ATLAS-SOFT-2002-004 ATLAS-DAQ-2002-014 ATLAS-ID-2002-008 June 6, 2002

Requirements for an Inner Detector Event Data Model

S. Armstrong (editor),¹ M. Elsing, D. Froidevaux, I. Gavrilenko, R. Hawkings, N. Konstantinidis, A. Poppleton European Organization for Nuclear Research (CERN) - EP Division

> W. Wiedenmann University of Wisconsin - Madison Department of Physics

 $^{^{1}}$ contact: Stephen.Armstrong@cern.ch

1 Introduction

This document summarizes requirements of an Event Data Model (EDM) for the Inner Detector (ID) suitable to the needs of existing Offline Reconstruction Packages (ORPs) (e.g., xKalman++ [1] and iPatRec [2]) and the High Level Trigger Selection Software (HLTSSW) [3]. The dual nature of these requirements stems from the need to run Offline algorithms in the HLTSSW environment and *vice versa*.

The scope of the EDM for the ID covers all of the data entities in an event and their relationships with each other. In the context of the Athena/Gaudi object-based paradigm separating *knowledge objects* from *data objects* [4, 5], the EDM specifies the nature and content of data objects, referred to as Event Data Objects (EDOs) which are passed between algorithms and software packages to communicate information about the event and hence are potentially persistifiable. A corollary to this approach is that data object classes contain minimal algorithmic content (*e.g.*, algorithms for finding, following, or fitting tracks are separated from methods in the class definition of Track itself).

Since existing ORPs function within the context of the ATLAS detector and its simulation, they should, in principle, draw upon common EDM components. Ideally the Requirements Capture and Design for such an EDM would be independent of any one package or packages. However, our goal is a Requirements Capture which maintains compatibility with and draws upon the development experience of the existing ORPs.²

Existing ORPs have developed almost independently of each other, and, in many cases, crucial design and implementation details were initially made purposefully orthogonal to create a robust environment that provides inherent cross-checking. Existing ORPs have already implemented components of an ID EDM (and Detector Description, see below), but these components are incompatible [6]. Further complications arise due to the fact that the ID consists of not one but three independent sub-detectors (*i.e.*, the Pixel Detector (Pixels), the Semiconductor Tracker (SCT), and the Transition Radiation Tracker (TRT)), and clients of ORPs include both physics analyses and alignment procedures [7] in the Offline environment, and HLTSSW in the Online environment [6, 8].

An important distinction exists between the EDM and what is referred to as *Detector Description* (DD). In this document, we assume that the components of the EDM are objects which are event features or participate directly in their formation (*e.g.*, Raw-DataObjects, Clusters, Tracks) while the components of the DD are separate objects and external services. However, these two entities are closely coupled with correlated requirements, and existing ORPs have implemented their own versions of DD upon which their EDOs closely rely. Hence, this document discusses DD components, which overlap with the EDM (DetectorElements and their groupings), as well as relevant DD services upon which these components are highly dependent (services for DetectorGeometry and

 $^{^{2}}$ In other words, a prime Use Case implicit throughout this document is that both xKalman++ and iPatRec use the same EDOs of a new ID EDM.

PassiveMaterial).

With the definitions and requirements of EDM components in hand, a coherent design and implementation of the EDO classes (or modifications to existing EDO classes) representing these components may be achieved. This would represent the general ATLAS ID EDM rather than a single ORP-view or version of it. It is anticipated that existing ORPs will migrate to these new EDM classes internally as the requirements and design will have been oriented towards them (*i.e.*, the ultimate long-term goal is that tracking algorithms make **direct** use of these classes and not rework or convert them into specialized internal classes).

An implicit assumption throughout this document is that the use of EDM components should not violate the computational resources of the High Level Trigger (HLT) or Offline environments (*i.e.*, the trade-off between computational time and storage media space). For the HLT, under present assumptions, this translates into access and processing of a full event will not exceed the maximal Event Filter (EF) trigger latency period of 1s extrapolated to the HLTSSW running on a 4 GHz Pentium processor with several GBytes of main memory.³ The complementary Offline assumption is that the EDM should not preclude compact persistency of EDOs.

Extensive work has been done on general ID EDM components and performance issues related to them [9, 10]. Where suitable, these classes could be used directly or with minimal modification without "re-inventing the wheel." However, crucial issues related to Analysis and Design of the ID EDM (*i.e.*, a class or package of classes meeting the requirements articulated in this document do not necessarily have a suitable design) cannot be neglected and should be the topic of a future document(s).

