www.linkedin.com/in/mazurov
github.com/mazurov
alexander.mazurov@gmail.com

Alexander (Sasha) MAZUROV

Education

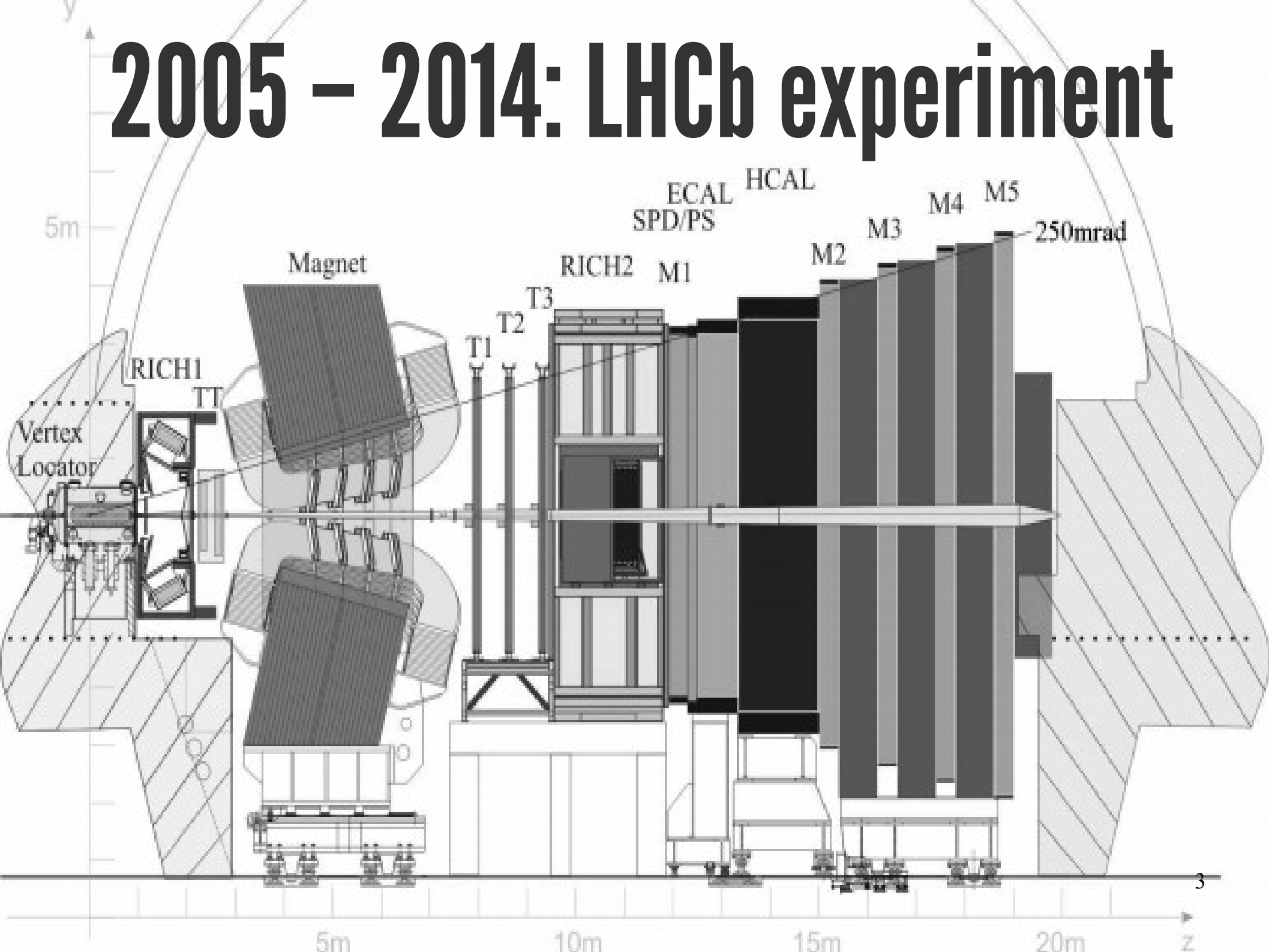
Ph.D in Physics

University of Ferrara (Italy), 2011-2014 (March)

