Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: September 3, 2020

S. Randriamasy
Nokia Bell Labs
R. Yang
Yale University
Q. Wu
Huawei
L. Deng
China Mobile
N. Schwan
Thales Deutschland

March 2, 2020

Application-Layer Traffic Optimization (ALTO) Cost Calendar draft-ietf-alto-cost-calendar-19

Abstract

This document is an extension to the base Application-Layer Traffic Optimization (ALTO) protocol. It extends the ALTO cost information service so that applications decide not only 'where' to connect, but also 'when'. This is useful for applications that need to perform bulk data transfer and would like to schedule these transfers during an off-peak hour, for example. This extension introduces ALTO Cost Calendar, with which an ALTO Server exposes ALTO cost values in JSON arrays where each value corresponds to a given time interval. The time intervals as well as other Calendar attributes, are specified in the Information Resources Directory and ALTO Server responses.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on September 3, 2020.

Table of Contents

1. Introduction

The base Application-Layer Traffic Optimization (ALTO) protocol specified in [RFC7285] provides guidance to overlay applications that need to select one or several hosts from a set of candidates able to provide a desired resource. This guidance is based on parameters that affect performance and efficiency of the data transmission between the hosts such as the topological distance. The goal of ALTO is to improve the Quality of Experience (QoE) in the application while optimizing resource usage in the underlying network infrastructure.

The ALTO protocol in [RFC7285] specifies a network map which defines groupings of endpoints in provider-defined network regions identified by Provider-defined Identifiers (PIDs). The Cost Map Service, Endpoint Cost Service (ECS) and Endpoint Ranking Service then provide ISP-defined costs and rankings for connections among the specified endpoints and PIDs and thus incentives for application clients to connect to ISP preferred locations, for instance, to reduce their costs. For the reasons outlined in the ALTO problem statement [RFC5693] and requirement AR-14 of [RFC6708], ALTO does not disseminate network metrics that change frequently. In a network, the costs can fluctuate for many reasons having to do with instantaneous traffic load or due to diurnal patterns of traffic demand or planned events such as network maintenance, holidays or highly publicized events. Thus, an ALTO application wishing to use the Cost Map and Endpoint Cost Service at some future time will have to estimate the state of the network at that time, a process that is, at best, fragile and brittle since the application does not have any visibility into the state of the network. Providing network costs for only the current time thus may not be sufficient, in particular for applications that can schedule their traffic in a span of time, for example by deferring backups or other background traffic to off-peak hours.

In case the ALTO Cost value changes are predictable over a certain period of time and the application does not require immediate data transfer, it can save time to get the whole set of cost values over this period in one single ALTO response. Using this set to schedule data transfers allows optimizing the network resources usage and QoE. ALTO Clients and Servers can also minimize their workload by reducing and accordingly scheduling their data exchanges.

This document extends [RFC7285] to allow an ALTO Server to provide network costs for a given duration of time. A sequence of network

costs across a time span for a given pair of network locations is named an "ALTO Cost Calendar". The Filtered Cost Map Service and Endpoint Cost Service are extended to provide Cost Calendars. In addition to this functional ALTO enhancement, we expect to further save network and storage resources by gathering multiple Cost Values for one cost type into one single ALTO Server response.

In this document, an "ALTO Cost Calendar" is specified in terms of information resources capabilities that are applicable to time-sensitive ALTO metrics. An ALTO Cost Calendar exposes ALTO Cost Values in JSON arrays, see [RFC8259], where each value corresponds to a given time interval. The time intervals as well as other Calendar attributes are specified in the Information Resources Directory (IRD) and in the Server response to allow the ALTO Client to interpret the received ALTO values. Last, the extensions for ALTO Calendars are applicable to any Cost Mode and they ensure backwards compatibility with legacy ALTO Clients - those that only support [RFC7285].

In the rest of this document, Section 3 provides the design characteristics. Sections Section 4 and Section 5 define the formal specifications for the IRD and the information resources. IANA, security and operational considerations are addressed respectively in sections Section 6, Section 7 and Section 8.

1.1. Some recent known uses

A potential use case is the SENSE project, see [SENSE-sdn-e2e-net], who is implementing smart network services to dynamically build end-to-end virtual guaranteed networks across administrative domains, with no manual intervention. The initial SENSE services include informing applications on the availability of bandwidth resources or feasibility of some requested Time-Bandwidth-Product (TBP) during a specific time period. ALTO Calendars can support these services if the Calendar start date and duration cover the period of interest of the requesting applications.

The need of future scheduling of large scale traffic that can be addressed by the ALTO protocol is also motivated by Unicorn, a unified resource orchestration framework for multi-domain, geo-distributed data analytics, see [I-D.xiang-alto-multidomain-analytics].

1.2. Terminology

- ALTO transaction: A request/response exchange between an ALTO Client and an ALTO Server.

- Client: When used with a capital "C", this term refers to an ALTO Client.

- Calendar, Cost Calendar: When used with capitalized words, these terms refer to an ALTO Cost Calendar.

- Endpoint (EP): An endpoint is defined as in Section 2.1 of [RFC7285]. It can be, for example, a peer, a CDN storage location, a physical server involved in a virtual server-supported application, a party in a resource-sharing swarm such as a computation grid, or an online multi-party game.

- Server: When used with a capital "S", this term refers to an ALTO Server.

2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

When the words appear in lower case, they are to be interpreted with their natural language meanings.

3. Overview of ALTO Cost Calendars and terminology

This section gives a non-normative overview of the design. It assumes the reader is familiar with the ALTO protocol [RFC7285]. Normative specifications are given in the following sections.

3.1. ALTO Cost Calendar overview

An ALTO Cost Calendar provided by the ALTO Server provides 2 information items:

- an array of values for a given metric, where each value specifies the metric corresponding to a time interval, where the value array can sometimes be a cyclic pattern that repeats a certain number of times.

- attributes describing the time scope of the Calendar, including the size and number of the intervals and the date of the starting point of the Calendar, allowing an ALTO Client to interpret the values properly.

An ALTO Cost Calendar can be used like a "time table" to figure out the best time to schedule data transfers and also to proactively manage application traffic given predictable events such as expected spike in traffic due to crowd gathering (concerts, sports, etc.), traffic-intensive holidays and network maintenance. A Calendar may be viewed as a synthetic abstraction of, for example, real measurements gathered over previous periods on which statistics have been computed. However, like for any schedule, unexpected network incidents may require the current ALTO Calendar to be updated and re-sent to the ALTO Clients needing it. The "ALTO Incremental Updates Using Server-Sent Events (SSE)" Service [I-D.ietf-alto-incr-update-sse] can be used to update the calendar faster if supported by both the Server and the Client.

