tools for developers

Empowering our volunteer developers to write better code that can work across wikis is going to be a key factor in helping us gather the sum of all knowledge. Wikis need code contributors as much as they need content contributors. Templates, gadgets, and bots act as superpowers in making editors more efficient at their tasks. Experienced editors use these tools to create and maintain backlogs, keep track of quality of incoming edits, perform mass actions on pages, ward off vandalism and more. However this superpower is limited to wikis which have contributors able to write code for the site. This creates disparity in the resources available to wikis. Bringing these important resources to all wikis is fundamental to bridging the equity gap across all language wikis.

Sections

Tools 02

Pan-wiki Tools Platform

Empowering our volunteer developers to write better code that can work across wikis is going to be a key factor in helping us gather the sum of all knowledge. Wikis need code contributors as much as they need content contributors. Templates, gadgets, and bots act as superpowers in making editors more efficient at their tasks. Experienced editors use these tools to create and maintain backlogs, keep track of quality of incoming edits, perform mass actions on pages, ward off vandalism and more. However this superpower is limited to wikis which have contributors able to write code for the site. This creates disparity in the resources available to wikis. Bringing these important resources to all wikis is fundamental to bridging the equity gap across all language wikis.

This paper advocates for building a platform that can support tools which work on all our wikis seamlessly. Right now a lot of developer code lives on the wikis (gadgets, Lua modules, templates) where it really isn't possible to do any type of testing, code reviews or debugging; nor is there any straightforward way to add localization or RTL support. This often leads to issues like security vulnerabilities,[1] conflicts with MediaWiki deployed extensions,[2] and bugs due to lack of maintenance. Also, in its current state, having code hosted on the wikis (in a per-project fashion) makes it hard to get in the mindset of having the code work across wikis. It's easy to get sucked into customization and forget to think about things like RTL rendering or localization.

One aspect of the future platform depends on services being available to the developers and communities which they can use for building better tools. These may be services which can be used to do better copyright violation detection, vandalism detection, and image recognition, and provide access to better statistics, and so on. Part of the growth of services involves better partnerships with companies like Google, Turnitin and others providing such services. Another very important aspect of the platform is for Wikimedia engineering to collaborate with our volunteer developer communities to come up with documentation and best practices for creating new tools.

An example of this can be tutorials and guidelines on how gadgets can make use of OOUI [3] to standardize our interfaces and make them more accessible for everyone. Tools that facilitate communication among engineers and volunteer developer communities is key to achieving this goal.

Bots : Potential and Risks

[not edited] Bots are an important part of a wiki's infrastructure. They perform a lot of repetitive tasks that need much effort from the editors.

Bots can can have a big influence in shaping a wiki's community practices and the encyclopedia itself. This can be dangerous because most bots are programmed by individual programmers without a lot of community input.

Gadgets & Gadget Usage

[not edited] 16 gadgets are enabled by default for all users on English WP [4]

8 other gadgets are manually enabled by more than 30,000 active users

The numbers are comparatively low on other projects like German[5], Hebrew,[6] Tamil,[7] Italian [8] etc.

Several prominent gadgets have been adopted by several projects. Like HotCat, Navigation popups, UTCLiveClock, WikEd etc. have been made available on several different projects by volunteers who are familiar with programming.

There is no good way to test gadgets before deploying them. This means that gadgets can break things quite easily for users while being hard to detect as the source of the breakage. They also conflict with MediaWiki extensions at times, causing broken behavior. [9] [10] [11]

Most importantly, gadgets do not go through any sort of review process before being deployed which means they are a potential vector for abuse.

Conclusions & Implications

[not edited] Gadgets are very heavily used on most Wikipedias

Bigger projects (with more language speakers) like English have a lot more gadgets at their disposal than smaller projects.

Knowledge of some basic programming seems is an inhibiting factor for porting gadgets to new wikis. This is because very often gadgets are written in a way that is custom to the wiki they are written for. Internationalizing these gadgets is hard and is seldom paid attention to. Similarly, the UI is also commonly customized to the wiki it's written for and does not work well for RTL/LTR wikis.

