# Notifications

## Introduction

One of the actions that a user can specify as a part of action handlers is to notify an external entity that an event occurred. The notifications carry a JSON payload that the user supplies from the event handler. Notifications are published to Topics. Topics must be defined administratively in Eventing server before they can be used by function handlers.

## Intended Use

Notifications published by Eventing server are intended to be used in the same tier in which other Couchbase programmatic APIs are used, such as Key Value APIs. This primary use case for the document involves notifying a middle tier entity such as Weblogic.

## Topics

A topic is a named containers of event notification, and specifies both quality of service attributes and requirements of agents that receive notifications.
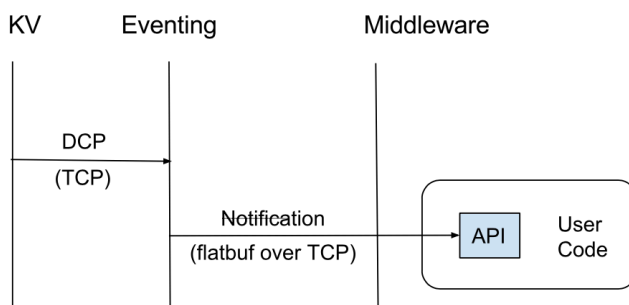
## Quality of Service

Notifications can be configured to be either persistent or ephemeral. Persistent notifications are queued up to configured thresholds while ephemeral notifications are not queued. Persistent notifications do not support broadcast, and require exactly one agent to read them. Ephemeral notifications can be read by any number of agents.

## Filtering

The programming language elements available to event handler on server side provides filtering capabilities and the intent of the project is to optimize the handler execution language to ensure suitable performance of filtering elements that will be expressed in the JS language.

## Flows

Notification flows are similar to existing control flows:

## Transport

Eventing server will provide notifications through APIs to be used by middle tier platforms, and so the transport will itself not form an external interface. The first version will utilize flatbuffer formatted data over TCP to internally deliver events to the APIs.

## Boundary

The feature authoritatively covers the topics of generation of events and delivery of notifications into an API deployed in a middle tier. The following topics are treated as integrations with existing tools:

- Propagation to end agents such as the browsers or displays: Sync Gateway
- Publish/subscribe semantics: Kafka, RabbitMQ, NATS and other popular products
- Further filtering, including parametric filtering in middleware: TBD

## Administrative Interface

Topics can be created from Admin Console and couchbase-cli. In addition, topics can be programmatically created via the eventing API and via OnDeploy event handlers.