Most likely, the ALTO Cost Calendar would be used for the Endpoint Cost Service, assuming that a limited set of feasible Endpoints for a non-real time application is already identified, that they do not need to be accessed immediately and that their access can be scheduled within a given time period. The Filtered Cost Map Service is also applicable as long as the size of the Map allows it.

3.2. ALTO Cost Calendar information features

The Calendar attributes are provided in the Information Resources Directory (IRD) and in ALTO Server responses. The IRD announces attributes without date values in its information resources capabilities, whereas attributes with time dependent values are provided in the "meta" section of Server responses. The ALTO Cost Calendar attributes provide the following information:

- attributes to describe the time scope of the Calendar value array:
  - applicable time interval size for each Calendar value, defined in seconds, that can cover a wide range of values.
  - duration of the Calendar: e.g., the number of intervals provided in the Calendar.
- "calendar-start-time": specifying when the Calendar starts, that is to which date the first value of the Cost Calendar is applicable.
- "repeated": an optional attribute indicating how many iterations of the provided Calendar will have the same values. The Server may use it to allow the Client to schedule its next request and thus save its own workload by reducing processing of similar requests.

Attribute "repeated" may take a very high value if a Calendar represents a cyclic value pattern that the Server considers valid for a long period. In this case, the Server will only update the Calendar values once this period has elapsed or if an unexpected event occurs on the network. See Section 8 for more discussion.

3.3. ALTO Calendar design characteristics

The extensions in this document encode requests and responses using JSON [RFC8259].

In the base protocol [RFC7285] section 11.2.3.6, an ALTO cost is specified as a generic JSONValue [RFC8259], to allow extensions. However, that section 11.2.3.6 states: "An implementation of the protocol in this document ([RFC7285]) SHOULD assume that the cost is a JSONNumber and fail to parse if it is not, unless the implementation is using an extension to this document that indicates when and how costs of other data types are signaled".

The present document extends the definition of a legacy cost map given in [RFC7285] to allow a cost entry to be an array of values, with one value one per time interval, instead of being just one number. Therefore the implementor of this extension MUST consider that a cost entry is an array of values. Specifically, an implementation of this extension MUST parse the "number-of-intervals" attribute of the "calendar-attributes" in an IRD entry announcing a service providing Cost Calendar. The implementation then will know that a cost entry of the service will be an array of values, and the expected size of the array is that specified by the "number-of-intervals" attribute.

To realize an ALTO Calendar, this document extends: the IRD and the ALTO requests and responses for Cost Calendars.

This extension is designed to be lightweight and to ensure backwards compatibility with base protocol ALTO Clients and with other extensions. It relies on section 8.3.7 "Parsing of Unknown Fields" of [RFC7285] that writes: "Extensions may include additional fields within JSON objects defined in this document. ALTO implementations MUST ignore unknown fields when processing ALTO messages."

The Calendar-specific capabilities are integrated in the information resources of the IRD and in the "meta" member of ALTO responses to Cost Calendars requests. A Calendar and its capabilities are associated with a given information resource and within this information resource with a given cost type. This design has several advantages:

- it does not introduce a new mode,
- it does not introduce new media types,
- it allows an ALTO Server to offer Calendar capabilities on a cost type, with attributes values adapted to each information resource.

The applicable Calendared information resources are:

- the Filtered Cost Map,
- the Endpoint Cost Map.

The ALTO Server can choose in which frequency it provides cost Calendars to ALTO Clients. It may either provide Calendar updates starting at the request date, or carefully schedule its updates so as to take profit from a potential repetition/periodicity of Calendar values.

3.3.1. ALTO Cost Calendar for all cost modes

An ALTO Cost Calendar is well-suited for values encoded in the "numerical" mode. Actually, a Calendar can also represent metrics in other modes considered as compatible with time-varying values. For example, types of Cost values such as JSONBool can also be calendared, as their value may be 'true' or 'false' depending on given time periods or likewise, values represented by strings, such as "medium", "high", "low", "blue", "open".

Note also that a Calendar is suitable as well for time-varying metrics provided in the "ordinal" mode, if these values are time-varying and the ALTO Server provides updates of cost value based preferences.

3.3.2. Compatibility with legacy ALTO Clients

The ALTO protocol extensions for Cost Calendars have been defined so as to ensure that Calendar capable ALTO Servers can provide legacy ALTO Clients with legacy information resources as well. That is, a legacy ALTO Client can request resources and receive responses as specified in [RFC7285].

A Calendar-aware ALTO Server MUST implement the base protocol specified in [RFC7285].

As a consequence, when a metric is available as a Calendar array, it also MUST be available as a single value as required by [RFC7285]. The Server, in this case, provides the current value of the metric to

either Calendar-aware Clients not interested in future or time-based values, or Clients implementing [RFC7285] only.

For compatibility with legacy ALTO Clients specified in [RFC7285], calendared information resources are not applicable for full cost maps for the following reason: a legacy ALTO Client would receive a calendared cost map via an HTTP 'GET' command. As specified in section 8.3.7 of [RFC7285], it will ignore the Calendar Attributes indicated in the "meta" of the responses. Therefore, lacking information on Calendar attributes, it will not be able to correctly interpret and process the values of the received array of Calendar cost values.

Therefore, calendared information resources MUST be requested via the Filtered Cost Map Service or the Endpoint Cost Service, using a POST method.

4. ALTO Calendar specification: IRD extensions

The Calendar attributes in the IRD information resources capabilities carry constant dateless values. A Calendar is associated with an information resource rather than a cost type. For example, a Server can provide a "routingcost" Calendar for the Filtered Cost Map Service at a granularity of one day and a "routingcost" Calendar for the Endpoint Cost Service at a finer granularity but for a limited number of endpoints. An example IRD with Calendar specific features is provided in Section 4.3.

4.1. Calendar attributes in the IRD resource capabilities

A Cost Calendar for a given cost type MUST be indicated in the IRD by an object of type CalendarAttributes. A CalendarAttributes object is represented by the "calendar-attributes" member of a resource entry. Each CalendarAttributes object applies to a set of one or more cost types. A cost type name MUST NOT appear more than once in the "calendar-attributes" member of a resource entry; multiple appearances of a cost type name in the CalendarAttributes object of the "calendar-attributes" member MUST cause the ALTO Client to ignore any occurrences of this name beyond the first encountered occurrence.

An ALTO Server SHOULD specify the "time-interval-size" in the IRD as the smallest it is able to provide. A Client that needs a longer interval can aggregate multiple cost values to obtain it.

