Physics Analysis Tools for Beauty Physics in ATLAS

Christos Anastopoulos\textsuperscript{1}, Eva Bouhova-Thacker\textsuperscript{2}, James Catmore\textsuperscript{2}, Steve Dallison\textsuperscript{3}, Frederick Derue\textsuperscript{4}, Brigitte Epp\textsuperscript{5}, Patrick Jussel\textsuperscript{5}, Anna Kaczmarska\textsuperscript{6}, Lee de Mora\textsuperscript{2}, Holger von Radziewski\textsuperscript{7}, Pavel Řezníček\textsuperscript{8}, Thorsten Stahl\textsuperscript{7}

(on behalf of the ATLAS collaboration)

\textsuperscript{1}Physics Department, Aristotle University Of Thessaloniki
\textsuperscript{2}Department of Physics, Lancaster University
\textsuperscript{3}Particle Physics Department, CCLRC Rutherford Appleton Laboratory
\textsuperscript{4}LPNHE, IN2P3 - CNRS - Universités Paris VI et Paris VII
\textsuperscript{5}Institute for Astro- and Particle Physics, University of Innsbruck
\textsuperscript{6}Institute of Nuclear Physics, Polish Academy of Sciences
\textsuperscript{7}Department of Physics, University of Siegen
\textsuperscript{8}IPNP, Faculty of Mathematics and Physics, Charles University in Prague

E-mail: pavel.reznicek@cern.ch

Abstract. The Large Hadron Collider experiments will search for physics phenomena beyond the Standard Model. Highly sensitive tests of beauty hadrons will represent an alternative approach to this research. The analysis of complex decay chains of the beauty hadrons have to efficiently extract the detector tracks made by these reactions and reject other events in order to make sufficiently precise measurement. This places severe demands on the software used to analyze the B-physics data. The ATLAS B-physics group has written a series of tools and algorithms for performing these tasks, to be run within the ATLAS offline software framework Athena. This paper describes this analysis suite, paying particular attention to mechanisms for handling combinatorics, interfaces to secondary vertex fitting packages, B-flavor tagging tools and finally Monte Carlo true information association to pursue simulation data in process of the software validations which is an important part of the development of the physics analysis tools.

1. Introduction

ATLAS is a general purpose detector \cite{1} designed to exploit the full discovery potential of the Large Hadron Collider (LHC) - CERN’s 7 + 7 TeV proton-proton collider with a design luminosity of $10^{34}$ cm$^{-2}$s$^{-1}$. The experiment is expected to begin data taking in summer 2008 \cite{2}. Its wide program consists of both discovery physics and precision Standard Model measurements.

Physics of beauty quark systems is a part of the ATLAS physics program \cite{3}. It involves measurements of CP-violation, $B$-oscillations, $B$-hadron properties and rare decays with the aim to observe New Physics effects. The main part of the B-physics program is concentrated for the initial low-luminosity ($L = 2 \cdot 10^{33}$ cm$^{-2}$s$^{-1}$) stage of LHC running. This implies that B-physics will serve as one of the early test beds for understanding the detector properties. Therefore the studies of the ATLAS B-group include detector commissioning with early data, trigger, tracking and muon system calibrations and precision measurements of well known properties of $B$-particles like mass and lifetime. The expected rate of $b\bar{b}$ quark pairs produced

\textsuperscript{©} 2008 IOP Publishing Ltd
within the ATLAS detector acceptance during the low luminosity stage of the LHC is $\sim 20$ kHz. Since for all the ATLAS physics only about 100 Hz are committed to disk out of which $\sim 10$ Hz are devoted for B-physics, a highly selective trigger must be used. The B-physics triggers are generally based on detection of a single-muon, di-muon or a muon and a calorimetry cluster [4, 5].

In order to conduct the B-physics measurements and achieve sensitivity to New Physics effects, the analysis has to efficiently separate the tracks of the complex $B$-hadron decay chains from other reactions. An analysis suite consisting of a set of common tools, specialized data structures and a list of algorithms for each exclusive decay channel has been written by the ATLAS B-physics group to fulfill this requirement.

### 2. ATLAS Software and Data Flow Overview

The B-physics analysis suite is implemented within the ATLAS software framework Athena [6], which primarily uses object-oriented C++. The Athena applications are built up from a sequence of dynamically loadable components driven by Python configuration scripts. The framework provides common data-processing support, with major components including list of tools (e.g. vertexing tools), services (e.g. logging facility, histogram and ntuple services), data representation converters, common interfaces for per-event processing algorithms etc., all driven by an application manager. The B-physics analyses are coded in pure C++. The main processing classes are derived from an Athena algorithm extended by robust handling of ROOT-ntuple output [7].

