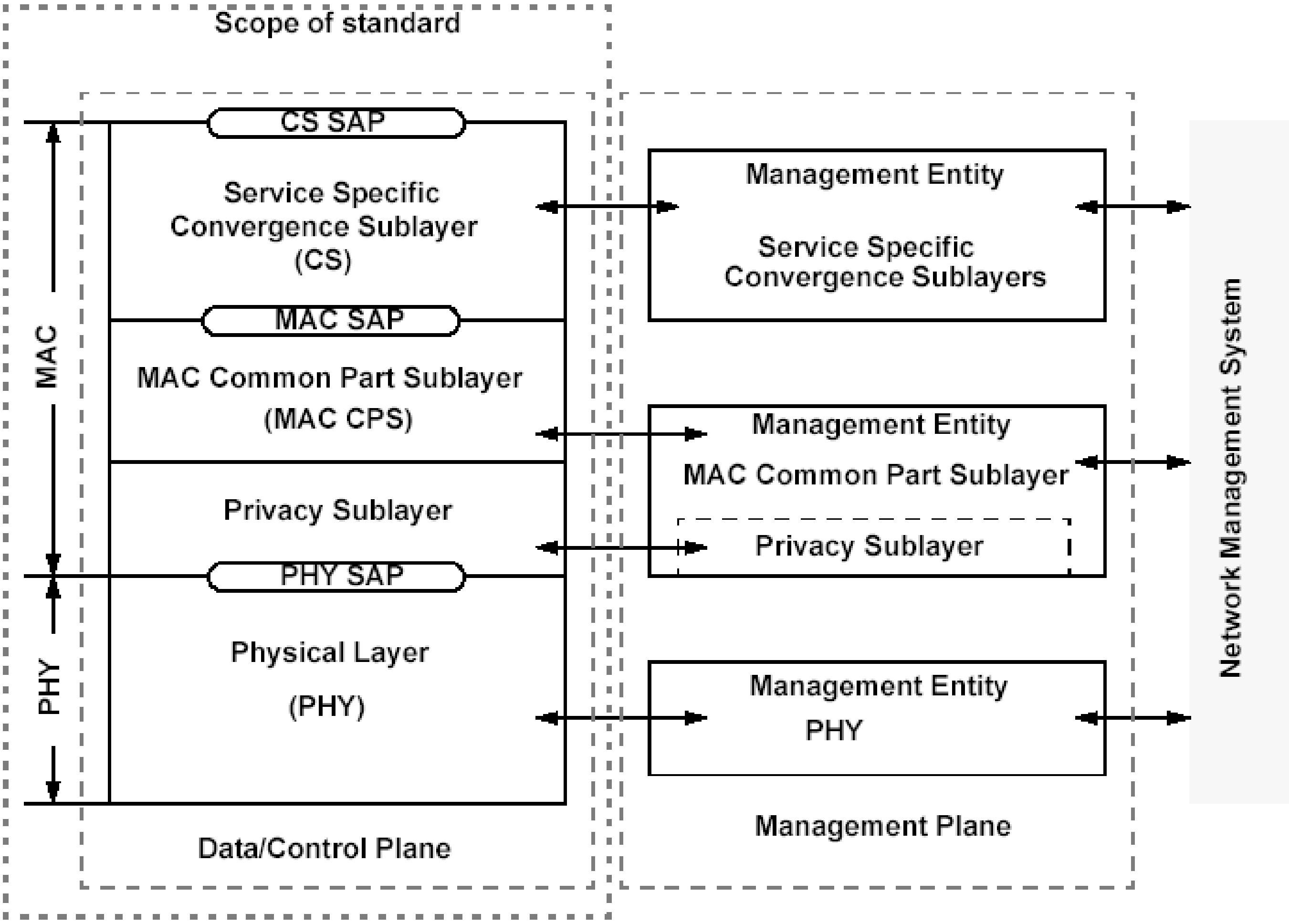
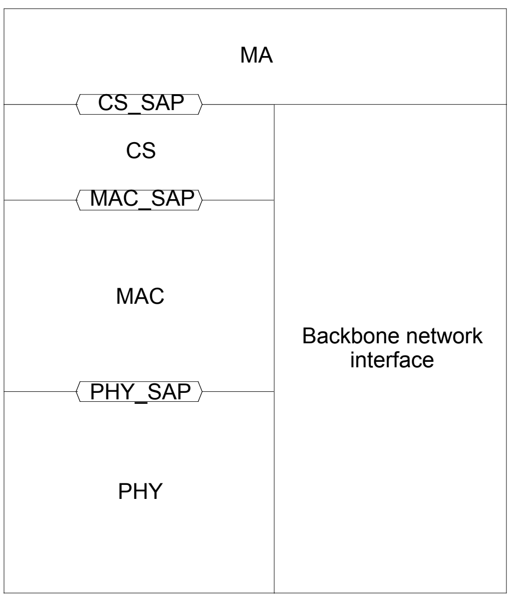
Project	**IEEE 802.16 Broadband Wireless Access Working Group <http://ieee802.org/16>**
Title	**Improved Stack Diagram**
Date Submitted	**[2004-03-13]**
Source(s)	David Johnston
Re:	Sponsor Ballot Comment against P802.16-REVd_D3 by David Johnston
Abstract	
Purpose	To furnish the stack layering diagram with the correct interfaces and scope and to define the CS SAP and the ISS correctly.
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>.

