Intro to UIC RED
red.uic.edu

- The WordPress platform for creating and managing UIC websites

- UIC branded theme framework for a consistent look across department websites

- Includes a variety of page layouts and content components that are designed to create responsive, accessible and usable websites without extensive knowledge of website development

- Allows departments to host sub-unit websites nested within departmental sites

614 production sites,
300 dev and archive sites in a single installation

Red Multisite

UIC WordPress

Start here to learn →
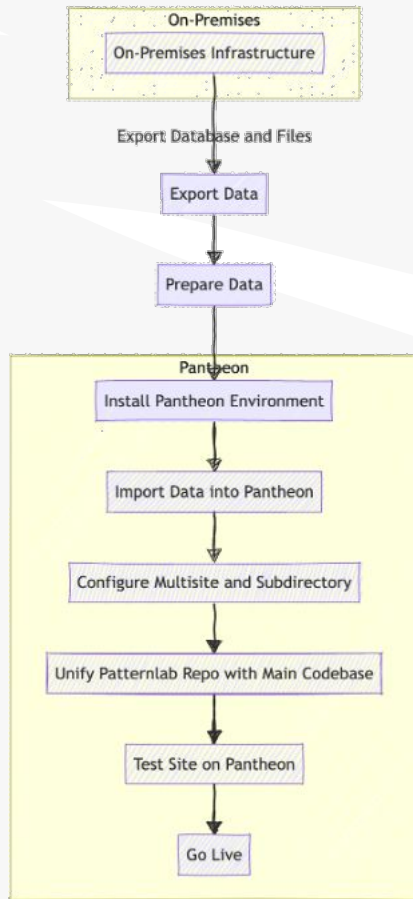
⬚ Tutorial   ⬚ What's Available in RED   ✓ Requirements   ⚑ Accessibility

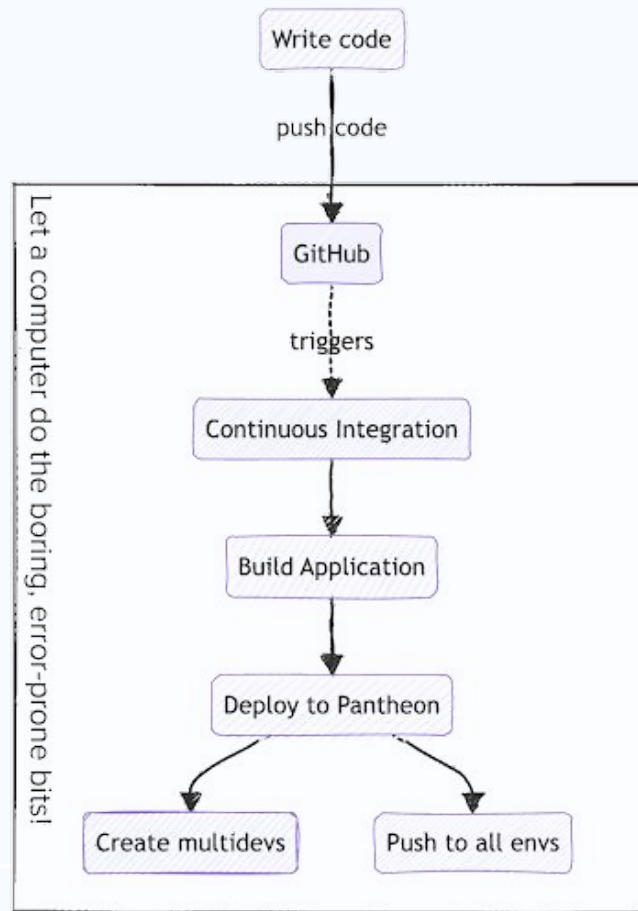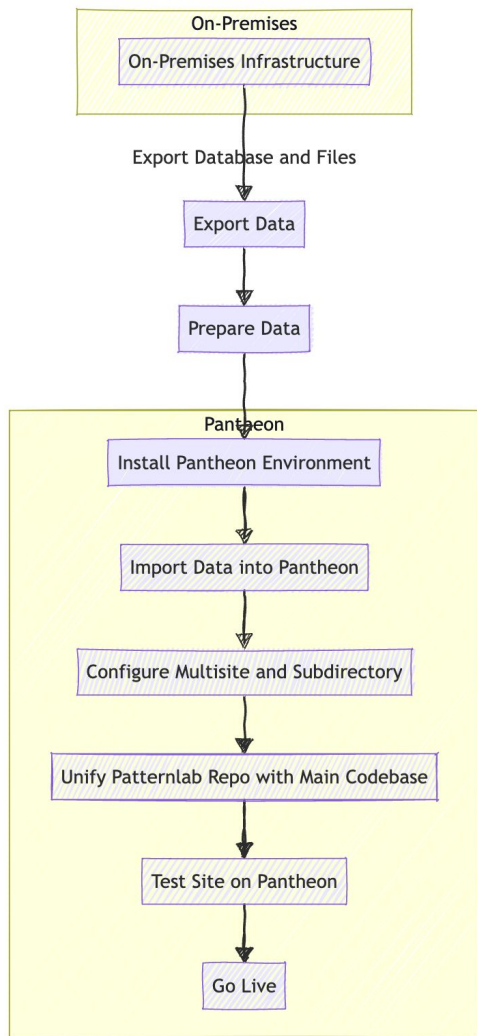Big Network, Small Team

