PYDECAY/GRAPHPHYS

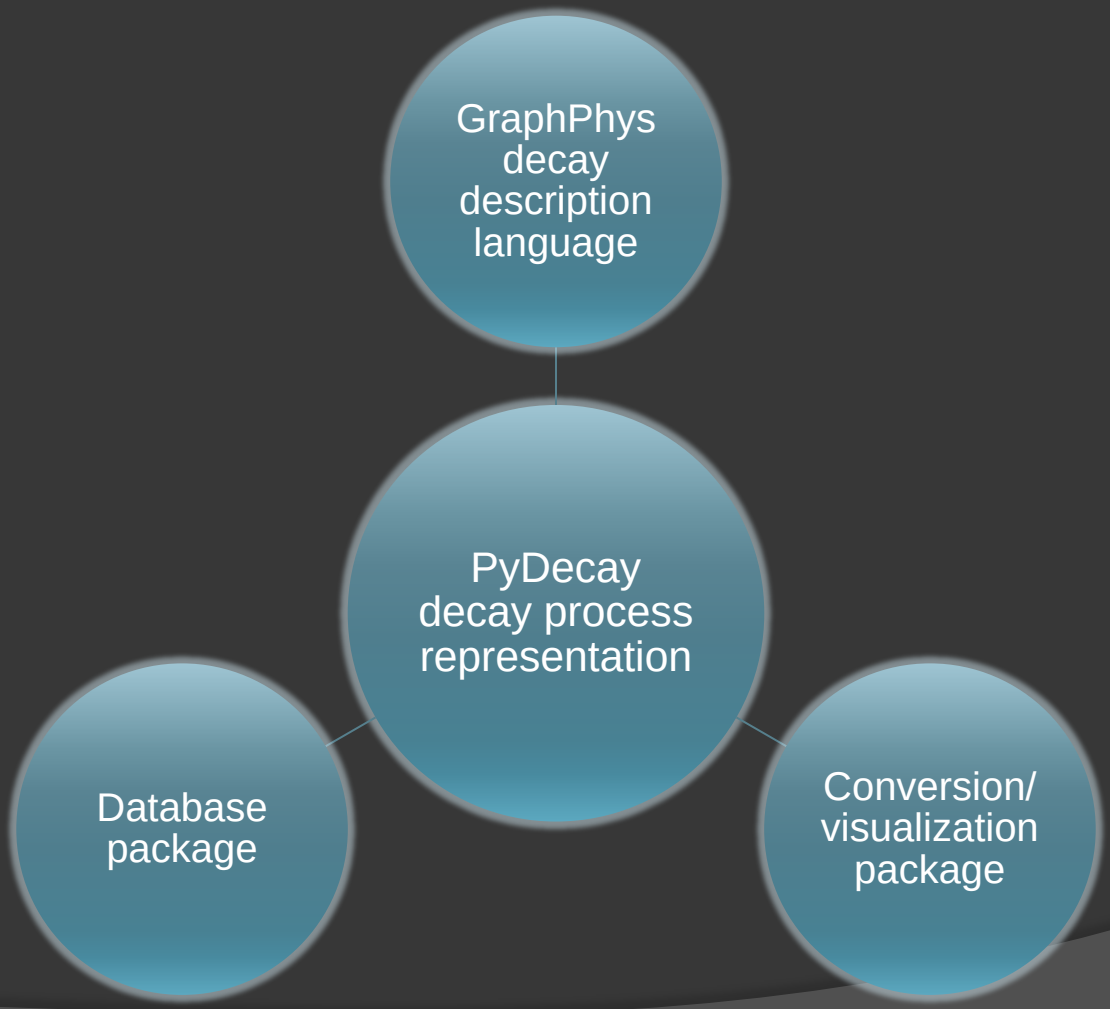
A Unified Language and Storage System
for Particle Decay Process Descriptions

