Internet Engineering Task Force (IETF)
Request for Comments: 6807
Category: Experimental
ISSN: 2070-1721

D. Farinacci
G. Shepherd
S. Venaas
Cisco Systems
Y. Cai
Microsoft

December 2012

Population Count Extensions to Protocol Independent Multicast (PIM)

Abstract

This specification defines a method for providing multicast distribution-tree accounting data. Simple extensions to the Protocol Independent Multicast (PIM) protocol allow a rough approximation of tree-based data in a scalable fashion.

Status of This Memo

This document is not an Internet Standards Track specification; it is published for examination, experimental implementation, and evaluation.

This document defines an Experimental Protocol for the Internet community. This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Not all documents approved by the IESG are a candidate for any level of Internet Standard; see Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc6807.

Table of Contents

1. Introduction

This document specifies a mechanism to convey accounting information using the Protocol Independent Multicast (PIM) protocol [RFC4601] [RFC5015]. Putting the mechanism in PIM allows efficient distribution and maintenance of such accounting information. Previous mechanisms require data to be correlated from multiple router sources.

This mechanism allows a single router to be queried to obtain accounting and statistic information for a multicast distribution tree as a whole or any distribution sub-tree downstream from a queried router. The amount of information is fixed and does not increase as multicast membership, tree diameter, or branching increases.

The sort of accounting data this specification provides, on a per-multicast-route basis, are:

1. The number of branches in a distribution tree.

2. The membership type of the distribution tree, that is, Source-Specific Multicast (SSM) or Any-Source Multicast (ASM).

3. Routing domain and time zone boundary information.

4. On-tree node and tree diameter counters.

5. Effective MTU and bandwidth.

This document defines a new PIM Join Attribute type [RFC5384] for the Join/Prune message as well as a new Hello option. The mechanism is applicable to IPv4 and IPv6 multicast.

This is a new extension to PIM, and it is not completely understood what impact collecting information using PIM would have on the operation of PIM. This is an entirely new concept. Many PIM features (including the core protocols) were first introduced in Experimental RFCs, and it seems appropriate to advance this work as Experimental. Reports of implementation and deployment across whole distribution trees or within sub-trees (see Section 6) will enable an assessment of the desirability and stability of this specification. The PIM Working Group will then consider whether to move this work to the Standards Track.

This document does not specify how an administrator or user can access this information. It is expected that an implementation may have a command-line interface or other ways of requesting and

displaying this information. As this is currently an Experimental document, defining a MIB module has not been considered. If the PIM Working Group finds that this should move on to Standards Track, a MIB module should be considered.

1.1. Requirements Notation

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].

1.2. Terminology

This section defines the terms used in this document.

Multicast Route: An (S,G) or (*,G) entry regardless of whether the route is in ASM, SSM, or BIDIR mode of operation.

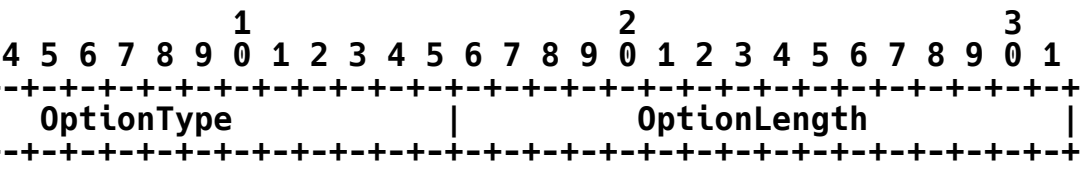
Stub Link: A link with members joined to the group via IGMP or Multicast Listener Discovery (MLD).

Transit Link: A link put in the oif-list (outgoing interface list) for a multicast route because it was joined by PIM routers.

Note that a link can be both a Stub Link and a Transit Link at the same time.

