

# EvtGen Status



Anders Ryd  
Cornell U.  
Jan 13, 2009

*Alea iacta est*

# EvtGen Status

## Outline:

- EvtGen Status and Background
- EvtGenLHC
- Recent developments

# Motivation

- EvtGen provides two main tools
- Detailed decay tables for decays of Upsilon states and lighter particles (e.g. B and D mesons, Charmonium)
- Tools for simulation of kinematics based on amplitudes.
  - This uses a modular framework to assemble amplitudes to calculate the total probability.

# Sequential Decays

- Many decays have interesting sequential decay chains:

$$B \rightarrow D^* \ell \nu \quad B \rightarrow D^* \ell \nu$$
$$\downarrow_{D\pi} \quad \downarrow_{D\gamma}$$

$$B \rightarrow D^* D^* \quad B \rightarrow D^* D^*$$
$$\downarrow_{D\pi} \downarrow_{D\pi} \quad \downarrow_{D\pi} \downarrow_{D\gamma}$$

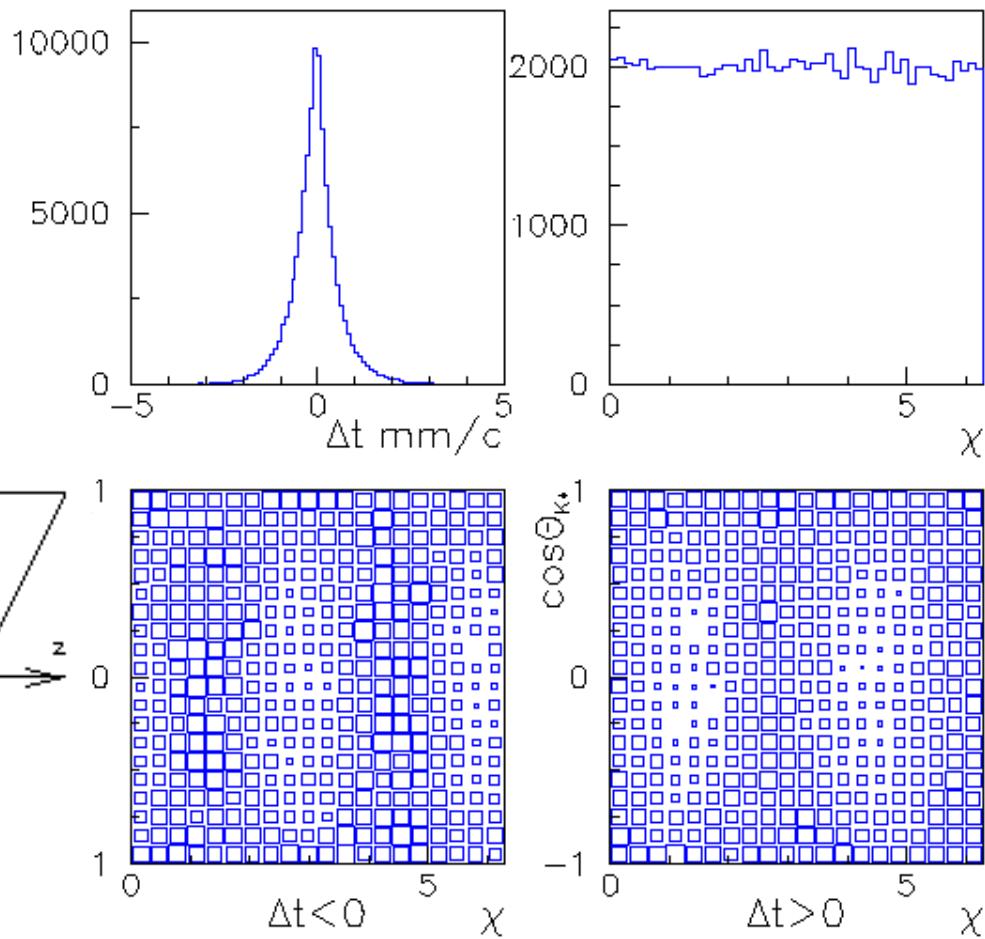
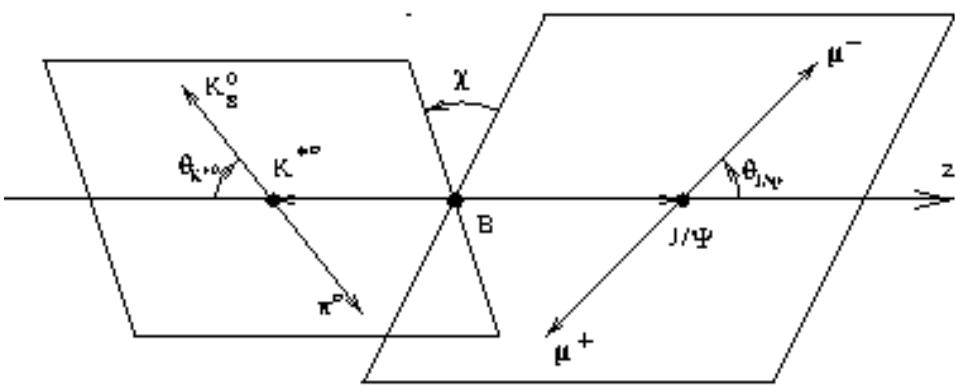
- Want to correctly simulate these decay chains while only implementing the nodes in the decay tree.

ie nodes in the decay tree:

$$B \rightarrow D^* \ell \nu \quad B \rightarrow D^* D^*$$
$$D^* \rightarrow D\pi \quad D^* \rightarrow D\gamma$$

# CP Violating Decays

- $B \rightarrow J/\psi K^{*0}$  ( $K^{*0} \rightarrow K^0 \pi^0$ )
  - Angular correlations and time dependence



# EvtGenLHC

- The version of EvtGen used at LHC was split off from the development at BABAR in 2003.
  - The LHC version does not have any improvements made at BABAR, or elsewhere, for the last 5+ years!

# Code Developments since 2003

- Mass generation improved; 2003 version used Breit-Wigner lineshape for all resonances. Now includes much more information about phase space and partial waves when calculating the lineshape.
- Addition of new decay models:
  - ◆ Semileptonic B and D decays
  - ◆ Three-body 'Dalitz' decays for B and D decays
- Many code fixes
  - ◆ HelAmp and PartWave models
  - ◆ SSDCP

# Decay Table Updates

- There has been several significant updates to the decay table:
  - ◆ Major updates in 2004 from BABAR
  - ◆ Updates from CLEO-c and CDF (around 2004-5)
  - ◆ Smaller updates from BABAR in 2006
  - ◆ CLEO-c made a major tuning of charm and charmonium decays in 2007. These changes are so far not merged into the BABAR version.
    - Plan to merge these changes in the next month or so.
  - ◆ CDF also has updates that are not yet merged.
- I think that most of the interest here would be in the improved tuning of the lepton spectra and particle multiplicities.

