Results of Ballot on Draft Standard D3.0

Comments on clause 9 and Resolutions

Seq. #	Section number	your ini-tials	Cmnt type E, e, T, t	Part of NO vote	Comment/Rationale	Corrected Text	Disposition/Rebuttal
1.	9	msu	t	Y	The current draft specifies that the 1 Mbps modulation shall be 2GFSK with BT = 0.5. The current level of -60 dBc for N >= M+/-3 is not achievable using a filtering method that addresses size and implementation restraints and takes into consideration production variations.	**Change** the formulas to read: Channel N = M +/-2 -20 dBm or -40 dBc, whichever is the lowest power N = M +/- 3,4,5 -30 dBm or -50 dBc, whichever is the lowest power N >= M +/- 6 -40 dBm or -60 dBc, whichever is the lowest power	**Addressed by Clause 14 subgroup**
2.	9	msu	T	Y	The current draft does not specify an algorithm for switching between available rates. An algorithm is required to accommodate the large number of users who require a combination of speed and range.	**Delete** the following sentence: "The algorithm for selecting this rate is implementation dependent and is beyond the scope of this standard."	**Declined** **A common algorithm is not needed to assure interoperability.**
3.	**9.1 10**	**WD**	**E**	**n**	**The figures 35 (MAC Architecture Block Diagram) and 53 (GET and SET Operations) do not match. In particular, figure 35 shows a Sublayer Management interface that is not described in section 10. It is suggested to delete this interface from the figure 35.**	**Delete Sublayer Management interface from figure 35.**	**Accepted** **Need to edit figure**
4.	**9.1**	**rw**	**T**	**y**	The MAC architecture must be able to handle more than one outstanding transmit frame. This is not reflected in clause 9.1, in clause 9.2.5.2 which defines the backoff procedure, or in the MAC transmit state machine in Annex C. This is very important in an infrastucture based system. If an AP is trying to transmit a frame to a STA in poor coverage and has to backoff and retry, the MAC must be able to transmit another frame during the backoff.	The MAC architecure allows a STA to process more that one transmit frame at a time. This allows a STA to transmit a frame while another frame is in backoff due to not receiving an ACK.	**Decline text Addition - cannot transmit during backoff period. (see 9.2.4 second sentence)**

					If this is not done, a STA in poor cover will decrease the through-put of the entire BSS		
5.	**9.1**	**db**	**T**	**Y**	**figure 35, appears to be a hold over from the state machine stuff that was in this clause in D2 - the rest was moved to an annex, but this was left. I think it should be moved also - this picture of a MAC archatecture is not relevant and represents the patitions assumed by the state machine annex.**	**remove this figure from the draft or place it in the state machine annex where it belongs.**	**Accepted:** **- In addition move all text in 9.1 prior to figure35 to annex as well.** **- delete "Viewed along a different axis",** **- delete "Alternative View of" in Figure 36 heading.**
6.	**9.14**	**TT**	**T**	**Y**	There is currently no valid reason why broadcast and multicast frames are required to be fragmented if their size exceeds aFragmentation_Threshold. The only reason for fragmentation is: - to improve reliability of MSDU delivery in a noisy medium Therefore given a certain chance of a bit error it does not make any sense to add more bits to a broadcast frame, which fragmentation does, when any one of these bits received with error, will cause the whole MSDU to be discarded. The often quoted reason of PHY's not being able to transmit MPDUs larger than a certain size would be valid, except that all the PHYs in the current standard quote a maximum MPDU size the PHY **shall** be capable of sending, that is larger than the maximum MSDU size. i.e. 4095 in the FH PHY 65000 in the DS PHY 2500 in the IR PHY I've heard people say that some PHYs cannot transmit continuously for the max length frame time but then these PHYs cannot be 802.11 therefore we don't have to worry about them.	Add new paragraph after first paragrah: **Only Directed Frames shall be fragmented. Broadcast/Multicast frames shall not be fragmented even if their length exceeds aFragmentation_Threshold.**	**Accepted** **Alter 9.4 to reflect this change.**

					So if the PHY can transmit a max length MPDU and fragmenting broadcast frames decreases the probability they get through, then why fragment them. From the implementation point of view, it is simpler to qualify the MSDU length check against aFragmentation_Threshold with the fact the MSDU is a broadcast, than create a whole new TX state machine to transmit framgents back to back.		
7.	**9.14**	**TT**	**T**	**Y**	There is currently no valid reason why broadcast and multicast frames are required to be fragmented if their size exceeds aFragmentation_Threshold. The only reason for fragmentation is: - to improve reliability of MSDU delivery in a noisy medium Therefore given a certain chance of a bit error it does not make any sense to add more bits to a broadcast frame, which fragmentation does, when any one of these bits received with error, will cause the whole MSDU to be discarded. The often quoted reason of PHY's not being able to transmit MPDUs larger than a certain size would be valid, except that all the PHYs in the current standard quote a maximum MPDU size the PHY **shall** be capable of sending, that is larger than the maximum MSDU size. i.e. 4095 in the FH PHY 65000 in the DS PHY 2500 in the IR PHY I've heard people say that some PHYs cannot transmit continuously for the max length frame time but then these PHYs cannot be 802.11 therefore we don't have to worry about them. So if the PHY can transmit a max length MPDU and fragmenting broadcast frames decreases the probability	Add new paragraph after first paragrah: **Only Directed Frames shall be fragmented. Broadcast/Multicast frames shall not be fragmented even if their length exceeds aFragmentation_Threshold.**	**Duplicate**

					they get through, then why fragment them. From the implementation point of view, it is simpler to qualify the MSDU length check against aFragmentation_Threshold with the fact the MSDU is a broadcast, than create a whole new TX state machine to transmit framgents back to back.		
8.	9.1.1	jz	t		Replace "ad hoc" with "independent" or "autonomous".		**Accepted use word IBSS**
9.	9.2	**BO**	**T**	**Y**	All references to multirate suppport shall be deleted. There is no mechanism described to allow any determination of interoperability to be made.	~~The medium access protocol allows for stations to support different sets of data rates. All STAs must receive all the Basic Rate Set and transmit at one or more of the Basic Rate Set data rates. To support the proper operation of the RTS/CTS and the Virtual Carrier Sense mechanism, all STAs must be able to detect the RTS and CTS frames. For this reason the RTS and CTS frames must be transmitted at one of these mandatory rates.~~	**Declined by MAC Group vote to adopt Multirate support as described in 96/79r1**
10.	**9.2 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	frame and the returning ACK frame. All stations within the reception range of either the originating station (which transmits the RTS) or the destination station (which transmits the CTS) shall~~will~~ learn of the medium reservation. Thus a station may~~ean~~ be "hidden" from the originating station and still know about the impending use of the medium to transmit a data frame.	**Accepted**
11.	**9.2**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	The RTS/CTS exchange also performs a type of fast collision detection and transmission path check. If the return CTS is not detected by the STA	**Accepted**

						originating the RTS, the originating STA may~~ean~~ start the process over (after observing the other medium use rules) more quickly than if the long data frame had	
12.	**9.2**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	overlap. The medium reservation mechanism works across the BSA boundaries. The RTS/CTS mechanism may~~ean~~ also improve operation in a typical situation where all STAs may~~ean~~ hear the AP but not all other STAs in the BSA.	**Accepted**
13.	**9.2 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	The RTS/CTS mechanism shall~~ean~~ not be used for broadcast and multicast frames because there are multiple destinations. This mechanism need not be used for every data frame transmission. Because the	**Accepted**
14.	**9.2**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	The use of the RTS/CTS mechanism is under control of the RTS_Threshold attribute. This parameter is a manageable object and may~~ean~~ be set on a per station basis. This mechanism allows stations to be	**Accepted**
15.	**9.2 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	A STA configured not to initiate the RTS/CTS mechanism shall~~must~~ still update its Virtual Carrier Sense mechanism with the duration information contained in an RTS or CTS frame, and shall~~must~~ always respond to an RTS addressed to it with a CTS.	**Accepted**
16.	**9.2 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	The medium access protocol allows for stations to support different sets of data rates. All STAs shall~~must~~ receive all the Basic Rate Set and transmit at one	**Accepted**

						or more of the Basic Rate Set data rates. To support the proper operation of the RTS/CTS and the Virtual Carrier Sense mechanism, all STAs shall~~must~~ be able to detect the RTS and CTS frames. For this reason the RTS and CTS frames shall~~must~~ be transmitted at one of these mandatory rates.	
17.	**9.1.2 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	of the distributed coordination function. This access method uses a point coordinator, which shall~~must~~ operate at the access point of the BSS, to determine which station currently has the right to transmit. The	**Accepted**
18.	**9.1.2**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	use of an access priority mechanism, aided by the virtual carrier sense mechanism. Different classes of traffic may~~can~~ be defined through the use of different values for Inter Frame Spacing (IFS), thereby creating prioritized access to the medium for those classes with a shorter IFS. The point coordination	**Accepted**
19.	**9.1.2**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	allowed to begin their transmissions under the DCF access method. The point coordinator may~~can~~ then control the frame transmissions of the stations so as to eliminate contention for a limited period of time.	**Accetped**
20.	**9.2.1**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	The NAV state is combined with physical carrier sense to indicate the busy/free state of the medium. The NAV may~~can~~ be thought of as a counter, which is counting down. When the counter is zero the virtual carrier	**Accepted**

						sense indication is free.	
21.	9.2.1	jz	T	**Y**	Multicast/Broadcast reliability is compromised by the power save mechanism. We should adopt the mechanism is 96/15 and 96/16 to fix this. My "No" vote will only change to a "Yes" vote if we adopt these changes or else mandate the use of a stripped-down PCF to enhance multidestination reliability.	<<adopt text from 95/15 for this subclause>>	**Unresolved. Plenary did not accept MAC group recommendation to adopt 96/15 and 96/16.**
22.	9.2.1	jz	T	**Y**	Multicast/Broadcast reliability is compromised by the power save mechanism. We should adopt the mechanism is 96/15 and 96/16 to fix this. My "No" vote will only change to a "Yes" vote if we adopt these changes or else mandate the use of a stripped-down PCF to enhance multidestination reliability.	<<Adopt changed text for this section from 96/15 and 96/16.>>	**DUPLIC ATE**
23.	**9.2.1, 9.3.2.2, 9.4, 14.4.2.2 , 15,2,3,5**	**vz**	**E**		On page 72, under 9.2.1 there is a reference to a clause with no number following it. Please identify the clause or subclause number. The same occurs on page 85 under 9.3.2.2, and on page 90 under 9.4, on page 188 under 14.4.2.2, on page 220 under 15.2.3.5.		**Accepted**
24.	**9.2.10**	**ch**	**e**		**grammer problems**	All timings are referenced from the end of the transmission, which is ~~are~~ referenced from the last symbol of a frame on the medium.	**Accepted**
25.	**9.2.10**	**ch**	**e**		**Figure 47 uses wrong MIB variable name**	aMAC_Prc~~x~~_Delay~~Time~~	**Accepted**
26.	**9.2.10**	**ch**	**e**		**Fix the funny capitalization of aSlot_Time**	DIFS = aSIFS_Time + 2 * a~~A~~Sl~~L~~ot~~T~~_time Tx_PIFS = Tx_SIFS + a~~A~~Sl~~L~~o~~T~~_time	**Accepted**
27.	**9.2.10**	**ch**	**T**	**Y**	**inconsistant definition of aSlot_Time - the picture include aMAC_Prc_Time in Slot_Time but the text does not. The PHY MIB defintion in 13.1.4.4 matches the text here.**	ASLoT_time is: aCCA_Asmnt_Time + aRxTx_Turnaround_Time + aAir_Propagation_Time +	**Accepted** **Also recommend change to 13.1.4.4 to add**

