XTools

Release 3.0.0

Aug 31, 2017

Table of Contents

1 Pre-requisites **3**
1.1 Databases . . . 3

2 Installation **5**
2.1 Single wiki . . . 5
2.2 Wiki family . . . 5

3 Configuration **7**
3.1 Databases . . . 7
3.2 Authentication and Email . . . 8
3.3 Application . . . 8
3.4 Tools . . . 8

4 Opting in to restricted statistics **11**
4.1 How to opt in . . . 11
4.2 How to opt out . . . 11

5 Tools **13**
5.1 Edit Counter . . . 13
5.1.1 General Statistics . . . 13
5.1.2 Namespace totals . . . 14
5.1.3 Timecard . . . 14
5.1.4 Year counts . . . 14
5.1.5 Month counts . . . 14
5.1.6 Latest global edits . . . 14
5.1.7 Automated edits . . . 14
5.2 Top Edits . . . 14
5.3 Simple Counter . . . 15

6 Development **17**
6.1 Style Guidelines . . . 17
6.2 Running Development server . . . 17
6.3 Developing against WMF databases . . . 18
6.4 Table mappings . . . 18
6.5 Caching . . . 19
6.6 Writing the docs . . . 19
6.7 Releases . . . 19

6.8 Additional Help . . . 20

7 The tools **21**
7.1 Edit Counter . . . 21
7.2 Admin Score . . . 21
7.3 Admin Stats . . . 21
7.4 Article Info . . . 21
7.5 Auto Edits . . . 22
7.6 Bash . . . 22
7.7 Blame . . . 22
7.8 EC . . . 22
7.9 Pages . . . 22
7.10 RFA . . . 22
7.11 RFAP . . . 22
7.12 Simple Counter . . . 22
7.13 Top Edits . . . 22

8 Help **23**

XTools is a suite of statistics tools for MediaWiki wikis, users, pages, and more. It is in operation for Wikimedia wikis and can also be installed for any MediaWiki installation.

Quick links:

- Demonstrations:
  - Wikimedia installation: xtools.wmflabs.org
  - Development installation: xtools-dev.wmflabs.org
- This documentation: xtools.readthedocs.io
- Source code: github.com/x-tools/xtools-rebirth
- Issue tracker: phabricator.wikimedia.org
- IRC: #wikimedia-xtools on Freenode

CHAPTER 1

Pre-requisites

XTools requires the following to run:

- A recent version of Linux (Windows servers are supported, however; you must enable the `app.load_stylesheets_from_cdn` if you want it to look nice).
- PHP 5.5.9+ (not tested on PHP7)
  - JSON must be enabled.
  - ctype needs to be enabled
  - You must have `date.timezone` set in `php.ini`.
  - PDO including the driver for the database you want to use
  - Curl must be enabled.
- Composer 1.0.0+

Databases

1. One or more project databases. These should be current mediawiki installations. The meta database should point to them.
2. A Meta database. If you are running more than one wiki (`app.is_single_wiki` set to false), information on each wiki must be stored in a meta database. XTools uses one modeled after the WMF Labs database..

   This database must live on the same machine as the project databases.

   See the *installation documentation* for more details if you don't already have this database available.
3. An optional Tools' database, where other MediaWiki tools store their data.

CHAPTER 2

Installation

To install XTools, please follow these steps:

1. Download the repository into a web-accessible location. If you're using Git, the `master` branch is always stable so you can clone that.
2. Ensure that `var/` and all files within it (other than `var/SymfonyRequirements.php`) are writable by the web server.
3. Run `composer install` and be prompted to enter database details and other configuration information.
4. Open XTools in your browser; you should see the XTools landing page.

To update the cache after making configuration changes, run `./bin/console cache:clear`.

Single wiki

To use XTools for a single wiki, set the following variables in `parameters.yml`:

- `app.single_wiki` to `true`
- `wiki_url` to the full URL of your wiki
- `api_path` to the path to the root of your wiki's API

Wiki family

To use XTools for a family of wikis, set `app.single_wiki` to `false` in `parameters.yml`.

You will also need to create a new database table to record the meta information about your wikis. It can live wherever you want; just set the `database_replica_*` variables accordingly in `parameters.yml`.

The table must be called `wiki` and have the following structure:

```
CREATE TABLE `wiki` (
    `dbname` varchar(32) NOT NULL PRIMARY KEY,
    `lang` varchar(12) NOT NULL DEFAULT 'en',
    `name` text,
    `family` text,
    `url` text
);
```

(The WMF version of this table can be browsed at Quarry #4031.)