M.Sc in Mathematics & Computer Science

Lomonosov's Moscow State University (Russia), 1997-2002

2005 – 2014: LHCb experiment



2005-2007: Summer student, INTAS-CERN fellowship

- New Gaudi Job Options Service's string parsers (based on Boost.Spirit v1.x library)
  - String parsers in GaudiKernel
  - Job options file parser

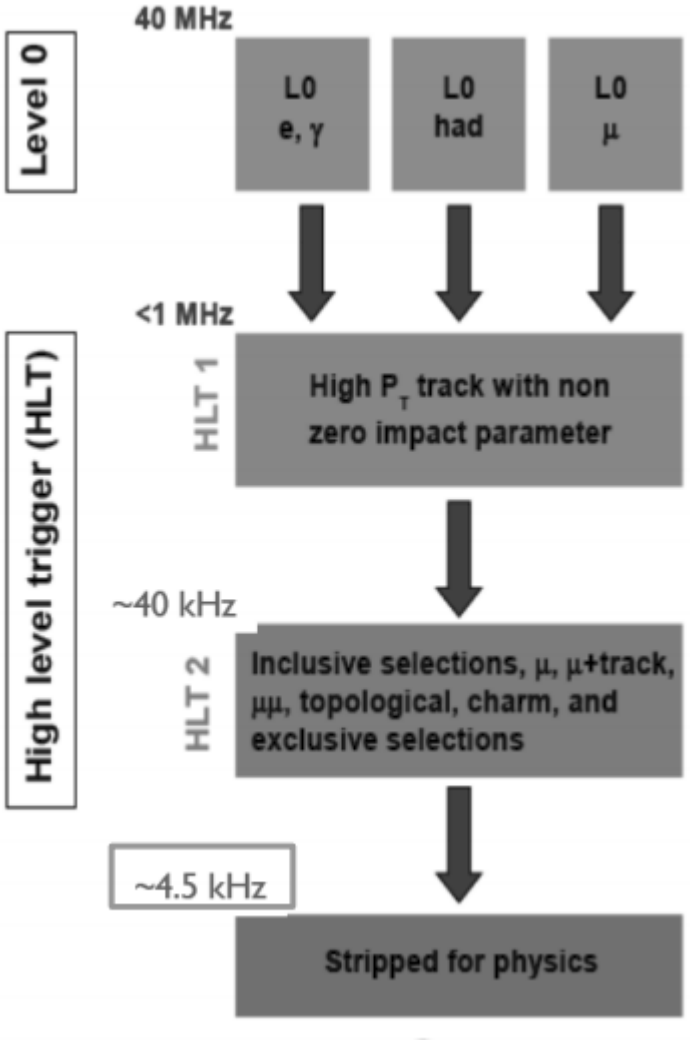
2007-2010: Project Associator at the Online group

- LHCb User management interface. User migration to Active Directory
- Setup and configuration of Helpdesk System, Twiki, web services.
- Run database web interface
- CastorFS (FUSE file system). Poster at CHEP 2009, Prague
  (source code: https://github.com/mazurov/castorfs)

2011-2014: Doctoral student

- Gaudi Parsers v2.0 (based on Boost.Spirit v2.x library)
- Gaudi Profiler Auditor. Talk at CHEP 2012, New-York
- Thesis: Profiler & study of χ_b production.

Trigger CPU profiling



Most experiments require a trigger in order to record interesting events at a suitable rate.