The encoding format for object CalendarAttributes, using JSON [RFC8259], is as follows:

CalendarAttributes calendar-attributes <1..*>;

```
object{
  JSONString cost-type-names <1..*>;
  JSONNumber time-interval-size;
  JSONNumber number-of-intervals;
} CalendarAttributes;
```

o "cost-type-names":

  * An array of one or more elements indicating the cost-type-names in the IRD entry to which the capabilities apply.

o "time-interval-size":

  * is the duration of an ALTO Calendar time interval in a unit of seconds. A "time-interval-size" value contains a non-negative JSONNumber. Example values are: 300 and 7200, meaning that each Calendar value applies on a time interval that lasts 5 minutes and 2 hours, respectively. Since an interval size (e.g., 100 ms) can be smaller than the unit, the value specified may be a floating point (e.g., 0.1). Both ALTO Clients and Servers should be aware of potential precision issues caused by using floating point numbers; for example, the floating number 0.1 cannot be represented precisely using a finite number of binary bits. To improve interoperability and be consistent with [RFC7285] on the use of float point numbers, the Server and the Client SHOULD use IEEE 754 double-precision floating point [IEEE.754.2008] to store this value.

o "number-of-intervals":

  * is a strictly positive integer (greater or equal to 1), that indicates the number of values of the Cost Calendar array.

- Providing attribute "cost-type-names" together with "time-interval-size" and "number-of-intervals" improves the readability of the Calendar attributes specified for an IRD resource and avoids confusion with Calendar attributes of other cost types.

- Multiplying 'time-interval-size' by 'number-of-intervals' provides the duration of the provided Calendar. For example, an ALTO Server may provide a Calendar for ALTO values changing every 'time-interval-size' equal to 5 minutes. If 'number-of-intervals' has the value 12, then the duration of the provided Calendar is "1 hour".

4.2. Calendars in a delegate IRD

It may be useful to distinguish IRD resources supported by the base ALTO protocol from resources supported by its extensions. To achieve this, one option, is that a "root" ALTO Server implementing base protocol resources and running at a given domain, delegates "specialized" information resources such as the ones providing Cost Calendars, to another ALTO Server running in a subdomain. The "root" ALTO Server can provide a Calendar-specific resource entry where it specifies the URI allowing to retrieve the location of a Calendar-aware Server and relevant resources. This option is described in Section 9.2.4 "Delegation using IRDs" of [RFC7285].

This document provides an example, where a "root" ALTO Server runs in a domain called "alto.example.com". It delegates the announcement of Calendars capabilities to an ALTO Server running in a subdomain called "custom.alto.example.com". The location of the "delegate Calendar IRD" is assumed to be indicated in the "root" IRD by the resource entry: "custom-calendared-resources".

Another benefit of delegation is that some cost types for some resources may be more advantageous as Cost Calendars and it makes few sense to get them as a single value. For example, if a cost type has predictable and frequently changing values, calendared in short time intervals such as a minute, it saves time and network resources to track the cost values via a focused delegate Server rather than the more general "root" Server.

4.3. Example IRD with ALTO Cost Calendars

This section provides an example ALTO Server IRD that supports various cost metrics and cost modes. In particular, since [RFC7285] makes it mandatory, the Server uses metric "routingcost" in the "numerical" mode.

For illustrative purposes, this section introduces 3 other fictitious example metrics and modes that should be understood as examples and should not be used or considered as normative.

The cost type names used in the example IRD are as follows:

- "num-routingcost": refers to metric "routingcost" in the numerical mode as defined in [RFC7285] and registered with IANA.

- "num-owdelay": refers to fictitious performance metric "owdelay" in the "numerical" mode,to reflect the one-way packet transmission delay on a path. A related performance metric is currently under definition in [I-D.ietf-alto-performance-metrics].

- "num-throughputrating": refers to fictitious metric "throughputrating" in the "numerical" mode, to reflect the provider preference in terms of end to end throughput.

- "string-servicestatus": refers to fictitious metric "servicestatus" containing a string, to reflect the availability, defined by the provider, of for instance path connectivity.

The example IRD includes 2 particular URIs providing Calendars:

- "https://custom.alto.example.com/calendar/costmap/filtered": a filtered cost map in which Calendar capabilities are indicated for cost type names: "num-routingcost", "num-throughputrating" and "string-servicestatus",

- "https://custom.alto.example.com/calendar/endpointcost/lookup": an endpoint cost map in which Calendar capabilities are indicated for cost type names: "num-routingcost", "num-owdelay", "num-throughputrating", "string-servicestatus".

The design of the Calendar capabilities allows some Calendars with the same cost type name to be available in several information resources with different Calendar Attributes. This is the case for Calendars on "num-routingcost", "num-throughputrating" and "string-servicestatus", available in both the Filtered Cost map and Endpoint Cost Service, but with different time interval sizes for "num-throughputrating" and "string-servicestatus".

--- Client to Server request for IRD ----------

```
GET /calendars-directory HTTP/1.1
Host: custom.alto.example.com
Accept: application/alto-directory+json,application/alto-error+json
```

--- Server response to Client -----------------

```
HTTP/1.1 200 OK
Content-Length: 2542
Content-Type: application/alto-directory+json

{
  "meta" : {
    "default-alto-network-map" : "my-default-network-map",
    "cost-types": {
      "num-routingcost": {
        "cost-mode" : "numerical",
        "cost-metric" : "routingcost"
      },
```

```
        "num-owdelay": {
          "cost-mode"  : "numerical",
          "cost-metric": "owdelay"
        },
        "num-throughputrating": {
          "cost-mode"  : "numerical",
          "cost-metric": "throughputrating"
        },
        "string-servicestatus": {
          "cost-mode"  : "string",
          "cost-metric": "servicestatus"
        }
      }
    },
    "resources" : {
      "filtered-cost-map-calendar" : {
        "uri" :
          "https://custom.alto.example.com/calendar/costmap/filtered",
        "media-type" : "application/alto-costmap+json",
        "accepts" : "application/alto-costmapfilter+json",
        "capabilities" : {
          "cost-constraints" : true,
          "cost-type-names"  : [ "num-routingcost",
                                 "num-throughputrating",
                                 "string-servicestatus" ],
          "calendar-attributes" : [
            {"cost-type-names" : [ "num-routingcost",
                                   "num-throughputrating" ],
             "time-interval-size" : 7200,
             "number-of-intervals" : 12
            },
            {"cost-type-names" : [ "string-servicestatus" ],
             "time-interval-size" : 1800,
             "number-of-intervals" : 48
            }
          ]
        },
        "uses": [ "my-default-network-map" ]
      },
      "endpoint-cost-calendar-map" : {
        "uri" :
        "https://custom.alto.example.com/calendar/endpointcost/lookup",
        "media-type" : "application/alto-endpointcost+json",
        "accepts" : "application/alto-endpointcostparams+json",
        "capabilities" : {
          "cost-constraints" : true,
          "cost-type-names" : [ "num-routingcost",
                                "num-owdelay",
```

```
                              "num-throughputrating",
                              "string-servicestatus" ],
          "calendar-attributes" : [
            {"cost-type-names" : [ "num-routingcost" ],
             "time-interval-size" : 3600,
             "number-of-intervals" : 24
            },
            {"cost-type-names" : [ "num-owdelay" ],
             "time-interval-size" : 300,
             "number-of-intervals" : 12
            },
            {"cost-type-names" : [ "num-throughputrating" ],
             "time-interval-size" : 60,
             "number-of-intervals" : 60
            },
            {"cost-type-names" : [ "string-servicestatus" ],
             "time-interval-size" : 120,
             "number-of-intervals" : 30
            }
          ]
        }
      }
    }
  }
```

