中国科学院高能物理研究所
Institute of High Energy Physics
Chinese Academy of Sciences

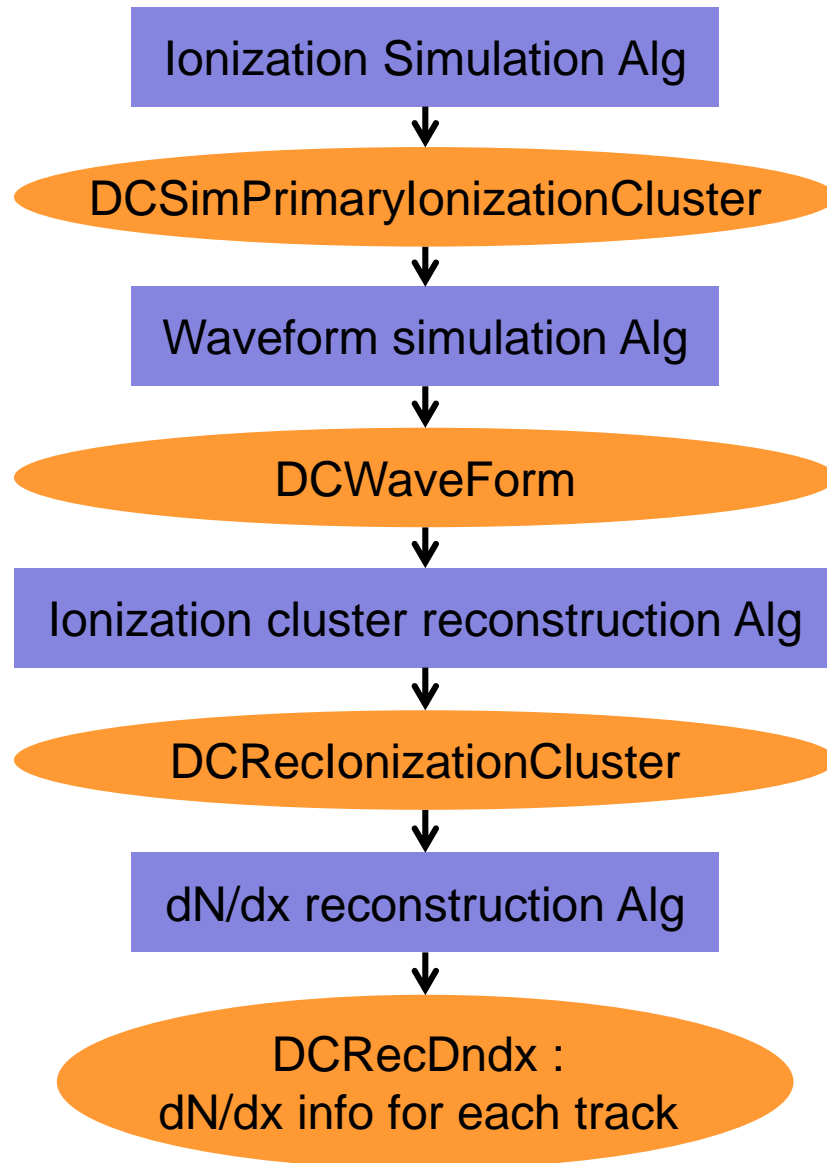# EDM for dN/dX study within the CEPCSW

Wenxing Fang (IHEP)

Cluster counting meeting (2022.09.15)

Introduction

❖ As the dN/dx method has great potential for PID, studying dN/dx using full simulation of CEPC detector should be supported

❖ Try to develop the chain of dN/dx study based on CEPCSW

❖ CEPCSW is fully integrated with the key4hep, and the edm4hep is used for the event data model

❖ Currently, edm4hep does not include EDM for drift chamber study

❖ Try to develop a common EDM for the drift chamber based on PODIO
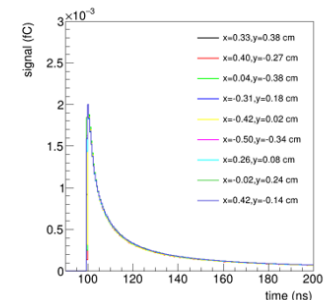
Chain of dN/dx study
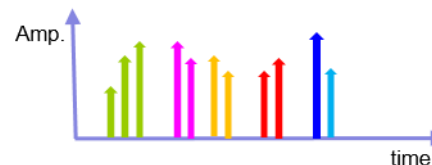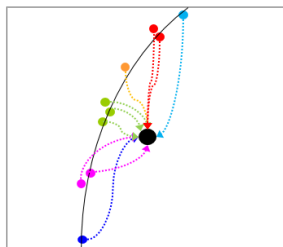
Ionization Simulation Alg