					I think the picture is correct, aSlot_time also includes aMAC_Prc_Delay.	aMAC_Prc_Delay	**aMAC_Prc_Delay**
28.	**9.2.10**	**ch**	**T**	**Y**	**Remove this sentance because there is no reason why this should be fixed - it should be a per PHY value. It is not fixed according to the definition in 13.1.4.19**	~~aAir_Propagation_Time is fixed at 1 usec.~~	**Accepted** **Specified in MIB already.**
29.	**9.2.10 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	and the different MAC Slot Boundaries Tx_SIFS, Tx_PIFS and Tx_DIFS. These Slot Boundaries define when the transmitter shall~~can~~ be turned on by the MAC to meet the different IFS timings on the medium,	**Accepted**
30.	**9.2.10 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	The tolerances are specified in the MIB, and shall~~will~~ only apply to the SIFS specification, so that tolerances shall~~will~~ not accumulate.	**Accepted**
31.	9.2.10	jz	T	**Y**	The paragraph "The following equations..." claims that the slot definitions take timing variability into account. I think this should be clarified. In any case, it should indicate that it is the *PHY MIB* that defines the numbers.	<<I will write text at the La Jolla meeting after the MAC group has discussed SIFS "slop" and timing variability>>	**Declined** **Text already says 'Slot time is fixed per PHY'**
32.	**9.3**	**AS**	**t**	**y**	**The PC does not gain priority access due to the use of PIFS but due to the fact that everybody else has their NAV set during the CFP.**	**Original Text:** All STA inherently obey the medium access rules of the PCF, because these rules are based on the DCF, with the Point Coordinator gaining priority access to the medium using a PCF IFS (PIFS) which is smaller than the DCF IFS (DIFS) used by the DCF to access the medium. **Replacement Text:** All STA inherently obey the medium access rules of the PCF, because these rules are based on the DCF, and they set their NAV at the beginning of each CFP.	**Accepted** **new text is better.**

33.	**9.3**	**ch**	**T**	**Y**	**According to subclause 5.5, Class 3 frames, whcih include the CFP control frames, can only be sent when associated. According to subclause 5.4.2.2, association is a service between a statino and an AP.** **I think this means that only an AP can be a Point Coordinator (in fact, it says that a few paragraphs later, but I had fun figuring it out the hard way!).**	It is an option for an AP ~~STA~~ to be able to become the Point Coordinator(PC).	**Accepted** **see 34**
34.	**9.3**	**TT**	**t**	**Y**	Section 9.3.2 indicates that the PC is in the AP. Therefore non-AP STAs cannot be the PC. Stronger wording to ensure only one frame is transmitted on a CF-Poll. Also how a CF-Aware station handles the need to retransmit is not explictly described. How retries are handled during the CFP is not mentioned in this standard. I believe the assumption was that the PC	Rewrite second sentence '~~It is an option for~~ ....' as follows: **The Point Coordinator(PC) must reside in the AP. It is an option for an AP to become the PC.** Change text in first paragraph: .....in the contention free period. When polled by the Point Coordinator, a CF-Aware station may transmit **only** one frame to any destination (not just to the Point Coordinator), and may "piggyback" the acknowledgment of a frame received from the Point Coordinator using particular data frame subtypes for this transmission. **If the data frame is not in turn, acknowledged theCF-Aware station shall not re-transmit the frame until it is polled again by the Point Coordinator. The CF-Aware station shall maintain the same sequence number in subsequent transmissions of the same frame even though it may have transmitted them in other CFPs or even the Contention Period.** If the addressed recipient of a CF ..... Add new paragraphs after 1st paragraph:	**Accepted** **Accepted** **only one MSDU which can be to any destination.....** **Accepted** **change until to unless.** **Delete second sentence.** **Add:** **, or it decides to retransmit during the Contention the Period.**

					can move on with its polling list rather than retrying an unacknowledged frame. Since this is somewhat different to the DCF rules it should be stated explicitly.	**A PCF that is maintaining a polling list shall not perform a DCF retry on an unacknowledged frame transmission during the CFP. The frame can be transmitted again the next time the particular SID is at the top of polling list. The AP shall maintain the same sequence number in subsequent transmissions of the same frame even though it may have transmitted other new frames.** **A PCF may re-transmit an unacknowledged frame during the CFP after a PIFS time.**	not perform a "backoff" on an... Delete The AP....
35.	**9.3**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	The rules under which multiple, overlapping point-coordinated BSSs mayeen coexist are presented in	**Accepted**
36.	**9.3**	**TT**	**t**	**Y**	Section 9.3.2 indicates that the PC is in the AP. Therefore non-AP STAs cannot be the PC. Stronger wording to ensure only one frame is transmitted on a CF-Poll. Also how a CF-Aware station handles the need to retransmit is not explictly described.	Rewrite second sentence 'It ~~is an option for~~ ....' as follows: **The Point Coordinator(PC) must reside in the AP. It is an option for an AP to become the PC.** Change text in first paragraph: .....in the contention free period. When polled by the Point Coordinator, a CF-Aware station may transmit **only** one frame to any destination (not just to the Point Coordinator), and may "piggyback" the acknowledgment of a frame received from the Point Coordinator using particular data frame subtypes for this transmission. **If the data frame is not in turn, acknowledged theCF-Aware station**	**DUPLICATE**

					How retries are handled during the CFP is not mentioned in this standard. I believe the assumption was that the PC can move on with its polling list rather than retrying an unacknowledged frame. Since this is somewhat different to the DCF rules it should be stated explicitly.	**shall not re-transmit the frame until it is polled again by the Point Coordinator. The CF-Aware station shall maintain the same sequence number in subsequent transmissions of the same frame even though it may have transmitted them in other CFPs or even the Contention Period.** If the addressed recipient of a CF ..... Add new paragraphs after 1st paragraph: **A PCF that is maintaining a polling list shall not perform a DCF retry on an unacknowledged frame transmission during the CFP. The frame can be transmitted again the next time the particular SID is at the top of polling list. The AP shall maintain the same sequence number in subsequent transmissions of the same frame even though it may have transmitted other new frames.** **A PCF may re-transmit an unacknowledged frame during the CFP after a PIFS time.**	
37.	**9.4**	**maf**	**T**	**Y**		**allow reception of a minimum of 3 MSDUs instead of 6**	**Accepted**
38.	**9.4**	**maf**	**T**	**Y**	**Last paragraph implies that multiple MSDUs may be oustanding in Transmission. This means multiple MACs residing in a single antenna. The word "each" implies that there could be more than one MSDU outstanding. How is it possible that a STA is allowed to have multiple MSDUs outstanding? How do I intersperse the transmission attempts for each MSDU? Do I have spearate backoff functions for each MSDU that is pending? This would be tantamount to having multiple MACs residing within**	**Last paragraph should be replaced with the following text (note that the only actual change to this paragraph is changing the word "each" to the word "the"):** The source station shall maintain a Transmit MSDU Timer for *the* MSDU being transmitted. The attribute aMax_Transmit_MSDU_Lifetime	**Unresolved, but will probably be Declined by Main MAC Group vote This comment implies the author interprets the standard requires only one MSDU is being transmitted at the same time. Othes in the group interpret the standard says (or should say)**

					a single antenna - I would end up with one MSDU being transmitted during the backoff of another, which would be very unfair. This is just wrong.	specifies the maximum amount of time allowed to transmit a MSDU. The timer starts on the attempt to transmit the first fragment of the MSDU. If the timer exceeds aMax_Transmit_MSDU_Lifetime then all remaining fragments are discarded by the source station and no attempt is made to complete transmission of the MSDU.	**that multiple MSDUs can be transmitted at the same time.**
39.	**9.3.1 9.3.3.4**	**WD**	**E**	**n**	**This section uses the CFP_Rate field name, whereas this is specified as the CFP Period field in section 7.3.2.5**	**Change all occurrences of CFP_Rate into CFP_Period.**	**Accepted**
40.	**9.3.1**	**ch**	**t**	**Y**	**Subclause 7.3.2.5 says that the field in the DTIM beacon is CFP_Period (not rate) and is defined in units of DTIM Intervals (not beacon intervals).** **Correspoding comment has been made in 11.4.4.1.24 to change the MIB definition of CFP_Rate**	This value, in units of DTIM~~beacon~~ intervals, is communicated to other stations in the BSS in the CFP_Period~~Rate~~ field of the CF Parameter Set Element of Beacon frames.	**Accepted**
41.	**9.3.1**	**ch**	**t**	**Y**	**Says rate, really means duration**	If the CFP Duration ~~Rate~~ is greater than the beacon interval, the PC shall transmit beacons at the appropriate times during the CFP	**Accepted**
42.	**9.3.1 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	delay. In the case of a busy medium due to DCF traffic, the beacon shall~~will~~ be delayed for the time required to complete the current DCF frame exchange. The longest delay ~~will~~ occurs when~~if~~ the current frame exchange is an MSDU which is larger than both aRTS_Threshold and aFragment_Threshold. In	**Accepted**
43.	**9.4**	**amb**	**e**		**"Error! Reference..." should be corrected**		**Accepted**
44.	**9.4**	**ch**	**e**		**grammer**	The fragmentation and reassembly mechanisms allows for fragment retransmission.	**Accepted**

45.	**9.4**	**db**	**E**	**n**	**2ND paragraph auto ref bad.**	**fix reference**	**Accepted**
46.	**9.4**	**sb**	**e**	**n**	**Minor editorials in the second paragraph of this section. Three periods and an erroneous reference.**	**Correct.**	**Accepted**
47.	**9.4**	**TT**	**t**	**Y**	The text in this section was confusing as it refered to payload which was not defined. Since fragments are MPDUs and its the MPDU length that is set to aFragmentation_Threshold the text needs rewording. Since only FH radios have dwell time boundaries the text should explicitly say its talking about an FH radio.	Change text of second paragraph: The ~~payload~~ **size** of a fragment **MPDU** shall be an equal number of octets for all fragments except for the last, which may be smaller. The ~~payload~~ **size** of a fragment **MPDU** shall never be larger than aFragmentation_Threshold unless WEP is invoked..... Change text of third paragraph: When data is to be transmitted, the number of octets in the ~~payload~~ **fragment** (pre WEP processing) ~~of the fragment~~ shall be determined by ..... Change text of fourth paragraph: The number of data octets in ~~the payload of~~ a fragment **MPDU** shall depend on **aFragmentation_Threshold and the number of octets in the MPDU that have not yet been assigned to a fragment** ~~the values of the following variables~~ at the instant the fragment is contsructed for the first time.~~:~~ ~~a) aFragmentation_Threshold~~ ~~b) The number of octets in the MSDU that have not yet .....~~ Change text of second last paragraph: ~~Since the~~ **In an FH PHY station,** control of the channel will be lost	**Accepted**

