GFP Considerations for RPR

Angela T. Faber
afaber@telcordia.com

Agenda

- GFP Background
- Why GFP?
- GFP Core Header
- GFP Payload Area
- GFP Options
  - Signal Adaptation (Transparent GFP and Frame-mapped GFP)
  - Error Detection
  - Extension Header (Ring-based and Null)
- Conclusions

GFP Background

- T1X1.5 Generic Framing Procedure (GFP) Draft Revision 4 (T1X1.5/2001-024R4)
  - "GFP provides a generic mechanism to adapt traffic from higher-layer client signals over an octet synchronous transport network. Client signals may be PDU-oriented (such as IP/PPP or Ethernet MAC), block-code oriented (such as Fibre Channel or ESCON), or a constant bit rate stream."
  - GFP is used to delineate octet-aligned, variable-length payloads from higher-level client signals for subsequent mapping into octet-synchronous payload envelopes such as those defined in ANSI T1.105.02 (SONET) and ITU-T G.709 (OTN).

Why GFP? (1)

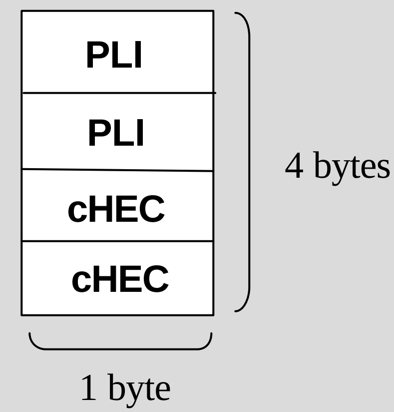
- RPR can use GFP for mapping RPR packets into SONET/SDH
  - GFP would provide frame delineation
  - GFP may add idle frames depending on the rate of RPR packets and the rate of the SONET/SDH signal
- GFP framing is expected to be THE standard for mapping any PDU-based signal into a Constant Bit Rate server layer (e.g., SONET/SDH)
- But RPR can also use Packet over SONET/SDH (POS) for that

Why GFP? (2)

- Main advantages of GFP compared to POS
  - Bandwidth Expansion
    - Deterministic bandwidth: Byte-stuff HDLC has unpredictable bandwidth inflation due to the need for escape characters whenever the client data emulates flag/control characters. But it is not a big issue in networks.
    - Network vulnerability standpoint: Malicious user can send max. length frames with payloads consisting entirely of flag/control characters, thus virtually doubling the bandwidth required by that packet. GFP prevents that possibility.
  - Ability to multiplex different protocols
    - GFP allows the multiplexing of multiple protocol or multiple instances of the same protocol onto the same SONET/SDH interface (while POS allows multiplexing of different protocols onto the same SONET/SDH interface)
    - However, multiplexing of RPR signals with other client signals is not an advantage if delay-sensitive clients are being supported in RPR

GFP Core Header

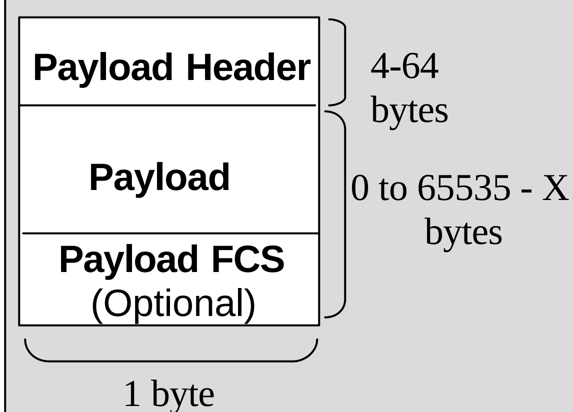
- GFP Core Header is composed of PDU Length Indication (PLI) and Core Header Error Check (cHEC)
  - used for frame delineation
- PLI contains the GFP payload length (in octets)
  - GFP uses the PDU Length to find the end of the GFP frame (for delineation)
  - RPR will have to pass its packet length to GFP
    - Do we have situations where RPR may not know the size of its packet?
    - If so, can RPR layer simply not tell the packet length and leave it for GFP? (i.e., GFP buffers it and check the length)



GFP Core Header Format

GFP Payload Area

- GFP Payload Area consists of Payload Header and Payload field, with an optional Payload Frame Check Sequence (FCS).
- Although Payload Area supports PDUs up to 64K, GFP implementations should support reception of GFP frames with GFP Payload Areas of at least 1600 bytes
  - RPR packets will probably be under 1600 bytes
- Problem if Jumbo frames are allowed
  - GFP draft v4 has that “prior arrangements between two GFP implementations will be needed”
  - However, GFP has no negotiation mechanisms for that (and this is considered to be a client issue)
  - Do we need fragmentation?



