ALTO Incremental Updates
draft-schwan-alto-incr-updates-02

Nico Schwan, W. Roome
Presented by Michael Scharf <michael.scharf@alcatel-lucent.com>
IETF 86, Orlando, Florida, USA, March 2013
Content and Objectives

• Problem statement: Huge size of ALTO cost maps (15 MB for 1000 PID)
• Determine client map version options
  • If-Modified-Since HTTP Header
  • If-None-Match HTTP Header with ETags
  • Version-based incremental updates as ALTO extension, i.e., query for map-vtag
• Incremental update options
  • (HTTP compression)
  • JSON patch
    • Pro: Standard mechanism for JSON documents
    • Con: Special care needed for network maps ➔ Guidance needed even if vanilla mechanism applied
    • Con: Significantly larger overhead (see draft and IETF 85 presentation)
• ALTO extension for incremental update service
  • Pro: Similar to filtered network/cost map message (using map-vtag), i.e., simple to implement
  • Pro: Factor 2 more efficient (see draft and IETF 84 presentation)
  • Con: Extensibility, possibly updates required if future ALTO extensions get standardized
Incremental Update Examples

- **Request**
  - New MIME Type: "application/alto-update-param+json"
  - Input parameter: {"reference-tag": "1266506140"}

- **Response for JSON patch**
  - JSON Patch (draft-ietf-appsawg-json-patch-02)
  - Supports "add", "remove", "replace", "move", "copy" or "test" operations
  - Example: `{ "replace": "meta/data/map/SRC-PID/DEST-PID", "value": 123 }`

- **ALTO Extension**
  - Syntactically equal to Filtered Map services
  - Contains only changed values (or -1 for delete)
  - Example: "SRC-PID": {"DEST-PID": 123}
Next Steps

• Draft recently expired due to new affiliation of Nico Schwan
  ➔ New version to be submitted

• HTTP/1.1 useful for conditional requests
  ➔ Specification instead of survey

• Incremental update options: JSON patch vs. ALTO extension
  • No clear winner, but some WG interest in JSON patch at IETF 84
  • More experience with running code required
  ➔ Too early for decision