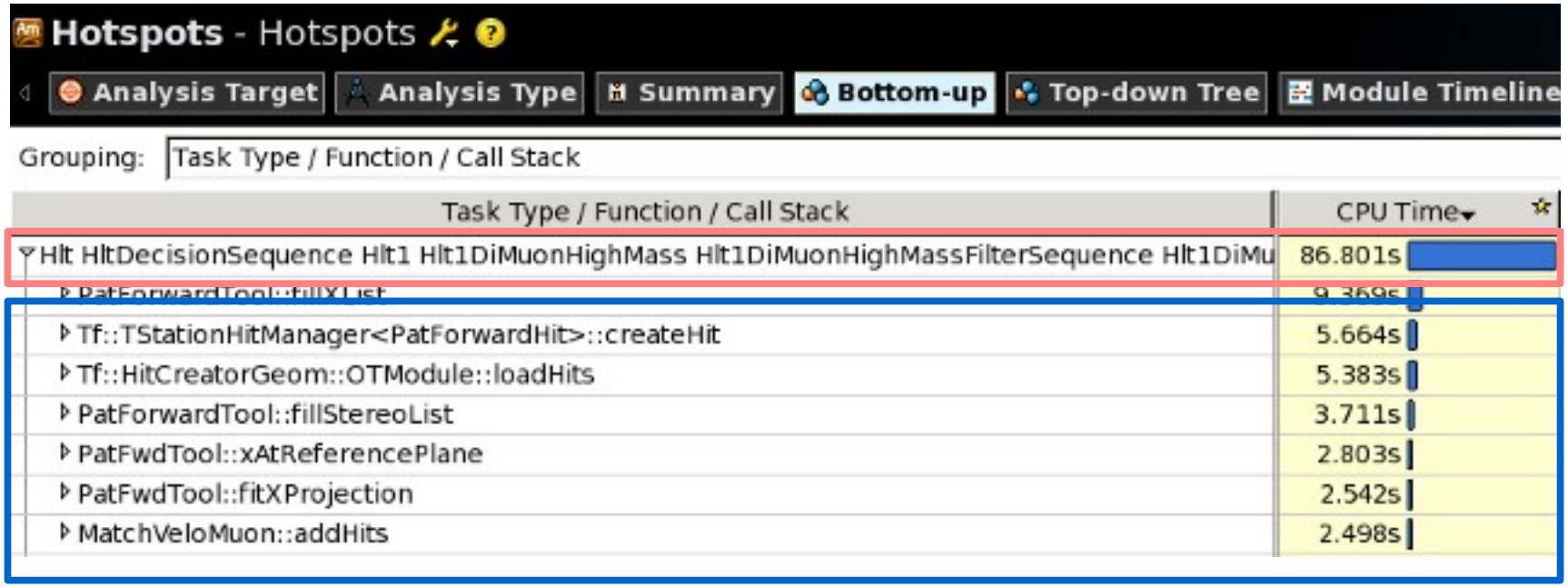
- L0 Hardware Trigger 40 MHz -> 1 MHz. Search for high pt, mu, e, gamma, hadron candidates.
- High Level Software Trigger Farm
  - HLT1: Add Impact parameter cuts
  - HLT2: Global event reconstruction

  - 100 man/years work that has only 20-30 ms to process an average event.
  - 29K CPUs or 1700 servers

The trigger needs fast algorithms!

