FECFRAME Introduction

Marshall Eubanks
(with help from Greg Shepherd and Mark Watson)
IETF-66

Why FEC ?

- Forward Error (or Erasure) Correction (FEC) is a means of correcting packet loss through built-in redundancy
  - Multicast, UDP Streaming will not protect from packet loss.
  - If all data was on dedicated wireline networks, then a simple "Checksum" approach would suffice
    - Say, for every N packets, XOR them and send that as the Nth+1 packet - this will work well if the packet loss rate is << 1/N or less, and packet losses are Poisson in nature
  - Real networks are not so kind…

Why a FEC Framework WG ?

- Forward Error (or Erasure) Correction (FEC) is a means of correcting packet loss through built-in redundancy
  - Multicast, UDP Streaming all will not protect from packet loss.
- The RMT WG has produced FEC Building Blocks for FEC of streams
  - draft-ietf-rmt-fec-bb-revised-03.txt
  - Basically at the application layer
- There is great interest in applying this at the transport layer
  - The first task is to set up a Framework.
  - Next task will be to set up Schemes to instantiate that framework.
- Both together will give us FEC Transport

Terminology

- **Source data flow**: The packet flow or flows to which FEC protection is to be applied.
- **Repair data flow**: The packet flow or flows carrying forward error correction data
- **Source protocol**: A protocol used for the source data flow being protected - e.g. RTP.
- **Transport protocol**: The protocol used for transport of the source data flow being protected - e.g. UDP (unicast or multicast), DCCP.
- **Application protocol**: Control protocols used to establish and modify the source data flow being protected - e.g. RTSP.
- **FEC Code**: An algorithm for encoding data such that the encoded dats flow is resilient to data loss or corruption.
- **FEC Scheme**: A specification which defines the additional protocol aspects required to use a particular FEC code with the FEC framework, or (in the context of RMT), with the RMT FEC Building Block.
- **Source Block**: the group of source data packets which are to be FEC protected as a single block
- **Protection amount**: The relative increase in data sent due to the use of FEC.

Architecture question:

What should be the split of responsibility between FEC Framework and FEC Scheme ?

FEC Framework

Common to all FEC codes

Preferably “IPR free”

FEC Scheme

Specific to one FEC code

May include IPR

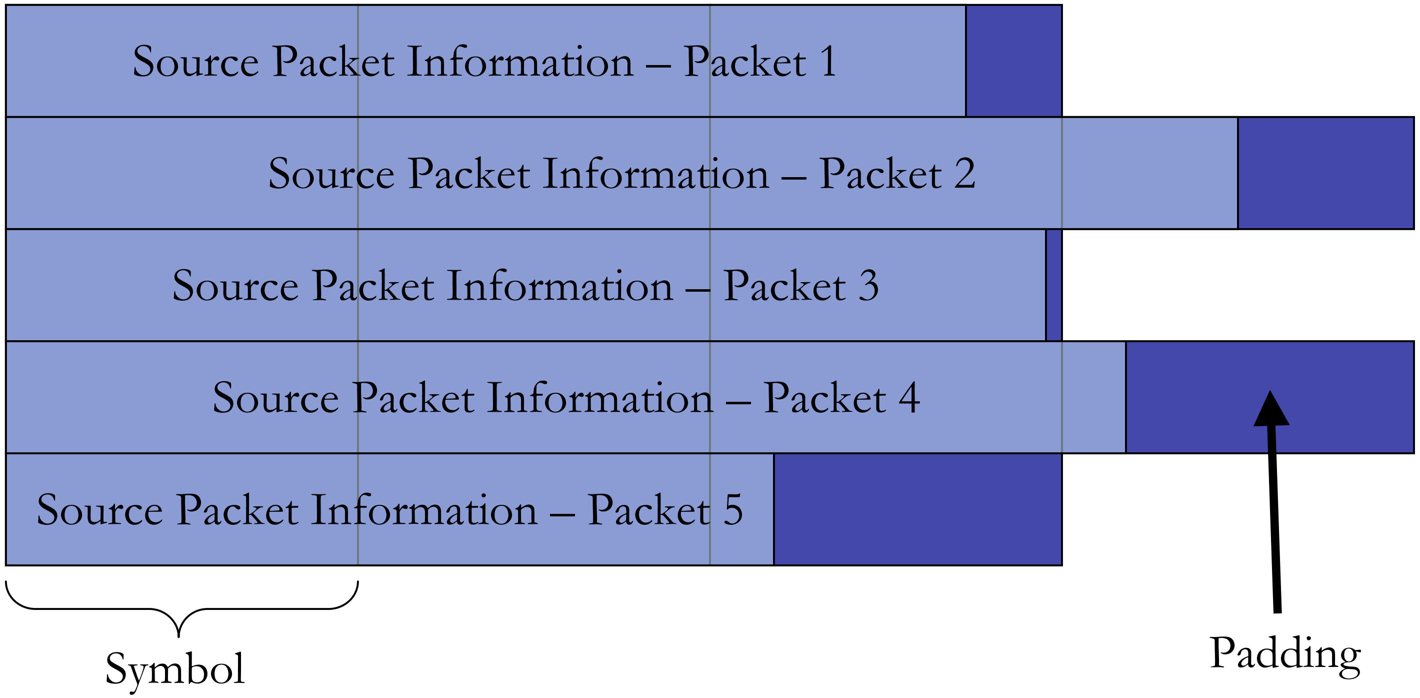
Work Elsewhere

- What are the potential customers for FEC Transport ?
- Multicast or Unicast IPTV
  - High bit rate, relatively low loss regime
  - There is a new SMPTE standard being voted on
    - Not suitable for general use, tailored for MPEG-2
- Wireless streaming
  - There is a new 3GPP FEC standard
    - From Digital Fountain
  - One goal is to have IETF FEC Transport scheme have the 3GPP standard as a special case.

Construction of source block

- In 3GPP specification, the FEC framework is responsible for constructing a source block
  - Source block is a sequence of fixed length "symbols"
  - Padding added to each packet so that packets start on symbol boundaries
  - Symbols are passed to FEC Scheme for encoding/decoding

Source block example
(3GPP framework)


Source Packet Information – Packet 1
Source Packet Information – Packet 2
Source Packet Information – Packet 3
Source Packet Information – Packet 4
Source Packet Information – Packet 5
Symbol
Padding

FEC Framework knows about "symbols" and adds padding to form source block:
Source block is the concatenation of the 17 symbols above

Alternative proposal

- Responsibility for "padding" and knowledge of "symbols" transferred to FEC Schemes
- Information passed from FEC Framework to FEC Scheme is just the "Source Packet Information" for each packet
    - Symbol size, padding strategy etc. left to FEC Scheme

Next steps

- Had our first meeting
  - Seems to be interest in this work.
- Update requirements draft
- Start work on architecture
  - Add to requirements draft ?
  - New draft ?