# Summary

- EvtGen has slowly been developed at BABAR, CLEO-c and elsewhere.
- I would like to encourage that the EvtGen version use was updated to the latest version.

# Decay amplitudes are used instead of probabilities

- EvtGen works with amplitudes to correctly handle sequential decays:

$$\begin{array}{c} B \rightarrow D^* \quad \tau \nu \\ \downarrow D\pi \quad \downarrow \pi \nu \end{array}$$

$$d\Gamma = |A|^2 d\phi \quad A = \sum_{\lambda_{D^*} \lambda_\tau} A_{\lambda_{D^*} \lambda_\tau}^{B \rightarrow D^* \tau \nu} A_{\lambda_{D^*}}^{D^* \rightarrow D\pi} A_{\lambda_\tau}^{\tau \rightarrow \pi \nu}$$
$$A_{\lambda_{D^*} \lambda_\tau}^{B \rightarrow D^* \tau \nu} \equiv \langle \lambda_{D^*} \lambda_\tau | H | B \rangle \quad \sum_{\lambda_{D^*}} |\lambda_{D^*} \rangle \langle \lambda_{D^*}| = I$$

- Nodes in the decay tree are implemented as “models”. The framework of EvtGen handles the bookkeeping needed to correctly generate the full decay tree.

# Selection algorithm (I)

- Generate the  $B \rightarrow D^* l \nu$  decay

$$P = \sum_{\lambda_{D^*} \lambda_\tau} |A_{\lambda_{D^*} \lambda_\tau}^{B \rightarrow D^* \tau \nu}|^2$$

- Compare with maximum probability and accept or reject generated  $B \rightarrow D^* l \nu$  decay.
  - Maximum probability specified in code.
    - Can instead be generated on the fly, however this leads to the output of event  $N$  depending on the random number sequence used to determine the max probability.
- Regenerate  $B \rightarrow D^* l \nu$  decay until combination is accepted.

# Selection algorithm (II)

- Average over  $\tau$  spin and calculate the  $D^*$  spin density matrix:

$$\rho_{\lambda_{D^*} \lambda'_{D^*}}^{D^*} = \sum_{\lambda_\tau} A_{\lambda_{D^*} \lambda_\tau}^{B \rightarrow D^* \tau \nu} (A_{\lambda'_{D^*} \lambda_\tau}^{B \rightarrow D^* \tau \nu})^*$$

- Generate the  $D^* \rightarrow D\pi$  decay

$$P = \sum_{\lambda_{D^*} \lambda'_{D^*}} \rho_{\lambda_{D^*} \lambda'_{D^*}}^{D^*} A_{\lambda_{D^*}}^{D^* \rightarrow D\pi} (A_{\lambda'_{D^*}}^{D^* \rightarrow D\pi})^*$$

- Compare with maximum probability and accept or reject generated  $D^* \rightarrow D\pi$  decay
- Regenerate  $D^* \rightarrow D\pi$  decay until accepted. The  $B \rightarrow D^*/\nu$  decay is **not** regenerated.

# Selection algorithm (III)

- Calculate the spin density matrix for the  $\tau$

$$\rho_{\lambda_\tau \lambda'_\tau}^\tau = \sum_{\lambda_{D^*} \lambda'_{D^*}} \hat{\rho}_{\lambda_{D^*} \lambda'_{D^*}}^{D^*} A_{\lambda_{D^*} \lambda_\tau}^{B \rightarrow D^* \tau \nu} (A_{\lambda'_{D^*} \lambda'_\tau}^{B \rightarrow D^* \tau \nu})^*$$

- Where:

$$\hat{\rho}_{\lambda_{D^*} \lambda'_{D^*}}^{D^*} \equiv A_{\lambda_{D^*}}^{D^* \rightarrow D\pi} (A_{\lambda'_{D^*}}^{D^* \rightarrow D\pi})^*$$

- Generate the  $\tau \rightarrow \pi \nu$  decay

$$P = \sum_{\lambda_\tau \lambda'_\tau} \rho_{\lambda_\tau \lambda'_\tau}^\tau A_{\lambda_\tau}^{\tau \rightarrow \pi \nu} (A_{\lambda'_\tau}^{\tau \rightarrow \pi \nu})^*$$

- Compare with maximum probability and accept or reject generated  $\tau \rightarrow \pi \nu$  decay.
- Regenerate  $\tau \rightarrow \pi \nu$  decay until accepted. The  $B \rightarrow D^* l \nu$  and  $D^* \rightarrow D\pi$  decays are not regenerated.

# Advantages to using decay amplitudes

- Implementation of decay models is simplified by using amplitudes instead of probabilities.
- Keeping track of the spin density matrices allows us to generate each node of the decay chain independently.
  - More efficient
  - Avoids the need to determine uncountable # of maximum probabilities
- Generalizes to arbitrarily long decay chains
- Calculation of probabilities and spin density matrices are done by the framework. Models specify only the decay amplitudes.
- **However: No interference between particles on different branches of decay tree.**

# States in EvtGen

- EvtGen works with amplitudes. The amplitudes are specified as amplitudes between the initial and final state in a set of basis vector provided by EvtGen.
- EvtGen uses the following representation for the lower spin states:

Class name	Rep.	J	States	Example
EvtScalarParticle	1	0	1	$\pi, B^0$
EvtDiracParticle	$u_\alpha$	$1/2$	2	$e, \tau$
EvtNeutrinoParticle	$u_\alpha$	$1/2$	1	$\nu_e$
EvtVectorParticle	$\epsilon^\mu$	1	3	$\rho, J/\Psi$
EvtPhotonParticle	$\epsilon^\mu$	1	2	$\gamma$
EvtTensorParticle	$T^{\mu\nu}$	2	5	$D_2^*, f_2$

- Also  $J=3/2$  EvtRaritaSchwinger 4 states
- Higher spin states are represented by a generic helicity state basis