Jesse Dunietz
BaBar Physics Meeting
August 17, 2010

Difficulties in HEP Analyses

- Formats don't match each other *or* the picture in analysts' heads
- BaBar-specific: tcl/.DEC formats not ideal
  - High learning curve
  - Won't work in classrooms
  - Difficult to share files
  - Not easily searchable
  - Hard to read
- Some tasks are impossible by computer
  - E.g. kinematics checking, branching fractions

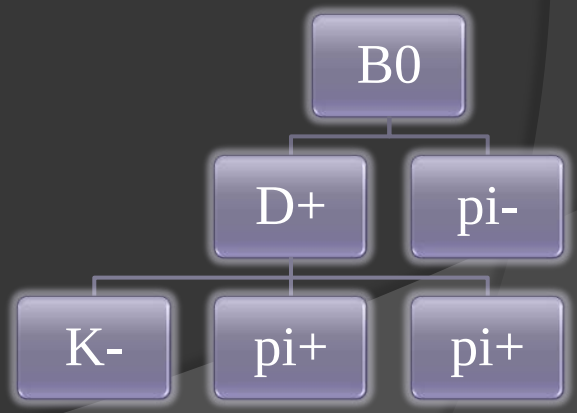
The Solution: A Decay Specification Framework



Example:

$B^0 \rightarrow D^+\pi^-$

$D^+ \rightarrow K^-\pi^+\pi^+$



The GraphPhys Language

- Inspired by GraphViz Dot
- Easy to read/write
- Arbitrary parameters on:
  - Particles
  - Decays
  - Entire processes
- Parameters can be nested
- File-wide defaults

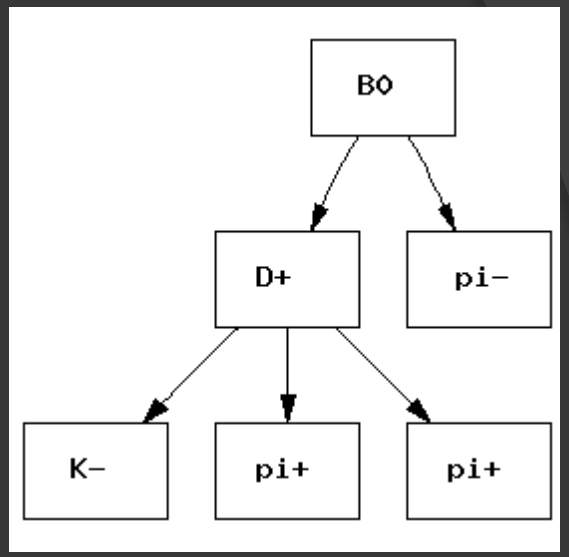
```
pip0 [type="pi+"];
pip1 [type="pi+"];

B0 -> {"D+" "pi-"};
"D+" -> {"K-" pip0 pip1};
```

Visualization

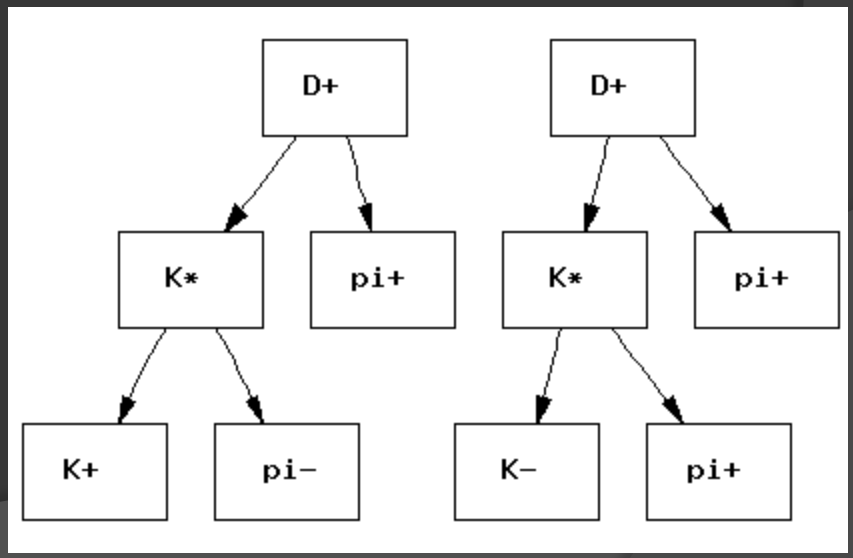
```
pip0 [type="pi+"];
pip1 [type="pi+"];

B0 -> {"D+" "pi-"};
"D+" -> {"K-" pip0 pip1};
```




```
pip0 [type="pi+"];
pip1 [type="pi+"];
pim1 [type="pi-"];
pim2 [type="pi-"];

"D+" -> {"K*" pip0};
"K*" -> {"K+" pim2};
"K*" -> {"K-" pip1};
```



