

Goal: Multitenant, Multilingual, multi-geographical SAAS platform for questionnaire/survey

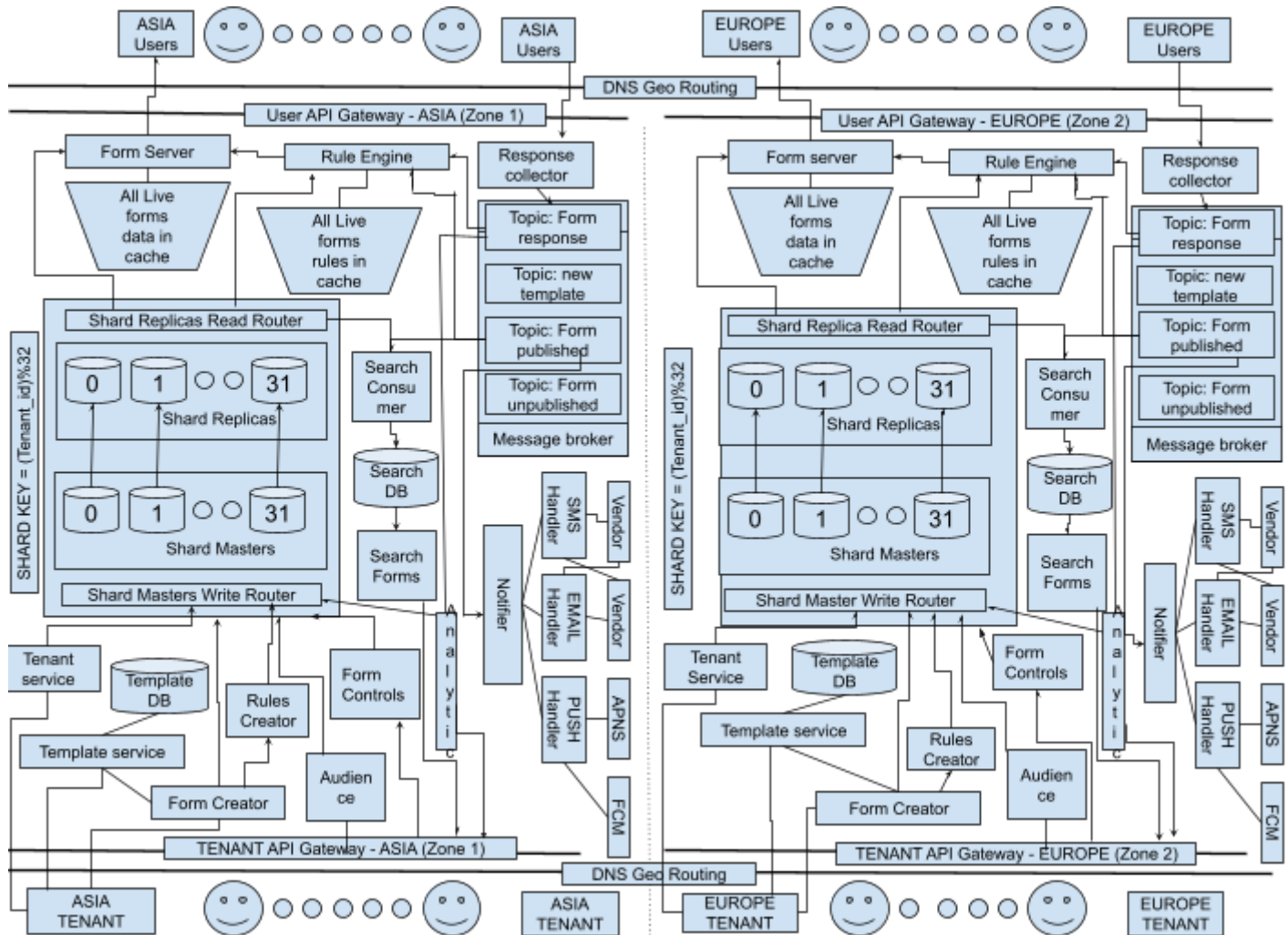
Assumption: Shared database, shared schema, all data for a tenant resides in a single shard

Functional requirements (from the tenant use case lens):

Onboarding, create/view/update/delete/search templates, create/view/update/delete/search form, create/view/update/delete target users, create/view/update/delete rules to populate additional questions based on the last response, edit controls for each form, publish a form to selected/all users, view collected responses, export collected response to 3rd party integrations

Non-functional requirements (from the system architecture lens):

Low latency, High Availability, Fault tolerance, Scalable, Secure



Publish flow: Tenant onboards the SAAS platform via tenant service, can manage his own templates via template service, can create form via form creator and stores form on the basis of the current state of form in an appropriate table in the same shard, can create AND/OR rules to populate next question on the basis of last response using rules creator, can create an audience and can publish the form and notify users via Kafka events, can create an audience, can search forms by status, can edit form wise controls, can view analytics being aggregated automatically by consuming live responses. All writes are happening in shard masters while reads are happening from shard replicas. 3rd party integrations can be done similar to how notifier works.

Data Tables in each shard:

Tenant_onboarding_info
Tenant_forms_indraft
Tenant_forms_Audience

Tenant_forms_live
Tenant_forms_rules
Tenant_forms_analytics

Tenant_forms_completed
Tenant_forms_controls
Tenant_forms_response

(Drafted by Nitin Mishra, 20 Apr 2023)