In this example IRD, for the Filtered Cost Map Service:

- the Calendar for "num-routingcost" and "num-throughputrating" is an array of 12 values each provided on a time interval lasting 7200 seconds (2 hours).

- the Calendar for "string-servicestatus": "is an array of 48 values each provided on a time interval lasting 1800 seconds (30 minutes).

For the Endpoint Cost Service:

- the Calendar for "num-routingcost": is an array of 24 values each provided on a time interval lasting 3600 seconds (1 hour).

- the Calendar for "owdelay": is an array of 12 values each provided on a time interval lasting 300 seconds (5 minutes).

- the Calendar for "num-throughputrating": is an array of 60 values each provided on a time interval lasting 60 seconds (1 minute).

- the Calendar for "string-servicestatus": "is an array of 30 values each provided on a time interval lasting 120 seconds (2 minutes).

5. ALTO Calendar specification: Service Information Resources

This section documents extensions to two basic ALTO information resources (Filtered Cost Maps and Endpoint Cost Service) to provide calendared information services for them.

Both extensions return calendar start time (calendar-start-time, a point in time), which MUST be specified as an HTTP "Date" header field using the IMF-fixdate format specified in Section 7.1.1.1 of [RFC7231]. Note that the IMF-fixdate format uses "GMT", not "UTC", to designate the time zone, as in this example:

Date: Tue, 15 Nov 2014 08:12:31 GMT

5.1. Calendar extensions for Filtered Cost Maps (FCM)

A legacy ALTO Client requests and gets Filtered Cost Map responses as specified in [RFC7285].

5.1.1. Calendar extensions in Filtered Cost Map requests

The input parameters of a "legacy" request for a filtered cost map, defined by object ReqFilteredCostMap in section 11.3.2 of [RFC7285], are augmented with one additional member.

A Calendar-aware ALTO Client requesting a Calendar on a given Cost Type for a filtered cost map resource having Calendar capabilities MUST add the following field to its input parameters:

JSONBoolean calendared<1..*>;

This field is an array of 1 to N boolean values, where N is the number of requested metrics. Each entry corresponds to the requested metric at the same array position. Each boolean value indicates whether or not the ALTO Server should provide the values for this cost type as a Calendar. The array MUST contain exactly N boolean values, otherwise, the Server returns an error.

This field MUST NOT be included if no member "calendar-attributes" is specified in this information resource.

If a value of field 'calendared' is 'true' for a cost type name for which no Calendar attributes have been specified: an ALTO Server, whether it implements the extensions of this document or only implements [RFC7285], MUST ignore it and return a response with a single cost value as specified in [RFC7285].

If this field is not present, it MUST be assumed to have only values equal to 'false'.

A Calendar-aware ALTO Client that supports requests for only one cost type at a time and wants to request a Calendar MUST provide an array of 1 element:

"calendared" : [true];

A Calendar-aware ALTO Client that supports requests for more than one cost types at a time, as specified in [RFC8189] MUST provide an array of N values set to 'true' or 'false', depending whether it wants the applicable cost type values as a single or calendared value.

5.1.2. Calendar extensions in Filtered Cost Map responses

In a calendared ALTO Filtered Cost Map, a cost value between a source and a destination is a JSON array of JSON values. An ALTO Calendar values array has a number of values equal to the value of member "number-of-intervals" of the Calendar attributes that are indicated in the IRD. These attributes will be conveyed as metadata in the Filtered Cost Map response. Each element of the array is valid for the time-interval that matches its array position.

The FCM response conveys metadata among which:

- some are not specific to Calendars and ensure compatibility with [RFC7285] and [RFC8189]
- some are specific to Calendars.

The non Calendar specific "meta" fields of a calendared Filtered Cost Map response MUST include at least:

- if the ALTO Client requests cost values for one cost type at a time only: the "meta" fields specified in [RFC7285] for these information service responses:
  - "dependent-vtags ",
  - "cost-type" field.
- if the ALTO Client implements the Multi-Cost ALTO extension specified in [RFC8189] and requests cost values for several Cost Types at a time: the "meta" fields specified in [RFC8189] for these information service responses:
  - "dependent-vtags ",

* "cost-type" field with value set to '{}', for backwards compatibility with [RFC7285].

* "multi-cost-types" field.

If the Client request does not provide member "calendared" or if it provides it with a value equal to 'false', for all the requested Cost Types, then the ALTO Server response is exactly as specified in [RFC7285] and [RFC8189].

If the value of member "calendared" is equal to 'false' for a given requested cost type, the ALTO Server MUST return, for this cost type, a single cost value as specified in [RFC7285].

If the value of member "calendared" is equal to 'true' for a given requested cost type, the ALTO Server returns, for this cost type, a cost value Calendar as specified above in this section. In addition to the above cited non Calendar specific "meta" members, the Server MUST provide a Calendar specific metadata field.

The Calendar specific "meta" field that a calendared Filtered Cost Map response MUST include is a member called "calendar-response-attributes", that describes properties of the Calendar and where:

o member "calendar-response-attributes" is an array of one or more objects of type "CalendarResponseAttributes".

o each "CalendarResponseAttributes" object in the array is specified for one or more cost types for which the value of member "calendared" is equal to 'true' and for which a Calendar is provided for the requested information resource.

o the "CalendarResponseAttributes" object that applies to a cost type name has a corresponding "CalendarAttributes" object defined for this cost type name in the IRD capabilities of the requested information resource. The members of a "CalendarResponseAttributes" object include all the members of the corresponding "CalendarAttributes" object.

