New Specification of Current 802 LLC

Document Number: maint-Marks-llc-spec-0420-copyright

Date Submitted:
2020-04-21

Source:
Roger B. Marks
EthAirNet Associates
404 Montview Blvd
Denver, CO 80207 USA

Voice: +1 802 227 2253
E-mail: roger@ethair.net

Re: 802.1 Maintenance, related to IEEE Std 802-2014, IEEE Std 802.1AC-2016, and IEEE Std 802.1Q-2018
Venue:
802.1 Maintenance TG

Abstract
This document proposes a detailed description of LLC, as commonly implemented in 802, to replace 802.2. It supports only the protocol multiplexing function of 802.2 and also supports protocol multiplexing based on Ethertype, providing a detailed specification of LPD and EPD and including the architectural role of VLAN tagging. The results are not intended to provide any novel protocols but instead simply to specify architecture and terminology in accordance with current usage. This contribution is a followup to maint-Marks-hlpde-0919-redacted.pdf, maint-Marks-epd-lpd-0719-v02, and maint-Marks-hlpde-spec-0420-copyright-v01.

Note:
This document represents the current views of the author only and is offered as a basis for discussion. More development is needed.

The author appreciates the valuable contributions of Norm Finn and Mick Seaman on clarifying protocol identification.

Copyright information

Per IEEE copyright policy:

- this document is “Previously Published”
- the contributor is fulfilling his responsibility to immediately inform the WG Chair that the contribution requires permission from copyright owner(s) and cannot be presented or included in the draft until that permission is granted, and offering to assist the WG Chair in requesting the permission, if possible
- the WG Chair is responsible to use the IEEE-SA Permission Request and Response Form Templates to request permission (http://standards.ieee.org/develop/stdsreview.html)

New Specification of Current 802 LLC

Roger B. Marks
EthAirNet Associates
2020-04-21
IEEE 802.1 Maintenance TG

Summary

- A prior contribution <maint-Marks-hlpde-0919-redacted.pdf> said:
  - *The de facto LLC is the HLPDE. The root of the problem is that the HLPDE is not specified. It should possible to specify the HLPDE for clarification, without altering current understanding of the expected operation.*
- maint-Marks-hlpde-spec-0420-copyright-v01 proposed to specify HLPDE so that HLPDE and 802.2 comprise the LLC.
- Contributors have noted defects with 802.2 and the obsolescence of the standard; however, 802.2 cannot be maintained since the standard was withdrawn.
- Instead of specifying specify HLPDE, this contribution proposes to delete the reference to 802.2 and instead provide a specification of the LLC, as currently understood in 802, that incorporates only the protocol multiplexing function of 802.2 and also supports protocol multiplexing based on Ethertype, including a detailed specification of LPD and EPD.
- The results are not intended to provide a novel protocol but instead simply to specify architecture and terminology in accordance with current usage.

IEEE 802 LLC

- LLC is a core concept in IEEE 802
  - Per IEEE Std 802, DLC has two sublayers: LLC and MAC
  - LLC is the client of the MAC service
- LLC is not specified in IEEE Std 802
  - it vaguely discusses a “higher layer protocol discrimination entity”
  - examples show components of MAC frames without naming their origins
  - frame examples are complicated because they aggregate all layers
- IEEE Std 802 has one mention of IEEE Std 802.2 (“Logical link control”):
  - *IEEE Std 802.2™-1989 (reaffirmed 2003) was administratively withdrawn as an IEEE standard on 11 January 2011 in deference to the stabilized standard ISO/IEC 8802-2:1998 where the same material continues to be available.*

- The MAC standards refer to LLC
  - 802.3: “The services provided by the MAC sublayer allow the local MAC client entity to exchange LLC data units with peer LLC sublayer entities.”
- The LLC data unit (MSDU), per 802.3, is not a plain opaque packet; instead, it is presumed to have structure. This is not documented in IEEE Std 802 (nor 802.2).
  - 802.11 also specifies assumptions about the MSDU format
  - But we have no LLC specification of how that MSDU is formed

- Many (most) aspects of 802.2 are not currently implemented.
- A withdrawn standard cannot be revised or amended.

802.2 Functionality

- 802.2 supports three LLC types
  - Type 1: connectionless and unacknowledged
  - Type 2: connection-oriented
  - Type 3: connectionless but acknowledged
- This LLC specification supports only Type 1 multiplexing function
  - does not support Type 2 or 3
- 802.2 supports four “classes” of LLC operation
- This LLC specification supports only Class I
  - Class I is “data-link connectionless-mode”
- Per the 802.2 Type 1 description:
  - *There exists for each MAC service access point one and only one LLC entity, consisting of the various operating components.*
  - *In Class I LLC operation, each LLC can have zero or more SAPs being serviced (i.e., active) at any one time, independent of each other, which are differentiated by the DSAP address.*

HLPDE per IEEE Std 802.2014

The LLC sublayer contains a variety of entities, as illustrated in Figure 6.