Migration to Pantheon

Issues (on-prem hosting):

- Infrastructure update and Maintenance
- Non Scalable
- Security
- Logging and Monitoring.

Collaboration with Human Made

Challenges on Pantheon:

- Performance optimization
- Support for subdirectory multisite structure and subdomain multisite networks
- Ensuring single source code management
- Security
- MultiDev setup

On-Premises

On-Premises Infrastructure

Export Database and Files

Export Data

Prepare Data

Pantheon

Install Pantheon Environment

Import Data into Pantheon

Configure Multisite and Subdirectory

Unify Patternlab Repo with Main Codebase

Test Site on Pantheon

Go Live

Write code

push code

Let a computer do the boring, error-prone bits!

GitHub

triggers

Continuous Integration

Build Application

Deploy to Pantheon

Create multidevs

Push to all envs

Prepare the codebase

getcomposer.org
packagist.org ▪ wpackagist.org

```
composer require wpackagist-plugin/query-monitor
composer update wpackagist-plugin/safe-svg
```

```
wp-content/

    mu-plugins/

        first-party-plugin/

        third-party-plugin/    # Composer-installed, .gitignore'd

        autoloader.php         # Versioned file which loads mu-plugins

    plugins/

        first-party-plugin/

        third-party-plugin/    # Composer-installed, .gitignore'd

    themes/

        atom/

    uploads/                   # .gitignore'd entirely
wordpress/                     # Composer-installed, .gitignore'd
vendor/                        # Dev-time dependencies (PHPCS, etcetera), .gitignore'd
index.php                      # Versioned in Git
wp-config.php                  # Can be locally committed OR .gitignore'd
```

```json
"repositories": {
  "wpackagist": {
    "type": "composer",
    "url": "https://wpackagist.org",
    "only": [ "wpackagist-plugin/*", "wpackagist-theme/*" ]
  }
},
"require": {
  "johnpbloch/wordpress": "~6.6.0",
  "wpackagist-plugin/web-stories": "^1.35.0",
  "wpackagist-plugin/wordpress-seo": "^23.0"
},
"extra": {
  "installer-paths": {
    "wp-content/plugins/{$name}/": [ "type:wordpress-plugin" ]
  },
  "wordpress-install-dir": "wp"
},
```

Dependabot

dependabot (bot) commented on behalf of **github** on May 15   ⋯

Bumps wpengine/advanced-custom-fields-pro from 6.2.6.1 to 6.2.10.

🤖 compatibility unknown

Dependabot will resolve any conflicts with this PR as long as you don't alter it yourself. You can also trigger a rebase manually by commenting `@dependabot rebase`.

▸ Dependabot commands and options

Merge repositories

Those who cannot remember the past are condemned to repeat it...