↓

DCSimPrimaryIonizationCluster

↓

Waveform simulation Alg

↓

DCWaveForm

↓

Ionization cluster reconstruction Alg

↓

DCRecIonizationCluster

↓

dN/dx reconstruction Alg

↓

DCRecDndx :
dN/dx info for each track

Ionization simulation
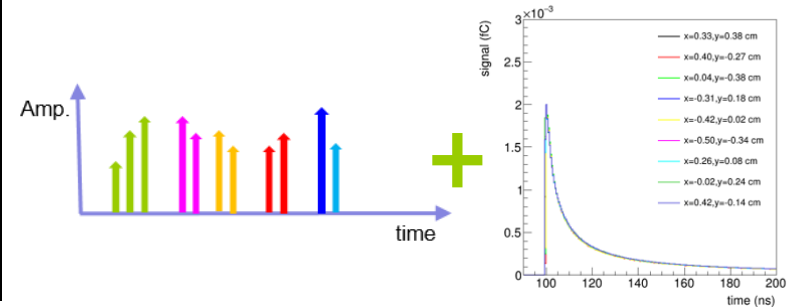
Geant4+TrackHeed+pulse_simulation(NN)

```
#----------- DCSimPrimaryIonizationCluster
edm4dc::DCSimPrimaryIonizationCluster:
  Description: "Simulated Primary Ionization"
  Author : "Wenxing Fang, IHEP"
  Members:
    - unsigned long long cellID      //ID of cell for this produced primary ionization.
    - float time                     //the primary ionization producing time in the lab frame in [ns].
    - int   type                     //type.
    - edm4hep::Vector3d position     //the produced primary ionization's position in [mm].
  VectorMembers:
    - unsigned long long ionCellID   //ID of cell for this produced ionization.
    - float ionTime                  //the ionization producing time in the lab frame in [ns].
    - edm4hep::Vector3d ionPosition  //the ionization's position in [mm].
    - float pulseTime                //the pulse producing time in the lab frame in [ns].
    - float pulseAmplitude           //the pulse's amplitude.
  OneToOneRelations:
    - edm4hep::MCParticle MCParticle //MCParticle that caused the hit.
```

Waveform simulation


```
#---------- DCSimPrimaryIonizationCluster
edm4dc::DCSimPrimaryIonizationCluster:
  Description: "Simulated Primary Ionization"
  Author : "Wenxing Fang, IHEP"
  Members:
    - unsigned long long cellID      //ID of cell for this produced primary ionization.
    - float time                     //the primary ionization producing time in the lab frame in [ns].
    - int    type                    //type.
    - edm4hep::Vector3d position     //the produced primary ionization's position in [mm].
  VectorMembers:
    - unsigned long long ionCellID   //ID of cell for this produced ionization.
    - float ionTime                  //the ionization producing time in the lab frame in [ns].
    - edm4hep::Vector3d ionPosition  //the ionization's position in [mm].
    - float pulseTime                //the pulse producing time in the lab frame in [ns].
    - float pulseAmplitude           //the pulse's amplitude.
  OneToOneRelations:
    - edm4hep::MCParticle MCParticle //MCParticle that caused the hit.
```
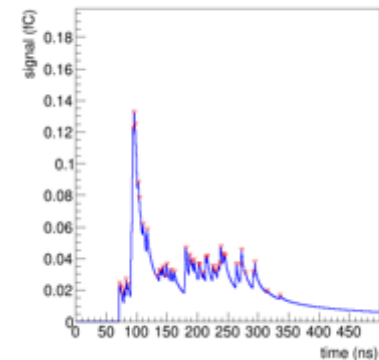
Wavefrom simulation Alg

```
#----------   DCWaveform
edm4dc::DCWaveform:
  Description: "Waveform"
  Author : "Wenxing Fang, IHEP"
  Members:
    - unsigned long long cellID  //detector specific cell id.
    - int type                   //type.
    - float beginTime            //begin time of the waveform.
    - float interval             //interval of each sampling in [ns].
  VectorMembers:
    - float rawData              //charges.
```

