ALTO Incremental Updates Using Server-Sent Events (SSE)

draft-ietf-alto-incr-update-sse-17

W. Roome
Y. Richard Yang

IETF 104
March 26, 2019
Prague
Updates Overview (v13-v17)


- Two major types of changes
  - Fixed issues identified in WGLC reviews – most are organizing and clarification issues
  - Fixed issues motivated by handling PV – a systematic approach to handling `multipart/related`

- Huge group efforts by many, in several weekly meetings
Update: Updated Draft Structure
## Update: Highlight of Major Changes

### 2. Major Changes Since Version -01

**To RFC editor:** This will be removed in the final version. We keep this section to make clear major changes in the technical content.

- Incremental encoding: Added JSON patch as an alternative incremental delta encoding.
- Update concurrent requests of the same resource: The client now assigns a unique client-id to each resource in an update stream. The server puts the client-id in each update event for that resource (before, the server used the server's resource-id). This allows a client to use one update stream to get updates to two different requests with the same server resource-id; before, that required two separate update streams.
- Control: Defined a new "stream control" resource (Section 8) to allow clients to add or remove resources from a previously created update stream. The ALTO server creates a new stream control resource for each update stream instance, assigns a unique URI to it, and sends the URI to the client as the first event in the stream.

- Incremental encoding using JSON patch: Added JSON patch as an alternative incremental delta encoding than Merge patch.
- Substream-id to allow concurrent updates of the same server resource: This design allows an ALTO client to assign a unique substream-id when requesting a resource in an update stream. The server puts the substream-id in each update event for that resource (before, the server used the server's resource-id). This allows a client to use one update stream to receive updates to multiple requests for the same server resource, for example, for a POST-mode resource with different input parameters; before, that required separate update streams.

- Multipart resources: Use generic `data-id` subfield of the `event` field to identify the data to be updated. For all multipart existing data, data-id is the substream-id, but it allows support of multipart as well, by adding content-id.
- Flexible control: Defined a new "stream control" resource (Section 8) to allow a client to add or remove resources from a previously created update stream. The ALTO server creates a new stream control resource for each update stream instance, assigns a unique URI to it, and sends the URI to the client as the first event in the stream.
Update: Restructure the Order of Terms

This document uses the following terms: Update Stream, Update Server, Update Message, Data Update Message, Full Replacement, Incremental Change, Update Stream Server, Update Stream Control Service, Update Stream Control, Control Update Message, Stream Control Service.

Update Stream: An update stream is an HTTP connection between an ALTO client and an ALTO server so that the server can push a sequence of update messages using SSE to the client.

Update Stream Server: We refer to an ALTO server providing an update stream as an ALTO update stream server, or update stream server for short. Note that the ALTO server mentioned in this document refers to a general server that provides various kinds of services; it can be an update stream server or stream control server (see below); it can also be a server providing ALTO IBD information.

Update Message: An update message is either a data update message or a control update message.

Data Update Message: A data update message is for a single ALTO information resource and sent from the update stream server to the ALTO client when the resource changes. A data update message can be either a full-replacement message or an incremental-change message. Full replacement is a shorthand for a full-replacement message, and incremental change is a shorthand for an incremental-change message.

Full Replacement: A full replacement for a resource encodes the content of the resource in its original ALTO encoding.

Incremental Change: An incremental change specifies only the difference between the new content and the previous version. An incremental change can be encoded using either JSON merge patch or JSON patch in this document.

Update Stream Server: An update stream server is an ALTO server that provides update stream service.

Stream Control Service: An stream control service provides an HTTP URI so that the ALTO client of an update stream can use it to send stream control requests on the addition or removal of resources receiving update messages.

Stream Control: A shorthand for stream control service.

Control Update Message: A control update message is for the update stream server to notify the ALTO client of related control information of the update stream. The first control update message provides the URI using which the ALTO client can send stream control requests to the stream control server. Additional control update messages allow the update stream server to notify the ALTO client of status changes (e.g., the server will no longer send update messages for an information resource).
Update: Clarify Interaction

---

In addition to state changes triggered from the update stream server side, in a flexible design, an ALTO client may initiate changes as well, in particular, by adding or removing ALTO resources receiving updates. An ALTO client initiates such changes using the stream control service. For an update stream service supporting update stream control, the update stream server responds by sending an event (a control update message) with the URI of the stream control service. The ALTO client can then use the URI to ask the update stream server to (1) send data update messages for additional resources, (2) stop sending data update messages for previously requested resources, or (3) gracefully stop and close the update stream altogether. Figure 2 shows the complete ALTO SSE architecture.
Update: Clarify Security Considerations

- Clarify that original security issues still apply

- Categorize security issues into three aspects: stream server, ALTO client and stream control server
Update: Provide Better Wording on Alternative Design

Note that the ALTO client cannot suppress full replacement.
6. Update Messages: Data Update and Control Update Messages

We now define the details of ALTO incremental update. Specifically, an update stream consists of a stream of data update messages (Section 6.2) and control update messages (Section 6.3).

6.1. ALTO Update Message Format

Data update and control update messages have the same basic structure. The data field is a JSON object, and the event field contains the media type of the data field and an optional client-id.

Data update messages use the client-id to identify the ALTO resource to which the update message applies. Client-ids MUST follow the rules for ALTO ResourceIds (Section 10.2 of [RFC7285]). Client-ids MUST be unique within an update stream, but need not be globally unique. For example, if an ALTO client requests updates for both a cost map and its dependent network map, the ALTO client might assign client-id "1" to the network map and client-id "2" to the cost map. Alternatively, the ALTO client could use the client-ids for those two maps.

JSON specifications use the type ClientId for a client-id, and the type ClientId conforms to the specification of ResourceId as defined in Section 10.2 of [RFC7285].

The two sub-fields (media-type and client-id) of the event field are encoded as comma-separated strings:

```
media-type [ ',' client-id ]
```

Note that media-type names may not contain a comma (character code 0x2c). [Dawn: may not or MAY NOT]

A data update message needs additional information to identify the ALTO data to which the update message applies. For example, an ALTO client can request updates for both a cost map and its dependent network map in the same update stream. The ALTO client assigns substream-id "1" in its request to receive updates to the network map; and substream-id "2" to the cost map. For this example, the substream-id defines the data to be updated and need to be indicated in a data update message.

Hence, the event field of ALTO update message can include two sub-fields (media-type and data-id), where the two sub-fields are separated by a comma:

```
media-type [ ',' data-id ]
```

To allow non-ambiguous decoding of the two sub-fields, the media-type name used by ALTO SSE MUST NOT contain a comma (character code 0x2c), and the string before the comma is the media-type name. [To RFC editor: please check this conforms to Section 4.2 of [RFC6838] and confirms to IANA.]
Integrated Handling of Multipart
Summary of Status

• The revision has addressed all issues identified so far

• Plan:
  – Take a pass to address any WG comments, if raised: in 1 week after IETF
  – Work through the rest of the process