Tools Built on this Framework

- Proof-of-concept MC simulator
- Kinematics checker
- Visualization generator
- tcl file generator

Generating a .tcl File

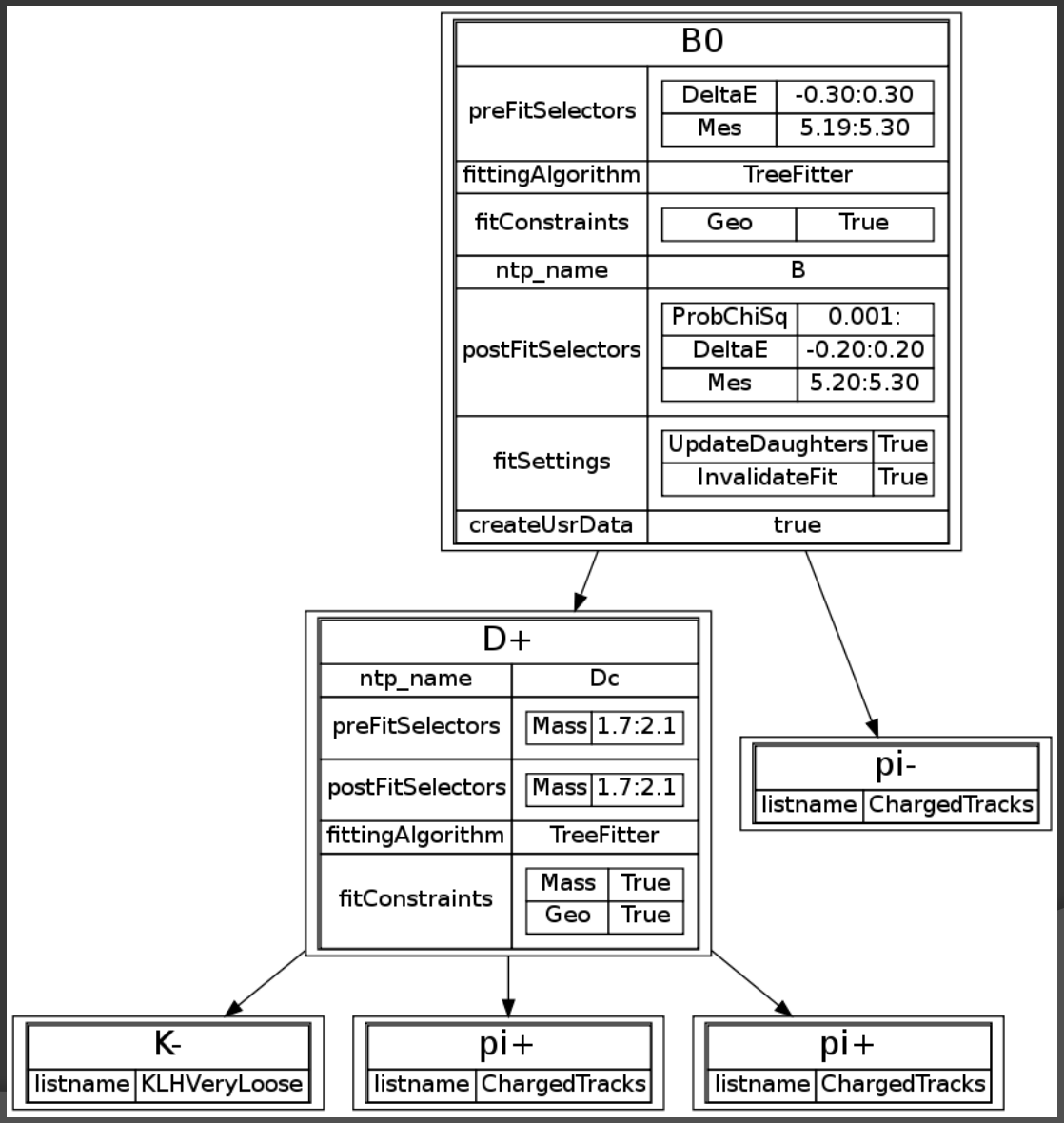
```
K    [type="K-",  listname="KLHVeryLoose"];
pip0 [type="pi+", listname="ChargedTracks"];
pip1 [type="pi+", listname="ChargedTracks"];
pim  [type="pi-", listname="ChargedTracks"];

D [type="D+", ntp_name="Dc",
     fittingAlgorithm="TreeFitter",
     fitConstraints=["Geo", "Mass"],
     preFitSelectors=[Mass="1.7:2.1"],
     postFitSelectors=[Mass="1.7:2.1"]];

B [type="B0", ntp_name="B",
   fittingAlgorithm="TreeFitter",
   fitConstraints=["Geo"],
   preFitSelectors=[DeltaE="-0.30:0.30", Mes="5.19:5.30"],
   postFitSelectors=[ProbChiSq="0.001:",
                     DeltaE="-0.20:0.20", Mes="5.20:5.30"],
   fitSettings=["InvalidateFit","UpdateDaughters"],
   createUsrData="true"];

B -> {D pim};
D -> {K pip0 pip1};
```