How do you maintain `git` history when merging unrelated codebases?

brew install git-filter-repo

```
% git checkout you@host.com:your/theme-repo.git
% cd theme-repo
% git filter-repo --to-subdirectory-filter \
                  wp-content/themes/ourtheme
```

before

```
[16:09:02] % tree | head -n4
.
├── LICENSE
├── bin
│   └── install-wp-tests.sh

[16:09:04] % git log --oneline --graph
* 656a7e3 tools(ci): Add a benign commer
* 36a5a70 tools(build): install jQuery
* 2cf191c chore: Fix syntax warning abou
* ed287c7 chore: Adding some global lice
* d51e9fd chore: moved wp-sri plugin url
* 684b193 chore: repatriate semicolons
* a42fcb1 feat: ignore emergency alert u
```

after

```
[16:10:11] % tree | head -n7
.
└── wp-content
    └── themes
        └── uic-red
            ├── LICENSE
            ├── bin
            │   └── install-wp-tests.sh
[16:10:27] % git log --oneline --graph
* 905fead tools(ci): Add a benign commer
* 2e261de tools(build): install jQuery
* f311960 chore: Fix syntax warning abou
* 7b4b349 chore: Adding some global lice
```

git merge
--allow-unrelated-histories

```
% cd new-monorepo
% git remote add theme ../theme-dir/.git
% git fetch theme
% git merge --allow-unrelated-histories theme/main
```
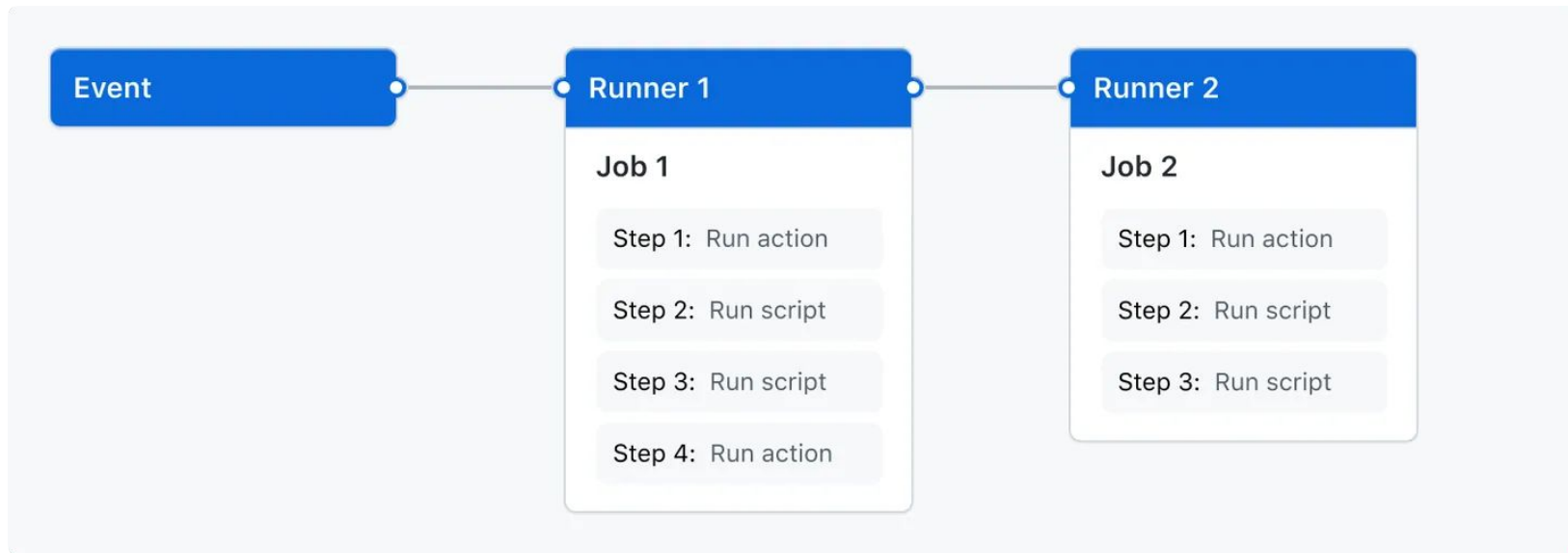
Codebases: Merged

