Drupal Training

Ross Brodbeck

Agenda

- What is Drupal?
- How does it work?
- Why Drupal?
- Drupal Concepts
- Drupal Architecture

WHAT IS DRUPAL?

What is Drupal?

- Open Source Web Content Management System
- Written in PHP
- Fairly mature product
- Pluggable architecture using "modules"
- 648,625 people in 228 countries speaking 181 languages who work with Drupal
- Drupal 6 to Drupal 7 transitional phase (concepts, work)

What Can you do with Drupal?

- **Blog**
- **Corporate Websites**
- **Photo Galleries**
- **Video Galleries**
- **Forums**
- **E-commerce**
- **E-learning**
- **Intranet**
- **Wiki**
- **Applications**

HOW DOES IT WORK?

Framework or CMS?

- Drupal as a framework
  - Functionality extended via modules and code
- Drupal as a CMS
  - Start with the basics, build it yourself
- Distributions
  - Prepackaged Drupal instances
  - "out of the box functionality

Drupal Distributions

- Pressflow
  - "fast" version of base Drupal
- Acquia Commons
  - Wiki's, discussions, etc out of the box
- OpenPublish
  - Run a news publishing site
- OpenPublic
  - Run a government collaboration site
- OpenAtrium
  - Collaboration Suite

Drupal as Open Source

- Drupal as an Open Source product
- Vibrant community of developers
- Many add-ons, projects
  - Pro: lots to choose from
  - Con: lots to choose from (and not always good)

WHY DRUPAL?

Drupal Strengths

- Large community of developers
- Always improving (open source)
- Many modules – first step to creating functionality: find a module
- Well documented
- Many large scale implementations (proven)
- Accepted by many government agencies

Drupal Weaknesses

- Not a full scale CMS

- Always changing

- More maintenance work than a traditional website

- No single vendor

- Learning curve

Who is using Drupal?

- Government
  - White House.gov
  - Energy.gov
  - Dutch government
- Entertainment
  - Grammy's (grammy.com)
  - MTV (mtv.co.uk/)
  - WWE (wwe.com/)
- Corporate Website
  - AT&T
  - Intel

DRUPAL CONCEPTS

Drupal is "content soup".

Nodes

- Primary Building blocks of all content
- Nodes have "Fields"
- Example node: Blog Post
  - Blog post has a title (field)
  - Blog post has a body (field)
  - Blog post has an attachment (field)
- Each node is a page
  - (but can be show in lists as well)
- Nodes can (and will) have multiple URLs

Content Types

- All nodes in a system are organized by content type
  - Not by hierarchy or taxonomy!
- First question on creation is:
  - "what type of node do you want?"
- Content types are global throughout the system

Menus

- All pages in drupal are managed by their URL:
  - www.drupal.com/url1
  - www.drupal.com/url2/argument
- Menus attach nodes to various URLs
- Menus create a "heirarchy" of URLs
  - Main Menu
    - Page 1 (attached to node 1)
    - Page 2 (attached to node 2)