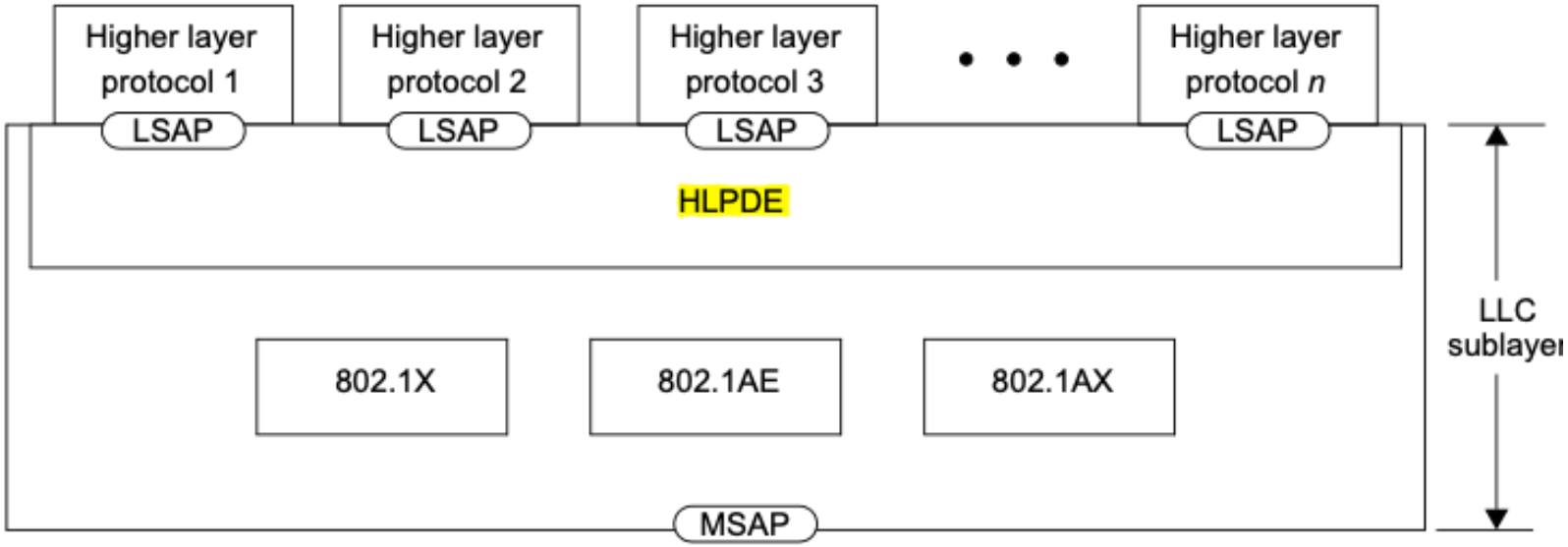


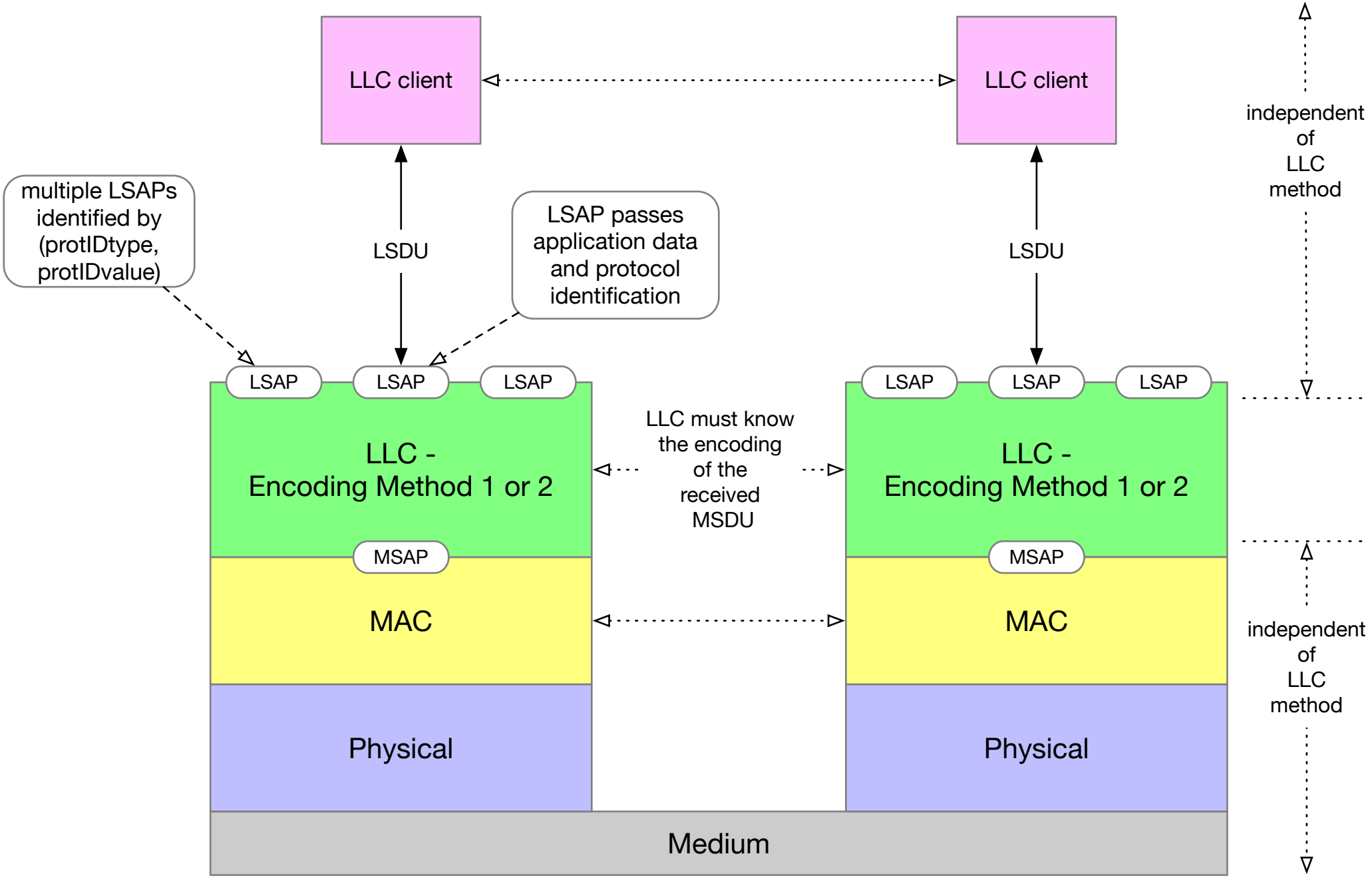
Figure 6—LLC sublayer in 802 RM

IEEE Std 802-2014:
The higher layer protocol discrimination entity (HLPDE) is used by the LLC sublayer to determine the higher layer protocol to which to deliver an LLC sublayer protocol data unit (PDU). Two methods may be used in the HLPDE. The two methods are:

1) EtherType protocol discrimination (EPD), which uses the EtherType value made available to the LLC sublayer through the MSAP
2) LLC protocol discrimination (LPD), which uses the addresses defined in ISO/IEC 8802-2, including the Subnetwork Access Protocol (SNAP) format

- As discussed in prior contributions, the descriptions of EPD and LPD in IEEE Std 802-2014, 802.1AC, and 802.1Q are inconsistent.
- Since EPD and LPD are HLDPE methods, this contribution seeks to resolve the issues by developing a detailed specification of the LLC incorporating higher layer protocol discrimination.

New LLC Spec in the 802 Architecture

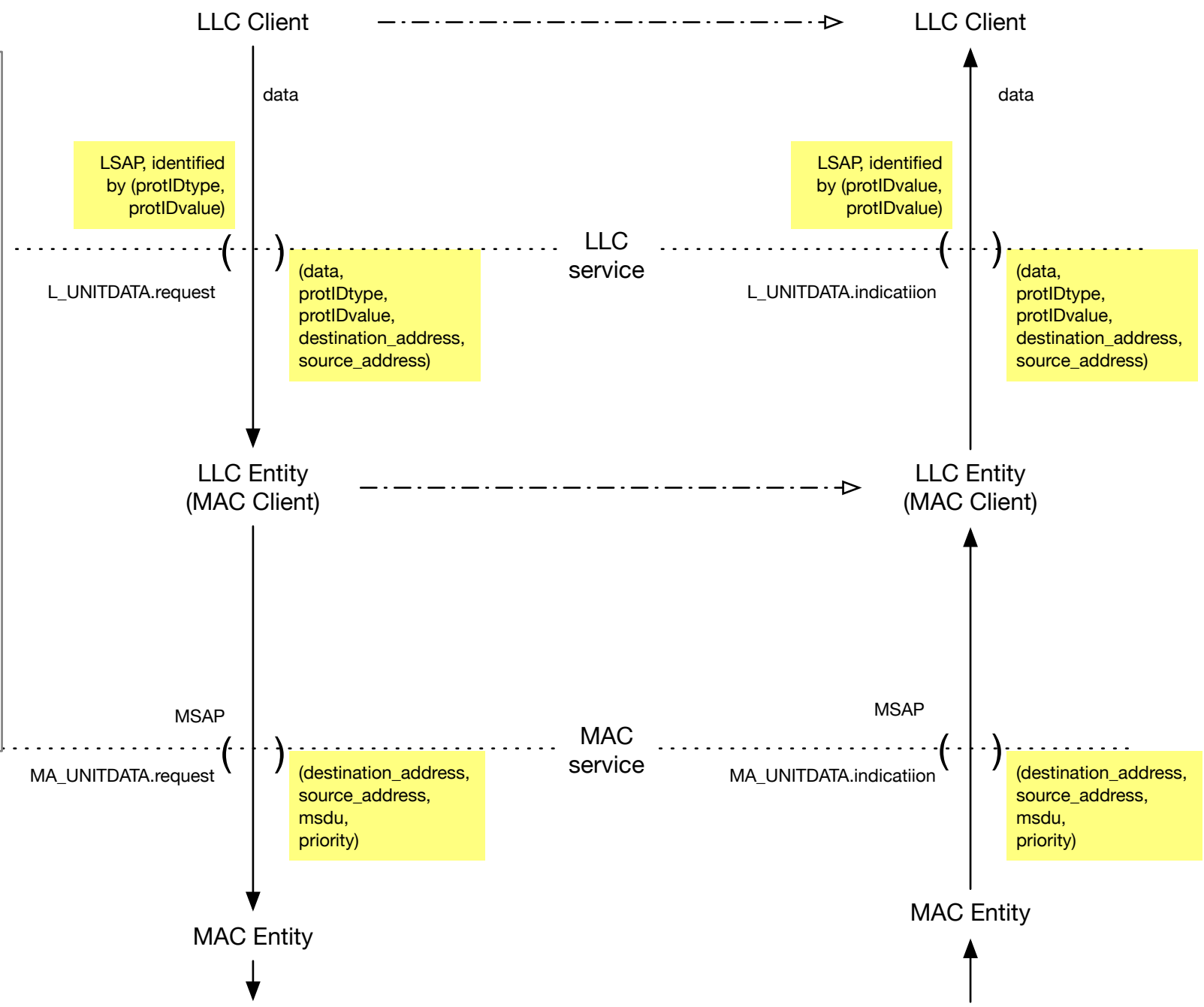


LSAP Service Primitives in New LLC Spec

Analogous 802.2 primitive is DL-UNITDATA request (source_address, destination_address, data, priority)

Per 802.2:
The "source_address" and "destination_address" parameters provide at a minimum the logical concatenation of the MAC address field (SA and/or DA) and the LLC address field (SSAP and/or DSAP).

