

WGLC Status for New Transport

draft-ietf-alto-new-transport-07

Roland, Richard, Kai, Lauren and Lachlan

ALTO Interim, April 11, 2023

Summary: WGLC Status

Reviews

[RTGDIR Early review by Russ White](#)

Has Nits

[SECDIR Early review by Donald Eastlake](#)

Has Nits

[OPSDIR Early review by Sheng Jiang](#)

Has Nits

[ARTART Early review by Spencer Dawkins](#)

Ready with Issues

[TSVART Early review by Bernard Aboba](#)

Not Ready

[HTTPDIR Early review by Martin Thomson](#)

Not Ready

[ARTART Early review \(of -01\) by Spencer Dawkins](#)

Ready with Issues

- RTGDIR (responded, waiting for feedback)
- SECDIR (responded, considered as resolved)
- OPSDIR (responded, waiting for feedback)
- ARTART (responded, waiting for feedback)
- TSVART (responded, waiting for feedback)
- HTTPDIR (responded, still debating)
- WGLC review from Luis

RTGDIR Early Review

- Lachlan has replied and explained that the push order does not need to be strictly in the correct order
- Waiting for further feedback

Nits:

This document assumes data will be pushed in the correct order, but it doesn't seem to say this is a requirement. Since the document is dealing with changes to a network topology (for instance), it seems like this would be a requirement to mention someplace. It could be that I missed this requirement, however.

SECDIR Early Review

- Some nits are raised and already fixed

Nits:

Section 1.0, Page 4:

OLD

functioning for HTTP/1.x. TIPS also provides an ALTO server to

NEW

functioning for HTTP/1.x. TIPS also provides for an ALTO server to

Section 2.1.1, Page 8: Seems too vague. A sentence about tips-view-uri wouldn't hurt. At the bottom it says "Use the URI as above". Which URI above? What exactly does "use" mean?

Section 2.2, Page 9, Figure 3: Figure looks kind of incomplete. Shouldn't there be arrows from R1 to R2/R3?

Section 2.3, Page 10: In the text on "Information Resource Directory" the first sentence is confusing. What is the thing that is requested to discover? Maybe you should replace "Requested" at the start of the sentence with "Produced when a server is requested"...

Section 2.3, Page 11 at top: That's Figure 4, not 1.

Section 2.4, Page 12, 1st paragraph: I think a service runs "over" a connection not "inside" a connection

Hi Lachlan,

I'm happy with all of your responses and consider my comments to have been resolved.

Thanks,
Donald

=====

ble. How about, given that a thing like the following, which ins a use count. (This document need

rver will not want to delete

rver MUST NOT delete

OPSDIR Early Review

- Mostly nits
- Fixed but need another round of revision (missing references)

This document intends for Standards Track. It document specifies a new ALTO Transport Information Publication Service (TIPS), which takes advantage of HTTP/2 and later versions already support concurrent, non-blocking transport of multiple streams in the same HTTP connection. This document is well-written and READY with minor Nits (below) for publication.

In section 2.5 "TIPS with HTTP/1.x". This document claims "TIPS still functions with HTTP/1.x for client pull functionality only, with the limitation that it cannot cancel any outstanding requests or fetch resources concurrently over the same connection." However, it is unclear what operations/procedure defined in this document will fail when TIPS work over HTTP/1.x (also whether there are scenarios the server/client have to fall back to HTTP/1.x), the signal or error code the TIPS will provide, and how the server and client should act when these failures happen.

This looks like an operational hole that authors must fill before publication. There is no scenarios that the server/client have to handle. The authors may simply declare the TIPS SHOULD not

Hi, Lachlan,

Thanks for your reply and address my comments. The sentence "The HTTP version a connection uses is negotiated between client and server based on the respective HTTP RFC documents." assuming that "HTTP RFC documents" are some references in which the negotiation procedures have been defined.

Regards,

ARTART Early Review

- Different ways to initiate server push with HTTP/2 and /3
- Some clarification questions and nits
- Responded with proposed changes and waiting for feedback

I know a lot of things changed in this draft since I early-reviewed -01 - I'm now early-reviewing -07 - but I'm not sure that my previous observation about this HTTP setting

0x02 SETTINGS_ENABLE_PUSH (a [BCP14](#) "MUST")

has been addressed.

This set may be any subset of the ALTO server's network information resources and may include resources defined in linked IRDs. However, it is RECOMMENDED that the ALTO server selects a set that is closed under the resource dependency relationship. That is, if a TIPS' "uses" set includes resource R1 and resource R1 depends on ("uses") resource R0, then the TIPS' "uses" set SHOULD include R0 as well as R1. For example, a TIPS for a cost map SHOULD also provide a TIPS view for the network map upon which that cost map depends.

I have the same question about R1 and R0, but let's start with a specific case. If a TIPS for a cost map does not also provide a TIPS view for the underlying network map, what happens next?