Ionization cluster reconstruction


```
#---------   DCWaveform
edm4dc::DCWaveform:
  Description: "Waveform"
  Author : "Wenxing Fang, IHEP"
  Members:
    - unsigned long long cellID  //detector specific cell id.
    - int type                   //type.
    - float beginTime            //begin time of the waveform.
    - float interval             //interval of each sampling in [ns].
  VectorMembers:
    - float rawData              //charges.
```
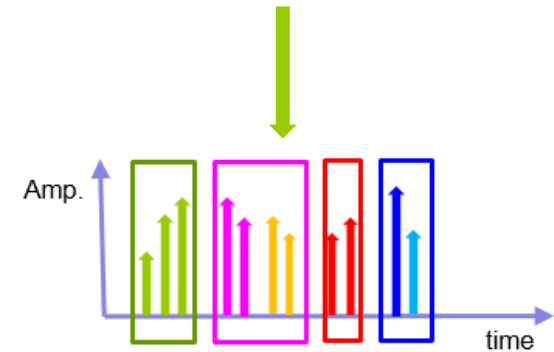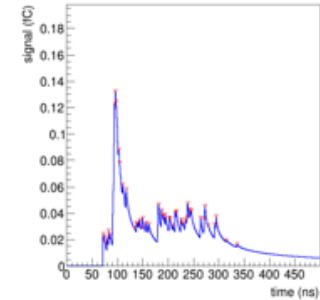
Ionization cluster reconstruction Alg

```
#---------   DCRecIonizationCluster
edm4dc::DCRecIonizationCluster:
  Description: "Reconstructed Ionization Cluster"
  Author : "Wenxing Fang, IHEP"
  Members:
    - unsigned long long cellID  //ID of cell for this reconstructed primary ionization cluster.
    - int type                   //type.
    - float significance         //significance of the reconstructed primary ionization cluster.
  VectorMembers:
    - float pulseTime            //the pulse producing time in the lab frame in [ns].
    - float pulseAmplitude       //the pulse's amplitude.
    - float pulseSignificance    //significance of the pulse.
```
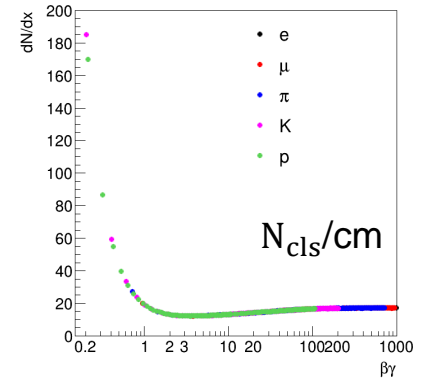
dN/dx reconstruction

rec ionization cluster collection

dN/dx reconstruction Alg

rec track collection

N_{cls}/cm


```
#-------- Track
edm4hep::Track:
  Description: "Reconstructed track"
  Author : "F.Gaede, DESY"
  Members:
    - int32_t type                    //flagword that defines the type of t
    - float chi2                      //Chi^2 of the track fit
    - int32_t ndf                     //number of degrees of freedom of the
    - float dEdx                      //dEdx of the track.
    - float dEdxError                 //error of dEdx.
    - float radiusOfInnermostHit      //radius of the innermost hit that has be
  VectorMembers:
    - int32_t subDetectorHitNumbers   //number of hits in particular subdet
    - edm4hep::TrackState trackStates //track states
    - edm4hep::Quantity dxQuantities // different measurements of dx quantities
  OneToManyRelations:
    - edm4hep::TrackerHit trackerHits //hits that have been used to create this
    - edm4hep::Track tracks           //tracks (segments) that have been combin

# quantity with an identifier, a value and an error
edm4hep::Quantity:
  Members:
    - int16_t type // flag identifying how to interpret the quantity
    - float value  // value of the quantity
    - float error  // error on the value of the quantity
```

```
#------------- DCRecDndx
edm4dc::DCRecDndx:
  Description : "DC dN/dx info of Track."
  Author : "Wenxing Fang, IHEP"
  Members :
    - int type                    //type.
    - float dNdx                  //reconstructed dNdx.
    - float dNdxError             //error on reconstructed dNdx.
    - int particleType            //particle type, e(0),mu(1),pi(2),K(3),p(4).
  VectorMembers:
    - unsigned long long  cellID   //DC cell id
    - float  N                    //number of reconstructed primary ionization in DC cell
    - float  edep                 //energy deposite in DC cell
    - float  pathL                //path length in DC cell in [mm]
    - float  chi                  //chi for e(0), mu(1), pi(2), K(3), p(4)
    - float  dNdxExpect           //expected dNdx for e(0),mu(1), pi(2), K(3), p(4)
    - float  dNdxSigma            //expected sigma of dNdx for e(0),mu(1), pi(2), K(3), p(4)
  OneToOneRelations:
    - edm4hep::Track track        //track that created the DC info.
```