Furthermore, although this document concentrates on requirements rather than implementation issues, an awareness of potential restrictions of the Transient and Persistent Event Stores [5] should accompany the design and implementation phases of the EDM (e.g., achieving bi-directional associations).

Finally, the requirements of both ID and Muon Systems could be merged into a common EDM framework. Along these lines, several classes are already shared between the ORPs iPatRec (ID) and Moore (Muons) [11]. It is hoped that future work can yield further EDM unification [12]

The subsequent sections of this document are organized as follows. Section 2 presents a definition of terms. Section 3 consists of subsections each of which concentrates on the requirements of a single component class of the ID EDM. Section 5 discusses requirements for Object Organization and Relationships. Finally, Section 6 provides some exemplary Use Cases.

 $^{^{3}\}mathrm{There}$ are also CPU time restrictions for Offline reconstruction; however, the HLT requirement is the more stringent.

2 Definition of Terms

This section defines, in conceptual order, terms relevant to the ID EDM. Terms with a † next to them are the EDOs.

- **RawData**: Read-Out Buffer (ROB)-formatted data produced by the ATLAS detector or its simulation.⁴
- RawDataObject(RDO)[†]: Uncalibrated RawData converted into an object representing a set of readout channels.⁵ Historically this has been referred to as a *Digit*. It is the representation of RawData which is put into the Transient Event Store and is potentially persistifiable.
- DetectorElement(DE):
 - Pixel Detector: A module, equivalent to a single Silicon wafer.
 - SCT: One side of a module, equivalent to a bonded pair of wafers whose strips are oriented in a single direction (*i.e.*, axial or stereo).
 - TRT: a planar set of straw tubes representing one row at a given radius of straws in a barrel module (*i.e.*, a plane corresponding to the tangential direction in the barrel) and 1/32 in $r\phi$ at a given z of straws in an end-cap wheel.
- Cluster[†]:
 - *Pixel Detector*: A two-dimensional group of neighbouring readout channels in a DE.
 - SCT: A one-dimensional group of neighbouring readout channels in a DE.
 - TRT: A calibrated measurement in a single straw.
- Global Coordinate System: A right-handed Cartesian coordinate system where the origin is centered around the nominal interaction point (*i.e.*, the mean beam collision point) with the z-component parallel to the axial component of the magnetic field produced by the solenoid.
- Local Coordinate System: A DE-dependent coordinate system related to the Global Coordinate System by an Euler transformation.
- SpacePoint(SP)[†]:

 $^{^{4}}$ The format of the ROB-data is subdetector dependent and is not yet fixed. Definitive proposals of the format and data content for each subdetector exist and are use-case dependent. For a summary of on-going discussions, see [13].

⁵These RDOs may or may not be produced. In the HLT environment, a conversion from RawData directly to Clusters occurs.

- Pixels: A single Cluster in the Global Coordinate System.
- SCT: A combination of one axial Cluster and one stereo Cluster.
- ClusterOnTrack(COT)[†]: A modified Cluster with its parameters and errors corrected given knowledge of a track intersect and direction accounting for DE alignment and calibration.
- Track[†]: In general, an object containing a parameterization of a hypothesized particle trajectory through space relating groups of Clusters together. As discussed and defined elsewhere [14], a Track trajectory consists of three position, two direction, and one curvature⁶ parameters. If specified at an intersecting surface, there are five parameters and a covariance matrix. At various stages of the reconstruction process, different versions of Track exist:
 - TrackSegment: a local combination of Clusters which may span subdetectors, also referred to as a Track *seed*.
 - TrackCandidate: a Track contained internally within reconstruction algorithms and subject to further processing.
 - OutputTrack The object for use by physics analysis packages containing consistent sets of parameters with errors evaluated at specified point(s) and knowledge of the fitted combination of COT(s) from which they are derived.
- **DetectorGeometry**: A static view (*i.e.*, not reinitialized or recalculated for a set of events in which conditions and alignment are assumed to be constant) of the ATLAS detector containing the positions and geometric parameters of each DE in the ID, and an associated hierarchical organization of DEs.
 - Layer A level in the hierarchical structure consisting of an aggregation of DEs at similar radial positions (barrel) or z (endcap).
 - PassiveMaterial: A reconstruction-oriented view of the material within the ATLAS tracking volume of appropriate granularity in which all detector and service material is parameterised into a set of surfaces with a radiation length, an interaction length, and a minimum-ionising particle energy-loss coefficient.

