

AAI Historical Tracking 2018-03-29 Meeting Notes

Recordings

AAI Historical Tracking Recorded Session 2018-03-28

Attendees


- James Forsyth
- Manisha Aggarwal
- Adrian Batos-Parac
- William Reehil
- Pavel Paroulek
- Adrian Slavkovsky
- Yugandhar Guntaka
- Robby Maharajh

Date

28 Mar 2018

Duration 60 minutes

Discussion items

Time	Item	Who	Notes
	Review Proposed Epic	William Reehil	<p>Epic:</p> <p>A subsystem is needed to track the history of the A&AI real-time database at an attribute/entity level for every resource (nodes & edges). The following is what would need to be tracked for the attribute value and entity assertions</p> <ul style="list-style-type: none">• source of change (provenance)• timestamp of when the change (add/update/delete) was made in the database• timestamp of when the change (add/update/delete) was made in the network as per provided by the client (preferred that client sends this but not mandatory) <p>The client will need a streamlined way to access this history via URIs and a GUI. For Casablanca only singular entity history queries will be supported showing the state of the entity at a given time. The design needs to be extensible enough to easily support history queries over a range of time, retrieving complicated topology history, traversal mS endpoints (custom query/dsl), and aggregations post Casablanca.</p> <p>Any data populated in A&AI before this feature was implemented, would not have historical information.</p> <p>If the results have a subset of attributes that did not exist in the range of the data requested, A&AI will display the partial history available</p> <p>Data retention limits will be enforced on the historical database to truncate history, this data retention timeframe will be configurable</p> <div> AAI-953 - AAI subsystem for tracking history of the AAI real-time DB CLOSED</div>
	Review Proposed User Stories	William Reehil	<p>Stories:</p> <ul style="list-style-type: none">• Create the A&AI history subsystem framework (scaffolding)• Develop functionality for the history subsystem to handle storing the historical data within the database layer, recording the database entry time as a meta property on a value/entity/relationship update along with the source of the update• Develop functionality for the history subsystem to handle storing the network sequence time as a meta property on a value/entity/relationship update (if the network sequence timestamp was passed in), also store the source of the update• Develop functionality for the history subsystem to retrieve the state of an entity/relationship at a given database timestamp• Develop functionality for the history subsystem to retrieve the state of an entity/relationship at a given network sequence timestamp (will only pull entity/relationships that have a network sequence timestamp)• Develop functionality for the history subsystem to truncate history data after a configurable range of time• Update the existing A&AI GUI to include functionality to query historical data for an entity/relationship at a particular point in time• Ensure the history subsystem is properly updating the logs per ONAP guidelines

Action items

- ✓ James Forsyth will create the epic and skeleton stories for this initiative, Due 02 Apr 2018