The format of member "CalendarResponseAttributes is defined as follows:

CalendarResponseAttributes calendar-response-attributes <1..*>;

```
object{
  [JSONString cost-type-names <1..*>];
  JSONString calendar-start-time;
  JSONNumber time-interval-size;
  JSONNumber number-of-intervals;
  [JSONNumber repeated;]
} CalendarResponseAttributes;
```

Object CalendarResponseAttributes has the following attributes:

- "cost-type-names": is an array of one or more cost-type-names to which the capabilities apply and for which a Calendar has been requested. The value of this member is a subset of the "cost-type-names" array specified in the corresponding IRD Calendar attributes.

- "calendar-start-time": indicates the date at which the first value of the Calendar applies. The value provided for the "calendar-start-time" attribute SHOULD NOT be later than the request date.

- "time-interval-size": as specified in Section 4.1 and with the same value.

- "number-of-intervals": as specified in Section 4.1 and with the same value.

- "repeated": is an optional field provided for Calendars. It is an integer N greater or equal to '1' that indicates how many iterations of the Calendar value array starting at the date indicated by "calendar-start-time" have the same values. The number N includes the iteration provided in the returned response.

For example: suppose the "calendar-start-time" member has value "Mon, 30 Jun 2014 00:00:00 GMT", the "time-interval-size" member has value '3600', the "number-of-intervals" member has value '24' and the value of member "repeated" is equal to '4'. This means that the Calendar values are the same on Monday, Tuesday, Wednesday and Thursday on a period of 24 hours starting at 00:00:00 GMT. The ALTO Client thus may use the same Calendar for the next 4 days starting at "calendar-start-time" and will only need to request a new one for Friday July 4th at 00:00:00 GMT.

Attribute "repeated" may take a very high value if a Calendar represents a cyclic value pattern that the Server considers valid for a long period and hence will only update once this period has elapsed or if an unexpected event occurs on the network. In the latter case, the Client will be notified if it uses the "ALTO Incremental Updates Using Server-Sent Events (SSE)" Service, specified in

[I-D.ietf-alto-incr-update-sse]. To this end, it is RECOMMENDED that ALTO Servers providing ALTO Calendars also provide the "ALTO Incremental Updates Using Server-Sent Events (SSE)" Service that is specified in [I-D.ietf-alto-incr-update-sse]. Likewise, ALTO Clients capable of using ALTO Calendars SHOULD also use the SSE Service. See also discussion in Section 8 "Operational Considerations".

5.1.3. Use case and example: FCM with a bandwidth Calendar

An example of non-real time information that can be provisioned in a Calendar is the expected path throughput. While the transmission rate can be measured in real time by end systems, the operator of a data center is in the position of formulating preferences for given paths, at given time periods to avoid traffic peaks due to diurnal usage patterns. In this example, we assume that an ALTO Client requests a Calendar of network provider defined throughput ratings, as specified in the IRD, to schedule its bulk data transfers as described in the use cases.

In the example IRD, Calendars for cost type name "num-throughputrating" are available for the information resources: "filtered-cost-calendar-map" and "endpoint-cost-calendar-map". The ALTO Client requests a Calendar for "num-throughputrating" via a POST request for a filtered cost map.

We suppose in the present example that the ALTO Client sends its request on Tuesday July 1st 2014 at 13:15. The Server returns Calendars with arrays of 12 numbers for each source and destination pair. The values for metric "throughputrating", in this example, are assumed to be encoded in 2 digits.

```
POST /calendar/costmap/filtered HTTP/1.1
Host: alto.example.com
Content-Length: 208
Content-Type: application/alto-costmapfilter+json
Accept: application/alto-costmap+json,application/alto-error+json

{
  "cost-type" : {"cost-mode" : "numerical",
                 "cost-metric" : "throughputrating"},
  "calendared" : [true],
  "pids" : {
    "srcs" : [ "PID1", "PID2" ],
    "dsts" : [ "PID1", "PID2", "PID3" ]
  }
}

HTTP/1.1 200 OK
```

```
Content-Length: 1013
Content-Type: application/alto-costmap+json

{
  "meta" : {
    "dependent-vtags" : [
      {"resource-id": "my-default-network-map",
       "tag": "3ee2cb7e8d63d9fab71b9b34cbf764436315542e"
      }
    ],
    "cost-type" : {"cost-mode" : "numerical",
                   "cost-metric" : "throughputrating"},
    "calendar-response-attributes" : [
      {"calendar-start-time" : "Tue, 1 Jul 2014 13:00:00 GMT",
       "time-interval-size" : 7200,
       "number-of-intervals" : 12}
    ]
  },
  "cost-map" : {
    "PID1": { "PID1": [ 1, 12, 14, 18, 14, 14,
                        14, 18, 19, 20, 11, 12],
              "PID2": [13,  4, 15, 16, 17, 18,
                        19, 20, 11, 12, 13, 14],
              "PID3": [20, 20, 18, 14, 12, 12,
                        14, 14, 12, 12, 14, 16] },
    "PID2": { "PID1": [17, 18, 19, 10, 11, 12,
                        13, 14, 15, 16, 17, 18],
              "PID2": [20, 20, 18, 16, 14, 14,
                        14, 16, 16, 16, 14, 16],
              "PID3": [20, 20, 18, 14, 12, 12,
                        14, 14, 12, 12, 14, 16] }
  }
}
```

5.2. Calendar extensions in the Endpoint Cost Service

This document extends the Endpoint Cost Service, as defined in {11.5.1} of [RFC7285], by adding new input parameters and capabilities, and by returning JSONArrays instead of JSONNumbers as the cost values. The media type {11.5.1.1} and HTTP method {11.5.1.2} are unchanged.

5.2.1. Calendar specific input in Endpoint Cost requests

The extensions to the requests for calendared Endpoint Cost Maps are the same as for the Filtered Cost Map Service, specified in section Section 5.1.1 of this draft.

The ReqEndpointCostMap object for a calendared ECM request will have the following format:

```
object {
  [CostType cost-type;]
  [CostType multi-cost-types<1..*>;]
  [JSONBoolean calendared<1..*>;]
  EndpointFilter endpoints;
} ReqEndpointCostMap;

object {
  [TypedEndpointAddr srcs<0..*>;]
  [TypedEndpointAddr dsts<0..*>;]
} EndpointFilter;
```

5.2.2. Calendar attributes in the Endpoint Cost response

The "meta" field of a calendared Endpoint Cost response MUST include at least:

- if the ALTO Client supports cost values for one cost type at a time only: the "meta" fields specified in {11.5.1.6} of [RFC7285] for the Endpoint Cost response:
  - "cost-type" field.
- if the ALTO Client supports cost values for several cost types at a time, as specified in [RFC8189] : the "meta" fields specified in [RFC8189] for the the Endpoint Cost response:
  - "cost-type" field with value set to '{}', for backwards compatibility with [RFC7285].
  - "multi-cost-types" field.