The physics analysis runs on reconstructed data devoted for physics analysis - Analysis Object Data (AOD), the final product of the ATLAS reconstruction chain [6]. The chain starts with RAW data (the output of the final stage of the ATLAS High Level Trigger) that are reconstructed producing Event Summary Data (ESD), intended to make access to RAW data unnecessary for most physics applications. The AOD are then derived from ESD, representing reduced event information suitable for analysis, such as reconstructed particles and b-tagging information. Along side with AOD also Tag Data (TAG) are extracted collecting event-level metadata to support an efficient identification and selection of the events of interest to a given analysis. All the data from the ATLAS detector are distributed using a hierarchical four-tier structure of facilities ([6] p. 8) connected by a Grid infrastructure. Both AOD and ESD are held on Tier-1 facilities, while AOD are copied to Tier-2 sites that are accessible for physicists to perform the analyses.

The Tier-2 sites also provide all the required simulation capacity for the experiment. The Monte Carlo (MC) simulation chain consists of an event generation, Geant4 [8] simulations of the particles’ passage through the ATLAS detector and finally detector response simulation. The output of the last stage (digits) is then used to produce ESD and the rest of the chain is similar as with real data.

To avoid large amounts of data being copied over to local computers, the analysis software is sent to the data on the Tier-2 sites. The jobs are distributed via a Grid middleware to which several interfaces exist, the B-physics group is presently using Ganga [9]. A large Computing System Commissioning (CSC) MC production of B-physics channels was performed via this interface in 2007: 450,000 of signal and 700,000 of background events were generated, simulated and reconstructed and are analyzed by individual physicists on selected data samples appropriate to the studied channels. The signal datasets consist of events containing a particular decay channel, while the backgrounds are of inclusive type like e.g. any event where $b\bar{b}$ pair was produced and there were two muons in the final state with $p_T > 4$ GeV [10].

### 3. B-Physics Analysis

At the LHC the B-physics program focuses on the study of exclusive decay channels, therefore the analysis is characterized by many different topologies and constraints. In order to avoid
duplicated efforts, common tools, data structures and calculations were collected and unified over all the B-physics channels. The main task in the B-physics analysis is to identify a B-decay chain typically consisting of a cascade of several vertices, like the topologies depicted in the following sketch:

\[ B_{s}^{0} \rightarrow J/\psi \rightarrow \mu^{+} \mu^{-} \]

\[ B_{s}^{0} \rightarrow \phi \rightarrow K^{+} K^{-} \]

\[ B_{s}^{0} \rightarrow \mu^{+} \]

\[ B_{s}^{0} \rightarrow \pi^{+} \]

\[ D_{s}^{-} \rightarrow \phi \rightarrow K^{+} K^{-} \]

\[ K^{+} \rightarrow \mu^{-} K^{-} \]

\[ K^{+} \rightarrow K^{-} \phi \]

\[ \phi \rightarrow \pi^{+} \]

This requires the usage of a vertex finder devoted for offline data processing, a management of the combinatorics of tracks to form the candidates of the elements of the decay chain and an extraction of various properties of the fitted particle-like objects. In order to check the efficiencies, performance and sources of background, one needs the MC true information (MC truth) and its associations to the reconstructed objects. There were tools developed to ease the usage of these relations like an extraction of the tracks corresponding to the signal particles of the studied decay channel.

Since the B-physics analysis suite is a part of the Athena software framework, the Athena services are used to access the objects stored in the AOD files. Presently, the B-physics analyses rely on reconstructed Inner Detector [11] tracks, combined muon and electron objects, trigger decision information, reconstructed primary vertices and particle jets.

The analysis extracts the B-decay candidates and produces ROOT ntuples, that can be transferred to the users’ private locations and the final tuning is performed using a collection of ROOT scripts.

3.1. Typical Analysis Procedure

The analyses of the various decays have a common structure, that is here illustrated by means of the reconstruction of the \( B_{s}^{0} \rightarrow J/\psi (\mu^{+} \mu^{-}) \phi (K^{+} K^{-}) \) channel. The analysis steps are:

(i) Get the collections of the analysis objects in a given event.

