# *Full MC chain for diffractive processes: current status*
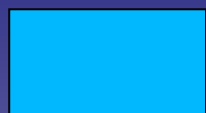
R. Ryutin (on behalf of K. Datsko and A.Sobol)

IHEP, Protvino

# Full MC chain for diffractive processes: current status
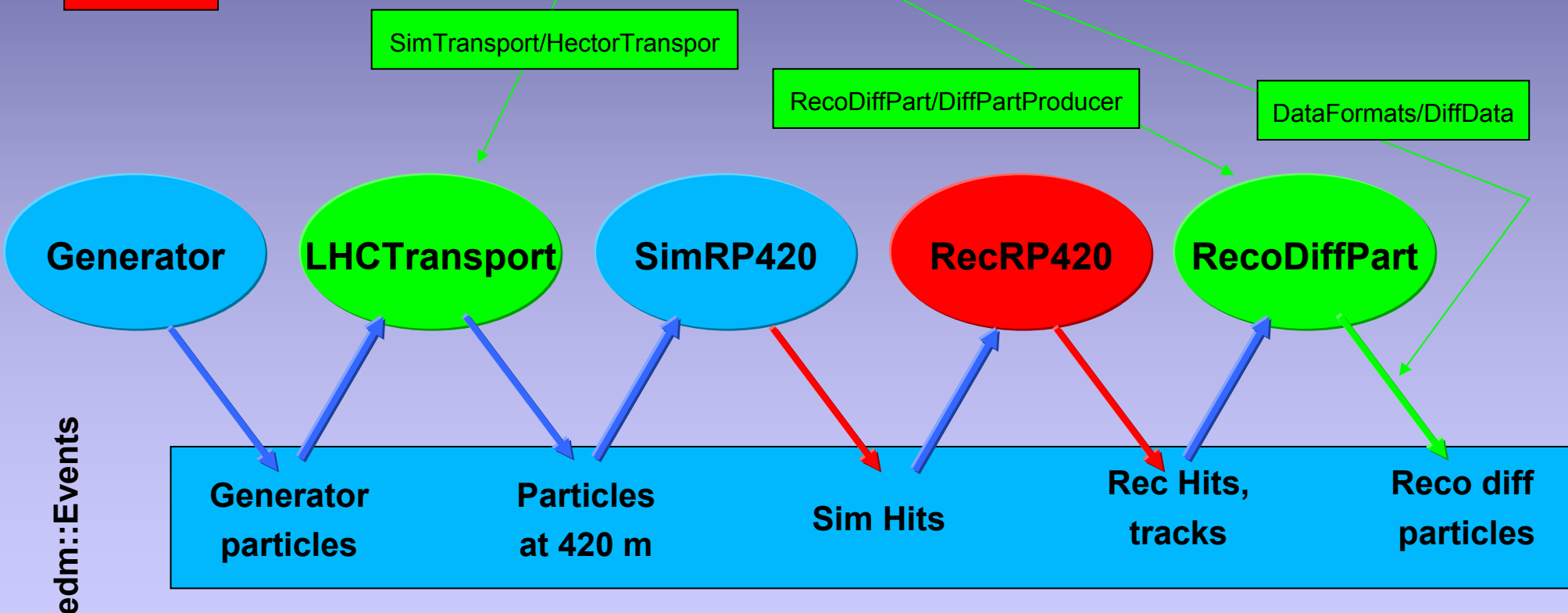


completely implemented in CVS

code exist, package request was made, no answer yet

no codes yet

SimTransport/HectorTranspor

RecoDiffPart/DiffPartProducer

DataFormats/DiffData

**Generator**

**LHCTransport**

**SimRP420**

**RecRP420**

**RecoDiffPart**

edm::Events

Generator particles

Particles at 420 m

Sim Hits

Rec Hits, tracks

Reco diff particles

*K.Datsko /A.Sobol*

# Transport integration status

How to use:

- for compilation type in *SimTransport/HectorProducer*  *scramv1 b*
- for configuring put in to your *cfg* file:

  - if you want to use CMSSW vertex smearing

    *include "SimTransport/HectorProducer/test/HectorProdVtxSmear.cfi"*

    *path p1 = { VtxSmeared, LHCTransport, g4SimHits }*

  - if you want to use HECTOR vertex smearing

    *include "SimTransport/HectorProducer/test/HectorProdSelfSmear.cfi"*

    *path p1 = { LHCTransport, g4SimHits }*

  - *replace g4SimHits.Generator.HepMCProductLabel = "LHCTransport"*

  - for changing some parameters from *HectorProdVtxSmear.cfi or HectorProdSelfSmear.cfi* add something like

    *replace LHCTransport.Hector.RP420f = 416.* (for changing RP position)