If the Client request does not provide member "calendared" or if it provides it with a value equal to 'false', for all the requested Cost Types, then the ALTO Server response is exactly as specified in [RFC7285] and [RFC8189].

If the ALTO Client provides member "calendared" in the input parameters with a value equal to 'true' for given requested Cost Types, the "meta" member of a calendared Endpoint Cost response MUST include, for these cost types, an additional member "calendar-response-attributes", the contents of which obey the same rules as for the Filtered Cost Map Service, specified in Section 5.1.2. The Server response is thus changed as follows, with respect to [RFC7285] and [RFC8189]:

- the "meta" member has one additional field "CalendarResponseAttributes", as specified for the Filtered Cost Map Service,

- the calendared costs are JSONArrays instead of the JSONNumbers format used by legacy ALTO implementations. All arrays have a number of values equal to 'number-of-intervals'.

If the value of member "calendared" is equal to 'false' for a given requested cost type, the ALTO Server MUST return, for this cost type, a single cost value as specified in [RFC7285].

5.2.3. Use case and example: ECS with a routingcost Calendar

Let us assume an Application Client is located in an end system with limited resources and having access to the network that is either intermittent or provides an acceptable quality in limited but predictable time periods. Therefore, it needs to schedule both its resource-greedy networking activities and its ALTO transactions.

The Application Client has the choice to trade content or resources with a set of Endpoints and needs to decide with which one it will connect and at what time. For instance, the Endpoints are spread in different time-zones, or have intermittent access. In this example, the 'routingcost' is assumed to be time-varying, with values provided as ALTO Calendars.

The ALTO Client associated with the Application Client queries an ALTO Calendar on 'routingcost' and will get the Calendar covering the 24 hours time period "containing" the date and time of the ALTO Client request.

For cost type "num-routingcost", the solicited ALTO Server has defined 3 different daily patterns each represented by a Calendar, to cover the week of Monday June 30th at 00:00 to Sunday July 6th 23:59:

- C1 for Monday, Tuesday, Wednesday, Thursday, (weekdays)

- C2 for Saturday, Sunday, (weekend)

- C3 for Friday (maintenance outage on July 4, 2014 from 02:00:00 GMT to 04:00:00 GMT, or big holiday such as New Year evening).

In the following example, the ALTO Client sends its request on Tuesday July 1st 2014 at 13:15.

The "routingcost" values are assumed to be encoded in 3 digits.

```
  POST /calendar/endpointcost/lookup HTTP/1.1
  Host: alto.example.com
  Content-Length: 290
  Content-Type: application/alto-endpointcostparams+json
  Accept: application/alto-endpointcost+json,application/alto-error+json

  {
    "cost-type" : {"cost-mode" : "numerical",
                   "cost-metric" : "routingcost"},
    "calendared" : [true],
    "endpoints" : {
      "srcs": [ "ipv4:192.0.2.2" ],
      "dsts": [
        "ipv4:192.0.2.89",
        "ipv4:198.51.100.34",
        "ipv4:203.0.113.45",
        "ipv6:2001:db8::10"
      ]
    }
  }

  HTTP/1.1 200 OK

  Content-Length: 1318
  Content-Type: application/alto-endpointcost+json

  {
    "meta" : {
      "cost-type" : {"cost-mode" : "numerical",
                     "cost-metric" : "routingcost"},
      "calendar-response-attributes" : [
        {"calendar-start-time" : "Mon, 30 Jun 2014 00:00:00 GMT",
         "time-interval-size" : 3600,
         "number-of-intervals" : 24,
         "repeated": 4
        }
      ]
    },
    "endpoint-cost-map" : {
      "ipv4:192.0.2.2": {
        "ipv4:192.0.2.89"    : [100, 100, 100, 100, 100, 150,
                                200, 300, 300, 300, 300, 250,
                                250, 300, 300, 300, 300, 300,
                                400, 250, 250, 200, 150, 150],
        "ipv4:198.51.100.34" : [ 80,  80,  80,  80, 150, 150,
                                250, 400, 400, 450, 400, 200,
                                200, 350, 400, 400, 400, 350,
                                500, 200, 200, 200, 100, 100],
```

```
        "ipv4:203.0.113.45"  : [300, 400, 250, 250, 200, 150,
                                150, 100, 100, 100, 100, 100,
                                100, 100, 100, 100, 100, 150,
                                200, 300, 300, 300, 300, 250],
        "ipv6:2001:db8::10"  : [200, 250, 300, 300, 300, 300,
                                250, 300, 300, 300, 300, 350,
                                300, 400, 250, 150, 100, 100,
                                100, 150, 200, 250, 250, 300]
      }
    }
  }
```

When the Client gets the Calendar for "routingcost", it sees that the "calendar-start-time" is Monday at 00h00 GMT and member "repeated" is equal to '4'. It understands that the provided values are valid until Thursday included and will only need to get a Calendar update on Friday.

5.2.4. Use case and example: ECS with a multi-cost Calendar for routingcost and owdelay

In this example, it is assumed that the ALTO Server implements multi-cost capabilities, as specified in [RFC8189] . That is, an ALTO Client can request and receive values for several cost types in one single transaction. An illustrating use case is a path selection done on the basis of 2 metrics: routing cost and owdelay.

As in the previous example, the IRD indicates that the ALTO Server provides "routingcost" Calendars in terms of 24 time intervals of 1 hour (3600 seconds) each.

For metric "owdelay", the IRD indicates that the ALTO Server provides Calendars in terms of 12 time intervals values lasting each 5 minutes (300 seconds).

In the following example transaction, the ALTO Client sends its request on Tuesday July 1st 2014 at 13:15.