# EvtGen decay algorithm

1. Input: Parent particle Id and p4



2. Determine decay tree (completely)

Input from  
DECAY.DEC



3. Determine mass of each particle in tree

Input from  
evt.pdl



4. Accept/reject to determine kinematics

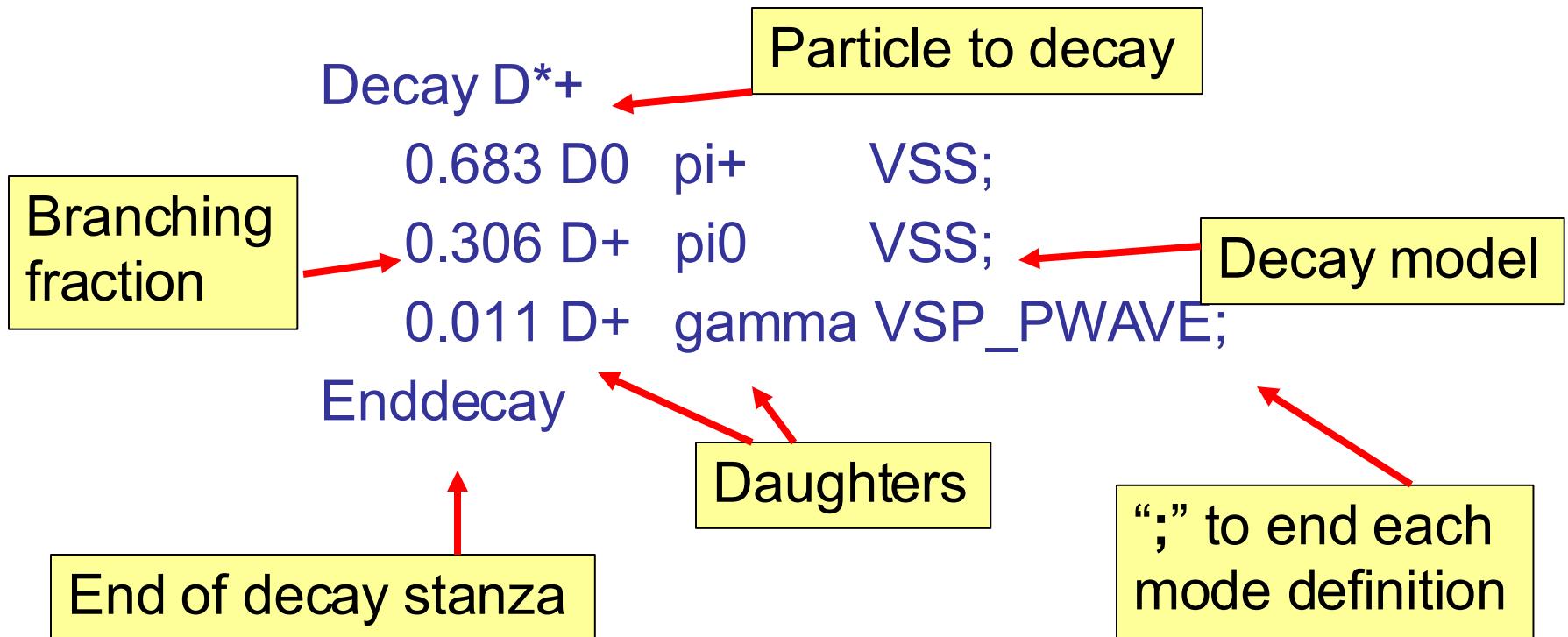
- Configuration specified by input files at run time.
  - Users override generic DECAY.DEC to generate MC as needed.

# The decay table (DECAY.DEC)

- We continue to increase the ability to control EvtGen via `$C3_DATA/DECAY.DEC`
  - Decays and branching fractions
  - Particle masses, widths, lineshapes
  - Try to avoid hardwiring numbers that control decay models, instead specifying them as arguments.
  - Control of usage of PHOTOS packages

Additional control avoids the need to change software to produce MC for systematic studies

# Defining particle decays



Defines three decay modes of the D\*+

Branching fractions will be rescaled to sum to 1.0

# Particle “aliases”

Alias MyD<sup>\*</sup>+ D<sup>\*</sup>+

Decay B0

1.0 MyD<sup>\*</sup>+ pi-      SVS;

Enddecay

Decay MyD<sup>\*</sup>+

1.0 D0      pi+      VSS;

Enddecay

- In this case, all B0s will decay to D<sup>\*+</sup> $\pi^-$ , with D<sup>\*+</sup> $\rightarrow$ D<sup>0</sup> $\pi^+$ . However, other D<sup>\*+</sup> in the event will decay as defined in DECAY.DEC.

# Model arguments

Some models takes arguments:

Decay B0

1.00 D\*- e+ nu\_e

Enddecay

HQET parameters



PHOTOS HQET 0.92 1.18 0.72;

These arguments can be accessed in the model using the methods:

getNArg() returns the number of arguments

getArg(i) returns the ith argument

# evt.pdl format

Particle properties are defined in **\$C3\_DATA/evt.pdl**:

Add p Lepton mu-	13	0.1056584	0	0	-3	1	658654.	13
Add p Lepton mu+	-13	0.1056584	0	0	3	1	658654.	0
Add p Meson pi+	211	0.139570	0	0	3	0	7804.5	101
Add p Meson pi-	-211	0.138570	0	0	-3	0	7804.5	0
Add p Meson rho+	213	0.7685	0.151	0.4	3	2	0	121
Add p Meson rho-	-213	0.7685	0.151	0.4	-3	2	0	0

.....

- 4<sup>th</sup> column=particle name, 5<sup>th</sup>=stdhep number, 6<sup>th</sup>=mass (GeV/c<sup>2</sup>), 7<sup>th</sup>=Width (GeV/c<sup>2</sup>), 8<sup>th</sup>=Mass cutoff, 9<sup>th</sup>=3\*charge, 10th 2\*spin, 11<sup>th</sup>=ct (mm), 12<sup>th</sup> Lund-KC number (for Pythia interface)

# Available decay models

- General purpose models that decay according to specified helicity or partial wave amplitudes
  - Handle decays to two body final states with arbitrary spins. Amplitudes specified at run time.
- Specific CP violating models
- Semileptonic form-factor models
- Dalitz decays
  - Specific:  $D$ ,  $\eta$ ,  $\pi^0$ ,  $\omega$
  - General Pseudoscalar  $\rightarrow$  3 Pseudoscalar
- $B \rightarrow K\bar{l}l$ ,  $b \rightarrow s\gamma$
- Use PHOTOS package for final state radiation.
  - On by default for all decays.

# Semileptonic decays

- HQET - Heavy Quark Effective Theory inspired form factor param.
- ISGW, ISGW2 - Quark model based prediction, Isgur, Scora et al.
- MELIKHOV - Quark model based prediction
- SLPOLE - Generic specification of form factors based on a lattice inspired parametrization.
- VUB - For generic  $b \rightarrow ul\nu$  decays, uses JetSet for fragmentation.
- GOITY\_ROBERTS - Decays to non resonant  $D^{(*)}\pi l\nu$ .

BABAR uses, HQET, ISGW2, VUB, and GOITY\_ROBERTS in its simulation.

ISGW2 should support  $D$ ,  $D_s$  and  $B_s$  decays as well as  $B$  decays.

