CLAS12 Simulations

CLAS12Tags releases

OSG, Containers, Priority

GEMC3

Streaming Readout

Maurizio Ungaro

CLAS12 Tags: *https://github.com/gemc/clas12Tags*

Production:

- 4.3.2 (paired with Coatjava version 6.5.3)
  - FILTER_HADRONS option to write out events that have hit from specific hadrons in them
  - Rich sector 4 passive materials
  - FMT use "rgf_spring2020" variation with 3 layers and in retrieving Z0 in the digitization
  - RTPC geometry and digitization for the Bonus experiment
  - Target geometry for the Bonus experiment
  - GUI background color changed to white
  - Neutrals particles color changed to blue
  - Double radius for hits above thresholds
  - allow two sequential rotations in the detector definition
  - TOFs resolutions pars from CCDB
  - Move LUND vertex based on gcard entry
  - Detector time signal shift to match data: FTOF and DC

GCARDS are updated with the "_mc" variation: use these only with 4.3.2 and 6.5.3

Interactive Container: **jeffersonlab/clas12tags:4.3.2**

CLAS12 Tags: *https://github.com/gemc/clas12Tags*

In development:

- 4.4.0:
  - geant4 10.6 support
  - conform all detectors to read RUNNO and DIGITIZATION_VARIATIONS in the digitization
  - add time offsets for: EC, LTCC

Numbering scheme changes: hipo4 breaks backward compatibility. So this release is "major". Also, from now on we go to two numbers only.

- 5.0:
  - Hipo 4 output ➡ SOON

- Future releases:
  - Background merging memory check ➡ SOON
  - Rich sector 4 digitization ➡ SOON
  - 3D Cylindrical map field ➡ SOON
  - BMT digitization with global coordinates instead of locals ➡ SOON
  - Time propagation in DC digitization ➡ SOON
  - Detector time signal shift to match data ➡ SOON

Containers Generators

Generators are collected at https://github.com/JeffersonLab/clas12-mcgen/

Generators

name	summary description	maintainer	email	requirements met
clasdis	clas SIDIS MC based on PEPSI LUND MC	Harut Avakian	avakian@jlab.org	✅
claspyth	SIDIS full event generator based on PYTHIA	Harut Avakian	avakian@jlab.org	✅
dvcsgen	DVCS/pi0/eta generator based on GPD and PDF parameterizations	Harut Avakian	avakian@jlab.org	✅
genKYandOnePion	🔴 no description	Valerii Klimenko	valerii@jlab.org	✅
inclusive-dis-rad	generates inclusive electron and optionally radiative photon using PDFs	Harut Avakian	avakian@jlab.org	✅
TCSGen	Timelike Compton Scattering	Rafayel Paremuzyan	rafopar@jlab.org	✅
JPsiGen	J/Psi	Rafayel Paremuzyan	rafopar@jlab.org	✅

If you want to include a generator in the container, please visit the page for the requirements, and contact Nathan Baltzell or Mauri

clas12tags:4.3.2, OSG

- Interactive containers are tagged, i.e. ***jeffersonlab/clas12tags:4.3.2***
- OSG Container is not tagged. “devel” will replace “production” after testing.

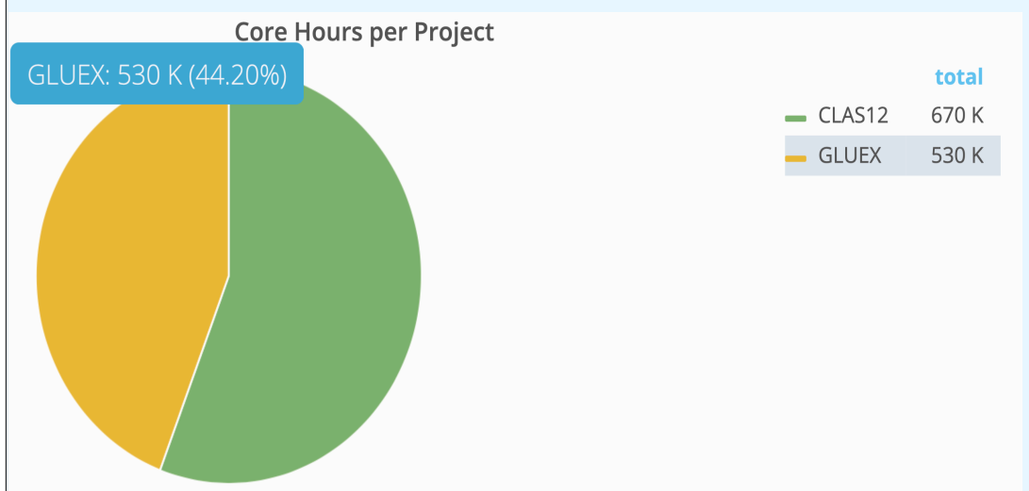
- CentOS……..centos7
- JLAB_VERSION 2.3…….data.jlab.org
- CLAS12TAG 4.3.2…….https://github.com/gemc/clas12Tags.git
- CCDBSQLITE ccdb_2020-03-15.sqlite……https://clasweb.jlab.org/clas12offline/sqlite/ccdb/
- COATJTAG 6.5.3…….https://clasweb.jlab.org/clas12offline/distribution/coatjava/
- JAVATAG 11.0.5…….https://www.oracle.com/technetwork/java/javase/downloads/index.html