(ii) Select tracks matching the kinematic cuts for the muon candidates and consult the muon combined objects to retain tracks belonging to the identified muons only. Combine such muon candidates into opposite charged pairs (\( J/\psi \) pre-candidates) and possibly cut out the pairs not satisfying some additional conditions like e.g. the muon tracks invariant mass window (before fitting a common vertex).

(iii) Perform vertexing of all the preselected muon pairs and apply appropriate quality, position, mass window and other cuts on the \( J/\psi \) candidates.

(iv) Select the \( \phi \) candidates using similar procedure (combine and use for vertexing the tracks being identified neither as muons nor as electrons).

(v) Combine the \( J/\psi \) and \( \phi \) candidates to create a track quadruplet, fit a common vertex of the whole \( B_{s}^{0} \) decay and again reject low-quality, out of the mass window and other unlikely \( B_{s}^{0} \) candidates.

(vi) Calculate additional variables (e.g. \( B_{s}^{0} \) isolation, flavor tagging etc.) to be either used for further rejection cut or to be just stored alongside with the \( B_{s}^{0} \) candidates in the output ntuple for a consequent ROOT analysis.

4. B-Physics Analysis Tools

The analysis suite consists of a set of predefined decay identification algorithms, their component tools and data structures simplifying the management of the selected candidates. The software is organized into three separate packages [12]:

International Conference on Computing in High Energy and Nuclear Physics (CHEP’07) IOP Publishing Journal of Physics: Conference Series 119 (2008) 032003 doi:10.1088/1742-6596/119/3/032003
(i) **BPhysAnalysisObjects** - contains non-trivial data structures used by the analysis programs and the tools: vertex fitting and flavor tagging results and a class collecting properties of a composite (decayed) particle.

(ii) **BPhysAnalysisTools** - collection of tools for building the decay channel analysis algorithm. This includes track combinatorics and selection tools, secondary vertex fitters, truth finding and association, B-flavor tagging and helper routines to calculate proper time, transverse decay length and other variables.

(iii) **BPhysExamples** - holds the main analysis algorithms and a code skeleton for creating a new user analysis. The structure of the code has one algorithm per decay channel, with the present implementation of:

- \( B^+ \rightarrow J/\psi(\mu^+\mu^-)K^+ \)
- \( B^0_d \rightarrow J/\psi(\mu^+\mu^-)\phi(K^+K^-) \)
- \( B^0_s \rightarrow D_s^- (\phi(K^+K^-)\pi^-)\pi^+ \)
- \( B^0 \rightarrow D_s^- (\phi(K^+K^-)\pi^-) a_1^+(\rho(\pi^+\pi^-)\pi^+) \)
- \( \Lambda^0_b \rightarrow \Lambda^0 (p,\pi^-)\mu^+\mu^- \)

The main part of the analysis is performed by the **BPhysExamples** algorithms, which call the tools from the **BPhysAnalysisTools** package as needed. The output of this part of analysis is stored into a flat ntuple, whose contents and structure is decided by the author(s) of the particular algorithm. The output ntuples are further processed by ROOT scripts, presently held in physicists’ private areas.

4.1. The Data Structures

There are three classes extensively used in the analysis algorithms and the tools:

(i) **Vertex** - a data structure that collects results from the vertex fit, using data members natural for analysis. In comparison to similar objects at the reconstruction stage, this class provides particle-like objects - e.g. tracks are having mass hypotheses associated, the vertex holds its invariant mass, etc.:

- vertex position, fit covariance matrix, fit quality, fit error flag, vertexing algorithm used,
- total vertex (composite particle) momentum, charge and mass,
- refitted tracks (tracks in vertex) parameters and their \( \chi^2 \),
- separation from the mother vertex,
- refitted primary vertex and
- variables specific to the selected vertexer (see section 4.2(ii)).

One important function of this class is to unify (if possible) the output of the different vertexers currently available, so that their differences in the output format are not visible from the main analysis algorithms. More complex members of this class are represented by objects from CLHEP [13] - the official ATLAS mathematical tools library of choice.

(ii) **VertexAndTracks** - designed to represent a composite particle candidate, keeping the relevant reconstructed, truth and genealogy information. It holds also results which are used to decide about whether the candidate should be rejected (is outside the requested cuts). The present contents are:

- the fitted vertex,
- original tracks of the stable daughters,
- composite mother and daughters candidates,
- MC truth particles associated with the stable daughter tracks and
- variables for the cut-based selection of the candidates: e.g. invariant mass, proper time, impact parameter, total \( p_T \).
The original daughter tracks, the composite mother and daughter candidates and the MC generator particles are only referenced using pointers.

