What will you be learning in the next hour

- The Various Tools used for Building EA4 RPMS
- How to patch apache in EA4
- How to package a PHP Extension
- Some pro tips we learned while working with OBS

cPanel

FEEL FREE TO STOPUS TO EXPLAIN WTF WE ARE DOING!!!!
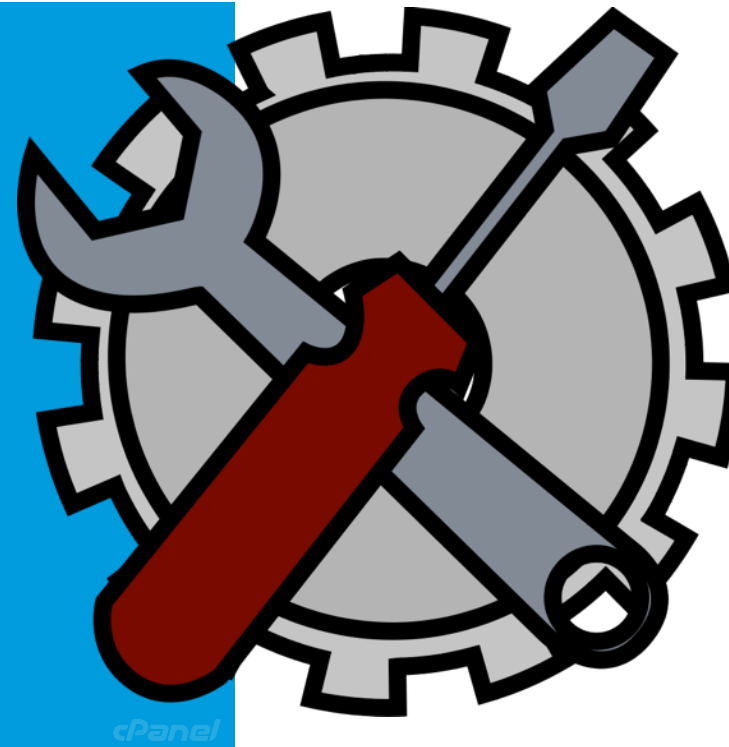
Design goals of EA4

- Do not invent our own standards
- Easy for hosting companies & integrators to manage their own stack
- Works with multiple repositories from multiple vendors

The Tools

- Open Build Service
- git
- osc
- rpmbuild
- Software Collections

There are numerous ways to solve the problem of building downstream RPMs from us. If you know a better way or already have your own infrastructure for doing so, PLEASE USE IT. What we are doing is by no means specific to cPanel, one of our goals in EA4 was that you didn't need to use processes specific to cPanel for managing your web stack.

Creating Your Development
Sandbox

cPanel

When working with EA4 packages you will find that there are various things that you need to do, in the next few slides we'll be discussing how to get started with OBS and some of the basic tools for working with Customizing tEA4.

We strongly recommend performing these changes on a CentOS 7 image, however the instructions that follow should work for just about any linux distribution, or even mac (with a bit of TLC to get the correct software installed)

Create an Open Build Service Account

https://build.opensuse.org

Open Build Service is an excellent free services provided by SuSE for building packages on numerous different distros from a single set of sources. We looked at numerous options for building packages and OBS appeared to be the only one that met all of our needs.

Install the osc utility

download.opensuse.org/repositories/openSUSE:/Tools/

This is the osc utility, it is the CLI interface to work with OBS. Packages are provided here for most common linux distributions.

