Table of Contents

AppConfig and job specifications

Last update : **23 November 2009**

AppConfig is the package where configurations for production are kept for the different LHCb applications. The options directory is structured with subdirectories where the options specific to a given application are kept (Gauss, Boole, Brunel, DaVinci). A subdirectory called Conditions contains files where the tags for DDDB and CondDB are given. Some global options are found in the top level directory

What is available in the various 'subdirectories' is described below.

Top level

Name	Comments
DisableLFC.py	Disable the usage of LFC to get the credentials to access Oracle servers
UseOracle.py	Force usage of Oracle database for Conditions
UseSQLDDB.py	Force usage of SQLDDDB for Conditions

Conditions

Predefined options to select combinations of DDDB and CondDB tags for a given well defined geometry are available in AppConfig and listed below. Combined with the Gauss, Boole or Brunel options they allow to configure those application for various beam, geometry and with truth/without truth in the case of simulated data.

Name	Geometry	Velo Position	Magnetic Field (crossing angle)	Comments
MC09-200900602-vc-md100.py	MC09 detector	closed	full field downward polarity	
MC09-200900602-vc-moff.py	MC09 detector	closed	field off	
MC09-200900602-vc-mu100.py	MC09 detector	closed	full field upward polarity	
Upgrade-20090917-md100-gauss.py	Upgrade detetector v1.1 without aerogel	closed	full field downward polarity	only to be used with Gauss
Upgrade-20090917-md100.py	Upgrade detetector v1.1	closed	full field downward polarity	

Gauss

AppConfig v3rX is used by Gauss v37r3 and higher to set the Simulation Conditions specifying the 'SIMCOND' settings and the beam settings.

The list of the avalaible steering options for is provided below. See presentation at *Tuesday Meeting 24 April 2009* for details on configuration.

Although they start with MC09 they are not specific to the MC09 production nor the detector described there. For the MC09 detector they need to be used with Conditions/MC09-....py for the Upgrade detector with

Conditions/Upgrade-....py and possibly options added below.

Additional options can be put in upon request for productions.

Name	Beam energy	Magnetic Field (crossing angle)	nu (Number of collisions/bunch)	Comments
MC09-b5TeV-md100.py	5 TeV	Full field downward direction	1	beta*=2m, bunch spacing more than 50 ns (no spillover), maximum possible external crossing angle
MC09-b5TeV-md100-nu2.py	5 TeV	Full field downward direction	2	as above
MC09-b5TeV-md100-nu3.py	5 TeV	Full field downward direction	3	as above
MC09-b5TeV-md100-nu4.py	5 TeV	Full field downward direction	4	as above
MC09-b5TeV-moff.py	5 TeV	Field off	1	no internal nor external crossing angle, otherwise as above
MC09-b450GeV-moff.py	450 GeV	Field off	1	no internal nor external crossing angle, beta*=10m, emittance and derived parameters adapted to energy
MC09-b7TeV-md100-nu0,7-25ns	7 TeV	Full field downward direction	0.7	beta*=31m, nominal effective crossing angle, spill-over as for 25ns bunch spacing, equivalent to L=2x10^32 cm^-2s^-1 for nominal machine
MC09-b7TeV-md100-nu1,7-25ns	7 TeV	Full field downward direction	1.7	beta*=12.5m, nominal effective crossing angle, spill-over as for 25ns bunch spacing, equivalent to L=5x10^32 cm^-2s^-1 for nominal machine
MC09-b7TeV-md100-nu3,4-25ns	7 TeV	Full field downward direction	3.4	beta*=6m, nominal effective crossing angle, spill-over as for 25ns bunch spacing, equivalent to L=10x10^32 cm^-2s^-1 for nominal machine
MC09-b7TeV-md100-nu6,8-25ns	7 TeV	Full field downward direction	6.8	beta*=3m, nominal effective crossing angle, spill-over as for 25ns bunch spacing, equivalent to L=20x10^32 cm^-2s^-1 for nominal machine
MC09-b3,5TeV-md100-nu1	3.5 TeV	Full field downward direction	1	beta*=2, head on collisions hence only internal crossing angle as for up to 156 bunches (no spill-over)

1. The configurations for 7 TeV beams at nominal machine for average luminosities/bunch of $2, 5, 10$ and $20 \times 10^{32} cm^{-2} s^{-1}$ have been requested by Franz Muheim. They are available from AppConfig v3r0.

2. The configurations for 3.5 TeV beams at startup conditions (head on, nu=1) have been requested by Olivier Schneider. They are available from AppConfig v3r9

Boole

Name	DataType	Output format	Comments
EnableSpillover.py			Additional options to digitize also `Prev`, `PrevPrev`, `Next` events
FEST.py	Default	.mdf	Standard simulation
FEST-L0Yes.py	Default	.mdf	Standard simulation, writing to .mdf file only events passing L0, or with simulated random trigger.
MC09-CaloMisCalibration.py	Default	not defined	Standard simulation with Calo miscalibrated. To be used together with e.g. `MC09-NoTruth.py`
MC09-NoTruth.py	Default	'Minimal' .digi	Standard simulation, saving only RawEvent and pileup multiplicity
MC09-WithTruth.py	Default	"Default" .digi	Standard simulation, saving RawEvent and MC truth containers
2009-WithTruth.py	Default	"Default" .digi	Standard simulation, saving RawEvent and MC truth containers, with L0 trigger emulation compatible with 2009 data
Upgrade-Extended-v19r6patch1.py	Upgrade	"Extended" .digi	Same as Upgrade-WithTruth.py but saving also MCHits, for use with Boole v19r6 only
Upgrade-NoTruth.py	Upgrade	'Minimal' .digi	LHCb upgrade simulation, Spillover enabled, saving only RawEvent and pileup multiplicity
Upgrade-WithTruth.py	Upgrade	"Default" .digi	LHCb upgrade simulation, Spillover enabled, saving RawEvent and MC truth containers