CHAPTER 3

Configuration

As part of the installation of XTools, `composer install` or `composer update` may prompt you for configuration options. This is a definition of those options.

Databases

XTools' own database:

- **database_host** - Hostname for the server with the XTools database
- **database_port** - Port for the server with the XTools database
- **database_name** - Database name of the XTools database
- **database_user** - Username for the XTools database
- **database_password** - Password for the user for the XTools database

The projects' databases:

- **database_replica_host** - Hostname for the server with the MediaWiki databases
- **database_replica_port** - Port for the server with the MediaWiki databases
- **database_replica_name** - Database name of any one of the MediaWiki databases (usually the default, or the 'meta'; it doesn't matter which).
- **database_replica_user** - Username for the MediaWiki databases
- **database_replica_password** - Password for the user for the MediaWiki databases

The 'meta' database:

- **database_meta_name** - Database Name for the server with the meta_p table (this is not required if `app.single_wiki` is set)

Other tools' database (e.g. checkwiki):

- **database_toolsdb_host** - MySQL host name

- **database_toolsdb_port** - MySQL port number
- **database_toolsdb_name** - Username to connect as
- **database_toolsdb_password** - Password to use for the user

Authentication and Email

The Oauth details need to be requested from `Special:OAuthConsumerRegistration` on your default wiki.

- **auth_key** - Oauth consumer key
- **auth_secret** - Oauth consumer secret
- **mailer_transport** - Software for the mailer
- **mailer_host** - Hostname for the mailer
- **mailer_user** - Username for the mailer software
- **mailer_password** - Password for the mailer software

Application

- **secret** - A secret key that's used to generate certain security-related tokens
- **app.noticeDisplay** - Display the notice or not
- **app.noticeStyle** - Style of the notice banner. Available options: "error," "warning," "succeess," "info."
- **app.noticeText** - Message shown to the user. If you provide a valid intuition key, it will display that message instead
- **app.replag_threshold** - Number of seconds to consider the replicas as "lagged", and show a warning to the user that the data may be out of date
- **app.load_stylesheets_from_cdn** - Whether to load our stylesheets and scripts from a CDN. This is required if XTools is installed on a Windows server
- **app.single_wiki** - Point XTools to a single wiki, instead of using a meta database. This ignores database_meta_name above.
- **app.is_labs** - Whether XTools lives on the Wikimedia Foundation Labs environment. This should be set to false.
- **wiki_url** - URL to use if app.single_wiki is enabled. The title of pages is attached to the end.
- **api_path** - The API path for the project, usually /w/api.php
- **opted_in** - A list of database names of projects that will display *restricted statistics* regardless of individual users' preferences

Tools

- **enable.ec** - Enable "Edit Counter" tool
- **enable.articleinfo** - Enable "Article Information" tool
- **enable.pages** - Enable "Pages Created" tool

- **enable.topedits** - Enable "Top Edits" tool
- **enable.blame** - Enable "Article Blamer" tool
- **enable.autoedits** - Enable "Automated Edits" tool
- **enable.adminstats** - Enable "Admin Statistics" tool
- **enable.adminscore** - Enable "Admin Score" tool
- **enable.rfa** - Enable "RfX Analysis" tool
- **enable.rfap** - Enable "RfX Vote Calculator" tool
- **enable.bash** - Enable "Quote Database" tool
- **enable.sc** - Enable "Plain, Dirty, Simple Edit Counter" tool

CHAPTER 4

Opting in to restricted statistics

Some statistics are considered private by some users, such as the times of the day or year that they edit most or the pages they've made most contributions to.

Although the data for these statistics is made available via MediaWiki's API, users must explicitly opt in to make the aggregate forms available in XTools. Alternatively, a whole project can be opted in via the `opted_in` *configuration variable*.

The affected tools are as follows:

- *Edit Counter*:
  - Monthly counts bar chart
  - Timecard punch chart
- *Top Edits*:
  - Top edits per namespace

How to opt in

To opt in, a user must create `User:<username>/EditCounterOptIn.js` on each wiki they want to opt in for. This page should be created with any content (it just has to have *some* content).

To opt in on all projects, they must create `User:<username>/EditCounterGlobalOptIn.js` on the default project (or, in the case of the WMF Labs installation, on Meta Wiki). Again, the actual content of this page is irrelevant.

How to opt out

To opt out the relevant user page (single-wiki or global; see above) should be blanked or deleted.

CHAPTER 5

Tools

Edit Counter

The edit counter tool provides detailed summary statistics about a single user on a single project.

General Statistics

The general statistics section contains lots of statistics about the user and their work on the project, as well as some data about other projects that they're active on.