CPU profiler tool is vital for trigger optimization

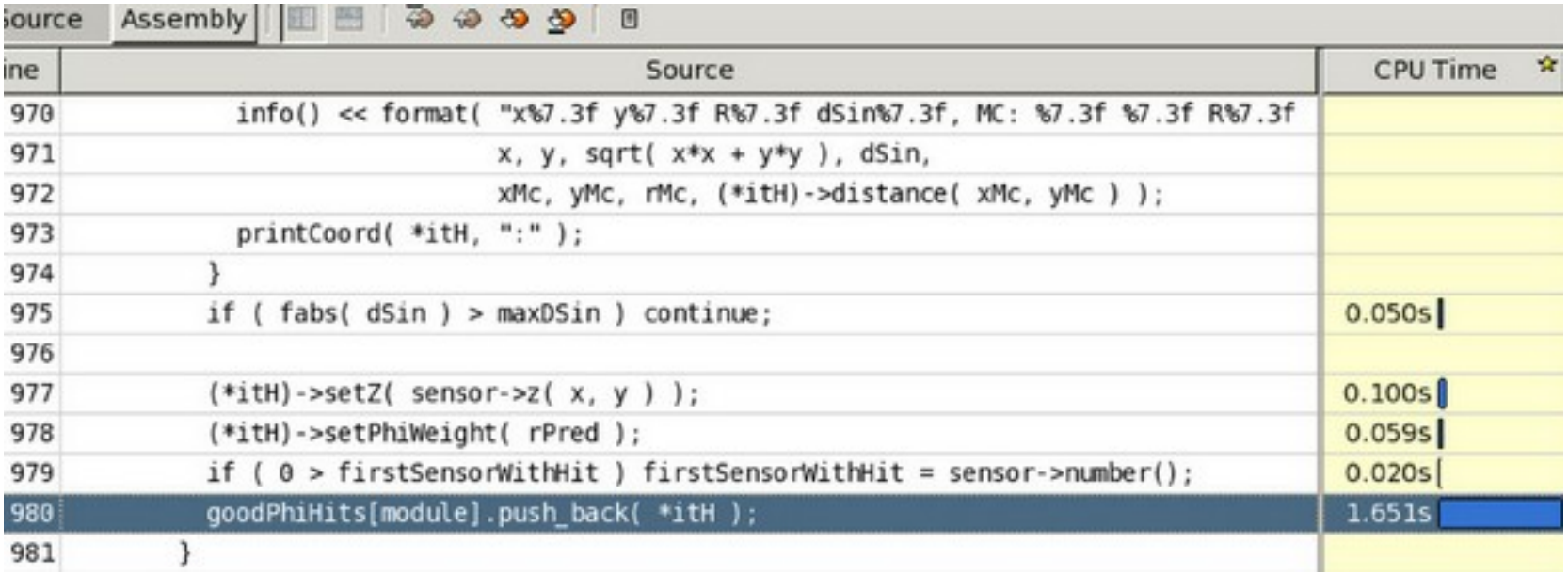
- Gaudi Intel Profiler Auditor (C++ library)
  - Deployed into the core software framework in LHCb – Gaudi.
  - Based on Intel VTune Amplifier XE User API.



Reports grouping:

- Static code properties: name of method or class
- Dynamic code properties: property values (value of algorithm's "Name" property)

CPU consumption by source code lines



Example of HLT hotspot

operatornew from libstdc++ library:

Grouping: Function / Call Stack

Function / Call Stack	CPU Time	Module
PatForwardTool::fillXList	23.460s	libPatAlgorithms.so
FastVeloTracking::makeSpaceTracks	19.788s	libFastVelo.so
operatornew	18.696s	libstdc++.so.6
Tf::HitCreatorGeom::OTModule::loadHits	10.870s	libTfTools.so
Tf::TStationHitManager<PatForwardHit>::createHit	7.981s	libTfTools.so

tc_new from tcmalloc library:

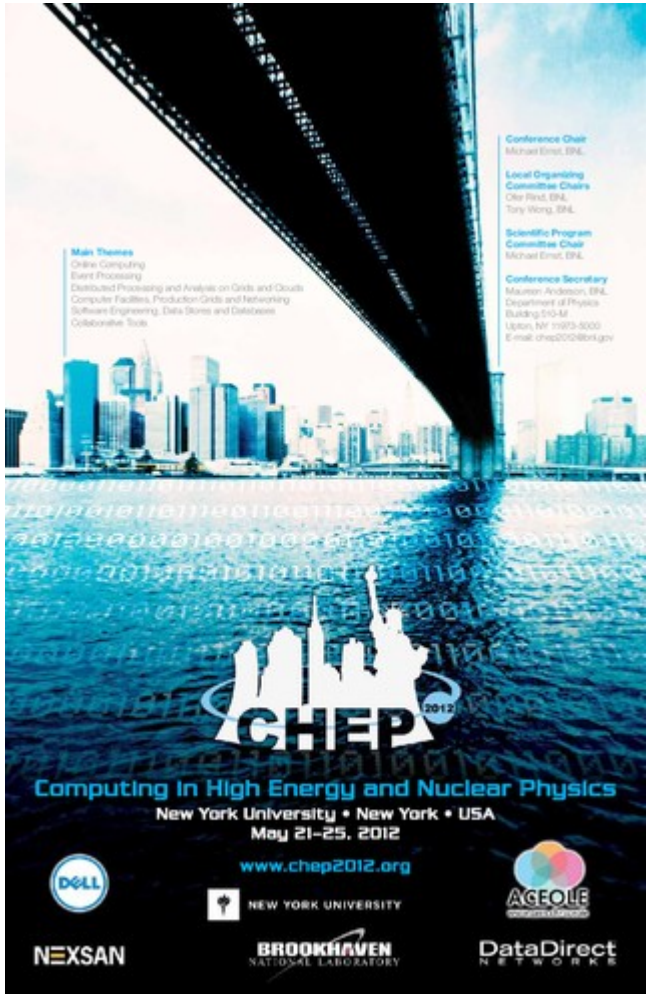
Grouping: Function / Call Stack

Function / Call Stack	CPU Time	Module
PatForwardTool::fillXList	22.483s	libPatAlgorithms.so
FastVeloTracking::makeSpaceTracks	16.963s	libFastVelo.so
Tf::HitCreatorGeom::OTModule::loadHits	8.465s	libTfTools.so
tc_new	8.157s	libtcmalloc.so.0.1.0
Tf::TStationHitManager<PatForwardHit>::createHit	7.374s	libTfTools.so

tc_new uses twice less time then operatornew

- Hotspot was detected
- Total CPU consumption decreased by 5%

CHEP2012: Talk and Paper

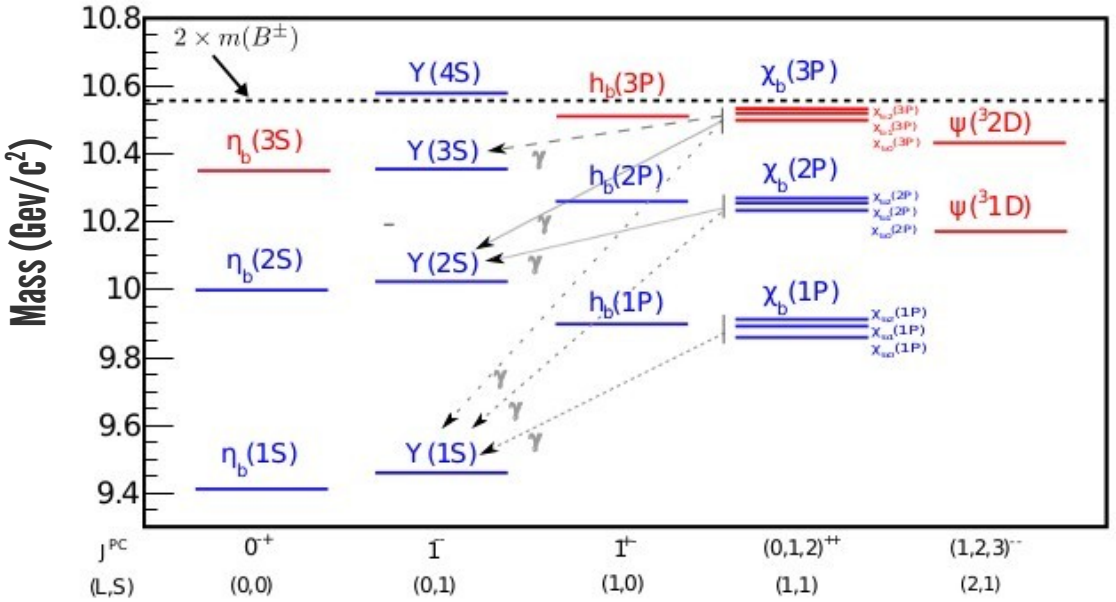


A. Mazurov and B. Couturier, “Advanced modular software performance monitoring”,
Journal of Physics: Conference Series 396 (2012), no. 5 052054.