https://github.com/wenxingfang/CEPCSW/blob/master/Edm/edm4dc.yaml

7

TPC

EUDET-Report-2007-04   MarlinTPC

Data structure	Processor name	input/output collection name
TrackerRawData		TPCRawData
	TrackerRawDataToDataConverterProcessor	
TrackerData		TPCConvertedRawData
	PedestalSubtractorProcessor	
	TimeShiftCorrectorProcessor	
TrackerData		TPCData
	PulseFinderProcessor	
	ChannelMapperProcessor	
	CountsToPrimaryElectronsConverterProcessor	
TrackerPulse		TPCPulses
	HitTrackFinderTopoProcessor	
TrackerHit, Track		TPCHits, TPCTrackCandidates
	TrackSeederProcessor	
Track		TPCSeedTracks
	TrackFitterLikelihoodProcessor	
Track		TPCTracks

Table 1: Present MarlinTPC reconstruction processors

Public Member Functions

	TrackerRawDataImpl ()	Default Constructor - initializes all data to 0's.
virtual	**~TrackerRawDataImpl** ()	Destructor.
virtual int	**id** () const	Returns an object id for internal (debugging) use in LCIO.
virtual int	**getCellID0** () const	Returns the first detector specific (geometrical) cell id.
virtual int	**getCellID1** () const	Returns the second detector specific (geometrical) cell id.
virtual int	**getTime** () const	Returns the time.
virtual const EVENT::ShortVec &	**getADCValues** () const	The measured ADC values.
void	**setCellID0** (int cellID0)	
void	**setCellID1** (int cellID1)	
void	**setTime** (int time)	
void	**setADCValues** (const EVENT::ShortVec &adc)	Set the ADC vector by copying the values.
EVENT::ShortVec &	**adcValues** ()	Allows direct access to the adc vector.

Protected Attributes

int	_cellID0
int	_cellID1
int	_channelID
int	_time

Public Member Functions

	TrackerDataImpl ()	Default Constructor - initializes all data to 0's.
virtual	**~TrackerDataImpl** ()	Destructor.
virtual int	**id** () const	Returns an object id for internal (debugging) use in LCIO.
virtual int	**getCellID0** () const	Returns the first detector specific (geometrical) cell id.
virtual int	**getCellID1** () const	Returns the second detector specific (geometrical) cell id.
virtual float	**getTime** () const	Returns the time.
virtual const EVENT::FloatVec &	**getChargeValues** () const	The calibrated ADC values.
void	**setCellID0** (int cellID0)	
void	**setCellID1** (int cellID1)	
void	**setTime** (float time)	
void	**setChargeValues** (const EVENT::FloatVec &charge)	Set the charge vector by copying the values.
EVENT::FloatVec &	**chargeValues** ()	Allows direct access to the charge vector.

Public Member Functions

	TrackerPulseImpl ()	Default Constructor - initializes all data to 0's.
	TrackerPulseImpl (const **TrackerPulseImpl** &)	default copy constructor - use with care
TrackerPulseImpl &	**operator=** (const **TrackerPulseImpl** &)	default assignment operator - use with care
virtual	**~TrackerPulseImpl** ()	Destructor.
virtual int	**id** () const	Returns an object id for internal (debugging) use in LCIO.
virtual int	**getCellID0** () const	Returns the first detector specific (geometrical) cell id.
virtual int	**getCellID1** () const	Returns the second detector specific (geometrical) cell id.
virtual float	**getTime** () const	The time of the pulse.
virtual float	**getCharge** () const	The integrated charge of the pulse // FIXME: unit ?.
virtual const EVENT::FloatVec &	**getCovMatrix** () const	Covariance matrix of the charge (c) and time (t) measurements.
virtual int	**getQuality** () const	The quality bit flag of the pulse - use the defined constants for referring to the bits.
virtual EVENT::TrackerData *	**getTrackerData** () const	Optionally the TrackerData that has been uesed to create the pulse can be stored with the pulse - NULL if none.
void	**setCellID0** (int cellID0)	
void	**setCellID1** (int cellID1)	
void	**setTime** (float time)	
void	**setCharge** (float charge)	
void	**setCovMatrix** (const float *cov)	
void	**setCovMatrix** (const EVENT::FloatVec &)	
void	**setQuality** (int quality)	
void	**setQualityBit** (int bit, bool val=true)	
void	**setTrackerData** (EVENT::TrackerData *corrData)	