Visualizing our GraphPhys File



Converting to the tcl File

```
...

#  All the composition modules get added to this sequence
sequence create AnalysisSequence
path append Everything AnalysisSequence

mod clone SmpMakerDefiner My_Dc_to_Kcpicpic
seq append AnalysisSequence My_Dc_to_Kcpicpic
talkto My_Dc_to_Kcpicpic {
    decayMode             set "D+ -> K- pi+ pi+"
    daughterListNames     set "KLHVeryLoose"
    daughterListNames     set "ChargedTracks"
    daughterListNames     set "ChargedTracks"
    preFitSelectors       set "Mass 1.7:2.1"
}
mod clone SmpRefitterDefiner My_Dc_to_Kcpicpic_Constrained
seq append AnalysisSequence My_Dc_to_Kcpicpic_Constrained
talkto My_Dc_to_Kcpicpic_Constrained {
    unrefinedListName     set "My_Dc_to_Kcpicpic"
    preFitSelectors       set "Mass 1.7:2.1“

...
```

The Database System

- Previously examined decays (“instances”)
- PDG info (“types”)
- Fully searchable via Python
- Null/dictionary-based implementations available for lightweight jobs

Name	Mass	Charge	Width	…
pi-	139.57	-1	$2.6(10^{-8})$	…
D+	1869.6	+1	$1.04(10^{-12})$	…

ID #	Type	Product of
242	pi-	Instance #243
243	D+	NULL

Validating the tcl File Substitute

```
jessed@bbr-uci-1:~/Documents/particle_decay_syntax/tools$
    ./kinematics_check.py ~/Documents/test.dot
All decays specified are kinematically possible.
```

…or, if the check fails:

```
jessed@bbr-uci-1:~/Documents/particle_decay_syntax/tools$
    ./kinematics_check.py ~/Documents/test.dot
Impossible decay specified: B0 -> D+ B-
Impossible decay specified: D+ -> D- K+ pi+
```

Conversion

- Visitor design pattern: “converter” objects
- Converters can share code through inheritance
- Converters have interchangeable APIs
- Most converters use internal PyDecay objects as intermediate representation

Potential and Future Work

- Interface with existing projects' software
- Framework for future projects (LHC, SuperB…)
- Outreach for students
- Possible PDG interface
- Google Code repository
  - Development will continue, albeit at a slower pace