FEC BB Evolution

Framework

- **FEC BB will be resubmitted as draft to move forward to "Standard Track"**

- **Experience with it is showing some issue:**
  - *no good guidance on FEC ID drafts*
  - *probably missing pieces of general-purpose specification. E.g. the splitting of objects into payload defined in Flute.*

- **A primary goal is to update the FEC-BB in a backward-compatible way.**
  - *may have to change dependent RFC (Flute ..)*
  - *must keep backward compatibility "on the wire" for already defined protocols*

FEC BB Evolution -- Objectives

Fixes

- **Require that a Fully-Specified Encoding ID defines**
  - encoding and decoding algorithms
  - FEC Payload ID (packet format)
  - FEC OTI (fields names and description)
  - algorithm for partitioning objects into source blocks
- **Add general-purpose algorithm to the FEC BB for re-use in FEC ID specs.**
  - suggested algorithms for partitioning an object into multiple source blocks.
  - ...
- **Define the FEC OTI in a more rigorous way**
  - list parameters names and clear description of each
  - foster name reuse across FEC IDs (for parameters that have the same semantic)
- **Make clear that an FEC payload may contain multiple FEC symbols**
  - provide guideline on how to use the FEC Payload ID to identify all the encoding symbols in the payload

FEC BB Evolution -- Objectives

Fixes (Cont.)

- **Provide guidelines on how to write FEC IDs**
  - list required specification and uniform formatting / style
  - require separation of FEC ID specification (one per RFC or one per section)
- **Clarify the semantic of under-specified FEC IDs**
  - FEC Instance ID is not the next level in the hierarchy but a (Encoding ID, Instance ID) defines the actual scheme
  - Clarify the requirement for re-use of the FEC Payload ID format
  - Workout the requirements for OTI reuse and possible per-instance ID customization

FEC BB Evolution - Objectives

Enhancements

- **Allow for systematic codes to define two different FEC Payload ID within the same FEC Encoding ID (one for original source symbols and one for the repair symbols)**

  - ■ *It is the responsibility of the upper layer protocol to distinguish source from repair packets.*

- **Add specification to LCT on how to achieve the above (reserved header bits)**

  - ■ *ALC, FLUTE .. will inherit this automatically*

- **Revise the FEC Encoding ID range to allow more than 256 IDs**

FEC BB Evolution - Objectives

Streaming Enhancements

- **Explicitly allow the use of certain FEC Encoding IDs for streaming / dual use (e.g. for integration with RTP)**
- **Workout a strategy for defining "Streaming Profiles" for FEC Encoding IDs.**