Simulation tools for physics

Matteo Rama - INFN Lab. Nazionali Frascati
for the FastSim working group

SuperB meeting, Orsay, Feb. 2009

FastSim – fast simulation of SuperB detector

As of today FastSim is the only tool available to do Physics simulations

Main features of FastSim design:

- Detector elements description
  - cylinders, cones, rings,…
  - xml-based configuration (geometry, material, resolutions)
- Passage of particles through matter
  - energy loss, multiple scattering, bremsstrahlung, Compton scattering, pair production, …
- Detector response
  - track hit resolution, cluster shape, Cherenkov photon resolution, …
- Particle reconstruction
  - charged tracks, calorimeter clusters, Cherenkov rings, …

Outline

- Physics analysis with FastSim in 5 steps
  - Setup the release
  - Configure the detector
  - Choose the event generator
  - Configure your signal selector
  - Run the simulation
- Status of the art and future developments

I - Setup the release

- The FastSim release system uses subversion as code repository technology
- FastSim currently still requires the BaBar software release framework to work.
  The plan is to announce a BaBar-independent release within ~2 months
- Setup instructions in FastSim User Guide:
  http://mailman.fe.infn.it/superbwiki/index.php/SuperB_fast_simulation_User_Guide

snapshot of instructions to setup a release

```
newrel -s $BFROOT/build/<first_letter>/<username> -t 24.3.5 2435
cd 2435
srtpath <enter> <enter>
addpkg workdir
gmake workdir.setup
```

create test rel.

```
setenv SVNROOT https://opteron05.lbl.gov/svn
wget http://opteron05.lbl.gov/~brownd/SuperB/sbboot
chmod u+x ./sbboot
./sbboot
SvnTools/newrel FastSim/V0.0.1
SvnTools/addpkg
```

download code from repo

```
make ldlink
gmake installdirs
gmake lib
gmake PacMC.bin
```

link and compile

2 – Configure the detector (if you need to do it)

- XML-based description of geometry, materials and resolutions

measurement parameters (DCH in this case):

```
<device name="Axial"
      type="DriftChamber"
       rms_par0="0.0125"
       rms_par1="0"
       rms_par2="0"
       rms_par3="0"
       rms_par4="0"
       rms_par5="0"
       eff="0.99"
       angle="0"  />
```

layers of wires and material definition:

```
<cyl name="DchLayer" id="1" zmin="-101.5" zmax="174.9" radius="26.040" thick="0.10" meas="Axial" />
<cyl name="DchLayer" id="2" zmin="-101.5" zmax="174.9" radius="27.230" thick="0.10" meas="Axial" />
<cyl name="DchLayer" id="3" zmin="-101.5" zmax="174.9" radius="28.420" thick="0.10" meas="Axial" />
<cyl name="dch-Wires" id="151" zmin="-101.5" zmax="174.9" radius="28.420" thick="0.0120" mat="dch-Wires_12" gap="0.946" />
<cyl name="dch-He-Ibu" id="152" zmin="-101.5" zmax="174.9" radius="28.421" thick="5.7300" mat="dch-He-Ibu_12" />
<cyl name="DchLayer" id="4" zmin="-101.5" zmax="174.9" radius="29.610" thick="0.10" meas="Axial" />
<cyl name="DchLayer" id="5" zmin="-101.5" zmax="174.9" radius="31.240" thick="0.10" meas="Stereo+" />
```

3 – Choose the event generator

- Simulation is largely configurable via tcl files (which avoids recompiling after every change). The event generator is set via tcl, with parameters stored in .dec files

```
Alias MyB0  B0
#
Decay Upsilon(4S)
1.000   MyB0        anti-B0                   VSS_MIX dm;
Enddecay
#
Decay MyB0
1.0000 pi+ pi-                      PHSP;
Enddecay
```

```
disableGenerators 0
module enable GfiEvtGen
talkto GfiEvtGen {
    GENERATE set Upsilon(4S)
    DECAY    set RELEASE/PacMC/DECAY.DEC
    UDECAY   set PARENT/ProdDecayFiles/B0barallB0pipi_crn.dec
}
```

- Event generators:

events	generator	status
e+e-→Y(4S)→Bbbar	EvtGen	ok
e+e-→qqbar (q=u,d,s,c)	EvtGen	ok
e+e→Psi(3770)→DDbar	EvtGen	ok
e+e-→tau+tau- including polarized beams	kk2f	setup for FastSim starting

Do we urgently need other generators?

4 - Configure your signal 'selector'

- The output of FastSim is a list of *BtaCandidates* (inherited by BaBar), objects representing *reconstructed particle candidates*.

