dCache events for users

Paul Millar

dCache Workshop 2018 at DESY, Hamburg, 2018-05-28
https://indico.desy.de/indico/event/19920/

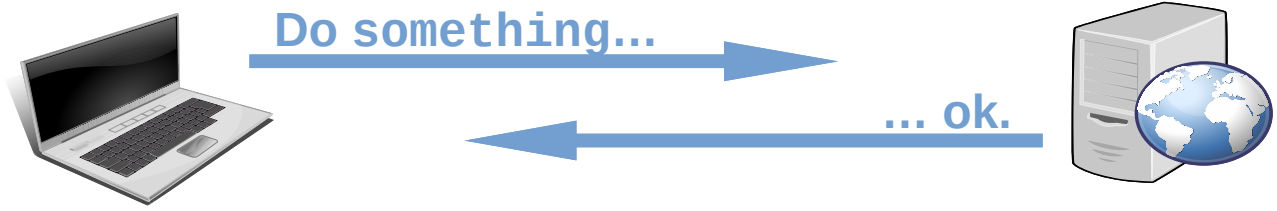
What problems are we trying to solve?

- Users have an **enriched view** of data
  - How to keep this up-to-date?
- Users want to process **incoming data**
  - How to trigger analysis / metadata extraction / derived data ?
- Users want to **stage files** from tape efficiently
  - How to process files quickly once they become available?
- Users want to innovate with (many) existing storage systems
  - How to make this **Just Work™** ?

Standard HTTP & the notification problem

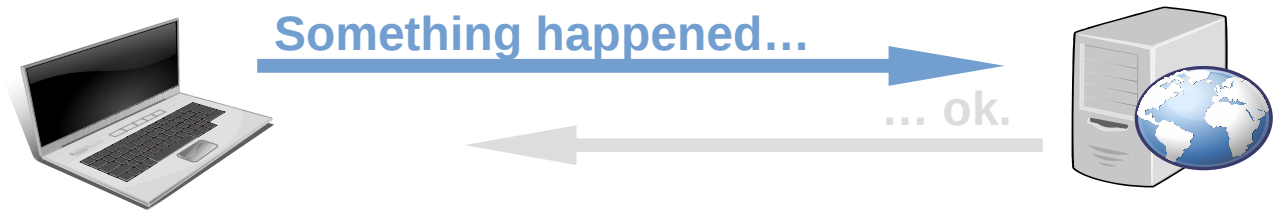
Interactions:

Works “out of the box”



Client events:

Yes, but kinda badly.



Server events:

Not at all

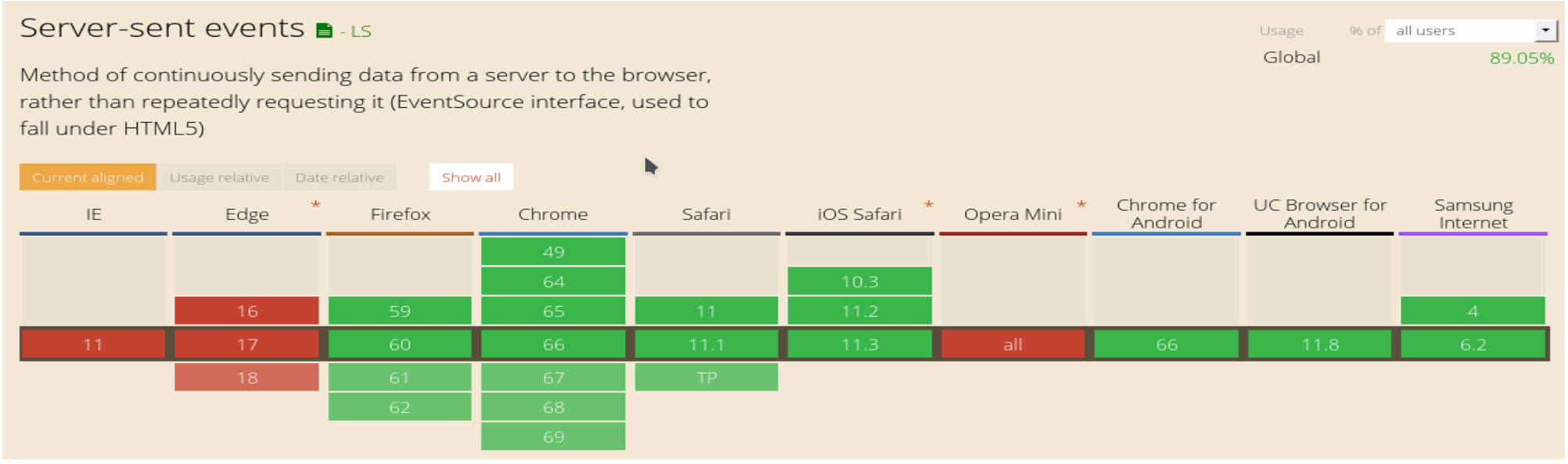
How to solve the “server events” problem

