cazh1 Web Development Administration Guide

Check the Scotland development box for updates to this document.
The **Last Update**: date in the lower right corner is your indication of freshness.

Overview

cazh1.com represents a collection of web sites, applications, and other web-related projects. In support of these efforts, we have established these Development Standards to simplify prototyping and development, rollout and scale up, extensions, and ongoing maintenance.

This document captures the critical processes and steps required ...

- *Technical Architecture* information (how does this stuff work?)
- *Process and Procedure* - basic administration tasks (what do I have to do today?)
- *Metrics and Measurements* (what have we accomplished today?)
- *Technical Support* - contact information (who can I call / where can I go for help?)

Note that this is not meant to be an all-encompassing “book of knowledge” for web site services; we’ll capture the critical tasks / steps, and attempt to provide a single resource for this basic information.

This document is not meant to replace the various systems engineering, administration manuals, and other generally available resources. We will refer to the proper documentation for standard and/or more detailed processes.

This is a task-oriented document – many of the sections contain checklists, designed to be printed and used again and again. The idea is the same behind a pilot’s takeoff ritual, checking off the various switches and settings that must be “just so” before leaving the ground. When dealing with a complex arrangement of systems, settings, and software, it makes sense to follow a standard checklist, so we can get it right, every time!

Installation Notes

PHP5

- Command Line Interpreter: You will want to install a command line version of PHP for scripts and script debugging.
  - Via Synaptic Package Manager, install `php5-cli`, the command-line interpreter for the PHP5 language
  - Make sure Eclipse is pointing to the correct place. From within Eclipse, go to Window > Preferences > PHP > PHP Executables. There need only be one item there. The executable path should be `/usr/bin/php5`.
- Web Server: To correctly configure a web server for PHP web applications ...
  - Make sure Eclipse is pointing to the correct place. From within Eclipse, go to Window > Preferences > PHP > PHP Servers. There need only be one item there, typically called Default PHP Web Server.
    - On the Server tab, the URL for the document root of this server should be `http://localhost`
    - On the Path Mapping tab, there should be a mapping for the server root (Path on Server = \) - set this to Path in the File System = `/media/data01/home/jpmacl/WebDev/Dev`

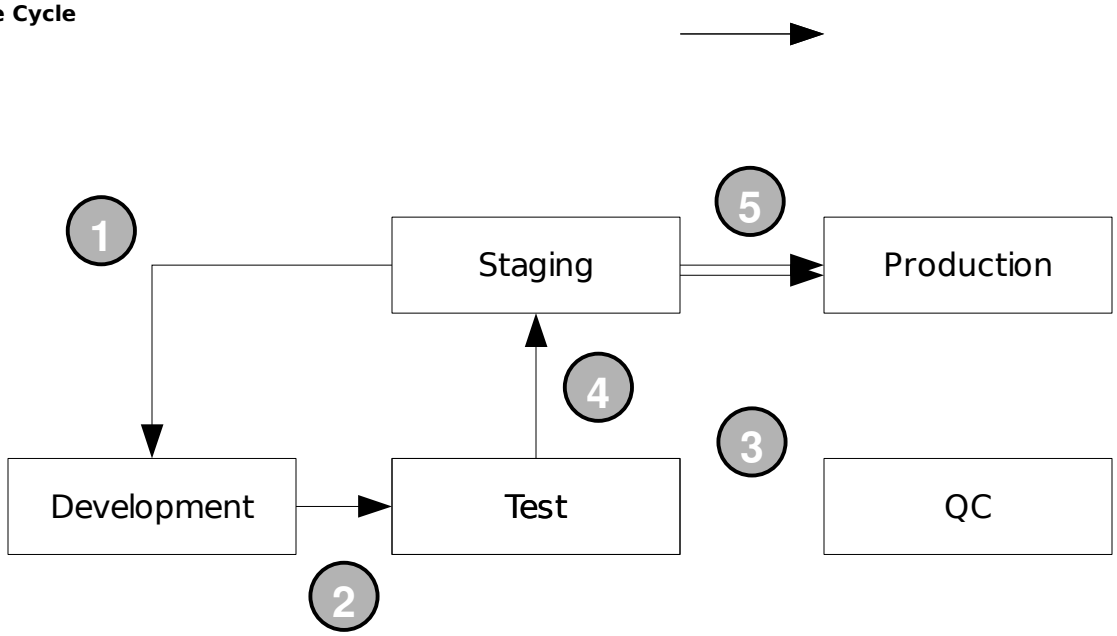
Development Life Cycle

Environments

Cazh1 web applications use a typical set of environments for change control. We'll start in the middle ...

- *Staging*: (aka 'the Golden Copy', 'Gold', 'Gold Code') This is the current, best, cleanest copy of the production version of this web site / application.
- *Development*: (*aka* '*Dev*') A very “disposable” environment, there can be any number of Dev environments for a given web app, These are typically created by copying the current Gold code to a new folder set in the coder's Home directory.
- *Test*: (*aka* '*SysTest*') Local test, a relatively clean environment where you can run the web application through it's paces.
- *QC*: (*aka* '*Q*', '*Quality*', '*Volume Test*') Production test – for externally hosted
- *Production*: (aka 'Prod')

Life Cycle



The numbers refer to the diagram above. Detailed process steps for the *Processes* mentioned here are found in the Process and Procedure section.

1. When it's time to start a new web project, a full set of environments are created in all of the various locations – creating an empty Dev environment. If you are going to update an existing web project, you can re-load the Dev environment with a copy of the gold code.
2. Unit testing can happen in you Dev environment, but before the work is put into production, the code must be promoted to test, and all of the proper testing completed.
3. For sites that will be hosted on an external server – typical for sites visible to the public internet – code is moved to the a QC environment. This is your last check to make sure the code behaves properly when running on the ultimate destination server – there may be diofferences in the way the web server is configured, and these can be checked here.
4. When your work is ready to be put into production, it is copied into the Staging environment. Note that we do not copy stuff directly to production – this happens automatically.
5. The Staging environment is synchronized with Production automatically, every 15 minutes – or on demand if there is some particular rush. Sync'g continuously allows us to keep an eye out for changes to the production folders that are visible to the public internet; if someone is planting spurious code in the production sites, this process will automatically correct the changes and notify that changes are happening.