*K. Datsko*

# Data formats

LHCTransport saves data in to *edm::Event* like HepMCProduct format. Therefore we don't need design new data format.
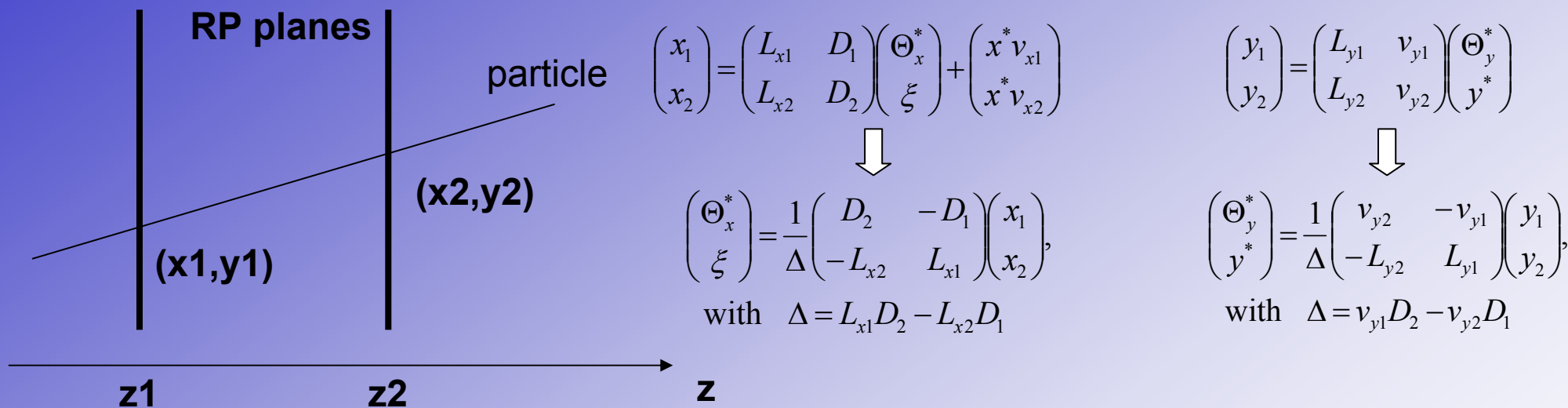
For RecoDiffPart module we need new data type. Therefore it has been designed new fomats of data DiffPartCollection (which is just *std::vector<DiffPart>*) and DiffPart. The DiffPart consist of:

- for momentum of diffractive particle at IP (*CLHEP::HepLorentzVector*)
- pointer (bar code) of mather particle in the initial MC event
- *possible something else?*

| Name | Title |
|---|---|
| @size | size of the collection |
| DiffParticles_RecoDiffPart__TEST.obj.ee | ee[DiffParticles_RecoDiffPart__TEST.obj_] |
| DiffParticles_RecoDiffPart__TEST.obj.mother_pointer | mother_pointer[DiffParticles_RecoDiffPart__TEST.obj_] |
| DiffParticles_RecoDiffPart__TEST.obj.pp.dx | dx[DiffParticles_RecoDiffPart__TEST.obj_] |
| DiffParticles_RecoDiffPart__TEST.obj.pp.dy | dy[DiffParticles_RecoDiffPart__TEST.obj_] |
| DiffParticles_RecoDiffPart__TEST.obj.pp.dz | dz[DiffParticles_RecoDiffPart__TEST.obj_] |

# DataProducer

We use Hector for reconstruction of diffractive particles at IP.

RP planes

particle

(x2,y2)

(x1,y1)

z1　　　z2　　　z

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} L_{x1} & D_1 \\ L_{x2} & D_2 \end{pmatrix}\begin{pmatrix} \Theta_x^* \\ \xi \end{pmatrix} + \begin{pmatrix} x^* v_{x1} \\ x^* v_{x2} \end{pmatrix}$$

$$\begin{pmatrix} \Theta_x^* \\ \xi \end{pmatrix} = \frac{1}{\Delta}\begin{pmatrix} D_2 & -D_1 \\ -L_{x2} & L_{x1} \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix},$$

with $\Delta = L_{x1}D_2 - L_{x2}D_1$

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} L_{y1} & v_{y1} \\ L_{y2} & v_{y2} \end{pmatrix}\begin{pmatrix} \Theta_y^* \\ y^* \end{pmatrix}$$

$$\begin{pmatrix} \Theta_y^* \\ y^* \end{pmatrix} = \frac{1}{\Delta}\begin{pmatrix} v_{y2} & -v_{y1} \\ -L_{y2} & L_{y1} \end{pmatrix}\begin{pmatrix} y_1 \\ y_2 \end{pmatrix},$$

with $\Delta = v_{y1}D_2 - v_{y2}D_1$

Hector reconstructs parameters of diffractive particles at IP by measurements at 2 planes. First one at the enter in RomanPot, second one at the exit. Since we can get some tracks (Reco/Sim Hits) from Sasha's Sim/RecoRP420 we can reconstruct kinematics at IP and store this information (*DiffParticleCollection*) in to *edm::Event* (ROOT file) for future using in physical analysis.

*K. Datsko*