtrace',
   'filename' => '/Users/yohgaki/php-5.2.14/ttt.php',
   'linenum' => 4,
);
```

- ログ形式はPHP配列形式（PHPで簡単に処理可能）
- 集取されたデータは変換ツールを使い、SQLiteデータベースへ変換される
- SQLiteを利用する事によりVALIDATIONモードで名前空間を綺麗に保てる
- SQLiteデータベースエンジンはPROVEモジュールに組み込み

PHP技術者認定機構の紹介

PHP技術者認定機構とは

- PHPエンジニアのスキルを認定する公正・中立なNPO法人
- できる限り安価な仕組み（公式教材が一般書籍）
- ベータ試験を今週開催
- これからWeb業界に入る方向け
  - 特に学生、PHPの実務経験が無い方向け
- 上級、セキュリティなどの認定試験も将来実施予定

ご清聴ありがとうございました