2. Pop-Count-Supported Hello Option

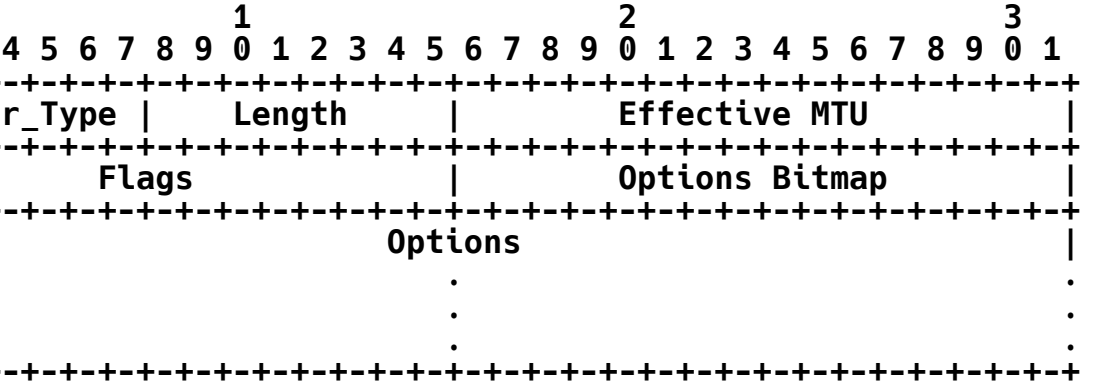
A PIM router indicates that it supports the mechanism specified in this document by including the Pop-Count-Supported Hello option in its PIM Hello message. Note that it also needs to include the Join-Attribute Hello option as specified in [RFC5384]. The format of the Pop-Count-Supported Hello option is defined to be:



OptionType = 29, OptionLength = 0. Note that there is no option value included. In order to allow future updates of this specification that may include an option value, implementations of this document MUST accept and process this option even if the length is non-zero. Implementations of this specification MUST accept and process the option ignoring any option value that may be included.

3. New Pop-Count Join Attribute Format

When a PIM router supports this mechanism and has determined from a received Hello that the neighbor supports this mechanism, and also that all the neighbors on the interface support the use of join attributes, it will send Join/Prune messages that MAY include a Pop-Count Join Attribute. The mechanism to process a PIM Join Attribute is described in [RFC5384]. The format of the new attribute is specified in the following.



The above format is used only for entries in the join-list section of the Join/Prune message.

F bit: 0 (Non-Transitive Attribute).

E bit: As specified by [RFC5384].

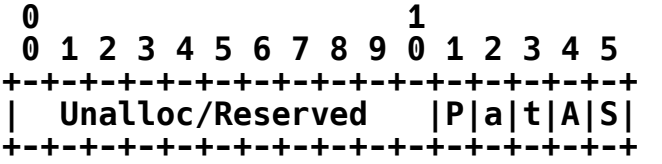
Attr_Type: 3.

Length: The minimum length is 6.

Effective MTU: This contains the minimum MTU for any link in the oif-list. The sender of a Join/Prune message takes the minimum value for the MTU (in bytes) from each link in the oif-list. If this value is less than the value stored for the multicast route (the one received from downstream joiners), then the value should be reset and sent in a Join/Prune message. Otherwise, the value should remain unchanged.

This provides the MTU supported by multicast distribution tree when examined at the first-hop router(s) or for sub-tree for any router on the distribution tree.

Flags: The flags field has the following format:



Unallocated/Reserved Flags: The flags that are currently not defined. If a new flag is defined and used by a new implementation, an old implementation should preserve the bit settings. This means that a router MUST preserve the settings of all Unallocated/Reserved Flags in PIM Join messages received from downstream routers in any PIM Join sent upstream.

S flag: This flag is set if an IGMPv3 or MLDv2 report with an INCLUDE mode group record was received on any oif-list entry or the bit was set from any PIM Join message. This bit should only be cleared when the above becomes untrue.

A flag: This flag is set if an IGMPv3 or MLDv2 report with an EXCLUDE mode group record, or an IGMPv1, IGMPv2, or MLDv1 report, was received on any oif-list entry or the bit was set from any PIM Join message. This bit should only be cleared when the above becomes untrue.

