IEEE P802.15
Wireless Personal Area Networks

Project	IEEE P802.15 Working Group for Wireless Personal Area Networks (WPANs)	
Title	**Draft running comment resolution**	
Date Submitted	[10 January, 2004]	
Source	[James P. K. Gilb] [Appairent Technologies] [16990 Via Tazon, #125, San Diego, CA 92127]	Voice: [858-485-6401] Fax: [858-485-6406] E-mail: [last name at ieee dot org]
Re:	[]	
Abstract	[This document is a record of comment resolutions for draft DF8 of 802.15.3b.]	
Purpose	[To provide a record of the comment resolutions for draft DF8 of 802.15.3b.]	
Notice	This document has been prepared to assist the IEEE P802.15. 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 acknowledges and accepts that this contribution becomes the property of IEEE and may be made publicly available by P802.15.	

1. Comment resolution in Monterey

1.1 Monday

Called to order at 4:09 pm PST

Page 4, line 23, “that is” appears three times, rewrite to read more nicely, perhaps “that is not operating as an 802.15.3 piconet,”

CID 121: ACCEPT

CID 11: ACCEPT IN PRINCIPLE. The TE will renumber the subclauses to make more sense. Discuss with ADH.

CID 12: Table until Tuesday.

CID 13: REJECT. ASIEs are used exclusively in the beacon and can be requested by a DEV to be placed there by the PNC. ASIE is intended to be used by applications. On the other hand, the Vendor Specific IE is used for MAC to MAC communications only.

CID 15: Table until Peter J. can discuss

CID 16: Table until Monday night ad-hoc

CID 17: Table until Peter J. can discuss

CID 18: REJECT. Although this would remove one of the MLMEs, it would require adding one for starting a child or neighbor piconet.

CID 19: ACCEPT.

CID 20: ACCEPT.

CID 21: ACCEPT IN PRINCIPLE. Change the description in Table 3g for VendorSpecificIE to read "The Vendor Specific IE, if present, in the Association Response command, 7.5.1.2."

CID 22: ACCEPT IN PRINCIPLE. Change the name of the primitive to MLME-BSID-CHANGE.yyy, change the parameters as necessary, e.g., RequestType is not necessary. Change the MSC and any associated text in clause 8.

Meeting recessed at 6:00 pm PST.

Ad-hoc meeting

Suggest accept CID 66, allows replace for faster changes of the ASIE (instead of add/remove). Remove count value field and its associated text. Put “replace” in the place of “count” for the allowed values for the Request Type field. In 7.5.9.4 Change “define” to “defined”. Change the name of “Application specific ID” to be “Application Data ID” to match clause 6. Also add this field to the ASIE Request command. 7.4.7 does not define this field, so the definition would be “a unique number generated by the requesting DEV to differentiate ASIEs. This should be changed whenever the data in the Application Specific Data changes.” Add field to the ASIE Response command named “Previous Application Data ID”. Also add “PreviousApplicationDataID” to the MLME-APPLICATION-SPECIFIC.confirm.

0xFE - Add ASIE

other - modify

zero length Application Data indicates remove regardless of the Application Data ID.

A DEV reports up all ASIEs that have new Application Data IDs. If one goes missing, it reports via the MLME-APPLICATON-SPECIFIC.indication to the FCSL and ApplicationSpecificDataSet that has a member that is zero length. Add the Application Data ID to the ApplicationSpecificDataSet.

Now Request Type is ASIE Request ID, change Application Data ID to be ASIE Index. There are two numbers, the ASIE Request ID and the Application Data ID, used in a similar manner as the Stream Request ID and the Stream Index.

In 7.4.7, move the Application specific ID to the first field (after type and length) and rename it the ASIE Index.

ASIEs are not maintained during handover.

JPKG to make edits, add to the end of this document so that everyone can see it in one place.

Questions:

1) If the PNC accepts the request, is it required to keep it in every beacon?

2) If not, does the PNC remove it permanently if it can't put it in or does it just not send it for some number of beacons, after which time it re-appears?

