Parsoid

Quarterly review Q1 2013/2014

Agenda

- Our objectives
- Progress Q1 2013/14
- High-level goals for 2013/14
- Tasks Q2
- Questions and discussion

We deal with Wikitext,
so you don’t have to.

Our objectives

Make it easy and efficient to view, reuse, and edit content.

1. Convert wikitext content faithfully to semantic HTML+RDFa
2. Support HTML editing without dirty wikitext diffs
3. Use HTML in MediaWiki core
4. Support wikitext editing with Parsoid

Goals Q1

- Image editing refinements [straightforward]
- Provide public HTML API [straightforward]
- Research: Language variant support [hard, Q1-Q2?]
- Research: Support switching between HTML and Wikitext within one edit [hard, Q1-Q3?]

Goals Q1

- HTML / Wikitext compound storage; support Flow [medium, Q1-Q2]
- Enforce proper nesting of transclusions [hard, Q1-Q2]
- Testing infrastructure improvements [straightforward, Q1-Q2]
- Performance: More efficient template updates [straightforward]

Progress Q1

- Image editing refinements [straightforward]
  - Some progress, but not yet done
  - Still straightforward

- Provide public HTML API [straightforward]
  - Simple public API being deployed
  - Longer-term storage API: Rashomon

Progress Q1

- HTML / Wikitext compound storage; support Flow [medium, Q1-Q2]
  - Storage API, named “Rashomon” (née “Storoid”)
  - Both internal and external service
  - Uses Cassandra as (initial) backend
    - Test results so far encouraging
  - Buy-in from ops, analytics
  - On track for Q2

Progress Q1

- Research: Language variant support [hard, Q1-Q2?]
  - Relatively straightforward plan for basic editing support: Q2
  - Will require some wikitext fixup of unbalanced constructs
  - Also ideas for longer-term plan, but low priority

Progress Q1

- Research: Support switching between HTML and Wikitext within one edit [hard, Q1-Q3?]
  - Work underway for stable element id preservation
  - elegant solution, also handles cross-{page,wiki} copy & paste
  - but hard

Progress Q1

- Enforce proper nesting of transclusions
  [hard, Q1-Q2]
  - Infrastructure now in place
  - Realized that we'll also need to enforce some content model constraints (<p>-in-<p>, <a>-in-<a>) and possibly some syntactic constraints
  - Trade-off between flexibility and ease of editing:

```
* Foo
{{echo|* Bar}}
* Baz
```

Progress Q1

- Testing infrastructure improvements
  [straightforward, Q1-Q2]
  - Round-trip testing (Marc!)
    - SQLite to MySQL port + other fixes lets us RT 160k pages overnight
    - New performance tracking feature caught several performance regressions
    - More work on RT testing needed, some as OPW project
  - Next step: test full stack including web API middleware
  - Better VE / Parsoid integration testing (betalabs)

Progress Q1

- Performance: More efficient template updates [straightforward]
  - Blocked on revision storage, nesting and content model constraint enforcement
  - Likely Q3

Progress Q1: Post-release work

- Cleanup and technical debt
  - Lot of bug fixes
  - Improved support for complicated template uses
  - DOM spec cleanup
  - Removed internal HTML5 fork (Arlo!), contributed changes back to several upstream projects
  - Couple of ancient bugs fixed in PHP parser (Scott) which makes it compatible with Parsoid.

Progress Q1: Bonus features

- Split job queue
  - Prioritize page edits over template/image updates
- Serialize to XHTML5
  - Easier postprocessing with XML libraries
  - Useful for section editing in VE + mobile views
- Mathoid
  - MathML + SVG rendering using MathJax on node.js and phantomjs

Progress Q1: More users

- Flow
  - APIs for (non-page) wikitext/HTML handling
  - Help with architecture
- Kiwix
  - New labs VMs provisioned
  - Exported the French Wikipedia
- PDF rendering
  - First investigations using phantomjs and Parsoid DOM

High-level goals for 2013/14

- Continue to support VE with editing features, bug fixes, etc.
- Start to leverage HTML in MediaWiki core
  - HTML and page property storage (also for Flow?)
  - HTML diffing, basic authorship maps
  - Parsoid HTML for page views (stretch goal)
- Investigate HTML-based templating

See https://www.mediawiki.org/wiki/Parsoid/Roadmap

Tasks Q2: Features

- Image editing refinements [Q2]
- Implement basic language variant support [Q2]

Tasks Q2: API and storage

- Simple HTML API (public Parsoid cluster)
  [being deployed as we speak]
- Rashomon storage API [prototype testing, Q2]
  - Long-term public HTML API
  - Likely using Cassandra backend
  - Needed for performance, mobile, new metadata

Tasks Q2: Rich metadata & wikitext

- Research: Preserve rich HTML metadata across wikitext edits [hard, Q2-Q3]
  - Settled on unique id attributes
  - Enables copy & paste across pages, wikis while preserving arbitrary associated metadata
    - authorship maps
    - annotations
    - internal data like data-parsoid

Tasks Q3 2013: DOM

- Parse most transclusion parameters to DOM once type info is available [medium, likely Q2]
  - blocks on TemplateData (in progress)
- Enforce proper nesting of transclusions [hard, Q2/Q3]
  - Also enforce content model constraints
  - More focused editing
  - Improves performance through expansion reuse
- Getting editor community input / buy in
- OPW tasks

Tasks Q3 2013: Ongoing work

- Testing infrastructure improvements
  - Better integration testing of full stack
- Compatibility: iterate on long tail
- Performance

Other tasks on the horizon

- HTML-only wiki support [hard, Q3-Q4]
- Non-Wikipedia projects [likely hard, Q3]
- DOM-based templating [hard, Q3/Q4]
- Use Parsoid HTML for all page views [hard, stretch goal, Q4]

Thanks!