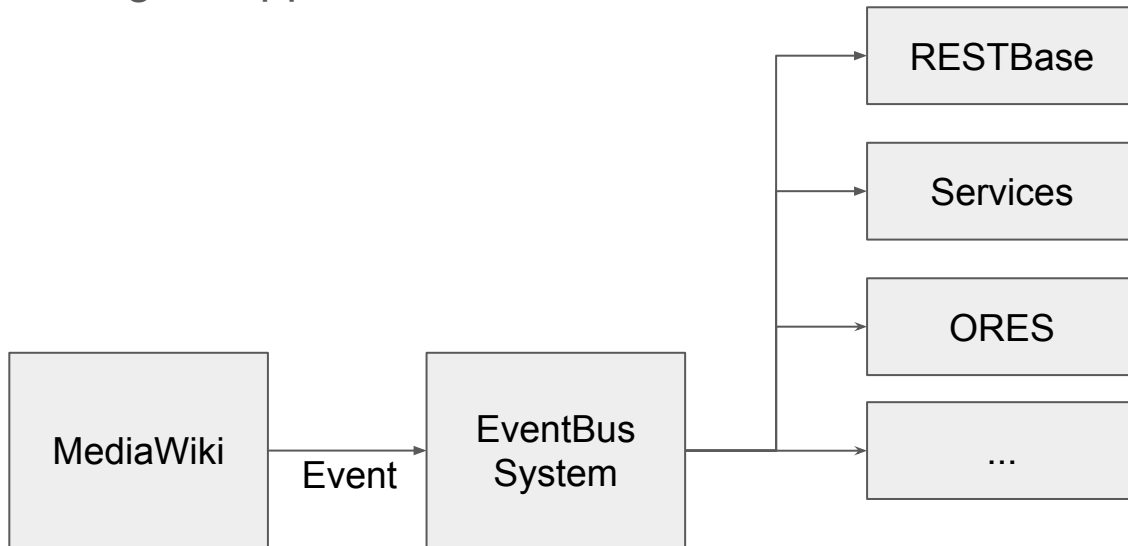


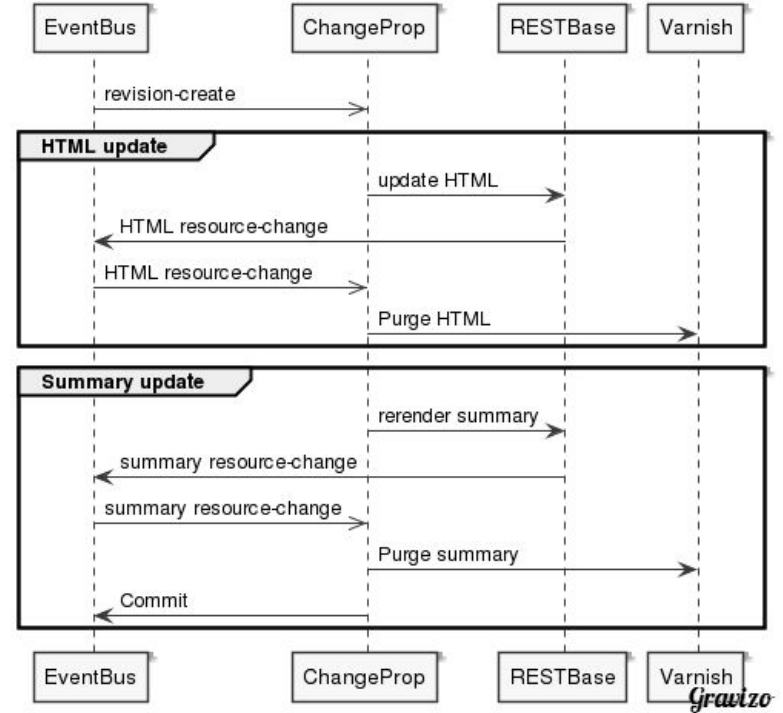
Propagating events between services

- Various kinds of events might happen in MediaWiki
 - revision-create
 - revision-visibility-set
 - page-delete
 - page-undelete
 - page-move
 - user-block-change



