IEEE P802.11
Wireless LANs

|  |
| --- |
| Additional REVmd LB236 EDITOR2 ad-hoc related comment resolutions |
| Date: 2019-03-01 |
| Author: |
| Name | Affiliation | Address | Phone | Email |
| Edward Au | Huawei Technologies | 303 Terry Fox Drive, Suite 400, Ottawa, Ontario K2K 3J1 |  | edward.ks.au@huawei.com  |

##### This submission present proposed resolution for CIDs 2095, 2096, and 2659. The proposed changes are based on REVmd/D2.1.

##### Revision history:

##### R0 – initial version

R1 – CID 2013 is pulled out for further discussion.

R2 – CIDs 2041 and 2042 are pulled out and transferred to PHY. Resolution for CID 2095 is updated.

R3 – CID 2222 is transferred to MAC.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| CID | Clause | Page | Line | Comment | Proposed Change |
| 2095 | 10.43.6.4.4 | 2028 | 17 | Change the following instances of "that has received" to "that receives": 10.43.6.4.4 (P2028L17 and P2028L40), 11.2.7.3.3 (P2187L29). | As suggested, first one listed only.Note we have many instances of "that has received", which are correctly used to indicate reception is a first-stage condition before another condition (often another reception) that the sentence is describing next (often in the format "that has received x and that receives y"), e.g.,"A STA that has received a PSMP frame and that receives a QoS Data frame with the Ack Policy field equal to PSMP Ack or that receives a Basic BlockAckReq or Multi-TID BlockAckReq frame shall transmit a Basic BlockAck frame or Multi-TID BlockAck frame..."When reception is stated not as a first-stage condition to a second reception or similar, "that receives" is preferred. |
| 2096 | 11.22.7.3 | 2353 | 29 | Normative language; change "that has received" to "that receives" | Change "A non-AP STA's SME that has received an MLME-BTM.indication primitive forwards the MLME-BTM.indication parameters to ..." to "A non-AP STA's SME that receives an MLME-BTM.indication primitive shall forward the MLME-BTM.indication parameters to ..." |

***Discussion:***

Per the note as provided by the commenter in his proposed change:

*Note we have many instances of “that has received”, which are correctly used to indicate reception is a first-stage condition before another condition (often another reception) that the sentence is describing next (often in the format “that has received x and that receives y”), e.g.,*

*“A STA that has received a PSMP frame and that receives a QoS Data frame with the Ack Policy field equal to PSMP Ack or that receives a Basic BlockAckReq or Multi-TID BlockAckReq frame shall transmit a Basic BlockAck frame or Multi-TID BlockAck frame...”*

*When reception is stated not as a first-stage condition to a second reception or similar, “that receives” is preferred.*

There are 9 instances of “that has received” in D2.1.

[1] 773.50 in clause 9.2.2 (Conventions)



This is a declarative description about the convention being used in the standards. No change is needed.

[2] 1710.30, clause 10.3.2.9 (CTS and DMS CTS procedure)



Reception of the RTS frame is stated as a condition before another condition (i.e., the status of NAV and CCA). No change is needed.

[3] 1888.16, clause 10.31.2.7 (PSMP acknowledgment rules)



Reception of the PSMP frame is stated as a first-stage condition before the second condition (i.e., the STA receives a QoS Data frame). No change is needed.

[4] 1922.18, clause 10.37.1 (HT NDP sounding protocol)



Reception of the HT NDP announcement frame is stated as a first-stage condition before the second condition (i.e., the STA does not receive the HT NDP). No change is needed.

[5] 2028.17, clause 10.43.6.4.4 (Antenna configuration setting during a beam refinement transaction)



There is no second condition for the first condition (i.e., the STA receives a beam refinement transmit training request). Change is required per CID 2095.

[6] 2028.40, clause 10.43.6.4.4 (Antenna configuration setting during a beam refinement transaction)



There is no second condition for the first condition (i.e., the STA receives a BRP-RX PPDU from a peer STA). Change is required per CID 2095.

[7] 2188.14, clause 11.2.7.3.3 (PCP operation with a wakeup schedule)



There is no second condition for the first condition (i.e., the STA receives a PCP WS). Change is required per CID 2095.