48.	**9.4 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	is invoked for the MPDU.. . If WEP is active for the MPDU, then the MPDU shall~~will~~ be expanded by IV and ICV (see clause **Error! Reference source not found.**), this may~~can~~ result in a fragment larger than aFragmentation_Threshold.	**Accepted**
49.	**9.4 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	Since the control of the channel is~~will be~~ lost at a dwell time boundary and the station shall~~will~~ have to contend for the channel after the dwell boundary, it is required that the acknowledgment of a fragment be	**Accepted**
50.	**9.4**	**mif**	**T**	**Y**	The provision that the frame body of all fragments, except the final fragment of an odd-length MSDU, shall be an even number of octets is no longer present in this sub-clause. This provision was a fundamental aspect of my votes in favor of the fragmentation proposal at the July, 1994 Plenary meeting, and in the successful resolution of some of my letter ballot comments relating to fragmentation in earlier letter ballots. Its omission in D3.0 renders the entire fragmentation mechanism unacceptable. If fragmentation is to be retained, all fragments, other than the final fragment, should be required to be both **equal in length** and **an even number of octets in length**. The added overhead in many implementations of reassembling fragments of odd length is unnecessary and unjustifiable, especially considering that only 1 of the 3 PHYs has a major need for fragmentation, and 1 of the other PHYs has no need for fragmentation, so the facility is present in the MAC for (at most) 1.5 out of 3 PHYs. (This text change also corrects an editorial problem with a dangling reference.)	The payload of a fragment shall be an equal number of octets for all fragments except the last, which may be smaller. The payload of a fragment shall always contain an even number of octets, except for the last fragment of an odd-length MSDU, which shall contain an odd number of octets. The payload of a fragment shall never be larger than aFragmentation_Threshold unless WEP is invoked for the MPDU.. . If WEP is active for the MPDU, then the MPDU will be expanded by IV and ICV (see clause 8.), this can result in a fragment larger than aFragmentation_Threshold.	**Accepted**
51.	**9.4**	**TT**	**t**	**Y**	The text in this section was confusing as it refered to payload which was not defined. Since fragments are	Change text of second paragraph:	**DUPLICATE**

					MPDUs and its the MPDU length that is set to aFragmentation_Threshold the text needs rewording. Since only FH radios have dwell time boundaries the text should explicitly say its talking about an FH radio.	The ~~payload~~ **size** of a fragment **MPDU** shall be an equal number of octets for all fragments except for the last, which may be smaller. The ~~payload~~ **size** of a fragment **MPDU** shall never be larger than aFragmentation_Threshold unless WEP is invoked..... Change text of third paragraph: When data is to be transmitted, the number of octets in the ~~payload~~ **fragment** (pre WEP processing) ~~of the fragment~~ shall be determined by ..... Change text of fourth paragraph: The number of data octets in ~~the payload of~~ a fragment **MPDU** shall depend on **aFragmentation_Threshold and the number of octets in the MPDU that have not yet been assigned to a fragment** ~~the values of the following variables~~ at the instant the fragment is contsructed for the first time.~~:~~ ~~a) aFragmentation_Threshold~~ ~~b) The number of octets in the MSDU that have not yet .....~~ Change text of second last paragraph: ~~Since the~~ **In an FH PHY station,** control of the channel will be lost	
52.	**9.5**	**maf**	**t**	**Y**	**This is an implementation issue and should not be specified here.**	**Strike the sentence: All stations shall support the simultaneous reception of a minimum of 6 MSDU's.**	**Declined** **A minimum level of performance is needed therefore a number**

							must be specified. Author's previous comment allready accepted to change this from 6 to 3. See comment 37
53.	**9.5**	**maf**	**T**	**Y**	**Text as written implies that STA must maintain as many timers as there are incoming MSDU's, and this could be a very large number in the worst case, and if the worst case happens, then everyone is non-compliant. Also, the text does not currently state what a STA shall do with a new MSDU when it runs out of timer hardware to monitor yet another simultaneous reception.**	**second from last paragraph, add text after the first sentence, as shown: "The destination station will maintain a aReceive_MSDU_Timer attribute for each MSDU being received, for a minimum of 3 MSDUs. The STA may implement additional timers to be able to receive additional simultaneous MSDUs. The receiving station shall discard all fragments that are part of an MSDU for which a timer is not maintained."**	**Accepted reword for grammar**
54.	**9.1.4**	**ch**	**t**		**Second paragraph, if the MSDU is too long, the MSDU must be fragmented, not the 'frame'**	When a MSDU~~frame~~ is received from the LLC with a ~~MSDU~~ size greater than aFragmentation_Threshold, the MSDU~~frame~~ must be fragmented	**Accepted**
55.	**9.1.4**	**AS**	**t**	**y**	**Only the last fragment is allowed to be smaller than aFragmentation_Threshold**		**Declined does not impact interoperability or receiver design**
56.	**9.1.4**	**TT**	**T**	**Y**	The following comment essentially wishes to add text which says that only DATA frames are fragmented. All Control and Management frames are not. The issue of whether to fragment Control and Management frames is only relevant for Beacon frames. All Control frames are less than 256 bytes long, therefore will never be fragmented. Similarily all Management frames except an AP Beacon, are also less than 256 bytes long (the minimum fragmentation threshold size). Since the Beacon MPDU is a broadcast frame with a maximum length of 355 bytes the value of fragmenting this frame if the threshold is below this amount is questionable. Especially since the element that will be split by the fragmentation is the TIM which will require	Add new paragraph after first paragraph: **Only DATA frames shall be fragmented. All Control and Management frames shall not be fragmented, even if their length exceeds aFragmentation_Threshold.**	**Declined Covered in comment 6**

					the beacon be re-assembled first before an STA can determine if its SID bit is set.		
57.	**9.1.4 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	When a frame is received from the LLC with a MSDU size greater than aFragmentation_Threshold, the frame shall~~must~~ be fragmented. The MSDU is divided into MPDUs. Each MPDU is a fragment with a	**Accepted**
58.	**9.1.4**	**TT**	**T**	**Y**	The following comment essentially wishes to add text which says that only DATA frames are fragmented. All Control and Management frames are not. The issue of whether to fragment Control and Management frames is only relevant for Beacon frames. All Control frames are less than 256 bytes long, therefore will never be fragmented. Similarily all Management frames except an AP Beacon, are also less than 256 bytes long (the minimum fragmentation threshold size). Since the Beacon MPDU is a broadcast frame with a maximum length of 355 bytes the value of fragmenting this frame if the threshold is below this amount is questionable. Especially since the element that will be split by the fragmentation is the TIM which will require the beacon be re-assembled first before an STA can determine if its SID bit is set.	Add new paragraph after first paragraph: **Only DATA frames shall be fragmented. All Control and Management frames shall not be fragmented, even if their length exceeds aFragmentation_Threshold.**	**DUPLICATE**
59.	**9.2.3**	**ch**	**e**		**extra word**	PHY MIB parameters ~~are~~ specify IFS values.	**Accepted**
60.	9.2.3	jz	T	**Y**	Treating SIFS as a constant value in the MAC is wrong. Implementations must be allowed a certain amount of "slop" for interframe timings. They must ensure that their frames don't start too soon after a previous frame (or else the intended recipient may not yet be ready to receive), nor too long (or someone else may grab the medium). We need three SIFS values: min-SIFS, nominal-SIFS and max-SIFS. The duration field should be encoded based on	Each PHY shall define aRxTx_Turnaround_Time in terms of a nominal value plus/minus some tolerance. A conformant 802.11 implementation shall ensure that, when transmitting a frame after a SIFS, transmission does not occur before the minimum allowable duration of a SIFS	**Accepted** **Subject was refered to PHY groups during full WG meeting.**

					the *maximum* length of time we allow to elapse between frames (max-SIFS). But the MAC should only wait min-SIFS before telling the PHY to transmit. Basically, the standard has an idealized notion of a MAC that instantaneously commands the PHY to do something, and the PHY instantaneously responds. Real implementations may not be able to ensure sub-microsecond repeatability in timings. There needs to be a (small) window within which frame transmission can commence. Add this paragraph at the end of the subclause:	nor after the maximum allowable duration of a SIFS.	
61.	**9.3.2**	**AS**	**t**	**y**	**Contention in the CF period is prevented because everybody set their NAV**	**Original Text:** This prevents most contention by preventing non–polled transmissions by stations which received the beacon, whether or not they are CF–Aware. **Replacement Text:** This prevents most contention by preventing non–polled transmissions by stations whether or not they are CF–Aware.	**Accepted**
62.	**9.5**	**TT**	**t**	**Y**	Incorrect text.	Change More Fragments Indicator description as follows: More Fragments Indicator: Indicates to the destination station that this is **not** the last fragment.....	**Accepted**
63.	**9.5 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	MSDU. Only the last or sole fragment of the MSDU shall~~will~~ have this bit set to zero. All other fragments of the MSDU shall~~will~~ have this bit set to one.	**Accepted**
64.	**9.5 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	The destination station shall~~can~~ reconstruct the MSDU by combining the fragments in order of Fragment Number portion of the Sequence Control Field. If WEP has been applied to the fragment it shall be	**Accepted**

65.	**9.5 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	not yet complete. As soon as the station receives the fragment with the More Fragments bit set to zero, the station knows that no more fragments may~~will~~ be received for the MSDU.	**Accepted**
66.	**9.5 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	To properly reassemble MPDUs into an MSDU, a destination station shall~~must~~ discard any duplicated fragments received. If a station receives a fragment with the same Source, Destination, and Sequence Control Field as a previous fragment, then the station shall~~must~~ discard the duplicate fragment. However an acknowledge shall~~must~~ be sent in response to a duplicate fragment of a directed MSDU.	**Accepted**
67.	**9.5**	**TT**	**t**	**Y**	Incorrect text.	Change More Fragments Indicator description as follows: More Fragments Indicator: Indicates to the destination station that this is **not** the last fragment.....	**DUPLICATE**
68.	**9.2.3.1 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	The SIFS timing shall~~will~~ be achieved when the transmission of the subsequent frame is started at the Tx_SIFS Slot boundary as specified in clause **Error! Reference source not found.**.	**Accepted**
69.	**9.2.4**	**amb**	**e**		**Figure 39 shows Cwmin to be 31. Everywhere else it is set to 7.**	**Show it as 7 in figure**	**Accepted**
70.	**9.2.4**	**ch**	**e**		**CW values 7 and 15 are missing from figure 39..**	add values 7 and 15 to figure 39	**Accepted**
71.	**9.2.4**	**ch**	**e**		**sentance should not be underlined**	of aCWmax. ~~A retry is defined as the entire sequence of frames sent to~~	**Accepted**

						~~attempt to deliver an MPDU.~~ A retry is defined as the entire sequence of frames sent to attempt to deliver an MPDU. The CW will remain at a value of aCWmax for the remaining retries.	
72.	9.2.4	jjk	e		**Figure 39 is incorrect and does not reflect the values of 7 and 15 for Cwmin. Also the last sentence uses the word should. It shall be changed to shall.**	aCWmin and aCWmax are MAC constants that shall~~should~~ be fixed for all MAC implementations, because they effect the access fairness between stations.	**Accepted**
73.	9.2.4	RM	e		**Figure 39: revise to correct CWmin**		**Accepted**
74.	9.2.4	ch	t		**requirement - needs to be 'shall' instead of 'will'**	The CW shall~~will~~ remain at a value of aCWmax for the remaining retries.	**Accepted**
75. 76.	9.2.4 11.4.4.2 .27	WD	T	Y	**The initial aCWmin default should be increased. This parameter determines the residual collision probability during the collision avoidance process of selecting the backoff delay after a defer. A high collision probability does directly influence the successrate of Broadcast and Multicast traffic, including the Beacon frame used within 802.11. It will further have a negative effect on the efficiency of medium use, resulting in a lower overall throughput of the total system, as demonstarted in the simulations as described in doc P802.11 95/80. The simulation shows a very high "lost Frame" probability for the Cwmin parameter as is currently specified. It is therefore suggested to increase the CWmin parameter as suggested in doc 95/80. The subject of Contention resolution, and Lost frame probability was also addressed in doc 95/182 and 183, with suggestions to decrease the collision probability that was based on the already suggested much larger Cwmin =32. HIPERLAN uses a different mechanism, but their goal is to achieve a maximum collision probability of 3.5 % maximum. The currently specified Cwmin=7 does represent a much much higher collision probability in the 20-30% range.**	**Change 9.2.4, just above figure as follows: The set of CW values are CW=2^k*Cwmin-1, with k ranging from 0 to a value that results in a CW=255. CWmin should be 32 for a DS PHY. CWmin should be TBD for a FH PHY. Cwmin should be TBD for an IR PHY.**	**Declined Resolved by comment 78 with consent of author.**

