



Dancing With the Stars (Without Stepping on Your Own Toes)

How to Update a WordPress Site

Marissa Martínez, GreenFerret, LLC

James Coletti, Boston WP Meetup Organizer

What You Will Know at the End of This Presentation

- Why use production and development environments
- Two installation/upgrade approaches
 - Subversion as an installation tool
 - Tarball installation
- Handling code vs. content (files vs. data)
- Including testing passes
- Backup and rollback strategies

GreenFerret



© 2010
GreenFerret LLC

What We Will Not Cover

- Multi-site installations
- Multi-server installations
- Load balancing challenges
- Testing theory
 - Testing automation
- Step by step instructions for
 - Stack and WordPress installation
 - Creating a backup
 - Marking tables for export and exporting data
 - Merging files in source control
 - Importing data
- All the other ways to do upgrades

GreenFerret



© 2010
GreenFerret LLC

Three Environment Approach

- Sandbox
 - Developer work area
- Development/staging
 - Passes unit testing
- Production
 - Live site

GreenFerret



© 2010
GreenFerret LLC

Local Machine (sandbox):

- Typical installation
 - One each of OS, MySQL, php, Webserver (Apache), CMS (WordPress)
 - Multiple customer installations
- Tools to consider
 - Mac: stack comes by default but may want to upgrade to newer versions, XAMPP or MAMP
 - Windows: XAMPP
 - XAMPP, MAMP (and others) offer all-in-one installation



Development/Staging

- Merge work of multiple developers
- Allow customer data access
 - Testing features
 - Creating release data

GreenFerret



© 2010
GreenFerret LLC

Production

- Upload WordPress to existing stack
- Fantastico to simplify
 - Mixed reviews
- Make files, Python scripts
 - Simplifies standard installations
 - Automation prevents errors
 - Maintain in source control