3) How does the PNC remove an ASIE from the beacon? Does it use a null ASIE like stream termination?

1.2 Tuesday

ACK Policy - table to 8 am.

CID 65:

2. Misc. resolutions

2.1 Syncrhonize MLME in clause 8 with clause 6.

8.1 Introduction

Change the seventh paragraph as shown:

An example MSC is shown in Figure 91 that illustrates two MLME requests and the associated timeouts. In the first case, the request completes before the timeout expires and so the confirm returns with the ResultCode set equal to COMPLETED. In the second case, the requested action completes unsuccessfully before the timer expires and so the confirm primitive is returned with the ResultCode set equal to FAILURE and the ReasonCode indicates the reason for the failure, if known. In the third case, the requested action does not complete before the timeout expires and so the confirm primitive is returned with the ResultCode set equal to FAILURE and the ReasonCode set equal to TIMEOUT.

Throughout this clause, the procedures and MSCs are written as though the optional MLME SAP as defined in 6.3 is exposed and is supported by the MLME. For each procedure shown to be initiated by the FCSL, unless it is stated otherwise, the reader shall assume that the procedure may also be executed by the MLME without interaction with the FSCL. For those procedures not shown to be initiated by the FCSL, the reader shall assume that the procedure is executed by the MLME without interaction with the FSCL.

8.2.1 Scanning through channels

Change first paragraph as shown:

All DEVs shall use passive scanning to detect an active piconet. That is, DEVs shall be in receive mode for a period of time in a channel no less than mMinChannelScan ~~, as specified in the MLME-SCAN.request,~~ to look for beacon frames from a PNC. If a particular BSID, PNID, or PNC MAC address to scan for is not specified with a MLME-SCAN.request ~~open scan is specified in the MLME-SCAN.request~~, the DEV searches for any beacon frame. If a particular BSID, PNID, or PNC MAC address to scan for is ~~open scan is not~~ specified, the DEV shall ignore all received frames not matching ~~the PNID and BSID~~ parameter or parameters contained in the MLME-SCAN.request.

Change third and fourth paragraphs as shown:

DEVs search for piconets by traversing through all available PHY ~~the indexed~~ channels ~~indicated in the MLME-SCAN. request~~. A DEV may search the channels in any order as long as all valid channels are contained in the search pattern. The result of a scan shall include information on any parent, child, as described in 8.2.5, or 802.15.3 neighbor, as described in 8.2.6, piconets that were detected. This provides a complete inventory of each channel.

While searching, if any frame is received, the searching DEV shall stay in the channel for a minimum of mMinChannelScan from the time of reception of first frame and look for a beacon from the PNC. ~~If the DEV finds only a frame and no beacon it shall report it as a part of the MLME-SCAN.confirm primitive.~~ The DEV shall scan all indicated channels to find piconets before returning the scan information via the MLME-SCAN.confirm primitive. The DEV shall only report piconets found due to the reception of a beacon frame as a part of the MLME-SCAN.confirm primitive.

8.2.2 Starting a piconet

Change the first three paragraphs as shown:

A DEV that is instructed to start a piconet through MLME-START.request, as described in 6.3.3.1, shall try only to start its own piconet and shall not attempt to associate with an existing piconet. ~~The DME shall have recently completed a scan procedure and will have chosen the channel in which to start the piconet.~~

The ~~DME~~ MAC should choose the channel with the least amount of interference to start the piconet based on the results of a recent scan (either FCSL initiated or MAC initiated) ~~the ChannelRatingList returned in the MLME-SCAN.confirm primitive, as described in 6.3.2.2~~.

