



CloudEvents in Iroha 2.0

Nikita Puzankov

Senior Rust Developer
puzankov@soramitsu.co.jp

Iroha 2 updates

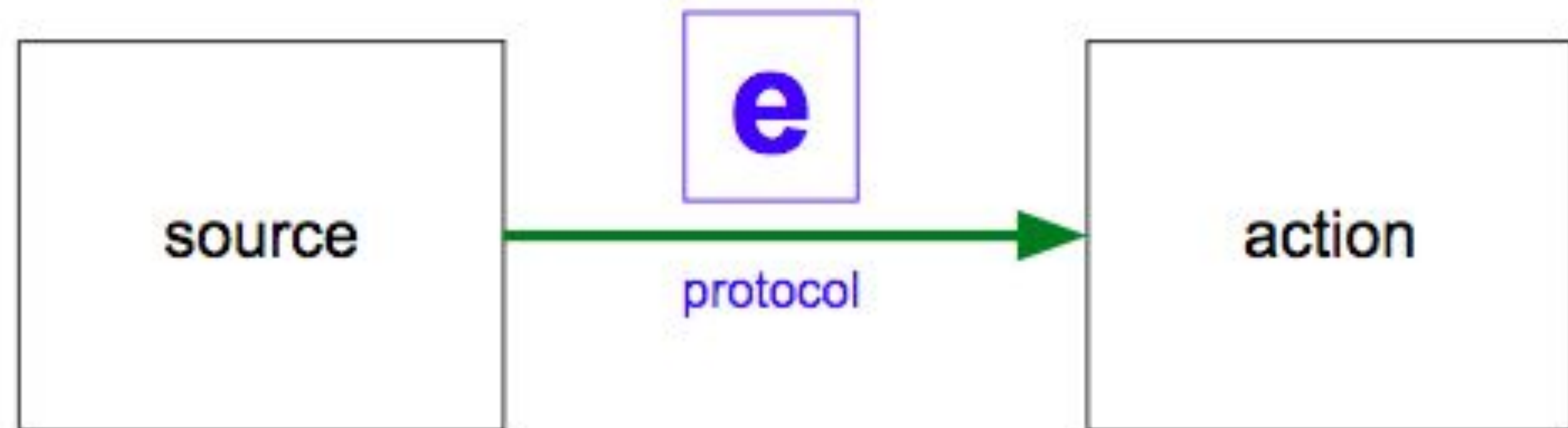
- . Agreement about Artifacts Deployment
 - Docker images on hub.docker.com
 - Binaries on crates.io
 - Technical Documentation on docs.rs
 - User Guide on iroha.readthedocs.io
- . Blocks Synchronization
- . Maintenance Endpoint
- . Bridge Iroha Special Instructions

Goal

... allow for services to be loosely coupled during development, deployed independently, and later can be connected to create new applications.