					Subsequent simulation results will be presented at the meeting where feasible. **Several users that gained experience with the access method using prototype implementations have testified to me that the suggested Cwmin =7 is too low.** **This Cwmin parameter should be the same for all stations that do contend for the medium within the same area, because they affect the access fairness between stations, and can therefore be specified on a per PHY basis, unlike described in section 9.2.4, which specifies this value to be the same accross all PHY's.**		
77.	**9.2.4**	**AS**	**t**	**y**	**Since aCWmin and aCWmax are MAC constants that effect fairness they should be fixed and not be get/replace in the MIB.**	**Original Text:** aCWmin and aCWmax are MAC constants that should be fixed for all MAC implementations, because they effect the access fairness between stations. **Replacement Text:** aCWmin and aCWmax are MAC constants that are fixed for all MAC implementations, because they effect the access fairness between stations.	**Declined,** **Text was changed to accept coment 78.**
78.	**9.2.4** **7.3.1.11**	**TT**	**t**	**Y**	See 7.3.1.11 for detail comment. Immediately after Figure 39 which shows the Exponential increase of CW there is the statement: 'aCWmin and aCWmax are MAC constants that should be fixed for all MAC implementations, beacuse they effect the access fairness between stations.' This statement is totally true however aCWmin and aCWmax are GET-REPLACE MIB variables. The optimum setting for these, especially aCWmin, is different depending on: - the number of active STAs in a BSS - the percentage of these STAs that on average have	Change last sentence of 9.2.4 to say: "aCWmin and aCWmax are **settable** MAC constants that ~~should~~ **shall** be ~~fixed for~~ **common to** all ~~MAC implementations, beacause they effect the access fairness between stations.~~ **STAs within a given BSS. Each STA will update its aCWmin and aCWmax variables from the CW field contained in each Beacon frame received from its AP."**	**Accepted** **strike "shall be common to all STAs within a given BSS".** **change constants to parameters.** **change variables to parameters.** **Add statement that in IBSS value shall be fixed to default MIB values.** **Default MIB value shall be 31.**

					data to send. Since each collision wastes bandwidth, reducing the number of collisions should improve the overall BSS throughput, therefore aCWmin and aCWmax should be controlled by the AP of a BSS by including these parameters in each Beacon frame.		
79.	**9.2.4 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	deliver an MPDU. The CW shall~~will~~ remain at a value of aCWmax for the remaining retries. This	**Accepted**
80.	9.2.4	jz	T	**Y**	Multicast/Broadcast reliability is compromised by the power save mechanism. We should adopt the mechanism is 96/15 and 96/16 to fix this. My "No" vote will only change to a "Yes" vote if we adopt these changes or else mandate the use of a stripped-down PCF to enhance multidestination reliability.	<<Adopt changed text for this section from 96/15 and 96/16.>>	**Unresolved. Plenary did not accept MAC group recommendation to adopt 96/15 and 96/16.**
81.	**9.2.4 7.3.1.11**	**TT**	**t**	**Y**	There is a need to be able to control the aCWmin and aCWmax values on a per BSS basis. In addition, this control must be fair to all nodes in the BSS. The Current CWmin default of 7 will work fine for a few nodes in a BSS but when the number gets large (>50) then the number of collisions would increase dramatically. Simply making aCWmin = 31 as Wim has asked may times will improve this situation, however it is very inefficient for an STA who is the only associated STA in a BSS to have to wait an average of 15 slot times to transmit each frame. The tradeoff between the individual STA's response time vs BSS throughput will change depending on the application, therefore CW should be a dynamic variable. The current standard does not have any way for aCWmin to be adjusted by any management entity. Putting the fields in the Assocation Response and Beacon frame would allow a management entity to set these **on a per BSS basis in a fair manner**. The MIB variables are already GET-REPLACE.	Add the fixed field: CW (Contention Window) which contains: CWmin CWmax A STA receiving a management frame with a valid BSSID and with this fixed field shall set its MIB variables aCWmin and aCWmax to these values.	**Accepted** **see comment 78**

					The default setting should be defined in the MIB and used unless the AP has the capability (and the user has a need) to alter the numbers. From the MAC point of view it does not care what the algorithm is that sets the CW's, but how and where it gets the values to use, as long as everyone in the BSS uses the same numbers. Simple algorithms, which are outside the scope of this standard, could base CW on the number of associated STAs, the current traffic statistics, the number of retry attempts, etc. All of these are, or can be, known by the AP which is the one who should set the CW for its BSS.		
82.	**9.2.4** **7.3.1.11**	**TT**	**t**	**Y**	See 7.3.1.11 for detail comment. Immediately after Figure 39 which shows the Exponential increase of CW there is the statement: 'aCWmin and aCWmax are MAC constants that should be fixed for all MAC implementations, beacuse they effect the access fairness between stations.' This statement is totally true however aCWmin and aCWmax are GET-REPLACE MIB variables. The optimum setting for these, especially aCWmin, is different depending on: - the number of active STAs in a BSS - the percentage of these STAs that on average have data to send. Since each collision wastes bandwidth, reducing the number of collisions should improve the overall BSS throughput, therefore aCWmin and aCWmax should be controlled by the AP of a BSS by including these parameters in each Beacon frame.	Change last sentence of 9.2.4 to say: "aCWmin and aCWmax are **settable** MAC constants that ~~should~~ **shall** be ~~fixed for~~ **common to** all ~~MAC implementations, beacause they effect the access fairness between stations.~~ **STAs within a given BSS. Each STA will update its aCWmin and aCWmax variables from the CW field contained in each Beacon frame received from its AP."**	**DUPLLICATE**
83.	**9.2.4, 11.4.2.2**	**ch**	**t**		**aCWmin and aCWmax are fixed, aren't they? If they're not, isn't an unfair advantage gained by**	9.2.4: aCWmin and aCWmax are MAC	**Declined**

	.1, 11.4.4.2 .27, 11.4.4.2 .28				**someone who chooses to use 31 as a minimum instead of 7?**	constants that ~~are~~should be ~~fixed~~ for all MAC implementations, because they effect the access fairness between stations. 11.4.2.2.1: aCW_max GET-REPLACE, aCW_min GET-REPLACE, 11.4.4.27 "This attribute indicates the maximum size of the contention window, in slots. The ~~default~~ value of this attribute shall be 255." 11.4.4.28: "This attribute indicates the minimum size of the contention window, in slots. The ~~default~~ value of this attribute shall be 7."	**due to accepting comment 78**
84.	9.6	**BO**	**T**	**Y**	Remove all reference to multirate support. The draft provides no mechanism, other than this meager attempt at window dressing, to ensure interoperability and to ensure that attempts to use multiple rates do not consume more bandwidth than they save. The only mechanisms for choosing a particular transmission rate that have been discussed thus far have all been heuristic, depending on learning that a station is no longer capable of communicating at other than the basic rate(s) by failing to receive acknowledgments when communicating at higher rates. The only mechanism to learn that communication at a higher rate is possible, is to attempt to communicate at that higher rate. Both of these methods lead to contradictory requirements to increase throughput (which is the real aim of using	~~***Multirate Support***~~ ~~The following set of rules must be followed by all the stations to ensure coexistence and interoperability on Multirate Capable PHYs.~~ ~~All Control Frames are transmitted at the aBSS_Basic_Rate_Set (which as specified before belongs to the ESS_BASIC_RATE) so they will be understood by all the stations in the ESS.~~ ~~All Multicast and Broadcast Frames are transmitted at the aBSS_Basic_Rate_Set, regardless of~~	**Declined by MAC Group vote to adopt Multirate support as described in 96/79r1**

					multiple rates). First, assuming that communication at a higher rate has been previously established, when that high rate communication fails, several retransmissions will take place, consuming a great deal of the available bandwidth of the BSS. Only after a number of retransmissions, will an attempt be made at a basic rate. Assuming that the basic rate transmission is successful, the bandwidth gain for this frame is negative. Assuming max length frames (the most efficient usage of multiple rates), the time to transmit this frame after N transmissions at the higher rate will be approximately (0.6*N + 1) times the time to transmit at the basic rate. Similarly, trying to establish communication at the higher rate (and failing) will consume the same amount of time (0.6*N + 1 times the time to transmit at the basic rate). As can easily be seen, with N>1 the time consumed to learn that a frame can not be transmitted at the higher rate more than doubles the time required to transmit the frame at the basic rate. The egregious offender here is not the station that has been in previous communication at the higher rate (although its waste of bandwidth is indeed offensive), but the many stations that have only been communicating at the basic rate and desire to communicate at the higher rate. Unless an unambiguous mechanism is described that will prevent the described behavior, the throughput of a multirate BSS will be significantly less than that of a basic rate only BSS.	~~their type.~~ ~~Unicast Data and/or Management Frames are sent on any available transmit rate. The algorithm for selecting this rate is implementation dependent and is beyond the scope of this standard.~~	
85.	**9.6 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	The following set of rules shall~~must~~ be followed by all the stations to ensure coexistence and interoperability on Multirate Capable PHYs. All Control Frames are transmitted at the aBSS_Basic_Rate_Set (which as specified before belongs to the ESS_BASIC_RATE) so they shall~~will~~	**Accepted**

						be understood by all the stations in the ESS.	
86.	9.6	jz	T	**Y**	Multirate is broken. We should adopt the text suggested in document 96/8 to fix it. Each PHY should define a Basic_Rate_Set at which all implementations must be able to send/receive. Individual APs can be configured for a primary rate that is different (higher or lower).	<<adopt changed text for this section from 96/8, and change the term "aBSS_Basic_Rate_Set" (which is not defined anywhere) to "either one of the rates defined in the PHY MIB's BSS_Basic_Rate set or the STA's Primary Rate".>>	**Accepted with different text by MAC Group vote to adopt Multirate support as described in 96/79r1 (author agrees that the desired effect of the comment has been achieved)**
87.	**9.2.5**	**maf**	**t**	**Y**		**allow backoff values greater than those specified**	**Accepted** **Add 511 and 1023 to set of CW values.** **add The CW shall take the next value "or higher" in the series** **.....** **to second sentence.** **Figure 39 needs to be edited to reflect above. VISIO cannot edit!**
88.	**9.7**	**maf**	**T**	**Y**	**The MAC state machines provide a mechanism for creating a concise, logical, self-consistent description of the standard.** **Textual descriptions elsewhere in the document are so spread out that it is difficult to maintain consistency across all descriptions of a partcular subfunction - e.g. NAV operation is not fully described anywhere, but instead, bits and pieces are spread around multiple locations.** **Information as to which frame responses use SIFS, or DIFS, or PIFS is spread around.** **802.3 is cited as a precedent in establishing state machine pseudo-code as the golden mean for possible inconsistency in the standard.**	**The MAC state machine diagrams with the accompanying text should be the golden standard for this specification and not the textual descriptions of functionality as found in the sections outside of section 6.7. The following text should be added:** **The state machine representations and the accompanying text that describes the state machines is the correct embodiement of the standard; Where inconsistencies between other text in the document and the state machine diagrams or their accompanying text arise, then**	**Declined** **No Text provided. May reconsider this in future if correct state machines available.**

