

Table of Contents

ST 2010 Analysis Helpline.....	1
Tags.....	1
Data.....	1
Simulation.....	1
Databases.....	1
ST VCSEL death scenarios.....	1
ST Charge Calibration (Before June 2010).....	1
Alignment.....	2
Online.....	2
TT Pitch fix.....	2
Trigger.....	2
Pileup.....	3
Selecting Beam-Beam crossings.....	3
Removing Lumi events.....	3
Book-keeping.....	3
Field Map.....	3
Example 1: Using the Tosca map.....	4
Example 2: Using the up map in the conditions for down data.....	4
Example 3: Using a different set of maps.....	4
Links.....	4

ST 2010 Analysis Helpline

Tags

You can find RecommendedTags here

Data

- DataType is now 2010
- XXXX.DDDBtag = "head-20100407"
- CondDBtag = "head-20100414"

Simulation

- 3.5 TeV min bias MC
- DataType is now 2010
- Remember Simulation = True

Some old tags ! For the latest look at RecommendedTags

- XXXX.DDDBtag = "head-20100119"
- CondDBtag = "sim-20100321-vc-md100"

Databases

ST VCSEL death scenarios

To simulate scenarios with TED VCSELs I made two scenarios, optimistic (96.8 % working) and pessimistic (93.7 % working)

Name	Text Info	File
Optimistic	optimistic.txt	optimistic.db
Pessimistic	pessimistic.txt	pessimistic.db

Both start from the July 8th tag of the database (98.5 % working detector). See: Savannah #4256

ST Charge Calibration (Before June 2010)

Official Databases

Meaning	Validity	Type	File
Latest slice for data (correct noise, charge calibration, cluster threshold info, efficiency maps)	April 2010 - inf	LHCBCOND	STCOND10.db
Mirror of STCOND10 slice for SIMCOND	April 2010 - inf	SIMCOND	SIM10C.db

To use:

```
from Configurables import ( CondDB, CondDBAccessSvc )
stcalib = CondDBAccessSvc('stCalib')
stcalib.ConnectionString = 'sqlite_file:/File/Type'
CondDB().addLayer(stcalib )
```

In addition, I prepared various intermediate slices

Meaning	Validity	Type	File
2009 charge calibration, noise + clustering thresholds	Nov 2009 - Dec 2009	SIMCOND	calib09.db
MC09 charge calibration, noise + clustering thresholds	MC09	SIMCOND	CON1.db
MC09 charge calibration, noise + 2010 clustering thresholds	-	SIMCOND	CON2.db
MC09 charge calibration, noise + 2009 clustering thresholds	-	SIMCOND	CON3.db

Alignment

If you use ~ Brunel v37r1 era data you need to do nothing. If you use earlier versions read on

Wouter has databases in his area. Last 2009 database is tagged as v2.3, first 2010 as v2.4.

```
from Configurables import ( CondDB, CondDBAccessSvc )
mycalib = CondDBAccessSvc('myCalib')
mycalib.ConnectionString = 'sqlite_file:/afs/cern.ch/user/w/wouter/public/AlignDB/v2.4.db/LHCb'
CondDB().addLayer( mycalib )
```

Online

If the data is new you need to either use Oracle[?] or Matt has a slice he tries to keep up to date (use at own risk):

```
from Configurables import ( CondDB, CondDBAccessSvc )
myOnline = CondDBAccessSvc( 'MyOnline' )
myOnline.ConnectionString = 'sqlite_file:/afs/cern.ch/lhcb/group/tracking/voll/mneedham/ONLINE'
CondDB().addLayer( myOnline )
```

TT Pitch fix

The pitch was wrong, a fix + fixed alignment is available. FIXED by ~ Brunel v37r1

```
from Configurables import ( CondDB, CondDBAccessSvc )
mycalib = CondDBAccessSvc('myCalib')
mycalib.ConnectionString = 'sqlite_file:/afs/cern.ch/user/w/wouter/public/AlignDB/v2.4.db/LHCb'
CondDB().addLayer( mycalib )
ttPitchFix = CondDBAccessSvc( 'TTPitchFix' )
ttPitchFix.ConnectionString = 'sqlite_file:/afs/cern.ch/user/w/wouter/public/AlignDB/TTPitchFix'
```

Trigger

For many studies [e.g. occupancies] you should select a trigger line.