CloudEvents

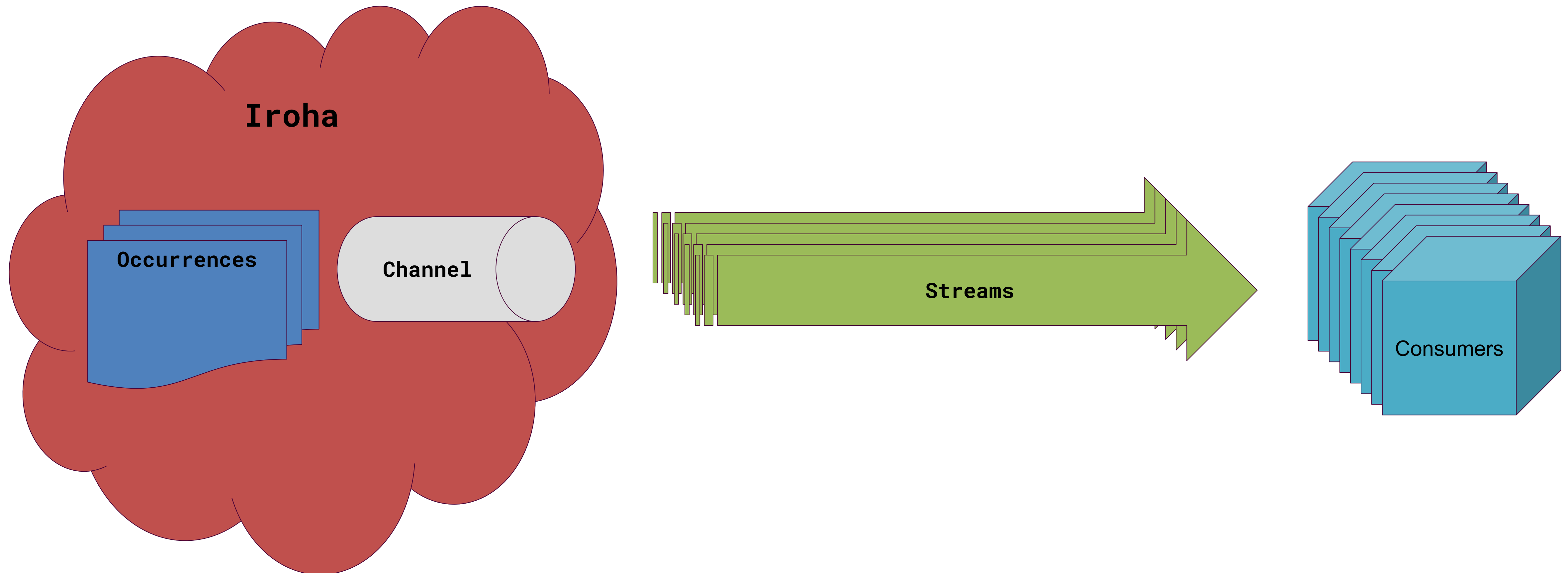
Specification



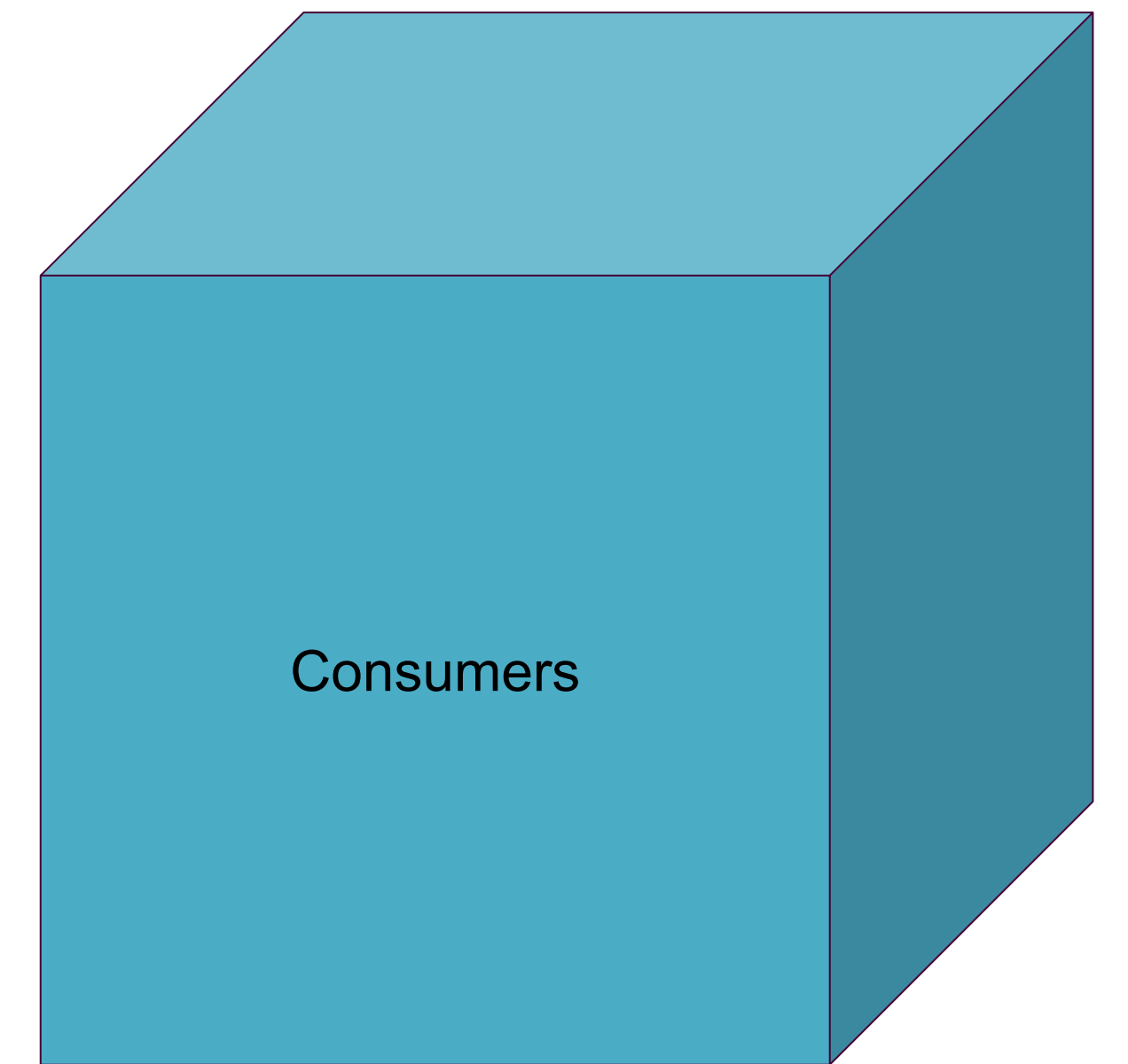
Iroha Maintenance Endpoint

[Initial RFC](#)

