IEEE P802.11  
Wireless LANs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Clause 6 – Investigation | | | | |
| Date: 2022-02 | | | | |
| Author(s): | | | | |
| Name | Affiliation | Address | Phone | email |
| Graham SMITH | SR Technology | Sunrise, FL, USA. | 916 799 9563 | gsmith@srtrl.com |

Abstract

Look in depth at Clause 6

Rev 1 START AT PAGE 9

**Background:**

Contribution 21/1822 noticed that Clause 6 consumes 434 pages MORE THAN Clause 11 (398 pages) and questioned if it was really worth it.

Contribution 21/1822 also looked at refeerrences in text to Clause 6 and proposed an approach along the lines of

**ACTION.request primitives are detailed in this clause when they do not directly correspond to frame described in “Clause 9” or Clause 11 (note some might be elsewhere), or where the primitive parameters significantly differ.**

**Hence, first short list might be:**

6.3.2 MLME-POWERMGT

6.3.3 MLME SCAN

6.3.4 MLME JOIN (synchronization)

6.3.5 MLME AUTHENTICATE

6.3.6 MLME DEAUTHENICATE

6.3.7 MLME ASSOCIATE

6.3.8 MLME REASSOCIATE

6.3.9 MLME DISASSOCIATE

6.3.10 MLME RESET

6.3.11 MLME START

6.3.12 MLME STOP

6.3.19 MLME-SETKEYS

At Feb 24 telecon meeting:

Consensus on worthwhile work.

Noted diagrams in 6.3.13 and TDLS and Timing. As these are “abstract” not sure these figures are needed here. Should fit the standard model.

Possible Way ahead is to expand the Introduction to 6.3. Add diagrams showing the two different models, internal command, and request/response.

Possibly a list of all “standard” promitives meeting one or other model. Maybe with reference to Clause 9 or 11 where the respective packet is defined.

First some acronyms:

SME – Station management entity

MLME – MAC sublayer management entity

PLME – PHY layer management entity

SAP – Service Access Point

**6.3 MLME SAP interface**

The services provided by the MLME to the SME are specified in this subclause. These services are

described in an abstract way (following the model described in ITU-T Recommendation X.210 [B55]) and

do not imply any particular implementation or exposed interface. **MLME SAP primitives are of the general**

**form ACTION.request primitive followed by ACTION.confirm primitive (for an exchange initiated by the**

**SAP client) and ACTION.indication primitive followed by ACTION.response primitive (for an exchange**

**initiated by the MLME)**. The SME uses the services provided by the MLME through the MLME SAP

NOTE: “These services are described in an abstract way…and do not imply any particular implementation…”

Let’s list this out clearer:

**MLME SAP primitives are of the general form**

For an exchange initiated by the SAP client

* ACTION.request primitive,
* ACTION.confirm primitive

For an exchange initiated by the MLME

* ACTION.indication primitive
* ACTION.response primitive

Now most exchanges are of the form “Request / Response”

BTW I coud not find any similar drawing or indeed anything useful in the ITU document.

Based on this “General Form” I interpret this as following diagram:

**FIGURE 1 – General form of MLME SAP Primitives for Request/Response**



**FIGURE 2 – General form of MLME SAP Primitives for SME requests MLME to** something that does not initiate a packet to a peer STA, e.g. START



AND there is a third

**FIGURE 3 – General form of MLME SAP Primitives for request to perform a requested Action and report on it.**



Soo…let’s look at an example that we should all understand. ASSOCIATE.

**MLME-ASSOCIATE.request**

This primitive requests association with a specified peer MAC entity that is within an AP.

When generated:

This primitive is generated by the SME when a STA wishes to establish association with an AP or PCP.

Effect of receipt

This primitive initiates an association procedure. In the case that a response is received from the responder STA, the MLME subsequently issues an MLME-ASSOCIATE.confirm primitive that reflects the results.