GreenFerret



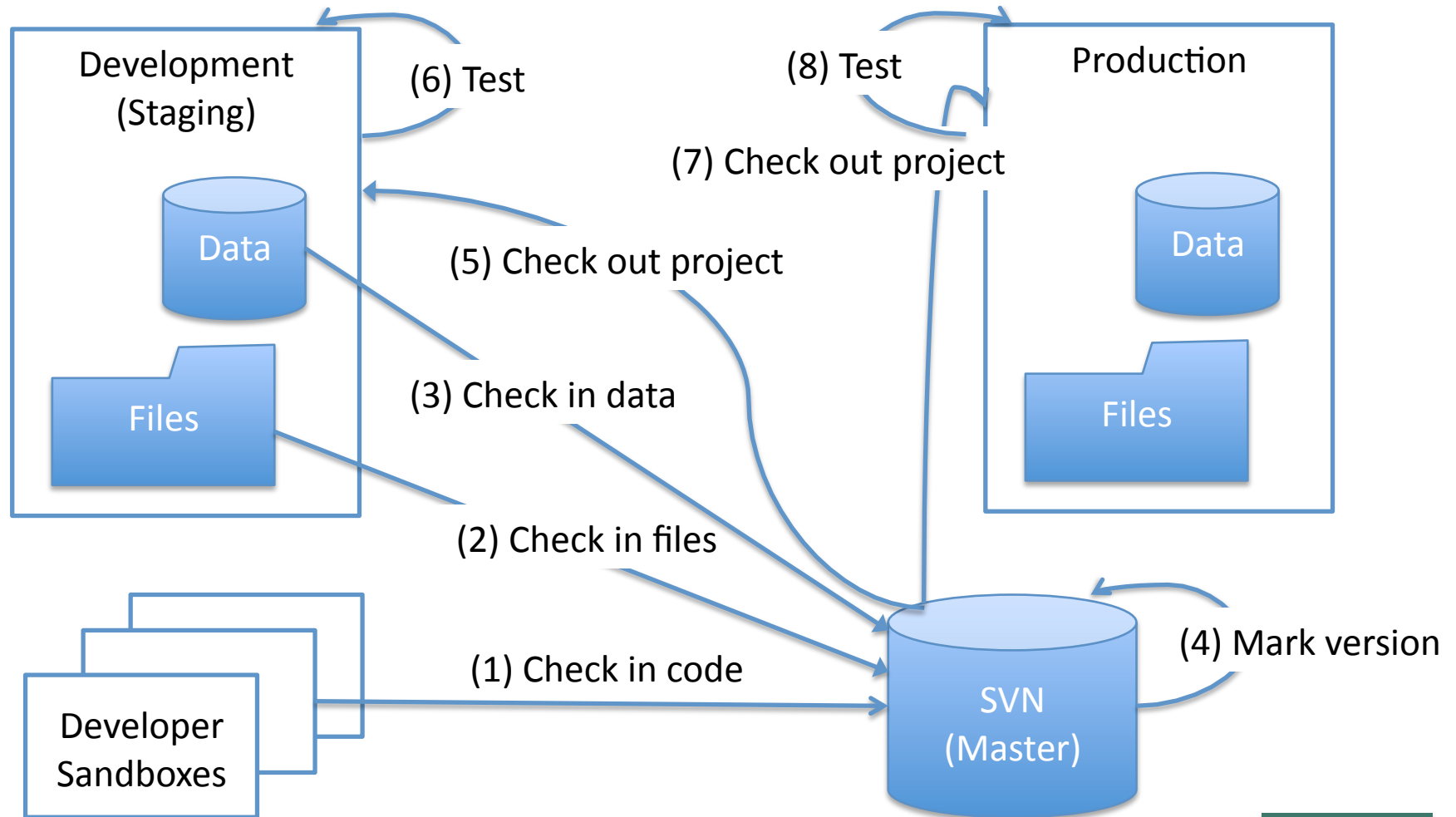
© 2010
GreenFerret LLC

Source Code Management

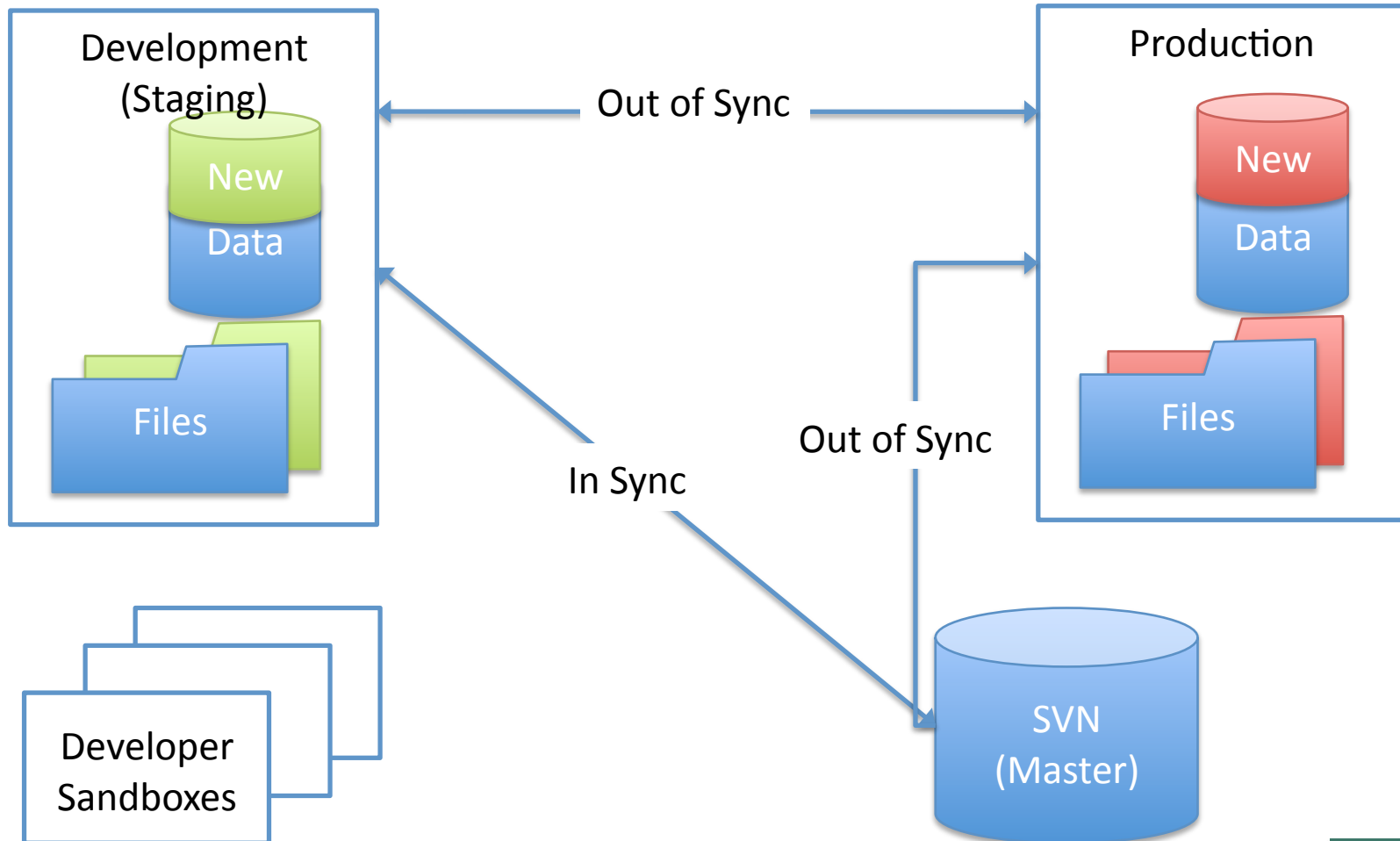
- SVN (Subversion)
- Production and staging environments are SVN Users
 - Check out and check in
 - Releases are tagged
 - Assumes command line access to servers
- Snapshot of Customer Installation
 - Data
 - Customer-uploaded files