						the state machines shall be considered the correct emodiement.	
89.	9.2.3.2	jz	T	**Y**	Multicast/Broadcast reliability is compromised by the power save mechanism. We should adopt the mechanism is 96/15 and 96/16 to fix this. My "No" vote will only change to a "Yes" vote if we adopt these changes or else mandate the use of a stripped-down PCF to enhance multidestination reliability.	<<Adopt changed text for this section from 96/15 and 96/16.>>	**Unresolved. Plenary did not accept MAC group recommendation to adopt 96/15 and 96/16.**
90.	**9.3.2.2**	**mif**	**e**	**N**	fix dangling reference	reference should be to clause 11.1.2.1	**Accepted**
91.	**9.3.2.2**	**ch**	**t**	**Y**	**There is no CFP_Rate in the CF parameter set, the re is Period and Count.** **The STA needs to be prepared to set its NAV at TBTT, based upon when the Beacon_Interval times CFP_Count. This needs to be recalculated with every Beacon received, just in case something changed.**	at which a Contention Free Period is scheduled to start (based on the CFP_Count~~Rate~~ in the CF Parameter Set Element of the beacons from this PC)	**Declined,** **Change rate to period.**
92.	**9.3.2.2**	**ch**	**T**	**Y**	**This subclause says that STA must update their NAV according to the CF_Dur_Remaining in any Beacon, even one from another BSS.** **This subcaluse does not say whether a STA should preset its NAV at TBTT when it is known from information in Beacons for another BSS that that BSS is going to start a CFP.** **If this is the case, some limit needs to be set here, because it is going to require one timer for each of these potential TBTTs at which the STA may have to preset its NAV due to a CFP start.** **I think this is all asking too much, and a STA should only have to pay attention to the Beacon information from its own BSS. Supposedly the fact that the PCF is built on the DCF is going to stop STA from interfering with any CFP that it can hear. If a STA can hear the Beacon, then it can hear half of most of the traffic going on during the CFP, and using the frame duration properly will take care of this.**	Each non–PC station shall update its NAV using the CF_Dur_Remaining value in any error–free CF Parameter Set Element of the beacon frame containing such an element that the station receives. ~~This includes CF_Dur_Remaining values in CF Parameter Set Elements from beacons received from other (overlapping) BSSs. This prevents stations from taking control of the medium during the CFP, which is especially important in cases where the CFP spans multiple medium-occupancy intervals, such as dwell periods of an FH PHY.~~ This setting of the NAV ~~also~~ reduces the risk of hidden stations sensing a DIFS during the CFP and possibly corrupting a transmission in progress.	**Declined,** **Add text at beginning indicating that CFPs are coordinated.** **Change last sentence of this section to indicate receipt of CF-ACK from any BSS will reset the NAV.** **Statement added to say coodination mechanism is beyond the scope of standard.**

93.	**9.3.2.2**	**TT**	**t**	**Y**	If the assumption is that hearing a foreign BSS's beacon with a valid CF_Dur_Remaining value should set the NAV to prevent interference with the foreign BSS's CFP, then it is best to play it safe and not reset the NAV until it expires. (I think it's too much to ask an STA to also be able to clear a NAV set by a foreign BSS when it hears a CF_End from that foreign BSS.)	Add to end of last paragraph: Receipt of either of these frame shall reset the NAV of all stations in the BSS, **unless the NAV was set by a Beacon from an overlapping BSS in which case the NAV shall be allowed to expire normally.**	**Declined,** **Resolved in comment 92, accepted by author of comment.**
94.	**9.3.2.2**	**TT**	**t**	**Y**	If the assumption is that hearing a foreign BSS's beacon with a valid CF_Dur_Remaining value should set the NAV to prevent interference with the foreign BSS's CFP, then it is best to play it safe and not reset the NAV until it expires. (I think it's too much to ask an STA to also be able to clear a NAV set by a foreign BSS when it hears a CF_End from that foreign BSS.)	Add to end of last paragraph: Receipt of either of these frame shall reset the NAV of all stations in the BSS, **unless the NAV was set by a Beacon from an overlapping BSS in which case the NAV shall be allowed to expire normally.**	**DUPLICATE**
95.	**9.3.2.2, 9.3.3.1**	**ch**	**t**	**Y**	**Receipt of a CF-End should only reset the NAV if the NAV is set because of the CFP. If your NAV was set by the CFP, then set to longer due to something else you can hear, clearing it will cause you to destroy that other thing.**	The PC shall transmit a CF–End or CF-END+ACK frame at the end of each CF-Period. Receipt of either of these frames shall reset the NAV of all stations in the BSS, for STA at which the CFP is the only reason the STA has the NAV set at the time the CF-End or CF-End+ACK frame is received. Also the last sentance of 9.3.3.1: All stations of the BSS receiving a CF-End or CF-END+ACK, at which the CFP is the only reason the STA has the NAV set at the time the CF-End or CF-End+ACK frame is received, reset their NAVs so they may attempt to transmit during the contention period.	**Declined,** **Resolved in comment 92, accepted by author of comment.**
96.	**9.3.3.1**	**ch**	**t**	**Y**	**CF_Max_Duration may span more than one beacon interval, so this text must be wrong.**	The CFP ends when the CFP_~~Max~~_Dur_Remaining~~ation~~ time has elapsed since the last Beacon or when the PC has no further frames to	**Accepted**

						transmit nor stations to poll.	
97.	**9.3.3.1 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	A CF-Poll bit in the Subtype field of these frames shall~~will~~ allow the stations to send their data frames if any. Stations shall respond to the CF-Poll immediately when a frame is queued, by sending this frame	**Accepted**
98.	**9.3.3.1 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	addressed to a different station than the one being acknowledged. This shall~~can~~ only occur if the acknowledged frame/fragment was marked as last fragment in the frame control. CF-Aware stations that	**Accepted**
99.	**9.3.3.1 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	A CF-Aware station shall~~must~~ respond to a CF-Poll. If the station has no frame to send when polled, the response shall be a Null frame. If the station has no frame to send when polled, but an acknowledgment is	**Accepted**
100.	9.7	ge	t		last three table items should not have a frames in sequence value	should be a note in the table entries that refers to text defining <CF-Sequence> just above table 20	**Declined last table items already corect.**
101.	**9.7**	**WD**	**T**	**Y**	**It is currently ambiguous what happens when the PS-Poll is followed by an erroneous Data frame. Because the Data frame is not successfully received, in response to the PS-Poll, then the PS-Poll will be retransmitted according to the normal retransmission rules. However if the AP did send Data directly after the SIFS in response to the PS-Poll, but did not receive the Ack, then this migth mean that the Data frame is to be retransmitted after a backoff. It should be noted that this is a special case for the AP, sinse it does not go through an access procedure to send the data, but instead generates it in direct response to the PS-Poll from the station, who did go through the access procedure. In general the retransmission responsibility is usually assigned to the station that did do the initial access procedure, and not by the responding station. this for instance also**	**Modify entry 6 in table 19 into: PS-Poll - Data(dir)**	**Declined, Sugested solution withdrawn by author, retry ambiguity needs to be resolved.**

					applies to the PCF. It is therefore suggested to prevent the ambiguity by deleting the Ack from the PS-Poll - Data-Ack sequence, so that only there will be a PS-Poll - Ack, or PS-Poll - Data sequence. this will clearly give the station the responsibility to regenerate the PS-Poll when the data transfer was not successfull.		
102.	**9.7**	**AS**	**T**	**y**	**Delete the sequence:** PS-Poll – [Data(dir) – ACK –] Data(dir) – ACK This sequence has a number of problems. The basis of which is that PS-Poll frames do not have sequence numbers. This means that the AP has no way of determining if a PS-Poll is a retransmission or a request for the next frame.		**Declined, because text has been added that resolves the ambiguity between PS-Poll transmissions being retries or new ones.**
103.	**9.7**	**TT**	**T**	**Y**	Under the current DCF rules it is not possible to correctly perform the PS-Poll - Data - ACK sequence. Since the PS-Poll is a directed frame that must have a response, there must be a timeout that the source STA must use before doing a DIFS and random backoff. Since the response is a data frame of unkown length, this timeout value is unknown. Currently the only other two timeouts are ACK timeout and CTS timeout, which end at the precise moment where the ACK and CTS frames were supposed to end. I believe it is preferable to eleminate this particular frame sequence rather than change the response timeout rules to wait until a response frame is fully received before you can tell if it is a true response to the frame you sent. It is also not mentioned in the standard, what happens when the DATA is not ACKed. Does the AP retry the data frame, or does it wait for another PS-Poll? If the ACK was transmitted but not received by the AP, then this PS-Poll would not happen until after the next Beacon frame was seen with the appropriate TIM set.	Remove entry: PS-Poll-[Data(dir)-ACK]Data(dir)-ACK from Table 19 Frame Sequences.	**Withdrawn**

					I believe that a much cleaner solution is to have only the PS-Poll - ACK sequence and use the proposed solution described in my comments on clause 11.2.1.4 and 11.2.1.6.		
104.	**9.7**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	Where "DATA*" mayeaa be any of the DATA sub–types, "DATA/END*" mayeaa be any of the DATA or CF–END sub–types, and "*CF-ACK" mayeaa be DATA+CF–ACK or CF–ACK(no data).	**Accpeted**
105.	9.7	jz	t	**Y**	We should add a clarification that only fragments of the *same MSDU* may be transmitted with a SIFS between them. The current text implies that, under some circumstances, Data/Managent may be sent back-to-back.		**Accepted by MAC group adoption of submission 96/76, as amended.**
106.	**9.7**	**TT**	**T**	**Y**	Under the current DCF rules it is not possible to correctly perform the PS-Poll - Data - ACK sequence. Since the PS-Poll is a directed frame that must have a response, there must be a timeout that the source STA must use before doing a DIFS and random backoff. Since the response is a data frame of unkown length, this timeout value is unknown. Currently the only other two timeouts are ACK timeout and CTS timeout, which end at the precise moment where the ACK and CTS frames were supposed to end. I believe it is preferable to eleminate this particular frame sequence rather than change the response timeout rules to wait until a response frame is fully received before you can tell if it is a true response to the frame you sent. It is also not mentioned in the standard, what happens when the DATA is not ACKed. Does the AP retry the data frame, or does it wait for another PS-Poll? If the ACK was transmitted but not received by the AP, then this PS-Poll would not happen until after the next Beacon	Remove entry: PS-Poll-[Data(dir)-ACK]Data(dir)-ACK from Table 19 Frame Sequences.	**DUPLICATE**

