

PCE Working Group
Internet-Draft
Intended status: Informational
Expires: July 14, 2014

R. Casellas, Ed.
CTTC
C. Margaria
Coriant
A. Farrel
Old Dog Consulting
O. Gonzalez de Dios
Telefonica I+D
D. Dhody
X. Zhang
Huawei Technologies
January 10, 2014

**Current issues with existing RBNF notation for PCEP messages and
extensions
draft-cmfg-pce-pcep-grammar-02**

Abstract

The PCEP protocol has been defined in [[RFC5440](#)] and later extended in several RFCs. This document aims at documenting inconsistencies when implementing a set of extensions and at providing a reference, complete and formal RBNF grammar for PCEP messages, including object ordering and precedence rules.

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 <http://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 July 14, 2014.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

- [1.](#) Requirements Language [2](#)
- [2.](#) Introduction and Motivation [3](#)
 - [2.1.](#) Object Ordering [3](#)
 - [2.2.](#) Inconsistent Naming [5](#)
 - [2.3.](#) Semantics and Exclusive Rules [6](#)
- [3.](#) Initial Considerations [8](#)
- [4.](#) RBNF Grammars [8](#)
 - [4.1.](#) Common Constructs [8](#)
 - [4.1.1.](#) Object Sequences [8](#)
 - [4.1.2.](#) Synchronized Vectors [8](#)
 - [4.1.3.](#) Monitoring Metrics [9](#)
 - [4.1.4.](#) Monitoring Requests and Responses [9](#)
 - [4.2.](#) PCEP Open Message [10](#)
 - [4.3.](#) PCEP Keep Alive (KeepAlive) Message [10](#)
 - [4.4.](#) PCEP Request (PCReq) Message [10](#)
 - [4.5.](#) PCEP Reply (PCRep) Message [12](#)
 - [4.6.](#) PCEP Monitoring Request (PCMonReq) Message [13](#)
 - [4.7.](#) PCEP Monitoring Reply (PCMonRep) Message [13](#)
 - [4.8.](#) PCEP Notify (PCNtf) Message [14](#)
 - [4.9.](#) PCEP Error (PCErr) Message [14](#)
 - [4.10.](#) PCEP Report (PCRpt) Message [15](#)
 - [4.11.](#) PCEP Update (PCUpd) Message [15](#)
- [5.](#) Management Considerations [15](#)
- [6.](#) Contributing Authors [15](#)
- [7.](#) Acknowledgments [15](#)
- [8.](#) Normative References [15](#)
- Authors' Addresses [17](#)

1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [[RFC2119](#)].

2. Introduction and Motivation

The RBNF notation, defined in [RFC5511], is used to specify the message format for the Path Computation Element Communication Protocol (PCEP). The core of PCEP has been defined in [RFC5440] and later extended in [RFC5441], to support the Backward Recursive Path Computation (BRPC) procedure; in [RFC5455], adding a CLASSTYPE object to support Diffserv-aware Traffic Engineering (DS-TE); in [RFC5520], for topology confidentiality by means of Path keys; in [RFC5521], in support of exclusions; in [RFC5541] to convey specific objective functions; in [RFC5557], for Global Concurrent Optimization, in [RFC5886], for monitoring and in [RFC6006] for point-to-multipoint (P2MP) computation.

Most PCEP RFCs describe specific protocol extensions and, as such, they focus on their constructs extending some base RFCs. Although it is not the intention of each individual draft or RFC to provide the latest and most complete/full definition of the protocol messages, in practice combining all the extensions as defined in the respective RFCs is complex.

Message rules are sometimes provided within the text, resulting in ambiguity. Moreover, the fact that extensions may be defined in parallel may be a problem. The canonical example is the case where RFC X defines construct $p ::= A$ and subsequent RFC Y extends RFC X stating that object C MUST follow object A and RFC Z also extends RFC X stating that object D MUST follow object A.

2.1. Object Ordering

The use of RBNF [RFC5511] states that the ordering of objects and constructs in an assignment is explicit, and protocol specifications MAY opt to state that ordering is only RECOMMENDED (the elements of a list of objects and constructs MAY be received in any order).

The core PCEP document [RFC5440] states in [Section 6](#) that an implementation MUST form the PCEP messages using the object ordering specified in [RFC5440].