This example assumes that the values of metric "owdelay" and "routingcost" are encoded in 3 digits.

```
POST calendar/endpointcost/lookup HTTP/1.1
Host: alto.example.com
Content-Length: 373
Content-Type: application/alto-endpointcostparams+json
Accept: application/alto-endpointcost+json,application/alto-error+json

{
```

```
    "cost-type" : {},
    "multi-cost-types" : [
      {"cost-mode" : "numerical", "cost-metric" : "routingcost"},
      {"cost-mode" : "numerical", "cost-metric" : "owdelay"}
    ],
    "calendared" : [true, true],
    "endpoints" : {
      "srcs": [ "ipv4:192.0.2.2" ],
      "dsts": [
        "ipv4:192.0.2.89",
        "ipv4:198.51.100.34",
        "ipv4:203.0.113.45",
        "ipv6:2001:db8::10"
      ]
    }
  }

  HTTP/1.1 200 OK
  Content-Length: 2111
  Content-Type: application/alto-endpointcost+json

  {
    "meta" : {
      "multi-cost-types" : [
        {"cost-mode" : "numerical", "cost-metric" : "routingcost"},
        {"cost-mode" : "numerical", "cost-metric" : "owdelay"}
      ],
      "calendar-response-attributes" : [
        {"cost-type-names" : "num-routingcost",
           "calendar-start-time" : "Mon, 30 Jun 2014 00:00:00 GMT",
           "time-interval-size" : 3600,
           "number-of-intervals" : 24,
           "repeated": 4 },
        {"cost-type-names" : "num-owdelay",
           "calendar-start-time" : "Tue, 1 Jul 2014 13:00:00 GMT",
           "time-interval-size" : 300,
           "number-of-intervals" : 12}
      ]
    },
    "endpoint-cost-map" : {
      "ipv4:192.0.2.2": {
        "ipv4:192.0.2.89"    : [[100, 100, 100, 100, 100, 150,
                                 200, 300, 300, 300, 300, 250,
                                 250, 300, 300, 300, 300, 300,
                                 400, 250, 250, 200, 150, 150],
                                [ 20, 400,  20,  80,  80,  90,
                                 100,  90,  60,  40,  30,  20]],
        "ipv4:198.51.100.34" : [[ 80,  80,  80,  80, 150, 150,
```

```
                                250, 400, 400, 450, 400, 200,
                                200, 350, 400, 400, 400, 350,
                                500, 200, 200, 200, 100, 100],
                               [ 20,  20,  50,  30,  30,  30,
                                 30,  40,  40,  30,  20,  20]],
        "ipv4:203.0.113.45"  : [[300, 400, 250, 250, 200, 150,
                                150, 100, 100, 100, 100, 100,
                                100, 100, 100, 100, 100, 150,
                                200, 300, 300, 300, 300, 250],
                               [100,  90,  80,  60,  50,  50,
                                 40,  40,  60,  90, 100,  80]],
        "ipv6:2001:db8::10"  : [[200, 250, 300, 300, 300, 300,
                                250, 300, 300, 300, 300, 350,
                                300, 400, 250, 150, 100, 100,
                                100, 150, 200, 250, 250, 300],
                               [ 40,  40,  40,  40,  50,  50,
                                 50,  20,  10,  15,  30,  40]]
      }
    }
  }
```

When receiving the response, the Client sees that the Calendar values for metric "routingcost" are repeated for 4 iterations. Therefore, in its next requests until the "routingcost" Calendar is expected to change, the Client will only need to request a Calendar for "owdelay".

Without the ALTO Calendar extensions, the ALTO Client would have no clue on the dynamicity of the metric value change and would spend needless time requesting values at an inappropriate pace. In addition, without the Multi-Cost ALTO capabilities, the ALTO Client would duplicate this waste of time as it would need to send one request per cost metric.

6. IANA Considerations

This document does not define any new media types or introduce any new IANA considerations.

7. Security Considerations

As an extension of the base ALTO protocol [RFC7285], this document fits into the architecture of the base protocol, and hence the Security Considerations (Section 15) of the base protocol fully apply when this extension is provided by an ALTO Server. For example, the same authenticity and integrity considerations (Section 15.1 of [RFC7285] still fully apply; the same considerations for the privacy of ALTO users (Section 15.4 of [RFC7285]) also still fully apply.

The calendaring information provided by this extension requires additional considerations on three security considerations discussed in the base protocol: potential undesirable guidance to Clients (Section 15.2 of [RFC7285]), confidentiality of ALTO information (Section 15.2 of [RFC7285]), and availability of ALTO (Section 15.5 of [RFC7285]). For example, by providing network information in the future in a Calendar, this extension may improve availability of ALTO, when the ALTO Server is unavailable but related information is already provided in the Calendar.

For confidentiality of ALTO information, an operator should be cognizant that this extension may introduce a new risk: a malicious ALTO Client may get information for future events that are scheduled through Calendaring. Possessing such information, the malicious Client may use it to generate massive connections to the network at times where its load is expected to be high.

To mitigate this risk, the operator should address the risk of ALTO information being leaked to malicious Clients or third parties. As specified in Section 15.3.2 ("Protection Strategies") of [RFC7285], the ALTO Server should authenticate ALTO Clients and use the Transport Layer Security (TLS) protocol so that Man In The Middle (MITM) attacks to intercept an ALTO Calendar are not possible. [RFC7285] ensures the availability of such a solution in its Section 8.3.5. "Authentication and Encryption", which specifies that: "ALTO Server implementations as well as ALTO Client implementations MUST support the "https" URI scheme of [RFC2818] and Transport Layer Security (TLS) of [RFC5246]".

[RFC8446] specifies TLS 1.3 and writes in its section 1: "While TLS 1.3 is not directly compatible with previous versions, all versions of TLS incorporate a versioning mechanism which allows Clients and Servers to interoperably negotiate a common version if one is supported by both peers". ALTO Clients and Servers SHOULD support both TLS 1.3 [RFC8446] and TLS 1.2 [RFC5246], and MAY support and use newer versions of TLS as long as the negotiation process succeeds.

The operator should be cognizant that the preceding mechanisms do not address all security risks. In particular, they will not help in the case of "malicious Clients" possessing valid credentials to authenticate. The threat here can be that legitimate Clients have become subverted by an attacker and are now 'bots' being asked to participate in a DDoS attack. The Calendar information would be valuable information for when to persecute a DDoS attack. A mechanism such as a monitoring system that detects abnormal behaviors may still be needed.

To avoid malicious or erroneous guidance from ALTO information, an ALTO Client should be cognizant that using calendaring information can have risks: (1) Calendar values, especially in "repeated" Calendars may be only statistical, and (2) future events may change. Hence, a more robust ALTO Client should adapt and extend protection strategies specified in Section 15.2 of the base protocol. For example, to be notified immediately when a particular ALTO value that the Client depends on changes, it is RECOMMENDED that both the ALTO Client and ALTO Server using this extension support "ALTO Incremental Updates Using Server-Sent Events(SSE)" Service [I-D.ietf-alto-incr-update-sse].

8. Operational Considerations

It is important that both the operator of the network and the operator of the applications consider both the feedback aspect and the prediction-based (uncertainty) aspect of using the Cost Calendar.

First consider the feedback aspect and consider the Cost Calendar as a traffic-aware map service (e.g., Google Maps). Using the service without considering its own effect, a large fleet can turn a not-congested road into a congested one; a large number of individual cars each choosing a road with light traffic ("cheap link") can also result in congestion or result in a less optimal global outcome (e.g., the Braess' Paradox [Braess-paradox]).