					frame was seen with the appropriate TIM set. I believe that a much cleaner solution is to have only the PS-Poll - ACK sequence and use the proposed solution described in my comments on clause 11.2.1.4 and 11.2.1.6.		
107.	**9.2.5.1**	**ch**	**e**		**missing ".", middle of second paragraph**	when it detects the free medium for greater than or equal to a DIFS. If, under these conditions,	**Accepted**
108.	**9.2.6**	**WD**	**T**	**Y**	**There is currently no CTS procedure described. This is of particular interrest, because the CTS may only be returned by a addressed station, when the NAV indicates a free medium, while there is no time to react on the physical CCA signal, because the CTS is to be returned after a SIFS.**	**Add the following text, preferably in section that is inserted in between 9.2.6.1 and 9.2.6.2. - CTS Procedure: A station that is addressed by the RTS frame, will transmit a CTS frame after SIFS, but only when the NAV does indicate that the medium is free. The CTS shall be addressed to the TA address present in the RTS frame. The duration field in the CTS frame shall be the duration field from the received RTS frame, adjusted by substraction of SIFS and CTS time duration.**	Accepted
109.	**9.3.3.2**	**ch**	**t**	**Y**	**This subclause implies that if a STA to STA transfer is fragmented and sent during the CFP, each fragment (i.e. Data/Ack pair) can only be sent after a CF-Poll from the PC - i.e. the two STAs cannot do repeated Data/Ack transactions following a CF-Poll. Is this true?**		**Withdrawn by author**
110.	9.3.4.1	**BO**	**T**	**Y**	Remove vestiges of time bounded services.	The PC shall issue polls to stations ~~whose entries~~ on the polling list ~~are for reasons other than time-bounded service connections~~ in order by ascending SID value.	*Accepted*
111.	9.3.4.1	**BO**	**T**	**Y**	Restrict and clarify usage of CFP	While time remains in the CFP, the	*Accepted*

						delivery of all CF frames has been completed and all stations on the polling list have been polled, the PC may generate one or more CF-Polls to *any* stations on the polling list. While time remains in the CFP, the delivery of all CF frames has been completed and all stations on the polling list have been polled, the PC *may* send Data or Management frames to *any* stations.	
112.	**9.2.5.2**	**maf**	**T**	**Y**	**If a TX is queued just a bit time after the end of a successful TX, then the newly queued transmission will follow the first one WITHOUT A BACKOFF HAVING BEEN EXECUTED!**	**In the 5th paragraph, strike the words: "and has another MSDU ready to transmit (queued)" Add text: A backoff should be performed immediately after the end of every transmission, even if the transmission was successful, and even if no additional transmissions are currently queued. If the transmission was successful, the CW value reverts to CWmin before the random backoff iterval is chosen. This assures that TX frames are always separated by a backoff.**	**Accepted**
113.	**9.2.5.2**	**maf**	**t**	**Y**		**This section does not mention that backoff is also used when a collision is interrepted to have occurred. Clause 6.2.5.3 alludes to collisions, so perhaps a reference to clause 6.2.5.3. would suffice.**	**Accepted (assuming author means 9.2.5.3, not 6.2.5.3)**
114.	**9.2.7.**	**maf**	**t**	**Y**		Broadcast/multicast are almost guaranteed to be NOT delivered, since the time following a beacon is likely to be flooded with asynch upbound traffic (in the absence of a CF period). A possible solution to make broadcast go from almost guaranteed failed delivery (assuming a few STA with traffic to	**Unresolved. Plenary did not accept MAC group recommendation to adopt 96/15 and 96/16.**

						send) to "pretty good" delivery is to require the use of the PIFS to send broadcast/multicast (i.e. force an "unannounced" CF period after every beacon that has broadcast/multicast to be sent) - this would make PIFS capability a requirement of APs. An alternative is that a *portion* of the PCF could be required - i.e. AP would set a PCF period, and would use it for multicast traffic. If there was no multicast, then it would send CF-end. Note that this CF period may be used for actual CF traffic, but with the restriction that multicast traffic must be transmitted first. Broadcast/multicast are now only lost by adjacent interfering BSS's, other ISM devices and noise sources. Another option is to turn off all other TIM bits when SID=0 is set. This prevents most PS-POLL traffic from interfering with the multicasts, but does not prevent asynchronous up-traffic from interfering. Another option is for the AP to choose at random, the address of an associated STA and send the RTS for a multicast frame to that STA. The DATA frame would then contain the multicast address and would be received by all appropriate STA - no ACK would be sent, but at least the NAVs of STA would prevent the majority of collisions. Alternatively, an ACK could be generated by the lucky STA that was randomly selected - although this doesn't really prove that all STA got the frame.	
115.	9.2.5.2	**BO**	**E**		count and time are used interchangeably when describing	A STA in backoff must monitor the	**Accepted**

					backoff.	medium for carrier activity during backoff slots. If no carrier activity is seen for the duration of a particular slot, then the random backoff process shall decrement its backoff time~~count~~ by aSlot_time.	
116.	9.2.5.2	**BO**	**T**	**Y**	This is patently untrue and must be deleted. Consider the case where two STAs have collided on their initial attempt to transmit. Both will select a random backoff period between 0 and 7. A third station that makes its initial attempt at transmission after this collision event has ended will be able to use the medium after a DIFS with probability 1 when each of the colliding stations will be able to access the medium at that same time only with probability 1/8. This clearly favors newcomers over past colliders.	~~The advantage of this approach is that stations that lost contention will defer again until after the next medium busy event, and will then likely have a shorter backoff delay than new stations entering the backoff procedure for the first time. This method tends toward fair access on a first come, first served basis.~~	**Accepted**
117.	**9.2.5.2 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	A STA in backoff shall~~must~~ monitor the medium for carrier activity during backoff slots. If no carrier	**Accepted**
118.	**9.2.5.2 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	the backoff timer shall not be decrement for that slot; The medium shall~~must~~ be sensed as idle for the duration of a DIFS period before the backoff procedure is allowed to resume. Transmission shall	**Accepted**
119.	**9.2.5.2 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	The effect of this procedure is that when multiple stations are deferring and go into random backoff, then the station selecting the lowest delay through the random function shall~~will~~ win the contention. The advantage of this approach is that stations that lost contention shall~~will~~ defer again until after the next medium busy event, and will then likely have a shorter backoff delay than new stations entering the	**Accepted**

120.	9.2.5.2	jz	T	**Y**	Multicast/Broadcast reliability is compromised by the power save mechanism. We should adopt the mechanism is 96/15 and 96/16 to fix this. My "No" vote will only change to a "Yes" vote if we adopt these changes or else mandate the use of a stripped-down PCF to enhance multidestination reliability.	<<Adopt changed text for this section from 96/15 and 96/16.>>	**Unresolved. Plenary did not accept MAC group recommendation to adopt 96/15 and 96/16.**
121.	**9.2.6.1**	**jjk**	**e**		**Incorrect parameter in range specifier in second paragraph**	The aRTS_Threshold attribute shall be a managed object within the MAC MIB, and its value can be set and retrieved by the MAC LME. The aRTS_Threshold attribute shall be constrained to range (0 ... aMax_Frame_Length+1~~Maximum MPDU Length~~). The value 0 shall be used to indicate that all MPDU shall be delivered with the use of RTS/CTS. Values of aRTS_Threshold ≥~~≥~~ aMax_Frame_Length shall indicate that all MPDUs shall be delivered without RTS/CTS.	**Accepted**
122.	**9.2.6.1**	**ch**	**t**		**These two subclauses are cumbersome. It would be clearer with just one subclause describing Directed MPDU Transfer followed by the one describing Broadcast And Multicast MPDU Transfer.** **Also, Figure 46 and the paragraph immediately preceding it, describe the ACK procedure, and should be moved to clause 9.2.8. Also a few words added to that moved paragraph would help its clarity.**	**9.2.6 Directed MPDU Transfer Procedure** ~~**9.2.6.1 Directed MPDU Transfer Procedure Using RTS/CTS**~~ STA shall use an RTS/CTS exchange for directed frames only when the length of the MPDU is greater than the length threshold indicated by the aRTS_Threshold attribute. The aRTS_Threshold attribute shall be a managed object within the MAC MIB, and its value can be set and retrieved by the MAC LME. The aRTS_Threshold attribute shall be	**Accepted, with modified text that refers to the new CTS Procedure subclause**

						constrained to range (0 ... Maximum MPDU Length). The value 0 shall be used to indicate that all MPDU shall be delivered with the use of RTS/CTS. Values of aRTS_Threshold ≥ aMPDU_Max_lngthMax_Frame_Lengt h shall indicate that all MPDUs shall be delivered without RTS/CTS. When RTS/CTS are used tThe asynchronous payload frame (e.g. DATA) shall be transmitted after the end of the CTS frame and an SIFS period. No regard shall be given to the busy or free status of the medium. **9.2.6.2 Directed MPDU Transfer Procedure without RTS/CTS** When RTS/CTS are not used, Following the basic access mechanism, the source STA shall transmit the asynchronous payload frame (e.g. DATA) shall be transmitted following the basic access mechanism. –With or without use of the RTS/CTS mechanism, tThe destination STA which is the destination of a directed asynchronous payload frame shall follow the ACK Procedure. The source STA shall start its backoff time a DIFS after the end of the ACK or a DIFS after aACK_Timeout. Figure 46 Add to the end of subclause 9.2.8 Ack	

						Procedure: The source STA shall start its backoff time a DIFS after the end of the ACK or a DIFS after aACK_Timeout prior to accessing the medium again. Figure 46	
123.	**9.2.6.1**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	The aRTS_Threshold attribute shall be a managed object within the MAC MIB, and its value may~~can~~ be set and retrieved by the MAC LME. The aRTS_Threshold attribute shall be constrained to range (0 ...	**Accepted**
124.	**9.2.7 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	the MPDU is directed to the AP. The Broadcast/Multicast message shall~~will~~ be distributed into the BSS. The station originating the message shall~~will~~ receive the message as a Broadcast/Multicast message. Therefore all stations shall~~must~~ filter out Broadcast/Multicast messages which contain their address as the source address.	**Accepted**
125.	9.2.7	jz	T	**Y**	Multicast/Broadcast reliability is compromised by the power save mechanism. We should adopt the mechanism is 96/15 and 96/16 to fix this. My "No" vote will only change to a "Yes" vote if we adopt these changes or else mandate the use of a stripped-down PCF to enhance multidestination reliability.	<<Adopt changed text for this section from 96/15 and 96/16.>>	**Unresolved. Plenary did not accept MAC group recommendation to adopt 96/15 and 96/16.**
126.	**9.3.3.3**	**ch**	**T**	**Y**	**Normally the PC does not check the status of the medium before transmitting during the CFP. The desire here is not just for the PC to leave a gap of some length every aMedium_Occupancy_Time, but for it to then sense the medium before re-taking it after that gap. This is not specified here.**	To further reduce the susceptibility to inter-PCF collisions, the PC shall require the medium be free for a DIFS plus random (over range of 1 to aCW_min) number of slot times once every aMedium_Occupancy_Limit Kmicroseconds during the CFP. After	**Accepted the intent of the comment with this modified text (provided by commenter):** To further reduce the susceptibility to inter-PCF collisions, the PC shall **be required to sense** the medium free for a DIFS plus

						the medium as been unused by the PC for this amount of time, the PC must sense the medium to be free for a PIFS prior to seizing control again. This can only result in loss of control of the medium to overlapping BSS or hidden station traffic,	random (over range of 1 to aCW_min) number of slot times once every aMedium_Occupancy_Limit Kmicroseconds during the CFP. This **may** ~~can only~~ result in loss of control of the medium to overlapping BSS or hidden station traffic.
127.	**9.3.3.3**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	aMedium_Occupancy_Limit Kmicroseconds during the CFP. This ~~can only~~ results in loss of control of	**Accept**
128.	**9.2.5.3**	**maf**	**t**	**Y**	**Just being a stickler for details, I guess.**	**No reference is made to CRC error being interpreted as a collision. I.e. clause mentions "CTS may not be returned." Returned with CRC error is "returned" in my book. Let's be explicit and include a mention of CRC error as another reason for backing off.**	**Withdrawn**
129.	**9.2.5.3**	**sb**	**e**	**n**	**I assume here (but it does not seem to say explicitly) that the RTS and Data retry counts both increment independently while the sequence is still incomplete, ie the Data retry count does not get reset if an RTS gets retried.**	**-**	**(editorial, but resolved by processing other comments)**
130.	**9.2.5.3 11.4.4.2 .31 11.4.4.2 .32**	**WD**	**T**	**Y**	**The intend of having two Retry Limits is to cope with two significant different situations. One is that retries are needed to retry a transmission that failed primarily due to residual access collisions in the contention resolution process of CSMA/CA. The other case is primarily geared toward a "Hidden Station" situation, where frames are primarily lost, or CTS is not returned. because the medium is busy in the vicinity of the receive station. In the latter case the defer mechanism does not work for the stations that compete for the medium, and hence a higher value for the Retry Limit is needed to increase the probability that subsequent transmissions are separated in time so that they do not overlap and**	**Change text in section 9.2.5.3 Add the following at the end of the last sentence: , unless aRTS_Threshold is higher then 2304, in which case aLong_Retry_Limit should always be used. Change text in section 11.4.4.2.31: Change "aFragmentation_Threshold" into "aRTS_Threshold". Change the default value 5 into 7.**	**Accepted clause 11 parts, but declined clause 9 part as no longer necessary**