That is, it's not the job of the LLC to determine the MAC DA and SA; that needs to be passed down from LLC Client.

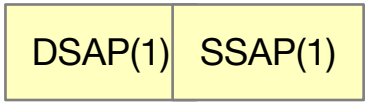
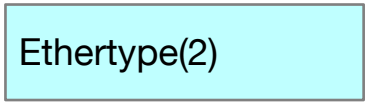
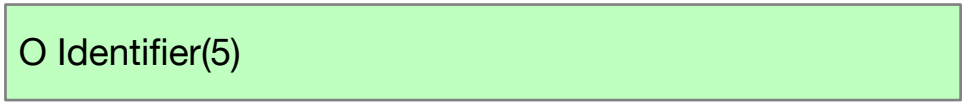


LLC Operation

- LSAP is identified by (protIDtype, protIDvalue)
- LSAP passes (data, protIDtype, protIDvalue), plus MAC DA/SA
- LLC encodes (data, protIDtype, protIDvalue) into the MSDU
- LLC sends MSDU to MSAP, along with MAC DA/SA
- Peer LLC, from MSAP, receives MSDU with MAC DA/SA
- From MSDU, LLC determines (data, protIDtype, protIDvalue)
    - then forwards data (& MAC DA/SA) to LSAP identified by (protIDtype, protIDvalue)

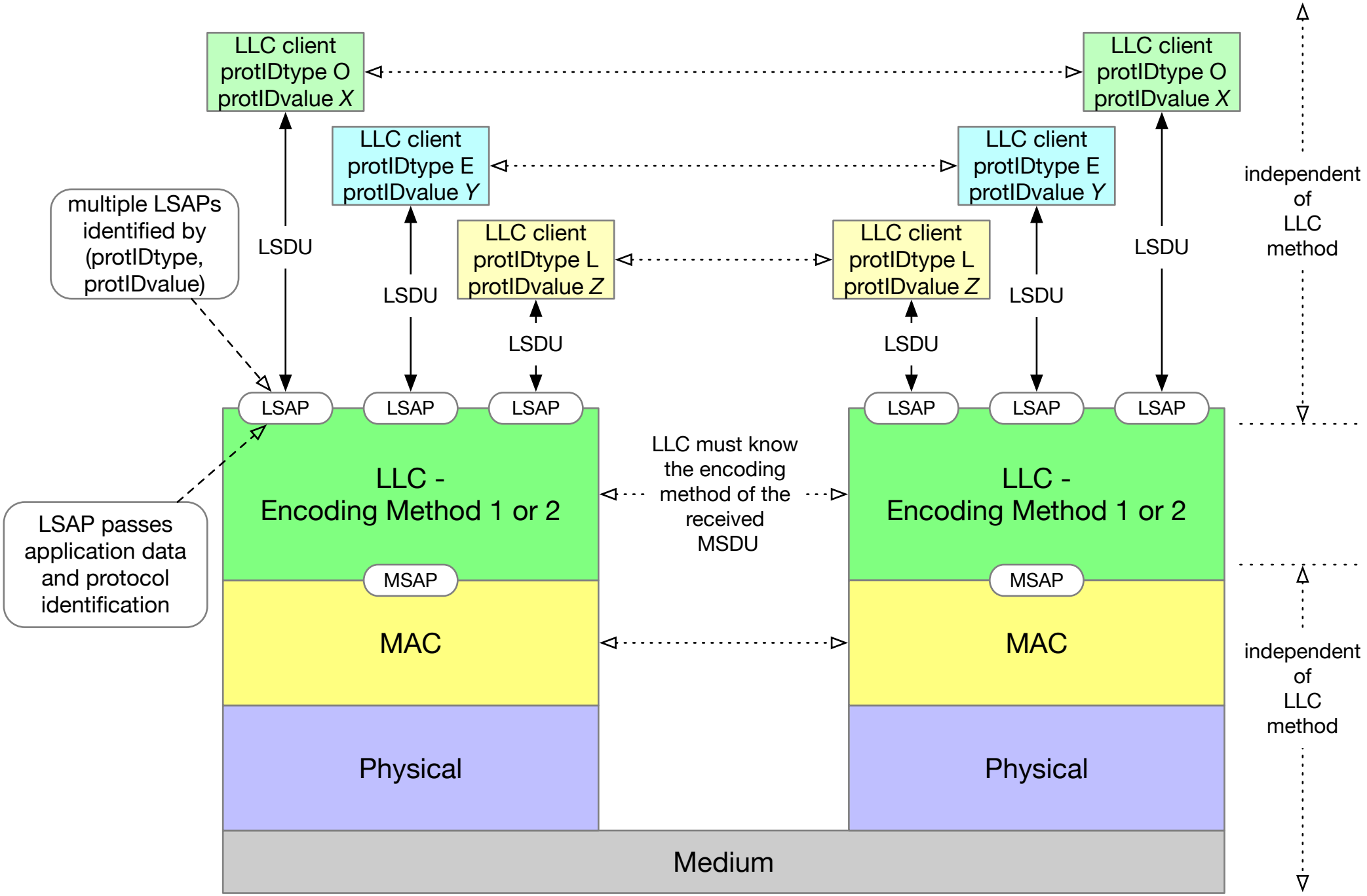
protIDtypes and protIDvalue formats

The protocol is identified at the LSAP using:
(a) protIDtype (either “L”, “E”, or “O”)
(b) protIDvalue, using a specific format for each protIDtype:

protIDtype	**protocol identifier format (PIF)**	
L	LSAP	
E	EtherType	
O	OUI Extended	

Note: The OUI/CID Extended identifier is specified in IEEE Std 802 to begin with a registered OUI, OUI-36, or CID, with additional bits specified by the assignee of that registered identifier to uniquely identify the protocol.
For LLC purposes, the structure of the O Identifier is irrelevant.

LLC Clients match by (protIDtype, protIDvalue)



LLC Encoding Methods

•Two LLC Encoding Methods are supported:
  -Method 1: EPD (Length/Type)
  -Method 2: LPD (LSAP/SNAP)

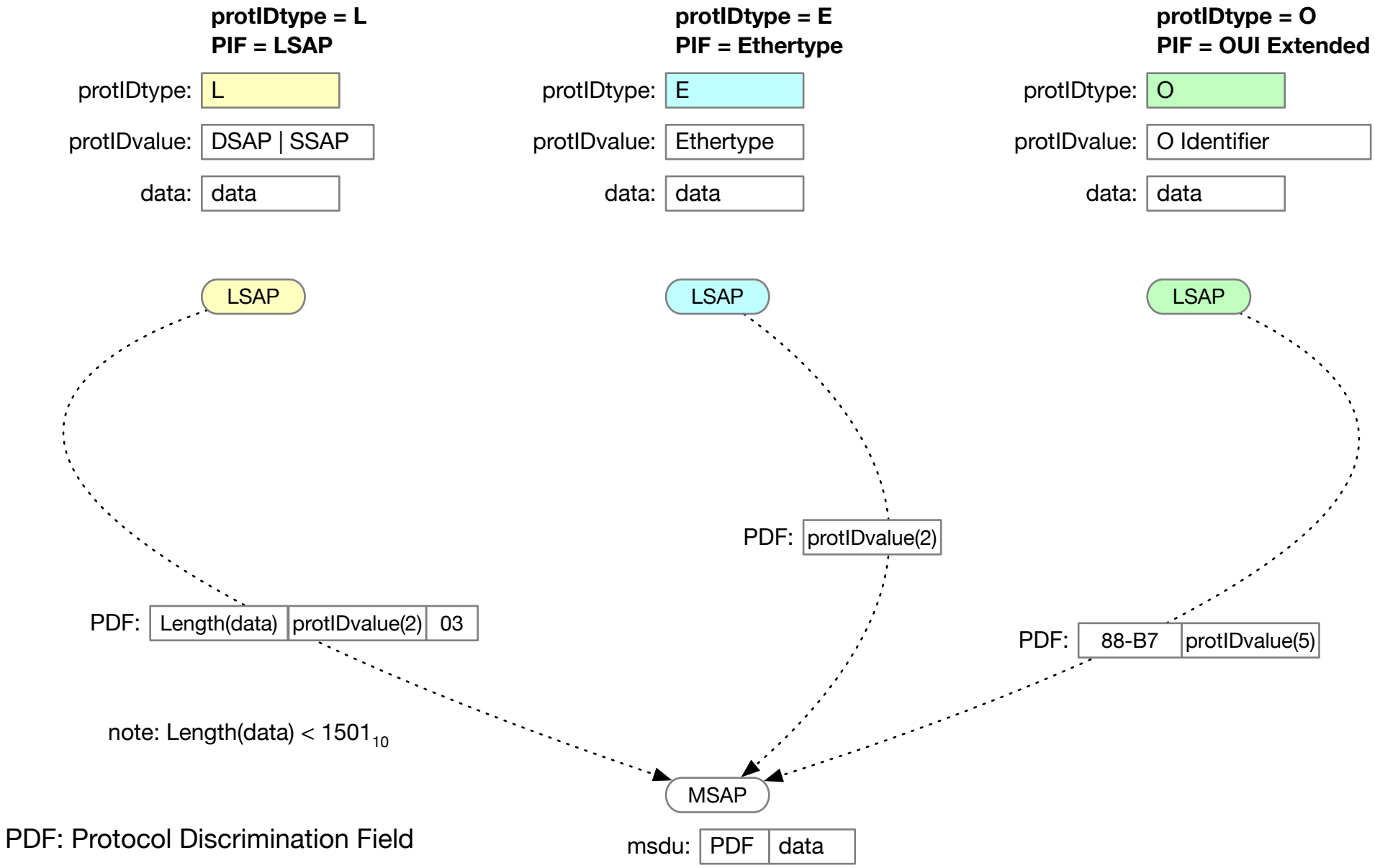
•The two Methods are different and incompatible, so the receiver LLC must know the LLC encoding of the received frame in order to be able to decode it.

