Simplified CF-Async Service

- Michael Fischer
  Digital Ocean, Inc.

CF-Async Fundamentals

- CF-Async is a delivery modality, not a service class.
  - Both contention-based Async and CF-Async deliver frames for the "asynchronous data" service class.
  - Time-Bounded Services involve one or more, service classes.
- CF-Async differs from contention-based async in media access and control communication:
  - Access to the medium for CF-Async transmission is controlled by the PCF. Stations receive permission to transmit when "polled" by the PCF.
  - CF-Async frames are delivered without contention delays and backoffs, but only during the CF-Period of each superframe.
  - CF-Async acknowledgements and polls are (generally) piggybacked on the headers of CF-Async data frames.

Things to Change About CF-Async

1. Unify asynchronous data handling:
   - Eliminate the separate CF frame type.
   - Adopt consistent acknowledgement timing.
   - Define rules under which all active stations can receive asynchronous data frames during the contention period and the contention free period.
2. Correct some problems:
   - Fix a mechanism broken in 20b3.
   - Fix a latent flaw present in 20bX.
   - Clarify the rules for usage of the CF-period.

Eliminating the CF Frame Type

CF-Async does not need a separate frame type:

- By defining some new subtypes, all asynchronous data frames can use the "Asynchronous" data type.
- This simplifies the use of either contention-based async or CF-Async for data frame transmission at stations that support both delivery modalities.
- This creates a reserved frame type, which may be useful for other service classes. (Time-Bounded Service may be able to benefit from using the spare frame type.)

Using a single frame type simplifies asynchronous data delivery and allows simpler control state machines, especially for IFS and Ack generation.

NOTE: The terms "CF-up" and "CF-down" that still appear in the draft are strictly descriptive (meaning to and from the PCF, respectively). The adoption of 94/236 eliminated the CF-up and CF -down frame types (which are unnecessary).

What is Optional About CF-Async?

- The ability for a station to operate as the PCF is optional.
- The options concerning contention-free "service" pertain to transmission during the CF-Period.
  - All stations can receive data frames sent as CF-Async.
  - Stations may optionally transmit data frames when polled by the PCF.
  - Stations that are not "CF–Aware" acknowledge CF-Data frame reception identically to contention-based data frames (by sending an ACK control frame).
- The elimination of the separate CF frame type simplifies the reception of asynchronous data frames by non–CF–Aware stations.

Fixing a Broken Mechanism

Draft changes from 94/236 broke a PCF mechanism:

- The CF-Poll frame control bit was eliminated, using the receipt of a CF frame from the PCF as an implicit CF-Poll.
- Without an explicit CF-Poll function, the PCF has no way to acknowledge a received CF frame without granting access to the medium. This means that, under certain conditions,
  THE CF–PERIOD MIGHT NEVER END!

The best way to correct this is to use one of the Subtype bits to encode the "CF-Poll" function. This is detailed in a subsequent slide.

Fixing a Latent Flaw

The PCF description in 93/20bX contains a flawed mechanism:

- The communication of CF-Poll and CF-Ack indications, when there is no CF-Data frame to transfer, is specified to occur using a CF-Data frame with a zero–length payload.
- This is a bad approach:
  - The receipt of a data frame generates an indication to the receiver's LLC, even for zero–length payloads.
  - There is no mechanism to distinguish zero–length MSDUs provided by the transmitting LLC and zero–length CF-Data frames generated by the MAC.

The best way to correct this is to use one of the Subtype bits to encode a "no payload" function. This is detailed in a subsequent slide.

Simplified IFS Usage During the CFP

The alternating use of PIFS and SIFS during the CF-Period is unnecessary and inefficient:

- The PIFS is required only in two situations:
  - At the beginning of the CF-Period:
    To allow the PCF to gain control of the medium.
  - During the CF-Period, when the recipient of a CF frame from the PCF fails to respond within an SIFS interval:
    To allow the PCF to retain control of the medium.
- The SIFS should be used between all other frames during the CF-Period:
  - The control state machine is simplified because all acknowledgements, whether in ACK control frames or piggybacked in CF-Data+Ack frames, are expected an SIFS–duration after the associated data frame.
  - The time required to send a given set of frames during the CF-Period is reduced.