					interfere with each other. So in general the Retry Limit needs to be a higher value in the cases when "Hidden Node" protection is targetted for. This can be detected by looking at the aRTS_Threshold parameter, which is 2305 or higher when the RTS/CTS mechanism is switched off. The current mechanism, together with the values specified in the MIB, causes a reverse behaviour. In addition, when the correct (changed) default values are specified in the MIB, then the effect is that the Short_Retry_Limit (the higher value) is then always used when the RTS/CTS mechanism is effectively turned off. The suggested text corrects this problem, by selecting the Short_Retry_Limit only when the RTS_Threshold parameter is lower then the default 2305. In addition it does reverse and change the defaults values specified in the MIB. It also corrects the problem in the MIB, which inadvertently defines aFragmentation_Threshold rather than RTS_Threshold.	**Change text in section 11.4.4.2.32: Change "aFragmentation_Threshold" into "aRTS_Threshold". Change the default value 7 into 4.**	
131.	**9.2.5.3**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	For instance, CTS may not be returned after the RTS transmission. This mayeanmay happen due to a collision with another RTS or a DATA frame, or due to interference during the RTS or CTS frame. It mayeanmay also be that CTS failed to be returned because the remote station has an active virtual carrier sense condition	**Accept**
132.	**9.2.5.3 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	required to transmit the ACK frame plus a SIFS . Since this pending transmission is a retransmission attempt the CW shallwillshall be increased (per the backoff rules). This process shall continue until the	**Accept**
133.	9.2.5.3	jz	t	**Y**	CTS_Timeout is not defined. Presumably, it should be SIFS plus however long it takes to detect the start of frame (I have made comments elsewhere that SIFS should really be a window of allowable times to account for		**Accepted**

					implementation jitter). The same goes for ACK_Timeout.		
134.	9.2.5.3	jz	T	**Y**	The last two paragraphs are confusing, and don't take into account the complicated possibilities for losing a couple of RTSs/CTSs, then getting a fragment through but losing the ACK, and so forth. That is, we need to clarify whether to ***add*** the number of retransmissions of the RTS to any retransmissions of the data before comparing to one of the Retry_Max numbers, and whether to start counting RTS retries over again if we don't get an ACK (i.e. does the sequence RTS...RTS...RTS/CTS/DATA...RTS...RTS leave us with two short retries and one long retry, or four short retries and one long retry, or five retries altogether or what?)	<<I assume we will discuss this at the meeting and I promise to write text at that time, once we agree on how it ought to work.>>	**Accepted**
135.	**9.2.5.3, 11.11.4.1.2.2, 11.4.2.2.1, 11.4.3.2.2, 11.4.4.2.30**	**ch**	**t**	**Y**	**9.2.5.3:** **CTS_TimeoutTimeout is misspelled, and not defined, and the value sof CW is not doubled** **Change the next paragraph to be consistant with the first and refer to the correct MIB variables, and add some punctuation for clarity** **The conditions for using aShort_Retry_limit and aLong_Retry_limit do not match what is described in the MIB definitions of those variables, so I suggest changing the text here.** **clause 11:** **there is no reason for aACK_Timeout to be a MIB variable. It is the sum of two other MIB variables and can be defined as such in the text.**	9.2.5.3: If after an RTS is transmitted, the CTS_Timeout~~Timeout~~ expires without reception of a CTS, then a new RTS shall be generated while following the basic access rules for backoff. Since this pending transmission is a retransmission attempt, the CW shall be increased (per the backoff rules)~~doubled~~ as per the backoff rules. This process shall continue until the number of attempts reaches aShort_Retry_Max. CTS_Timeout is equal to aCTS_Time plus aSIFS_Time. The same backoff mechanism shall be used when no ACK frame is received within a predetermined ACK_Timeout, after a directed DATA frame has been transmitted. ~~The~~ ACK_Timeout is equal to aACK_Time plus aSIFS_Time ~~value is the time required to transmit the ACK frame plus a SIFS~~. Since this pending transmission is a retransmission attempt the CW shall~~will~~ be increased (per the backoff rules). This process shall continue until the	**Accepted (desired effect has been achieved with the modified text)**

						number of attempts reaches either: aLong_Retry_Max for DATA frames the length of which exceed aFragmentation~~RTS~~_Threshold; or, aShort_Retry_Limit for DATA frames the length of which do not exceed aFragmentation~~RTS~~_Threshold. 11.4.1.2.2: aACK_Time, ~~aACK_Timeout,~~ aShort_Retry_Limit, 11.4.2.2.1: aACK_Time GET, ~~aACK_Timeout GET,~~ aShort_Retry_Limit GET-REPLACE, 11.4.3.2.2: aACK_Time, ~~aACK_Timeout,~~ aShort_Retry_Limit, 11.4.4.2.30: ~~aACK_Timeout~~ ~~aACK_Timeout ATTRIBUTE~~ ~~WITH APPROPRIATE SYNTAX~~ ~~integer;~~ ~~BEHAVIOUR~~ ~~"This attribute specifies the length of time, in microseconds, in which an ACK frame will be received in response to transmission of a frame which requires acknowledgment, timed from receipt of PHY_DATA.confirm at the MAC. The following equation is used to determine aACK_Timeout:~~	

						~~aSIFS_Time+aACK_Time";~~ ~~REGISTERED AS~~ ~~{ iso(1) member-body(2) us(840)~~ ~~ieee802dot11(10036) MAC(1)~~ ~~attribute(7) ack_timeout(29) };~~	
136.	9.2.8	**BO**	**T**	**Y**	Text is intended to be explanatory but winds up being confusing. Delete it.	~~This policy induces some probability that another frame could be corrupted by the generated ACK. However if no ACK is returned because a busy medium is detected, then it is guaranteed that a retransmission results.~~	**Accepted**
137.	**9.3.3.4** **&** **11.4.4.1** **.26**	**WD**	**T**	**Y**	**The current definition of the CFP_Max_Duration limit is not sufficient to allow non-CF_aware stations to succesfully transfer data, with such transfer delays that are acceptable to higher protocol layers.** **Known values of such timeout mechanisms are in the 400-600 msec range, after which a protocol layer message is expected to be received. This means that a station should at maximum have an opertunity to send every 200 msec or so, otherwise the higher layer times out, and retransmits the same message with a limited maximum retry limit.** **Currently the CFP_Period can be specified as multiple integers of the DTIM interval, where the MIB default is set to 5.** **We need to specify that the CFP_Period should be limited to 200 msec maximum.** **Change the MIB defaults such that this setting would not violate the 200 msec maximum**	**Add to the end of section 9.3.3.4:** **The CFP_period shall be no larger then 200 msec to allow sufficient response time for a non-CF-Aware station to access the medium.** **Modify section 11.4.4.1.24:** **Change the default value to 1** **Modify section 11.4.4.1.26:** **Change the default to 2.**	**Declined to change clause 9 text, because changing the defaults in clause 11 accomplish the desired effect without stipulating this (author agrees that changing 11.4.4.1.26 to require the default value to be 1 satisfies the intent of the comment).**
138.	**9.2.6.3**	**maf**	**T**	**Y**	**The slop in various carrier detection mechanisms will cause a problem unless the CTS_TIMEout (and ACK_timeout) are either increased, or are specifically called out to be interpreted as frame reception must have *STARTED* by the timeout expiration.**	**CTS_Timeout - value should include enough time to allow for slop in my start of timer vs actual possible end of reception of CTS frame, otherwise, if the last bit of CRC32 is even one bit time late, then the timer will beat the frame, and I'll pretend that I never heard it and go into backoff**	**Modified text accepted by commenter**

						and waste bandwidth Add text to indicate exactly how to interpret CTS_Timeout - if a CTS frame type is detected before the end of the timeout, but the entire frame, including a CRC has not yet been detected, then do I cancel the timeout, or this CTS reception doomed to failure, because there is no hope that the last bit CRC will make it to the receiver before the timeout, because the transmission started just one teensy itsy bit time too late?	
139.	**9.2.6.3**	**maf**	**T**	**Y**		**ACK_Timeout - see previous comment on CTS_Timeout**	**Modified text accepted by commenter**
140.	**9.2.5.4**	**ch**	**t**		**requirement - needs to be 'shall' instead of 'will'**	Stations receiving a valid frame shall~~ould~~ update their NAV with the information received in the Duration field, but only when the new NAV value is greater than the current NAV value and only when the frame is not addressed to the receiving STA.	**Accepted**
141.	**9.2.5.4**	**sb**	**e**	**n**	**Need to specify behaviour of NAV for the multirate case. Two possibilities are apparent: (1) set the NAV to cover the max packet length plus ACK; (2) don't worry about it and let CCA play an active role. The later is what will happen for a corrupted frame (FCS error for example). It is also what will happen for a PS-Poll-Data-Ack since the data frame length is unknown. I think the best option here is to rely on CCA. This requires no change to the text because it already has 'valid frame' in the text, but might benefit from a clarifying note.**	**Clarification note on imperfections in NAV and reliance on CCA under certain conditions.**	**Accepted by MAC Group vote to adopt Multirate support as described in 96/79r1**
142.	**9.2.5.4**	**WD**	**T**	**Y**	**There is a problem with the current RTS/CTS NAV setting procedure. There are cases where a CTS does not follow an RTS as is expected when the RTS collides in the vicinity of the receiver, or when at the receiver the NAV is set, such that it prevents the transmission of a CTS. The effect of this is that all**	**Add the following text at the end beyond figure 42: Stations that did set the NAV upon reception of an RTS may undo this setting when they do not detect a subsequent Data frame after a RTS**	**Accepted with different wording to resolve exactly when the actions resulting from the timeout take place.**

					traffic around the transmitter is prevented, because the NAV is set in all stations, but the medium is not used for the subsequent data, because the CTS is missing. The only traffic that is then possible is the retransmission of the RTS, which may again be failing because no CTS is returned, thereby only extending the NAV setting. **In the original proposal there were provisions that would allow stations that do hear an RTS, but no subsequent Data after a RTS Timeout period to undo the previous setting of the NAV.** **It should be allowed to implement that MAC such that a station can undo such a NAV setting when it was caused by an RTS (or Data frame when fragmentation is used), but not when the update was done by a CTS. All stations that do hear the RTS will also hear the subsequent Data if it is there, so lack of Data traffic after the RTS Timeout (2*SIFS + CTS + Slot) is a valid condition to undo the previous NAV setting.**	**Timeout period following the received RTS which has a duration of 2*SIFS+CTS+Slot time.**	
143.	**9.2.5.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	condition of the medium. **Error! Reference source not found.** indicates the NAV for stations that may~~can~~ hear the RTS frame, while other stations may only receive the CTS frame, resulting in the lower NAV bar as shown (with the	**Acepted**
144.	9.2.9	**BO**	**E**		Edit for clarity.	A destination STA shall reject a frame as a duplicate frame, any frame that has the RETRY bit set in the Frame Control field and matches a <source-address, sequence-number and fragment-number> tuple of an entry in the cache .	**Accepted (editorial)**
145.	**9.2.9**	**ch**	**e**		**duplicate word, second paragraph**	Duplicate frame filtering is facilitated through the inclusion of a **Sequence Control Field** (consisting of a sequence number and fragment number) ~~field~~ within Data and Management frames.	**Accepted**