Taxonomy

- Taxonomy is separate from menus
- Use tags to classify content
- Create multiple sets of tags using "vocabularies"
- Vocabulary 1
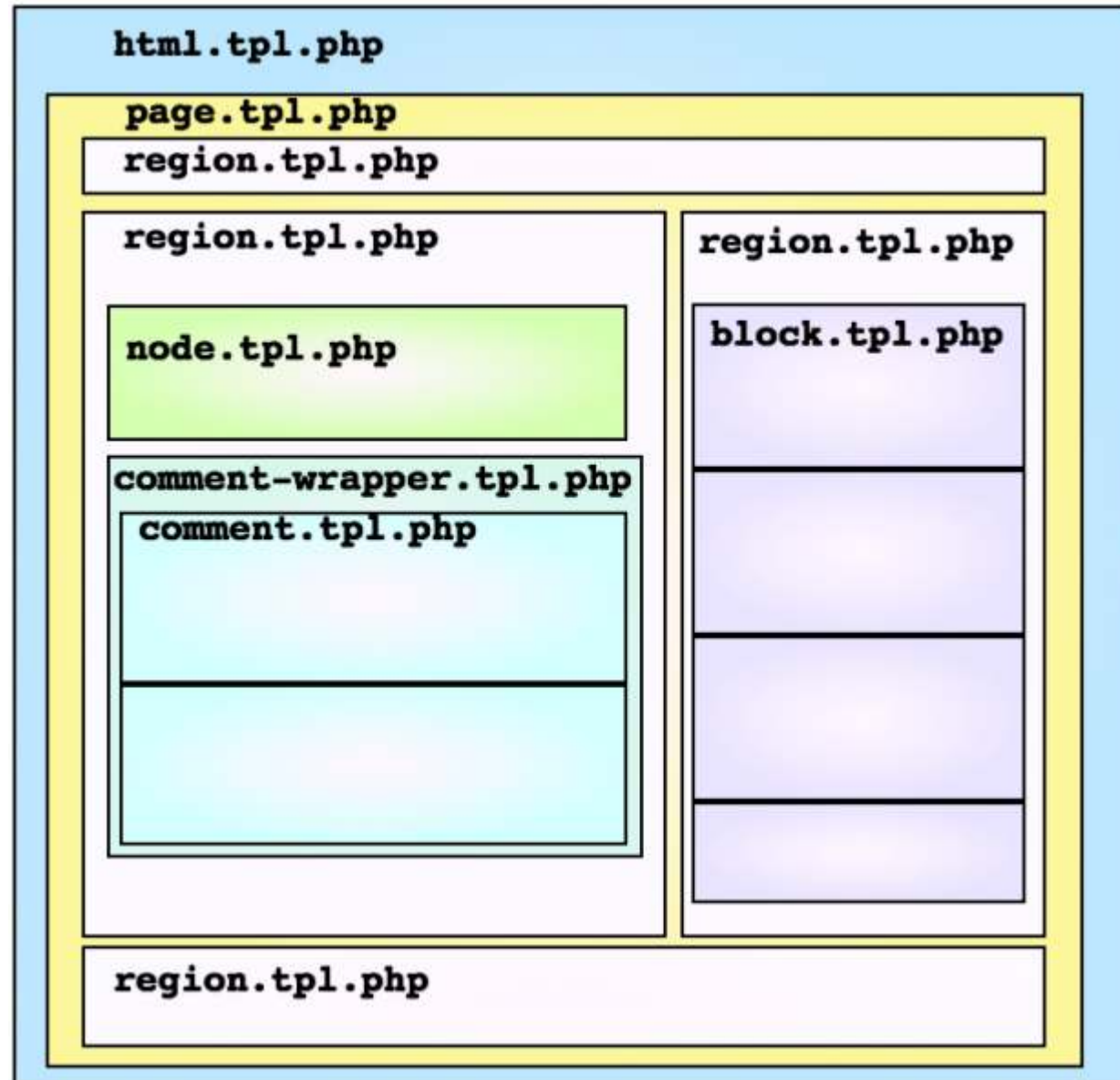 - Tag 1
  - Tag 2
- Vocabulary 2
  - Tag 1
  - Tag 2

Module

- Set of PHP code for extending Drupal

- 1000's of modules on Drupal.org (prebuilt)

- Use "hooks" concent to invoke Drupal API

- User interface for "enabling"/"disabling"

- Can drastically alter Drupal functionality

Themes

- Themes define how content looks
- Themes are sets of PHP pages with variables in them
- Themes exist in various layers
  - Theme parts of a page (region)
  - Theme parts of a module
  - Theme parts of a menu
- Themes define their own regions

Theme Layout

Block

- Part of a page

- Predefined piece of content

- Think "Portlet"

Important Things

- Drupal does not have a predefined "page heirarchy"
  - There can be multiple menus
  - There can be multiple themes
- Content is "floating in space"
  - Can be searched
  - Can be accessed via URL
  - Can be added to a menu

DRUPAL ARCHITECTURE

Drupal Concepts You Should Know

- Primarily a LAMP stack
  - Linux
  - Apache
  - MySQL
  - PHP
  - Features usually targeted here first
- Can run on Windows
  - IIS
  - PHP
  - MySQL
- Support for SQL Server, NoSQL databases coming in 7

Drupal Architecture

Drupal

PHP

Web Server

Database

Common Modules

- **CCK-** Allows you to add custom fields to existing content types.
- **Views -** Provides a flexible method for Drupal site designers to control how lists and table of content are presented
- **LDAP Integration-** Allows users to authenticate against multiple LDAP
- **Form Builder –** Allows you to build form easily using AJAX interface
- **WYSIWYG –** Allows user to enter content through a WYSIWYG interface.
- **SimpleNews-** Allows you to create newsletter
- **Organic Group –** Allows you create groups among users and have their own private sites.

BRINGING IT TOGETHER

Required Skills

- PHP Programming
- Jquery
- Javascript
- CSS
- Apache/IIS Knowledge
- MySQL/Database knowledge

- Go to Drupal.org. Get involved.
  - Blog aggregator
  - Module contributor

Typical Drupal Process

- Get the base code from Drupal's site
- Select the modules you want
- Code custom modules for missing functionality
- Configure
- Create a theme for custom look
- Tweak site based on issues/concerns
- Apply patches/updates