The Async Data Subtype Field

Async Subtype:

Reserved	No–Data	CF–Poll	CF–Ack

Subtype	Sent By	Meaning
0000	any STA	Data (same as 20b3)
0001	any CF-STA	Data+CF-Ack (same as 20b3)
0010	PCF	Data+CF-Poll
0011	PCF	Data+CF-Ack+CF-Poll
0100	none (any?)	null function
0101	any CF-STA	CF-Ack (no data)
0110	PCF	CF-Poll (no data)
0111	PCF	CF-Ack+CF-Poll (no data)
1xxx	none	reserved

- Non–CF–aware stations only need to test for "No–Data = 0" in the Subtype field of Async frames.
- Each CF–related combination already needs to be handled by the PCF and/or CF–aware stations (see 94/207A).

CF-Period Usage Rules

1. Only Data frames (and resulting Ack frames, if any) shall be sent during the CF-period. All management frames shall be sent during the contention period.
2. A PCF may send Async Data frames to any active station (not to stations in PSP or PSNP mode).
   - CF-aware stations shall acknowledge receipt of each Async Data frame (from the PCF) that has CF-Poll=1 using CF-Ack=1 in a Data frame (possibly with No-Data=1), sent after an SIFS–interval; and shall acknowledge the receipt of all other Async Data frames using ACK Control frames sent after an SIFS–interval.
   - Non-CF-aware stations shall acknowledge receipt of (all) Async Data frames using ACK Control frames sent after an SIFS–interval. (This is the same as these stations already do for contention–based async.)

CF-Period Usage Rules (cont.)

3. When polled by the PCF (CF-Poll=1 in the header of a directed Data frame), a CF-aware station may send one Data frame to any destination.
   - Such a frame directed to or through the PCF station shall be acknowledged by the PCF, using CF-Ack=1 in a Data frame (possibly with No-Data=1), sent after an SIFS–interval.
   - Such a frame directed to non–PCF stations shall be acknowledged using an ACK Control frame sent after an SIFS–interval. (This is the same as these stations already do.)
   - A polled CF-aware station with neither a Data frame nor acknowledgement to send shall not respond, permitting the PCF to resume transmission after a PIFS–interval.
4. The PCF shall not send Data frames with CF-Poll=1 if insufficient time remains in the current CF-Period to permit the polled station to transmit a Data frame containing a maximum–length MPDU.

Digital Ocean Simplified CF–Async IEEE P802.11–94/252A

MOTION #1

Slide 13

1. To restore the explicit CF-Poll capability lost in 93/20b3, using a bit in the Frame Subtype, as defined in document 94/252.

(motion not made due to approval of Motion #4)

Digital Ocean Simplified CF–Async IEEE P802.11–94/252A

MOTION #2

Slide 14

2. To eliminate the use of zero–length CF-Async frames for polling and acknowledgement when there is no CF-Async data, using instead a "no data" bit in the Frame Subtype, as defined in document 94/252.

(motion not made due to approval of Motion #4)

Digital Ocean Simplified CF–Async IEEE P802.11–94/252A

MOTION #3

Slide 15

3. To adopt the changes to contention free asynchronous operation, including frame types/subtypes, usage rules, IFS durations, etc. as defined in document 94/252.

(motion not made due to approval of Motion #4)

Digital Ocean Simplified CF–Async IEEE P802.11–94/252A

MOTION #4

Slide 16

4. To adopt the changes to contention free asynchronous operation, including frame types/subtypes, usage rules, IFS durations, etc. as defined in document 94/252; and to modify the updates to section 4.1.2.1.2. to desginate the "reserved" frame type (type code =11) as "time bounded" frame type.

(this motion was approved in MAC group 18-2-12)