# Generic amplitudes

- HELAMP, PARTWAVE - generic two-body decays specified by the helicity or partial wave amplitudes.
- SLN - Decay of scalar to lepton and neutrino.
- PHSP - N-body phase space.
- SVS, STS - Scalar decay to vector (or tensor) and scalar.
- VSS, TSS - decay of vector or tensor particle to a pair of scalars.
- VLL, SLL - Decay of vector or scalar to two leptons.
- VSP\_PWAVE, vector to scalar and photon, e.g.,  
 $D^* \rightarrow D\gamma$

# Special matrix elements

- **BTOXSGAMMA** -  $b \rightarrow X_s \gamma$  with JetSet fragmentation.
- **BTOXSLL** -  $b \rightarrow X_s ll$  with JetSet fragmentation.
- **D\_DALITZ** - 3-body  $D$ -decays with substructure.
- **ETA\_DALITZ** -  $\eta \rightarrow 3\pi$  with measured dalitz amplitude.
- **KSTARNNU** -  $B \rightarrow K^* \bar{n} n \bar{u} \bar{d}$
- **LNUGAMMA** -  $B \rightarrow l \nu \gamma$
- **OMEGA\_DALITZ** - Dalitz structure in the  $\omega \rightarrow 3\pi$  decay
- **PHI\_DALITZ** - Dalitz structure in the  $\phi \rightarrow 3\pi$  decay
- **PTO3P** - scalar to 3 scalars decay where you can specify intermediate resonances
- **TAUHADNU** - hadronic 1, 2, and 3 pion final states.
- **TAULNNU** - leptonic tau decays.
- **VSS\_BMIX** - Upsilon(4S) to  $B\bar{B}$ , including mixing.
- **VVPIPI** - decay of vector to vector and two pions, e.g.  
 $\psi' \rightarrow \psi \pi \pi$ .
- **VECTORISR** - ISR production of vector mesons:  
 $e^+e^- \rightarrow V\gamma$

# Writing new Physics Models

- This part of the tutorial deals with writing new models
  - ★ A model is a C++ class that implements the calculation of amplitudes for a given process.
  - ★ This class has to be registered with the frame work in order to be used.
  - ★ The model has a name which is used to indentify the model in the decay table.
- There are currently about 80 decay models implemented in EvtGen.

# Example decay: V->SS

To illustrate how a decay model is written we will use the example of the decay of a vector particle to two scalars. The amplitude for this decay is given simply by:

$$A = \varepsilon^\mu v_\mu$$

Where  $\varepsilon$  is the polarization vector of the initial vector meson and  $v$  is the four-velocity of one of the final state particles.

We will illustrate how we write the class, EvtVSS, to implement the calculation of this amplitude for a model named 'VSS'.

# EvtVSS.hh (simplified)

```
#ifndef EVTSS_HH
#define EVTSS_HH

#include "EvtGenBase/EvtDecayAmp.hh"

class EvtParticle;

class EvtVSS:public EvtDecayAmp {
public:
    EvtVSS() {}
    virtual ~EvtVSS();

    void getName(std::string& name);
    EvtDecayBase* clone();

    void decay(EvtParticle *p);
    void init();
    void initProbMax();

};

#endif
```

# EvtVSS.cc

```
#include <stdlib.h>
#include "EvtGenBase/EvtParticle.hh"
#include "EvtGenBase/EvtGenKine.hh"
#include "EvtGenBase/EvtPDL.hh"
#include "EvtGenBase/EvtVector4C.hh"
#include "EvtGenBase/EvtVector4R.hh"
#include "EvtGenBase/EvtReport.hh"
#include "EvtGenModels/EvtVSS.hh"
#include <string>

EvtVSS::~EvtVSS() {}

void EvtVSS::getName(std::string& model_name){
    model_name="VSS";
}

EvtDecayBase* EvtVSS::clone(){
    return new EvtVSS;
}

void EvtVSS::initProbMax() {
    setProbMax(1.0);
}

void EvtVSS::init() {
    // check that there are 0 arguments
    checkNArg(0);

    // check that there are 2 daughters
    checkNDaug(2);

    // check the parent and daughter spins
    checkSpinParent(EvtSpinType::VECTOR);
    checkSpinDaughter(0,EvtSpinType::SCALAR);
    checkSpinDaughter(1,EvtSpinType::SCALAR);
}

void EvtVSS::decay( EvtParticle *p){

    p->initializePhaseSpace(getNDaug(),getDaugs());

    EvtVector4R pdaug = p->getDaug(0)->getP4();

    double norm=1.0/pdaug.d3mag();
    vertex(0,norm*pdaug*(p->eps(0)));
    vertex(1,norm*pdaug*(p->eps(1)));
    vertex(2,norm*pdaug*(p->eps(2)));

    return;
}
```

# Registering the model

The last step to do before you can use a model is to register it with the EvtGen framework. This is done in the EvtModelReg.cc:

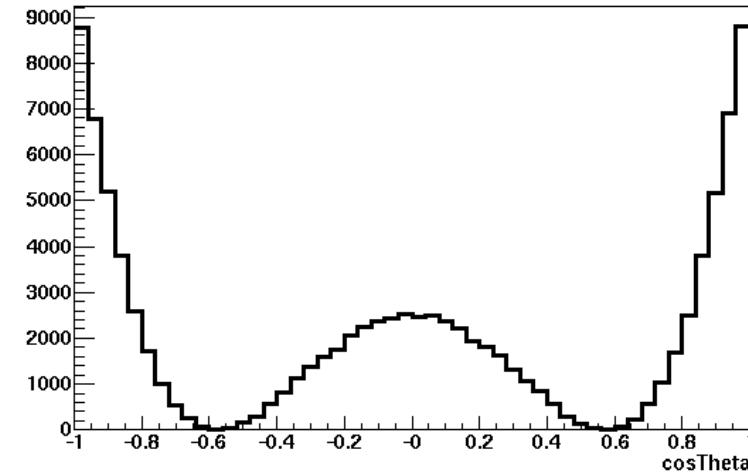
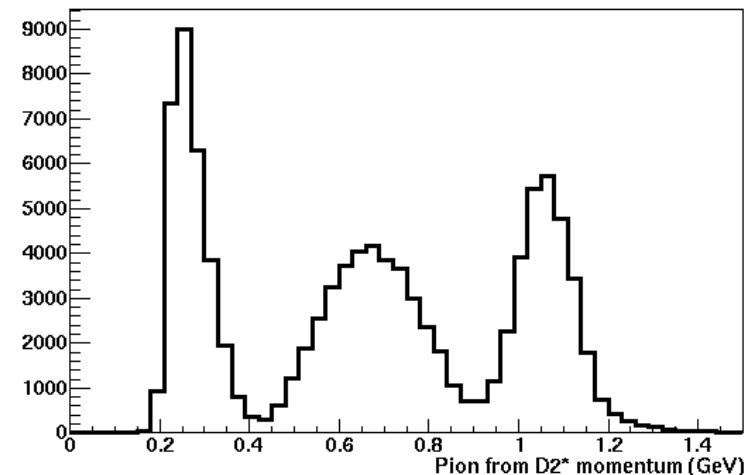
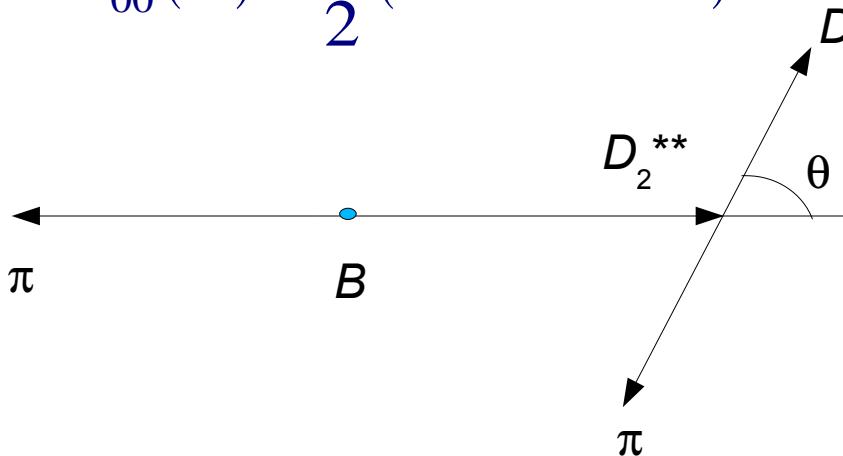
```
modelist.Register(new EvtVSS);
```