Next consider the prediction aspect. Conveying ALTO Cost Calendars tends to reduce the on-the-wire data exchange volume compared to multiple single cost ALTO transactions. An application using Calendars has a set of time-dependent values upon which it can plan its connections in advance with no need for the ALTO Client to query information at each time. Additionally, the Calendar response attribute "repeated", when provided, saves additional data exchanges in that it indicates that the ALTO Client does not need to query Calendars during a period indicated by this attribute. The preceding is true only when "accidents" do not happen.

Although individual network operators and application operators can choose their own approaches to address the aforementioned issues, this document recommends the following considerations. First, a typical approach to reducing instability and handling uncertainty is to ensure timely update of information. The SSE Service as discussed in Section 7 can handle updates, if supported by both the Server and the Client. Second, when a network operator updates the Cost Calendar and when an application reacts to the update, they should consider the feedback effects. This is the best approach even though there is theoretical analysis [Selfish-routing-Roughgarden-thesis] and Internet based evaluation [Selfish-routing-Internet-eval] showing

that uncoordinated behaviors do not always cause substantial sub-optimal results.

High-resolution intervals may be needed when values change, sometimes during very small time intervals but in a significant manner. A way to avoid conveying too many entries is to leverage on the "repeated" feature. A Server can smartly set the Calendar start time and number of intervals so as to declare them "repeated" for a large number of periods, until the Calendar values change and are conveyed to requesting Clients.

The newer JSON Data Interchange Format specification [RFC8259] used in ALTO Calendars replaces the older one [RFC7159] used in the base ALTO protocol [RFC7285]. The newer JSON mandates UTF-8 text encoding to improve interoperability. Therefore, ALTO Clients and Servers implementations using UTF-{16,32} need to be cognizant of the subsequent interoperability risks and MUST switch to UTF-8 encoding, if they want to interoperate with Calendar-aware Servers and Clients.

9. Acknowledgements

The authors would like to thank Fred Baker, Li Geng, Diego Lopez, He Peng and Haibin Song for fruitful discussions and feedback on earlier draft versions. Dawn Chan, Kai Gao, Vijay Gurbani, Yichen Qian, Juergen Schoenwaelder, and Brian Weis and Jensen Zhang provided substantial review feedback and suggestions to the protocol design.

10. References

10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC7231] Fielding, R., Ed. and J. Reschke, Ed., "Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content", RFC 7231, DOI 10.17487/RFC7231, June 2014, <https://www.rfc-editor.org/info/rfc7231>.

[RFC7285] Alimi, R., Ed., Penno, R., Ed., Yang, Y., Ed., Kiesel, S., Previdi, S., Roome, W., Shalunov, S., and R. Woundy, "Application-Layer Traffic Optimization (ALTO) Protocol", RFC 7285, DOI 10.17487/RFC7285, September 2014, <https://www.rfc-editor.org/info/rfc7285>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8189] Randriamasy, S., Roome, W., and N. Schwan, "Multi-Cost Application-Layer Traffic Optimization (ALTO)", RFC 8189, DOI 10.17487/RFC8189, October 2017, <https://www.rfc-editor.org/info/rfc8189>.

[RFC8259] Bray, T., Ed., "The JavaScript Object Notation (JSON) Data Interchange Format", STD 90, RFC 8259, DOI 10.17487/RFC8259, December 2017, <https://www.rfc-editor.org/info/rfc8259>.

10.2. Informative References

[IEEE.754.2008] "Standard for Binary Floating-Point Arithmetic, IEEE Standard 754", August 2008.

[RFC2818] Rescorla, E., "HTTP Over TLS", RFC 2818, DOI 10.17487/RFC2818, May 2000, <https://www.rfc-editor.org/info/rfc2818>.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, DOI 10.17487/RFC5246, August 2008, <https://www.rfc-editor.org/info/rfc5246>.

[RFC5693] Seedorf, J. and E. Burger, "Application-Layer Traffic Optimization (ALTO) Problem Statement", RFC 5693, DOI 10.17487/RFC5693, October 2009, <https://www.rfc-editor.org/info/rfc5693>.

[RFC6708] Kiesel, S., Ed., Previdi, S., Stiemerling, M., Woundy, R., and Y. Yang, "Application-Layer Traffic Optimization (ALTO) Requirements", RFC 6708, DOI 10.17487/RFC6708, September 2012, <https://www.rfc-editor.org/info/rfc6708>.

[RFC7159] Bray, T., Ed., "The JavaScript Object Notation (JSON) Data Interchange Format", RFC 7159, DOI 10.17487/RFC7159, March 2014, <https://www.rfc-editor.org/info/rfc7159>.

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/info/rfc8446>.

[I-D.ietf-alto-incr-update-sse]
Roome, W. and Y. Yang, "ALTO Incremental Updates Using Server-Sent Events (SSE)", draft-ietf-alto-incr-update-sse-20 (work in progress), February 2020.

[I-D.ietf-alto-performance-metrics]
WU, Q., Yang, Y., Lee, Y., Dhody, D., and S. Randriamasy, "ALTO Performance Cost Metrics", draft-ietf-alto-performance-metrics-08 (work in progress), November 2019.

[I-D.xiang-alto-multidomain-analytics]
Xiang, Q., Le, F., Yang, Y., Newman, H., and H. Du, "Unicorn: Resource Orchestration for Multi-Domain, Geo-Distributed Data Analytics", draft-xiang-alto-multidomain-analytics-02 (work in progress), July 2018.

[SENSE-sdn-e2e-net]
"SDN for End-to-End Networked Science at the Exascale (SENSE), http://sense.es.net/overview".

[Braess-paradox]
Steinberg, R. and W. Zangwill, "The Prevalence of Braess' Paradox", Transportation Science Vol. 17 No. 3, August 1983.

[Selfish-routing-Roughgarden-thesis]
Roughgarden, T., "Selfish Routing", Dissertation Thesis, Cornell 2002, May 2002.

[Selfish-routing-Internet-eval]
Qiu, L., Yang, Y., Zhang, Y., and S. Shenker, "Selfish Routing in Internet-LIke Environments", Proceedings of ACM SIGCOMM 2001, August 2001.

Authors' Addresses

Sabine Randriamasy
Nokia Bell Labs
Route de Villejust
NOZAY 91460
FRANCE

Email: Sabine.Randriamasy@nokia-bell-labs.com

Richard Yang
Yale University
51 Prospect st
New Haven, CT 06520
USA

Email: yry@cs.yale.edu

Qin Wu
Huawei
101 Software Avenue, Yuhua District
Nanjing, Jiangsu 210012
China

Email: sunseawq@huawei.com

Lingli Deng
China Mobile
China

Email: denglingli@chinamobile.com

Nico Schwan
Thales Deutschland
Lorenzstrasse 10
Stuttgart 70435
Germany

Email: nico.schwan@thalesgroup.com