**MLME-ASSOCIATE.confirm**

This primitive reports the results of an association attempt with a specified peer MAC entity that is in an AP or PCP.

When generated:

This primitive is generated by the MLME as a result of an MLME-ASSOCIATE.request primitive **or** receipt of an Association Response frame from the peer MAC entity to associate with a specified peer MAC entity that is in an AP or PCP.

*NOTE: This seems to indicate that the MLME sends a ‘confirmation’ of receipt back to the SME, and then later on, the confirmation that the Association Response frame has been received. Maybe this should have been “on” receipt.*

*NOTE 2 (3/7/meeting) Could be that a .confirm is sent if error in the .request or time out or ??? Might need look at the error codes and then sort out. But in any case, would need a lot more description – if it was worth it!*

Effect of receipt

The SME is notified of the results of the association procedure.

*NOTE: This does not align if really is “or”*

**MLME-ASSOCIATE.indication**

This primitive indicates that a specific peer MAC entity is requesting association with the local MAC entity, which is in an AP or PCP

When generated:

This primitive is generated by the MLME as a result of the receipt of an association request from a specific peer MAC entity

Effect of receipt

The SME is notified of the receipt of the association request

**MLME-ASSOCIATE.response**

This primitive is used to send a response to a specific peer MAC entity that requested an association with the

STA that issued this primitive, which is in an AP or PCP.

When generated:

This primitive is generated by the SME of a STA that is in an AP or PCP as a response to an MLMEASSOCIATE.indication primitive

Effect of receipt

This primitive initiates transmission of an AssociationResponse to the specific peer MAC entity that requested association.



Is this right? It fits exactly with the description. Does SME need a confirm?

Note

**MLME-REASSOCIATE.confirm**

This primitive is generated by the MLME as a result of an MLME-REASSOCIATE.request primitive to reassociate with a specified peer MAC entity that is in an AP or PCP.

*NOTE: REASSOCIATE not include the “immediate” confirm. Neither does the DISASSOCIATE.*

**FIRST PROPOSED CHANGE**

I suggest the “or” should be changed to “on”.

Then ASSOCIATE meets the general form FIG. 1.

Let’s look at another diagram that was inserted. TDLS direct link establishment. How many problems here?

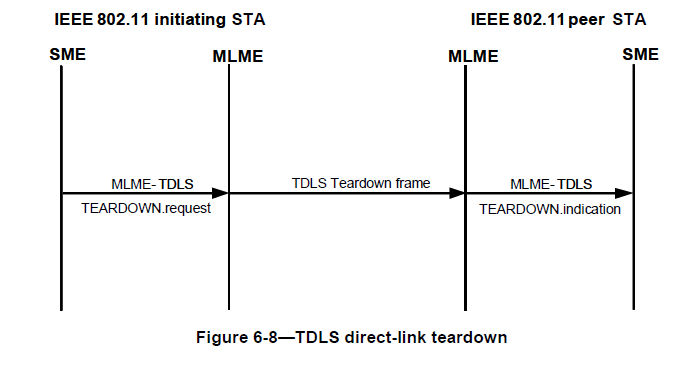


This is scary. I suspect it has never been looked at since it was written.

Do we need a primitive if no response happens?

Discussion (3/7/22) This is a 3 exchange scheme hence, 3 type 4 exchanges, single

Also look at Figure 6.8.