We don't currently have any instrumentation in place to gauge how often each individual gadget is actually used in practice. This is important for two reasons:

Such instrumentation is a great way to figure out which tools are desired by the communities which are not available as part of MediaWiki and the extensions. This will allow us to fill that gap by creating tools that work across all our projects well.

Instrumentation is also important because the default enabled gadgets are loaded for every single registered user. This can amount to a significant JavaScript load for every user that needs some investigation to gauge performance impact. [12]

Wishlist & Technical Requests

[not edited] One of the big realizations we have had working on the Community Tech team is that a big chunk of wishlist survey wishes every year already have an existing solution created by the community. Here's a few examples:

- Global cross-wiki watchlists asked for in the 2015 Wishlist survey were

worked on by a volunteer dev who made a tool called crosswatch.[13] It stopped working because of lack of maintenance.

- Pageview stats tool also asked for in the 2015 wishlist survey was already worked on by a volunteer.
- GlobalPreferences extension was built by Legoktm in his volunteer capacity in 2013, much before it got nominated for the wishlist.[14]
- TemplateWizard[15] has several local wiki equivalents - on Hebrew [16] and German [17] wiki, developed by volunteer developers.
- Blame tool has a few existing solutions built by the community already.[18] [19]
- English Wikipedia has a page to request bots for specific tasks.[20] Similar pages exist on other wikis. At a quick glance, one can see that most of the uncontroversial requests are pretty quickly fulfilled by volunteer developers.

Conclusions & Implications

- Volunteer developers are extremely good at helping our communities when they can.
- Often wikis wish something which is already implemented on another wiki as a gadget/bot but there is a communication gap between different wikis as each wiki has its own ecosystem of tools it uses. There is no single place to surface tools used by different wikis. Also when wikis do discover these tools, there is lack of documentation to guide non-technical folks on importing and enabling these tools on their own wikis.
- They have the potential to act as an extended arm of WMF, in helping our communities grow and become better.

Notes

[1] https://lists.wikimedia.org/pipermail/wikitech-l/2018-March/089636.html
[2] https://phabricator.wikimedia.org/T170896
[3] A UI component library created by WMF, aimed at providing a consistent UI experience that works well for all languages. https://www.mediawiki.org/wiki/OOUI
[4] https://en.wikipedia.org/wiki/Special:GadgetUsage
[5] https://de.wikipedia.org/wiki/Spezial:GadgetUsage
[6] https://he.wikipedia.org/wiki/מיוחד:GadgetUsage
[7] https://ta.wikipedia.org/wiki/சிறப்பு:GadgetUsage
[8] https://it.wikipedia.org/wiki/Speciale:GadgetUsage
[9] https://phabricator.wikimedia.org/T178348
[10] https://phabricator.wikimedia.org/T170896
[11] https://phabricator.wikimedia.org/T22134
[12] https://phabricator.wikimedia.org/T142461
[13] https://tools.wmflabs.org/crosswatch/welcome
[14] https://www.mediawiki.org/w/index.php?title=Extension:GlobalPreferences&oldid=838650
[15] https://www.mediawiki.org/wiki/Help:Extension:TemplateWizard
[16] https://he.wikipedia.org/wiki/מדיה_ויקי:Gadget-TemplateParamWizard.js
[17] https://de.wikipedia.org/wiki/Wikipedia:Technik/Skin/Gadgets/Vorlagenmeister
[18] https://github.com/wikiwho/WhoColor
[19] http://wikipedia.ramselehof.de/wikiblame.php
[20] https://en.wikipedia.org/wiki/Wikipedia:Bot_requests

Sources

N. Kohli, R. Kaldari and J. Matazzoni : Research and Insights, Other contributors[1]: A. Amaroni, A. Baso, B. Davis, A. Halfaker, J.Hare, D. Horn, R. Isler, J. Katz, J. Minor, T. Negrin, M. Novotny, N. Pangarkar, N. Wilson,

Critical Point of View: A Wikipedia Reader - The Lives of
Bots by R. Stuart Geiger https://en.wikipedia.org/wiki/Special:GadgetUsage

[1] If your name was left off the list by mistake please contact JMinor or MNovotny