- 2000: various solutions introduced

  Comet, BOSH, Bayeux, long-get, …

- 2006, W3C WHATWG standardised: Server-Sent Events (SSE)
  - Standard: HTML 5
  - Solves the server events by layering a new protocol on top of HTTP
  - Client can avoid loosing events when disconnected

SSE: is it supported?



Source: https://caniuse.com/#feat=eventsource

SSE: is it supported: libraries

22 libraries in
12 languages

Server-sent events

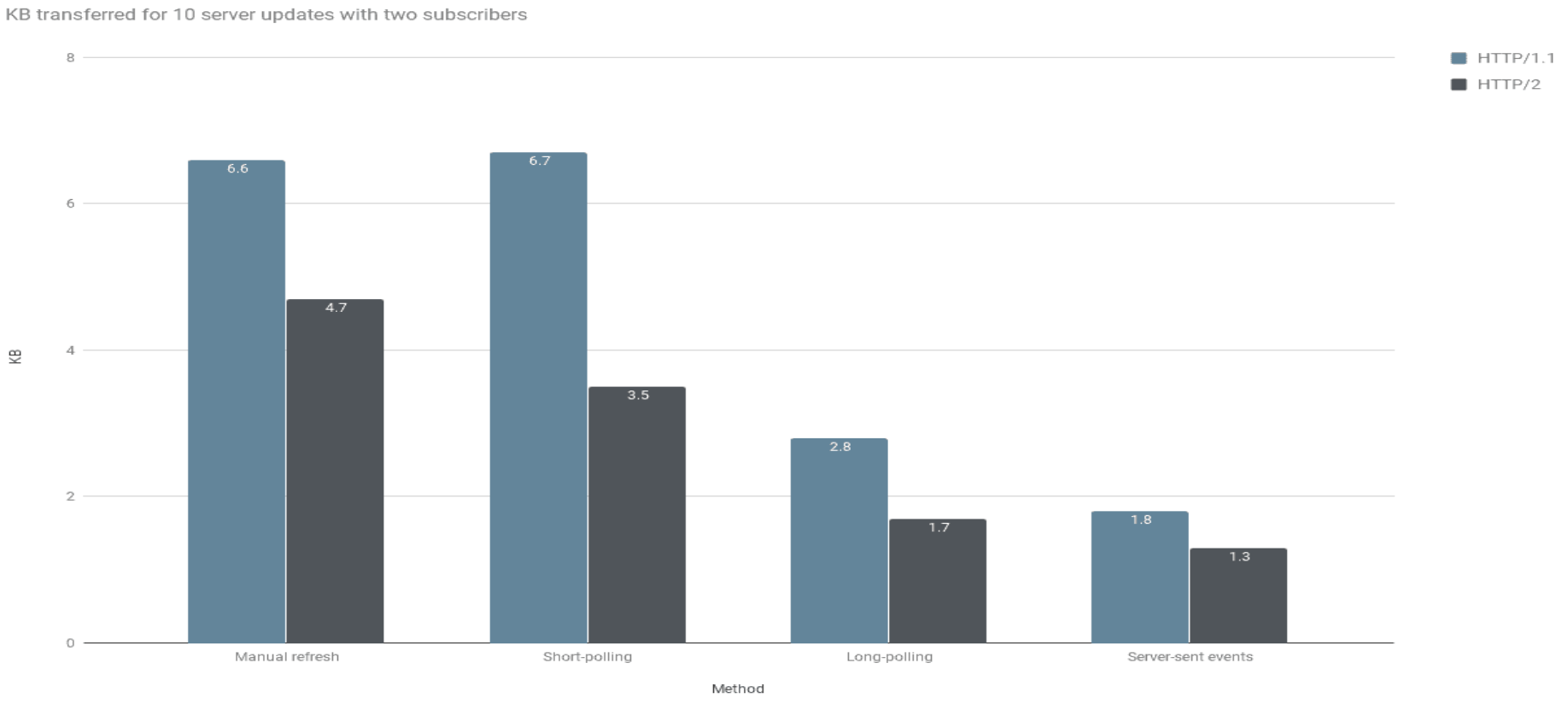
From Wikipedia, the free encyclopedia

Server-sent events (**SSE**) is a technology where a browser receives automatic updates from a server via HTTP connection. The Server-Sent Events EventSource API is standardized as part of HTML5[1] by the W3C.