- A Trigger twiki
- Tutorial on the trigger[?]
- The TCK is available in the Odin bank
- There is a useful command line tool in the DaVinci environment to give the lines in the TCK : at the prompt type TCKsh, then e.g. listHlt1Lines(0x00021810)
- Or use the Online TCK presenter[?]

To create and use a trigger filter in python you do something like:

```
from Configurables import LoKi__HDRFilter as HltFilter
trigFilter = HltFilter( 'HltPassFilter', Code="HLT_PASS('Hlt1MBMicroBiasRZVeloDecision')"
```

Pileup

The 2010 simulation has $\nu = 1$. To get the pile-up in the early data:

```
from Configurables import LoKi__VoidFilter
fltr = LoKi__VoidFilter( 'GenFilter' , Code = " 1 == CONTAINS('Gen/Collisions') ")
```

Selecting Beam-Beam crossings

The following filter does the job

```
from Configurables import LoKi__ODINFilter
fltr = LoKi__ODINFilter( 'O1' , Code = " ODIN_BXTYP == LHCb.ODIN.BeamCrossing " )
```

Removing Lumi events

The following filter does the job

```
# routing bit filter to remove lumi
from Configurables import HltRoutingBitsFilter
physFilter = HltRoutingBitsFilter( "PhysFilter", RequireMask = [ 0x0, 0x4, 0x0 ] )
```

Book-keeping

- [Web access](#)
- The RecommendedTags page explains how to find information on production conditions

Field Map

There are several field maps around. Though the original maps are used the modified maps are preferred and will be used in the future

Map	Polarity	Names	Comments
Tosca	Down	field047.cdf	One quadrant that is reflected
Original down	Down	field048.c1.vs.down.cdf	Used in Brunel v37r1
Original up	Up	field048.c1.an.up.cdf	Used in Brunel v37r1
Modified down	Down	c1_downward_20100106.cdf	Should improve on the original map
Modified up	Up	c1_upward_20100106.cdf	Should improve on the original map
Modified up/down	Down	c1_upward_and_downward_20100112.cdf	Use all the data in the parameterization (obsolete - use 0100517)
Modified up/down	Down	c1_upward_and_downward_20100517.cdf	Use all the data in the parameterization, better behaviour at acceptance edge

Comments:

- I put all the maps in `/afs/cern.ch/lhcb/group/tracking/vol1/mneedham/FieldMap/cdf`
- You can use an Up map for Down data by putting a negative scale factor
`MagneticFieldSvc().ForcedSignedCurrentScaling = -1`

Example 1: Using the Tosca map

```
from Configurables import MagneticFieldSvc
MagneticFieldSvc().UseConditions = False
MagneticFieldSvc().FieldMapFiles = ["/afs/cern.ch/lhcb/software/releases/DBASE/FieldMap/v5r3/cdf/
```

Example 2: Using the up map in the conditions for down data

```
from Configurables import MagneticFieldSvc
MagneticFieldSvc().ForceToUseUpMap = True
MagneticFieldSvc().ForcedSignedCurrentScaling = -1
```

Example 3: Using a different set of maps

```
from Configurables import MagneticFieldSvc
MagneticFieldSvc().UseConditions = False
MagneticFieldSvc().FieldMapFiles = ["/afs/cern.ch/lhcb/group/tracking/voll/mneedham/FieldMap/cdf/
"/afs/cern.ch/lhcb/group/tracking/voll/mneedham/FieldMap/cdf/
"/afs/cern.ch/lhcb/group/tracking/voll/mneedham/FieldMap/cdf/
"/afs/cern.ch/lhcb/group/tracking/voll/mneedham/FieldMap/cdf/
```

Nota Bene:

- In old versions of the code ForcedSignedCurrentScaling was called ScaleFactor
- Instead of ForceToUseUpMap and ForceToUseDownMap there was Polarity option [+/-1]

Links

- Lots of useful information on ST software can be found here [↗](#)
- Viper python scripts
- Tutorial on filters etc [↗](#)

-- MatthewNeedham - 08-Apr-2010

This topic: LHCb > LHCbSTSoftwareHelp2010

Topic revision: r17 - 2013-02-26 - SandraSaornil



Copyright &© 2008-2019 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

Ideas, requests, problems regarding TWiki? Send feedback