(Nit) is there a missing term after TIPS in "a TIPS for a cost map" in this paragraph?

In 4.4.

<<https://datatracker.ietf.org/doc/html/draft-ietf-alto-new-transport-07#section-4.4>>close Request

<<https://datatracker.ietf.org/doc/html/draft-ietf-alto-new-transport-07#name-close-request>>

An ALTO client can indicate it no longer desires to pull/receive updates for a specific network resource by "deleting" the TIPS view using the returned tips-view-uri and the HTTP DELETE method. Whether or not the server actually deletes the TIPS view is implementation dependent. Likely, a server will remove the client from a dependency set associated with the TIPS view. A server will not want to delete a TIPS view if another client is using it.

I'm guessing here, but this looks like it's conflating client usage with server storage management. If client A says "delete the TIPS view", and no other client is using it, that view is deleted, but if another client is

TSVART Early Review

- Issues:
 - Transport requirements not laid out
 - Backward compatibility not stated or justified
 - Must the increment between updates be 1?
 - Define more efficient coding schema?
- Responded with proposed changes

Overall, the document does not lay out the transport requirements explicitly, and so it is hard for me to tell whether the proposed design is optimal or not.

In particular, it would be useful to understand the desired reliability vs. latency tradeoff, as well as the requirements for backward compatibility (e.g. need to operate over HTTP 1.x as well as HTTP/2 and HTTP/3).

There seems to be a requirement for the protocol to support HTTP 1.x as well as HTTP/2 and HTTP/3, but this is not stated explicitly, nor is it justified. In

Also, the requirements for reliability and latency are not explicitly laid out. The diagrams show incremental update N+1 always depending on update N. Is this an inherent limitation imposed to meet a requirement, or is it a choice that could potentially be modified? For example, do bad things happen if a client

Are there circumstances where such a tradeoff would be desirable? As an example, there could be a base layer of updates where update N+2 depends on update N, and an extension layer (with higher update frequency) where update N+1 depends on update N. The client could request both layers if it could handle the higher frequency, or just the base layer if it could not. A diagram describing this kind of two-layer dependency structure is here:

https://www.w3.org/TR/webrtc-svc/#L1T2*

Changes for the TSVART Review

- Transport requirements not laid out
- Backward compatibility not stated or justified
- A new subsection is added in -08 to clarify the transport requirements
- Must the increment between updates be 1?
- Define more efficient coding schema?
- Example modified to show that incremental changes can be offered between non-neighbor versions
- Non-stepwise updates are left as server specific options

HTTPDIR Early Review

- A major road blocker at this point
- Three major technical issues
 - The use of HTTP/2 and /3 server push
 - Session bounded to a single (persistent) connection
 - “Meta layer” too complex
- Writing issues
 - inconsistency of terms
 - “specification by example”

The first of my issues makes me wonder if this has been implemented at all. And as I went through this, I found myself asking that same question again multiple times. Has it?

Finally, I feel obligated to point out that expending effort on HTTP server push is perhaps unwise given its relative adoption and success. That is, it has not
Server Push Usage

Section 7.1 says "A client can add itself explicitly to the receiver set or add itself to the receiver set when requesting the TIPS view." It describes two methods for doing this, but neither indicates which request will remain open so that the client can receive push promises.

HTTP server push requires that the server send pushes alongside an outstanding request, but aside from discussion of streams in Section 7.3.1, I can't work out how the client would do that. Section 2.4 also fails to make this clear.

consequently, I cannot convince myself that the primary feature of this document will work.

Connections and Clients

I can't pin this one down, but there seems to be some sort of assumption that there is a 1:1 correspondence between connections and clients. That is not how HTTP works. In HTTP, every request stands on its own. Though there might be linkages between requests, those linkages should not affect how HTTP itself operates, including server push. (You might detect a common theme here.)

As I noted, I'm not completely certain about raising this issue because of a lack of clarity about how the protocol is supposed to operate.

Road blocker #1

- The use of HTTP/2 and /3 server push
 - lack of implementation support
 - not the intended use of server push

Potential options:

- Option 1: We split (again) HTTP/2 and /3 as another document
- Option 2: We convince Martin that the use of server push can be justified, i.e., there is good implementation practice

Road blocker #2

- Session bounded to a single (persistent) connection
 - HTTP-based protocols should be based on requests rather than connections

Potential options

- Option 1: We adjust the protocol to allow the same user use multiple connections but RECOMMENDED to use a single one for performance enhancement
- Option 2: We justify the design with other working examples of mapping a connection to a session/user

Next step

- Follow up the WGLC inside the WG
- Follow up with RTGDIR, OPSDIR, ARTART, and TSVART reviewers
- Continue the discussion with Martin (HTTPDIR reviewer) and finalize the new revision

Thanks!