Wikibot

Tom Rochette <tom.rochette@coreteks.org>

December 21, 2025 — 77e1b28a

0.1 Context

0.2 Learned in this study

0.3 Things to explore

- Grammar/spelling/syntax check
- Check for dead links
  - Suggest replacements
- Find duplicate articles
- Merge duplicate articles
- Detect spam edits
- Suggest similar articles
- Monitor articles

1 Overview

1.1 High level overview

A bot must have various low level functions such as

- fetching (read) the content of an article
- submitting (write) the content of article
- improve the content
  - add references
  - add relevant information
  - prune dead links
  - remove advertisement/vandalism
  - normalize content

2 Difficulties to overcome

2.1 Wiki markup vs HTML

Wikipedia articles are written in Wiki markup, which is a custom language which is then translated into HTML.

When a bot fetches the content of a Wikipedia article, it is fetching the Wiki markup text. This means that a bot would have to be able to manipulate this language and not HTML, which may be an issue as it is more difficult to find libraries to read/manipulate Wiki markup than HTML.

As Wikipedia is written in PHP, it is possible to use the code that converts from Wiki markup to HTML and work directly with HTML. It is also possible to use one of the many alternative parsers that can be found

online. However, it is important that if you convert the Wiki markup into something else to process it, then you need to be able to convert it back into Wiki markup if you want to submit your changes to Wikipedia.