1. The Default DataType is set by the application. It is "2008" until Boole v19r4, "2009" from Boole v19r5. Currently only "DC06" DataType has a special meaning in Boole, used to select the DC06 L0 trigger setting. DataType also selects the default DDDB and SimCond tags if they are not explicitly set, e.g. by providing one of the options files in `AppConfig/options/Conditions` directory.

Brunel

Name	DataType	Input format	Output format	Comments
Default.py	Default	MDF	DST	Standard reconstruction for real data
FEST-Stripping.py	Default	ETC	DST	Standard reconstruction of FEST stripping ETC
MC09-NoTruth.py	MC09	'Minimal' .digi	DST	Standard MC09 reconstruction, without using MC truth, saving only real data and pileup multiplicity
MC09-Stripping.py	MC09	ETC	DST	Standard reconstruction of FEST stripping ETC
MC09-WithTruth.py	Default	"Default" .digi	DST	Standard MC09 reconstruction, saving also MC truth containers
StoreMuonAlignTracks.py				Additional options to add MuonsForAlignment to DST
Upgrade-noaerogel.py	Upgrade			Additional options to remove aerogel from RICH reco for Upgrade data produced with no aerogel

Upgrade-NoTruth.py	Upgrade	'Minimal' .digi	DST	Reconstruction for LHCb upgrade simulation, saving only real data and pileup multiplicity
Upgrade-WithTruth.py	Upgrade	"Default" .digi	DST	Reconstruction for LHCb upgrade simulation, saving also MC truth containers
Upgrade-XDST-v35r6p1patch1.py	Upgrade	"Extended" .digi	XDST	Same as Upgrade-WithTruth.py but saving also MCHits, for use with Brunel v35r6 only
v36r0-EarlyDataPatches.py	Default	MDF	DST	Additional options to be used in v36r0(p*) for earlyData tuning
addDownstreamTracks.py				additional options to add the Downstream tracks container to the DST
allTracksProtoP.py				Additional options to make ProtoParticles for all track types (Brunel default is just ["Long", "Upstream", "Downstream"]). Useful for data where Velo and/or TT are switched off. Do not use together with firstData.py
beamGas.py				Additional options for reconstructing beam gas data
firstData.py				Additional options to select tuning for first data. Available from Brunel v35r11
moff.py				Additional options for reconstruction with magnetic field off
richLooseTrackCuts.py				Additional options to loosen track selection cuts for the Rich. Do not use together with firstData.py
unpackedDST.py				Additional options to switch off packed DST
veloOpen.py				Additional options for reconstruction with velo open

1. The Default DataType is set by the application. It is "2008" until Brunel v35r3, "2009" from Brunel v35r4.
2. Use AppConfig v2r* for Brunel up to v34r7, AppConfig v3r* from Brunel v35r0. Note that in Brunel v34r*, MC09 DataType was not known, 2008 DataType was used instead.

DaVinci

Name	DataType	Input format	Output format	Comments
DVMonitor-RealData-Dst.py	2009	DST	Histograms	Standard monitoring for Real data DSTs
DVMonitor-RealData-rDst.py	2009	rDST	Histograms	Standard monitoring for Real data rDSTs
DVMonitor-RealData-rDst.py	2009	rDST	Histograms	Standard monitoring for Real data rDSTs
DVMonitor-EXPRESS-Dst.py	2009 (Sim = True)	DST	Histograms	Standard monitoring for FEST data DSTs
DVMonitor-FULL-rDst.py	2009 (Sim	rDST	Histograms	Standard monitoring for FEST data

	= True)			rDSTs
DVStrippingETC-MC09.py	2009	DST	ETC	Stripping into ETC for MC09 data (DaVinci prior to v24r4)
DVStrippingETC-MC09-v24r4.py	2009	DST	ETC	Stripping into ETC for MC09 data (DaVinci v24r4 and after)
DVStrippingEtc-FEST.py	2009	rDST	ETC	Stripping into ETC for FEST data (DaVinci prior to v24r4)
DVStrippingEtc-FEST-v24r4.py	2009	rDST	ETC	Stripping into ETC for FEST data (DaVinci v24r4 and after)
DVStrippingDST-MC09.py	2009	DST	DST	Stripping into DST for MC09 data (DaVinci prior to v24r4)
DVStrippingDST-MC09-v24r4.py	2009	DST	DST	Stripping into DST for MC09 data (DaVinci v24r4 and after)
DVStrippingDst-FEST.py	2009	rDST	DST	Stripping into ETC for FEST data (DaVinci prior to v24r2)
DVStrippingDst-FEST09.py	2009	rDST	DST	Stripping into ETC for FEST data (DaVinci v24r2 - v24r3)
DVStrippingTagger-MC09-v24r4.py	2009	Stripped DST	SETC	Tagger for MC09 data (DaVinci v24r4 and after)

Moore

-- GloriaCorti - 2009-10-05

This topic: LHCb > AppConfig
Topic revision: r15 - 2010-02-02 - MarcoCattaneo