

2021-09-27 Meeting: Activity Streams - Appendices for Use Cases

Charter	Project Page	Terminology	References	Working Documents	Deliverables	Meeting Notes
-------------------------	------------------------------	-----------------------------	----------------------------	-----------------------------------	------------------------------	-------------------------------

- [ACTION ITEMS:](#)
 - [Agenda](#)
 - [Meeting Materials](#)
 - [Recording](#)
 - [Notes](#)
 - [Full Cache](#)
 - [Notifications](#)
 - [Partial Cache](#)
 - [Future Topics](#)
-

ACTION ITEMS:

From Last Meeting

- *Lynette - Followup in slack about some of the remaining questions.*
- *Vitus - provide examples of types that would be used with entities and types that would be used with authorities*

This Week

Agenda

- **Review** [updated activity streams document](#)
 - Appendices for 3 primary use cases (i.e. full cache, partial cache, notifications)
 - Object types

Meeting Materials

- [Activity Streams - Extensions for Authoritative Data Change Management \(min extensions + instruments\)](#)
-

Recording

Recording: [Activity Streams - Appendices for Use Cases \(2021-09-27\)](#)

Notes

Notes were taken directly in the Activity Streams document as comments with general discussions captured here.

Full Cache

Alternate approach for full cache:

- take full dump
- one patch file for all changes since last dump
- new changes are appended to the end of the patch file as they occur
- process
 - apply a patch file to the last full dump

Might work well for smaller authorities. Or if it is small, they may just do a full replace instead of applying patches. Complexity of following a series of patch documents may be more overhead than processing a full download.

Could be useful for the first time grab of data. Take the last full dump. Apply the current full patch. From then on process the Activity Stream to keep it up to date.

What is the value of the detailed information in the Activity Stream for a full cache?

- for a large cache, the ingest is very time consuming
- the detailed information (e.g. object type, etc.) may not be used by the full cache model

Alternative: The provider periodically produces a large patch file that can be applied. The consumer could keep applying each in order.

- would these happen often enough?
- Activity Streams can update really frequently and possibly at transaction time. Common to release activity streams daily. LOC is looking at daily releases. Lynette will check with Getty to see how often they update.

Notifications

Example from Notification Stream for LOC

/authorities/genreForms/activitystreams/feed/27.json

- Add + uri
- Update + uri
- Remove + uri

/authorities/genreForms/activitystreams/labels/27.json (TBD)

- Not developed yet
- Looking at making this ordered to be able to track specific changes
- May include the text of the new label value (TBD)

Partial Cache

The notification approach might will work well for the Partial Cache use case as well, but might depend on what is being cached. The consumer knows which URIs are cached and what data about that URI is cached. They can dereference the URI and grab the new data to update the local cache.

If grabbing the entire set of data associated with a URI, will need to have a way to determine the edges of the graph for removing the outdated triples and adding the triples of the new full graph for the URI.

Description should include a description of the difference between full cache and notifications. Include that both are activity streams, but the second includes fewer details.

Consumers will need to be familiar with the shape of the data they are consuming. If they want the full graph of a URI, they will need to know this shape.

Producer may only push up the Activity Stream once a day. The only thing that is known is the dt-stamp of the last update.

This may mean that if something got update multiple times during the day, it may not matter. The consumer will need to know the last time they crawled the data and will need to start again at that date.

Consumers may just make a list of URIs that change and then bring in the changes through dereferencing and grabbing the data they need.

Can feeds come too frequently?

Beyond knowing the frequency, the consumer will need to be familiar with the service when it comes to types/modeling.

Future Topics

- *Notification to a single institution when a requested entity that was missing becomes available. This will likely be outside the main change management stream.*
- *Value of diagramming to express changes as a means of conveying information around where data is being produced, where it goes, etc.*
- *Partial splits and partial merges.*

Attendees:

Dave Eichmann
Christine Eslao
Nancy Fallgren
Steven Folsom
Kevin Ford
John Graybeal
Jim Hahn
Kirk Hess
Jesse Lambertson
Anna Lionetti
Alessandra Moi
Tiziana Possemato
Erik Radio
Lynn Ransom
Lynette Rayle
Greg Reeve
Amanda Sprochi
Vitus Tang
Emma Thomson

Absent:
