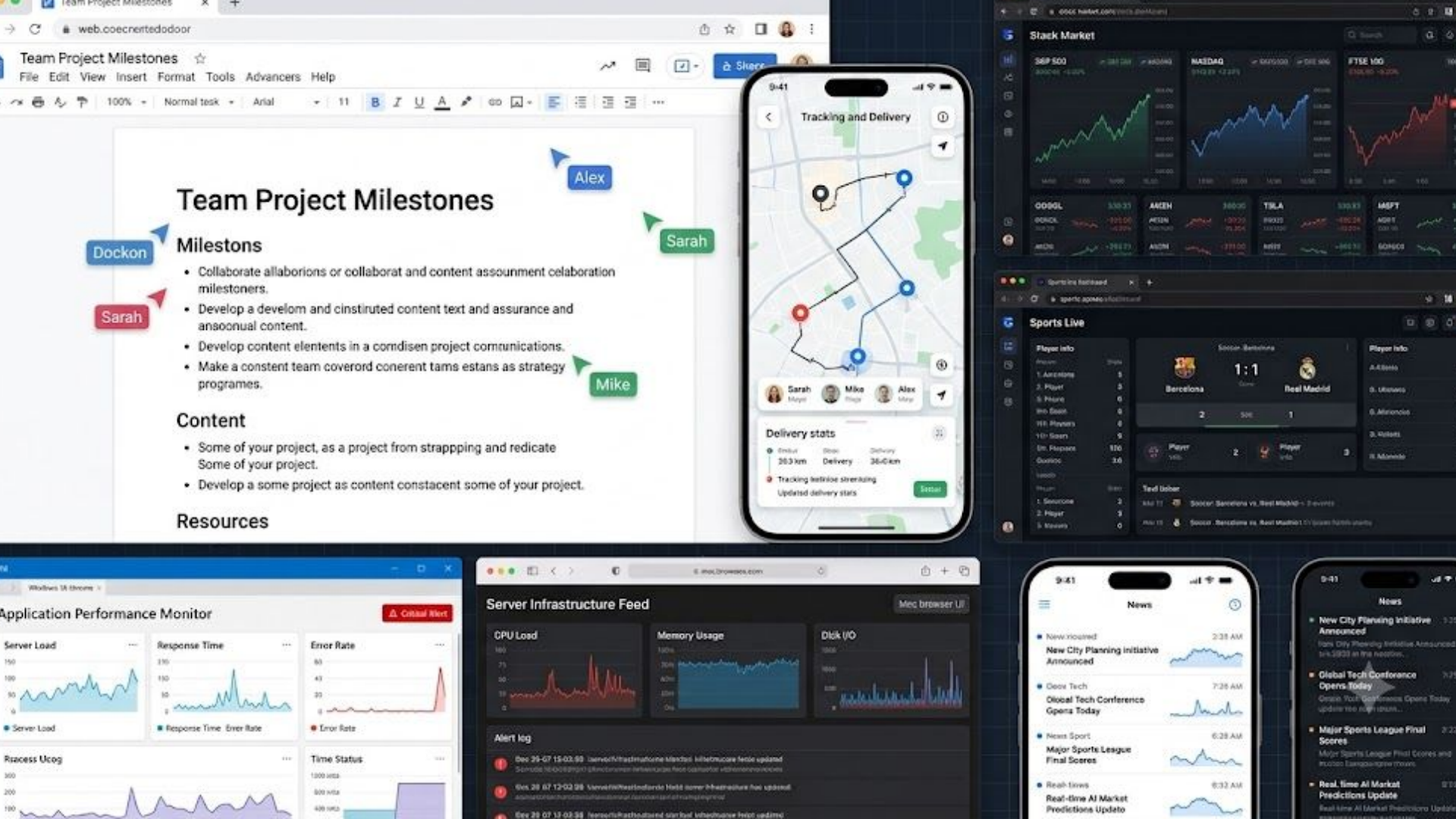
HTTP Events Query

Rahul Gupta

HotRFC, IETF 126


HTTP


web.coecnertedodoor
Team Project Milestones
File Edit View Insert Format Tools Advancers Help
Team Project Milestones
Alex
Dockon
Sarah
Milestons
Collaborate allaborions or collaborat and content assounment celaboration milestoners.
Develop a develom and cinstiruted content text and assurance and ansoonual content.
Sarah
Develop content elentents in a corndisen project comunications.
Make a constent team coverord conerent tams estans as strategy programes.
Mike
Content
Some of your project, as a project from strappping and redicate Some of your project.
Develop a some project as content constacent some of your project.
Resources
Tracking and Delivery
Sarah
Mike
Alex
Delivery stats
Slack Market
S&P 500
NASDAQ
FTSE 100
Sports Live
Barcelona
1:1
Real Madrid
Application Performance Monitor
Critical Alert
Server Load
Response Time
Error Rate
Server Infrastructure Feed
CPU Load
Memory Usage
Disk I/O
Alert log
News
New City Planning Initiative Announced
Global Tech Conference Opens Today
Major Sports League Final Scores
Real-time AI Market Predictions Update

Existing Notification Protocols

- WebSockets
  - Opaque tunnel; custom protocol, no intermediation (violates uniform-interface, layering)
- WebHooks
  - Inverts roles: client needs public URL (violates client-server)
- WebSub/WebPush/MQTT/Mercure
  - Requires Hub or Broker (violates layering)

Server-Sent Events/EventSource

- Only one media type
  - `text/eventstream`
- No request headers
- Unmaintained



Events Query

- Fetch notifications over HTTP using QUERY [RFC10008]
  - Not switch protocols
- Add notifications to any resource
  - Discourages (not prohibits) end-points
- Reliable and In-order notifications
- Fetch representation and notifications in a single request
- Content Negotiation
- Allows intermediaries to participate

Discovery

Request

```
HEAD /foo HTTP/1.1
Host: example.org
```

Response

```
HTTP/1.1 200 OK
Accept-Query: example/events-query+yaml
```

Streaming with Representation: Request

```
QUERY /foo HTTP/1.1
Host: example.org
Content-Type: example/events-query+yaml
Events: duration=1200

state:
  Accept: text/plain
events:
  Accept: example/event-notification
```

Streaming with Representation: Response

```
HTTP/1.1 200 OK
Accept-Query: example/event-query+yaml
Transfer-Encoding: chunked
Incremental: ?1
Events: duration=600
Content-Type: multipart/mixed; boundary="BOUNDARY"

--BOUNDARY
Content-Type: text/plain
Content-Length: 14

Hello World!
--BOUNDARY
```

Streaming with Representation: Response /2

```
Content-Type: example/event-notification
Content-Length: 31

published: 2025-01-02T10:11:12.345Z
event-id: 567
type: update
--BOUNDARY
```

Streaming with Representation: Response /2

```
Content-Type: example/event-notification
Content-Length: 31

published: 2025-01-02T10:11:12.345Z
event-id: 567
type: update
--BOUNDARY
Content-Type: example/event-notification
Content-Length: 31

published: 2025-01-02T11:12:13.456Z
event-id: 678
type: delete
--BOUNDARY--
```

Implementation

Express.js Demo by Rahul Gupta:
https://github.com/cxres/events-query-express-demo

CAMARA Demo by Jaime Jimenez:
https://camara.jaime.win

You?

Engagement

- Internet-Draft
  https://cxres.github.io/events-query
  https://datatracker.ietf.org/doc/draft-gupta-httpapi-events-query/
- Specification Repository
  https://github.com/CxRes/events-query
- Mailing List
  httpapi@ietf.org