For each instance of a decay in the decay table that uses the VSS model  
a new instance of the EvtVSS class is created using the clone method.

# HELAMP and PARTWAVE models

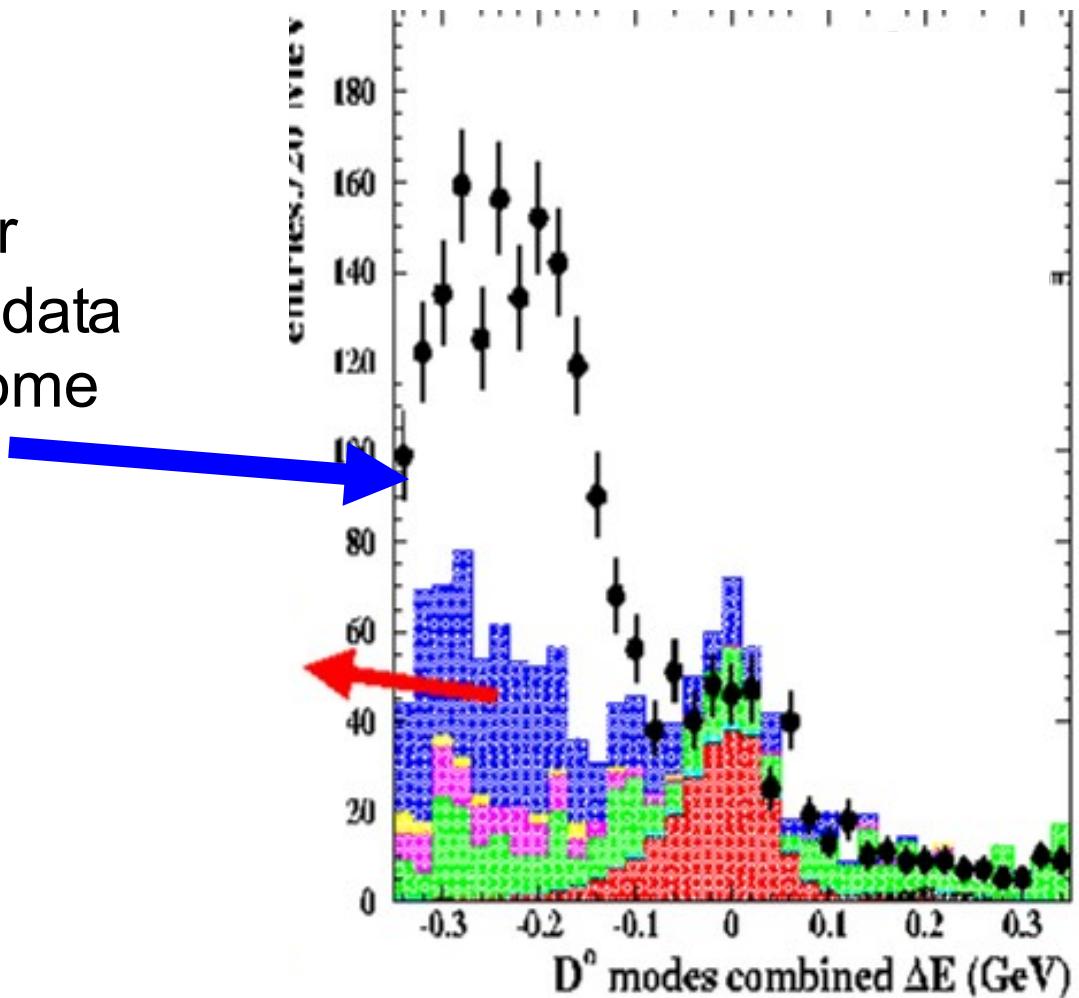
- $B \rightarrow D_2^{**} \pi \quad D_2^{**} \rightarrow D \pi$ 
  - Known and nontrivial kinematical distributions.
  - For decays with multiple allowed partial waves, amplitudes are specified as model argument

$$A = d_{00}^2(\theta) = \frac{1}{2}(3\cos^2 \theta - 1)$$



# Given large data sample, detailed effects must be modeled in generic $B$ Monte Carlo

Mixed up two decay amplitudes in  $B \rightarrow D^* \rho$  for generic MC led to large data vs MC differences for some analyses.

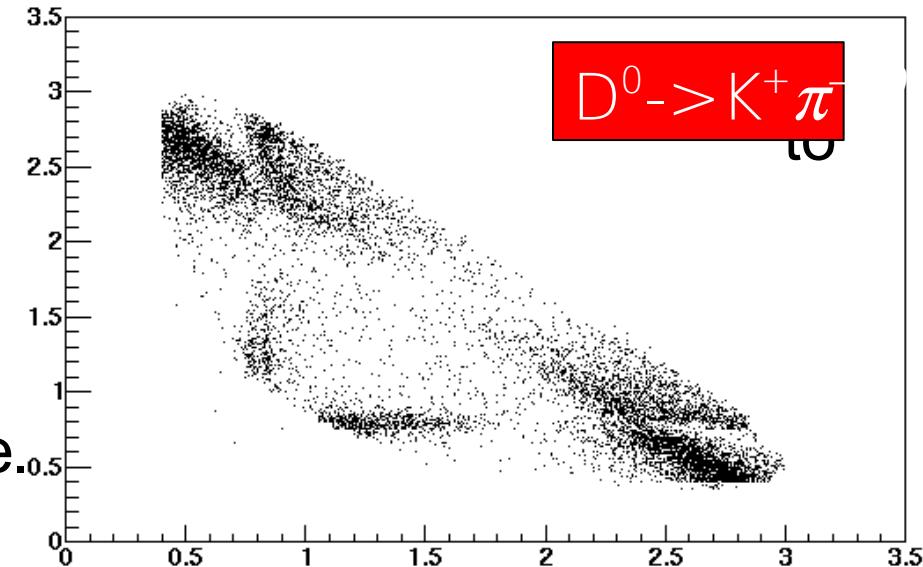


# Jetset 7.4 used for inclusive decay generation

- We rely on Jetset to handle  $ee \rightarrow qq$  fragmentation and  $B$  decays not specified in the decay table.
- $B$  decays:
  - Approximately 40% of the  $B$  decay width is not explicitly listed in decay table.
  - Pythia decays are accepted if generated mode is not specified in the decay table.
  - We have performed some tuning to improve the data vs. MC agreement
    - ↳ BF to charmless non-resonant states too big.
    - ↳  $D^*$  production in both  $B$  and  $ee \rightarrow cc$  decays

# Lineshapes and Dalitz plots

- Try to use relativistic Breit-Wigners for all particles with finite width.
  - ◆ Only for decays to two daughters
    - Otherwise non-rel BW.
  - ◆ Particles produced by Jetset have non-rel BW
- Include where possible
  - ◆ phase space factors, birth and decay form factors.
- Minimize use of mass cutoffs
  - ◆ Still needed in many cases prevent crashes due to pathological configurations.
- Moving towards integrated lineshape and Dalitz plot code.