3 Requirements

The subsections below discuss requirements for the EDOs of the ID EDM.

⁶The use of curvature assumes a homogeneous magnetic field in which case this quantity is constant. For ATLAS and its inhomogeneous magnetic field, this parameter may be replaced by an invariant quantity such as charge/p

3.1 Cluster

- UR1.1 Cluster from the Pixels and SCT shall provide a position measurement in Local and Global Coordinate Systems.
- UR1.2 Cluster shall provide a covariance matrix at its position in its Local Coordinate System.
- UR1.3 Cluster from the TRT shall provide the center of the drift circle inside a straw in Local and Global Coordinate Systems, the drift radius, and the uncertainty on the drift radius.
- UR1.4 Cluster for the TRT shall provide time-over-threshold and transition radiation information.
- UR1.5 Cluster for the Pixels (and SCT) shall provide its overall multiplicity and its two (one)-dimensional width.

Requirements related to Cluster Relationships (UR6.1-UR6.7) and Organization (UR7.1-UR7.4) are described in Section 5.

3.2 SpacePoint (SP)

- UR2.1 SP shall provide its position in Cartesian and Cylindrical coordinates.
- UR2.2 SP position shall be provided in the Global Coordinate System.
- UR2.3 SP shall provide a covariance matrix in the Global Coordinate System on demand.⁷

Requirements related to SP Relationships (UR6.2, UR6.5, UR6.6) and Organization (UR7.1-UR7.4) are described in Section 5.

3.3 ClusterOnTrack (COT)

- UR3.1 A service shall be provided that corrects a Cluster with respect to an external Track prediction which produces ClusterOnTrack.
- UR3.2 COT from the Pixels and SCT shall provide their corrected position in Local and Global Coordinate Systems.

⁷The Use Case for this requirement is pattern recognition in xKalman++ where cylindrical coordindates are used and only the diagonal terms (σ_{rr} , $\sigma_{\phi\phi}$, σ_{zz}) are required

- UR3.3 COT from the TRT shall provide the corrected center of the drift circle in Local and Global Coordinate Systems, the corrected drift radius and its uncertainty, choosing the appropriate side of the drift circle for a given Track.
- UR3.4 COT shall provide a covariance matrix.
- UR3.5 COT should provide additional information to support ID alignment; these are
 - Position with current alignment (see UR3.2);
 - Knowledge of its DE (see UR6.5);
 - Track fit residuals and uncertainty (one or two planes);
 - Track direction in Local Coordinate System;
 - Scattering angles at plane and expected RMS (derived from UR5.7);
 - Derivatives of residuals (calculable from above);
 - Drift time (for TRT) (see UR3.3).

Requirements related to COT Relationships (UR6.3, UR6.4, and UR6.6) are described in Section 5.

3.4 Track

The requirements below apply equally for OutputTrack, TrackCandidate, and TrackSegment (referred to in general as *Track*).

- UR4.1 A Track shall know its parameters and covariance matrix in the Global Coordinate System in the vicinity of the interaction region and at its end point.
- UR4.2 It must be possible to extrapolate a Track to any location or detector surface along its trajectory.
- UR4.3 A Track shall know its fit quality (*e.g.*, χ^2 , number of missing Clusters, number of shared/ambiguous Clusters, presence of kinks).
- UR4.4 It must possible to refit⁸ the Track.

Requirements related to Track Relationships (UR6.4 and UR6.6) are described in Section 5.

⁸Here the term *refit* covers a variety of Use Cases. In the context of Heavy Flavour triggers or analyses, a Track may be refit to a secondary or tertiary vertex constraint. Also a Track could be refit in the context of changes to alignment, calibration, magnetic field mapping, or clusterization algorithms. Furthermore, given a Track from one ORP, the need could exist to refit it using the algorithm of a different ORP or more refined stage of the same ORP.

4 DetectorElement (DE) and DetectorGeometry

As mentioned in Section 1, DEs and DetectorGeometry are not part of the EDM, but instead belong to DD, and, in a broader context, the general Data Model. However, as DEs contain Event Data (*i.e.*, RDOs and Clusters belong to a given DE), the EDM and the DD are closely coupled. In this section, we discuss requirements for DD components needed by the ID EDM.