As specified today, the stacks in 802.16 and 802.16e look like this:

802.16:



802.16e



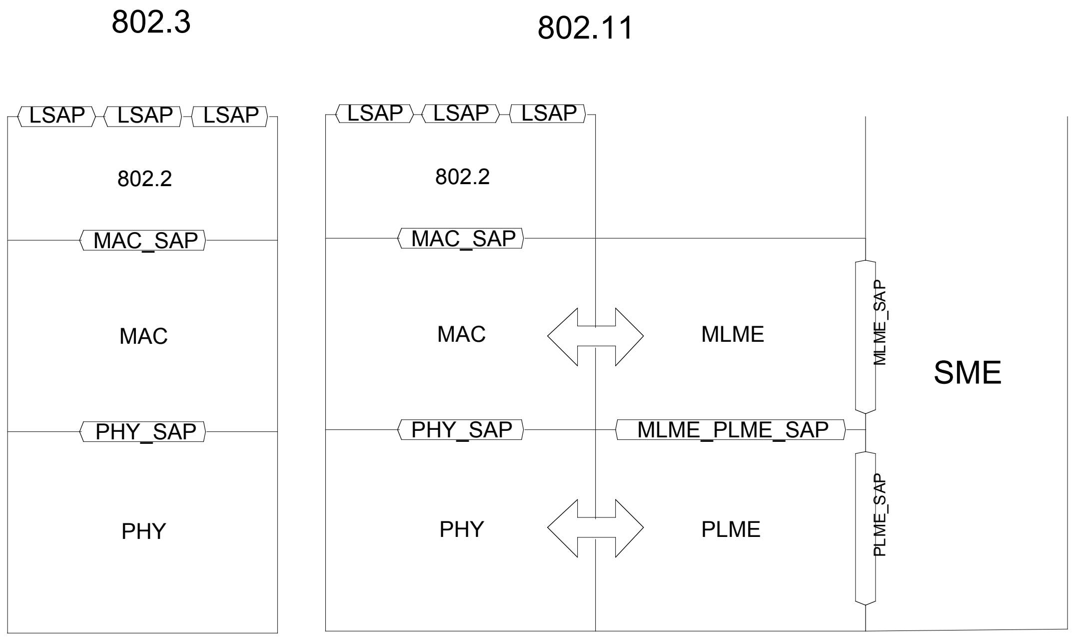
These diagrams have some interesting properties.

The CS in 802.16e does not have a CS_SAP. In both 802.16d and 802.16e, no CS_SAP primitives are defined. Instead, the packet CS specification has frames (IP, 802.1Q or 802.3) passing upwards and downwards out of and into the CS. The ATM CS does something similar with ATM connections. Each CS type and subtype defines a mapping between the frames and the MAC_SAP primitives.

In 802.16e, a CS_SAP has appeared as an interface between the CS and MA (Mobility Agent). The mobility agent is drawn as a bridging layer, between the CS and some other undefined backhaul network. There is no MA_SAP, nor is there any other type of service defined. 802.16e in this mode is either a network tunnel using the chosen CS protocol type (ATM, IP, 802.1Q or 802.3) as the encapsulating protocol.

Unfortunately, since no CS_SAP primitives are defined, so the model is incomplete.

There is a glaring difference between existing wireless 802 standards and the wired 802 standard:



The wired stack has no management plane. The PHY and MAC pretty much send one type of frame. The PHY might have some of its own special on-the-wire signaling for managing the link, but that is local between itself and peer PHYs on the wire.