Once the MAC has received ~~DME has chosen a channel, it shall issue~~ the MLME-START.request primitive ~~with the chosen channel. The DEV~~, it shall listen to the channel for mMinChannelScan duration to determine if the channel is still clear. If, at the end of this listening period, the ~~DEV~~ MAC determines that the channel is clear, the DEV, now the PNC, shall commence broadcasting its beacon once every superframe duration. ~~If, however, the DEV determines that the channel is no longer clear, it shall issue an MLME-START.confirm with a ResultCode indicating a failure to start the piconet. The DME then has options that include sending another MLME-START. request with a different ChannelIndex to start a piconet in a different channel, associating as a regular DEV and requesting the formation of a dependent piconet.~~ When the

piconet starts, the PNC allocates an additional DEVID to itself for the purposes of exchanging data with other DEVs that become members of the established piconet.

8.2.3 PNC handover

Change the eighth paragraph as shown:

Meanwhile the chosen PNC capable DEV, after receiving an ACK to its PNC Handover Response command, will prepare to broadcast its first beacon as the new PNC. The current PNC shall place the PNC Handover IE in the beacon with the Handover Beacon Number field set to the beacon number of the superframe in which the new PNC will send its first beacon. After sending the last beacon, the old PNC relinquishes control of the piconet~~, generates an MLME-PNC-HANDOVER.confirm to its DME,~~ and stops generating beacons. The new PNC shall broadcast its first beacon at the time the beacon would have been sent by the old PNC. This time may vary from the actual time due to clock inaccuracies of the old and new PNCs. The new PNC shall start sending beacons with the time token counter set to one more than the time token of the last beacon that will be sent by the old PNC. The new PNC shall begin using the PNCID as the SrcID for all beacon or command frames transmitted. The new PNC shall use the PNCID or its previously assigned DEVID as the SrcID for all data frames transmitted. When the PNC handover is successful, the association of the remaining DEVs with the piconet is unaffected and hence they are not required to re-associate with the new PNC.

Delete the eleventh paragraph as shown:

~~In the MSC, the MLME-PNC-HANDOVER.response is sent when the DME is ready for the handover and is not tied to the arrival of the PNC Handover Information commands or PS Set Information Response commands.~~

8.2.5 Child piconet

Delete the third paragraph as shown:

~~If the DEV receives a private CTA from the PNC, the DEV DME configures the child PNC parameters using the MLME-START-DEPENDENT.request and confirm primitives, as described in 6.3.3.3 and 6.3.3.4.~~

8.2.6 Neighbor piconet

Change the fourth paragraph as shown:

If the PNC permits the formation of a neighbor piconet and there is sufficient channel time available, the PNC shall allocate a private CTA using the NbrID as both the source and destination DEVID. ~~After receiving this channel time allocation in the beacon, the DEV DME configures the neighbor PNC parameters using the MLME-START-DEPENDENT.request and confirm primitives, as described in 6.3.3.3 and 6.3.3.4.~~

8.2.7.2 Parent PNC stopping a dependent piconet

Change the first paragraph as shown:

If the parent PNC wishes to stop a child piconet, it shall terminate the stream allocated to the child piconet using the isochronous stream termination procedure, as described in 8.5.1.3. If the parent PNC wishes to stop a neighbor piconet, it shall send a Disassociation Request command, as described in 7.5.1.3, to the neighbor PNC. In either case, the dependent PNC shall either change channels, join another piconet as a dependent piconet or immediately initiate its shutdown procedure, as described in 8.2.7.1. The parent PNC

shall listen for the dependent PNC shutdown beacon sequence to determine when the dependent piconet CTA should be removed. The parent PNC may set a maximum time for the completion of the dependent shutdown sequence, after which the CTA will be removed regardless of the completion of the dependent shutdown procedure. ~~In the case of a child piconet, this timeout is set by the MLME while for a neighbor piconet, this time is set via the MLME-DISASSOCIATE. request primitive, as described in 6.3.6.1.~~ If the dependent PNC is a neighbor that is ~~that is~~ not operating ~~a piconet that is not~~ an 802.15.3 piconet, the parent PNC shall provide the same time as it would allow for its own shutdown sequence for the neighbor PNC to cease operations as a dependent piconet of the parent piconet before removing its private CTA.