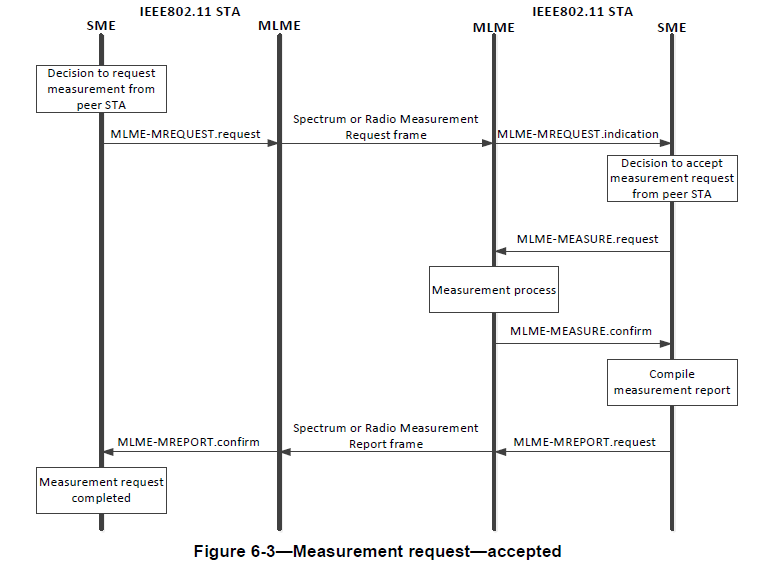
****

With no response at all, should this even be here? Note should be Type 4 case.

**Let’s discuss**.

There are 24 Figures in Clause 6.3. How many are really needed? If they meet one of the “General Cases” they need not be there?

Let’s now look at FIG. 6.3



This is classic case of General Form #3. The action is pretty clear from the text, i.e. get a request, do it, then send the response.

**NEXT STEPS**

Go through all figures in 6.3. Look at text to see if they meet one of the “General Forms” and how they are actually referred to.

Are they needed? Is description in Clause 9 or 11 sufficiently clear. What about the boxes?

We could in text refer to which “Form” of MLME SAP primitive is used.

Make decision if any do not fit the four “General Forms”. Is there another?

**OBJECTIVE**

Write Introduction with descriptions of the 4 General Forms and see if these adequately cover all the 125 cases minus the 12 listed earlier.

**In Clause 12.2.10. P3097.10, we have**

The SME of the non-AP STA may change the MAC address by generating an MLMEUPDATEMACADDRESS. request primitive containing the new MAC address. On receipt of an MLME-UPDATEMACADDRESS.request primitive, the MLME shall attempt to update the MAC address that is to be used by the MAC entity and shall generate an MLME-UPDATEMACADDRESS.confirm primitive to notify the SME whether the MAC address has been changed to the new value.

Note that it appears at 6.3.117 P815. In this case it is clear that we do not need 6.3.117.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Proposed new 6.3 Introduction

**6.3 MLME SAP interface**

**6.3.1 Introduction**

The services provided by the MLME to the SME are specified in this subclause. These services are described in an abstract way (following the model described in ITU-T Recommendation X.210 [B55]) and do not imply any particular implementation or exposed interface. MLME SAP primitives are of the general form ACTION.request primitive followed by ACTION.confirm primitive (for an exchange initiated by the SAP client) and ACTION.indication primitive followed by ACTION.response primitive (for an exchange initiated by the MLME). The SME uses the services provided by the MLME through the MLME SAP.

**6.3.1.1 Types of MLME-SAP interface primitive forms**

There are six general forms of MLME-SAP interface primitives.

Figure 6.x depicts Type 1. The Type 1 general form is used for the exchange of request/response frames between an initiating STA and a peer STA.

**FIGURE 6.x – Type 1 form of MLME SAP primitives for request/response process**



The .request primitive is generated by the SME of the initiating STA to request that a Request frame is sent to a peer STA.

The .indication primitive is generated by the MLME of the peer STA when the Request frame is received.

The .response primitive is generated by the SME of the peer to request that a Response frame be sent to the initiating STA.

The .confirm primitive includes a Result Code reporting success or failure of the request, and is generated by the MLME of the initiating STA when the either Response frame from the peer STA is acknowledged, or the (re)transmission of the request frame fails.

Figure 6.xx depicts Type 2. The Type 2 general form is used for the SME requesting a process to be initiated by the MLME.

**FIGURE 6.xx – Type 2 form of MLME SAP primitives for SME requesting MLME to perform a process**



The .request primitive is generated by the SME to request that a process is initiated by the MLME.

