Zend
The *php* Company

PHP 6 & The PHP Collaboration Project
Present & Future

Zeev Suraski
Co-founder & CTO, Zend Technologies
Co-architect of PHP

PHP 6 Key Improvements

Zend
The php Company

- Unicode support
- Polish up OO including __toString() support everywhere
- Improved SOAP support? (WS-Security, file attachments)
- Remove legacy junk such as register_globals & safe_mode
- The future will tell…

Internationalization

- **A definition for i18n:**
  - The effort to make the Web usable to everyone regardless of their preferred language or geographic location through the effort of making all technologies operable across all languages and writing systems.
- **PHP will support Unicode standard**
  - YAHOO! & Zend leading effort
- **Use the IBM ICU library:**
  - provides an open, flexible, portable foundation for applications to use for their software globalization requirements.
- **Bulk of core work already completed**

Unicode String Literals

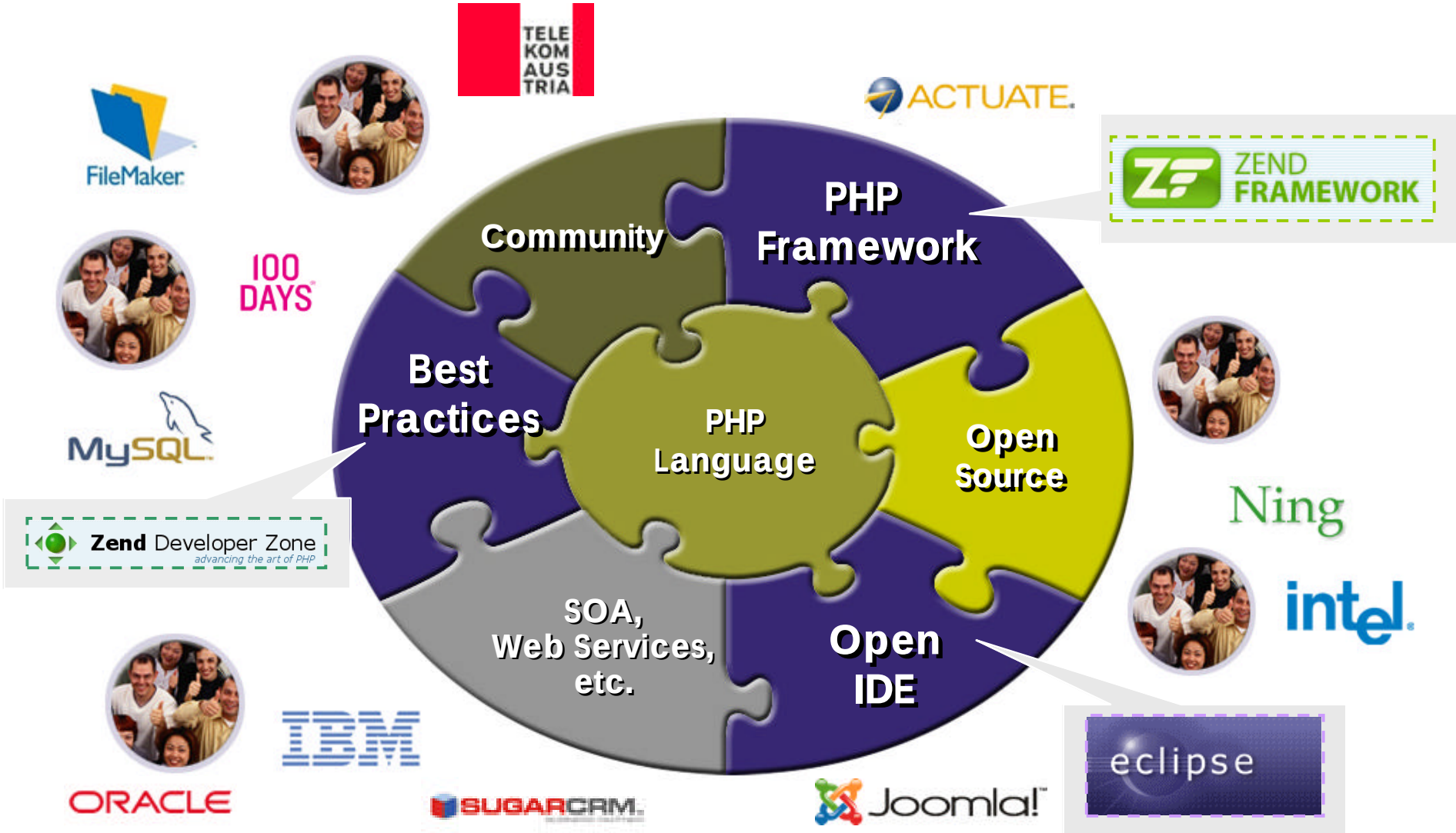
- **With unicode_semantics=on, string literals are of Unicode type**
- **1 character may be > 1 byte**

```
// unicode_semantics = on
$str = "hello world"; // Unicode
echo strlen($str);    // result is 11


$jp = "検索オプション"; // Unicode
echo strlen($str);    // result is 7
```