•Both Methods support all three protIDtypes.

•The client (LSAP and up) and the MAC (MSAP and down) are Method-independent
  -note: therefore there is, in general, no “EPD MAC” or “EPD medium” (nor LPD)
  -however, a MAC specification may specify which method it presumes used or how the method may be identified at a station

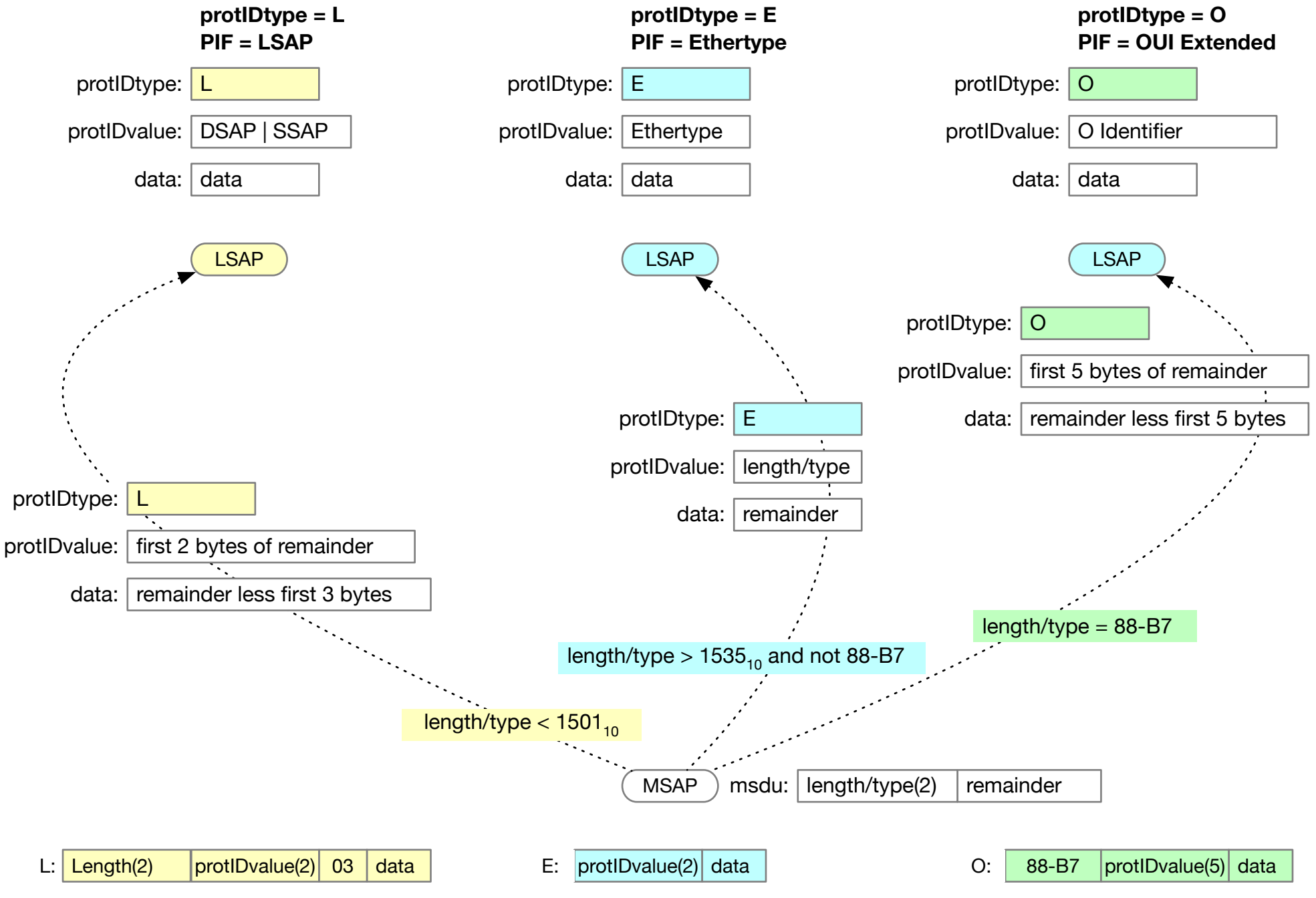
encoding formats	**protIDtype: L**	**protIDtype: E**	**protIDtype: O**
Method 1: Length/Type (EPD)	Length	Type	Type
Method 2: LLC/SNAP (LPD)	LSAP	SNAP	SNAP

