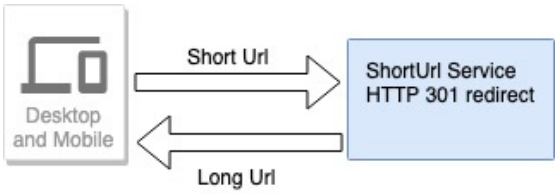
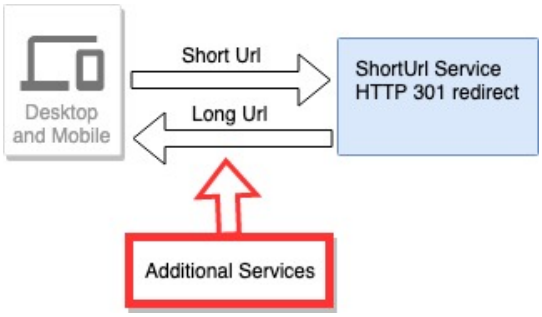
Short URL Service (clone of the bit.ly)

Approach

The architecture of the service is design under the initial assumption that the short URL request will return a 301 redirect status to the browser.





Initially, the service will process two “types” of short URLs, brand-specific or UID key based. The brand URL is processed in the same way as the unique key URL, but the first one could potentially include market information in the headers.

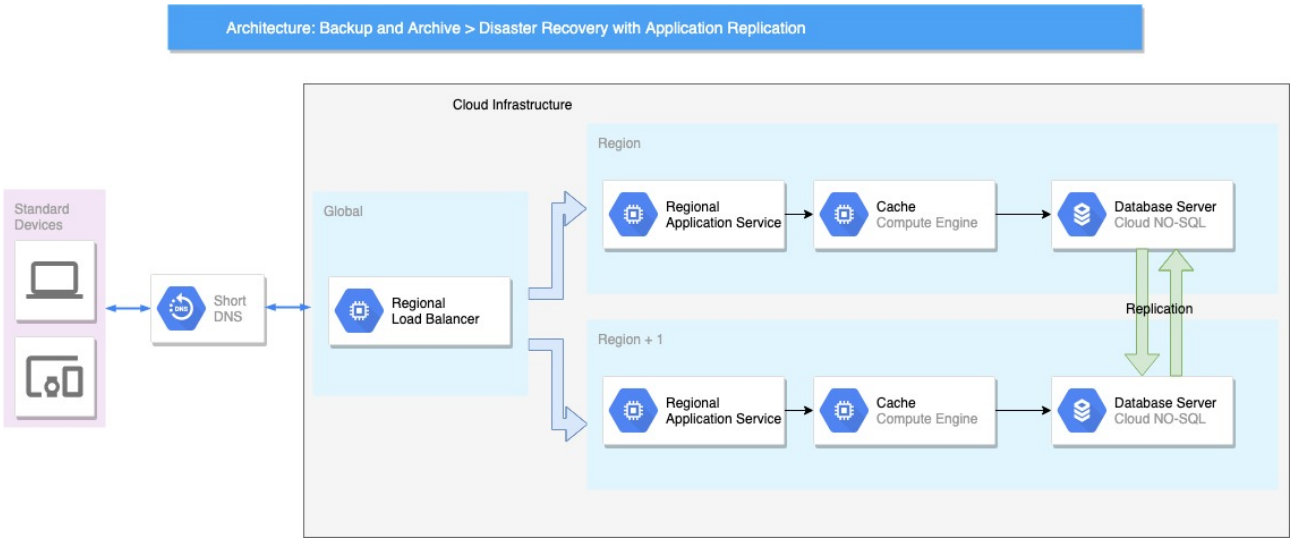
http://big.ly/<brand> *http://big.ly/<ukey>*

The design graph shows the “additional Service” box (see it in red) between the request and the "redirect" response. Any added value should be placed there.

The idea is that URI (context) will work as a key to extract an “object” or “entity” with the long URL. A high performant NoSQL database is a natural option for the implementation.
The diagram also shows a cache which will prevent repetitive request reducing the networking activity.