A combination of settings for these bits indicate:

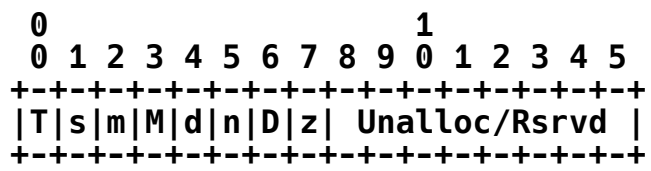
A flag	S flag	Description
0	0	There are no members for the group. ('Stub Oif-List Count' is 0)
0	1	All group members are using SSM.
1	0	All group members are using ASM.
1	1	A mixture of SSM and ASM group members.

t flag: This flag is set if there are any manually configured tunnels on the distribution tree. This means any tunnel that is not an auto-tunnel. If a manually configured tunnel is in the oif-list, a router sets this bit in its Join/Prune messages. Otherwise, it propagates the bit setting from downstream joiners.

a flag: This flag is set if there are any auto-tunnels on the distribution tree. If an auto-tunnel is in the oif-list, a router sets this bit in its Join/Prune messages. Otherwise, it propagates the bit setting from downstream joiners. An example of an auto-tunnel is a tunnel set up by the Automatic Multicast Tunneling [AMT] protocol.

P flag: This flag is set by a router if all downstream routers support this specification. That is, they are all PIM Pop-Count capable. If a downstream router does not support this specification, it MUST be cleared. This allows one to tell if the entire sub-tree is completely accounting capable.

Options Bitmap: This is a bitmap that shows which options are present. The format of the bitmap is as follows:



Each one of the bits T, s, m, M, d, n, D and z is associated with one option, where the option is included if and only if the respective bit is set. Included options MUST be in the same order as these bits are listed. The bits denote the following options:

bit	Option
T	Transit Oif-List Count
s	Stub Oif-List Count
m	Minimum Speed Link
M	Maximum Speed Link
d	Domain Count
n	Node Count
D	Diameter Count
z	TZ Count

See Section 3.1 for details on the different options. The unallocated bits are reserved. Any unknown bits MUST be set to 0 when a message is sent, and treated as 0 (ignored) when received. This means that unknown options that are denoted by unknown bits are ignored.

By using this bitmap we can specify at most 16 options. If there becomes a need for more than 16 options, one can define a new option that contains a bitmap that can then be used to specify which further options are present. The last bit in the current bitmap could be used for that option. However, the exact definition of this is left for future documents.

Options: This field contains options. Which options are present is determined by the flag bits. As new flags and options may be defined in the future, any unknown/reserved flags MUST be ignored, and any additional trailing options MUST be ignored. See Section 3.1 for details on the options defined in this document.

3.1. Options

There are several options defined in this document. For each option, there is also a related flag that shows whether the option is present. See the Options Bitmap above for a list of the options and their respective bits. Each option has a fixed size. Note that there are no alignment requirements for the options, so an implementation cannot assume they are aligned.

Transit Oif-List Count: This is filled in by a router sending a Join/Prune message indicating the number of transit links on the multicast distribution tree. The value is the number of oifs (outgoing interfaces) for the multicast route that have been joined by PIM plus the sum of the values advertised by each of the downstream PIM routers that have joined on this oif. Length is 4 octets.

Stub Oif-List Count: This is filled in by a router sending a Join/Prune message indicating the number of stub links (links where there are host members) on the multicast distribution tree. The value is the number of oifs for the multicast route that have been joined by IGMP or MLD plus the sum of the values advertised by each of the downstream PIM routers that have joined on this oif. Length is 4 octets.

Minimum Speed Link: This contains the minimum bandwidth rate for any link in the oif-list and is encoded as specified in Section 3.1.1. The sender of a Join/Prune message takes the minimum value for each link in the oif-list for the multicast route. If this value is less than the value stored for the multicast route (the smallest value received from downstream joiners), then the value should be reset and sent in a Join/Prune message. Otherwise, the value should remain unchanged. This, together with the Maximum

Speed Link option, provides a way to obtain the lowest- and highest-speed links for the multicast distribution tree. Length is 2 octets.

Maximum Speed Link: This contains the maximum bandwidth rate for any link in the oif-list and is encoded as specified in Section 3.1.1. The sender of a Join/Prune message takes the maximum value for each link in the oif-list for the multicast route. If this value is greater than the value stored for the multicast route (the largest value received from downstream joiners), then the value should be reset and sent in a Join/Prune message. Otherwise, the value should remain unchanged. This, together with the Minimum Speed Link option, provides a way to obtain the lowest- and highest-speed links for the multicast distribution tree. Length is 2 octets.