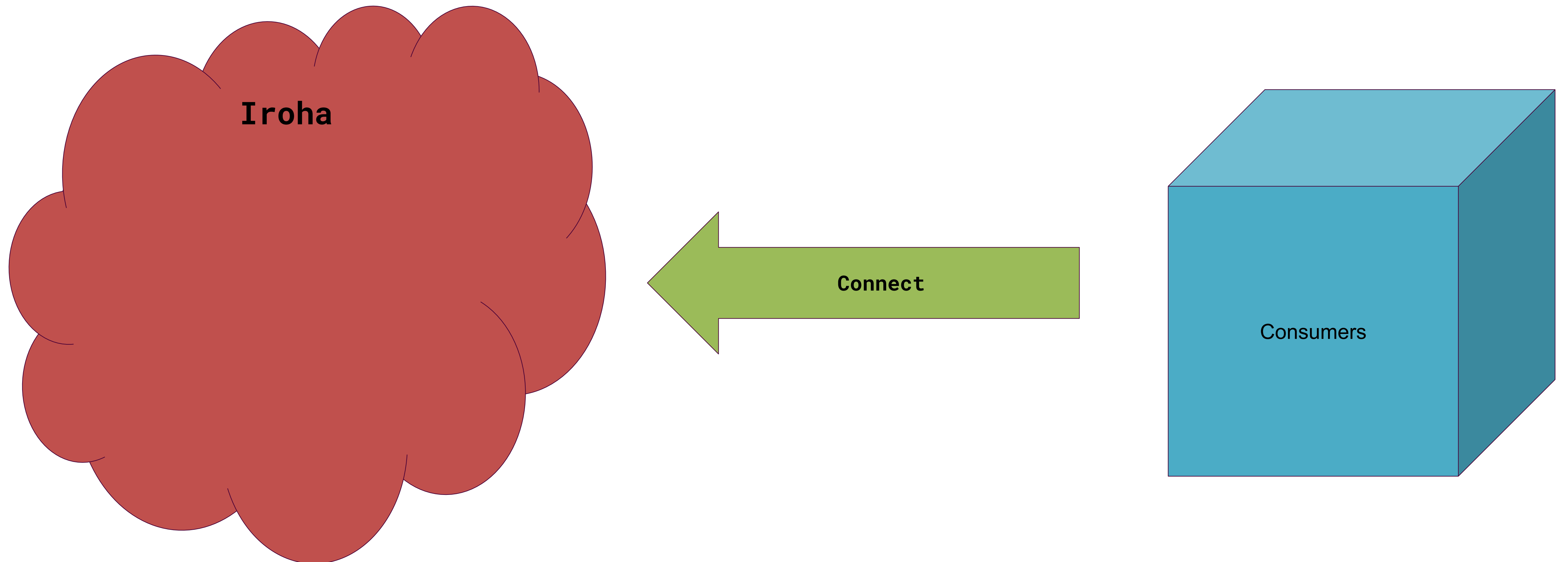
Maintenance Endpoint - Events Listeners



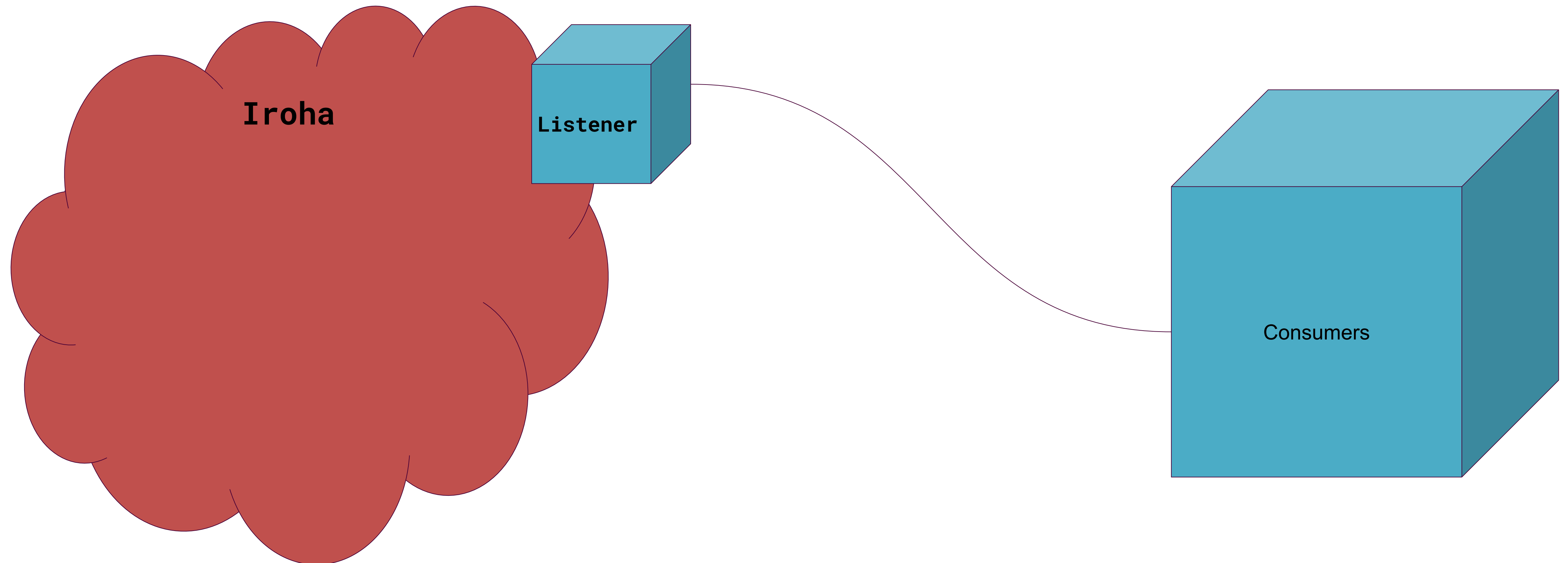
Events Listeners



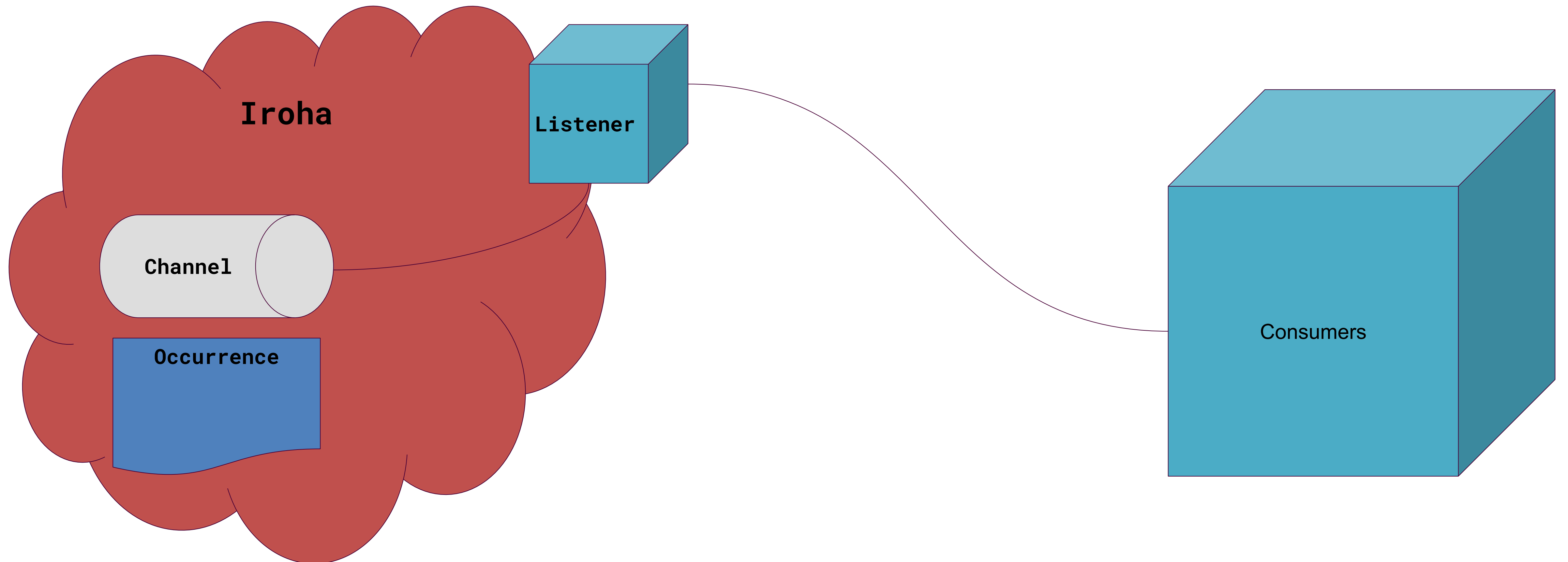
Consumer Connection



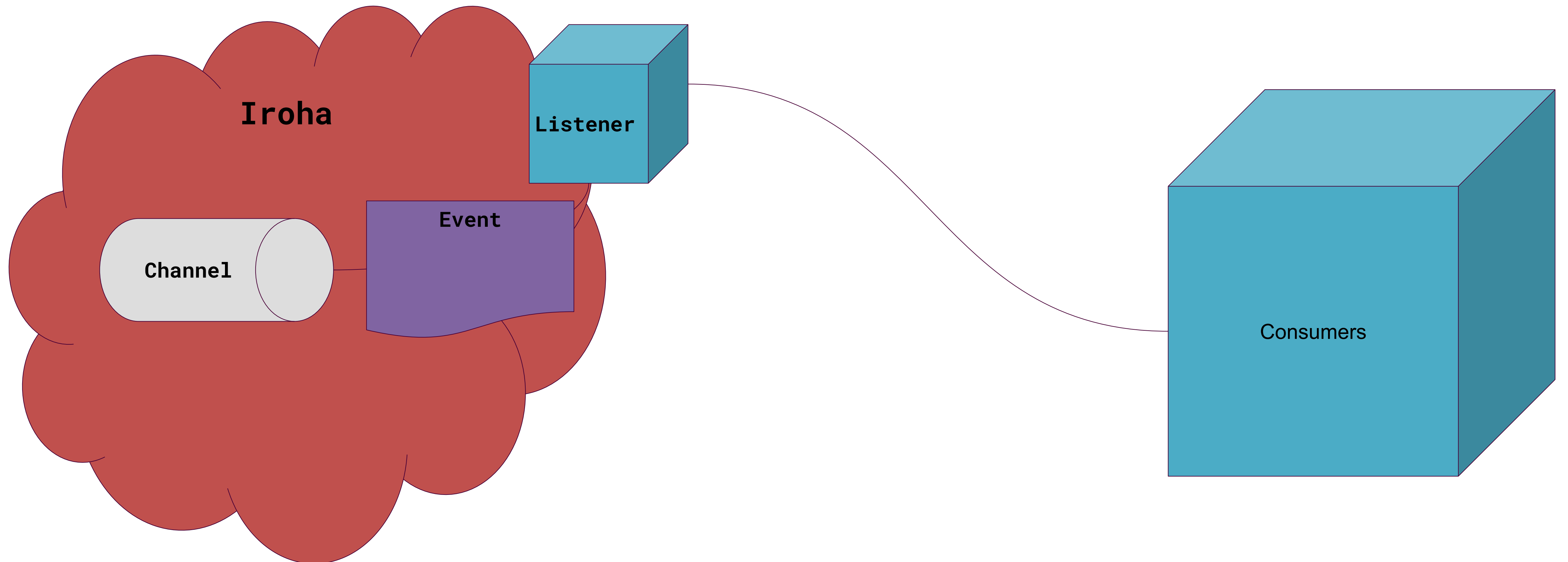