[RFC5886] equally states that "An implementation MUST form the PCEP messages using the object ordering specified in this document."

[RFC5521] only states that "the XRO is OPTIONAL and MAY be carried within Path Computation Request (PCReq) and Path Computation Reply (PCRep) messages." and no ordering is provided. It does not mention SVEC objects or rules.

[RFC5541] specifies that "the OF object MAY be carried within a PCReq message. If an objective function is to be applied to a set of synchronized path computation requests, the OF object MUST be carried just after the corresponding SVEC (Synchronization VECTOR) object and MUST NOT be repeated for each elementary request. Similarly, if a metric is to be applied to a set of synchronized requests, the METRIC object MUST follow the SVEC object and MUST NOT be repeated for each elementary request. (...) An OF object specifying an objective function that applies to an individual path computation request (non-synchronized case) MUST follow the RP object for which it applies". It should be understood that this last sentence must be relaxed or is in contradiction with the ENDPOINTS object.

RFCs that extend the core PCEP protocol are not consistent with the object ordering. For example, [RFC5520] defines:

```
<segment-computation> ::=
  <END-POINTS>
  [<LSPA>]
  [<BANDWIDTH>]
  [<BANDWIDTH>]
  [<metric-list>]
  (snip)
```

and states that "the format of the message for use in normal path computation is unmodified". However, [RFC5520] was not updated to reflect that the the BANDWIDTH object used for reoptimization was moved to appear after the RRO for which it applies, as given in [RFC5440] (updated in Errata ID: 3582):

```
<request> ::= <RP>
  <END-POINTS>
  [<LSPA>]
  [<BANDWIDTH>]
  [<metric-list>]
  [<RRO>[<BANDWIDTH>]]
  [<IRO>]
  [<LOAD-BALANCING>]
```

[RFC5541] in [section 3.2](#) is not consistent with the ordering of OF and metric-list:


```
<svec-list> ::= <SVEC>
              [<OF>]
              [<metric-list>]
```

```
<request> ::= <RP>
              (snip)
              [<metric-list>]
              [<OF>]
```

```
<attribute-list> ::= [<OF>]
                    [<LSPA>]
                    [<BANDWIDTH>]
                    [<metric-list>]
```

In view of the above considerations, this document aims at providing an object ordering for PCEP messages so implementations can interoperate. Implementations conforming to this document MUST use the object ordering specified here.

2.2. Inconsistent Naming

PCEP RFCs may use inconsistent or ambiguous naming. For example [[RFC5440](#)] defines the Open message as having a common header and an OPEN object, and later uses Open to refer to the object that may appear in a PCErr message.

```
<Open Message> ::= <Common Header>
                  <OPEN>
```

```
<PCErr Message> ::= <Common Header>
                   (<error-obj-list> [<Open>]) | <error>
                   [<error-list>]
```

It is common that a sequence or repetition of an object OBJ is noted as obj-list. It may happen that in extensions to core documents, the naming is kept although it no longer applies to such a sequence. For example, [[RFC5886](#)] states:

```
<svec-list> ::= <SVEC>
               [<OF>]
               [<svec-list>]
```

and later

```
<svec-list> ::= <SVEC>
               [<svec-list>]
```


2.3. Semantics and Exclusive Rules

The current RBNF notation does not capture the semantics/intent of the messages; notably, when two options are mutually exclusive and at least one is mandatory. In most cases, this is noted as both options being optional. For example [[RFC5440](#)] states:

```
<response> ::= <RP>
                [<NO-PATH>]
                [<attribute-list>]
                [<path-list>]
```

with this example, a message that contains a response of the form <RP><NO-PATH><ERO><..> (that is, a NO-PATH object followed by a path) is correct and successfully parsed. Likewise, a response with just an RP object is valid. Although the actual text within the RFC may state the intention and disambiguate the grammar, having a RBNF notation that better captures semantics, message structure and original intent, enables the development of automated parsers that closely map the specification.