Domain Count: This indicates the number of routing domains the distribution tree traverses. A router should increment this value if it is sending a Join/Prune message over a link that traverses a domain boundary. For this to work, an implementation needs a way of knowing that a neighbor or an interface is in a different domain. There is no standard way of doing this. Length is 1 octet.

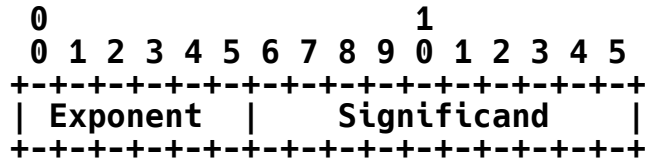
Node Count: This indicates the number of routers on the distribution tree. Each router will sum up all the Node Counts from all joiners on all oifs and increment by 1 before including this value in the Join/Prune message. Length is 1 octet.

Diameter Count: This indicates the longest length of any given branch of the tree in router hops. Each router that sends a Join increments the max value received by all downstream joiners by 1. Length is 1 octet.

TZ Count: This indicates the number of time zones the distribution tree traverses. A router should increment this value if it is sending a Join/Prune message over a link that traverses a time zone. This can be a configured link attribute, or using other means to determine the time zone is acceptable. Length is 1 octet.

3.1.1. Link Speed Encoding

The speed is encoded using 2 octets as follows:



Using this format, the speed of the link is Significand * 10 ^ Exponent kbps. This allows specifying link speeds with up to 3 decimal digits precision and speeds from 1 kbps to 10 ^ 67 kbps. A computed speed of 0 kbps means the link speed is < 1 kbps.

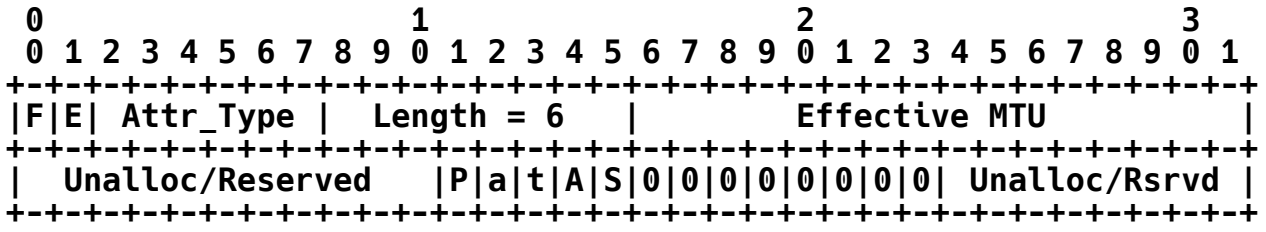
Here are some examples of how this is used:

Link Speed	Exponent	Significand
500 kbps	0	500
500 kbps	2	5
155 Mbps	3	155
40 Gpbs	6	40
100 Gpbs	6	100
100 Gpbs	8	1

