



# Command line tools for Podio

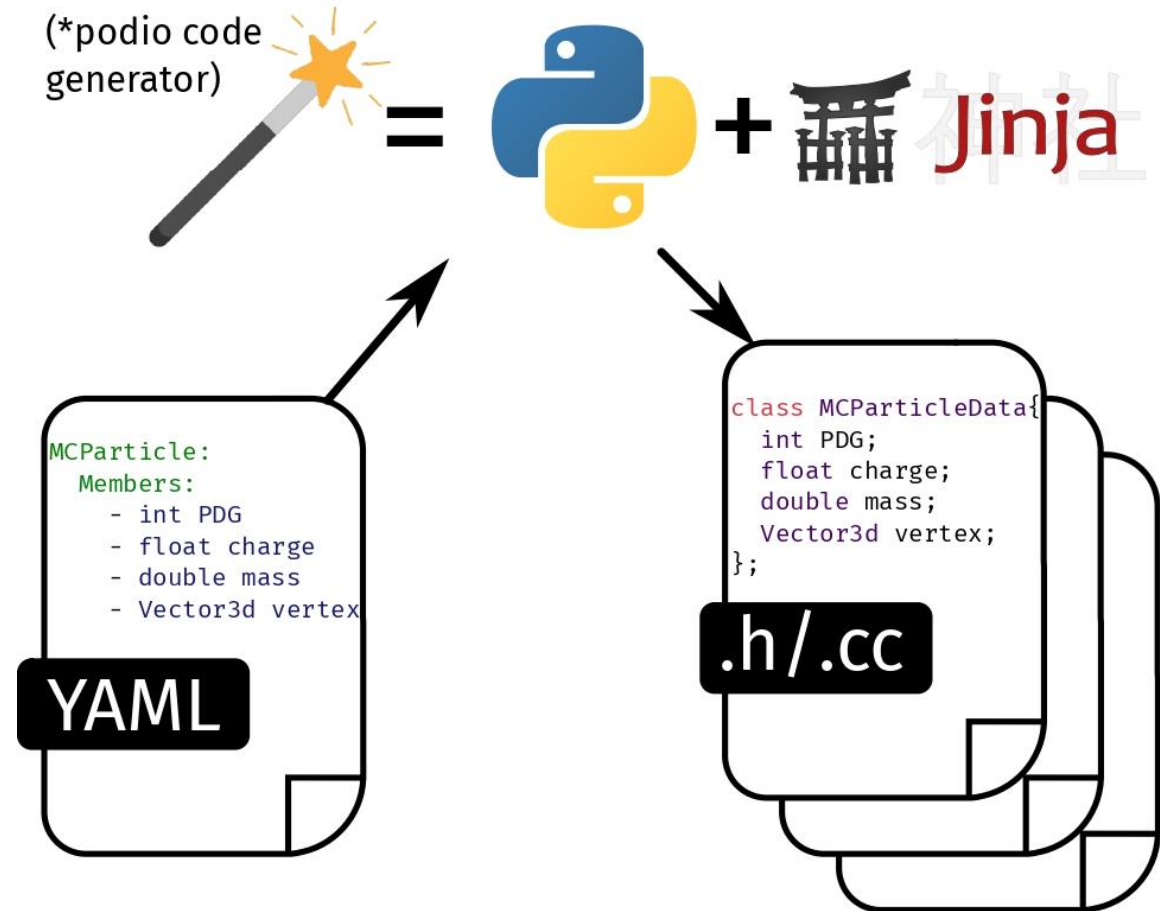
Juan Manuel Moreno Pérez  
for the Key4Hep meeting

Sep 7, 2021

---

# Podio

- Podio is a EDM tool that generates all code from a YAML description.
- Podio is used for EDM4hep in Key4hep
- Podio can be used to generate arbitrary datamodels
- Podio supports different I/O backend



# LCIO and iLC software

- LCIO is the event datamodel and persistency layer used by the iLC software.
- Development started 15 years ago and there are lots of tools around this.
- Two of these tools are **anajob** and **dumpevent** which allow you to inspect LCIO files from command line.
- These tools does not yet exist on podio.
- My work is to make a first version of these tools.