Synchronize changes

Resources

- github.com/newren/git-filter-repo
- Blog post: Merging git repos with `git filter-repo`
- GitHub documentation on removing sensitive information (using filter-repo)
- Interactive Rebasing tutorial on Bitbucket
- Blog post: Using Dependabot with WordPress

Continuous Integration

GitHub Actions

```yaml
name: Build & Deploy

on:
  workflow_dispatch:
  workflow_call:
  push:
    branches:
      - main
  pull_request:
    # Only run on a given PR if relevant source files have changed.
    paths:
      - '**.php'
      - 'etc'

jobs:
  build:
    uses: ./.github/workflows/build.yml
    secrets: inherit
```

.github/workflows/ci.yml
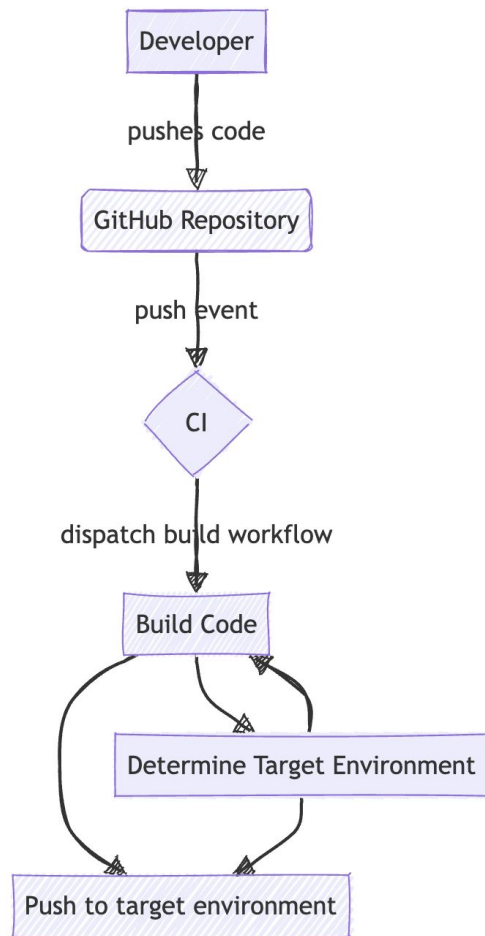
build.yml

1. Checkout codebase

2. Check whether workflow should run

3. Set up PHP (Composer dependencies)

4. Set up Node (Build dependencies)

5. Build project

6. Commit built files to target branch

7. Push updated deploy branch

```mermaid
flowchart TD
    A[Push to Main] -->|triggers| B[Cross-Site Deploy to Pantheon]
    B --> C[Network A Dev]
    B --> D[Network B Dev]
    B --> E[Network C Dev]
    C -->|Test and approve| F[Network A Staging]
    D -->|Test and approve| G[Network B Staging]
    E -->|Test and approve| H[Network C Staging]
```

Push to Main

triggers

Cross-Site Deploy to Pantheon

Network A Dev   Network B Dev   Network C Dev

Test and approve   Test and approve   Test and approve

Network A Staging   Network B Staging   Network C Staging

Choose Deploy Target

```yaml
- name: Identify target branches and Pantheon environment context
  id: environment
  uses: ./.github/actions/environment
  with:
    PANTHEON_PROJECT_IDS: ${{ vars.PANTHEON_PROJECT_IDS }}
```