- **To obtain length in bytes one would use a separate function**

PHP Collaboration Project



What do we need?

Zend
The php Company

- **SOA & Web Services:**
  - Finally software re-use truly seems to work. And not in its previous copy-on-write incarnation.
  - Software re-use works across companies.
  - Increasingly important to support all (popular) forms of Web Services
- **Data is (still) King:**
  - Need to support all data (relational, XML, legacy)
  - Data needs to be searchable

What do we need?

Zend
The php Company

- **Rich Interfaces:**
  - The world has acknowledged the deployment advantages of the browser.
  - Ajax is a response to the need for a richer and easily deployable interface in the browser.
- **Continue the low barrier to entry:**
  - Continue building on PHP's simplicity which has made it a success.
  - Open-source's distribution advantages enables companies to start playing and building on top of technology without a relationship with vendors.

What do we need?

Zend
The php Company

- **Need for Best Practices:**
  - It is not trivial to architect, build and deploy stable, secure and high-performing three tier Web apps (Javascript → PHP → DB)
- **RAD:**
  - Time-to-market on the Web is becoming increasingly critical.

The Importance of Simplicity

Zend
The php Company

> "Things should be made as simple as possible, but no simpler.“
>
> -- Albert Einstein

- **Simpler is easier to use.**
- **Simpler is more stable, and less prone to error.**
- **Simpler is more compatible.**
- **Simpler is easier to maintain.**

Plagiarized from http://www.pantos.org/atw/35504.html

Guiding Principles

Zend
The php Company

- **Keep it "extremely simple" – stick to 20%/80% rule and compensate by:**
  - Extensibility
  - Use-at-will architecture
  - Configuration-less
- **Cherry pick best-of-breed ideas**
- **Showcase current trends in Web development (Web Services, Ajax, Search, Syndication, …)**
- **Document development with use-cases**
- **Only high quality and necessary components**
- **Play nicely with others**

How does the PHP Collaboration Project fit in?

Zend
The php Company

- **SOA & Web Services:**
  - Zend Framework has a large focus on Web Services
    - Zend_Feed, Zend_Service_*
  - Eclipse (Web Tools Project):
    - WSDL/XSD graphical editors
    - UDDI/WSDL Explorer
    - PHP code completion for WSDL

```
$feed = new Zend_Feed_Atom('http://atom.example.com/feed/');
echo 'The feed contains ' . $feed->count() . ' entries.' . "\n\n";
foreach ($feed as $entry) {
    echo 'Title: ' . $entry->title() . "\n";
    echo 'Summary: ' . $entry->summary() . "\n\n";
}
```

How does the PHP Collaboration Project fit in?

Zend
The php Company

- **Data is (still) King:**
  - Zend Framework:
    - Zend_Db, Zend_Search_Lucene
    - Support for XML databases
  - Eclipse (DTP Project):
    - SQL Editing
    - SQL Visual Query building
    - SQL Debugging

```
require_once('Zend/Search/Lucene.php');

$index = new Zend_Search_Lucene('/data/my_index');

$hits = $index->find($query);

foreach ($hits as $hit) {
    echo $hit->id;
    echo $hit->score;

    echo $hit->title;
    echo $hit->author;
}
```

How does the PHP Collaboration Project fit in?

Zend
The php Company

- **Rich Interfaces**
  - Zend Framework:
    - Ajax and event model
    - Json, Widgets
  - Eclipse (ATF – Ajax Toolkit Framework)
    - Javascript editor
    - Javascript debugger
    - DOM inspector
    - Toolkit specific functionality (Rico, Zimbra, Dojo)
- **Best Practices**
  - Zend Developer Zone (devzone.zend.com)
  - Zend Framework

Zend
The *php* Company

Where are we heading?