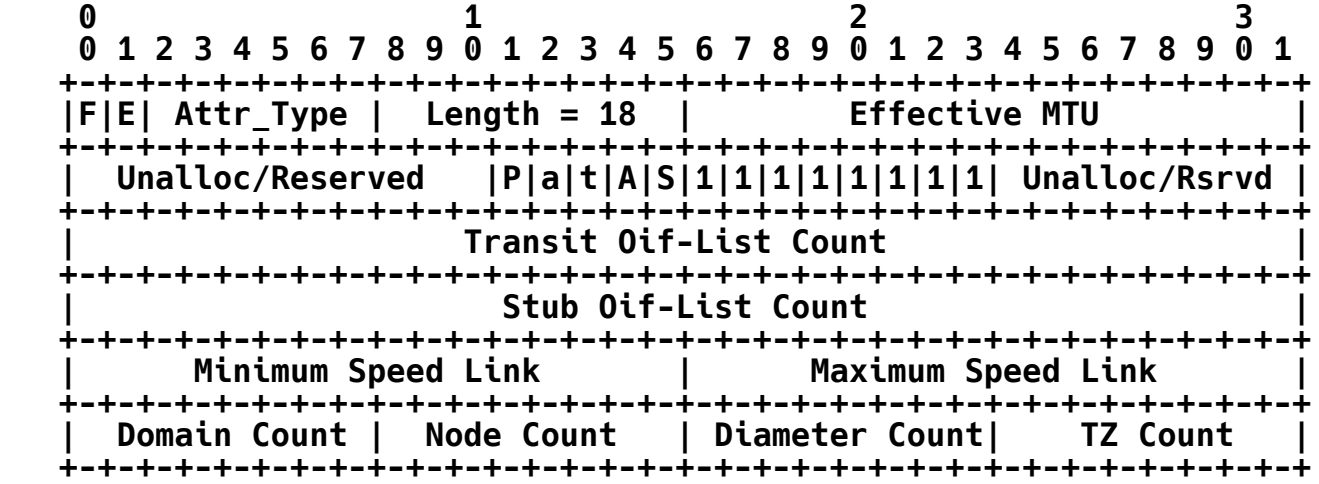
3.2. Example Message Layouts

Here, we will give a few examples to illustrate the use of flags and options.

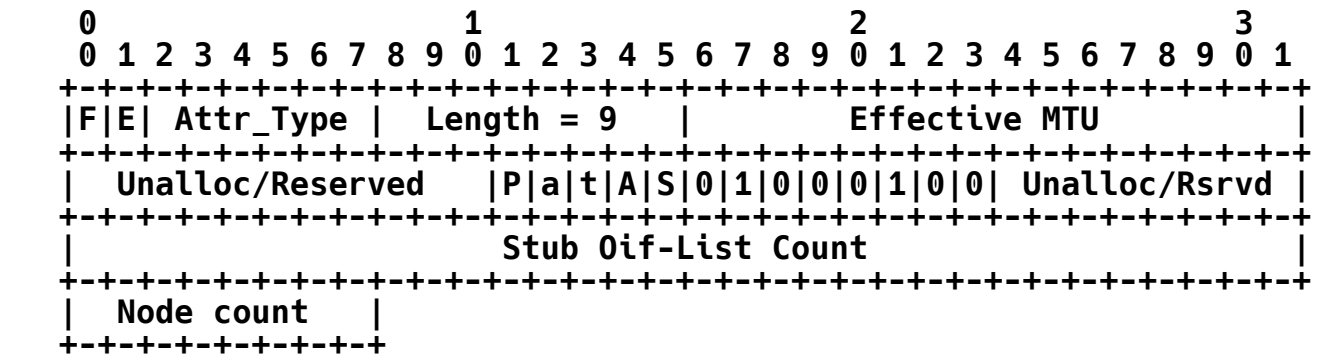
A minimum-size message has no option flags set and looks like this:



A message containing all the options defined in this document would look like this:



A message containing only Stub Oif-List Count and Node Count would look like this:



4. How to Use Pop-Count Encoding

A router supporting this mechanism MUST, unless administratively disabled, include the PIM Join Attribute option in its PIM Hellos. See [RFC5384] and "PIM-Hello Options" on [PIM-REG] for details.

It is RECOMMENDED that implementations allow for administrative control of whether to make use of this mechanism. Implementations MAY also allow further control of what information to store and send upstream.

It is very important to note that any changes to the values maintained by this mechanism MUST NOT trigger a new Join/Prune message. Due to the periodic nature of PIM, the values can be accurately obtained at 1-minute intervals (or whatever Join/Prune interval used).

When a router removes a link from an oif-list, it needs to be able to reevaluate the values that it will advertise upstream. This happens when an oif-list entry is timed out or a Prune is received.

It is RECOMMENDED that the Join Attribute defined in this document be used only for entries in the join-list part of the Join/Prune message. If the attribute is used in the prune-list, an implementation MUST ignore it and process the Prune as if the attribute were not present.

It is also RECOMMENDED that join suppression be disabled on a LAN when Pop-Count is used.

It is RECOMMENDED that, when triggered Join/Prune messages are sent by a downstream router, the accounting information not be included in the message. This way, when convergence is important, avoiding the processing time to build an accounting record in a downstream router and processing time to parse the message in the upstream router will help reduce convergence time. If an upstream router receives a Join/Prune message with no accounting data, it SHOULD NOT interpret the message as a trigger to clear or reset the accounting data it has cached.

