Event Structure and Reprocessing

Ray Cowan

Ground Rules:

- Objects and tree headers are never deleted individually:
  - Deletion may occur only if the owning event is deleted.
- Events in a collection may be extended but not modified.

We distinguish two cases:

- Input Collection `==` Output Collection (event extension).
- Input Collection `!=` Output Collection (event cloning).
- Events may not be modified once written; instead, they may be:
  - *Extended:* new objects not previously present may be added.
  - *Cloned:* an event may be fully or partially copied (cloned) maintaining references back to headers in the original event.

This requires a bit of bookkeeping:

- Objects must maintain ownership information and borrower information (a reference count).
  - A object may have one and only one owner, of course.
  - Must enforce that only the owner can extend a header, not a borrower.
  - Borrower information may only be updated if update access is authorized.
  - Update access follows System/Group/User authorization: higher authority can write into lower authority but not the other way around.

- More work is needed to investigate the owner/borrower/use count situation.

Other considerations regarding cloning:

- Should it be possible to specify which headers to copy in a clone?
  - Non-copied headers would not exist in the new clone.
  - These headers could be added later with new data (renewed).
  - A mechanism to specify which headers to clone would be needed.
- Should there be "built-in" rules at clone time to enforce proper relationships between cloned and renewed headers?
  - Example: if the tracking header is removed in the clone, should all headers downstream be forcibly renewed?
  - Probably too complex to enforce. Instead depend upon reasonable usage by the user.
- Will need protection against duplicating an event in an output collection;
  - Requires an event ID and set semantics. Not for MDC2.

Database state information:

- What if a job crashes while writing events to the database?
  - Must be able to recover: need to maintain other meta-data about the contents of an event collection.
  - Must be able to scan (iterate) over an input collection and find the last processed event added to it. A possible name: BdbInspector?

Limitations for MDC2:

- If input collection `!=` output collection can only clone with full copy:
  - There is not yet any way to specify which headers to renew.
  - Can only extend with a new header that does not yet exist.