8.3 Association and disassociation with a piconet

No change needed

8.3.1 Association

Delete the first paragraph as shown:

~~Prior to the association process, the DME issues an MLME-SYNC.request and receives an MLME-SYNC.confirm.~~

8.3.4 Disassociation

Change the fourth paragraph as shown:

If the beacons from the PNC are not received by the DEV for longer than the ATP, the DEV shall consider itself disassociated from the piconet and may try to associate again. ~~The DEV notifies the DME that the ATP expired using the MLME-ATP-EXPIRED.ind primitive.~~

8.4.3.5 Allocation of MCTAs

Change the second paragraph as shown:

The intent of the MCTA Allocation Rate field is to enable the DEVs in the piconet to approximately determine the length of time required to send a command to the PNC. This information might be used to set the timeout for commands sent by the MAC ~~parameters for the MLME primitives, as described in 6.3~~.

8.5.1.1 Isochronous stream creation

Delete the tenth paragraph as shown:

~~DEVs perform multicast negotiations at a higher layer. A DEV sets up a multicast stream at the request of the upper layer by sending a request to the PNC for a stream with the multicast ID as the destination. A DEV enables reception of a multicast stream by using the MLME-MULTICAST-RX-SETUP.request. This tells the MAC to receive frames from a particular source DEV with the DestD set to the MestID and with the stream index specified in the MLME.~~

8.13 Power management

Change the third paragraph as shown:

A DEV shall always establish membership with the piconet in ACTIVE mode. ~~If the DEV MLME changes its PM mode to ACTIVE without the prompting of the DME, it notifies the DME with the MLME-PM-MODE-ACTIVE.indication primitive as described in 6.3.22.7~~.

8.14 ASIE operation

This one has been completely replaced in DF8 and should be OK.

9.1.1 Security membership and key establishment

No change

9.1.4 Data integrity

Change the first paragraph as shown:

Data integrity uses an integrity code to protect data from being modified by parties without the cryptographic key. It further provides assurance that data came from a party with the cryptographic key. Integrity may be provided using a key shared by all piconet DEVs or using a key shared between only two DEVs. All secure data frames that fail integrity checks are passed to the FCSL ~~DME~~ using MLME-SECURITY-ERROR.indication~~indicate~~ and no other action is taken on the frame by the MLME.

9.1.6 Command integrity protection

Change the first paragraph as shown:

The integrity of commands may be protected just like any other data. Integrity protected commands sent between the PNC and a DEV shall be protected using the PNC-DEV management key. All secure commands that fail integrity checks are passed to the FCSL ~~DME~~ using MLME-SECURITY-ERROR.indication~~indicate~~ and no other action is taken on the frame by the MLME.

9.3.4 Membership update

Change the first paragraph as shown:

When the FCSL ~~DME~~ determines that there has been a change of membership status with a particular DEV or when a management or data key is changed, the FCSL ~~DME~~ shall issue an MLME-MEMBERSHIP-UPDATE.request to its MLME. This membership status change or key change may be the result of a successful establishment of a security relationship, key update process, termination of a security relationship or some other event. The process by which this change occurs is outside the scope of this standard.

9.3.5 Secure frame generation

Change the first paragraph as shown:

When a DEV wishes to send a secure frame, it shall use the keying material required for the type of frame and by the relationship between the sending DEV and the receiving DEV. For each security relationship, there are two keys used to protect secure frames: a management key and a data key. Table 62 provides a listing of which of the keys shall be used to protect secure frames and which frames shall be sent without security. A DEV shall not send a secure frame if the only key selection in Table 62 is none . A DEV shall not send an unprotected frame or a frame with an incorrect SECID when security is required for that frame. If the DEV is unable to find the corresponding key that is to be used, the MLME shall return an MLME-SECURITY-ERROR.indication to the FCSL ~~DME~~ with the ReasonCode set to UNAVAILABLE-KEY and shall not transmit the requested frame.