The .confirm primitive generally includes a Result Code reporting success or failure of the request, and is generated by the MLME when the requested action of process is completed, or fails.

*??? We also have several .request only, i.e., no .confirm. Do we need a separate figure/type for this, or just cover it in the text as follows.*

If the SME does not require a confirmation then the .confirm primitive may be omitted.

Figure 6.xxx depicts Type 3. The Type 3 general form is used for the exchange of request/response frames between an initiating STA and a peer STA where the peer STA is requested to perform an action before responding.

**FIGURE 6.xxx – Type 3 form of MLME SAP primitives for request to peer STA to perform an action before responding.**



The .request primitive is generated by the SME of the initiating STA to request that a Request frame is sent to a peer STA.

The .indication primitive is generated by the MLME of the peer STA when the Request frame is received.

The .request primitive is generated by the SME of the peer STA to request that a process is initiated by the MLME of the peer STA.

The .confirm primitive from the MLME of the peer STA generally includes a Result Code reporting success or failure of the request, and is generated by the MLME when the requested action of process is completed, or fails.

The .response primitive is generated by the SME of the peer STA to request that a Response frame be sent to the initiating STA.

The .confirm primitive includes a Result Code reporting success or failure of the request, and is generated by the MLME of the initiating STA when the either Response frame from the peer STA is acknowledged, or the (re)transmission of the request frame fails.

Figure 6.xxxx depicts Type 4. The Type 4 general form is used for the transmission of a frame from one STA to a peer STA that does not require a response from the peer STA.

**FIGURE 6.xxxx – Type 4 form of MLME SAP primitives for frame transmission not requiring a response**

The .request primitive is generated by the SME of the initiating STA to request that a Request frame is sent to a peer STA.

The .indication primitive is generated by the MLME of the peer STA when the Request frame is received.

Figure 6.xxxxx depicts Type 5. The Type 5 general form is used for the transmission of a frame that does not require a response, but does require a confirmation that it was sent.

**FIGURE 6.xxxxx – Type 5 form of MLME SAP primitives for a frame transmission from a STA, but does require a confirmation that it was sent**



The .request primitive is generated by the SME to request that the MLME transmits a frame.

The .confirm primitive generally includes a Result Code reporting success or failure of the request, and is generated by the MLME when the requested frame transmission is completed, or fails.

Figure 6.xxxxxx depicts Type 6. The Type 6 general form when the MAC informs the SME of an event.

**FIGURE 6.xxxxxx – Type 6 form of MLME SAP primitives for MAC informing SME of an event**



The .indication primitive is generated by the MLME to inform the SME of an event.

**6.3.1.2 MLME-SAP Primitives**

MLME-SAP primitives are detailed in this clause when they do not directly correspond to frame exchanges described in Clause 9 or Clause 11, where the primitive parameters differ significantly from the fields in the respective Action frames, or when the primitives may not be clear from the descriptions in Clauses 9 or 11.

**Include the following, renumbering where necessary:**

6.3.2 MLME-POWERMGT Type 2

6.3.3 MLME SCAN Type 2 plus STOP request (with no confirm)

6.3.4 MLME JOIN (synchronization) Type 2

6.3.5 MLME AUTHENTICATE Type 1

6.3.6 MLME DEAUTHENICATE request.confirm,indication

6.3.7 MLME ASSOCIATE Type 1

6.3.8 MLME REASSOCIATE Type 1

6.3.9 MLME DISASSOCIATE request.confirm,indication

6.3.10 MLME RESET request (Type 2 no confim)

6.3.11 MLME START Type 2

6.3.12 MLME STOP request (Type 2 no confirm)

6.3.13 Protocol layer model for spectrum management and radio measurement

6.3.19 MLME-SETKEYS (renumber to 6.3.14)

**Look through the rest, one by one to see if a case for keeping them or if adequately described in the relate text.**

**Look at recommendations in 21/1822 for text changes related to 6.3 references in text.**

6.3.14 Measurement Request

MREQUEST. Request and indication.