➔ Use the composition tools developed in BaBar.

```
## Make a B0->pipi from two charged tracks
mod clone SmpMakerDefiner SmpMyBtopipi
seq append MyAnalysisSequence SmpMyBtopipi
talkto SmpMyBtopipi {
    decayMode set "B0 -> pi+ pi-"
    daughterListNames set GoodTracksLoose
    daughterListNames set GoodTracksLoose
    fittingAlgorithm  set Cascade
    fitConstraints    set Geo
    postFitSelectors  set "Mes 5.2:5.3"
    postFitSelectors  set "DeltaE -0.1:0.1"
    createUsrData     set true
}
```

PacMC/example_PacTrk.tcl

- To store relevant quantities in root files for later analysis you can either use existing user packages, or write your own module.
  Ex: BtaTupleMaker for fully reconstructed B decays (configurable via tcl).

5 - Run the simulation

- run FastSim

```
../bin/Linux24SL3_i386_gcc323/PacMCApp ../PacMC/example_PacTrk.tcl
```

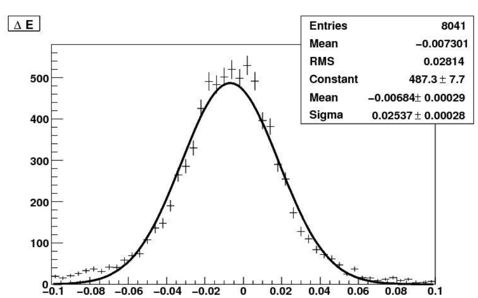
- Speed

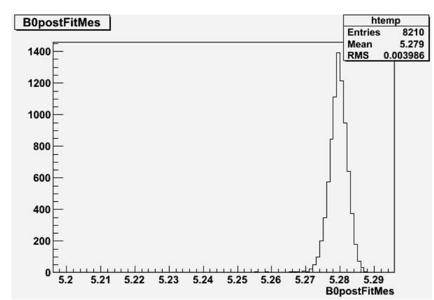
ms

```
AppAST:     0.00 |   10000 |      9.44 |        0.94400 |   0.00 |GoodTrackVeryLooseSelection
AppAST:     0.00 |   10000 |     15.17 |        1.51700 |   0.00 |RacTestInput
AppAST:     1.20 |   10000 |     27.09 |        2.70900 |   0.00 |GfiEvtGen
AppAST:     1.22 |   10000 |     28.12 |        2.81200 |   0.00 |PmcSimulate
AppAST:     0.00 |   10000 |     69.42 |        6.94200 |   0.00 |BtuTupleMaker
AppAST:     0.00 |   10000 |     97.70 |        9.77000 |   0.00 |PmcReconstruct
```

typical CPU time/evt ~20-25ms/evt (dual core cpus)

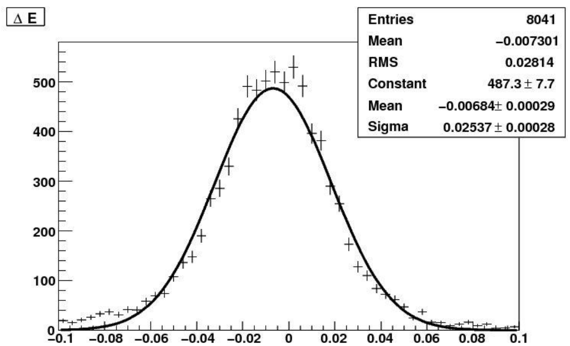
- Reconstructed signal





Tracking

- SVT: SuperB baseline and BaBar configuration available
- DCH: SuperB baseline and BaBar configuration available



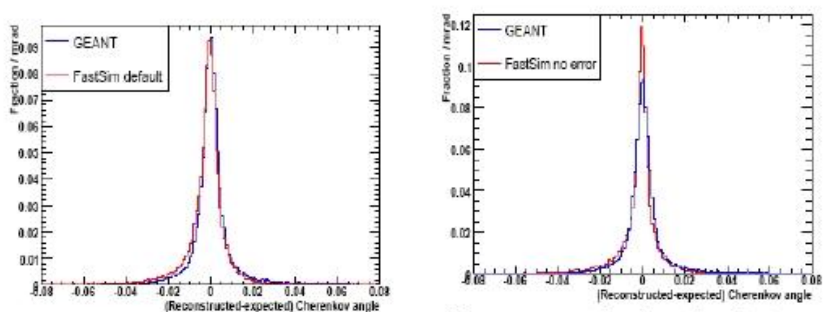
reso. [MeV]	FastSim	FullSim
ΔE (B→π+π-)	25.4±0.3	29.9±0.5
ΔE (B→Phi Ks)	15.6±0.2	21.8±0.3
Pt [1.0,2.0]	10.2±0.2	11.9±0.3
Pt [2.0,2.5]	13.4±0.1	16.7±0.3
Pt [2.5,3.0]	15.8±0.2	19.3±0.3

- 15-20% difference reso. difference between BaBar FastSim and BaBar full sim.