LLC Method 1 (EPD) Encoding

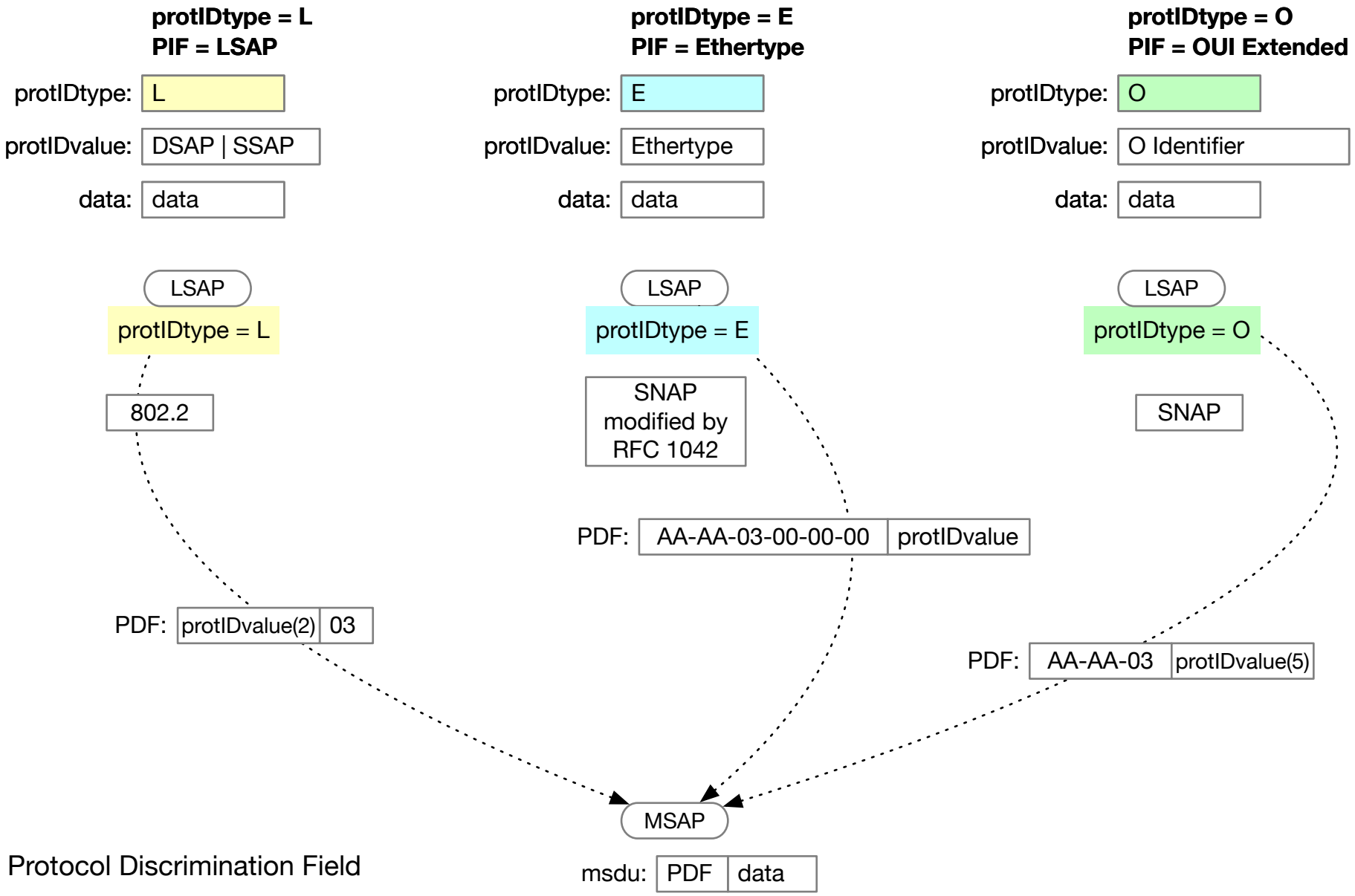


PDF: Protocol Discrimination Field

LLC Method 1 (EPD) Decoding

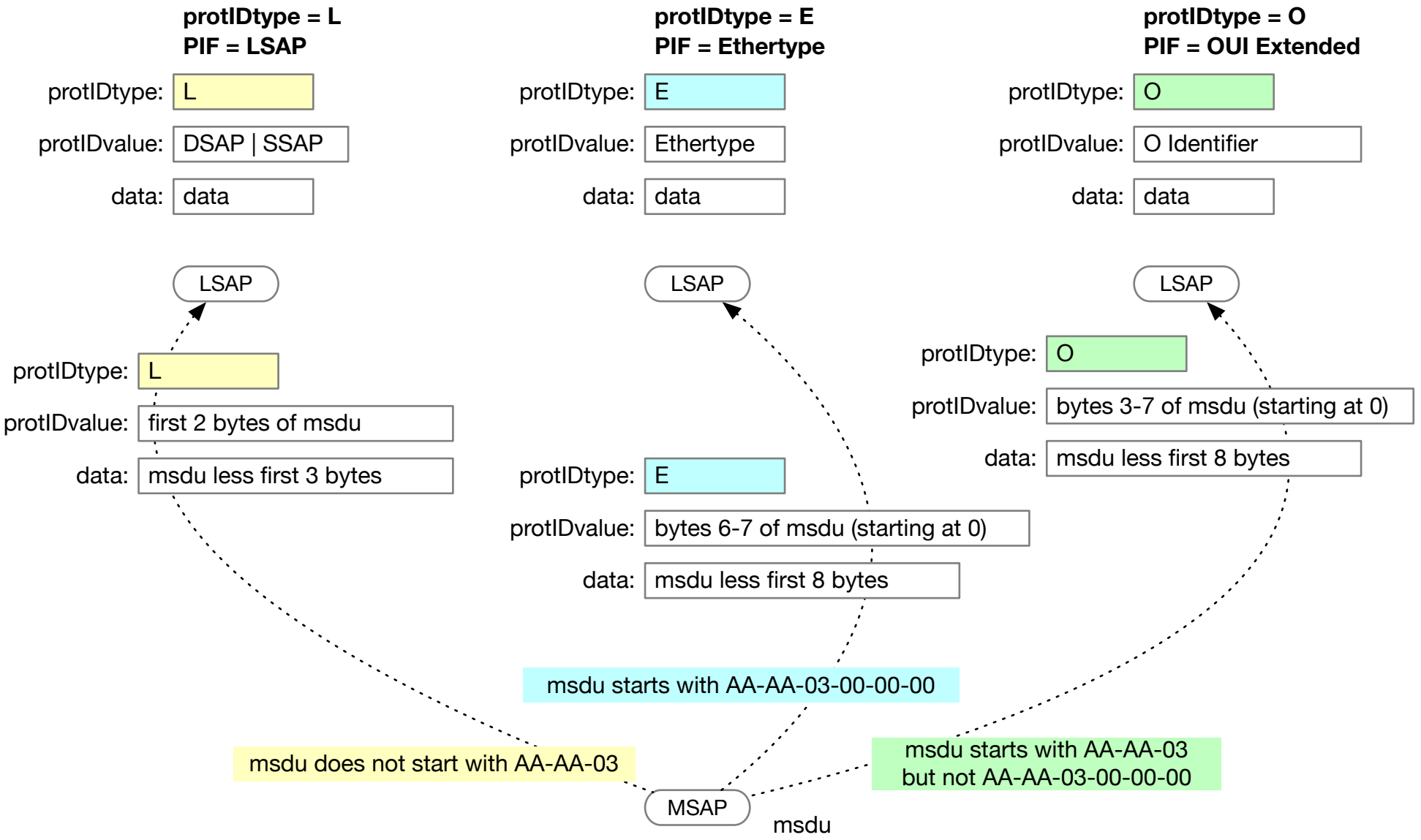


LLC Method 2 (LPD) Encoding



PDF: Protocol Discrimination Field

LLC Method 2 (LPD) Decoding



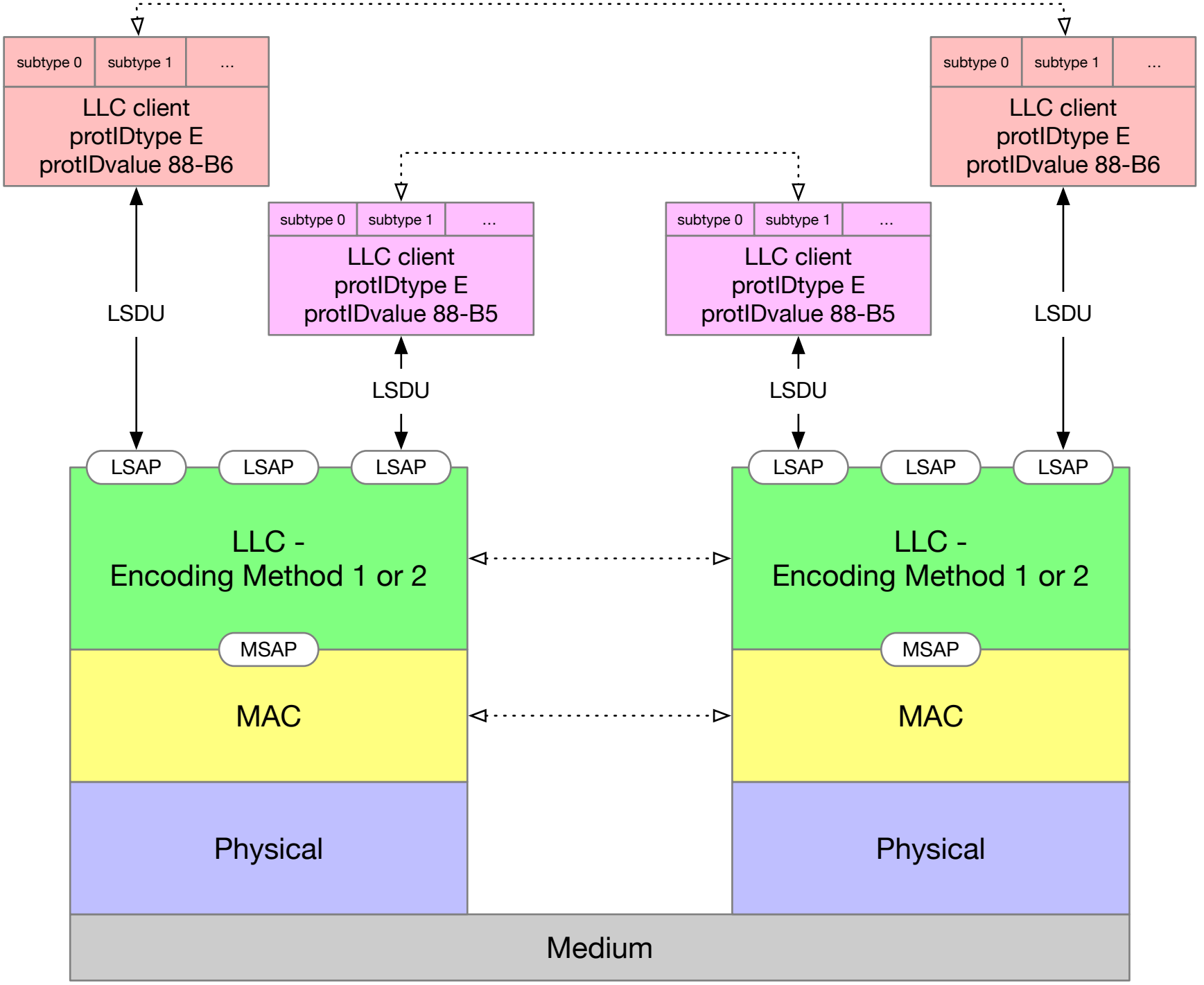
Playpen EtherType

Local Experimental EtherType (Playpen EtherType):