Type 4

6.3.15 Channel measurement

MEASURE. Request and confirm.

Type 2

6.3.16 Measurement report

MREPORT.request and indication

Type 4,

COMMENT: These measurement primitives seem that they should be better served by type 1. Discuss.

6.3.17 Channel switch

Type 1

6.3.18 TPC request

Type 2

11.7.7

6.3.20 Delete keys

Request only. Type 2 without the confirm. Is that right?

6.3.21 MIC failure

Indication only.

Type 6.

6.3.22 EAPOL

Type 2

6.3.23 Set Protection

Request only. Type 2 without the confirm

6.3.24 Protected frame dropped - (Note different heading!!)

Indication only.

Type 6

6.3.25 TS management

LOOK AT THIS ONE. Think all Type 1

6.3.26 Higher layer synchronization support

Type 4

6.3.27 Block Ack

ADDBA Type 1.

DELBA Type 4.

6.3.28 Schedule element management

Type 4

6.3.29 Vendor Specific action

Type 4

6.3.30 Neighbor report

Type 4

6.3.31 Neighbor report response

Type 4

COMMENT – Why two Type 4’s, surely should be Type 1 or 3

6.3.32 Link Measure request

Type 2

6.3.33 Resource request (again different heading style)

Type 1

Resource Request - Local

Type 2. (MIGHT NEED LOOKING AT) Why not separate?

6.3.34 Remote requests (Different heading format)

Type 4

6.3.35 Extbded channel switch announcement

Type 1

6.3.36 DSE power constraint announcement

Type 1

6.3.37 Enablement

Type 1

6.3.38 Deenablement

Type 4

6.3.39 SA Query support

Type 1

6.3.40 Get TSF timer

Type 2

6.3.41 Timing Advertisement

Type 4

6.3.42 TDLS Discovery

Type 1

6.3.43 TDLS direct link establishement

See discussion

6.3.44 TDLS direct link teardown

Type 4

6.3.45 TDLS peer U-APSD

Type 1

6.3.46 TDLS channel switching

Type 1

6.3.47 TDLS peer PSM

Type 1

6.3.48 Event request

Type 4

6.3.49 Event report

Type 4 Why two Type 4’s, surely should be Type 1 or 3

6.3.50 Event

Type 2

6.3.51 Diagnostic request

Type 4

6.3.52 Diagnostic reporo

Type 4 Why two Type 4’s, surely should be Type 1 or 3

6.3.53 Location configuration request

Type 1

6.3.54 Location track notification

Type 4

6.3.55 Timing measurement

Need to study this.

Type 4, followed by type 4 with an added .confirm. Begs the question why not a confirm on the first one. I.e. is an ACK always followed by a confirm? Seems superfluous. The more I look at these the worse it gets.

6.3.56 Fine timing measurement

Same as Timing.

6.3.57 BSS transition management procedure

Type 4 , then Type 1

Diagram appears useful, does it appear in similar form in text? No but primitives are described in text. (see 11.21.7.4) So decision is whether it is clear without the diagram in 6.3.57?

6.3.58 FMS setup

Type 1

6.3.59 Collocated interference request

Type 4

6.3.60 Collocated interference response

Type 4

AGAIN why two type 4’s. Should be Type 1 or 3

6.3.61 TFS setup

Type 1

6.3.62 WNM sleep mode request

Type 1

6.3.63 TIM broadcast setup

Type 1

6.3.64 QoS traffic capability update

Type 4

6.3.65 Channel Usage request

Type 1 (but really is a Type 3)