RAD

Zend
The php Company

- **Component architecture**
  - Both visual and non-visual components
- **Rich-design time experience ala Delphi and Visual Basic**
  - Drag&drop of components onto the Web page
- **Component is self-contained incl. design and run-time rendering & property editors**
  - IDEs don't have to be hard coded per-component
- **WYSIAWYG**
- **Programmatic model**

Key issues

Zend
The php Company

- **Differentiating between run-time and design-time rendering**
- **Isolation of HTML & component generated HTML**
- **Client & server side event model**
- **Property editing**
  - HTML?
- **Themes**
  - Standard CSS as a recommendation to component authors

Zend
The *php* Company

How does it all work?

Friendly Licenses

Zend
The php Company

- **Zend Framework – PHP-like license**
- **PHP IDE – Eclipse license**

Using these licenses bares great advantages:

- **Allows commercial use free of charge, no strings attached**
- **Attracts developers**
- **Licenses are well respected and widely proven**

Building the Community

Zend
The php Company

- **Industry-wide collaborative effort led by Zend including ISVs, integrators, experts both from small and large companies including:**
  - 100days.de
  - IBM
  - Ning.com
  - php | architect
  - Omni TI
  - Security experts (Chris Shiflett)
  - JamboWorks
  - and others...

Development Process (Framework)

Zend
The php Company

- **Components designed by small teams**
- **Proposals written and reviewed before coding**
- **All code starts from scratch**
- **Real Life Tests**

Quality Assurance

Zend
The php Company

- **Strict adherence to Zend Coding Standards**
- **All classes fully unit tested with PHPUnit2**
- **Peer-review and approval of all code**

Zend
The *php* Company

More Examples

ZActiveRecord

Zend
The php Company

Declare a class that extends ZActiveRecord for each database table.
Create a new instance of the class to make a new row.

The fields in the table become properties of the object automatically.

INSERT new records or UPDATE existing ones by calling save().

```php
<?php

class Person extends ZActiveRecord {}

$person = new Person();

$person->nameFirst = 'Andi';
$person->nameLast  = 'Gutmans';

$person->save();

?>
```

ZActiveRecord

Zend
The php Company

For simple queries, supply the search parameters to one of the find() methods without needing to write SELECT statements in SQL.

```php
<?php

class Person extends ZActiveRecord {}

$zeev = Person::findFirst(array('nameFirst' => 'Zeev',
                                'nameLast'  => 'Suraski'));

?>
```

For more complicated queries, KISS. No large OO-based query builder is provided. Just write the SQL and use findBySql(). Optionally, the ZDbAdapters include a simple tool for building SELECT statements to pass to findBySql().

ZMail

Zend
The php Company

Once the transport has been configured (defaults to SMTP), simply build the email and send() it.

```php
<?php
$mail = new ZMail();

$mail->setFrom('mike.n@zend.com', 'Mike Naberezny');
$mail->addTo('pmjones@zend.com',  'Paul M. Jones');
$mail->addTo('matthew@zend.com',  'Matthew Weier-O\'Phinney');

$mail->setSubject('ZMail Demo');
$mail->setBodyText('Hi All, here is some text.');
$mail->setBodyHtml('<blink>HTML</blink>');

$mail->send();

?>
```

In the example above, ZMail automatically builds the multipart email and passes it to the transport, which sends it over a persistent SMTP connection.

ZSearch

Zend
The php Company

Both a simple query API and an advanced OO-style query builder are provided. For searching simple words and phrases, use the find() method.

```php
<?php

$index = ZSearch::open('/tmp/index');

$hits = $index->find('zend');

foreach ($hits as $hit) {
    echo $hit->getDocument()->getFieldValue('title') . "\n";
}

?>
```

Each hit returns a document object. Documents are divided into fields that are chosen at the time of indexing. The documents in the index do not all need to be the same format – they may have any mix of fields.

ZSearch

Zend
The php Company

Complex query strings may be passed to ZSearch::find(). ZSearch includes a tokenizer and a parser to allows for Google-like query strings to be added to any website.

```php
<?php

$index = ZSearch::open('/tmp/index');

$hits = $index->find('zend php -java');

foreach ($hits as $hit) {
    echo $hit->getDocument()->getFieldValue('title') . "\n";
}

?>
```

The example above searches all documents of all fields for “zend” and “php” but not “java”.

Individual fields may also be specified: “-fieldName:content”.

Resources

- **Zend Framework -** 

  - http://framework.zend.com
- **PHP IDE project at Eclipse.org -** 

  - http://www.eclipse.org/php/ (TBD)
- **Zend Developer Zone -** 

  - http://devzone.zend.com

Zend
The *php* Company

Grazie!