- Introduced in IEEE Std 802a-2003 (“Ethertypes for prototype and vendor-specific protocol development”)
- specified in IEEE Std 802 with a subtype and version, of unspecified length, that are locally interpreted, not resolved in the LLC
- LLC directs the frame to the LSAP identified by 88-B5 or 88-B6
- The locally-specified application at that LSAP is responsible to take any appropriate actions corresponding to the local protocol subtype and protocol version fields.
- This is fully consistent with the proposed LLC architecture, without modification.

Local Experimental EtherType 1:	88-B5	protocol subtype	protocol version	data
Local Experimental EtherType 2:	88-B6	protocol subtype	protocol version	data

Playpen Ethertype



LLC encapsulation EtherType

LLC encapsulation EtherType is specified in IEEE Std 802.1AC with the value C9-D1, to indicate that the protocol carried is LPD.

As specified in IEEE Std 802.1AC, the LLC encapsulation EtherType is used only to support, in the EPD LLC, protIDtype=L frames with length > 1500_{10}, which is otherwise forbidden.

Notes:

•This effectively isolates LPD and EPD and could directly solve the problem of format identification, if all protIDtype=L frames were encoded this way (but they are not).

•This can be modeled as an LPD LLC stacked above the EPD LLC.

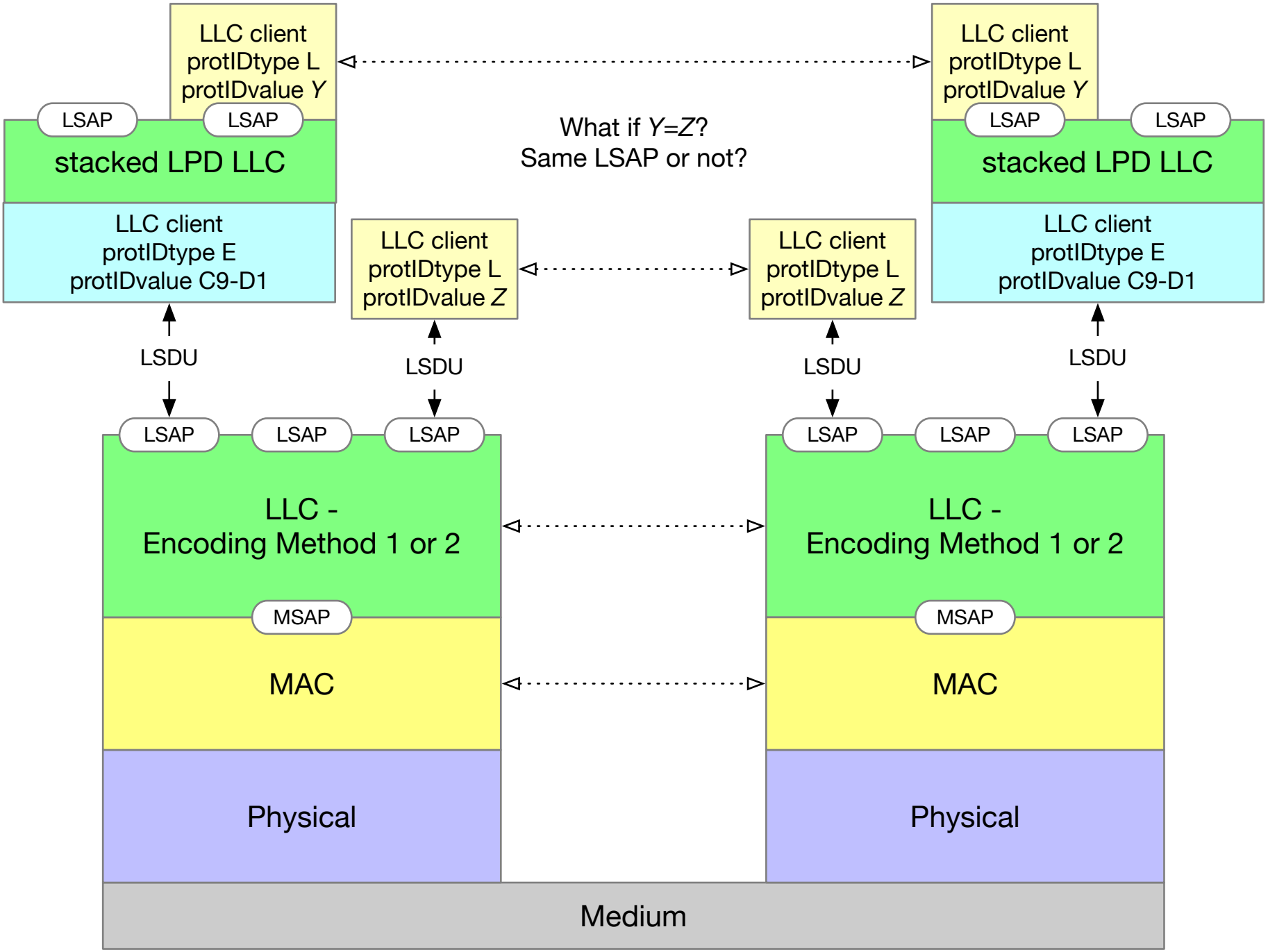
•This is an architectural modification, and an issue arises:

-Is the LSAP at protIDvalue=*Z* behind the LLC encapsulation EtherType the same LSAP as the one at the main protIDvalue=*Z?*

-Probably, yes, but it should be specified.

•Per 802.2, *There exists for each MAC service access point one and only one LLC entity,* this could be considered inconsistent.

LLC encapsulation EtherType

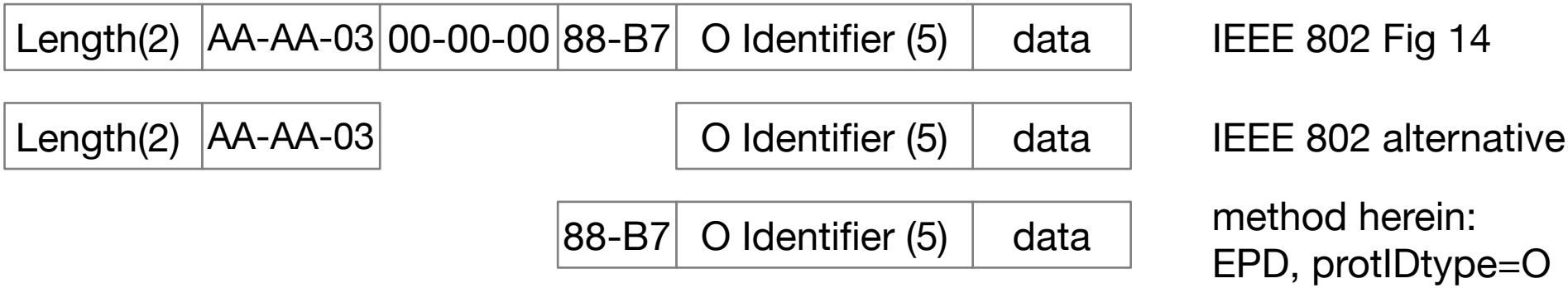


Redundant SNAP

IEEE Std 802-2014 (Fig. 14) documents a special form in which OUI Extended Ethernet format is embedded in the RFC 1042 form of SNAP in a Type/Length (EPD) frame. IEEE Std 802 says:

•*In this case, it would be more appropriate to use the SNAP identifier directly (i.e., omit the IETF RFC 1042 OUI and OUI Extended EtherType fields shown in Figure 14); however, this is a valid encoding of the OUI Extended EtherType that can result from the application of the encapsulation described in 9.4.*

However, the alternative example proposed in IEEE Std 802 is also inappropriate. The example illustrates EPD length encapsulation, but OUI Extended Ethertype is supported by an explicitly assigned Ethertype, which is directly supported by EPD type encapsulation (see IEEE Std 802 Fig. 13). In the LLC specified herein, this is exactly EPD with protIDtype=O:



SNAP is always redundant in EPD. SNAP in EPD was discouraged in IEEE Std 802.1H (ISO/IEC TR11802-5:1997), except in the case of one specific (and rather obsolete) protocol.

