News from PEAR

Welcome to "*PEAR and PHP5*"!
This presentation has been renamed to:

"**News from PEAR**"

Speaker: **Tobias Schlitt** <toby@php.net>
Location: **Karlsruhe**
Event: **Linuxtag X**
Date: **Apr 26, 2024**
Slides: http://pear.php.net/support/slides.php

About me

Student of Computer Science at the University of Dortmund

Besides: Freelancing IT Consultant (development / training)

Before that 4 years employed at Deutsche Bank AG (Developer, Software Architect)

Working with PHP for over 5 years

Member of the PEAR project since 2002

   Maintaining packages

   Working on the website

   Member of the Core-QA Team

Zend certified engineer

Agenda

PEAR Introduction (short!)
PEAR 1.4
PEAR Channel_Server
PEAR_Exception
[New PEAR packages]
Future outlook

PEAR Introduction

What is PEAR?

Collection of high quality PHP components

Standardization institution

Purely object oriented

Nearly 300 packages and growing fast

Almost 200 package maintainers, 500 contributors

100% free (only PHP, Apache, BSD, LGPL licenses)

Founded by Stig S. Bakken in 1999

Obtaining PEAR

PEAR installer shipped with PHP since 4.3.0

Automatically installed on Windows

Per default activated when compiling on *nix
(do not use "--without-pear")

For earlier PHP versions bootstrap from http://go-pear.org

On *nix try
```
lynx -source http://pear.php.net/go-pear | php -q
```

On Windows, save source and call PHP manually

Tip: Ensure, that your PEAR path is added to the
include_path directive in php.ini!

The PEAR Installer

Unified installation environment for installation of PHP code (PEAR) and C extensions (PECL, http://pecl.php.net)

Available for all major operating systems like

  Windows

  Linux

Different GUIs available:

  Console (built in)

  Web

  GTK

Handles dependencies between packages

Provides tools for developers

PEAR 1.4

Feature overview -1-

Channels
>   Open your own package repository

Automatic dependency resolving
>   --onlyreqdeps / -o
>   --alldeps / -a

Dependency group support

Dependencies to external packages
>   Refer to packages through URLs

Post install scripts, like
>   Setting up databases
>   Moving files to the web root

Feature overview -2-

PHAR support

    Running PEAR from a single file

Remote installation

    Install PEAR without shell access

New package.xml format

    More flexibility for your own packages

Install binary PECL packages

    Finally install PECL packages on Windows

Mirroring

    Wanna keep multiple servers up2date?

Multiple modules in one package

Channels

Set up your own package repository

> Will be covered later -> Channel Server

Receive packages from multiple sources

Cross channel dependencies

New commands for channels:

```
$> pear channel-discover http://example.com/channel.xml

$> pear channel-alias channel://... example

$> pear update-channels

...
```

Channels described in a channel.xml

Dependencies -1-

Dependencies in PEAR 1.3:

Package must be downloaded to see dependencies

No automatic resolving of dependencies

Installation fails if dependencies are not met

Only support for simple required /optional definition

Dependencies not possible to external packages

Unflexible describtion of dependencies in package.xml

Dependencies -2-

Dependencies in PEAR 1.4:

Dependency information available before downloading

Dependencies automatically resolved (if wished):

```
$ pear install –o My_Package
$ pear install –a My_Package
```

New package2.xml allows much more flexible and sensefull definition

Create dependency groups

Post install scripts

Allow you to perform powerful actions after package installation

Extreme flexible API, scripts to call are defined as in package2.xml

Tasks can be bound to specific files

Many predefined tasks available

Custom tasks implemented as simple PHP scripts

Task processing divided into steps

User interaction handled by PEAR installer

PEAR 1.3 communicated with the repository using an XML-RPC web service API

PEAR 1.4 will use ReST for this

- Less traffic

- Faster responses

- Easier API

- Much more flexible

Lear more on ReST:
http://www.xfront.com/REST-Web-Services.html

PEAR_Server

(currently Chiara_PEAR_Server)

What it can

Mirror the PEAR repository

   Faster internal updates & less external traffic

Provide your own library packages to the public

   Share yourmodules & frameworks as o.s.!

Create a module repository for internal use

   Automatic module updates for your servers

Distribute applications to your custumers

   Comfortable installation & automatic updates

Installation

Current Preconditions:

PHP 5 + PEAR >= 1.4.0a12

A domainname dedicated for your PEAR_Server

A MySQL database

Installation process:

```
$ pear channel-discover pear.chiaraquartet.net
$ pear install -a chiara/Chiara_PEAR_Server
$ pear run-scripts chiara/Chiara_PEAR_Server
```

Usage

Integrated web interface for administration

Simply discover your new channel by
```
$ pear channel-discover your.own.channel.com
```

Install the packages using
```
$ pear install your.own.channel.com/Your_Package
```

Upgrade packages using
```
$ pear upgrade your.own.channel.com/Your_Package
or
$ pear upgrade-all -c your.own.channel.com
```

PEAR_Exception

Exceptions in PHP5

PHP5 introduced the concept of exceptions

Exceptions are modeled after common OO languages

Example:

```
try {

    throw new Exception('Test');

} catch (Exception $e) {

    echo $e;

}
```

Features of PEAR_Exception

PEAR_Exception derives from Exception (PHP build-in class)

Shipped with the PEAR Installer since version 1.3.2

Use: `require_once 'PEAR/Exception.php';`

Implements some cool advanced features:

- Much prettier error output (web & shell)

- More context information on error output

- Support for nested exceptions

- Allows triggering of callbacks when exceptions are thrown (observer pattern)

Watch PEAR_Exception

See the life demo...