- UR5.1 DE shall know its position in the Global Coordinate System as well as its shape and geometrical description.
- UR5.2 DE shall know the transformation into its Local Coordinate System.
- UR5.3 DE shall know its position in a hierarchical structure.
- UR5.4 For a given DE, the neighbouring DEs shall be obtainable in an efficient way (*i.e.*, without iterating over all DEs in a subdetector or Layer). In the case of the SCT, DEs shall also know their axial or stereo DE counterpart in a similar manner.
- UR5.5 DE shall provide its orientation (*i.e.*, barrel versus endcap, tilt, and relevant stereo angle information).
- UR5.6 A service shall provide a list of DE(s) and/or PassiveMaterial intersected by a Track or a given volume in space.
- UR5.7 PassiveMaterial shall be grouped into surface elements, each of which is characterized by the energy loss for a minimum-ionising particle and radiation and interaction lengths for particles intersecting the surface at normal angle.

Requirements related to DE Relationships (UR6.7) and Organization (UR7.1-UR7.3) are described in Section 5.

5 Object Organization and Relationships

5.1 Relationships

- UR6.1 It shall be possible to iterate over all RDOs used to produce a Cluster.
- UR6.2 SP should know the Clusters that produced it.
- UR6.3 Effective bidirectional association⁹ between a Cluster and one or more COTs shall be possible.

⁹Here we use the term *effective bidirectional association* not to imply explicit bidirectional pointers or DataLinks, but the ability for two objects to know about each other via some feasible mechanism such as two uni-directional pointers or DataLinks or even a simple look-up table

- UR6.4 Effective bidirectional association between COTs and their associated Track shall be possible.
- UR6.5 Cluster, COT, and SP shall know the DE(s) to which they belong. [(s) for SCT]
- UR6.6 Appropriate Monte Carlo simulation truth information shall be associated to RDO, Cluster, SP, COT, and Tracks, but the possibility of such association will be active only for simulation.
- UR6.7 DE shall permit navigation to its event data (*i.e.*, RDO and Cluster).

5.2 Organization

- UR7.1 RDOs and Clusters shall be ordered by DEs.
- UR7.2 The collection or organizing structure for RDOs, Clusters, and SPs shall know the corresponding DE(s).
- UR7.3 It shall be possible to randomly iterate over all Clusters (SPs) corresponding to a given DE(s) [(s) for SP in SCT].
- UR7.4 Iteration over Clusters and SPs within a collection shall follow a defined ordering, if possible that of "byte stream" RawData.

6 Use Cases

- UC.1 Alignment: The Inner Detector alignment procedure is an iterative process with the following use cases.
 - a) Tracks need to provide their quality information (χ^2 , number of ambiguous hits, *etc.*);
 - b) Residuals between a Cluster or ClusterOnTrack with respect to a Track are recomputed;
 - c) Cluster global position and ClusterOnTrack are recomputed using alignment conditions from the previous iteration, and Tracks are refitted.

UC.2 Calibration and Evaluation of related Conditions Data:

- a) Given a set of new TRT time-distance relations, TRT Clusters as well as ClusterOnTrack and Track, which are derived from them, are all recomputed;
- b) Given a new dead or noisy channel map, SCT and Pixel Clusters are recomputed leading to a new processing (this also applies for changes in clusterization algorithms).

- UC.3 **Cluster Building** RDOs or RawData from neighbouring readout channels within a DE are grouped together with a sub-detector dependent algorithm accounting for calibration information (*i.e.*, dead or noisy channels) to form a Cluster which has knowledge of its position and is able to know its DE.
- UC.4 **SpacePoint Formation** Groups of Clusters within the SCT (or Pixels) are grouped together to form three dimensional space points with a sub-detector dependent algorithm.
- UC.5 **Track Finding** A Track seed is formed from a set of SpacePoints in SCT or Pixels forming a TrackSegment.
- UC.6 **Track Following** A TrackSegment is extrapolated through the Inner Detector tracking volume. A service provides an ordered list of DEs which intersect the trajectory. The Clusters on each DE which are closest to the Track's intersect are determined in an efficient way.
- UC.7 **Track Fitting** An algorithm is used to optimize a TrackCandidate's trajectory given knowledge of the residuals between itself and its Cluster as well as energy loss and multiple scattering occuring at any PassiveMaterial along its trajectory.
- UC.8 **ClusterOnTrack creation** ClusterOnTracks are created during reconstruction which contain the corrected position of a Cluster given a Track extrapolation and detailed information about alignment and calibration.
- UC.9 **Combined Reconstruction** A Track is associated with a Calorimeter object for the purposes of electron/photon identification. A Track from the Inner Detector is associated with a Track from the Muon System.
- UC.10 **Physics Analysis** A set of OutputTracks is used to derive quantities useful for physics analysis.