For LDP, a format like Fig. 14 (without the Length field) would be another example of redundant SNAP. In that case, the IEEE 802 Alternative would indeed be relevant for LPD; in fact, it is exactly the method specified herein for LPD with protIDtype=O.

The method herein specifies the encoding formats without support for redundant SNAP, which would be inconsistent with the architecture and is inefficient and unnecessary. It is possible to specify decoding to support obsolete frames with redundant SNAP, but it may not be worth putting the details into an LLC standard.

MSDU Content and Format in the MAC

•The MSDU is created by LLC, with a structure and format directed to be read by LLC.

•In principle, MAC sees MSDU as a data unit without structure or format.
-in accordance with good layering principles

•For example, IEEE Std 802.1AC says:
-"*the MAC Service provides for the transparent transfer of MAC Service user data. It does not restrict the content, format, or coding of the information, nor does it ever need to interpret its structure or meaning*."

•Therefore, the choice of LLC Encoding Method (EPD or LPD) should be MAC-independent.

•In practice, this is mostly true, but not entirely accurate.

•Let's take a closer look at 802.3.

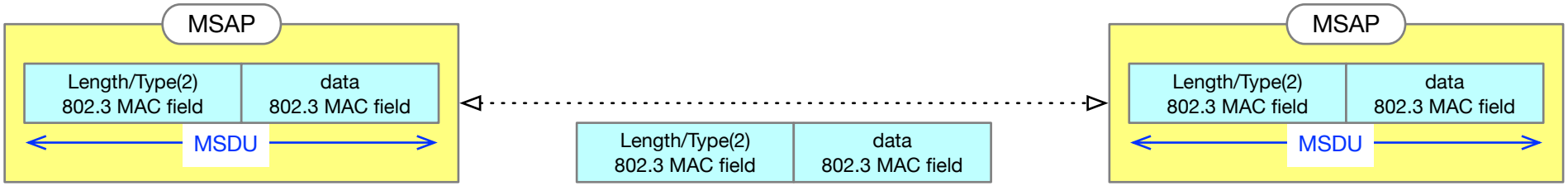
MSDU Content in the 802.3 MAC

The MAC service per IEEE 802.3 (normative subclause 4A.3.2) specifies:

-mac_service_data_unit：concatenation of the lengthOrType field and the data field parsed from the client request

-lengthOrType：The value of the first two octets at the start of the mac_service_data_unit

-data: The value of mac_service_data_unit excluding the first two octets (Length/Type field)



•Upon receipt from LLC, the MSDU is split into two MAC fields: (a) Length/Type; (b) data

•Before delivery to LLC, Length/Type and data fields are re-concatenated to form MSDU

-(these should be called out as two parameters passed over the MSAP; oh well)

•When the Length/Type is interpreted as a Type, it is not used by the 802.3 MAC, and the MSDU is delivered as received, so the structure and content of the MSDU are irrelevant.

•However, when the Length/Type is interpreted as a Length, it is used by the 802.3 MAC to remove the pad field

-802.3: “Length checking is provided for Length interpretations of the Length/Type field.”

-Otherwise, 802.3 takes no action in response to the Length/Type field.

•What if an LPD MSDU was sent to 802.3?

-If the first byte was $>1535_{10}$, it would be transferred without problem.

-If the first byte was $>1500_{10}$ and $<1536_{10}$, it could be rejected.

-If the first byte was $<1501_{10}$, then frame could be trashed (trimmed or dropped).

Theoretical: sending an LPD MSDU over Ethernet

•An LPD MSDU always begins with DSAP/SSAP
  -this may be "AA-AA"
    -AA-AA = 43960_{10} => Type interpretation
    -this may match an Ethertype, but the 802.3 MAC doesn't care; no problems arise

DSAP	SSAP	decimal value	Length/Type	DSAP Assignment*
00	00	0	Length	null address
01	00	256	Length	none
02	00	512	Length	802.1B (withdrawn)
03	00	768	Length	none
04	00	1024	Length	none
05	00	1280	Length/nulll	none
06	00	1536	Type	
07	00	00	Type	
08	00	00	Type	
>08	00	…	Type	
FF	00	65280	Type	

•An LPD MSDU can begin with two octets that look to Ethernet like a length or an invalid Type/Length field

•This happens only with *DSAP=00, 01, 02, 03, 04, 05

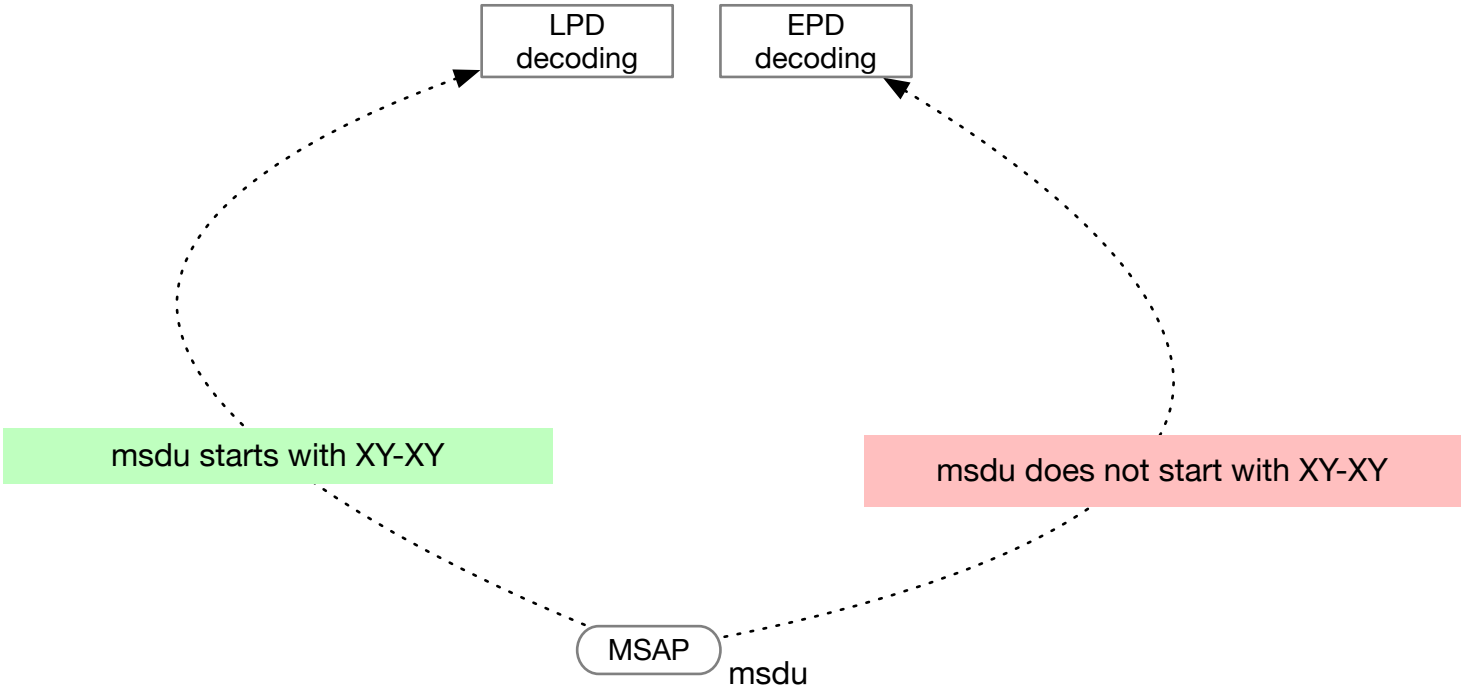
•But those DSAP values are not assigned to protocols.