Firstly, some basic **user information**: ID, username, and group membership (including globally, if CentralAuth is installed).

Then, **Edit counts** are displayed for:

- the last day, week, month, year, and all time (the latter also including addition counts of deleted edits);
- edits made with or without comments;
- edits that have been deleted;
- small (under 20 KB) and large (over 1000 KB) edits;
- minor/non-minor edits (as recorded by the user); and
- what semi-automating tools they used to edit.

Also, dates of activity on the project (earliest and latest) are displayed, and what this duration is in days.

Averages (per day) are given for some of the above metrics.

Next, **Page counts** are shown:

- pages created, imported, moved, deleted, and undeleted;
- total number of unique pages edited.

And lastly, **Log counts** are summarized:

- the number of times the user has thanked another user;
- pages reviewed, patrolled, protected, and unprotected;
- users blocked and unblocked;
- files uploaded (and also those uploaded to Commons, for the WMF Labs installation).

Namespace totals

Total edit counts in each namespace (from all time): a table ordered in decreasing number of edits; and a pie chart showing the relative number of edits.

Timecard

A 'punchcard' chart showing what days of the week and hours of the day the user made most edits. The times are given in the timezone of the XTools user (and keep in mind that the editing user may also have edited in multiple different timezones).

Year counts

A bar chart showing total edit counts made in each year, with each bar being divided into namespace sections so that it's possible to get an idea of how a user's namespace activity has changed over the years.

Month counts

The same as the year counts, except the columns are months instead of years.

Latest global edits

A list of the user's thirty most recent edits from all projects.

Automated edits

A summary table of the number of edits the user has made with any of the known semi-automated editing tools, sorted in decreasing order.

Top Edits

This tool queries a single project and displays

- a user's most-edited articles in one or all namespaces; or
- all of a user's edits on one article (in chronological order).

Simple Counter

The Simple Counter is a quicker way than *Edit Counter* to get a brief overview of a user's contributions.

It displays a user's total number of edits (live, deleted, and a grand-total), as well as their username, ID, and group membership.

CHAPTER 6

Development

To contribute to the development of XTools, you may fork us on GitHub. A few things to be aware of first:

1. XTools is based on Symfony 3. We use Twig as our template engine. Symfony is a full MVC system. a. The controllers are located at `src/AppBundle/controller`. They are sorted by tool. b. The twig templates are located at `app/resources/views`. They are sorted by tool.
2. We use the `@Route` syntax to configure routes.
3. Every tool requires a twig directory and one controller. Also, core parts of XTools require the tool to be registered within *app/config/tools.yml*.

Style Guidelines

- It's called "XTools", with two capital letters.
- XTools conforms to PSR2 coding standards; use `./vendor/bin/phpcs` to check your code.
- Functions and routes must begin with the tool name.
- Version numbers follow Semantic Versioning guidelines.

Running Development server

First make sure you meet the *Pre-requisites*, and then follow these steps:

1. Clone the repository: `git clone https://github.com/x-tools/xtools-rebirth.git && cd xtools-rebirth`
2. Run `composer install` and answer all the prompts.
3. Create a new local database: `./bin/console doctrine:database:create` (or `d:d:c`).
4. Run the database migrations: `./bin/console doctrine:migrations:migrate` (or `d:m:m`)

5. Launch PHP's built-in server: `./bin/console server:run` (or `s:r`).
6. Visit `http://localhost:8000` in your web browser.
7. You can stop the server with `./bin/console server:stop` (or `s:s`)

The *Simple Counter* is the simplest tool and should work as soon as you set up XTools. Test it by going to http://localhost:8000/sc and put in `Jimbo Wales` as the Username and `en.wikipedia.org` as the Wiki. After submitting you should quickly get results.

The development server does not cache data; any changes you make are visible after refreshing the page. When you edit the `app/config/parameters.yml` file, you'll need to clear the cache with `./bin/console c:c`.

Assets can be dumped with `./bin/console assetic:dump`, and if you're actively editing them you can continually watch for changes with `./bin/console assetic:watch`.

The logs are in `var/logs/dev.log`. If things are acting up unexpectedly, you might try clearing the cache or restarting the server.

Developing against WMF databases

If you want to use the WMF database replicas, open two tunnels with:

```
ssh -L 4711:enwiki.labsdb:3306 tools-login.wmflabs.org -N -l your-username-here
ssh -L 4712:tools.labsdb:3306 tools-login.wmflabs.org -N -l your-username-here
```

And set the following in `app/config/parameters.yml`:

```
app.is_labs: 1
database_replica_host: 127.0.0.1
database_replica_port: 4711
database_replica_name: meta_p
database_meta_name: meta_p
database_replica_user: your-uxxxx-username-here
database_replica_password: your-password-here
database_toolsdb_host: 127.0.0.1
database_toolsdb_port: 4712
database_toolsdb_name: toollabs_p
```

(Change the `your-*-here` bits to your own values, which you can find in your `replica.my.cnf` file on Tool Labs.)