Example: RESTBase Dependencies Updates

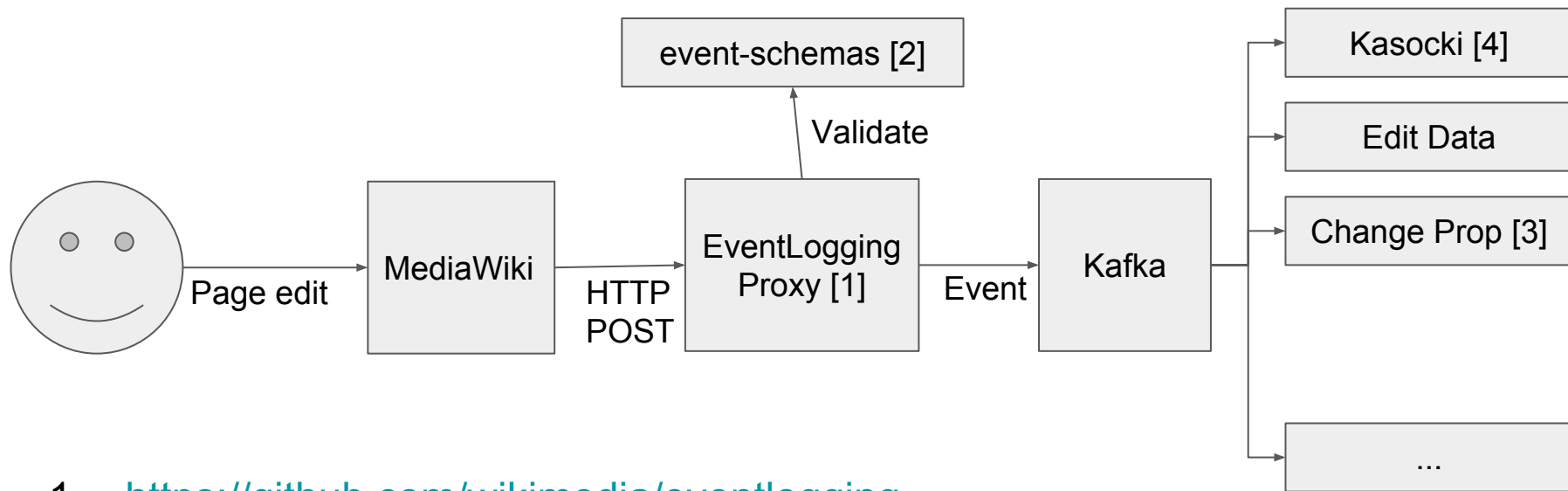
- RESTBase stores different formats
 - HTML
 - Summary
 - Mobile Sections
 - ...
- Representations depend on each other
 - Mobile uses RESTBase HTML
- Update in order by posting and reacting to events in **resource-change** topic



EventBus Requirements

- **Guaranteed *at-least-once* delivery**
- Automatic retries/error handling
- Ability to perform complex multi-stage updates
- Unlimited number of reactions per event
- Simple config-based rule setup
- Scale well to support current & future use cases
 - Currently the possible throughput is ten ~10.000 msg/s, so we have lots of headroom
- MultiDC support out of the box
- Detailed monitoring in [grafana](#)

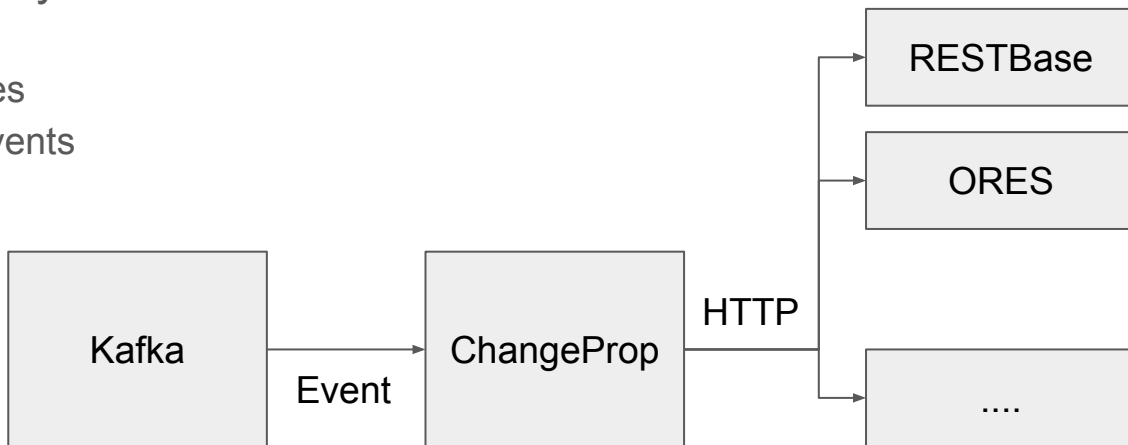
EventBus System



1. <https://github.com/wikimedia/eventlogging>
2. <https://github.com/wikimedia/mediawiki-event-schemas>
3. <https://github.com/wikimedia/change-propagation>
4. <https://github.com/wikimedia/kasocki>

Change Propagation Mission

- Direct consumption from kafka is hard
 - Complex protocol, complex client libs
 - Complex offset management
 - Error handling, retries, concurrency limits, deduplication etc etc
- Change Prop updates systems based on kafka events
 - Update RESTBase
 - Warm-up ORES caches
 - Post ReviewStream events
 - More to come



Change Prop Rule

Update RESTBase when a new page revision is created

```
page_edit:
topic: mediawiki.revision-create
retry_on:
status:
- '5xx'
- 404 # Sometimes occasional 404s happen because of the mysql replication lag, so retry
exec:
method: get
uri: 'https://{{message.meta.domain}}/api/rest_v1/page/html/{{message.page_title}}/{{message.rev_id}}'
headers:
cache-control: no-cache
x-restbase-parentrevision: '{{message.rev_parent_id}}'
if-unmodified-since: '{{date(message.meta.dt)}}'
query:
redirect: false
```