# Writing a user decay file

```
#  
Alias myD0 D0  
Alias myanti-D0 anti-D0  
#  
Decay vpho  
0.500  myD0  anti-D0      VSS;  
0.500  D0    myanti-D0    VSS;  
Enddecay  
#  
Decay myD0  
1.000  eta   pi0          PHSP;  
Enddecay  
#  
Decay myanti-D0  
1.000  eta   pi0          PHSP;  
Enddecay  
#  
End
```

# LundAreaLaw

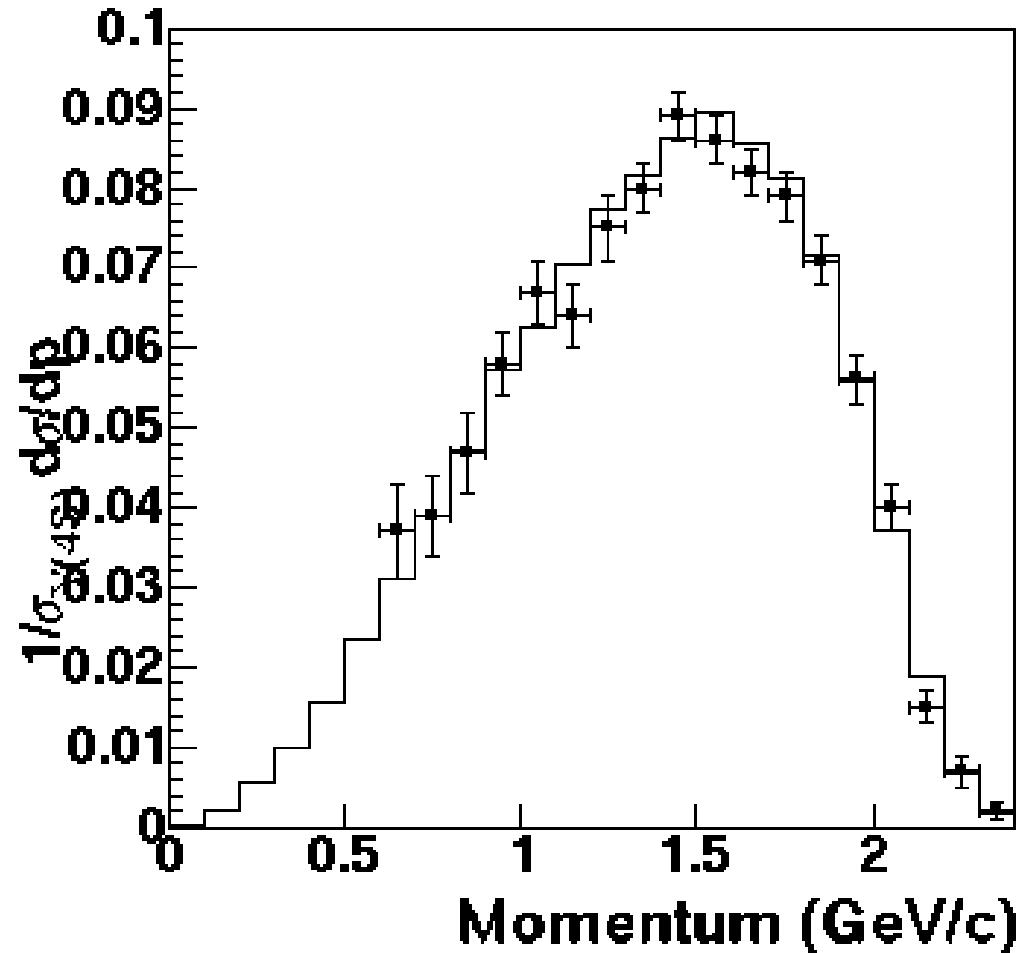
- Jim N. and I added the LundAreaLaw to EvtGen
  - The lund area law is a modified version of JetSet that should produce a more accurate fragmentation at low energy, in particular it should simulate baryon production better.
- To use the lund area law for the fragmentation in  $e^+e^-$ :

```
Decay vpho
1.000      LUNDAREALAW 0;
Enddecay
End
```

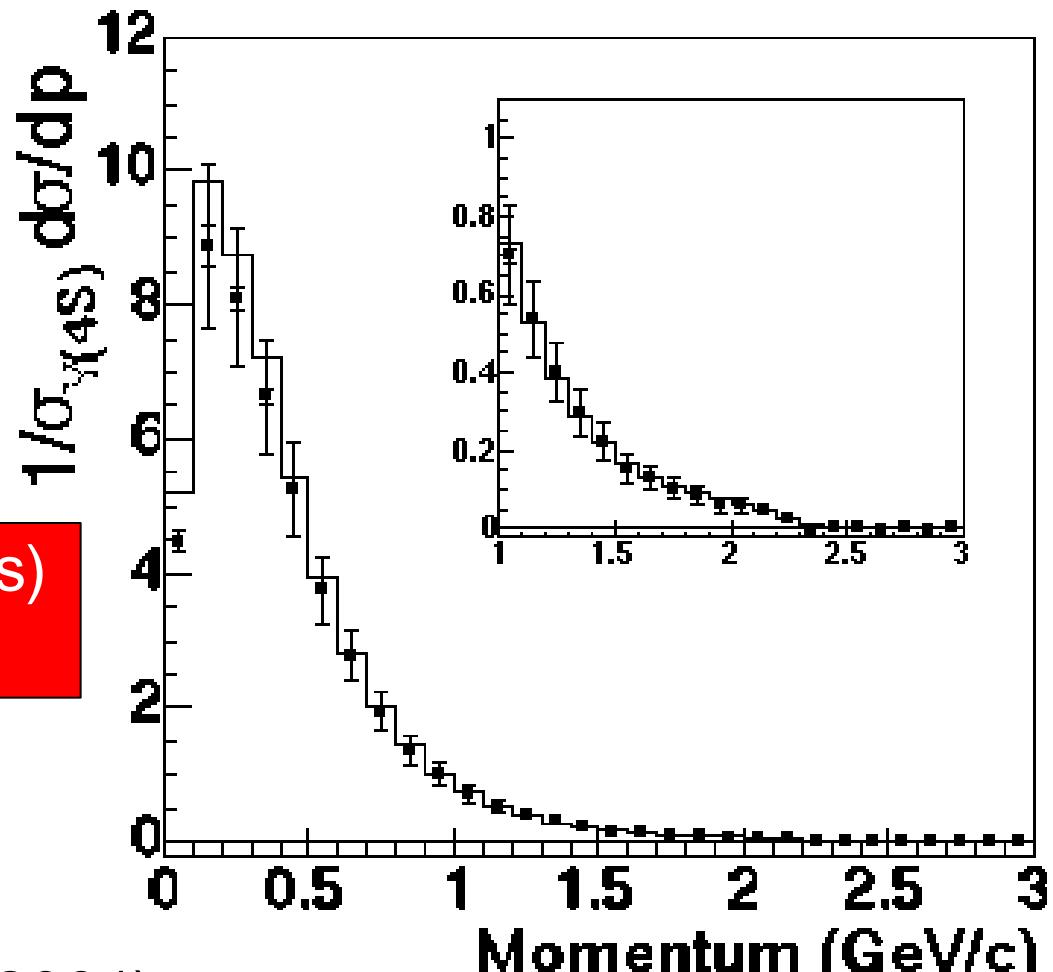
# $B \rightarrow X l \bar{\nu}$ lepton energy spectrum