9.3.6 Secure frame reception

Change the third, fourth and fifth paragraphs as shown:

When a DEV receives a secure beacon frame (a beacon with the SEC field in the Frame Control field set to one, the DEV shall determine if the received time token is greater than the CurrentTimeToken and less than the LastValidTimeToken + aMaxTimeTokenChange. If not, the MLME shall return an MLME-SECUITY-ERROR. indication to the FCSL ~~DME~~ with the ReasonCode set to BAD-TIME-TOKEN and shall not perform any additional operations on the received beacon. The DEV shall also determine if the SECID matches the SECID of the piconet group data key stored in the MAC/MLME, or the SECID of a valid old piconet group data key, as described in 9.2.5. If the SECID does not match, the DEV may set the CurrentTimeToken to the value in the beacon and request a new piconet group data key, as described in 9.3.2. If both of these checks succeed, the DEV shall check the integrity code on the beacon using the piconet group data key. If this succeeds, the DEV shall accept the beacon and set the LastValidTimeToken and CurrentTimeToken to be the time token in the beacon. If the DEV is able to determine that it missed a beacon or that the beacon was corrupted and if CurrentTimeToken is less than LastValidTimeToken + aMaxTimeTokenChange - 1, the DEV should increment the CurrentTimeToken to maintain synchronization with other DEVs in the piconet.

When a DEV receives a secure non-beacon frame, it shall use the appropriate keying material depending on the type of frame, SECID and source address found in the frame. If the SECID in the frame does not correspond to known keying material in the receiving DEV, the MLME shall return an MLME-SECURITYERROR. indication to the FCSL ~~DME~~ with the ReasonCode set to UNAVAILABLE-KEY and shall not perform any additional operations on the received frame.

If there are no previous security errors in the processing of the frame, the DEV shall apply the operations defined by the symmetric key security operations to the frame, see Table 10.3.2. If any of the security operations fail, the MLME shall return an MLME-SECURITY-ERROR.indication to the FCSL ~~DME~~ with the ReasonCode set to FAILED-SECURITY-CHECK and shall not perform any additional operations on the received frame. If the security operations have been successfully performed and the frame has been modified appropriately, the DEV may then continue to process the frame.

3. ASIE update

6.3.11 Application specific data management

These primitives are used to request that the PNC add or remove application specific data to the beacon and to report the reception of application specific data in a beacon as defined in 8.14.1. The parameters used for these primitives are defined in Table 3n.

6.3.11.1 MLME-APPLICATION-SPECIFIC.request

This primitive is used to request to add application specific data to the piconet beacon or to remove previously added application specific data from the beacon. The semantics of this primitive are:

```
MLME-APPLICATION-SPECIFIC.request(
                                    RequestType,
                                    ASIERequestID,
                                    ASIEIndex,
                                    VendorOUI,
                                    ApplicationDataLength,
                                    ApplicationData
                                    )
```

The primitive parameters are defined in Table 3n.

Table 3n—MLME-APPLICATION-SPECIFIC primitive parameters

Name	Type	Valid range	Description
RequestType	Enumeration	ADD, REMOVE	If ADD, a request that a new ASIE be placed in the beacon. If REMOVE, a request that a previously added ASIE be removed from the beacon.
ASIEIndex	Octet	0-255	An ID assigned by the PNC to each ASIE successfully added to the beacon. If RequestType is REMOVE, specifies the the ASIE to remove from the beacon.
ASIERequestID	Octet	0-255	A unique assigned by the requesting DEV to identify the request.
VendorOUI	As defined in 7.4.7.	As defined in 7.4.7.	If RequestType is ADD, the vendor OUI for the ApplicationData as defined in 7.4.7.
ApplicationDataLength	Integer	As defined in 8.14.1.	If RequestType is ADD, the length of the ApplicationData to add to the beacon in octets.
ApplicationData	Octet string	Any valid octet string of length up to ApplicationDataLength	If the RequestType is ADD, the application specific data to add to the beacon as defined in 8.14.1.
NumApplicationSpecificData	Integer	0-255	Indicates the number of ApplicationSpecificData in the piconet beacon.
ApplicationSpecificDataSet	Set of application specific data as defined in Table 3d	A set containing zero or more instances of a ApplicationSpecificData	The ApplicationSpecificDataSet is returned to indicate the ApplicationSpecificData in the piconet beacon.
ResultCode	Enumeration	SUCCESS, FAILURE	Indicates the result of the MLME request.
ReasonCode	Enumeration	REQUEST_TIMEOUT, NOT_ASSOCIATED, UNKNOWN_ID, PNC_DENIED, OTHER	Indicates the reason for a ResultCode of FAILURE.