  → likely main reason: hit confusion, not implemented yet -- Work in progress
- dE/dx measurement not implemented yet: goal is to have it on the Warwick meeting time scale

PID

- DIRC in BaBar default configurations
  - use ring dictionary, now outside the BaBar condition DB (important step towards standalone FastSim)



Default distributions.

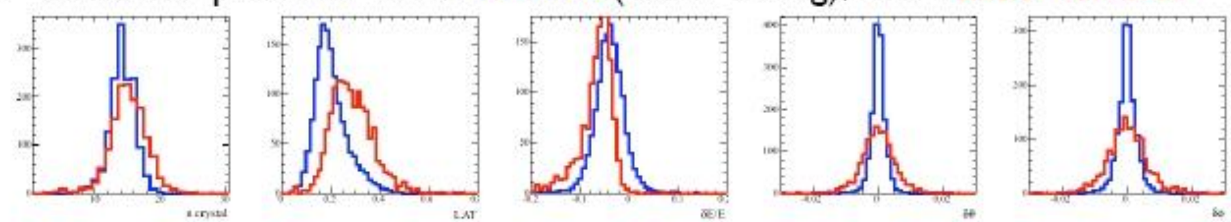
No errors - geometric, achromatic, and extra error terms all set to zero.

good agreement between fast and full simulation

- Forward PID options:
  - Work going on (see Dave's talk in prev. session)
- PID selectors: combine >1 PID information to provide the best guess on particle nature
  - generally have to deal with correlations
  - activity now started

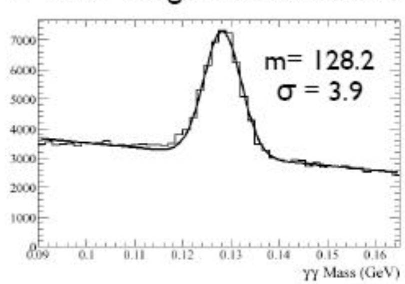
Calorimeters

- EMC simulation still needs some tuning but results are encouraging. Both forward (LYSO) and backward (Pb-scint.) endcap available.
  - One-GeV photons: Blue= FastSim (BaBar config); Red= BaBar full Sim



crystal   LAT moment   δEraw/Eraw   δθ   δφ

  - π^0 from BB generic simulation



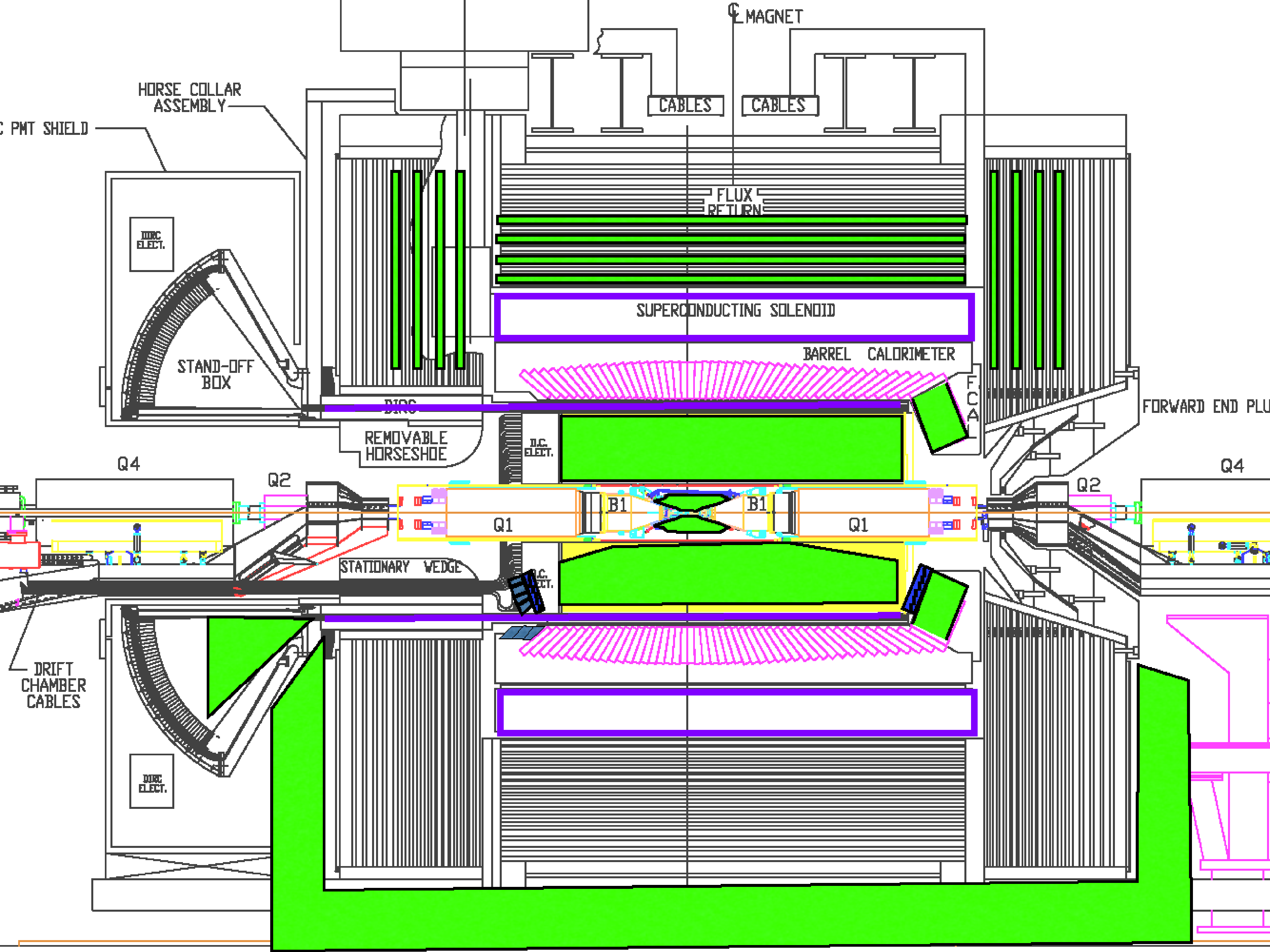
Resolution are still too optimistic.

