

Project	IEEE 802.16 Broadband Wireless Access Working Group < http://ieee802.org/16 >
Title	Incremental Approach for MMR Network Topology Discovery
Date Submitted	2007-03-05
Source(s)	G.Q. Wang, Wen Tong, Peiyong Zhu, Hang Zhang, David Steer, Gamini Senarath, Derek Yu, Mo-Han Fong Voice: 1-613-763-1315 [mailto:wentong@nortel.com] Nortel 3500 Carling Avenue Ottawa, Ontario K2H 8E9
Re:	A response to a Call for Technical Proposal, http://www.wirelessman.org/relay/docs/80216j-07_007r2.pdf
Abstract	This document provides text descriptions for relay path mgmt and routing sections defined in ToC of IEEE 802.16j-06/026r2
Purpose	
Notice	This document has been prepared to assist IEEE 802.16. It is offered as a basis for discussion and is not binding on the contributing individual(s) or organization(s). The material in this document is subject to change in form and content after further study. The contributor(s) reserve(s) the right to add, amend or withdraw material contained herein.
Release	The contributor grants a free, irrevocable license to the IEEE to incorporate material contained in this contribution, and any modifications thereof, in the creation of an IEEE Standards publication; to copyright in the IEEE's name any IEEE Standards publication even though it may include portions of this contribution; and at the IEEE's sole discretion to permit others to reproduce in whole or in part the resulting IEEE Standards publication. The contributor also acknowledges and accepts that this contribution may be made public by IEEE 802.16.
Patent Policy and Procedures	The contributor is familiar with the IEEE 802.16 Patent Policy and Procedures < http://ieee802.org/16/ipr/patents/policy.html >, including the statement "IEEE standards may include the known use of patent(s), including patent applications, provided the IEEE receives assurance from the patent holder or applicant with respect to patents essential for compliance with both mandatory and optional portions of the standard." Early disclosure to the Working Group of patent information that might be relevant to the standard is essential to reduce the possibility for delays in the development process and increase the likelihood that the draft publication will be approved for publication. Please notify the Chair < mailto:chair@wirelessman.org > as early as possible, in written or electronic form, if patented technology (or technology under patent application) might be incorporated into a draft standard being developed within the IEEE 802.16 Working Group. The Chair will disclose this notification via the IEEE 802.16 web site < http://ieee802.org/16/ipr/patents/notices >.

Incremental Approach for MMR Network Topology Discovery

G.Q. Wang, Wen Tong, Peiyong Zhu, Hang Zhang, Gamini Senarath,

David Steer, Derek Yu, Mo-Han Fong

Nortel

Introduction

In current 802.16j architecture, the MMR network topology is a tree-like hierarchy, and the communication connection is established with the selected path within the tree layer-by-layer. All the connections, no matter they are tunnel CIDs, or basic CIDs, have a binding relationship with existing MMR topology. We can use the pre-established connection and their topological property for the new node entry process. For example, when a RS receives RNG-req from multiple new nodes, it encapsulates RNG-req messages into the pre-established connections, and forwards the data upstream to BS. While BS receives the tunneled RNG-req, it would handle the entry process accordingly, and tunnel RNG-rsp back to the RS. Then the tunnel-end-point RS de-caps the tunnel and sends RNG-rsp messages to each new node. Note that the pre-established connection is not only used for RNG-req/rsp, it is actually also used for all the messages in entry process such as SBC and REG-req/rsp.

Due to that the pre-established connection is associated with an existing path, when BS receives the tunneled RNG-rsp from the new nodes, it would learn the new topology by combining the pre-established path and the newly attached nodes, and to form a new path in its routing database. As the MMR tree is built up layer-by-layer, it provides an incremental way to construct routing database and path/CID binding which eventually cover all the nodes in MMR tree.

The following sections describe the details how to use pre-established connection recursively to support MMR network topology discovery. In this approach, it only requires store tunnel CID or basic CID in all intermediate RS, and it does not change any entry process defined 802.16e-2005.

1. Initial entry and incremental topology creation

The following procedure systematically describes how BS learns MMR topology using pre-established connections and the attachment of new node; and how BS creates the new connection to the new node for the future topology discovery.

1. New node sends RNG-req with initial ranging CID, to the selected RS
2. The selected RS receives RNG-req and encapsulates it with other RNG-req messages (from other new node, if any) into tunneled data burst, and sends it upstream
3. Upon received tunneled data, BS handles RNG-req following the standard entry procedure, and then tunnels all RNG-rsp into data burst and sends it back to the designated RS. From the received CID and new node information, BS can create a new path in its routing database
4. Using the same procedure described above, BS and the designated RS would tunnel all the messages back and forth for the rest operations associated with ranging, capacity negotiation, security and registration from the new node.
5. After the new node gets into normal operation mode, if it is a RS node, BS should create a new path and populate path/CID binding data to all the RS along this new path. This makes the new path and new CIDs available for the future topology discovery.
6. In this process, only tunnel CID or basic CID need to be stored in all intermediate RS.