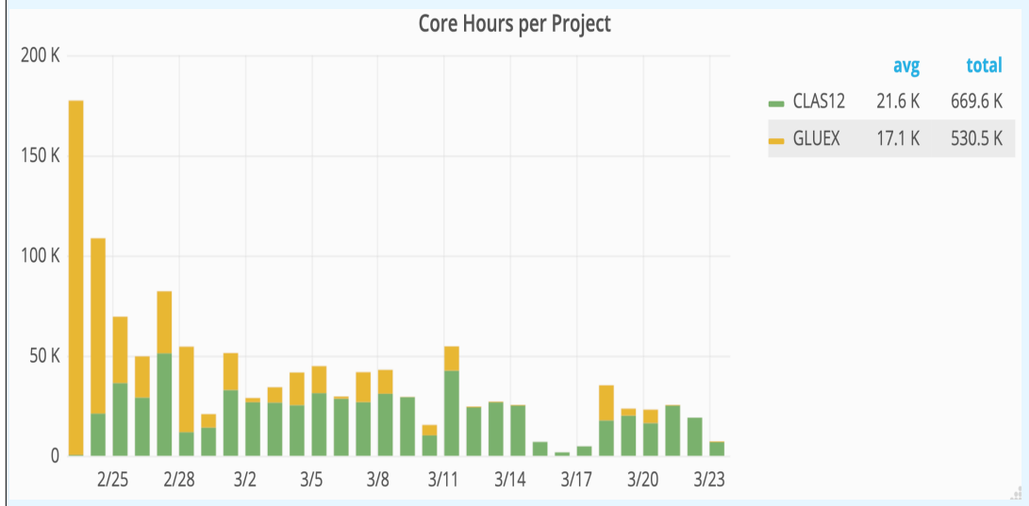
OSG Containers

- CED 1.4.48…….https://userweb.jlab.org/~heddle/ced/builds/
- CLARA + COATJAVA + JAVA: 4.3.12……https://claraweb.jlab.org/clara/_downloads/install-claracre-clas.sh