Other management tasks are done over these standard frames. Ethertypes are created, to route packets through 802.2 to the correct LSAPs and management functions are bolted to the top of 802.2. E.G. EAPoL as defined in 802.1X.

What is implicit in this model is that the link takes care of itself. Just plug in the wire and the packets can fly, all dealt with locally by the MAC and the PHY. The upper layer management functions can use the packets to carry their management payloads, differentiated by ethertypes.

The wireless channel is less accommodating. There is no guarantee of connectivity. There may be multiple options on connectivity (e.g. visible APs) that require the network topology to be constrained. The security issues of RF channels require that security be established before user packets fly, not after. There is rapid mobility where either the network or the mobile might be the optimal decision maker. Handovers need to be coordinated, the network does not have the user plugging wires to tell it what the current connectivity should be.

In order to address all this new management that must be performed, the 802.11 stack, consistent with other wireless protocols, defines a management stack in parallel with the user data stack. The complexity of these management procedures is high, so the behavior of each layer is bounded by the MSAPs attaching the management layer to those above and below it. The air side protocol defines a separate class of frames, management frames, just to carry the management messaging.

A simple way of viewing this is that the management in wireless protocols is sufficiently complex that it needs its own layered stack to organize it, whereas in the wired protocols, the assumption of connectivity allows much simpler management procedures to either be thrown in the data stack, ad-hoc, or put at a higher layer.

802.16 is in an intermediate state. Is does not have a well defined management stack, but is does have the idea of management entities and management frames and two management connections to carry them, along with a few others scattered around such as ranging and BW request channels that are really management functions.

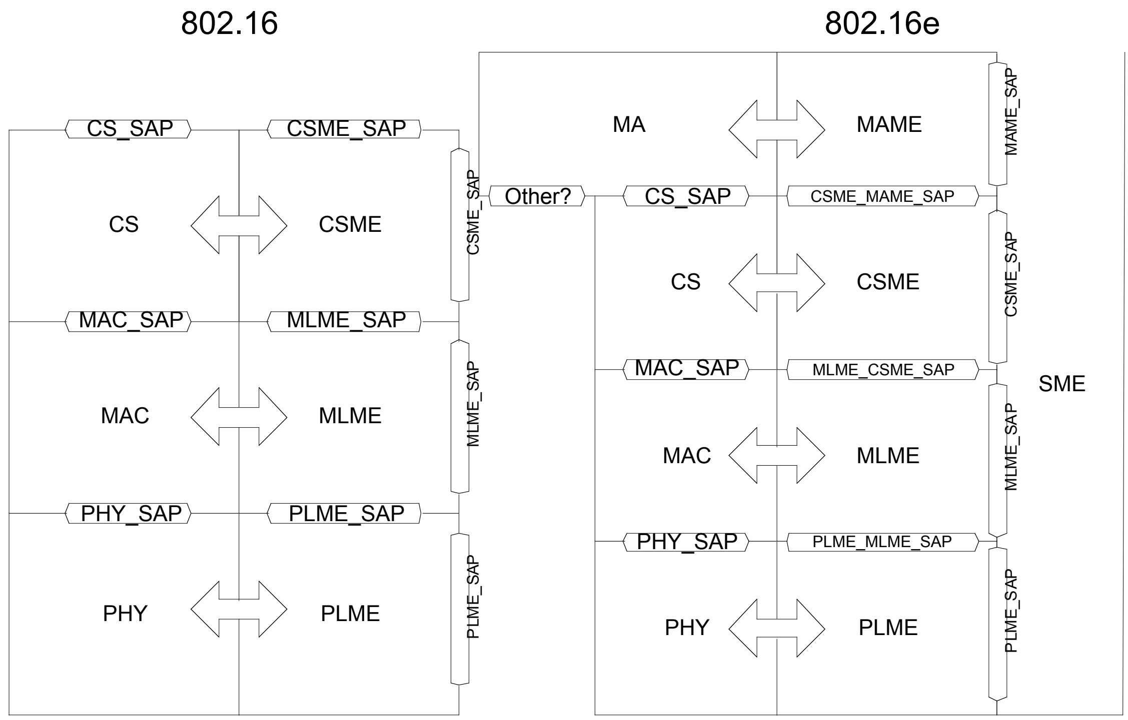
This leads to holes in the specification. E.G. we need inter-BS messages, but we have no SAP on which those messages are defined. The natural location for such a SAP would be on a management layer, but that SAP don’t exist so the primitives just float around with no clear place in the architecture.