Listener Registration



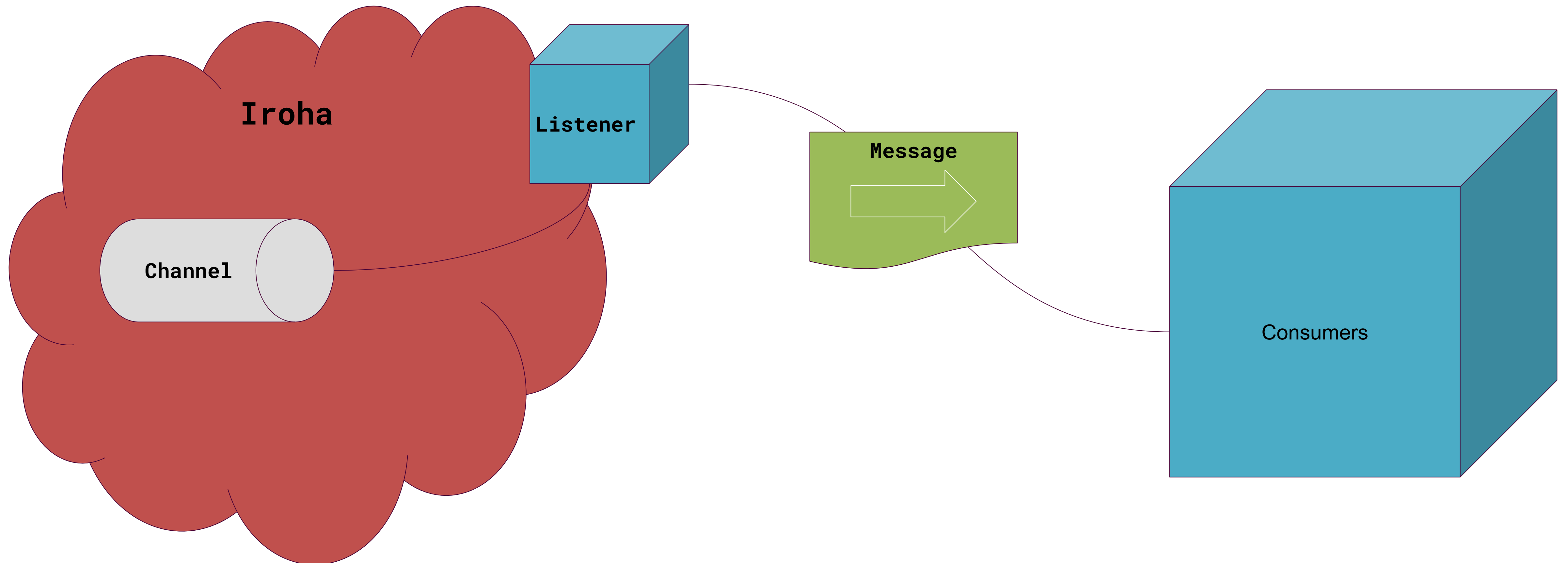
Occurrence



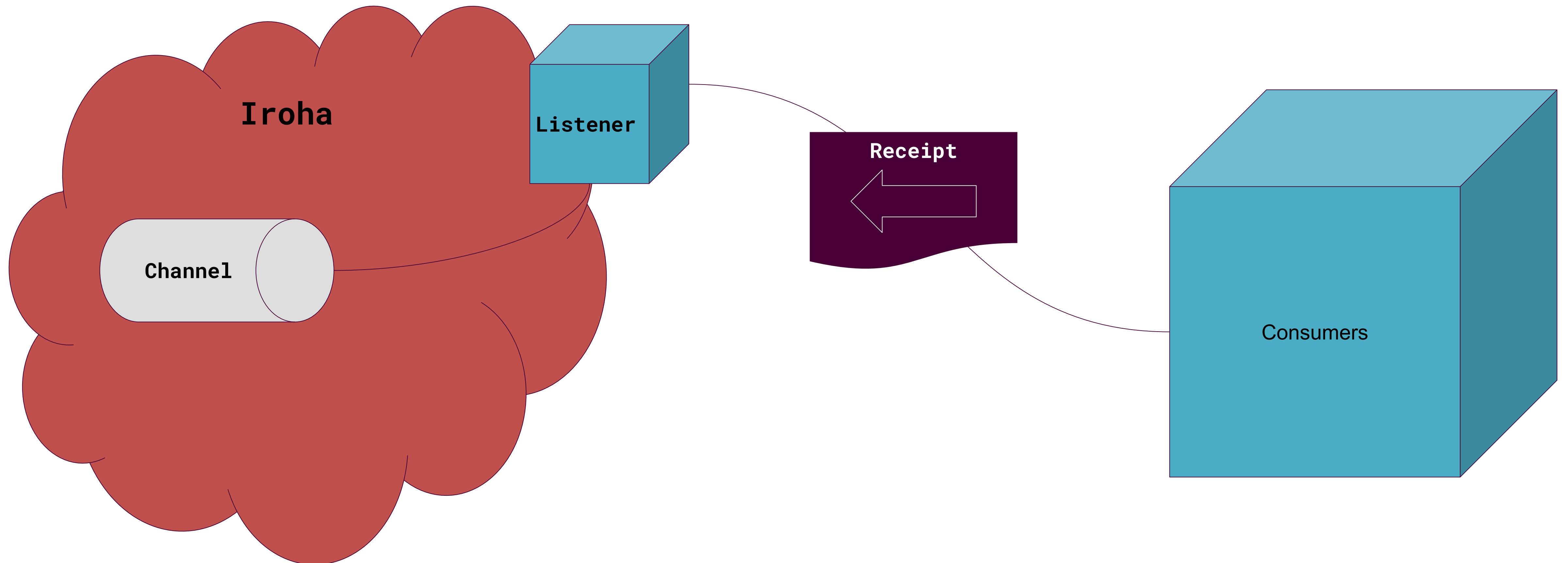
Event



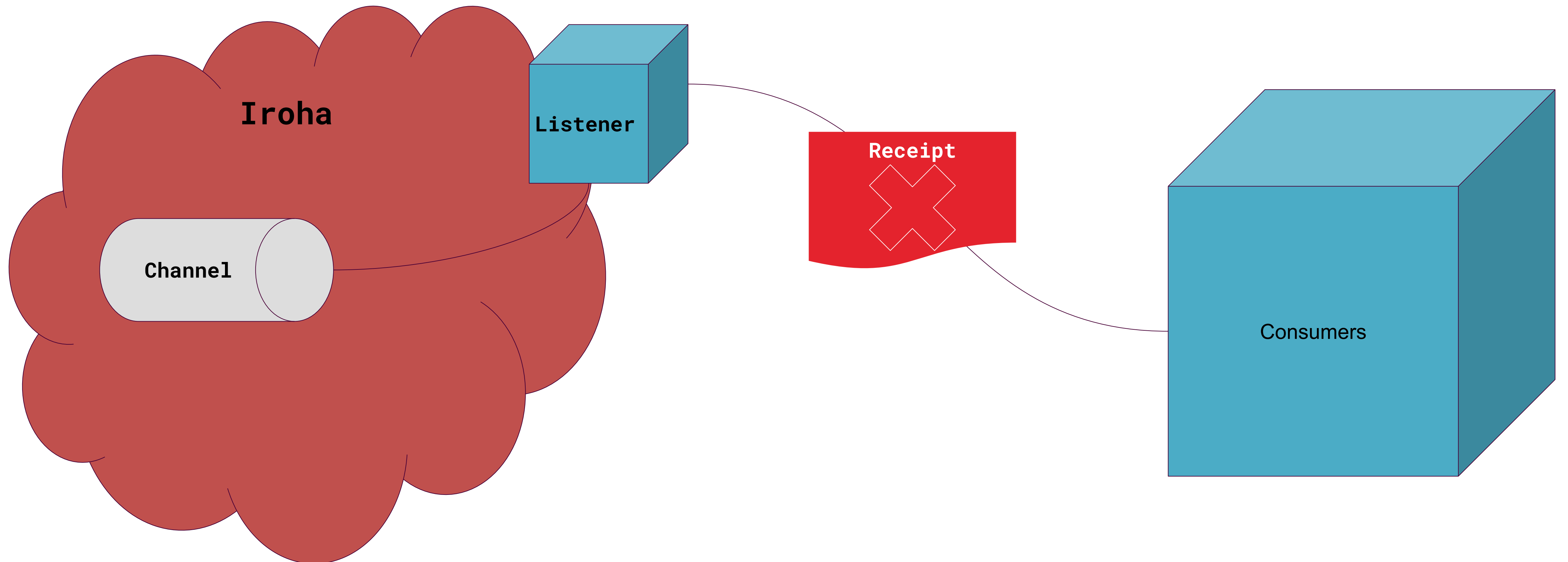
Message



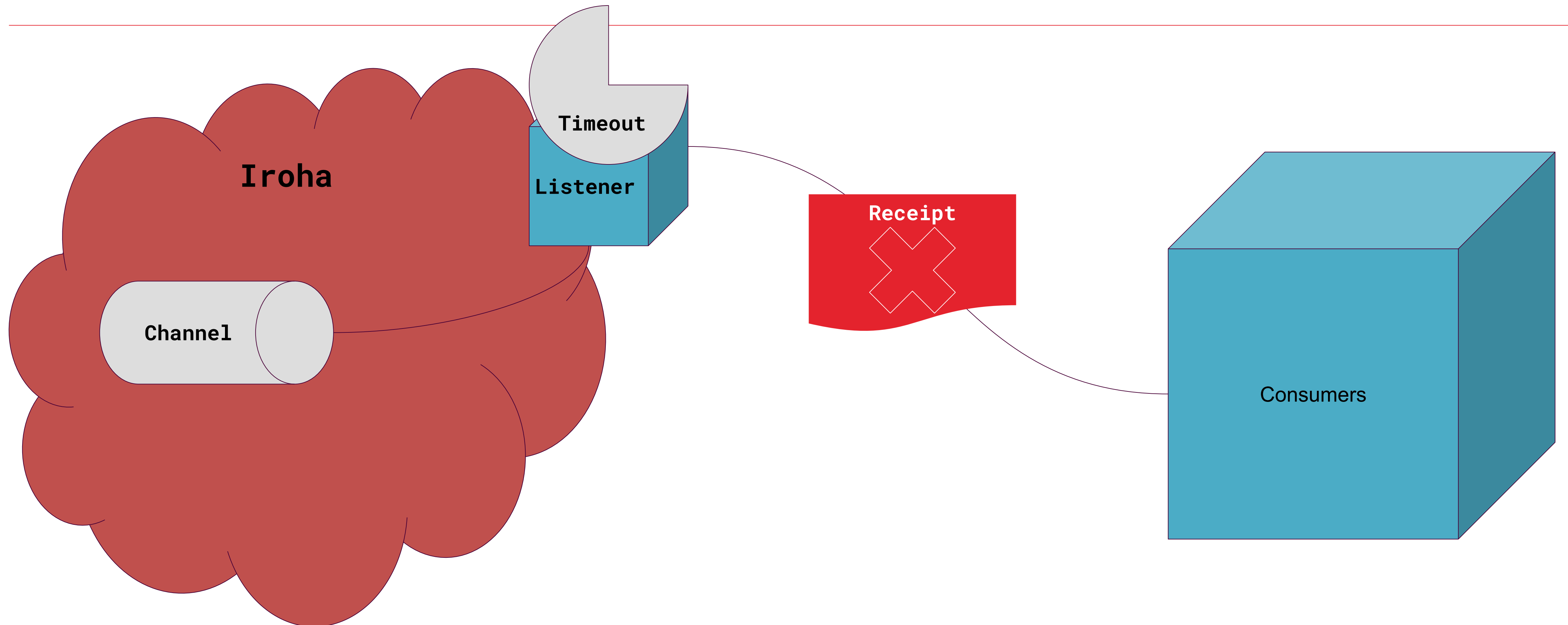