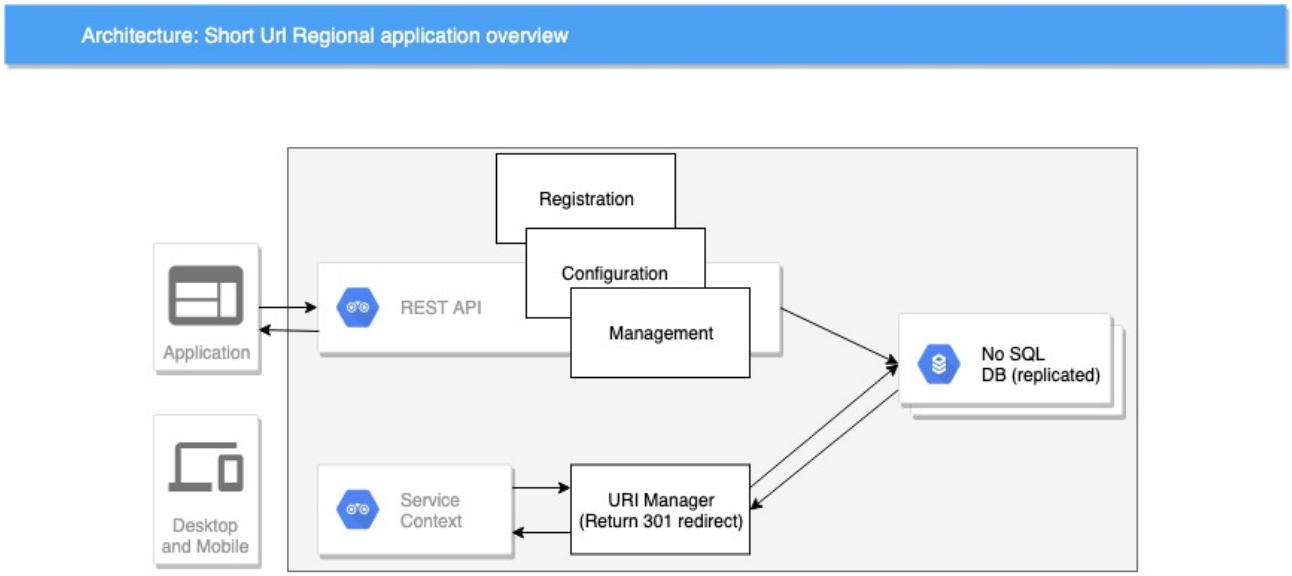
Technical challenges

At the highest level, the first challenge is to make the service worldwide, To handle that, a gateway service should load balance the requests by region.
Each region will host the service (here is the second assumption: the service is hosted in the cloud), and the NoSQL database changes will be replicated across all regions.
Note: The replication service will bring some latency issues.



Suggested ideas

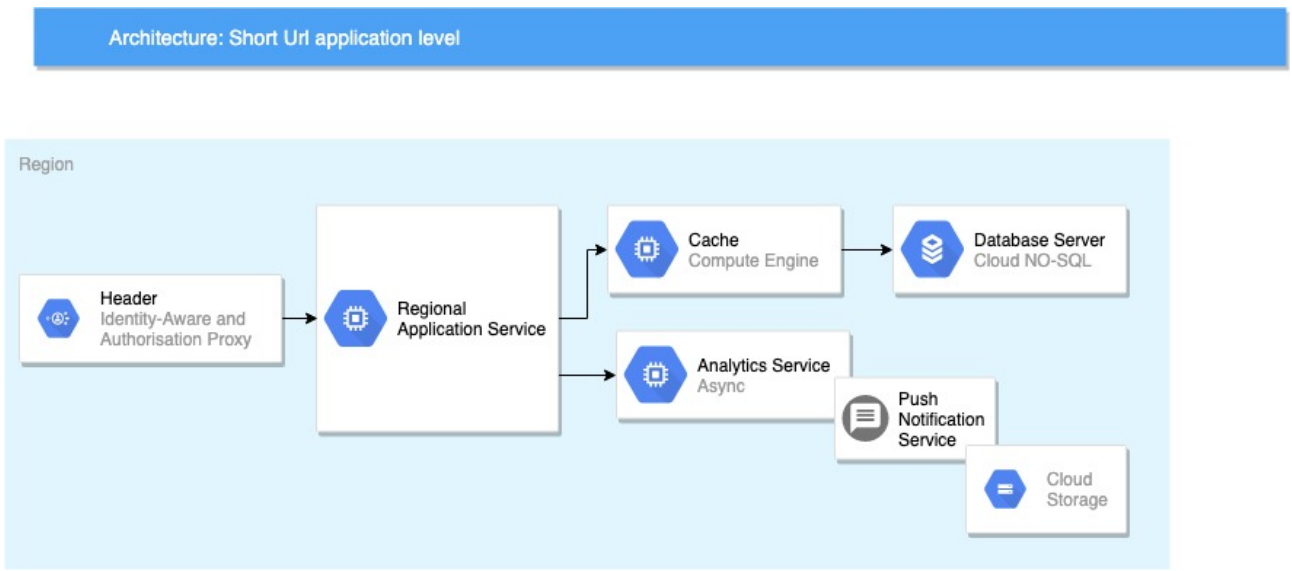
The main topic is to add value to the short URL concept. The added value could be administrated by a web mobile-friendly application, which can register, configure and manage an account with one or many short URLs.



The additional information is to handle the request with headers. Some links could include specific headers which can feed the analytics or **trigger advertisement actions**.

Solution and thoughts

Another aspect is to get authentication and authorisation headers. For the short URL with an expiration date, a **scheduler service** could be set up as part of the management service that will delete key (URI) from the database.



The last aspect (I don't have any more space) to point out is that the analytics service is an async push notification that dumps message into storage.