•LPD works fine over Ethernet for any valid DSAP/SSAP value.

*Possible exception if DSAP=00. Per 802.2, that null address represents the MSAP and may not be used as an address of an LSAP.

* https://standards.ieee.org/products-services/regauth/llc/public.html

LLC Universal Decoding?



The approach above doesn't quite work:

•On one hand:
- -as shown, DSAP/SSAP does not overlap Length values
- -there seem to be about 17 assigned Ethertypes of the form XY-XY
  - -none overlaps an existing active DSAP/SSAP assignment.

•On the other hand, about 11 of them overlap the "unreserved" LSAP range, which is available for local use.

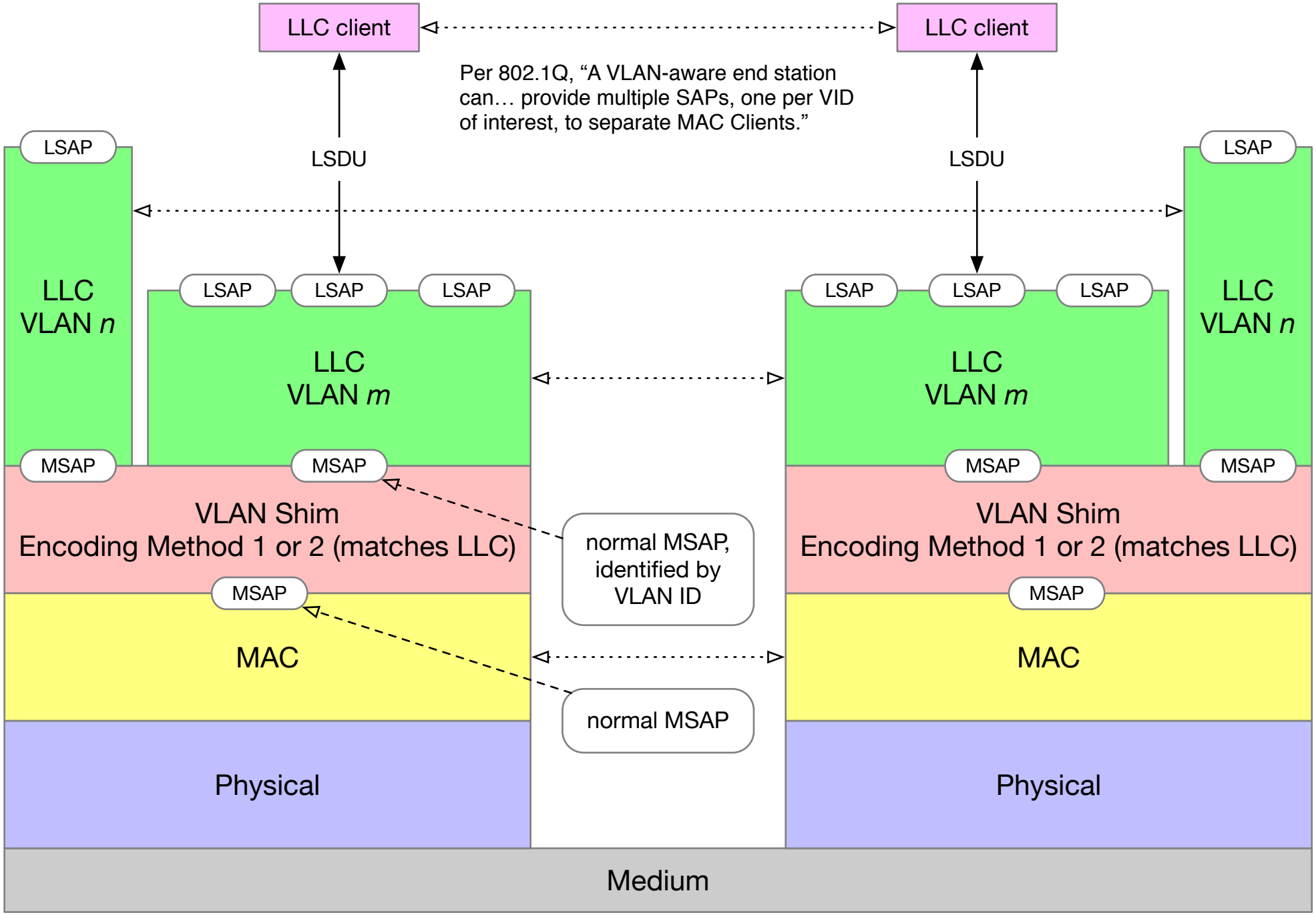
•If future Ethertype assignments continue to avoid overlap with XY-XY assignments (or if these are not supported in the implementation), and if local LSAP use of the XY-XY form is avoided, LLC Universal Decoding would be possible with this method.

•Probably not worth trying to find a standardized solution.

VLAN Tagging

- We should represent VLAN tagging in the architecture.

End Station VLAN Shim in the 802 Architecture



VLAN Tagging Methods

•Two VLAN Tagging Methods are supported:
  -Method 1: EPD (Length/Type): uses simple type-based VLAN tag
  -Method 2: LPD (LSAP/SNAP): uses SNAP-based VLAN tag

•Going down, the tag is added per the MSAP identifier.
•Going up, the tag is read and stripped; the data is passed to the identified MSAP.

•Per 802.1AC, the VLAN LPD method also changes the data's protocol identification encoding (added in the LLC sublayer) from LPD-like to EPD-like going down, reversing it going up.

•The two Methods are different and incompatible, so the receiver LLC must know the encoding of the received frame in order to be able to decode it.

•Both methods support all three protIDtypes.

encoding formats	**protIDtype: L**	**protIDtype: E**	**protIDtype: O**
Method 1: Length/Type (EPD)	Type	Type	Type
Method 2: LLC/SNAP (LPD)	SNAP	SNAP	SNAP

MSDU Translations

•It is possible to translate MSDU between LPD and EPD format.

•This can be useful when an MSDU encoded by an LPD LLC needs to be decoded by an EPD LLC, or vice versa.

•Such a translation could be useful, fro example, in a bridge in translating a frame from a network in which EPD is presumed to one in which LPD is presumed, or vice versa.

Proposals

(1) IEEE Std 802 should be thoroughly revised, with a detailed specification of the LLC that is independent of 802.2 and includes protocol discrimination functionality based on 802.2 as well as Ethertypes

(2) EPD and LPD should be specified as LLC methods so that the rest of the architecture is independent of the LLC method

-Note: EPD and LPD per IEEE Std 802 are completely different than proposed here

(3) IEEE Std 802 should clarify the architecture and detail the roles and functions of LLC and VLAN tagging

(4) IEEE Stds 802.1AC, 802.1Q, and 802.11 should be reviewed for consistency and clarified as necessary

Conclusion

- 802 is no longer reliant on the 802.2 LLC
- There is no LLC specified in active IEEE 802 standards
- LLC should be specified within the IEEE 802 family
  - Should go into a revision of IEEE Std 802
- It should possible to specify the LLC and its current role in protocol discrimination without altering the current understanding of the expected operation.
- This contribution proposes a view of the LLC specification.

Bibliography

- R. Marks, “What are EPD and LPD?”
  maint-Marks-epd-lpd-0719-v02.pdf
- R. Marks, “Clarifying EPD and LPD with an HLPDE Protocol”
  maint-Marks-hlpde-0919-redacted.pdf
- N, Finn, “Why the EPD/LPD information in IEEE 802, IEEE 802.1AC, and 802.1Q must be fixed”
  maint-finn-epd-lpd-errors-0919-v02.pdf
- R. Marks and N. Finn, “Clarifying EPD and LPD”
  maint-Marks-Finn-epd-lpd-1119-copyright.pdf
- R. Marks and N. Finn, “Fixing EPD and LPD in IEEE Std 802-2014”
  maint-Marks-802-epd-lpd-fix-0120-v02.pdf
- N. Finn, “EPD and LPD in IEEE Std 802 and Others”
  maint-finn-802-epd-lpd-0320-v01.pdf
- M. Seaman, “Protocol identification in 802 LANs”
  maint-seaman-protocol-identification-0420-v00.pdf
- R. Marks, “Proposed HLPDE Specification?”
  maint-Marks-hlpde-spec-0420-copyright-v01