Protected Attributes

int	_cellID0
int	_cellID1
float	_time
float	_charge
int	_quality
EVENT::FloatVec	_cov
EVENT::TrackerData *	_corrData

```
#---------- TPCHit

  edm4hep::TPCHit:

    Description: "Time Projection Chamber Hit"
    Author : "F.Gaede, DESY"
    Members:
      - uint64_t cellID   //detector specific cell id.
      - int32_t quality                //quality flag for the hit.
      - float time              //time of the hit.
      - float charge            //integrated charge of the hit.
    VectorMembers:
      - int32_t rawDataWords          //raw data (32-bit) word at i.
```
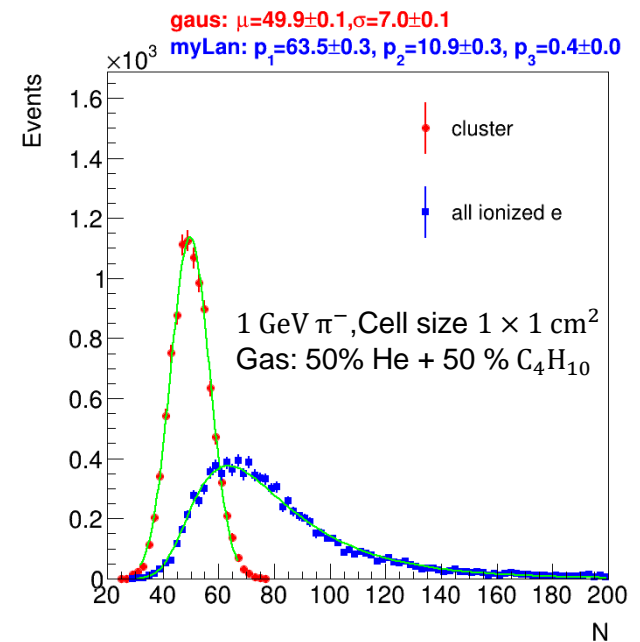
https://ilcsoft.desy.de/LCIO/current/doc/doxygen_api/html/classIMPL_1_1TrackerPulseImpl.html

Back up

Motivation

❑ The particle identification is very important for CEPC flavor physics study. Good hadron separation up to 20 GeV is essential

❑ Traditionally: using dE/dx method

  ❑ Due to the production of delta electron, the deposited energy follows Landau distribution

  ❑ Resolution is ~6%

❑ New technique: using dN/dx (cluster counting) method

  ❑ The number of primary ionization follows Poisson distribution

  ❑ Resolution could reaches <3%

❑ The dN/dx technique will be widely explored in CEPC drift chamber detector

gaus: μ=49.9±0.1,σ=7.0±0.1
myLan: p_1=63.5±0.3, p_2=10.9±0.3, p_3=0.4±0.0

cluster

all ionized e

1 GeV π^-, Cell size 1×1 cm^2
Gas: 50% He + 50 % C_4H_{10}

User extension data in EDM4hep

❖ As there is no waveform data format in EDM4hep yet, user extension data is a way to add additional data.