Similarly, if the intent is to specific a rule such as metric-pce which includes a PCE-ID object followed by a PROC-TIME object and/or an OVERLOAD object, the syntax:

```
<metric-pce> ::= <PCE-ID> [<PROC-TIME>] [<OVERLOAD>]
```

allows, amongst other combinations, that neither PROC-TIME nor OVERLOAD appears, which is not the intended behavior (there should be at least one metric). The alternative

```
<metric-pce> ::= <PCE-ID> <metric-argument-list>
<metric-argument-list> ::= <metric-argument> [<metric-argument-list>]
<metric-argument> ::= <PROC-TIME> | <OVERLOAD>
```

or equivalently

```
<metric-pce> ::= <PCE-ID> (<metric-argument>...)
<metric-argument> ::= <PROC-TIME> | <OVERLOAD>
```

does not reflect that each metric-argument should appear at most once. This can be addressed verbosely:


```
<metric-pce> ::= <PCE-ID>
                ( <PROC-TIME> | <OVERLOAD> | <PROC-TIME><OVERLOAD> )
```

```
<metric-pce> ::= <PCE-ID>
                ( <PROC-TIME>[<OVERLOAD>] | [<PROC-TIME>]<OVERLOAD> )
```

Here the semantic is that we require any object of the set {PROC-TIME, OVERLOAD} to be present, and there should be at least one. Note that currently there are only a few cases where the "non-empty set" case arises.

[Editor note/AF To make a normative or machine-readable definition, new notation could be defined:

- non-empty set, repetition not allowed


```
<set> ::= { <a> | <b> | <c> }
```
- non-empty set, repetition allowed


```
<set> ::= { <a> <b> <c> }
```
- also can be expressed using the previous definition with


```
<set> ::= { <a>... | <b>... | <c>... }
```

Note that the other options can already be handled

- non-repetition set allowed to be empty


```
<set> ::= [<a>] [<b>] [<c>]
```
- repetition set allowed to be empty


```
<set> ::= [<a>] [<b>] [<c>] [<set>]
```

The notation with "{" would be convenient to express implicit ordering (<a><a> ok but <a><a> not)].

A more condensed notation extension to the RBNF notation could also use a "sequential or" notation:

```
<a> || <b> is defined as <a> | <b> | <a><b>
```

```
<a> || <b> || <c> is defined as (assoc.)
```

```
(<a> | <b> | <a> <b>) | <c> | (<a> | <b> | <a> <b> ) <c> =
(<a> | <b> | <a> <b>) | <c> | (<a><c> | <b><c> | <a><b><c>) =
<a> | <b> | <c> | <a> <b> | <a> <c> | <b> <c> | <a> <b> <c>
```

The use of sequential-or notation allows writing:

```
<metric-pce> ::= <PCE-ID> ( <PROC-TIME> || <OVERLOAD> )
```


The goal of this document is then, first, to provide an (almost) formal (reasonably) complete definition of PCEP messages, checking the overall protocol and extensions consistency, defining an object ordering; and to set the basis for implementation agreements that aim at integrating published PCEP extensions. It is also a goal to provide alternative (although compatible) RBNF notations to be expressive enough to avoid invalid cases.

3. Initial Considerations

This document does not modify the content of defined PCEP objects and TLVs.

This document is not normative, the normative definition is included in the existing specs. This does not preclude integration with a future revision of such documents.

4. RBNF Grammars

This section provides the proposed RBNF notation for the PCEP messages. Specific constructs or grammar rules that appear in several messages or deserve special considerations are described first.

4.1. Common Constructs

4.1.1. Object Sequences

```
<of-list>           ::= <OF> [<of-list>]
<metric-list>      ::= <METRIC> [<metric-list>]
<vendor-info-list> ::= <VENDOR-INFORMATION> [<vendor-info-list>]
<pce-id-list>       ::= <PCE-ID> [<pce-id-list>]
    -- (note: named pce-list in original)
```

4.1.2. Synchronized Vectors

SVEC tuple:

A svec-tuple is a construct that associates a SVEC object with one or more constraining objects. The selected order follows the relative order of having OF and metric-list after the SVEC object, and the name svec-list has been changed since it no longer means a list of SVEC objects.


```
<svec-tuple> ::= <SVEC>
               [<OF>]
               [<metric-list>]
               [<vendor-info-list>]
               [<GC>]
               [<XRO>]
```

```
<svec-tuple-list> ::= <svec-tuple> [<svec-tuple-list>]
```