6.3.11.2 MLME-APPLICATION-SPECIFIC.confirm

This primitive is used report the result of the request to add application specific data to the piconet beacon or to remove previously added application specific data from the beacon. The semantics of this primitive are:

```
MLME-APPLICATION-SPECIFIC.confirm(
                                    ResultCode,
                                    ASIERequestID
                                    ASIEIndex,
                                    ReasonCode
                                    )
```

The primitive parameters are defined in Table 3n.

6.3.11.3 MLME-APPLICATION-SPECIFIC.indication

This primitive is used to indicate a change in the application specific data available in the beacon. The semantics of this primitive are:

```
MLME-APPLICATION-SPECIFIC.indication(
                                    NumApplicationSpecificData,
                                    ApplicationSpecificDataSet
                                    )
```

The primitive parameters are defined in Table 3n.

7.4.7 Application specific

Replace Figure 34 with the following:.

octets: L_n	**1**	**3**	**1**	**1**
Application specific data	Application specific ID	Vendor OUI	Length (=4+L_n)	Element ID

Figure 34—Application specific information element format

Change the second paragraph as shown:

The Vendor OUI field is the OUI assigned by the IEEE standards association registration authority committee (RAC), which shall be the sole registration authority. ~~A value of vendor OUI not understood by a receiving DEV causes the remainder of this IE to be ignored.~~ If a value of the Vendor OUI field is not understood by the receiving DEV, that DEV shall ignore the remainder of the associated ASIE.

Now Request Type is ASIE Request ID, change Application Data ID to be ASIE Index. There are two numbers, the ASIE Request ID and the Application Data ID, used in a similar manner as the Stream Request ID and the Stream Index.

7.5.9.3 ASIE request

The ASIE request command is used to send an ASIE to the PNC to be put in the beacon. The DestID shall be set to the PNCID. The ASIE request command shall be formatted as illustrated in Figure 90a.

octets: n	1	1	2	2
ASIE	ASIE Index	ASIE Request ID	Length (=2+n)	Command type

Figure 90a—ASIE request command format

The ASIE Request ID field is non-zero identifier generated by the originating DEV that is unique among the DEV's ASIE requests and is used to uniquely identify the DEV's request.

The ASIE Index field is used to identify the ASIE for the request. In the case where the DEV is requesting the addition of a new ASIE to the beacon, it is set to 0xFE, by the originating DEV. In the case where the DEV is requesting the modification of an existing ASIE, it is set to the ASIE index assigned by the PNC.

The ASIE field is defined in 7.4.7. If the length of Application Specific Data in the ASIE is zero, then the DEV is requesting that the ASIE be removed from the beacon.

7.5.9.4 ASIE response

The ASIE request command is used to send an ASIE to the PNC to be put in the beacon. The DestID shall be set to the PNCID. The ASIE request command shall be formatted as illustrated in Figure 90b.

octets: 1	1	1	2	2
Reason code	ASIE Index	ASIE Request ID	Length (=3)	Command type

Figure 90b—ASIE response command format

The ASIE Request ID field contains the value 7.5.9.3.

The ASIE Index field is define in 7.5.9.3.

The valid values of the Reason Code field are:

0 -> Success
1 -> Request rejected
2-254 -> Reserved
255 -> Other failure

8.14.1 ASIE operation

The ASIE is used to implement out of scope features that require additional functionality by both the PNC and one or more of its piconet member DEVs. The additional functionality is defined as an enhancement that does not violate the standard and allows DEVs that do not have the functionality to operate normally. The Application Specific Data field in this IE provides the messages that are only interpreted by the targeted DEV.