- WIP: https://github.com/key4hep/EDM4hep/pull/117 | Tao Lin

The proposed underlying data structure:

```
edm4hep::UserExt:
    Description: "A simple struct with user defined int/float/double"
    Author : "Tao Lin"
    VectorMembers:
        - int valI // data int
        - float valF // data float
        - double valD // data double
```

The proposed user APIs:

```
ud xyzi;
xyzi.reg("x", 1, 0)
    .reg("y", 1, 1)
    .reg("z", 1, 2)
    .reg("t", 2, 0)
    .reg("i", 0, 0);
```

```
xyzi.from(usrexts[i], 0)
    .get("x", x)
    .get("y", y)
    .get("z", z)
    .get("t", t)
    .get("i", iii);
```

```
xyzi.put("x", x)
    .put("y", y)
    .put("z", z)
    .put("t", t)
    .put("i", i);
```
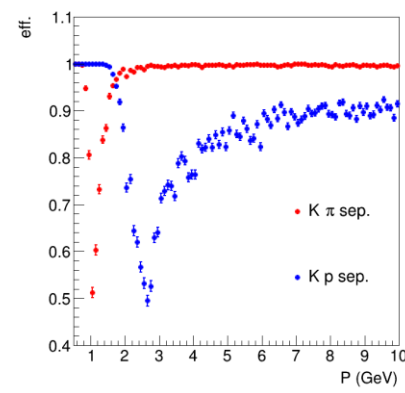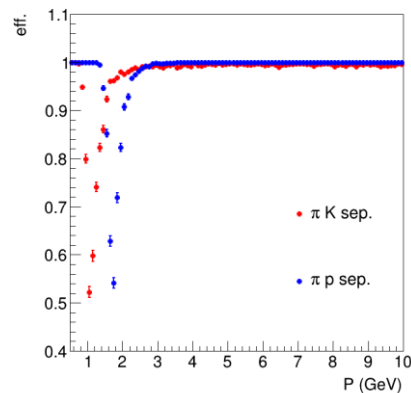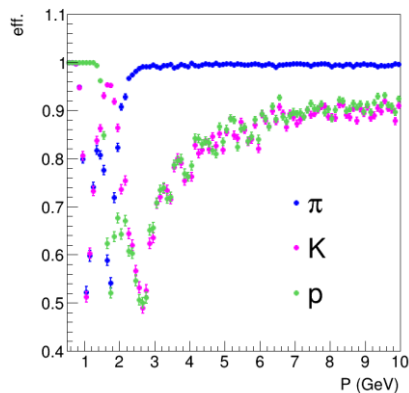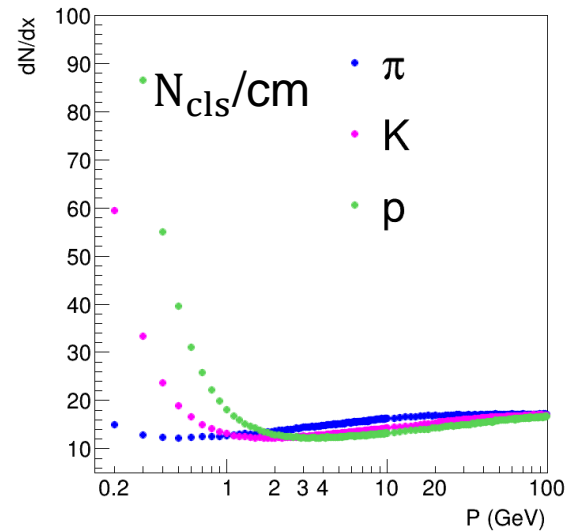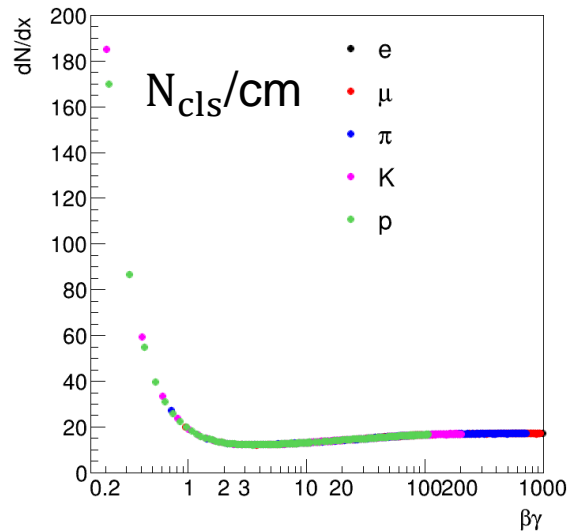
```
auto udv = usrexts.create();

xyzi.to(udv);
```

Runtime Type definition   Getters   Setters

Garfield++ simulation

90%He+10%C_4H_{10}

1 meter length