6.3.66 DMS or GCR request and response procedure (Again not standard heading)

GATS is type 1 or 3

GATS TERM is type 4

6.3.67 WNM notification request

Type 4

6.3.68 WNM notification response

Type 4 AGAIN why two type 4’s. Should be Type 1 or 3

6.3.69 Network discovery and selection support

Type 1

6.3.70 QoS Map element management

Type 4 (Note: Lots of normative text in the General. Is this the right place for this? Hopefully it is in the main text)

6.3.71 Mesh peering management

Type 1

6.3.72 Mesh power management

Type 2

6.3.73 Mesh neighbor offset synchronization

Three Type 2 START, CALCULATE, STOP

6.3.74 Mesh TBTT adjustment

Type 1

6.3.75 MCCA management interface

A single .request MAC to “activate MCCA” Type 2A

Then a Type 1

6.3.76 MBSS congestion control

Type 4

6.3.77 MBSS proxy update

Type 1

6.3.78 MBSS mesh gate announcement

Type 4

6.3.79 Mesh link metric

Type 1

6.3.80 HWMP mesh path selection

Type 4

6.3.81 QMF policy

POLICY Type 2

POLICYCHANGE Type 4

POLICYSET single .request why no .confirm? Type 2A

6.3.82 SCS request and response procedure

Type 1

Then Type 4

6.3.83 QLoad report management

Type 1

6.3.84 HCCA TXOP advertisement management

Type 1

6.3.85 GCR group membership management

Type 1

6.3.86 AP PeerKey management

Type 4

6.3.87 On-channel Tunneling operation

Type 6

6.3.88 Multi band operation

Type 1 SETUP

Type 1 ACK

Type 4 TEARDOWN, but think it should be Type 2?

INCOMING Single .request (no-confirm) Type 2A

6.3.89 DMG relay operation

SEARCH Type 1

RLS Type 1

TEARDOWN Type 4

6.3.90 Quieting adjacent BSS operation

Type 1

6.3.91 DMG beamforming

SISO Type 6

SISO MIMO Type 6

MIMO Beamforming Type 6

MIMO hybrid Type 6

MIMO hybrid beamforming Type 6

6.3.92 PN event report

Two .indications Type 6.

6.3.93 Channel availability query

Type 1

6.3.94 Channel schedule management

Type 1

6.3.95 Contact verification signal

Type 4

6.3.96 GDD Enablement

Type 1

6.3.97 Network channel control management

Type 1

6.3.98 White space map

Type 4

6.3.99 Estimated Throughput

Type 2

6.3.100 Get authentication and association state

Type 2

6.3.101 FILS Container

Type 1

6.3.102 Dynamic AID assignemtn operation

Type 1

6.3.103 Sync control

Type 4

6.3 104 STA Information Announcement

Type 4

6.3.105 EDCA Parameter Set update

Type 4

6.3.106 EL operation

Type 4

6.3.107 TWT setup

Type 1

6.3.108 TWT teardown

Type 4

6.3.109 Sectorized Group ID List management

Type 4

6.3.110 Header Compression procedure

Type 1

6.3.111 Reachhable Address update

Type 4

6.3.112 Control response MCS negotiation operation

Type 1

6.3.113 S1G relay

Type 1

6.3.114 DCS procedure

MEASUREMENT Type 1

DCS Type 1

6.3.115 Update

Type 2

6.3.116 MSCSrequest and response procedure

Type 1

MSCS TERM Type 4

6.3.117 MAC Address update

Type 2

6.3.118 Quiet time period

Type 1

6.3.119 TDD beamforming

Type 1 without the response. Is this another type?

6.3.120 TDD sector switch

Type 1 without the response. Is this another type?

6.3.121 TDD beam measurement

Type 1 without the response. Is this another type?

6.3.122 TDD structure and schedule

STRUCTURE Type 2

SCHEDULE Type 2

ANNOUNCE Type 1 without the response. Is this another type?

BANDWIDTH Type 1 without the response. Is this another type?

6.3.123 WUR mode set up

Type 1

6.3.124 WUR mode teardown

Type 4

6.3.125 WUR Discosvery

Type 2.