What should we do? Well for a start, we need to ‘close’ the layers. All interaction between a layer and anything else should be through SAPs. This means a CS_SAP with defined primitives, one set for each of the CS types.

This CS_SAP would define the user services that the standard provides.

To draw from the experience of other wireless standards, we could add in a structured management stack, with its own SAPs to formalize the scope of management behavior and the way it interacts with other entities.

So for both 802.16 and 802.16e, we can naively add a management stack and interfaces and we get the following:



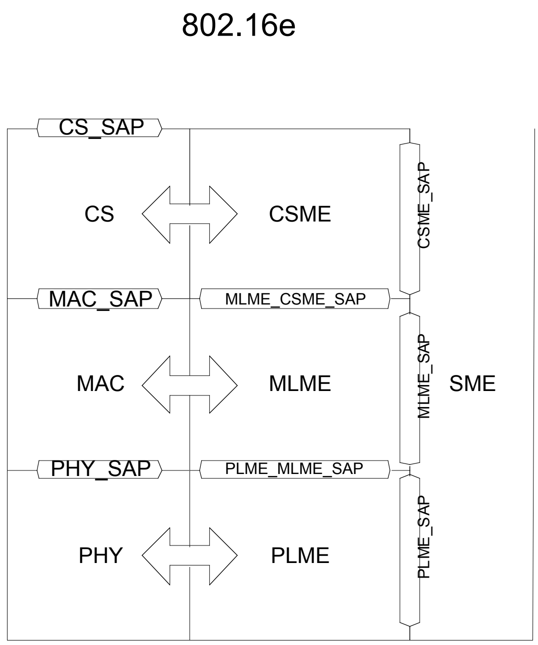
The 802.16e model still has problems. The MA is still defining the BS as being a bridge. There can be no services available at the top of the stack, since there is no upwards data SAP. There needs to be a separate MA defined for each CS.

The MA is defined as being something in the data path. This seems odd, given that an MA usually is concerned with making handover decisions and in some way managing a handover and address mapping. In general it looks like a management function.

In the specification, there are no specific procedures assigned to the MA, so it would be inappropriate for it to exist as a layer in the data stack. Hypothetically it should exist in the SME as an implementation defined function that will be related on the larger system architecture of the BS or MSS. The way that the MA can interact with the stack is through the CSME_SAP, MLME_SAP and PLME_SAP.

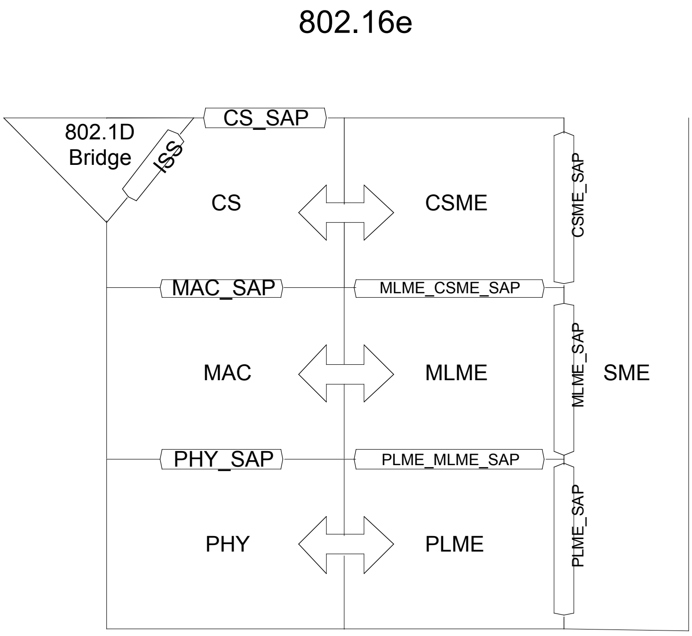
Narrowing in to just the .16e stack, since the MA is in the SME, we do not need to define it, nor do we need to define its interfaces. All we need to define is within the bounds of the CS, MAC, PHY, CSME, MLME and PLME and the interfaces to and between them.

So…



This appears to be much more tractable. We know where the MA does its thing (the SME to use 802.11 terminology). We know how it can influence the stack (Through the management SAPs). The stack looks very like the 802.11 stack, except with the addition of the 802.16 style CS. We know what the service is (the CS_SAP).