Note that [[I-D.ietf-pce-vendor-constraints](#)] defines:

```
<svec-list> ::= <SVEC>
              [<OF>]
              [<GC>]
              [<XRO>]
              [<metric-list>]
              [<vendor-info-list>]
              [<svec-list>]
```

The construct is updated to reflect the new name and to have the same relative order in the attributes that constrain a individual request

[4.1.3.](#) Monitoring Metrics

A metric-pce-id is a rule that associates a PCE identified by its PCE-ID to a list of metric arguments.

```
<metric-pce-id> ::= <PCE-ID>
                  ( <PROC-TIME> [<OVERLOAD>] |
                    [<PROC-TIME>] <OVERLOAD> )
```

```
<metric-pce-id-list> ::= <metric-pce-id> [<metric-pce-id-list>]
```

[4.1.4.](#) Monitoring Requests and Responses

See [[RFC5886](#)] for the definition of specific/general and in-band/out-of-band.


```

<monitoring> ::= <MONITORING> <PCC-ID-REQ>

<monitoring-request> ::= <monitoring> [<pce-id-list>]

<monitoring-response> ::= <monitoring>
    (<specific-monitoring-metrics-list> |
     <general-monitoring-metrics-list>)

<specific-monitoring-metrics-list> ::=
    <specific-monitoring-metrics>
    [<specific-monitoring-metrics-list>]

<general-monitoring-metrics-list> ::=
    <general-monitoring-metrics>
    [<general-monitoring-metrics-list>]

<specific-monitoring-metrics> ::=
    <RP> <monitoring-metrics>

<general-monitoring-metrics> ::=
    <monitoring-metrics>

<monitoring-metrics> ::=
    <metric-pce-id-list>

```

[4.2.](#) PCEP Open Message

```

<Open Message> ::= <Common Header>
    <OPEN>

```

[4.3.](#) PCEP Keep Alive (KeepAlive) Message

```

<KeepAlive Message> ::= <Common Header>

```

[4.4.](#) PCEP Request (PCReq) Message

Note that the actual parsing depends on the content (flags) of the Request Parameters (RP) object, notably expansion and P2MP. In some cases, this may be considered redundant, e.g. the presence of a PATH_KEY object and the corresponding flag.

[Editor's note: rom a notation perspective, we lack a way to express "if object a field x has value v then include object b, else include object c". A possible way would be to define new intermediate types
:

<a with x=v> and <a with x!=v> then

(<a with x=v>) | (<a with x!=v> <c>)

this issue is stil open.]

The PCReq message contains a possibly monitored list of requests, some of which may be grouped by SVEC tuples.

```
<PCReq Message> ::= <Common Header>
                    [<monitoring-request>]
                    [<svec-tuple-list>]
                    <request-list>
```

where:

```
<request-list>    ::= <request> [<request-list>]
```

-- A request is either an expansion, a P2P request or a P2MP request

```
<request>         ::= <expansion> |
                    <p2p_computation> |
                    <p2mp_computation>
```

```
<expansion>      ::= <RP><PATH-KEY>
```

```
<p2p_computation> ::= <RP><ENDPOINTS>
                    [<LSP>][<gen-bw>][<p2p-attributes>...]
```

```
<p2mp_computation> ::= <RP><tree-list>
                    [<p2mp-attributes>...]
```

-- For a P2P computation

```
<p2p-attributes> ::= <attributes>|<rro-bw-pair>
```

```
<attributes>    ::= <attribute> [<attributes>]
```

```
<attribute>     ::=
    <CLASSTYPE> |
    <LSPA> |
    <OF> |
    <metric-list> |
    <vendor-info-list> |
    <IRO> |
    <BNC> |
```



```
<XRO> |
<gen-load-balancing> |
<INTER-LAYER> |
<SWITCH-LAYER> |
<REQ-ADAP-CAP>

-- in RFC6006 there is a bw per tree,
-- it is intended to be an optimization for an RRO list

<rro-bw-pair> ::= <RRO> [<gen-bw>]

<rro-list-bw> ::= (<RRO>...)[<gen-bw>]

<tree> ::= <ENDPOINTS>(<rro-bw-pair>|<rro-list-bw>)

<gen-bw> ::= <BANDWIDTH>[<GENERALIZED-BANDWIDTH>...]
-- per RFC5440 section 7.7

<gen-load-balancing> ::= <LOAD-BALANCING> |
                        <GENERALIZED-LOAD-BALANCING>

-- For P2MP computations - note some atts (BNC) are only P2MP

<tree-list> ::= <tree> [<tree-list>]

<tree> ::= <ENDPOINTS> <rro_bw_pair>

<p2mp-attributes> ::= (<attribute> | <BNC>) [<p2mp-attributes>]
```

