Open issues from meeting

- There are two (in my mind) completely independent issues to resolve:
    - Are TrackSegment/TrackSegmentCollection the best way to store projections?
    - What is best way to implement detector/surface ID?

Storing track projections

Option 1

- Use existing TrackSegment
- Requires modified TrackPoint with surface ID

Option 2

- Create datatype ProjectedTrackPoint
  - Members:
    - Most TrackPoint members (e.g. position, momentum)
    - One-to-one relation to Track
    - Surface ID
  - This would allow a "flat" collection of points (not nested in TrackSegment)

Recommendation: modify TrackPoint and use existing TrackSegment. If desired, we can change the name to reflect usage (e.g. TrackProjection), but this is a very trivial change that we should not waste time debating.

Projection detector/surface ID

- Option 1 (existing PR): single 32-bit member, bits split between detector ID and surface ID
    - Requires well-defined bit fields, not the most transparent option
- Option 2: two 16-bit members for detector ID and surface ID
- Option 3: single 64-bit member for ACTS surface ID, use open 8 bits for detector ID
    - Limits usability outside of EICrecon, does not allow for arbitrary projections surfaces

Recommendation: add two 16-bit members to TrackPoint. One holds a detector ID as defined in epic/compact/definitions.xml. The other holds a surface ID (definition may depend on user needs/application).