Table mappings

Tool Labs has different versions of tables that utilize indexing to improve performance. We'll want to take advantage of that.

- Go to the config directory with `cd app/config`
- Create the file table_map.yml from the template: `cp table_map.yml.dist table_map.yml`
- Set the contents of the file to the following:

```
parameters:
  app.table.archive: 'archive_userindex'
  app.table.revision: 'revision_userindex'
  app.table.logging: 'logging_logindex'
```

Sometimes we want <code>logging_userindex</code> and not the logindex. This is handled in the code via the <code>getTableName()</code> function in [https://github.com/x-tools/xtools-rebirth/blob/master/src/Xtools/Repository.php#L144 Repository.php].

Caching

Caching should happen in helpers, with appropriate times-to-live.

Every helper should extend HelperBase, which has `cacheHas()`, `cacheGet()`, and `cacheSave()` methods. These should be used in this pattern:

```
public function doSomething($input)
{
    $cacheKey = 'something.'.$input;
    if ($this->cacheHas($cacheKey)) {
        return $this->cacheGet($cacheKey);
    }
    $something = 'big query here';
    $this->cacheSave($cacheKey, $something, 'P1D');
    return $something;
}
```

The cache key can be anything, so long as it is unique within the current class (the `cache*()` methods prepend the classname, so you don't have to). The TTL syntax is from the DateInterval class (e.g. `P1D` is one day, `PT1H` is one hour).

The above methods are just wrappers around a PSR-6 implementation, intended to reduce the repetition of similar lines of code. You can, of course, retrieve the underlying CacheItemPoolInterface whenever you want with `$container->get('cache.app')`.

Writing the docs

We use ReadTheDocs; it's great.

To build this documentation locally, you need `python-sphinx` installed, as well as the `sphinx_rtd_theme` theme.

Then, it's simply a matter of runnign `make html` in the `docs/` directory, and browsing to `xtools/docs/_build/html/` to view the documentation.

Documentation sections use the following (standard Python) hierarchy of section symbols:

- # with overline for parts
- * with overline for chapters
- = for sections
- – for subsections

Releases

Before tagging a new release:

- update the version numbers in `docs/conf.py` and `app/config/version.yml`;

- check the copyright year in `README.md`, `docs/conf.py`, and `app/Resources/views/base.html.twig`; and
- update `RELEASE_NOTES.md` with any notable new information for the end user.

Then tag the release (follow the Semantic Versioning guidelines, and annotate the tag with the above release notes) and push it to GitHub.

Lastly, update the `version` and `update` parameters at https://www.mediawiki.org/wiki/XTools

Additional Help

Please contact User:Matthewrbowker or User:MusikAnimal if you need help. Or, you are welcome to visit us on IRC (Direct link - Requires an IRC client).

CHAPTER 7

The tools

Here is a brief overview of all of the tools, with links to more detailed information. See the main menu in the side bar for more.

Edit Counter

Edit Counter provides summary information about a user and their activity on a project, such as the total numbers of certain types of edits; their most-edited pages; what semi-automating tools they've used to edit; and lots more. *Read more about Edit Counter...*.

Admin Score

Find out how admin-worthy a user is. Read more about Admin Score....

Admin Stats

Statistics about administrators' actions. Read more about Admin Stats....

Article Info

Get various statistics about the history of a page. Read more about Article Info....

Auto Edits

Explore the edits made by various semi-automated editing tools, from the point of view of pages or of users. Read more about Auto Edits....

Bash

A collection of humourous or insightful quotations about MediaWiki. Read more about Bash....

Blame

Find out who last changed a given part of a page. Read more about Blame....

EC

Edit Counter again? Read more about EC....

Pages

Information about pages that have been created by a user. Read more about Pages....

RFA

RFA Read more about RFA....

RFAP

@TODO Read more about RFAP....

Simple Counter

A simpler but quicker way to view edit counts (than Edit Counter, above). Read more about Simple Counter....

Top Edits

View the pages that a user has edited most often, or all of their edits on one page. *Read more about Top Edits....*.

CHAPTER 8

Help

For more help with XTools, there are several places you can ask:

- IRC (direct link requires an IRC client) — to chat with the developers and other users.
- Phabricator — if you've found a bug.