DEMOING OSC

What is isv?
osc ls isv:cpanel:EA4

osc co isv:cpanel:EA4 scl-php71
osc branch isv:cpanel:EA4 scl-php71
osc ci -m "lol"
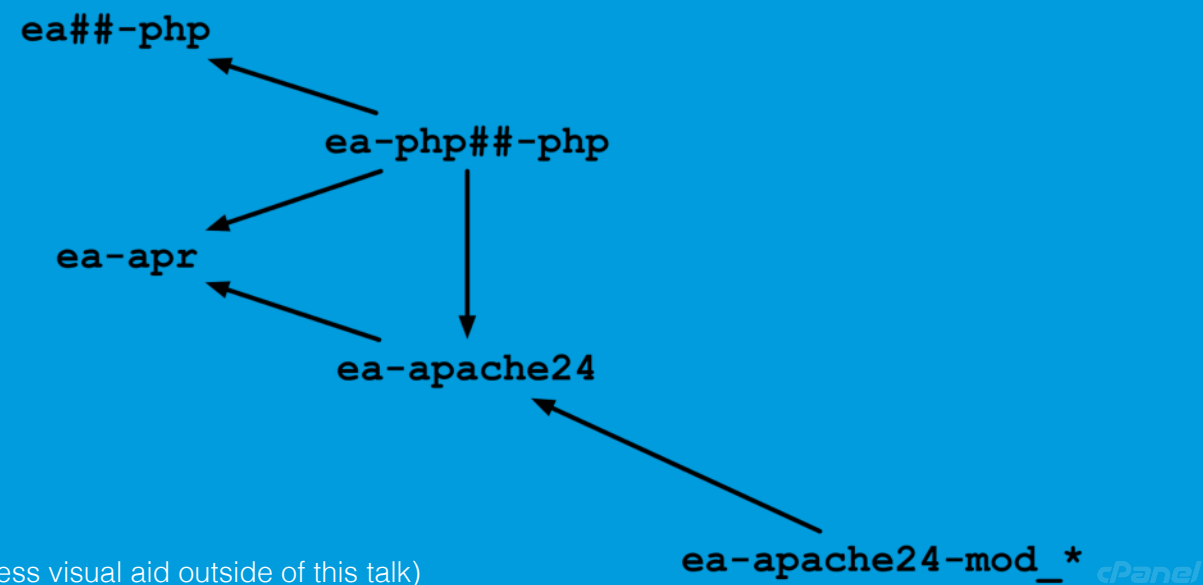
Patching is used when we want to make a small modification to a piece of code in an RPM.

Patching is used to change behavior, or a small piece of a piece of software.

Our goal with EA4 was to make management of these sorts of customizations as simple as possible to manage across multiple servers.

Dependencies

ea##-php

ea-php##-php

ea-apr

ea-apache24

ea-apache24-mod_*

(a worthless visual aid outside of this talk)

cPanel

If you decide to patch a package you may have to start rebuilding other things that depend on it depending on if the binary interface has been modified. This means that you may need to start shipping the entire stack if ea-apr has been updated in a manner that is no longer compatible with mod_dso or apache.

We are trying to do the right thing

- Mailing Lists
- Blog posts on Updates
- Hosted on github
- Hosted on OBS

cPanel

We are trying to behave the same as distributions on these matters, providing git repos, build services, etc.

DEMO
PATCHING

use symlink protection patch (BH patch)

branch

checkout

Edit spec - DISCUSS WHAT WE ARE CHANGING IN THE SPEC

* release #

* changeling

* Patch directives

osc add on the patch file
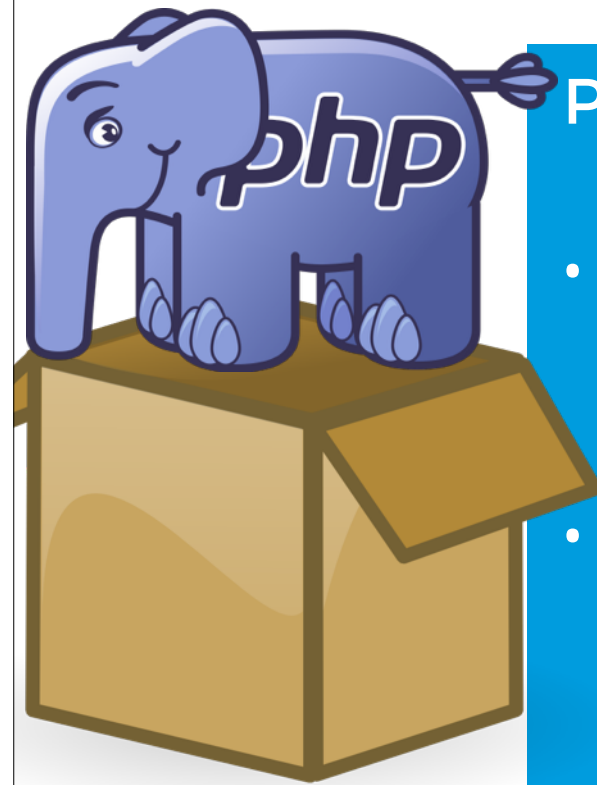
osc build (local build)

osc ci -m "LOL NEW" (build on server)

Creating a Package

Demoing how to build a PHP Extension

PHP packages

- Different extensions allow for specialized hosting options
- Customers always have that one "Critical" extension

cPanel

https://github.com/piannelli/htscanner-enhanced ?

Pro Tips!

cPanel

Run your own OBS

http://openbuildservice.org/

Linking OBS & git

```
>_  osc add git://lolololol
    osc service

    MATT TODO: flesh out this output
```

perk@cpanel.net
matt@cpanel.net