[4.5.](#) PCEP Reply (PCRep) Message


```
<PCRep Message> ::= <Common Header>
                    [<svec-tuple-list>]
                    <response-list>
```

-- Note: should clarify the use of SVEC tuple list

where

```
<response-list> ::= <response> [<response-list>]
```

-- An individual response may include monitoring info

```
<response> ::= <RP> [<monitoring>]
              (<success> | <failure>) [<monitoring-metrics>]
```

-- Note: should clarify P2MP attributes

```
<success> ::= <path-list>
```

```
<failure> ::= <NO-PATH> [<attributes>]
```

```
<path-list> ::= <path> [<path-list>]
```

```
<path> ::= <ERO> <gen-bw> [<attributes>]
```

4.6. PCEP Monitoring Request (PCMonReq) Message

The PCMonReq message is defined in [[RFC5886](#)] for out-of-band monitoring requests.

[RFC5886] specifies that there is one mandatory object but the grammar also includes PCC-ID-REQ as mandatory.

[Ed note:does it make sense to include a pce-id-list and a svec-list/request-list at the same time?]

```
<PCMonReq Message> ::= <Common Header>
                       <monitoring-request>
                       [[<svec-tuple-list>] <request-list>]
```

4.7. PCEP Monitoring Reply (PCMonRep) Message

The PCMonRep message is defined in [[RFC5886](#)] for out-of-band monitoring responses.

[RFC5886] specifies that there is one mandatory object but the grammar also includes PCC-ID-REQ as mandatory.

[RFC5886] does not allow bundling several specific monitoring responses. A PCMonReq message causes N PCMonRep messages.

```
<PCMonRep Message> ::= <Common Header>
                        <monitoring-response>
```

4.8. PCEP Notify (PCNtf) Message

```
<PCNtf Message> ::= <Common Header>
                    ( <solicited-notify> | <unsolicited-notify> )
```

where

```
<solicited-notify> ::= <request-id-list> <notification-list>
```

```
<unsolicited-notify> ::= <notification-list>
```

```
<request-id-list> ::= <RP> [<request-id-list>]
```

```
<notification-list> ::= <NOTIFICATION> [<notification-list>]
```

4.9. PCEP Error (PCErr) Message

Errors can occur during PCEP handshake, or bound to one or more requests.

An error during handshake is never solicited, i.e., not associated to a list of requests.

A solicited error binds one or more Requests (RPs) to one or more PCEP-ERROR objects.


```
<PCErr Message> ::= <Common Header>
    ( <solicited-error> | <unsolicited-error> )
```

where

```
<solicited-error> ::= <request-id-list> <pcep-error-list>
<unsolicited-error> ::= <handshake-error> | <pcep-error-list>
<handshake-error> ::= <pcep-error-list> <OPEN>
<request-id-list> ::= <RP> [<request-id-list>]
<pcep-error-list> ::= <PCEP-ERROR> [<pcep-error-list>]
```

[4.10.](#) PCEP Report (PCRpt) Message

TBD see [[I-D.ietf-pce-stateful-pce](#)].

[4.11.](#) PCEP Update (PCUpd) Message

TBD see [[I-D.ietf-pce-stateful-pce](#)].

[5.](#) Management Considerations

TBD

[6.](#) Contributing Authors

Robert Varga
Pantheon
robert.varga@pantheon.sk

[7.](#) Acknowledgments

This work was supported in part by the PACE Support Action (<http://ict-pace.net/>) project under grant agreement number 619712.

[8.](#) Normative References

[[I-D.ietf-pce-stateful-pce](#)]
Crabbe, E., Medved, J., Minei, I., and R. Varga, "PCEP Extensions for Stateful PCE", [draft-ietf-pce-stateful-pce-07](#) (work in progress), October 2013.