If we want an architecture that is a bridge, consistent with the larger 802 model, the path is clear. We define the top of the CS to be as in 802.2. If we need bridging, we define the ISS to be as in 802.1D. There is no need to call out framing for 802.1Q tagging. 802.1Q does the mapping from the ISS to the EISS for us. So we should be removing Q tagging from the 802.1Q packet CS and referring to 802.1Q, 802.1D and 802.2 to do the heavy lifting for us. This might look like this:

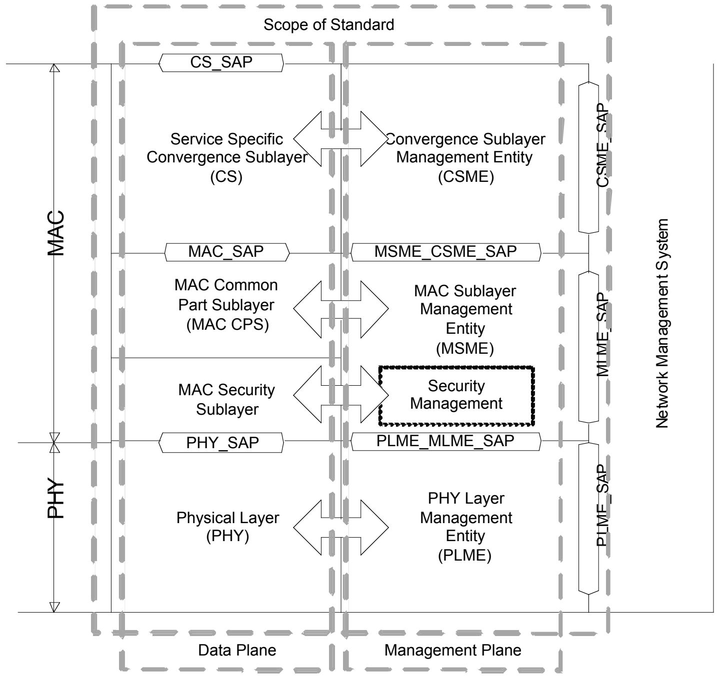


The specific difference between the 802.1D model and this is that to 802.1D, the CS is the MAC entity, since it is the CS that is making 802.16 look like an 802 MAC service. This gives 802.16 a headstart over a number of other 802 standards on convergence of the upper services, since by importing the ISS and 802.2 interfaces, we are supplying an exactly conformant 802 service, rather than an approximation to it that requires special mappings of the type that can be seen in 802.1Q for other media.

One property of the above two models is that the 802.16e model is identical to the 802.16d model when looking at the parts that are in scope for 802.16. This is a good property since it allows us to unify the model in both specifications. It allows us to delete some functionality (Q tagging, bridging) that is defined elsewhere. It is consistent with existing wireless management models that have been seen to work.

We also have a path over which to carry BS-BS primitives, namely the MLME_SAP. There is no reason that the CSME and/or the PLME should not talk to peer entities on another BS through BS-BS primitives. Thus we can define BS-BS primitives on all these interfaces and we can know which management entity they are destined for. The SME is responsible for arranging transport for BS-BS messages. The transport is out of scope, so is the SME functionality, so there is no conflict, we don’t need to define it.

So bringing in the extra information from figure 1 concerning the scope of the standard and the location of the data and management planes and renaming some of the 802.11 derived names to bring them into line with 802.16 terminology, I propose the following be substituted for figure 1.



The CS SAP does not need describing directly. It can be described by reference to 802.2. This is in “**2.3 LLC sublayer/MAC sublayer interface service specification**”
It defines the following primitives..

MA-UNITDATA request (
source_address,
destination_address,
routing_information,
data,
priority,
service_class
)

MA-UNITDATA indication (
source_address,
destination_address,
routing_information,
data,

reception_status,
priority,
service_class
)

and

MA-UNITDATA-STATUS indication (
source_address,
destination_address,
transmission_status,
provided_priority,
provided_service_class
)

The ISS is needed is 802.1D bridging is to be done. This is defined in 802.1D . The primitives are:

M_UNITDATA.indication (
frame_type,
mac_action,
destination_address,
source_address,
mac_service_data_unit,
user_ priority,
frame_check_sequence
)

and

M_UNITDATA.request (
frame_type,
mac_ action,
destination_address,
source_address,
mac_service_data_unit,
user_priority,
access_priority,
frame_check_sequence
)

Thus we need only an IP and an 802 mode for the packet CS. We do not need separate 802.1Q and 802.3 modes in the CS.