Continuous Integration
Quality at Speed" is the new norm in software development
More info forthcoming

- [https://jenkins.coeus.ca/job/tiki/](https://jenkins.coeus.ca/job/tiki/) builds master, sends mail upon breakage, and calculates test coverage.
- [https://gitlab.com/tikiwiki/tiki/pipelines](https://gitlab.com/tikiwiki/tiki/pipelines) works, gitlab seems to have fixed their Docker cache issue. It now works on both master and 19.x branch. When it breaks it seems to be on real bugs...
- [http://sonarqube.coeus.ca/dashboard?id=tiki](http://sonarqube.coeus.ca/dashboard?id=tiki) has useful output, and is connected to test coverage.

See also: [https://sourceforge.net/p/tikiwiki/code/HEAD/tree/trunk/.gitlab-ci.yml](https://sourceforge.net/p/tikiwiki/code/HEAD/tree/trunk/.gitlab-ci.yml)


Tiki releases every 6 months. However, the next release should ideally be "releasable at any time". Also, actual releases should be very easy (goal: 30 minutes for the technical part).

To get closer to these goals, the ideal would be community Continuous Integration server.

There are daily builds but if something breaks, there is no alert system to report the issue.

Wishlist for Tiki community
Pre-Dogfood Server
*For humans to be able to see yesterday's data with tomorrow's code*

- Pre-Dogfood Server

Profiles demo server

- Having a test server with main profiles applied regularly, for testing/demo.

Continuous build
*For the daily pre-release zip file to be as close as possible to the final one*

- Run the existing scripts we use at release time
- Automatic commits: we could register a "tikiwiki" user at Sourceforge.
  - update of changelog.txt (maybe once a week?)

Continuous Testing Server
*Machines testing code, according to a series of tests*

- Run all tests
  - Check PHP & Smarty Syntax, etc.
  - Check that all JavaScript can be safely minified with JSLint
  - Detect closing ?> tags from DevTips
- Run the security tests regularly (monthly?) and report to Security Team about new potentially risky files
- Ideas from How to improve the release process
php doc/devtools/translate.php englishupdate lag=10 audit --email=me@example.org to report if anyone breaks strings over the last 10 days
  etc

Please see Continuous Testing Server

Notes

Database Compare

In order to run the script doc/devtools/check_schema_upgrade.php which will compare your current database with what it should be you need to install DBDiff with this command:

```
php temp/composer.phar require "dbdiff/dbdiff:@dev"
```

Related links

- https://trunkbaseddevelopment.com/
- https://continuousdelivery.com/
- PHPCI is a free and open source continuous integration tool specifically designed for PHP.
  - https://www.ohloh.net/p/phpci
  - https://www.phptesting.org/hosted-phpci
- http://docs.codehaus.org/display/SONAR/PHP+Plugin
- https://www.ohloh.net/p/xinc-continuous-integration-for-php
- http://confluence.public.thoughtworks.org/display/CC/CI+Feature+Matrix
- https://github.com/wimg/PHPCompatibility
- https://medium.com/linagora-engineering/dockerized-ci-of-linagoras-james-team-b01bd34a2641
- https://github.com/phpstan/phpstan
- https://github.com/squizlabs/PHP_CodeSniffer

Related projects

CDash and CMake

GoCD

Buildbot

Jenkins

- http://jenkins-php.org
- Jenkins XMPP

PHPCI
CruiseControl (Java)

Travis CI

Strider

Integrity

http://integrity.github.io

Xinc

Apache Gump

GitLab Continuous Integration & Deployment

https://about.gitlab.com/features/gitlab-ci-cd/
https://docs.gitlab.com/ee/ci/pipelines.html

- That’s why we’ll be starting right away by bringing Gemnasium security checks to GitLab CI/CD. GitLab already supports security testing for JavaScript, Python and Ruby. The next version, GitLab 10.5, will include an implementation of Gemnasium, which will improve significantly the dependencies check for these languages. We expect to have coverage for PHP and Java in 10.6 (to be released on March 22).

Alias

- Continuous Integration and Continuous Delivery
- CI/CD