[8] 2354.29, clause 11.22.7.3 (BSS transition management request)



There is no second condition for the first condition (i.e., the STA receives an MLME-BTM.indication primitive). Change is required per CID 2096.

[9] 2480.7, clause 11.43.1 (General)



It is a declarative text. No change is needed.

***Proposed resolution for CID 2095:***

Revised

Change the following instances of "that has received" to "that receives": 10.43.6.4.4 (P2028L17 and P2028L40), 11.2.7.3.3 (P2187L29).

***Proposed resolution for CID 2096:***

Accepted

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| CID | Clause | Page | Line | Comment | Proposed Change |
| 2659 | 11.1.4.5 | 2141 | 33 | MIB variables in normative text sometimes cause strange effects in sentences, because they are long and typically non-breaking. See examples on 2267.25, 2363.32, 2373.19, 2374.25, 2404.34, 2422.60, 2475.53, etc. Worse yet, sometimes MIB variables are broken up with a hyphen added (like at 2307.46), which makes them invisible in a search for the MIB variable. | Find a way for MIB variables in normative text to be broken up while still searched for easily. Maybe, in normative text, write MIB variables with spaces between the parts that they are comprised of, similar to element and field names. But certainly re-unite MIB variables that are currently broken up. |

***Discussion:***

Per our Editor (Emily Qi)’s comment that I agree:

*In a table or figure, since it is predictable, we added a space in the MIB variable to make it looks good when it need to be break.*

*For a MIB variable in the text, it is unpredictable on where it will be. If we added a space in this revision, it may look okay in D2.1. However, in D2.2, it may look very bad (terrible) when it shows in the middle of line with a space.*

Per the commenter (Menzo)’s offline comment:

*MIB variables in figures and tables are an issue indeed, because fitting them would require extensive reformatting. This might be a fundamental problem with using MIB variables in normative text.*

*For the ones in text, there are some that Acrobat does find and some that it does not. The first ones are presumably automatically generated, and the latter ones handmade (hardcoded). I think we should fix at least the hardcoded ones. I'm listing those below (14 total).*

*As for a solution for future amendments, is it possible to make less use of MIB variables? Or not at all?*

*Another option might be to include a unique number in the MIB variable name. Which would not improve readability much though.*

It is the best that the editors can do (i.e., fixing the 14 instances provided by Menzo) because the others are most probably generated by the framemaker and out of the editors’ control.

***Proposed resolution:***

Revised

1. At 310.9, replace “dot11NonAPStationBestEffort-MSDUCount” with “dot11NonAPStationBestEffortMSDUCount”.
2. At 310.13, replace “dot11NonAPStationDroppedBestEffort-MSDUCount” with “dot11NonAPStationDroppedBestEffortMSDUCount”.
3. At 312.46, replace “dot11NonAPStationMulticast-MSDUCount” with “dot11NonAPStationMulticastMSDUCount”.
4. At 312.51, replace “dot11NonAPStationDroppedMulticast-MSDUCount” with “dot11NonAPStationDroppedMulticastMSDUCount”.
5. At 312.61, replace “dot11NonAPStationBestEffort-MSDUCount” with “dot11NonAPStationBestEffortMSDUCount”.
6. At 313.1, replace “dot11NonAPStationDroppedBestEffort-MSDUCount” with “dot11NonAPStationDroppedBestEffortMSDUCount”.
7. At 848.41, replace “dot11QosOption-Implemented” with “dot11QosOptionImplemented”.
8. At 848.45, replace “dot11QosOption-Implemented” with “dot11QosOptionImplemented”.
9. At 868.16, replace “dot11QosOption-Implemented” with “dot11QosOptionImplemented”.
10. At 882.33, replace “dot11CMMG OptionImplemented” with “dot11CMMGOptionImplemented”.
11. At 1052.40, replace “dot11STAStatisticsAverageAccessDelay-Voice” with “dot11STAStatisticsAverageAccessDelayVoice”.
12. At 1052.41, replace “dot11STAStatisticsAverageAccessDelay-Voice” with “dot11STAStatisticsAverageAccessDelayVoice”.
13. At 2296.28, replace “dot11DSEEnablement-TimeLimit” with “dot11DSEEnablementTimeLimit”.
14. At 2307.47, replace “dot11BSSWidthTriggerScan-Interval” with “dot11BSSWidthTriggerScanInterval”.