3 Analysis of existing bots

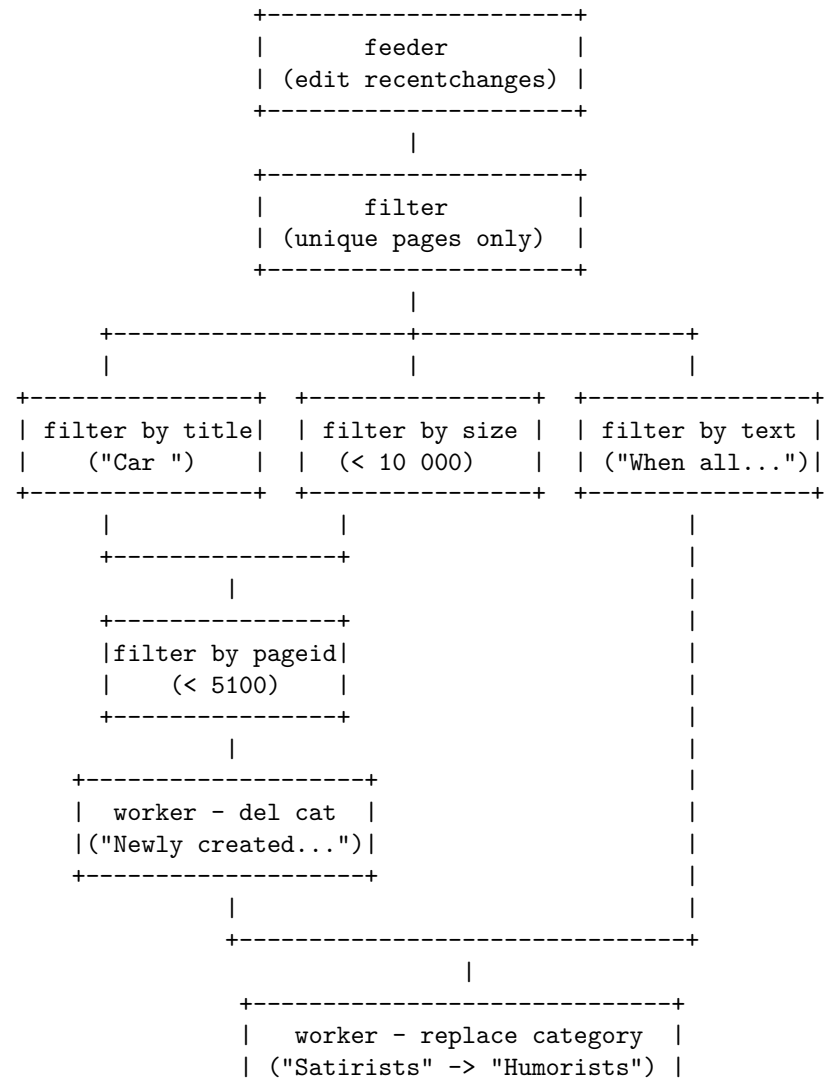
3.1 Apibot

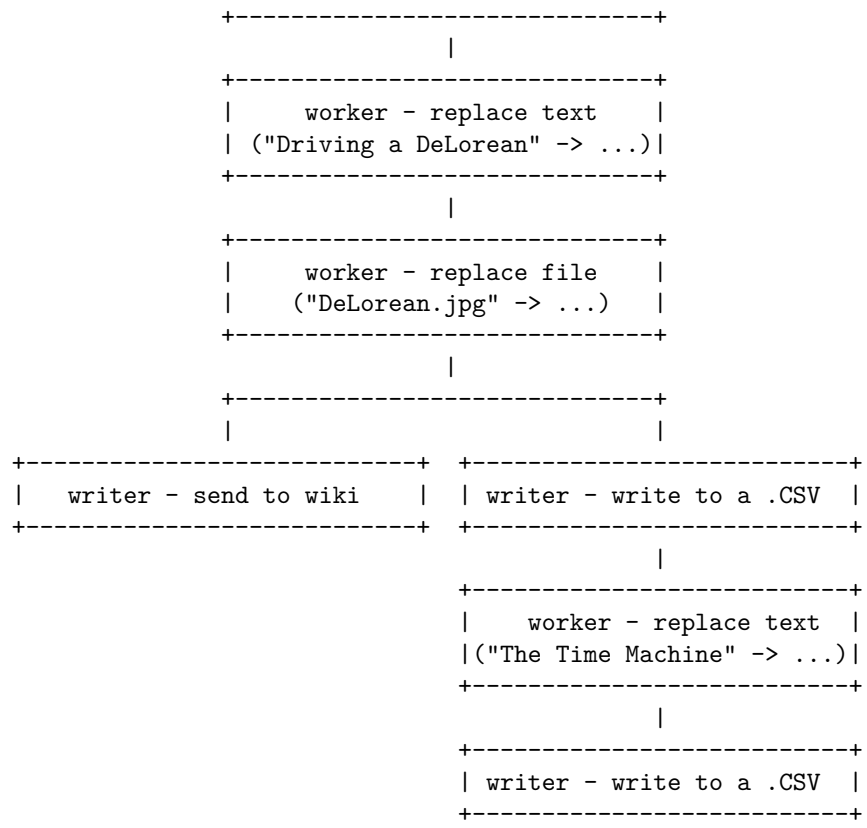
Apibot has a really interesting approach to the problem of processing wikipedia articles. They use what they call an **assembly line**, which is basically the pipeline design pattern.

> Every assembly line object belongs to one of the following types:
> Feeders - supply data to the assembly line
> Fetchers - fetch full wiki objects by some identifier (eg. pages by titles)
> Filters - let through only data that matches given criteria, or reorder the data
> Workers - process the data in one way or another
> Writers - write the processed data (back to the wiki, or to a file, etc.)

Source: http://apibot.zavinagi.org/index.php/Assembly_line_interface

To build an assembly line, you create instances of the objects listed above and you link them to one another by specifying the data source of each object (in other word, the input of the object).





4 See also

5 References

5.1 PHP Bot libraries

- Apibot
- Wikimate
- botclasses.php
- Peachy
- mediawiki-api-base
- mediawiki-api

5.2 Pipeline

- http://pipeline.thephpleague.com/
- http://martinfowler.com/articles/collection-pipeline/

5.3 Pasers

- https://www.mediawiki.org/wiki/Alternative_parsers