(iii) **BFlavourParticle** - a container with the lepton flavor tagging results (see section 4.2(iii)). It is a data structure that holds some properties of the tagging muon candidate and eventually the nearest jet.

### 4.2. Tools for building the decay channel analysis algorithm

The helper tools for the main analysis algorithm are divided into the following set of classes:

(i) **Toolbox**

The **BPhysToolBox** class is collecting all the common tools and calculation methods used among most or all the decay channels analyses. The methods of the toolbox can be divided into five groups:

- tracks selection methods, based on the kinematic parameters (basically $p_T$, $\eta$, charge),
- tracks combinatorics, creating pre-candidates of composite particles, e.g. making (opposite charge) pairs, triplets, etc.,
- particle identification functions returning the collection of tracks associated with the requested particle type (presently includes the muon identification only based on either muon detectors information or MC truth associations),
- methods for the usage of the MC truth - e.g. the association of tracks to MC particles, extraction of the requested decay chain (and also the associated tracks) from the MC truth collection, etc., and
- various parameter calculations, e.g. the proper time, forward-backward asymmetry of a B-decay, impact parameter, etc. not present in the analysis objects.

The methods from the first 4 groups are usually used only once by the main analysis algorithm and does not interact with each other, while the parameters calculations are extensively called by both the other toolbox methods and the main analysis algorithm.

(ii) **Vertexing Interfaces**

A robust offline vertex fitter supporting mass and pointing constraints is one of the key tools for the B-physics analyses. There are several vertex fitting algorithms in the ATLAS offline software, out of which the B-physics group presently uses two:

- **VKalVrt** - using Kalman filter method for the vertexing [14].
- **CTVMFT** - a FORTRAN based fitter developed by the CDF collaboration, extended by an interface to Athena [15].

Both the fitters are capable of complex mass, pointing (e.g. to the primary vertex) and conversion vertex type constraints. **VKalVrt** works with the detailed map of the ATLAS inner detector magnetic field, in contrast to the **CTVMFT** fitter which uses a constant field approach. The B-physics interfaces to these fitters are wrappers to their offline interfaces, which are presently different (a unified interface is under development). The purpose of this additional interface is to allow physics-objects-like input, e.g. defining a decay topology to be fitted, and unifying the output of the two fitters: the results are stored in the **Vertex** class. This approach also allows an easy connection to any eventual new fitter. Though the input methods for the two vertex fitters are not yet completely unified, an **XtoYZFinder** tool (inheriting from Athena **AlgTool** [6], p. 29) is already available and provides this unification for often fitted two-body decays.

(iii) **Tagging Tools**

Especially for mixing and CP-violation studies, the flavor of the B-meson at the time of production needs to be known. There are two techniques implemented in the B-physics analysis software (again the tools inherit from the Athena **AlgTool**):
The **Lepton Tagger** is using the opposite $b$-quark's weak decay characterized by a high $p_T$ lepton whose charge indicates the $b$-quark flavor as illustrated by the following diagrams:

The tagging algorithm inside a **BFlavourTagger** class is searching for high-$p_T$ muons in the event and eventually the associated jet. It scans the input muon collection, and eventually searches for the jet nearest to the tagging muon in the input particle jet collection, and stores the results into a list of **BFlavourParticles** sorted by decreasing muon $p_T$. The data access is handled by *set*- and *get*-methods. This lepton tagger tool also allows to perform internal histogramming of the tagging results which may be used e.g. for the calculation of the wrong-tag fraction.

The **Jet Charge Tagger** makes use of the associated jet production alongside with the production of the $B$-hadron of interest, whereas the charge of the jet is correlated with the flavor of the $b$-quark as illustrated by the following diagrams:

This method is implemented in the **BFlavourJetChargeTagger** class. The core algorithm collects tracks inside a predefined cone around the $B$-hadron, thus building the associated jet candidate and accepting it based on its total weighted charge. The class consists of the *set*-methods for the input track collection, the $B$-hadron momentum, and the configuration of the jet-building/rejection algorithm, the core calculation methods and finally the *get*-methods providing the tagging result and information about the tagging jet.