Main or Multidev?

```yaml
- name: Identify target branch names and multidev context
  id: branch_name
  shell: bash
  run: |
    SOURCE_BRANCH="${GITHUB_HEAD_REF:-${GITHUB_REF#refs/heads/}}"
    echo "source_branch=$SOURCE_BRANCH" >> $GITHUB_OUTPUT
    DEPLOY_BRANCH=$(bash .github/actions/environment/determine-target-branch.sh "$SOURCE_BRANCH")
    echo "deploy_branch=$DEPLOY_BRANCH" >> $GITHUB_OUTPUT
```

```yaml
outputs:
  source_branch:
    value: ${{ steps.branch_name.outputs.source_branch }}
    description: Name of source branch for build (either pushed branch or PR base branch).
  deploy_branch:
    value: ${{ steps.branch_name.outputs.deploy_branch }}
    description: Name of branch to deploy to, inferred from source_branch.
  is_multidev:
    value: ${{ steps.branch_name.outputs.is_multidev }}
    description: Whether a multidev should be created for this branch.
  pantheon_project_ids:
    value: ${{ steps.environment.outputs.pantheon_project_ids }}
    description: The specific Pantheon instance used as our main deploy target.
```

gist.github.com/kadamwhite/1c049512abcd8db99023e920771316b8
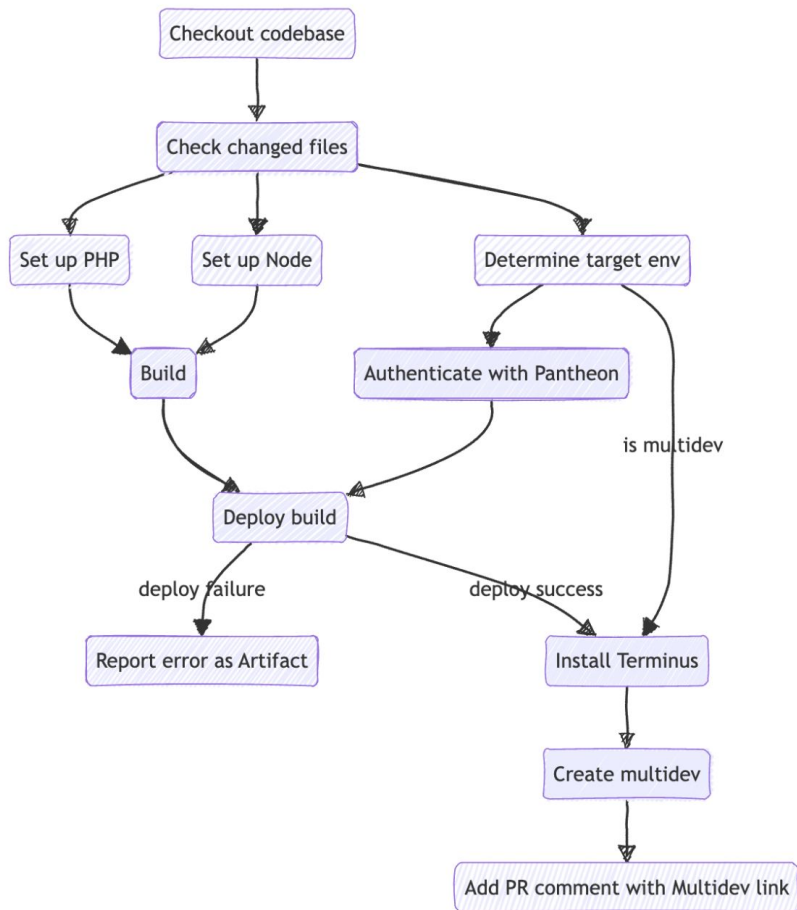
Full workflow

1. **Checkout & build**

2. **Determine target environment**

    a. Multidev or Main?

3. **Deploy to Pantheon env(s)**

4. **Use terminus to create multidev, if necessary**

5. **Report status as PR comment**

Resources

- [Blog Post: Getting started with CI and WordPress](#)
- [Understanding GitHub Actions](#)
- [GitHub Actions Documentation](#)
- [Pantheon terminus CLI documentation](#)
- ["Introducing GH Actions" on CSS Tricks](#)
  (slightly outdated but still good)

Looking forward

Further CI opportunities

Code style consistency
Security "sniffs"

Outcomes

Outcomes

- Simplified setup for devs
- Retained Pantheon multidev capabilities for review and testing
- Allows for single codebase deployed to multiple installations
- Introduce insert tests, security, and accessibility scanning into build process

Thank You!

UNIVERSITY OF ILLINOIS CHICAGO

Human Made