5. Implementation Approaches

This section offers some non-normative suggestions for how Pop-Count may be implemented.

An implementation can decide how the accounting attributes are maintained. The values can be stored as part of the multicast route data structure by combining the local information it has with the joined information on a per-oif basis. So, when it is time to send a Join/Prune message, the values stored in the multicast route can be copied to the message.

Or, an implementation could store the accounting values per oif and, when a Join/Prune message is sent, it can combine the oifs with its local information. Then, the combined information can be copied to the message.

When a downstream joiner stops joining, accounting values cached must be evaluated. There are two approaches that can be taken. One is to keep values learned from each joiner, so when the joiner goes away, the count/max/min values are known and the combined value can be adjusted. The other approach is to set the value to 0 for the oif, and then start accumulating new values as subsequent Joins are received.

The same issue arises when an oif is removed from the oif-list. Keeping per-oif values allows you to adjust the per-route values when an oif goes away. Or, alternatively, a delay for reporting the new set a values from the route can occur while all oif values are zeroed (where accumulation of new values from subsequent Joins cause repopulation of values and a new max/min/count can be reevaluated for the route).

6. Caveats

This specification requires each router on a multicast distribution tree to support this specification or else the accounting attributes for the tree will not be known.

However, if there is a contiguous set of routers downstream in the distribution tree, they can maintain accounting information for the sub-tree.

If there is a set of contiguous routers supporting this specification upstream on the multicast distribution tree, accounting information will be available, but it will not represent an accurate assessment of the entire tree. Also, it will not be clear how much of the distribution tree the accounting information covers.

7. IANA Considerations

A new PIM-Hello Option type, 29, has been assigned by IANA. Although the length is specified as 0 in this specification, non-zero length is allowed, so IANA has listed the length as being variable.

A new PIM Join Attribute type, 3, has been assigned by IANA.

8. Security Considerations

The use of this specification requires some additional processing of PIM Join/Prune messages. However, the additional amount of processing is fairly limited, so this is not believed to be a significant concern.

The use of this mechanism includes information like the number of receivers. This information is assumed to not be of a sensitive nature. If an operator has concerns about revealing this information to upstream routers or other routers/hosts that may potentially inspect this information, there should be a way to disable the mechanism or, alternatively, more detailed control of what information to include.

9. Acknowledgments

The authors would like to thank John Zwiebel, Amit Jain, and Clayton Wagar for their review comments on the initial versions of this document. Adrian Farrel did a detailed review of the document and proposed textual changes that have been incorporated. Further review and comments were provided by Thomas Morin and Zhaohui (Jeffrey) Zhang.

10. References

10.1. Normative References

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

[RFC4601] Fenner, B., Handley, M., Holbrook, H., and I. Kouvelas, "Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)", RFC 4601, August 2006.

[RFC5015] Handley, M., Kouvelas, I., Speakman, T., and L. Vicisano, "Bidirectional Protocol Independent Multicast (BIDIR-PIM)", RFC 5015, October 2007.

[RFC5384] Boers, A., Wijnands, I., and E. Rosen, "The Protocol Independent Multicast (PIM) Join Attribute Format", RFC 5384, November 2008.

10.2. Informative References

[AMT] Bumgardner, G., "Automatic Multicast Tunneling", Work in Progress, June 2012.

[PIM-REG] IANA, "Protocol Independent Multicast (PIM) Parameters", <http://www.iana.org/assignments/pim-parameters>.

Authors' Addresses

Dino Farinacci
Cisco Systems
Tasman Drive
San Jose, CA 95134
USA

EMail: dino@cisco.com

Greg Shepherd
Cisco Systems
Tasman Drive
San Jose, CA 95134
USA

EMail: gjshep@gmail.com

Stig Venaas
Cisco Systems
Tasman Drive
San Jose, CA 95134
USA

EMail: stig@cisco.com

Yiqun Cai
Microsoft
1065 La Avenida
Mountain View, CA 94043
USA

EMail: yiqunc@microsoft.com