Contents [hide]

Source: https://en.wikipedia.org/wiki/Server-sent_events

How does it perform?



Source https://aquil.io/articles/a-comparison-between-websockets-server-sent-events-and-polling

dCache implementation of SSE

- Requires **authentication**: no anonymous event delivery.
- Available to **all users** out-of-the-box: no admin configuration
- Management API is documented with **Swagger**

  Supports several, optional **advanced features**
- **Metronome**: a example event source for testing

  (pluggable interface – you can add your own events!)

dCache support for SSE

- Simple model:
  - Client creates a channel (the SSE endpoint)
  - Client subscribes to events for that channel
  - Events delivered to a channel

Compared to Kafka

- **Benefits**:
  - No extra service to install,
  - Built-in (user-driven) security model,
  - No admin effort needed before users can start,
  - Works with web-browsers,
- **Disadvantages**:
  - Fewer out-of-the-box integration options,
  - Event management API is dCache-specific,
  - “Catch-up” event storage is an in-memory ring-buffer.

What users can use SSE for...

- Processing **new data** as it is ingested.
- Avoiding **dark-data** and **dangling links** in catalogues.
- Enforcing **data placement rules**.
- Triggering **analysis** after staging data.
- Avoiding **custom clients**.

… plus many other things

What will be available: dCache v4.2?

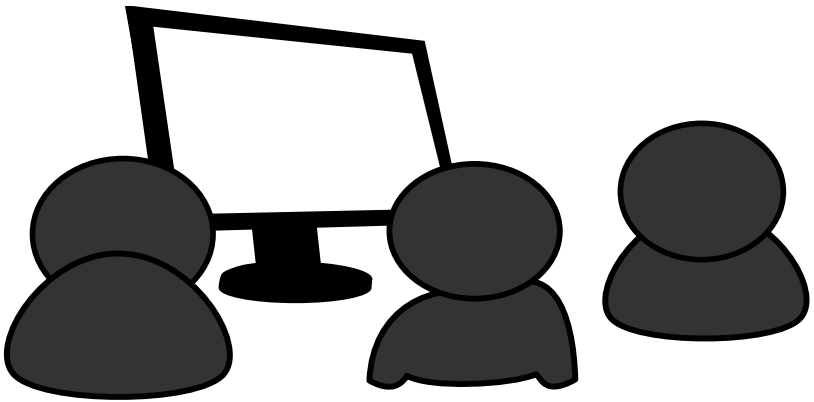
Simple example: **metronome**

- Send simple messages at a fixed rate

  configurable from many kHz to every x seconds.
- Can limit the number of messages
- Intended to for demonstrations and to test clients are working correctly

… yeah, not really all that useful.

Metronome demo…

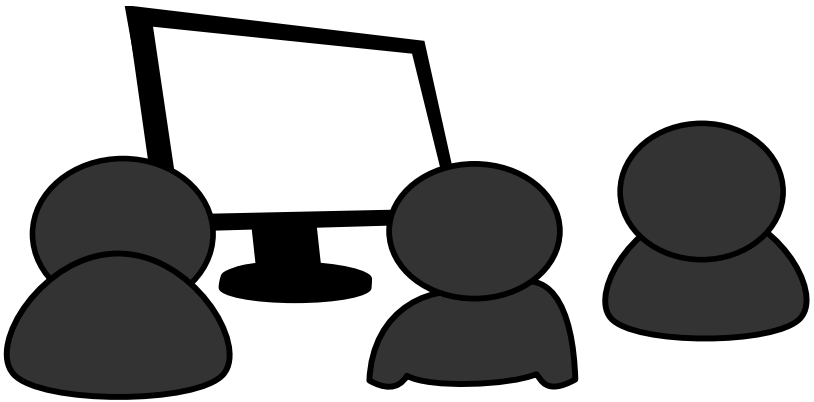


Coming soon: initial inotify

- See directory contents being created, deleted, renamed, …
- API is strongly based on Linux's **inotify**.
- All doors/protocols supported.
- There are some limitations:
  - Currently no **IN_OPEN**, **IN_MODIFY**, **IN_ACCESS**, **IN_CLOSE_NOWRITE** or **IN_CLOSE_WRITE** events for files.