Source code: https://github.com/mazurov/IntelProfiler

χ_b production study

$b\bar{b}$ system, which can be produced in different spin configurations, is ideal laboratory for QCD tests. It's like a hydrogen atom in QCD.



States with parallel quark spins (S=1):

- S-wave Υ state
- P-wave χ_b states, composed by 3 spin states $\chi_{b0,1,2}$. Can be readily produced in the radiactive decays of Υ

- χ_b(3P) state recently observed by ATLAS, D0 and LHCb.

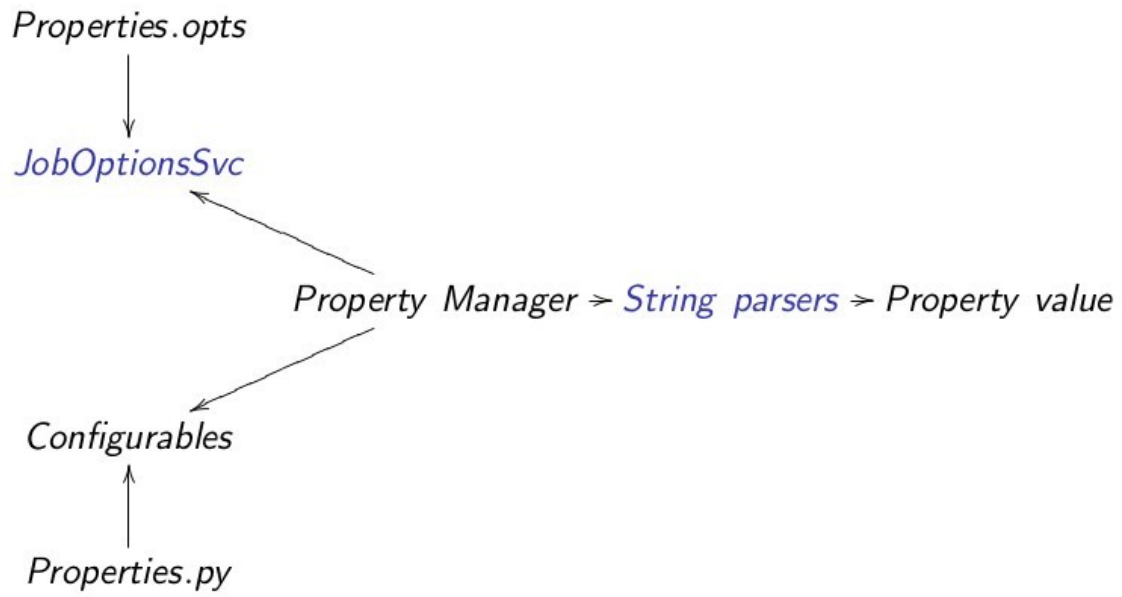
In the Thesis:

- Measurement for Υ(NS) (N=1, 2, 3) cross sections in χ_b decays as a function of pT(Υ)
- Measurement of χ_b(3P) mass.

http://amazurov.web.cern.ch/amazurov/mazurov-thesis-2.11.pdf

Gaudi String Parsers

LHCb Week Slides (2011): http://cern.ch/go/7VTl



- Flexible with respect to a addition of new types
- Readable and robust source code
- Based on Boost.Spirit v2, that is modern version with guaranteed long-term support
- Besides configuration parsers are used in DecayFinder and some other places.

StatusCode parse(TYPE& type, const std::string& input)

Standart C++ types

- int, double, bool, string
- vector, map, pair, set, list

Gaudi types

- StringKey
- Histo1Def
- XYZPoint, XYZVector
- LorentzVector

... and many compositions of this types.

Standalone version: https://github.com/mazurov/parsim

Thank you!