For Heavy Flavour analyses or Discovery physics analyses relying upon identification of b jets, displaced vertices are formed from sets of Tracks. Tracks are refit using the constraint that they originate from these displaced vertices. Track impact parameters are calculated with respect to the Primary Vertex and displaced vertices.

Acknowledgements

The authors would like to acknowledge useful discussions within the CERN Atlas Team Software and Physics Tools Working Group during January and February 2002. The editor would like to thank Brookhaven National Laboratory for its hospitality during May 2002 while the final draft of this document was prepared, and would also thank D. Adams, R. Clifft, W. Krasny, H. Ma, and S. Rajagopalan for useful discussions.

References

- [1] I. Gavrilenko, *xKalman++*, see http://maupiti.lbl.gov/atlas/xkal/xkalmanpp/index. en.html.
- [2] R. Clifft and A. Poppleton, *IPATREC: inner detector pattern-recognition and trackfitting*, see http://atlasinfo.cern.ch/Atlas/GROUPS/SOFTWARE/DOCUMENTS /IPATREC/ipatrec.html.
- [3] The PESA Software Group (ed. Markus Elsing), Analysis and Conceptual Design of the HLT Selection Software, May 2002.
- [4] M. Cattaneo *et al.*, GAUDI Data Processing Applications Framework, see http://proj-gaudi.web.cern.ch/proj-gaudi/GDG/v1/GDG.pdf.
- [5] P. Calafiura, H. Ma, S. Rajagopalan, and D. Rousseau, The ATLAS Data Model User's Guide, February 2002.
- [6] S. Armstrong, *Event Data Model for Inner Detector Reconstruction*, presentations given 5 March 2002 (ATLAS Software Week) and 18 April 2002 (ID EDM meeting), available via http://www.cern.ch/Steve.Armstrong/talks.
- [7] R. Hawkings, Alignment and Calibration Requirements for the Event Data Model and Detector Description, presentation given 18 April 2002 (ID EDM meeting), available via http://r.home.cern.ch/r/rhawking/www/atlas/idedm_apr02.pdf.
- [8] M. Elsing, General Event Data Model Issues and the Event Data Model for the Trigger, presentation given 27 February 2002 (PESA meeting), available via http://documents.cern.ch/archive/electronic/other/agenda/a02260/a02260s1t6 /transparencies/Markus.pdf.
- D. Calvet, ATLAS Silicon Trackers Software, available via http://atlas-proj-id-software.web.cern.ch/atlas-proj-id-software/ATLAS/index.html;
 D. Candlin and R. Candlin, The IDetSpacePoint Package, documentation available via http://atlas.web.cern.ch/Atlas/GROUPS/SOFTWARE/OO/dist/current /installed/share/doc/InDetSpacePoint/
- [10] N. Konstantinidis, Developments of Event Data Model for Inner Detector, presentation given 26 February 2002 (Inner Detector Software meeting), available via http://documents.cern.ch/AGE/current/fullAgenda.php?ida=a02353.
- [11] The Moore Group, *Moore Muon OO REconstruction for ATLAS*, see http://www.usatlas.bnl.gov/computing/software/moore/.

- [12] A. Zalite, М. Bosman, S. Armstrong, R. Hawkings, Ν. Konstantidnis, and D. Adams, presentations given inthe first Inner Detec-Muon Model Meeting, tor and Data 18April 2002,available via http://documents.cern.ch/AGE/current/fullAgenda.php?ida=a02605.
- [13] W. Krasny, Raw Event Structure, presentation in Event Data Model and Detector Description Meeting 5 March 2002, available via http://documents.cern.ch/cgibin/setlink?base=agenda&categ=a01922&id=a01922s6t20/transparencies.
- [14] D. Adams, Muon Requirements for the Common Track Class, presentation given in the first Inner Detector and Muon Data Model Meeting 18 April 2002, available via htp://documents.cern.ch/AGE/current/fullAgenda.php?ida=a02605.