# Current anajob on iLC software vs Podio Event Overview

```
////////////////////
EVENT: 2
RUN: 0
DETECTOR: ILD_l5_v02
COLLECTIONS: (see below)
////////////////////
```

COLLECTION NAME	COLLECTION TYPE	NUMBER OF ELEMENTS
BuildUpVertex	Vertex	2
BuildUpVertex_RP	ReconstructedParticle	2
BuildUpVertex_V0	Vertex	1
BuildUpVertex_V0_RP	ReconstructedParticle	1
ClusterMCTruthLink	LCRelation	318
DistilledPFOS	ReconstructedParticle	132
GammaGammaCandidateEtaPrimes	ReconstructedParticle	3
GammaGammaCandidateEtas	ReconstructedParticle	12
GammaGammaCandidatePi0s	ReconstructedParticle	11
GammaGammaParticles	ReconstructedParticle	2
KinkRecoParticles	ReconstructedParticle	1
KinkVertices	Vertex	1
MCParticlesSkimmed	MCParticle	415
MCTruthClusterLink	LCRelation	318
MCTruthMarlinTrkTracksLink	LCRelation	95
MCTruthRecoLink	LCRelation	337
MarlinTrkTracks	Track	86
MarlinTrkTracksKaon	Track	86
MarlinTrkTracksMCTruthLink	LCRelation	95
MarlinTrkTracksProton	Track	86
PandoraClusters	Cluster	127
PandoraPFOS	ReconstructedParticle	134
PrimaryVertex	Vertex	1
PrimaryVertex_RP	ReconstructedParticle	1
RecoMCTruthLink	LCRelation	337
SplitRecoParticles	ReconstructedParticle	1
SplitVertices	Vertex	1
V0RecoParticles	ReconstructedParticle	1
V0Vertices	Vertex	1

```
FileName: example.root
Number of events: 2000
```

Event Number 0	Type	Collection Size
info	EventInfo	1
mcparticles	ExampleMC	10
moreMCs	ExampleMC	10
mcParticleRefs	ExampleMC	0
hits	ExampleHit	2
clusters	ExampleCluster	3
refs	ExampleReferencingType	2
refs2	ExampleReferencingType	1
Component	ExampleWithComponent	1
OneRelation	ExampleWithOneRelation	2
WithVectorMember	ExampleWithVectorMember	2
WithNamespaceMember	ex42::ExampleWithNamespace	25
WithNamespaceRelation	ex42::ExampleWithARelation	5
WithNamespaceRelationCopy	ex42::ExampleWithARelation	5
strings	ExampleWithString	1
arrays	ExampleWithArray	1
fixedWidthInts	ExampleWithFixedWidthIntegers	3

Is actually pretty good right now

# Summary

- Podio is the EDM tool for the key4hep project and it supports multiple I/O backends.
- An equivalent to **anajob** has been developed, it required a few changes to the code itself.
- Both the current and the old code had to be working.
- An equivalent to dumpevent started to be developed but faced some issues due to the nature of podio.
- This challenges will still be worked upon.
- <https://github.com/AIDASoft/podio/pull/212>

# Appendix

# Podio Event Overview

```
37     if(argc==4){
38         try{
39             startEvent = std::stoi(argv[2]);
40             readEvent  = std::stoi(argv[3]);
41
42         } catch (std::invalid_argument& ex) {
43             std::cerr << "Cannot convert " << argv[2] << " to an event number: " << ex.what() << std::endl;
44             return 1;
45         }
46     }
```

```
48     if (startEvent>=readEvent){
49         std::cout<<"Can not use specified range"<<std::endl;
50     }
```

```
52     if (readEvent > eventNumber) {
53         std::cerr << "Only have " << eventNumber << " events to read. " << std::endl;
54         readEvent = eventNumber;
55     }
```

# Podio Event Overview

```
11     if(argc<2){
12         std::cout<<"ERROR FileName not received"<<std::endl;
13         return 1;
14     }
15     std::string FileName{argv[1]};
```

```
17     //Declaring the reader object and opening the file
18     auto reader = getReader(FileName);
19     reader->openFile(FileName);
20     int eventNumber=reader->getEntries();
21     int readEvent=1;
22     int startEvent=0;
```

```
32     } catch (std::invalid_argument& ex) {
33         std::cerr << "Cannot convert " << argv[2] << " to an event number: " << ex.what() << std::endl;
34         return 1;
35     }
36 }
```

```
23     if(argc==3){
24         try {
25             readEvent = std::stoi(argv[2])+1;
26             startEvent= readEvent-1;
27             if(readEvent==-1) {
28                 readEvent=eventNumber;
29                 startEvent=0;
30             }
```