New PEAR packages

PEAR_Delegator

More on that later…

Service_Ebay

Very comfortable access to the eBay web services

Currently supports more the 75% of the web services API

Makes extensive use of overloading, iterators, exceptions,….

Services_Yahoo

Access to the Yahoo! web service API

Makes extensive use of iterators and overloading

PHPUnit2

Second version of the famous testing framework

Uses all kinds of PHP5 features

XML_MXML

Allows the creation of flex (some macromedia stuff) documents

PEAR_Delegator

Implements the Delegator pattern for PHP as known from C# or similar

Allows emulation "pseudo multiple inheritance"

Much cleaner code than simply using callbacks

PHP_Archive -1-

```
$ php PEAR.phar
```
  Run the PEAR installer out of 1 file!
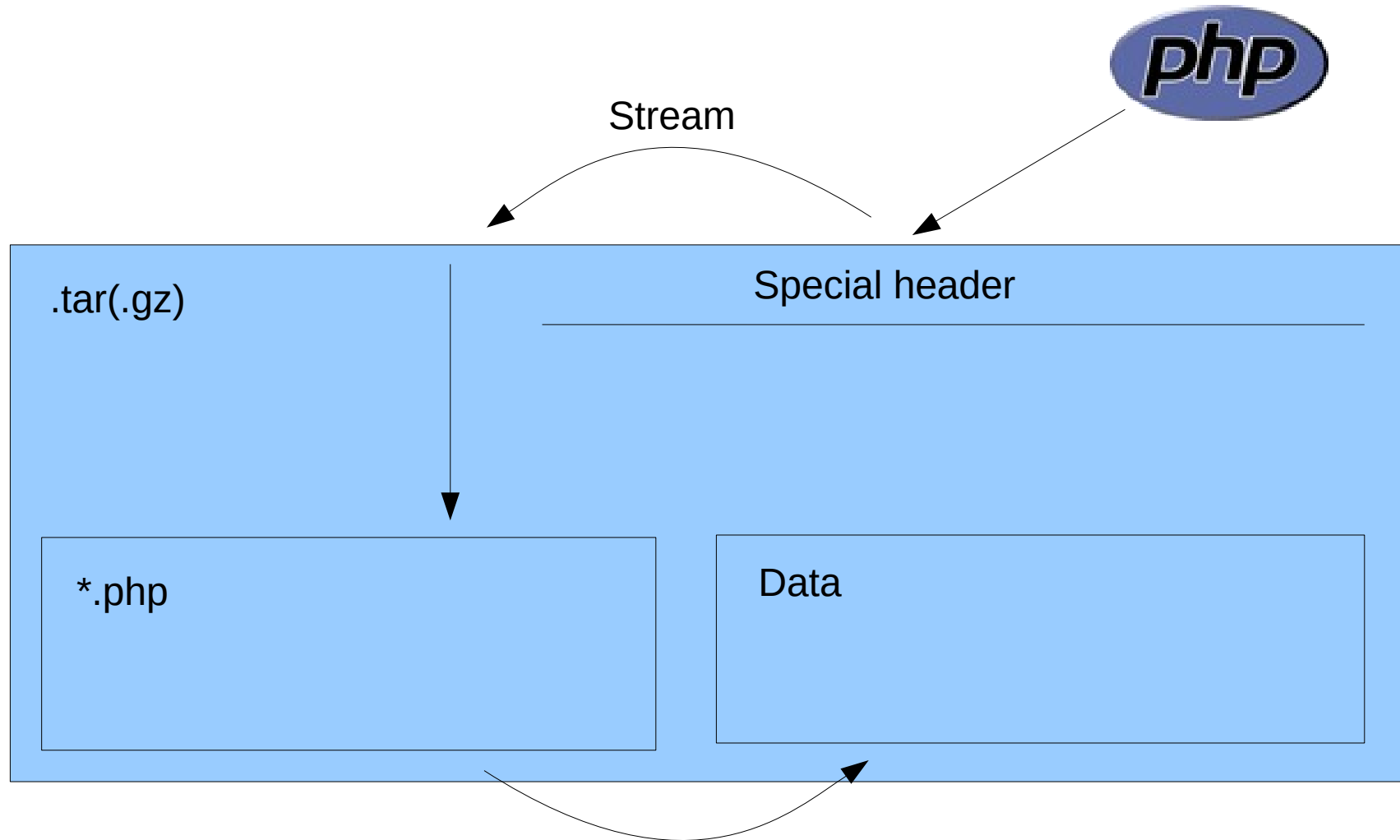
```
require_once 'phar://inc/myfile.php';
```
  Require PHP sources directly from PHARs

"New" file format PHAR

Allows packaging entire PHP applications into one file and run this directly

Completely implemented in PHP (not the best performance)

Runs with PHP >= 4.3.0 (5.0 increases speed)

PHP_Archive -2-

Future outlook

Future outlook

PEAR 1.4 is currently available in version 1.4.0a12

A beta version should occur during the next weeks (no promise!)

PEAR 1.4 should get stable during the summer (no promise!)

The end...

Thank you for your interest in PEAR!

Are there any questions left?
Do you want to give feedback?

You will us at the PEAR exhibition booth in the LAMP Area!

Useful information

License of this slides:
> OpenContent License (OPL) Version 1.0

Contact the speaker:
> Tobias Schlitt <toby@php.net>

The PEAR Website: http://pear.php.net
PEAR Mailinglist: pear-general@lists.php.net
PEAR IRC channel: #pear@EFNet

Planet PHP: http://www.planet-php.net

PEAR Seminars: http://www.akademie.de
> (Online seminars, starting in August 2005)
>   powered by