    Use **IN_ATTR** after **IN_CREATE** as alternative to **IN_CLOSE_WRITE**
  - Missing flags: **IN_EXCL_UNLINK**, **IN_DONT_FOLLOW**.
  - No events from chimera CLI or manually editing DB tables.

Inotify demo …



Coming in the future

- **Full inotify** event support.
- **Transfers** started/concluded/progress.
- Changes in **media availability**:

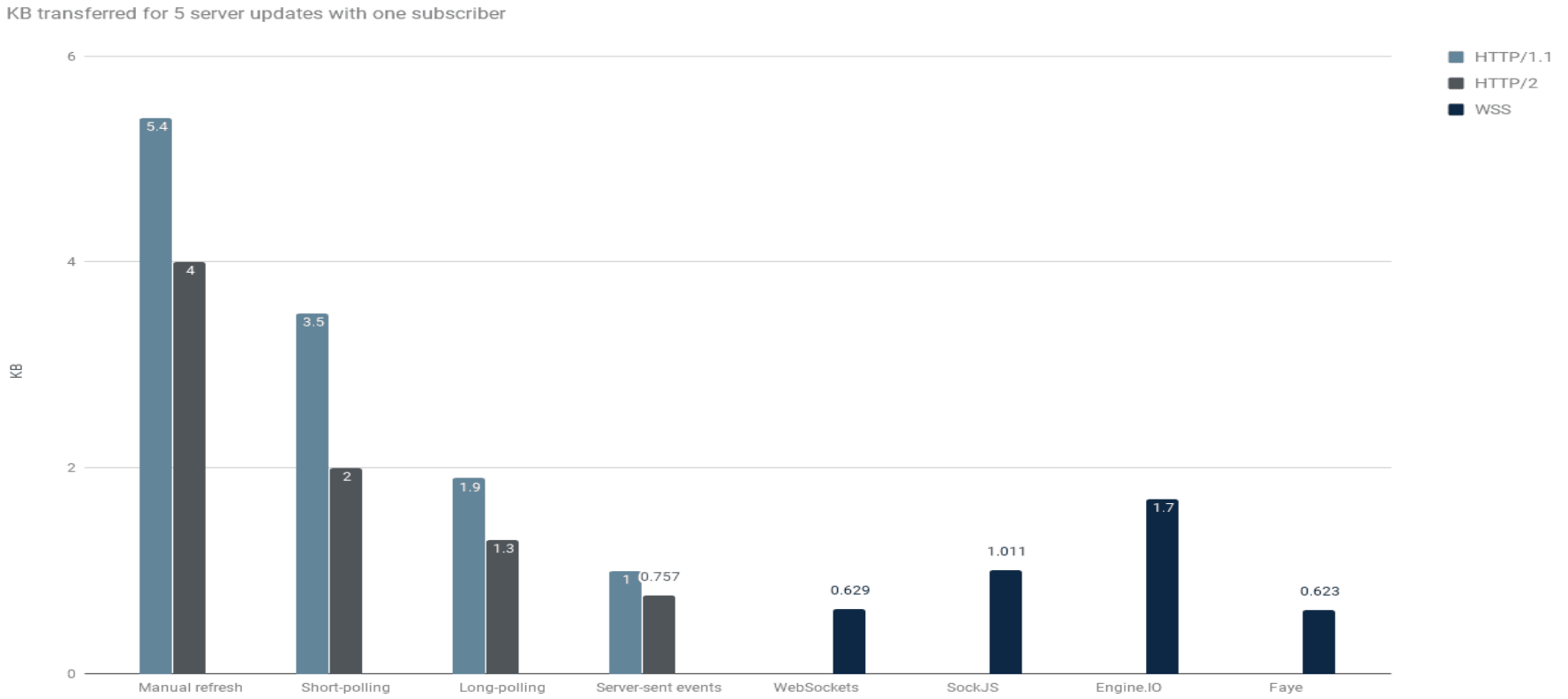
  Learn when data is staged or when last cache-copy is removed.
- **Quality-of-Service** (QoS) changes:

  Part of a larger work in revamping QoS support.

Thanks for listening!

Backup slides

How does it perform?



From "*A comparison between WebSockets, server-sent events, and polling*", by Alexis Abril
https://aquil.io/articles/a-comparison-between-websockets-server-sent-events-and-polling