- Lepton energy spectrum tuned using CLEO data.
  - PRL 76 1570 (1996)

Mode	BF (%)
$D^* l \bar{\nu}$	5.6
$D l \bar{\nu}$	2.1
$D_1^{**}(2420) l \bar{\nu}$	0.56
$D_0^{**} l \bar{\nu}$	0.2
$D_1^{***}(2460) l \bar{\nu}$	0.37
$D_2^{**}(2460) l \bar{\nu}$	0.37
$D^* \pi l \bar{\nu}$	0.3
$D \pi l \bar{\nu}$	0.9



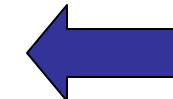
# $\pi^0$ momentum spectrum



- PRD 64, 072001, (2001)

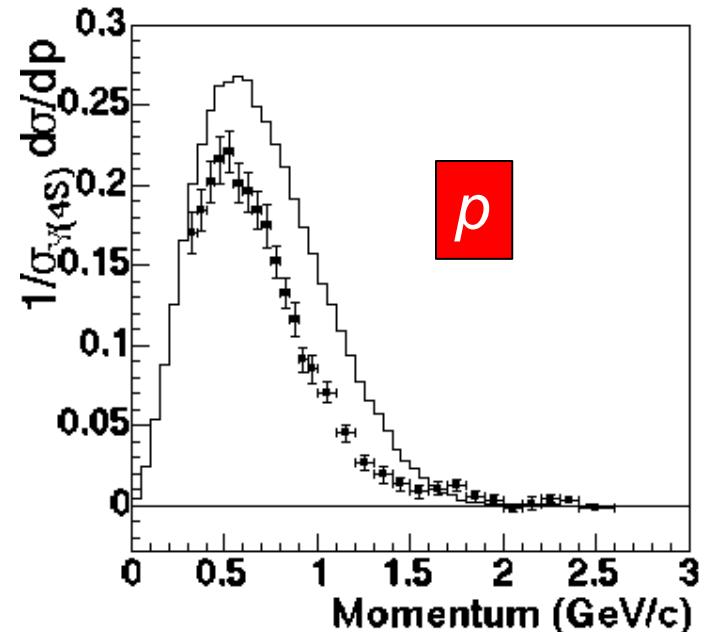
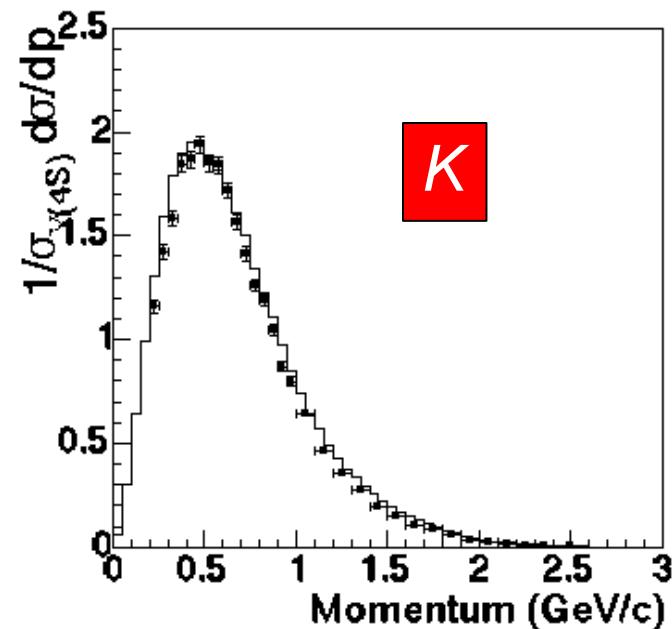
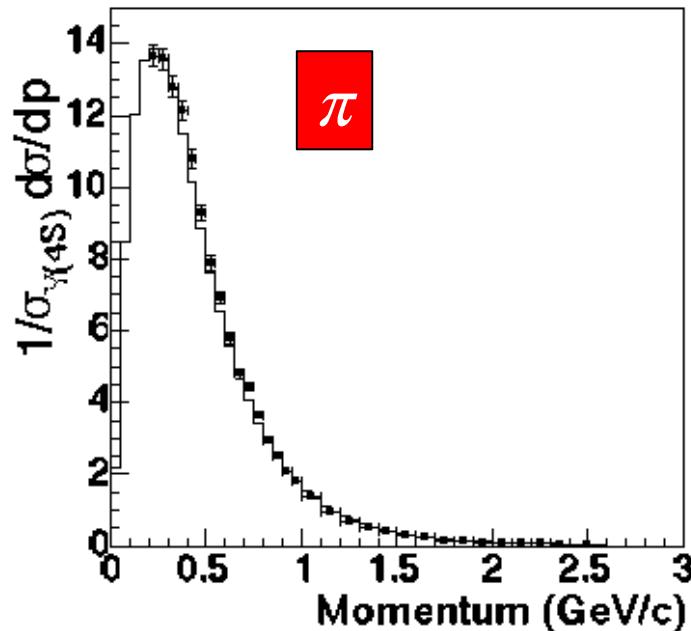
# Inclusive resonance production in B decays