- hadronic shower
  - Need to find better parameterization. Simulation output not usable yet for physics/optimization studies

Status of some important items

name	description	status
hit confusion	parametrization of effects given by pattern recognition	covered, work in progress
dE/dx	energy loss is simulated in FastSim, but not its measurement	not covered, but 1 person from next week!
tagging	need to setup it in FastSim and possibly implement new features	now 1 person!
PID selectors	needed to do analysis and optimization studies	work started, need complete set of PID info
tag vertex	see if we can exploit better vertex resolution	
Vertex and kinematic fits		ok
tuning of EMC response		work in progress
IFR response	need to find a way to improve IFR response	work in progress
forw. PID	implement response	work started

BACKUP



℄ MAGNET
HORSE COLLAR
ASSEMBLY
PMT SHIELD
CABLES
CABLES
FLUX
RETURN
DIRC
ELECT.
SUPERCONDUCTING SOLENOID
BARREL CALORIMETER
STAND-OFF
BOX
F
C
A
FORWARD END PLU
DIRC
REMOVABLE
HORSESHOE
D.C.
ELECT.
Q4
Q2
Q4
Q2
B1
B1
Q1
Q1
STATIONARY WEDGE
D.C.
ELECT.
DRIFT
CHAMBER
CABLES
DIRC
ELECT.

current activity

	People (22/jan/09)	Detector options	Optimization studies	Physics benchmarks	Items needing development
SVT	D. Brown, N. Neri, D. Roberts, G. Simi		internal geometry, radius of outer layer	B→Ksπ^0/Ks$\pi^0\gamma$, beta, Recoil, (tagging)	▪ dE/dx ▪ endcap PID response ▪ PID selectors ▪ tuning of EMC response ▪ hadron shower sim. ▪ Flavour tagging ▪ Tag vertex
DCH	M. Rama, G. Finocchiaro		longer DCH replacing forw. PID, inner radius	tracking performance, dE/dx	
PID	A. Stocchi, L. Burmistrov, N. Arnaud, A. Perez, A. Berdyugin, B. Meadows, F. Renga	forward PID yes/no, backward PID yes/no	angular and momentum coverage range, needed PID performance, #rad. length (impact on endcap EMC performance)	B→(d,s) l^+l^-, Recoil, tagging, B→Knunubar, B→tau nu	
EMC	C. Cheng, E. Manoni, A. Rossi	backward EMC yes/no	angular coverage of forw/back endcaps, needed performance, degradation due to endcap PID	B→Ksπ^0/Ks$\pi^0\gamma$, B→tau nu, b→sγ, B→K nu nubar Recoil, tagging	
IFR	G. Cibinetto, M. Rotondo		amount and distribution of absorber	beta, Recoil, tagging	
			Other: position of IR vertex		