5. The Analysis Output
The analysis algorithms use the Athena histogram and ntuple services to produce control histograms and an output ntuple collecting all the selected candidates of the analyzed decay chain in each event. Applying loose cuts during the analysis on the AOD allows a later tuning of the selection cuts on the ntuple level. Thus, to check results with only a slight change of the selection cuts, one does neither need to have the offline software installed nor does one need to rerun the analysis on the Grid. The AOD analysis is intended to extract the required $B$-hadron candidates per event. However, further statistical analysis on the datasets is performed on the output ntuples, naturally making use of existing statistical and mathematical tools available in ROOT.

5.1. **B-Physics Validation**
Since the mass, proper time resolution and other properties of the reconstructed decayed particles are very sensitive to the tracking and vertexing performance, the B-physics analyses
can serve as a reliable validation tool during the development of the offline software, addressing both the reconstruction and simulation issues. For continuously checking the offline releases, the Run Time Tester (RTT [16] and [6], p. 80) was developed by the ATLAS community, running a predefined set of tests on each daily development snapshot and reporting the results on automatically generated web pages. The B-physics part of the RTT repeats an analysis with fixed cuts on the $B^0_s \rightarrow J/\psi(\mu^+\mu^-)\phi(K^+K^-)$ decay on a predefined set of signal events using the B-physics analysis packages. The validation job itself monitors the performance in that channel and is driven by a set of Python scripts and ROOT scripts producing plots.

6. Summary
An analysis software suite for ATLAS B-physics has been developed, covering all the aspects of various decay topologies and constraints of B-physics. The code is integrated within the ATLAS software framework Athena, having a modular structure which contains a set of main algorithms (one for each decay channel studied) and common tools for building the decay chain. The tools are implemented in separate classes - for vertexing interfaces, flavour tagging and general methods including combinatorics, identification and Monte Carlo truth handling. The suite has been adopted by the ATLAS B-physics group in 2005, while it was still evolving and is presently heavily used during Computing System Commissioning ([6], p. 213) tasks. In the future, more decay channels will be included and appropriate tools added and ROOT scripts for the analysis of the output ntuples will be integrated into the analysis suite.

7. References
[1] ATLAS Collaboration, “ATLAS: Detector and physics performance technical design report. Volume 1, 2,” CERN-LHCC-99-14,15.
[2] http://lhc.web.cern.ch/lhc/, http://atlas.web.cern.ch/Atlas/index.html
[3] P. Ball et al., “B decays at the LHC,” arXiv:hep-ph/0003238, CERN-TH-2000-101.
[4] J. Baines [ATLAS Collab.], “ATLAS trigger for B physics,” Nucl. Instrum. Meth. A 462 (2001) 259.
[5] N. Panikashvili [ATLAS Collab.], “The ATLAS B-physics trigger,” Nucl. Phys. Proc. Suppl. 156 (2006) 129.
[6] G. Duckeck et al. [ATLAS Collab.], “ATLAS computing: Technical design report,” CERN-LHCC-2005-022.
[7] https://twiki.cern.ch/twiki/bin/view/Atlas/AtlasAwareNTuple
[8] S. Agostinelli et al. [Geant4 Collab.], “GEANT4: A simulation toolkit,” Nucl. Instr. Meth. A 506 (2003) 250.
[9] K. Harrison et al., “GANGA: A user-Grid interface for ATLAS and LHCB,” CHEP’03 proceedings
[arXiv:cs/0306085], http://ganga.web.cern.ch/ganga/
[10] https://twiki.cern.ch/twiki/bin/view/Atlas/TP出示ScAnalysisAndVertexing
[11] ATLAS Collaboration, “ATLAS inner detector: Technical design report. Vol. 1, 2,” CERN-LHCC-97-16,17.
[12] C. Anastopoulos et al., “Writing B-physics analysis code in Athena,” ATL-COM-SOFT-2006-014 (not publicly available).
http://twiki.cern.ch/twiki/bin/view/Atlas/PhysicsAnalysis/BPhys/Examples/doc
[13] L. Lonnblad, “CLHEP: A project for designing a C++ class library for high-energy physics,” Comput. Phys. Commun. 84 (1994) 307.
[14] V. Kostyukhin, “VKalVrt - package for vertex reconstruction in ATLAS,” ATL-PHYS-2003-031.
[15] J. Marriner, “Secondary vertex fit with mass and pointing constraint (CTVMFT),” CDF Internal Note 2724 (1994).
[16] http://www.hep.ucl.ac.uk/atlas/AtlasTesting