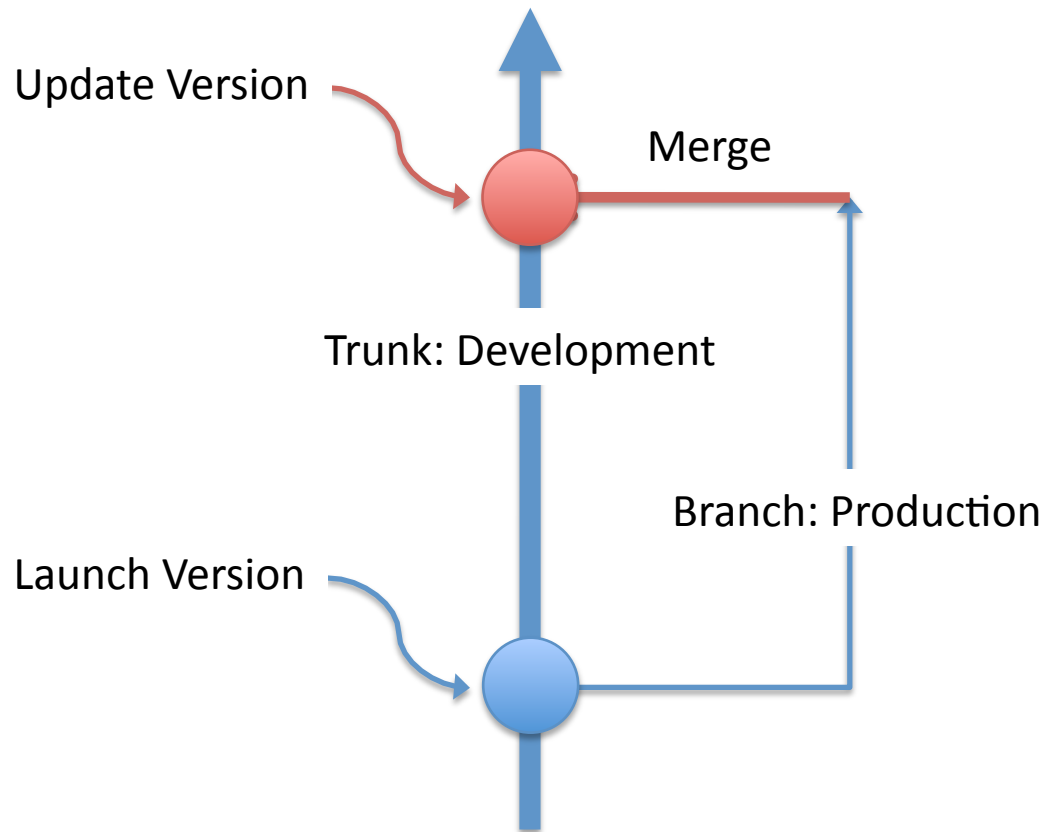
SVN Approach (Launch)



Time Passes....