2. Call flow for incremental network entry and topology discovery

The following is a figure to show the call flow of incremental network entry under and topology discovery. The described functional flow is applicable to both RS entry and MS entry. The description also reflects the routing database updates and path/CID binding relationship.

In this example, after RS1 attachment, the BS would create a tunnel connection for RS1. The RS2 entry process is tunneled between BS and RS1. Similarly, BS creates a tunnel connection for RS2, and MS entry process is tunneled between BS and RS2.

Note that the messages in this figure are only for illustrative purpose.

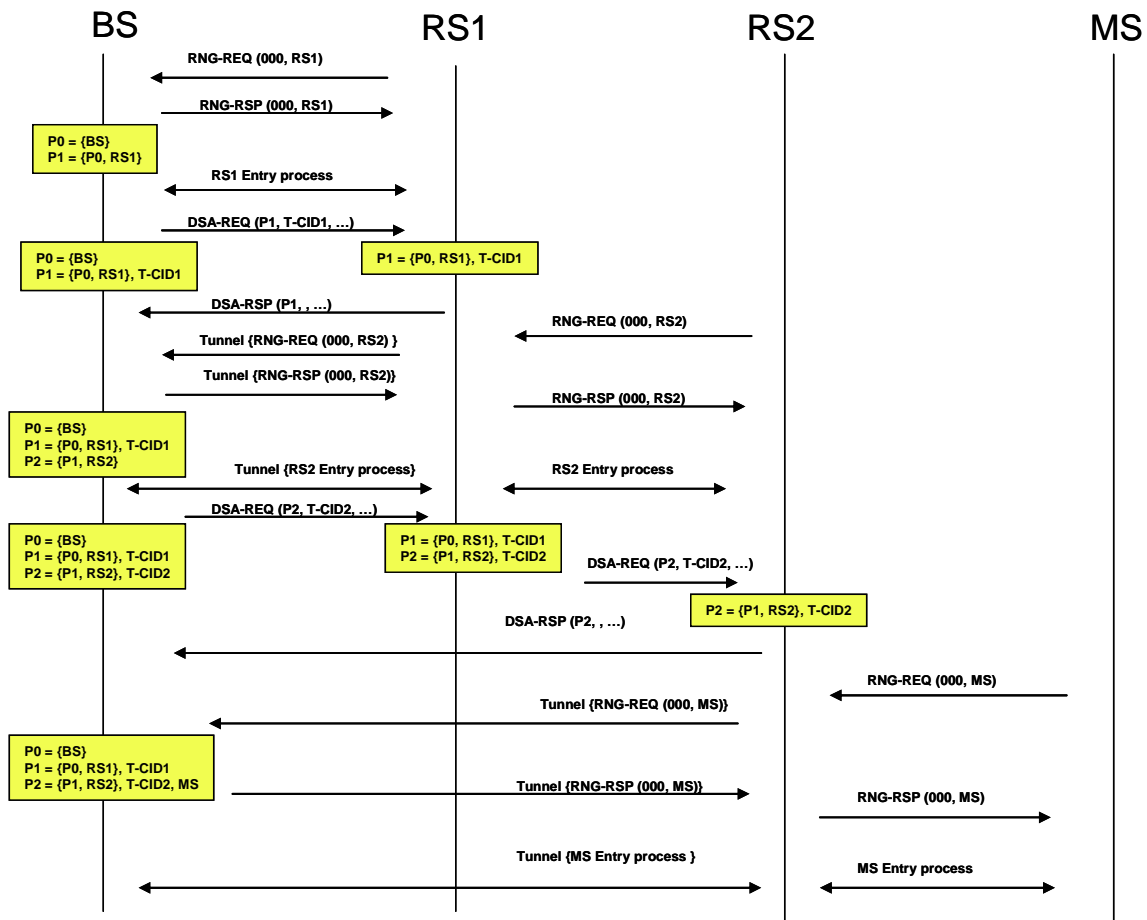


Fig 1. MMR topology discovery using tunnel connection

3. Proposed text changes

+++++++ start text proposal ++++++

[Insert the followings in sections of 6.3.25]

6.3.25.x MMR Initial entry and incremental topology discovery

The following procedure systematically describes how BS learns MMR topology using pre-established connections and the attachment of new node; and how BS creates the new connection to the new node for the future topology discovery.

1. New node sends RNG-req with initial ranging CID, to the selected RS
2. The selected RS receives RNG-req and encapsulates it with other RNG-req messages (from other new node, if any) into tunneled data burst, and sends it upstream
3. Upon received tunneled data, BS handles RNG-req following the standard entry procedure, and then tunnels all RNG-rsp into data burst and sends it back to the designated RS. From the received CID and new node information, BS can create a new path in its routing database
4. Using the same procedure described above, BS and the designated RS would tunnel all the messages back and forth for the rest operations associated with ranging, capacity negotiation, security and registration from the new node.
5. After the new node gets into normal operation mode, if it is a RS node, BS should create a new path and populate path/CID binding data to all the RS along this new path. This makes the new path and new CIDs available for the future topology discovery.
6. In this process, only tunnel CID or basic CID need to be stored in all intermediate RS.

+++++++ End of text proposal ++++++