Multiple ASIEs may be placed in the beacon by the PNC. The designer should minimize the size of each ASIE used to support the custom application.

The DEV sends the ASIE Request command with the ASIE Index field set to 0xFE to the PNC to place an ASIE in the beacons, as shown in Figure 147a. The DEV and PNC use the Application Specific ID to identify the ASIE to that its to be added to the beacon. If the PNC allows the request to place the ASIE in the beacon, it shall respond with an ASIE Response command to the originating DEV with the Request Type set to “add”, the Application Specific ID set to the same value as in ASIE Request command and the Reason Code set ot “success.” If the PNC refuses the request to place the ASIE in the beacon, it shall send an ASIE Response command to the originating DEV with the Request Type set to “add”, the Appplication Specific ID set to the same value as in the ASIE Request command and the Reason Code set to the appropriate value,

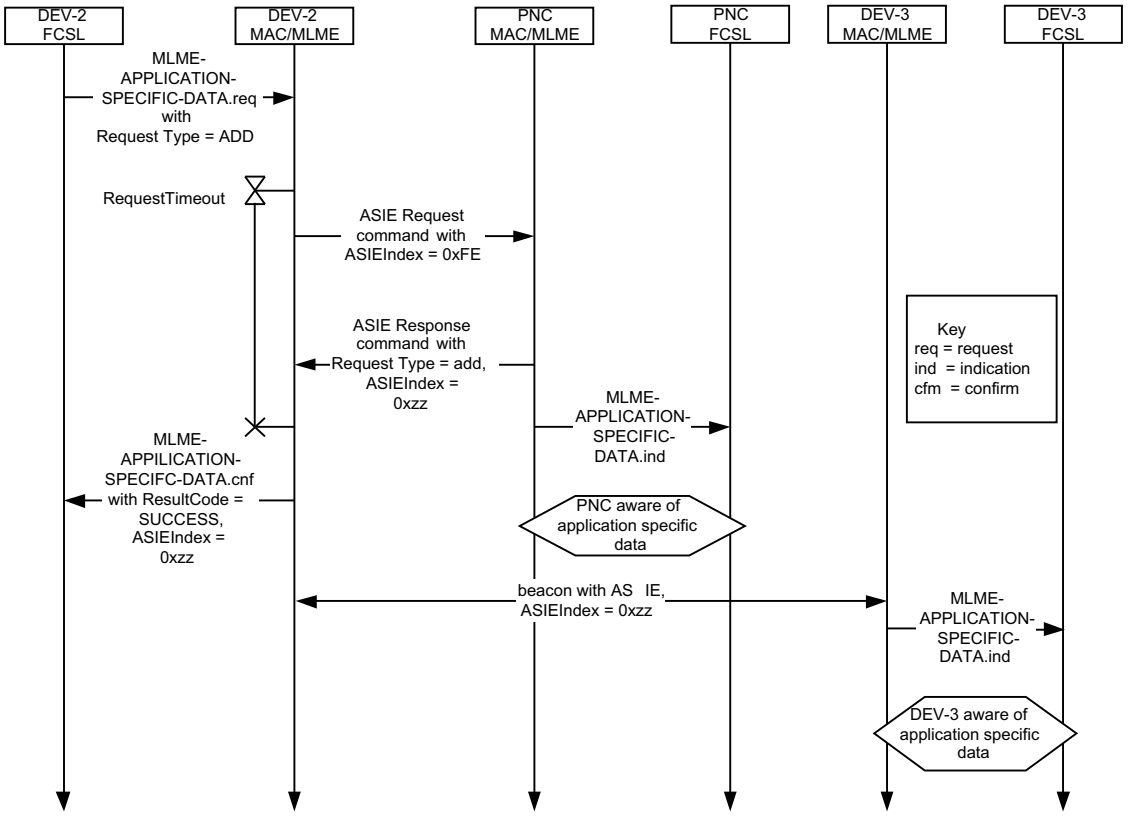

Figure 147a—Message sequence chart for a DEV adding an ASIE to the beacon.