[I-D.ietf-pce-vendor-constraints]

Zhang, F. and A. Farrel, "Conveying Vendor-Specific Constraints in the Path Computation Element Protocol", [draft-ietf-pce-vendor-constraints-10](#) (work in progress), April 2013.

[I-D.ietf-pce-wson-rwa-ext]

Lee, Y. and R. Casellas, "PCEP Extension for WSON Routing and Wavelength Assignment", [draft-ietf-pce-wson-rwa-ext-00](#) (work in progress), April 2013.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.

[RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", [RFC 4655](#), August 2006.

[RFC5440] Vasseur, JP. and JL. Le Roux, "Path Computation Element (PCE) Communication Protocol (PCEP)", [RFC 5440](#), March 2009.

[RFC5441] Vasseur, JP., Zhang, R., Bitar, N., and JL. Le Roux, "A Backward-Recursive PCE-Based Computation (BRPC) Procedure to Compute Shortest Constrained Inter-Domain Traffic Engineering Label Switched Paths", [RFC 5441](#), April 2009.

[RFC5455] Sivabalan, S., Parker, J., Boutros, S., and K. Kumaki, "Diffserv-Aware Class-Type Object for the Path Computation Element Communication Protocol", [RFC 5455](#), March 2009.

[RFC5511] Farrel, A., "Routing Backus-Naur Form (RBNF): A Syntax Used to Form Encoding Rules in Various Routing Protocol Specifications", [RFC 5511](#), April 2009.

[RFC5520] Bradford, R., Vasseur, JP., and A. Farrel, "Preserving Topology Confidentiality in Inter-Domain Path Computation Using a Path-Key-Based Mechanism", [RFC 5520](#), April 2009.

[RFC5521] Oki, E., Takeda, T., and A. Farrel, "Extensions to the Path Computation Element Communication Protocol (PCEP) for Route Exclusions", [RFC 5521](#), April 2009.

[RFC5541] Le Roux, JL., Vasseur, JP., and Y. Lee, "Encoding of Objective Functions in the Path Computation Element Communication Protocol (PCEP)", [RFC 5541](#), June 2009.

- [RFC5557] Lee, Y., Le Roux, JL., King, D., and E. Oki, "Path Computation Element Communication Protocol (PCEP) Requirements and Protocol Extensions in Support of Global Concurrent Optimization", [RFC 5557](#), July 2009.
- [RFC5886] Vasseur, JP., Le Roux, JL., and Y. Ikejiri, "A Set of Monitoring Tools for Path Computation Element (PCE)-Based Architecture", [RFC 5886](#), June 2010.
- [RFC5886] Vasseur, JP., Le Roux, JL., and Y. Ikejiri, "A Set of Monitoring Tools for Path Computation Element (PCE)-Based Architecture", [RFC 5886](#), June 2010.
- [RFC6006] Zhao, Q., King, D., Verhaeghe, F., Takeda, T., Ali, Z., and J. Meuric, "Extensions to the Path Computation Element Communication Protocol (PCEP) for Point-to-Multipoint Traffic Engineering Label Switched Paths", [RFC 6006](#), September 2010.

Authors' Addresses

Ramon Casellas (editor)
CTTC
Av. Carl Friedrich Gauss n.7
Castelldefels 08860 Barcelona
Spain

Phone: +34 93 645 29 00
Email: ramon.casellas@cttc.es

Cyril Margaria
Coriant
St.-Martin-Str. 76
Muenchen 81541
Germany

Phone: +49 89 5159 16934
Email: cyril.margaria@coriant.com

Adrian Farrel
Old Dog Consulting

Email: adrian@olddog.co.uk

Oscar Gonzalez de Dios
Telefonica I+D
Don Ramon de la Cruz 82-84
Madrid 28045
Spain

Phone: +34913128832
Email: ogondio@tid.es

Dhruv Dhody
Huawei Technologies
Leela Palace
Bangalore, Karnataka 560008
INDIA

Email: dhruv.dhody@huawei.com

Xian Zhang
Huawei Technologies

Email: zhang.xian@huawei.com