Effective Project Branching



Upgrade Steps

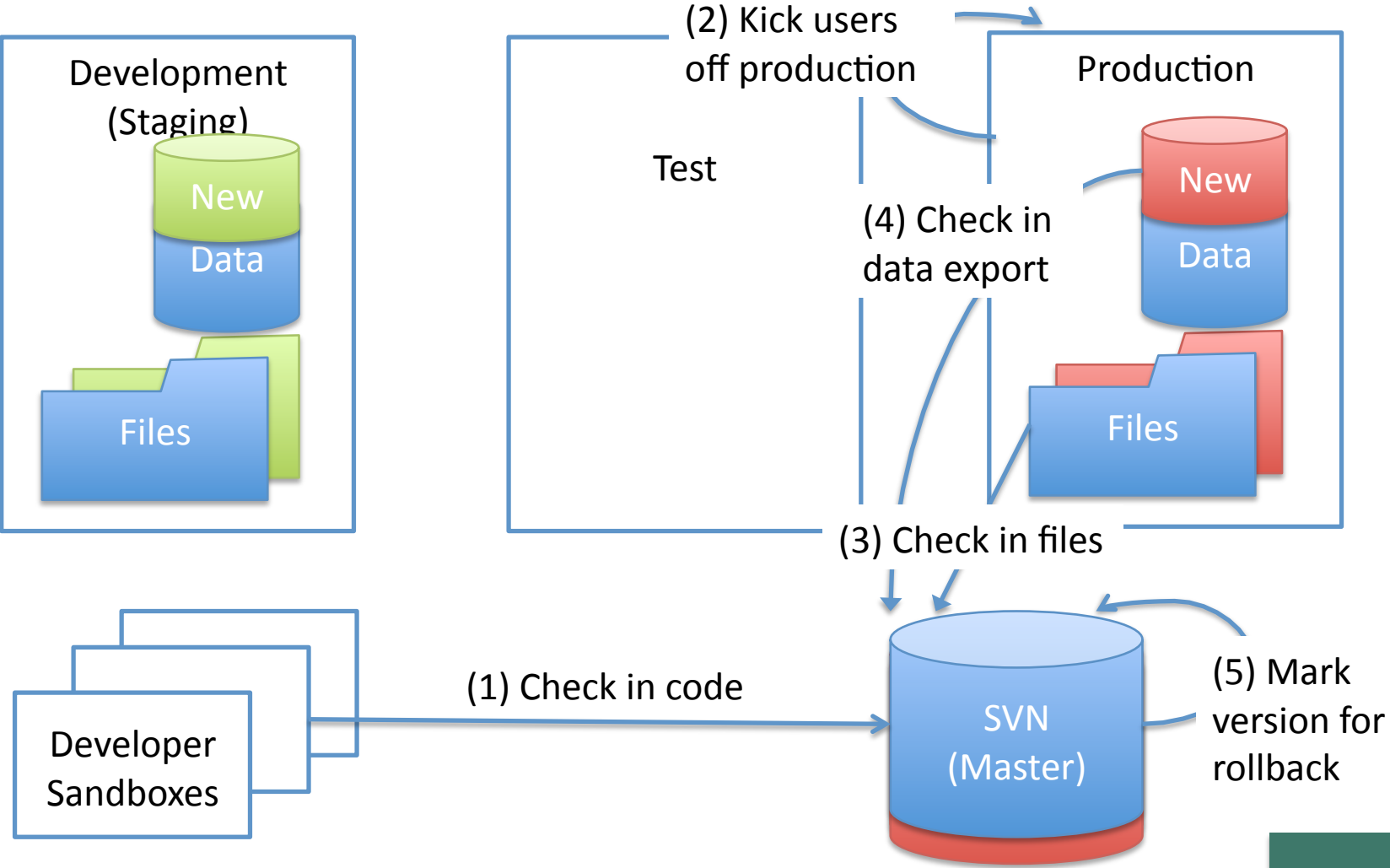
- Prep for Rollback Contingency
- Prep Upgrade
- Test the Upgrade
- Upgrade the Production Environment