The PNC FCSL requests the addition of an ASIE to beacon using the same MLME primitive, however, in this case, the primitive does not result in a command being sent over the medium.

A DEV may also request that the PNC remove the ASIE from the beacon by sending the ASIE Request command to the PNC with the ASIE Index field set to value for the ASIE that is to be removed and a zero length

Application Specific Data field in the ASIE. If the PNC receives this command, it shall no longer place that ASIE in the beacon. This procedure is illustrated in Figure 147b.

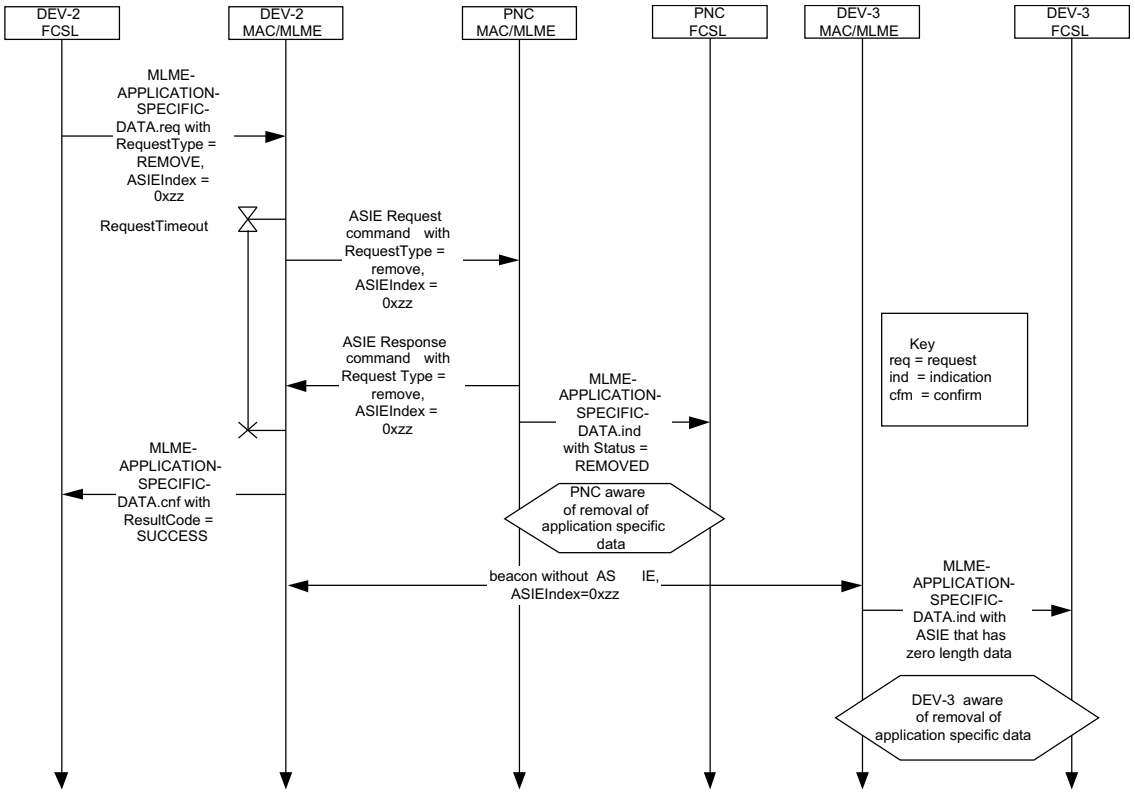

Figure 147b—Message sequence chart for a DEV removing an ASIE from a beacons

The MLME-APPLICATION-SPECIFIC-DATA.indication is used by the MLME of the DEV addressed in the ASIE, to pass the ASIE data up to the FCSL. The MLME shall pass the data to the DME only once for each ASIE that it receives. Since each ASIE may persist in multiple beacons, this persistence shall be detected in the DEV MLME.