GFP Payload Area Format

GFP Options

- GFP provides options in terms of signal adaptation, Error detection (FCS) and Extension Header
  – RPRWG has to decide which one is most applicable to RPR applications
- Signal Adaptation
  – GFP supports both Transparent-GFP and Frame-mapped GFP
- Error Detection
  – GFP provides a FCS for its payload, which can be turned on or off
- Extension Headers
  – Depending on the application, GFP provides (Linear), Ring-frame, and Null Extension Header
- Next slides will provide some advantages and disadvantages of those options, when used for RPR

Signal Adaptation - Transparent GFP (1)

- Intended to facilitate the transport of 8B/10B block-coded client signals for scenarios that require very low transmission latency
- Client/GFP adaptation function operates on the coded character stream
  - That means that GFP sends/receives bit streams to/from higher layers instead of PDUs
- Currently Transparent GFP supports only 8B/10B coded signals
- Theoretically it is possible to use transparent GFP in mapping the RPR client layer signal into SONET/SDH

Signal Adaptation - Transparent GFP (2)

- The following functions are needed if Transparent GFP is used
  - map the RPR packets into a 8B/10B physical signal (e.g., 1 GbE)
  - and then map this signal into a SONET/SDH signal using Transparent GFP
  - However, frame-mapped GFP with direct access to the RPR packets saves RPR the intermediate Ethernet PHY processing and line coding overhead
- RPR add/drops packets at the ring nodes and therefore requires access to the packet structure
  - RPR need to perform frame delineation if Transparent GFP is used
- What is the purpose of Transparent GFP for RPR?
  - RPR would be already providing frame delineation
  - The signal coming from RPR toward SONET/SDH would be a bit stream that could be mapped into the SONET/SDH payload

Signal Adaptation - Frame-mapped GFP

- Frame-mapped GFP uses a PDU-oriented client signal adaptation
- e.g., IP/PPP, Ethernet MAC
- Client/GFP adaptation function operates on the incoming client PDU
  - RPR layer sends/receives the RPR packet, i.e., no frame delineation is required for RPR to perform on the GFP outcome
- More efficient for RPR than Transparent GFP
  - No intermediate Ethernet PHY processing and line coding overhead (like the Transparent GFP case)

Error Detection - FCS (1)

- GFP provides an optional FCS [on/off] to protect GFP payload (i.e., RPR packet)
  - [on]: allows GFP to check if payload (i.e., RPR frame) is corrupted
  - [off]: corrupted RPR packet will only be checked at RPR layer
- T1X1.5 has not defined yet what to do with the GFP frame once it detects that the payload is corrupted (FCS [<u>on</u>])
- Allowing GFP to detect corrupt payload (and probably take action upon it, e.g., discarding GFP frame) may not give a chance to RPR to act upon it
  - If RPR packet is corrupted only in the payload, RPR may still want to deliver it
  - RPR will not be able to use it for monitoring the quality of the signal (based on corrupted CRC count accumulated on the node)

Error Detection - FCS (2)

- If FCS is [on], GFP can use it for span management purposes
  - Let GFP do span management (i.e., FCS [on]) but request that GFP do not discard the frame
  - is it feasible for GFP to detect corrupted payload but not discard it?
- However, RPR cannot rely on signal degradation to be detected at the physical layer
  - Even though GFP may be capable of providing detection for signal degradation via the FCS [on] capability, when using GbE as the physical layer such functionality may not provided
  - RPR should be able to do span management too
- Therefore RPR can have GFP FCS [on] as long as GFP do not discard it
  - otherwise turn it [off]

Extension Header - Ring Frame

- Ring Frame Extension Header allows for multiplexing of RPR packets together with other GFP client signals onto a single SONET/SDH interface
  - Multiplexing removes the total control of the bandwidth that RPR is planning on having
    - GFP provides no bandwidth reservation or priority capabilities, i.e., there is no way to guarantee capacity to the RPR client
    - Negative impact for bandwidth management mechanisms that are trying to arbitrate medium access
    - Also not good for delay bound control of certain classes of service, since the transmission time will depend on the traffic of all the client layers multiplexed together via GFP
- Ring Frame Extension Header allows RPR to use the ring frame Extension Header as the RPR header (i.e., frame for RPR)
  - Does it provide support for all functions RPR is planning on having?

Extension Header - Null

- Null Extension Header applies to logical point-to-point configuration
- Intended for scenarios where transport path is dedicated to one client signal
  - no multiplexing of client signal (Good!)
- Would that allow for simpler RPR implementation?
  - Can RPR have the same frame format whether it will be used with POS or GFP?
  - This may allow vendors to have the same RPR MAC implementation whether mapping POS or GFP

Extension Header - New Ring Frame

- We can get a new Ring Extension Header for RPR
  - It will support RPR functions better than the Ring Extension Header
  - No multiplexing of GFP higher client signals together with RPR
  - The new Ring Extension Header would actually be the RPR frame
    - Would that be more efficient for RPR?
    - Instead of Null Extension Header + RPR Packet (including RPR header + payload) it would have the RPR header INSIDE the Ring Extension Header + RPR payload
  - If it is actually more efficient, there is a trade-off between this and the simplicity of the Null Extension Header
  - Discussions, discussions, discussions!

Conclusions

- We need to discuss more these issues.
  - Form an ad-hoc group to discuss it during the meeting
- Initial Conclusions
  - If GFP does not discard corrupted frames, use FCS [on] (still to be decided), otherwise turn it [off]
  - For signal adaptation, Frame-mapped GFP should be used rather than Transparent GFP
  - RPR should use the GFP Null Extension Header rather than the Ring Frame (or Linear) Extension Header

References

- T1X1.5 Generic Framing Procedure (GFP) Draft Revision 4 (T1X1.5/2001-024R4)
- T1X1.5 Mailing List discussions