GreenFerret

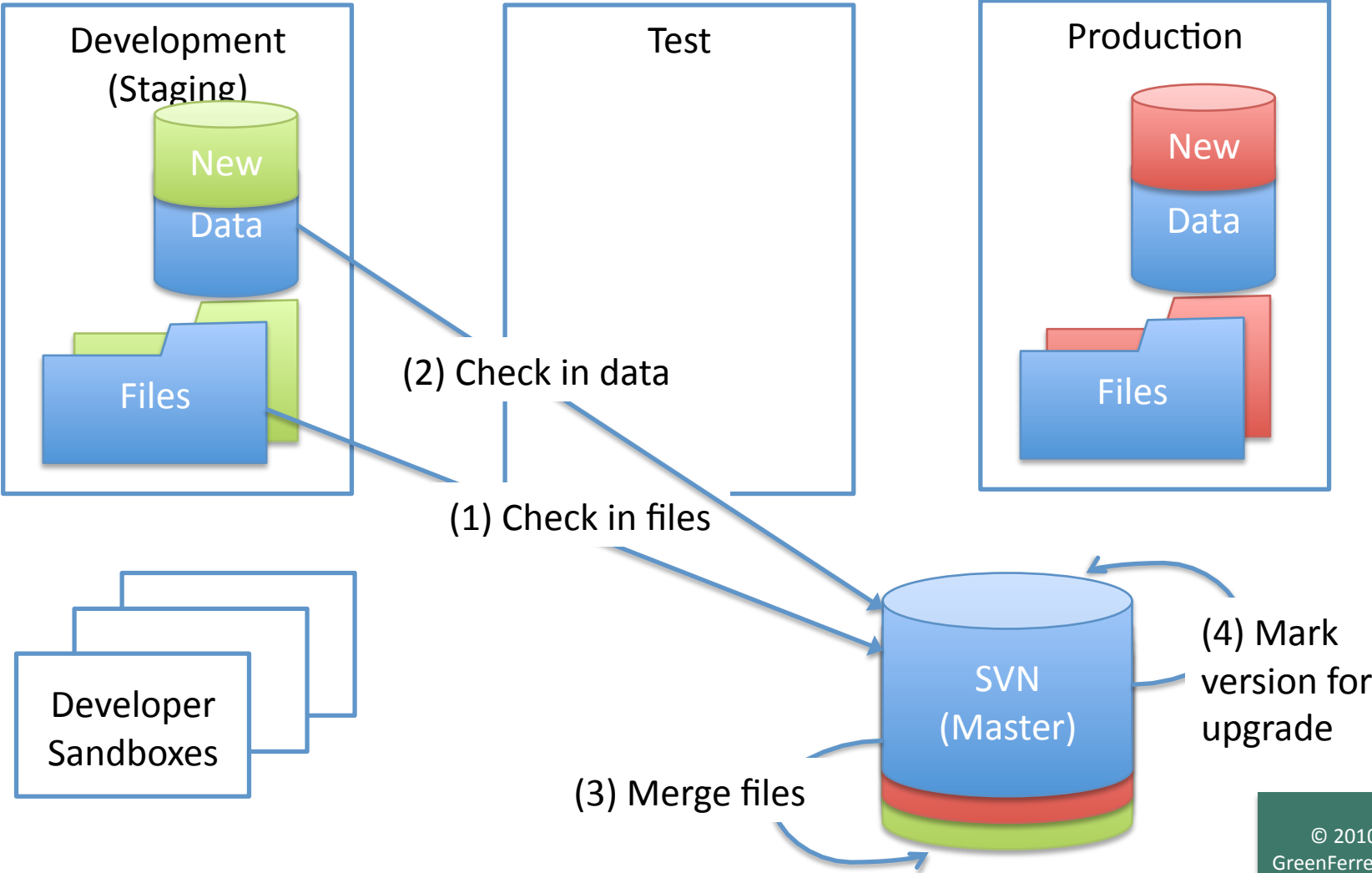


© 2010
GreenFerret LLC

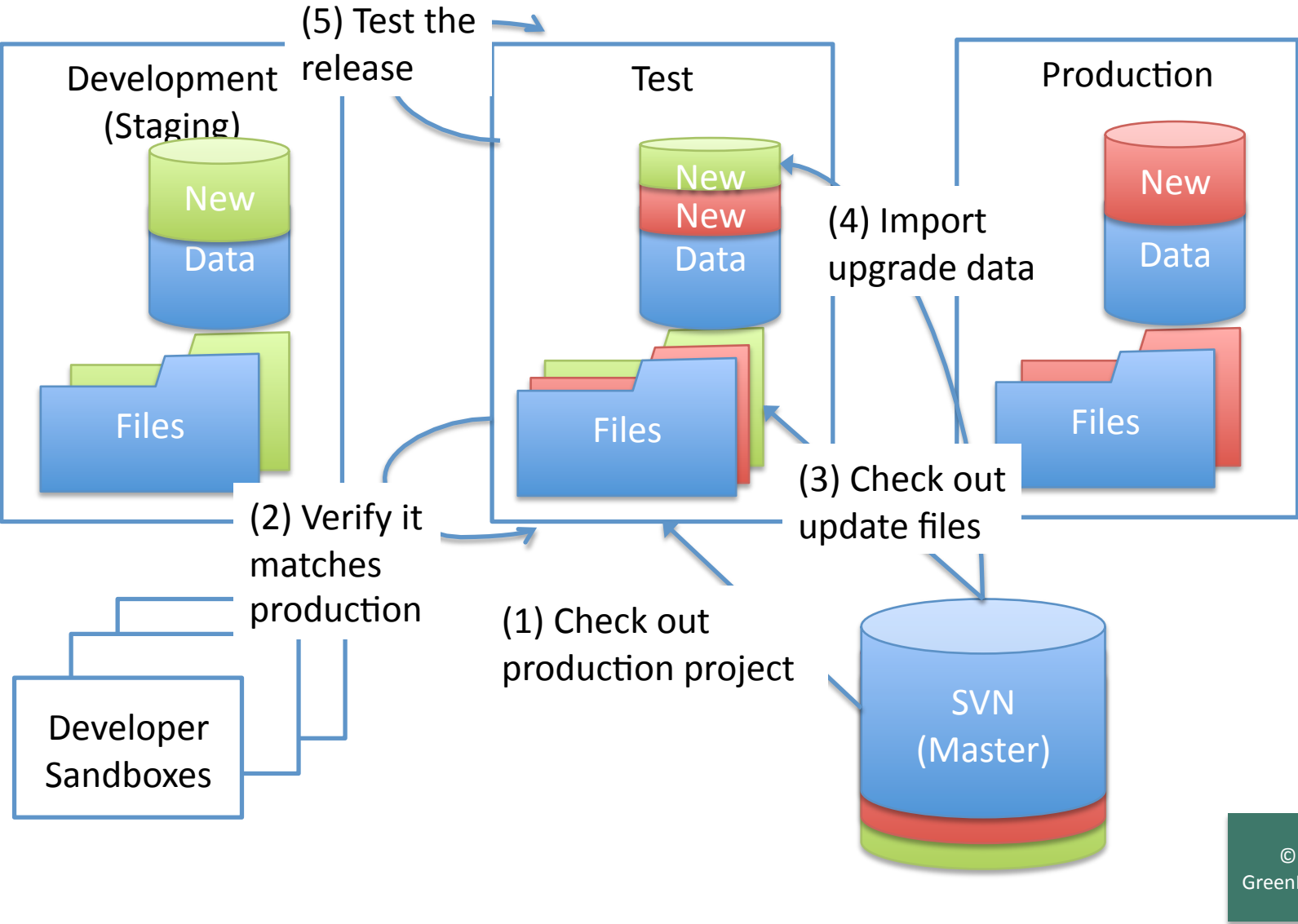
SVN Approach—Prep for Rollback



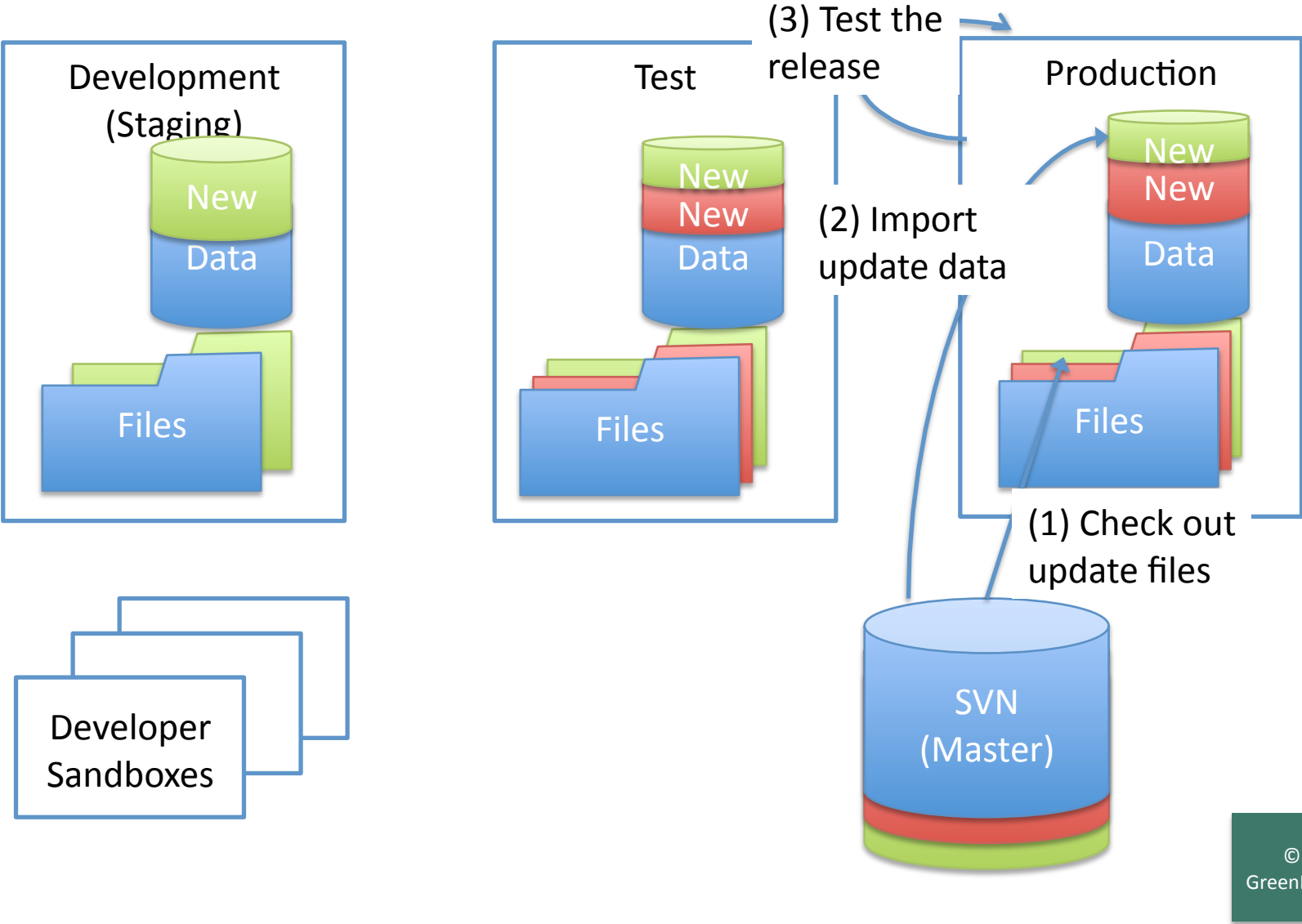
SVN Approach—Prep the Upgrade



SVN Approach—Test the Upgrade



SVN Approach—Upgrade Production



Everything works? Declare Release!



Don't forget to let the customer and users back onto Production!



Packaging and Deploying a Release (Code)

- Upgrading WordPress
 - Instructions:
[http://codex.wordpress.org/Upgrading WordPress Extended](http://codex.wordpress.org/Upgrading_WordPress_Extended)
- Customer release, Tarball and go!
- Capistrano for deployment
 - Make file concept (capfiles)
 - Ruby script originally for Rails apps but can deploy anything
 - Google: “capistrano wordpress” for instructions
- Careful with modified site files
 - wp-config.php, .htaccess, robots.txt
 - Other custom files

Capistrano Approach – Release via Tarball

- Latest code checked out from source control locally
- Any specified “exclude” files removed from release
- Release packaged (& compressed) into tarball
- Uploaded & unpackaged to dated release directory on one or more production servers (via scp, ftp)
- “Current release” link updated to point to new dated release
- Shared user-generated content directory links updated
- “Deploy” tasks executed (database changes, server restart, cache cleanup, etc.)