Receipt



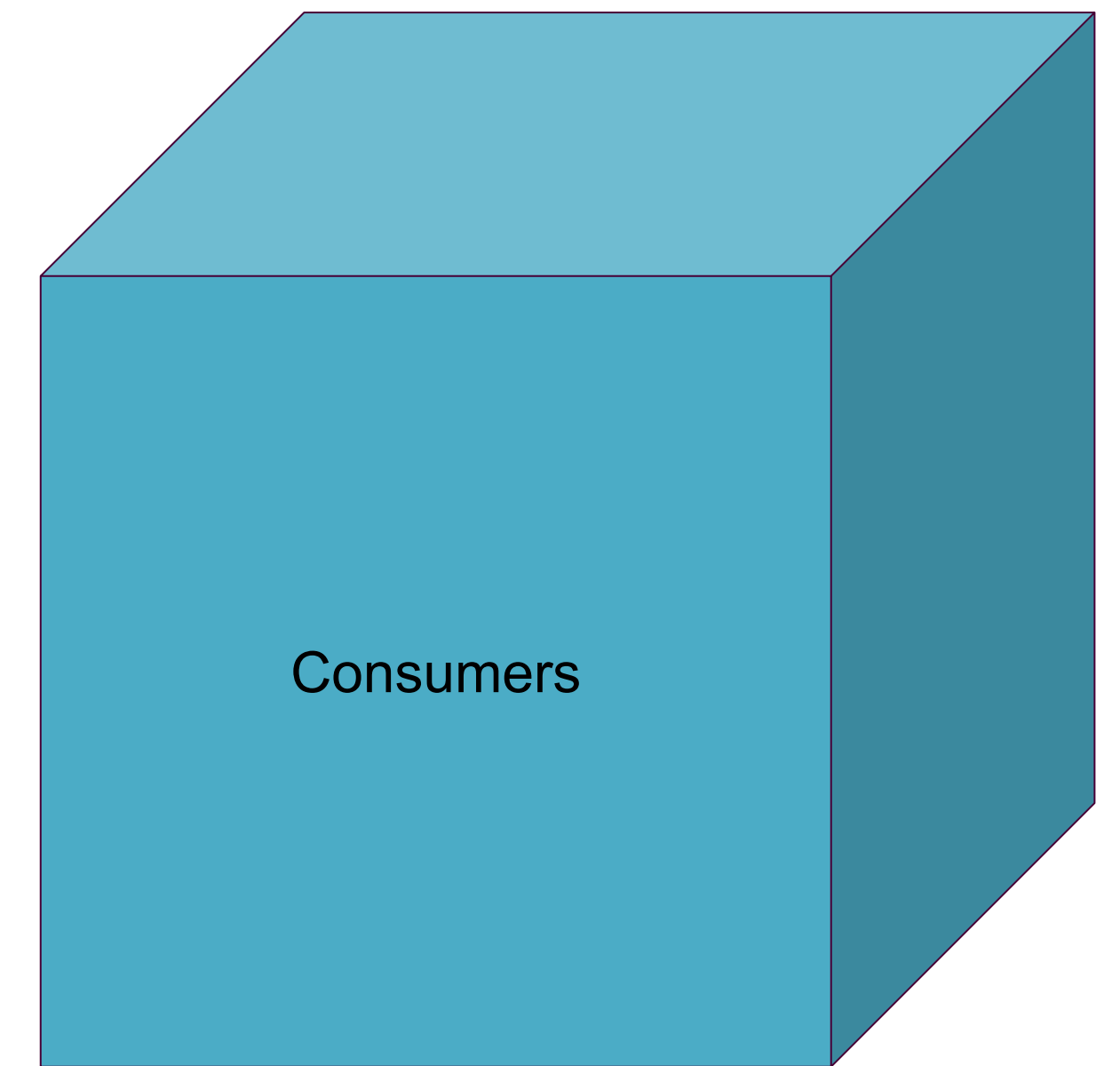
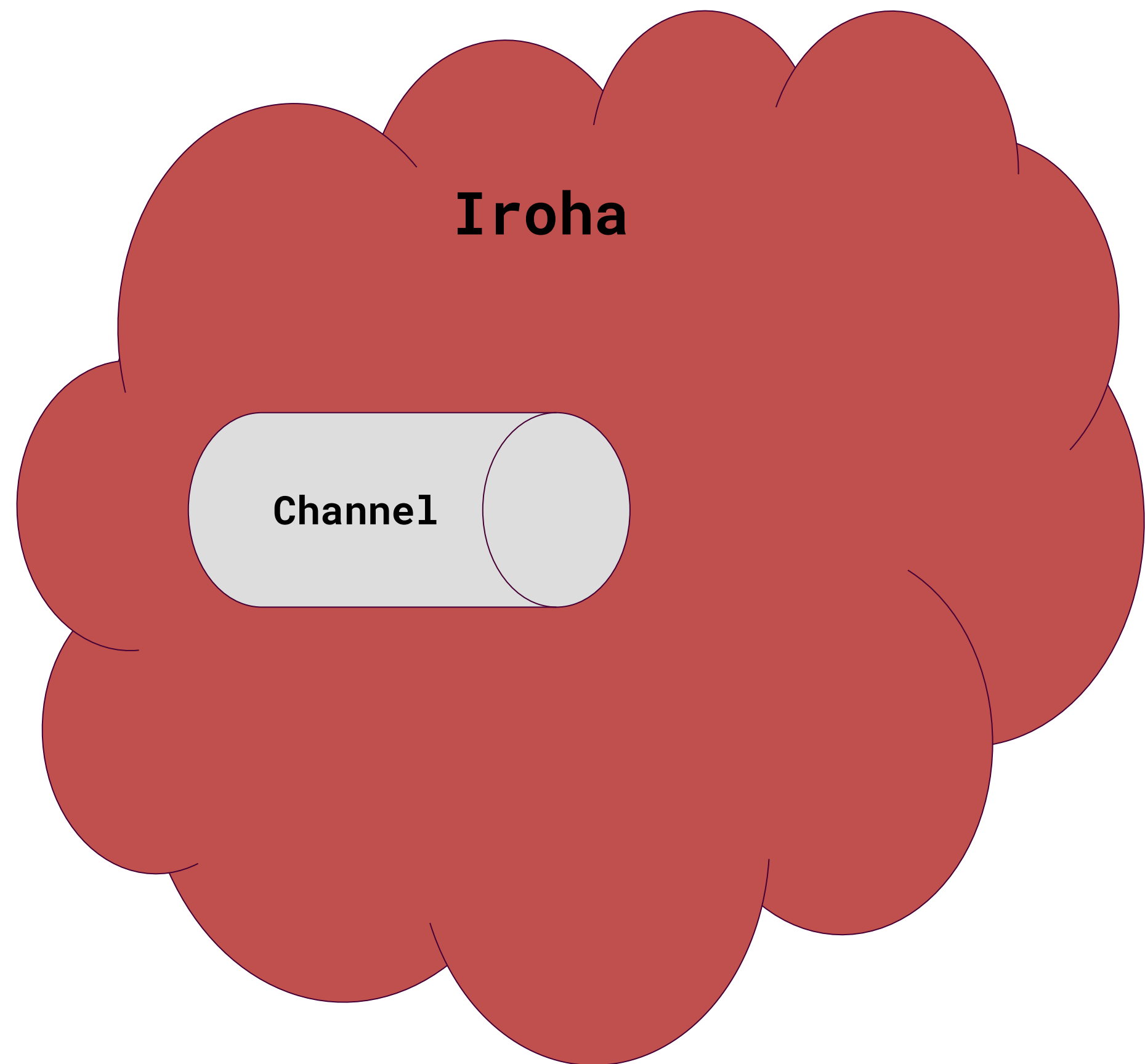
Failure



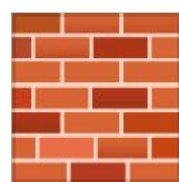



Timeout



Clean Up



Occurrences and Events

-  Creation of a new Block
- Transaction Validation
 -  Success
 -  Failure
- Asset Definition Registration
-  Time
- ...

Example

```
{
  "data": {
    "@type": "types.irohaapis.com/iroha.tx.v1.StatusMessage",
    "attributes": {
      "previous": "accepted",
      "next": "verified",
    },
    "messageId": "faa2e8da-9e4d-40c3-b4ed-5a975a3071b6",
    "publishTime": "2020-06-05T12:00:00.000Z",
    "data": "somebase64encodedmessage"
  },
  "context": {
    "eventId": "1ebf80a4-30e4-45c5-97f2-9d3a1f947fc0",
    "timestamp": "2020-06-05T12:00:00.000Z",
    "eventId": "iroha.tx.status.change",
    "resource": {
      "name": "domain/account",
      "service": "tx.irohaapis.com"
    }
  }
}
```




Questions?

