Mediawiki Schema Design Problems

Jaime Crespo

Wikimedia Developers Summit 2017

-San Francisco, 10 Jan 2017-

© 2017 Wikimedia Foundation & Jaime Crespo. http://wikimedia.org. License: CC-BY-SA-3.0

Myths: Let's rewrite Mediawiki from 0

- Mediawiki works more than fine in 899/902 wikis hosted at the Wikimedia Foundation
- Of course you can do it, it is "only" 400K lines of PHP on core
  - You will only **kill mediawiki, its extensions** (where the real deal is), and its ecosystem, with **all the wikis it hosts in the process**

© 2017 Wikimedia Foundation & Jaime Crespo. http://wikimedia.org. License: CC-BY-SA-3.0

Myths: The problem is the technology

- I will complain about the size of certain tables, but MySQL is not the limiting factor
  - External storage has 1.3 TB tables, but has no lag and no query issues because it is accessed in a key-value model
- **How we access the storage**/the storage model in a relational context is the problem

© 2017 Wikimedia Foundation & Jaime Crespo. http://wikimedia.org. License: CC-BY-SA-3.0

The main issue is optimizing 3 wikis

- enwiki
- commonswiki
- wikidata

© 2017 Wikimedia Foundation & Jaime Crespo. http://wikimedia.org. License: CC-BY-SA-3.0

Some peculiarities on those wikis:

- Large number of revisions
- High rate of revision creation
- Large number of images/image usages
- Large number of template usages (License templates)
- Large number of internal and external links counters
- Some accounts have large number of edits (bots)

© 2017 Wikimedia Foundation & Jaime Crespo. http://wikimedia.org. License: CC-BY-SA-3.0

Problem #1: Basic model over the years

- “Pages”
- Cur + old
- Page + revision + text + external storage
  - Good! We have solved the problem of having large amounts of wikitext in the same table
- Revision special partitioning hack for ‘contributions’ group
  - Ugh!

© 2017 Wikimedia Foundation & Jaime Crespo. http://wikimedia.org. License: CC-BY-SA-3.0

Problem #1: large & busy[1] revision table

- Many queries do large range scans on it
  - Not all rows are really that active
- Complex filtering & ordering options create suboptimal plans
  - Almost impossible to have good indexes for all potential options
- Large tables, specially if busy, are difficult to maintain (e.g. for schema changes)

[1] 1/3 of all query traffic uses this table

© 2017 Wikimedia Foundation & Jaime Crespo. http://wikimedia.org. License: CC-BY-SA-3.0

Problem #1: It is getting worse

- With all talks about multi-content revision, wikidata, structured content, smart category editing, the revision metadata problem will get worse
- We need a different model to accommodate 10x the number of revisions
- Forget about storage, think how we are going to **query metadata efficiently**

© 2017 Wikimedia Foundation & Jaime Crespo. http://wikimedia.org. License: CC-BY-SA-3.0

Problem #1: Some ideas towards a solution (I)

- Partition vertically/normalization:
  - Separate “comment” on a separate table
    - Possible deduplication + nice for other wishlist
  - Normalize “user”
    - More on that later
  - Separate individual columns + indexes
    - e.g. (page_id, revision_id, order)
    - Only makes sense in the context of horizontal part.

© 2017 Wikimedia Foundation & Jaime Crespo. http://wikimedia.org. License: CC-BY-SA-3.0

Problem #1: Some ideas towards a solution (II)

- Horizontal partitioning
  - It has to be integrated into mediawiki, and be the same among all servers
  - Preferably, mediawiki-handled rather than MySQL (it has lots of restrictions)
  - Which key? Do we need extra tables for different access methods? Sharding?

© 2017 Wikimedia Foundation & Jaime Crespo. http://wikimedia.org. License: CC-BY-SA-3.0

Problem #1: Some ideas towards a solution (III)

- Functional partitioning
  - It means separating revisions by a non-arbitrary reason, like old vs. new, namespace, type of revision, …
  - We are already doing this with the recentchanges table, should we focus on that -add more fields- to offload more load?

© 2017 Wikimedia Foundation & Jaime Crespo. http://wikimedia.org. License: CC-BY-SA-3.0

Problem #2: Lack of primary keys

- T17441
- Large amount of new functionality blocked on tables not having primary keys – allowing online schema changes and in general maintenance that requires uniquely identified rows

© 2017 Wikimedia Foundation & Jaime Crespo. http://wikimedia.org. License: CC-BY-SA-3.0

Problem #3: Gigantic *link tables

- Commonswiki: 1400 million rows on templatelinks – 220 GB table
- Cebwiki: 6 million rows on page – 1000 million of rows on templatelinks (120GB table)

© 2017 Wikimedia Foundation & Jaime Crespo. http://wikimedia.org. License: CC-BY-SA-3.0

Problem #3: Normalizing titles?

- Potential solution to links tables:
  - Make titles a first class entity with its own table
  - Substitute it with an id on page, *links
- `page: (Page_id: 1, page_title: 12345)`
- `title: (title_id: 12345, title_namespace: 0, title_text: “Batman”)`
- `pagelinks: (pl_id: 54321, pl_to: 12345)`

© 2017 Wikimedia Foundation & Jaime Crespo. http://wikimedia.org. License: CC-BY-SA-3.0

Problem #3: Compact link tables?

- Compact/JSONify/duplicate tables so tables have a predictable number of rows:
  - Template: 42, used on: ‘{123, 124, 125, ...}’
  - Page: 123, uses templates: ‘{42, 12, 53}’
- Limit the max usage list either on query or on storage (for very popular/problematic templates)

© 2017 Wikimedia Foundation & Jaime Crespo. http://wikimedia.org. License: CC-BY-SA-3.0

Problem #4: User_text

- The denormalization may be useful in some cases, but
  - It contributes to a larger revision table
  - It makes almost impossible or very difficult renaming users (which should be trivial) T33863
  - Not sure IPv6 makes the denormalization efficent

© 2017 Wikimedia Foundation & Jaime Crespo. http://wikimedia.org. License: CC-BY-SA-3.0

Problem #5: Too many filtering options

- There are 2 million non-trivial different queries for recentchanges: T101502#1471866
- BTREE indexes allow for multiple '=' and `'IN ()'` comparisons at the same time

© 2017 Wikimedia Foundation & Jaime Crespo. http://wikimedia.org. License: CC-BY-SA-3.0

Problem #6: Too many tables

- S3 has 99% of the objects of all databases, with ~900 wikis, with 78 wikis each

© 2017 Wikimedia Foundation & Jaime Crespo. http://wikimedia.org. License: CC-BY-SA-3.0