	PDG03	EvtGen
$B \rightarrow X e \nu$	10.7 +/- 0.28	10.6
$B \rightarrow D^+ X$	24.5 +/- 2.1	32.4
$B \rightarrow D^0 X$	64.0 +/- 2.9	68.2
$B \rightarrow D^{*+} X$	22.5 +/- 1.5	26.2
$B \rightarrow D^{*0} X$	26.0 +/- 2.7	25.7
$B \rightarrow D^{(*)} D^{(*)} K$	7.1+2.7-1.7	7.7
$B \rightarrow J/\psi X$	1.090 +/- 0.035	1.04
$B \rightarrow K^{**-} X$	18 +/- 6	17.5
$B \rightarrow \eta X$	17.6 +/- 1.6	22.8
$B \rightarrow \Lambda_c X$	6.4 +/- 1.1	3.7
$B \rightarrow \Lambda X$	4.0 +/- 0.5	4.6
$B \rightarrow \phi X$	3.5 +/- 0.7	4.7



PDG  $B \rightarrow D^{(*)}$  production  
 Bfs not consistent with  
 isospin (and  $B(B \rightarrow X) = 1$ )  
 at several sigma level

# Preliminary $Y(4S) \rightarrow \pi/K/p$ spectra from BABAR



	BABAR	EvtGen
$\pi$	$7.73 \pm 0.32$	7.98
$K$	$1.54 \pm 0.04$	1.61
$p$	$0.155 \pm 0.004$	0.224

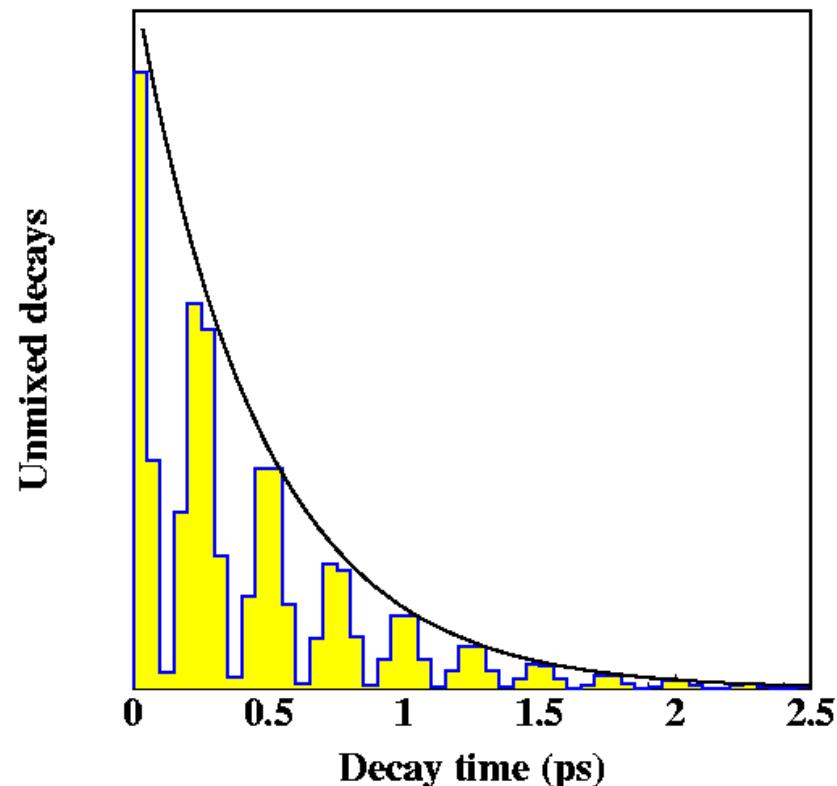
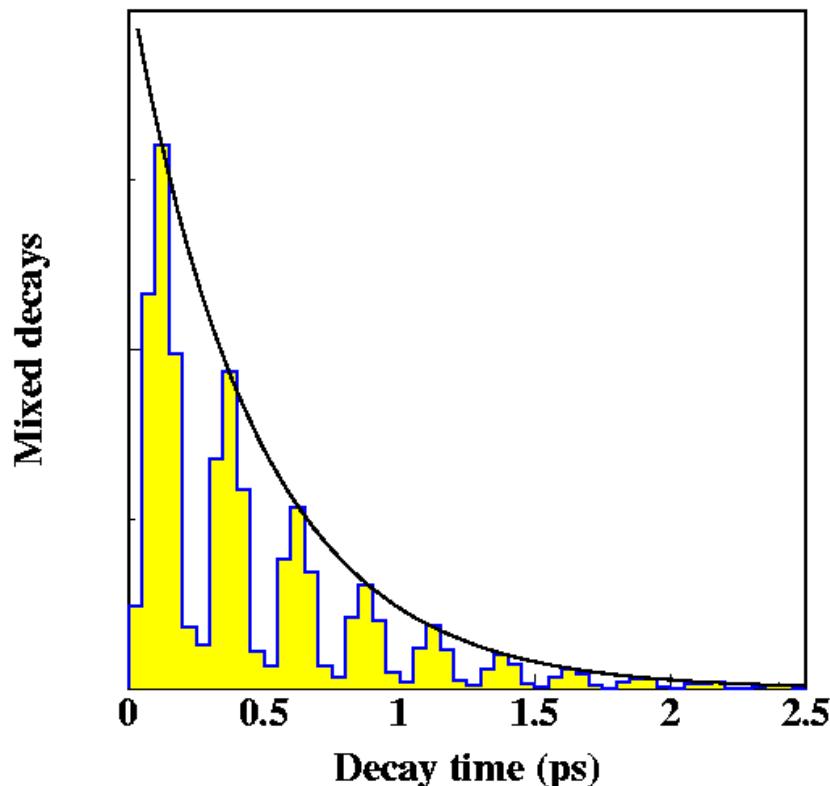
# Available decay modes (III)

- SSD\_CP model simulates CP violation for final states with a pseudoscalar + either a scalar, vector, or tensor.
  - $B \rightarrow \pi\pi$ ,  $B \rightarrow J/\psi K_S$ ,  $B \rightarrow D^*\pi$ , etc.
  - Specify in decay table:
    - $\Delta m$
    - $\Delta\Gamma/\Gamma$
    - $q/p$
    - $A(B \rightarrow f)$ ,  $A(B\bar{b} \rightarrow f)$ ,  $A(B \rightarrow f\bar{b})$ ,  $A(B\bar{b} \rightarrow f\bar{b})$
    - $z$
  - Flexible but relatively new model, so we are still gaining experience with all the possible use cases.

# $B_s$ physics in EvtGen

Items different wrt  $Y(4S) \rightarrow BB$  decays:

- Large # of common final states
- Incoherent mixing



Conclude about common final states

# Basic EvtGen interface (EvtGen.cc)

```
EvtGen myGenerator(  
    <DECAY.DEC location>,  
    <evt.pdl location>  
    <randomNumberEngine>.  
    <FSR generator>);
```

Optional: PHOTOS  
is default.

```
myGenerator.readUDecay(<user decay file>);
```

```
EvtParticle *myParentParticle;  
..... (Set up parent particle properties)....
```

```
myGenerator.generateEvent(myParentParticle,t_init);
```