Data Synchronization & Updates (Content)

- Manually via SQL or tools such as phpMyAdmin
- Mark tables for export on data source
- Export data
- Create new tables as needed
- Import data
- Data merging
 - Manage through offline workflow process
 - Bulk edit
 - Clean up permalinks, if necessary (find & replace)



Notes About Testing

- Production server is never the same as development servers
 - Test in both environments
- Browser compatibility
- Automated test tools
 - <http://www.opensourcetesting.org>

GreenFerret



© 2010
GreenFerret LLC

Backup and Rollback Strategies

- Ongoing backup (always a good idea) versus backup for upgrade procedure
- WordPress plugin WPDB-backup
 - Allows schedule, email, or FTP
 - Services (VaultPress, iDrive)
- Check your ISP for backup service
 - Charges
 - Test backups
- Cloud, servers, external HD for storage

GreenFerret



© 2010
GreenFerret LLC

Final Notes

- Test the release and upgrade process prior to launch date

GreenFerret



© 2010
GreenFerret LLC

Launch Checklist

1. Check in final code from development sandbox to SVN project
2. Check in “loose” customer files from development/staging environment
3. Check in data export
4. Mark release
5. Check out project to staging
6. Full test in staging
 - Bugs? Return to Step 1
7. Check out project to production
8. Test
 - Bugs? Remove from production and return to Step 1
9. Declare release!

GreenFerret



© 2010
GreenFerret LLC

Upgrade Checklist

1. Prep for Rollback Contingency
 1. Check in final code from development sandbox to SVN project
 2. Kick users off production and block them out
 3. Check in "loose" customer files from production environment
 4. Check in data export from development/staging environment
 5. Mark production release branch off previous release (This is the rollback version)
2. Prep upgrade package
 1. Check in "loose" customer files from development/staging environment
 2. Check in data export from development/staging environment for new tables
 3. Merge files
 4. Mark release and DB update (Ready for test run)
3. Test Pass
 1. Check out production project to testing environment (Start with production image)
 2. Verify the test environment matches production (to ensure your rollback will work)
 3. Check out upgrade project to testing environment
 4. Import DB update for upgrade to testing environment
 5. Full test in testing environment
 - Bugs? Remove user block in production environment and return to Step 1
 - Otherwise ready for official upgrade
4. Production Pass
 1. Check out upgrade project to production environment
 2. Import DB update for upgrade to testing environment
 3. Test
 - Bugs? Rollback to production snapshot, release user block, and return to Step 1
5. Declare release!
6. Let users back onto production environment

GreenFerret



© 2010
GreenFerret LLC

Tool Roundup

- LAMP installation: XAMPP, MAMP
 - <http://www.apachefriends.org/>
- Stack and WP installation: Fantastico
 - <http://www.netenberg.com>
 - <http://www.cpanel.net>
 - [One warning review](#) against Fantastico
- Source control: Subversion (SVN)
 - <http://subversion.apache.org/>
- Deployment, DB migration, Make files: Capistrano
 - www.capify.org
- Automated testing
 - <http://www.opensourcetesting.org>
- Backups
 - WPDB-backup: <http://wordpress.org/extend/plugins/wp-db-backup/>
 - http://codex.wordpress.org/WordPress_Backups
 - VaultPress: <http://www.vaultpress.com> (Automattic, Inc.)
 - iDrive: <http://www.idrive.com/wordpress.htm>



Summary

- Benefits of multiple environments
- Two approaches for installation/upgrades
- Differences between content and code
- Including testing into the process
- Preparing backups and rollbacks

GreenFerret



© 2010
GreenFerret LLC

Questions

GreenFerret



© 2010
GreenFerret LLC



Dancing With the Stars (Without Stepping on Your Own Toes)

How to Update a WordPress Site

LearnMore@greenferretllc.com

www.greenferretllc.com

617-306-1566