146.	**9.2.9 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	sequence number and fragment number) field within Data and Management frames. MPDUs which are part of the same MSDU shall have the same sequence number, and different MSDUs shall~~will~~ (with a high probability) have a different sequence number.	**Accepted**
147.	**9.2.9 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	There is the small possibility that a frame may~~will~~ be improperly rejected due to such a match; however, this occurrence would be rare and ~~will~~ simply results in a lost frame (similar to an FCS error in Ethernet).	**Accepted**
148.	**9.3.3.5**	**ch**	**e**		**punctuation and grammer**	Such a frame directed to a non–PCF station~~s~~ shall be acknowledged using an ACK Control frame sent after an SIFS (t~~T~~his is the same as these stations already do~~.~~).	**Accepted**
149.	**9.3.3.5**	**ch**	**t**	**Y**	**The first and second paragraphs contradicts the 2nd last paragraph of 9.3.3.1, which says that a CF-Poll can be answered with a Null Frame or a regular Ack. Which is correct?**		**Accepted, by adding Null to 9.3.3.5 list of allowable responses**
150.	**9.3.3.5**	**ch**	**t**	**Y**	**clarity and consiseness**	The PC shall not issue frames with a sub-type which includes CF-Polls if insufficient time remains in the current CFP to permit the polled station to transmit a Data frame containing a maximum length MPDU.	**Accepted**
151.	**9.2.5.5**	**ch**	**t**		**A lot of 'will' to 'shall', following the figure and delete some un-needed text**	When t~~T~~he source station transmits a fragment, ~~then releases the channel and waits for an acknowledgment. When the source station~~ it shall releases the	**Accepted in March**

						channel ~~following its fragment, it will~~ then immediately monitor the channel for an acknowledgment frame from the destination station. When the destination station has finished sending the acknowledgment, the SIFS following the acknowledgment shall be~~is then~~ reserved for the source station to continue (if necessary) with another fragment. The station sending the acknowledgment does not have permission to transmit on the channel immediately following the acknowledgment. The process of sending multiple fragments after contending for the channel is defined as a fragment burst. If the source station receives an acknowledgment but there is not enough time to transmit the next fragment and receive an acknowledgment due to an impending dwell boundary, it shall~~will~~ contend for the channel at the beginning of the next dwell time. If the source station does not receive an acknowledgment frame, it shall~~will~~ attempt to retransmit according to the backoff algorithm and ~~. When the time arrives to retransmit the fragment, the source station will contend for access in~~ the contention process~~window~~. After a station contends for the channel to retransmit a fragment of a MSDU, it shall~~will~~ start with the last fragment that	

						was not acknowledged. The destination station will receive the fragments in order (since the source sends them one at a time, in order). It is possible however, that the destination station may receive duplicate fragments. It shall be the responsibility of the receiving station to discard duplicate fragments. ~~This will occur if the destination station sends an acknowledgment and the source does not receive it. The source will retransmit the same fragment after executing the backoff algorithm and contending for the channel.~~ A station shall~~will~~ transmit after the SIFS only under the following conditions during a fragment burst: The station has just received a fragment that requires acknowledging. The source station has received an acknowledgment to a previous fragment, has more fragment(s) for the same MSDU to transmit, and there is enough time left in the dwell time to send the next fragment & receive an acknowledgment. The following rules also apply. When a station has transmitted a frame other than a fragment, it shall not transmit on the channel following the acknowledgment for that	

						frame, without going through a backoff. When an MSDU has been successfully delivered, and the station has a subsequent MSDU to transmit, then it shall go through a backoff. Only unacknowledged fragments are retransmitted. If a multiple fragment MSDU does not require an acknowledgment (for example, a broadcast/multicast packet transmitted by the Access Point), the source station shall~~will~~ transmit all fragments of the MSDU without releasing the channel, as long as there is enough time left in the dwell time. If there is not, the station shall~~will~~ transmit as many fragments as possible and recontend for the channel during the next dwell time. The spacing between fragments of a broadcast/multicast frame shall be equal to the SIFS period.	
152.	9.2.5.5	jz	t		I don't see that fragmenting broadcasts/multicasts serves any purpose. Since we can't retry them, their reliability is in fact reduced by adding all the extra header/CRC bits to their transmission. (Or is there some weird radio-physics thing that makes later bits in a frame more likely to get corrupted than early ones?)	Accepted See comment 6	Accepted
153.	**9.2.5.5**	**AS**	**t**	**y**	**If a fragment burst is interrupted the AP may not be able to resume sending the fragements if it has to transmit a beacon and possibly a CFP or broadcast frames.**	**Original Text:** Should the sending of the fragments be interrupted due to one of these reasons, when the next opportunity for transmission occurs the station shall	**Acepted, in effect but with different text, by MAC group vote to adopt 96/91r1. The new text resolves the conflict between requirement to start with the**

						resume sending the fragments . **Replacement Text:** Should the sending of the fragments be interrupted due to one of these reasons, the station shall resume sending the fragments at its earliest opportunity.	**same MPDU and other requirement to send buffered group-addressed frames.**
154.	**9.2.5.5**	**AS**	**t**	**y**	**The destination station will receive fragments for the same frame in order, but there may be an indeterminate number of fragmented frames received from the same station between two fragments of the same frame.**	**Original Text:** The destination station will receive the fragments in order (since the source sends them one at a time, in order). **Replacement Text:** The destination station will receive fragments of the same MSDU in order (since the source sends them one at a time, in order).	**Accepted**
155.	9.2.5.5	**BO**	**T**	**Y**	The rule is incomplete	When an MSDU has been successfully delivered or all retransmission attempts have been used, and the station has a subsequent MSDU to transmit, then it shall go through a backoff.	**Accepted**
156.	**9.2.5.5**	**TT**	**T**	**Y**	See Rationale in comment of Section 9.1.4 on not fragmenting broadcast frames.	Delete last paragraph of Section 9.2.5.5. ~~If a multiple fragment MSDU does not require.....~~	**Accepted**
157.	**9.2.5.5 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	MSDU have been sent, an acknowledgment is not received, or the station is restricted from ~~can not~~ sending any additional fragments due to a dwell time boundary. Should the sending of the fragments be	**Accepted**
158.	**9.2.5.5 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	When the source station releases the channel following its fragment, it shall~~will~~ immediately monitor the	**Accepted**
159.	**9.2.5.5 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	fragment and receive an acknowledgment due to an impending dwell boundary, it shall~~will~~ contend for	**Accepted**

160.	**9.2.5.5 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	If the source station does not receive an acknowledgment frame, it shall~~will~~ attempt to retransmit according to the backoff algorithm. When the time arrives to retransmit the fragment, the source station shall~~will~~ contend for access in the contention window.	**Accepted**
161.	**9.2.5.5 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	After a station contends for the channel to retransmit a fragment of a MSDU, it shall~~will~~ start with the last fragment that was not acknowledged. The destination station ~~will~~ receives the fragments in order (since the source sends them one at a time, in order). It is possible however, that the destination station may receive duplicate fragments. It shall be the responsibility of the receiving station to discard duplicate fragments. This may~~will~~ occur if the destination station sends an acknowledgment and the source does not receive it. The source shall~~will~~ retransmit the same fragment after executing the backoff algorithm and contending for the channel. A station shall~~will~~ transmit after the SIFS only under the following conditions during a fragment burst:	**Accepted**
162.	**9.2.5.5 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	If a multiple fragment MSDU does not require an acknowledgment (for example, a broadcast/multicast packet transmitted by the Access Point), the source station shall~~will~~ transmit all fragments of the MSDU without releasing the channel, as long as there is	**Accepted**

						enough time left in the dwell time. If there is not, the station shall~~will~~ transmit as many fragments as possible and recontend for the channel during the next dwell time. The spacing between fragments of a broadcast/multicast frame shall be equal to the SIFS	
163.	9.2.5.5	jz	T	**Y**	Multicast/Broadcast reliability is compromised by the power save mechanism. We should adopt the mechanism is 96/15 and 96/16 to fix this. My "No" vote will only change to a "Yes" vote if we adopt these changes or else mandate the use of a stripped-down PCF to enhance multidestination reliability.	<<Adopt changed text for this section from 96/15 and 96/16.>>	**Unresolved. Plenary did not accept MAC group recommendation to adopt 96/15 and 96/16.**
164.	**9.2.5.5**	**TT**	**T**	**Y**	See Rationale in comment of Section 9.1.4 on not fragmenting broadcast frames.	Delete last paragraph of Section 9.2.5.5. ~~If a multiple fragment MSDU does not require.....~~	**DUPLICATE**
165.	9.2.5.6	jz	E		The diagram is yucky. The NAVs are all one big black blob. It should be redrawn to clarify (in black/white) which parts of the NAV came from which frames' Duration field. Also, need to substitute "0" for "1" throughout the second paragraph.		Declined Turn on gray scale while printing.
166.	**9.2.5.6**	**WD**	**E**	**n**	**Change the fill pattern in figure 44 to show the actual NAV durations, and the RTS frame.**		**Declined** **Turnn on gray scale while printing**
167.	**9.2.5.6 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	The following is a description of using RTS/CTS for the first fragment of a fragmented MSDU. RTS/CTS may~~will~~ also be used for retransmitted fragments if their size warrants it. The RTS/CTS frames define the	**Accepted**
168.	**9.2.5.6 A.4.4**	**db**	**T**	**Y**	**w/o the requested change the Draft is technically incorrect - since approved "standard" language was not used the draft does not corectly convey operational requirements.**	Each frame contains information that defines the duration of the next transmission. The RTS shall~~will~~ update the NAV to indicate busy until the end of ACK 1. The CTS shall~~will~~ also update the NAV to indicate busy until the end of ACK 1. Both Fragment 1 and	**Accepted**

						ACK 1 shall~~will~~ update the NAV to indicate busy until the end of ACK 2. This is done by using the duration field in the DATA and ACK frames. This shall~~will~~ continue until the last Fragment which has a duration of one ACK time plus one SIFS time and its ACK which shall~~will~~ have the duration set to zero. Each Fragment and ACK acts as a virtual RTS and CTS, therefore no RTS/CTS frame needs to be generated even though subsequent fragments are larger the aRTS_Threshold. In the case where an acknowledgment is not received by the source station, the NAV shall~~will~~ be marked busy for next frame exchange. This is the worst case situation. This is shown in **Error! Reference source not found.**. If the acknowledgment is not sent by the destination station, stations that may~~can~~ only hear the destination station shall~~will~~ not update their NAV and be free to access the channel. All stations that hear the source shall~~will~~ be free to access the channel after the NAV from Frame 1 has expired.	

Seq. #	Section number	your ini-tials	Cmnt type E, e, T, t	Part of NO vote	Comment/Rationale	Corrected Text	Disposition/Rebuttal