Interactive Containers

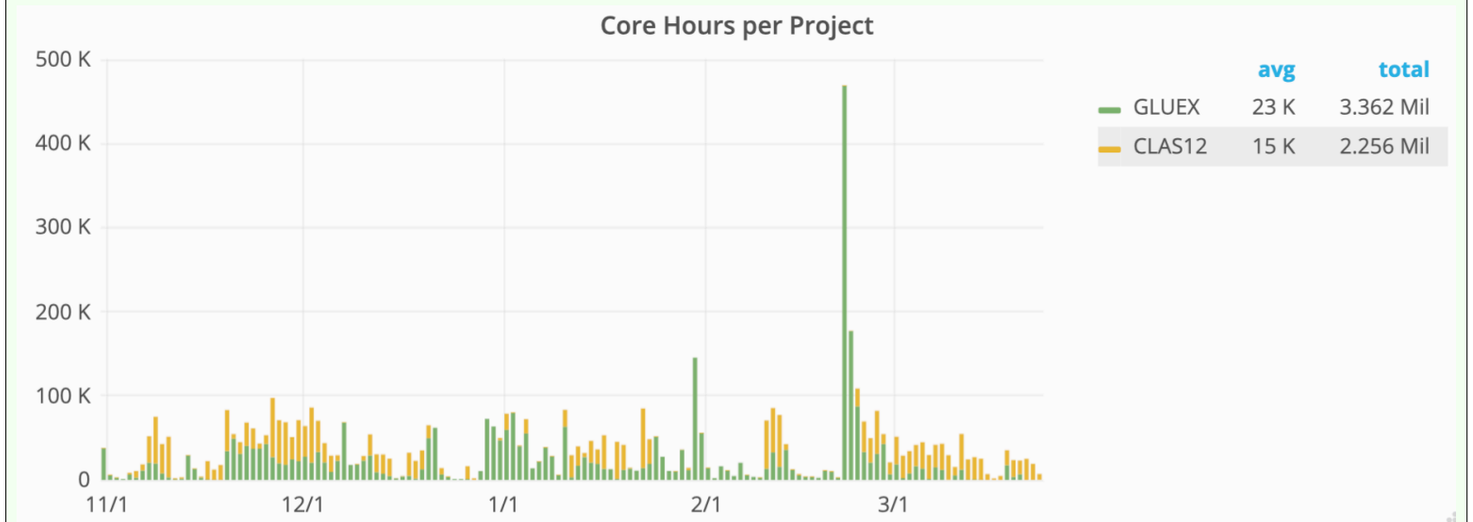
OSG Jobs Submission: Status

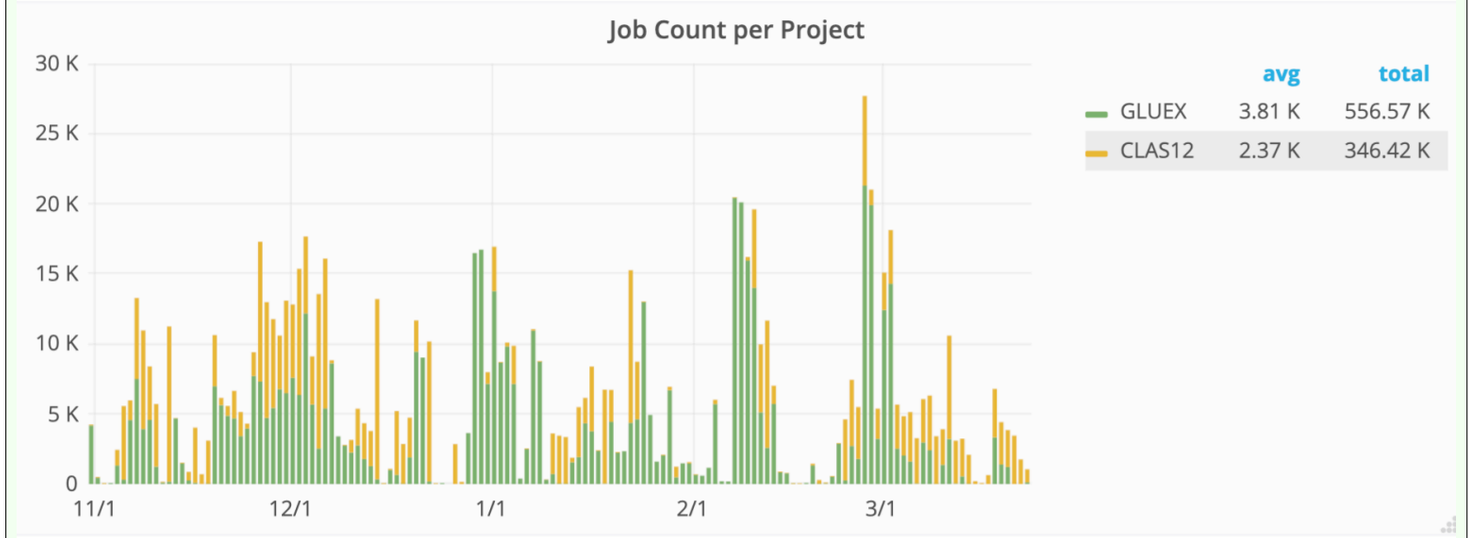




Last 30 days:

0.1% of OSG





Last 5 months

Opportunistic mode, 50% GridPP. Other offsite farms are being tested as we speak (INFN, ICJ Lab)

OSG Priority

System in place to assign priority to OSG ID based on JLAB *username* and # Running Jobs associated with *username*

- by default, all accounts have priority P=1:
- jobs of user “B" do not have to wait user “A” jobs to be finished to start
- at regime, users will have same amount of jobs running

user	job id	submitted	total	done	run	idle	osg id
gjwei	470	3/16 15:37	10	1	9	0	1962859
markov	478	3/18 20:35	5000	4979	21	0	1966243
dien	482	3/20 11:09	1000	908	92	0	1968809
dien	483	3/20 11:10	1000	462	508	30	1968810
dien	484	3/20 15:49	1000	0	361	639	1968814
dien	485	3/20 15:52	1000	1	0	999	1968815
gjwei	486	3/20 23:46	500	488	12	0	1968818
gjwei	487	3/21 01:01	500	211	289	0	1968819
gangel	488	3/21 02:39	40	36	4	0	1968820

2974 jobs; 0 completed, 0 removed, 1678 idle, 1296 running, 0 held, 0 suspended

gj, giovanni didn’t have to wait for dien’s jobs to finish

user	job id	submitted	total	done	run	idle	osg id
gjwei	470	3/16 15:37	10	0	10	0	1962859
markov	478	3/18 20:35	5000	4284	463	253	1966243
dien	482	3/20 11:09	1000	4	473	523	1968809
dien	483	3/20 11:10	1000	0	0	1000	1968810

2722 jobs; 0 completed, 0 removed, 1776 idle, 946 running, 0 held, 0 suspended

at regime, nick, dien, have the same amount of running jobs

OSG Priority Form

Priority Permission Increase Form

We request the increase of the user account _____ to perform CLAS12 simulations offsite, with the details below:

- Task name_____
- Channels _____
- Event generators_____
- Kinematics_____
- Detector configuration (for instance RGB-Spring19, …) _____
- Desired statistics (M = millions of events, B = billion, T = trillion) _____
- Disk space needed for storage _____ TB
- CPU time estimate _____ Million Core Hours
- Date of Start of High Priority: _____
- Duration of High Priority: _____ Days
- Test with actual configuration performed: YES NO

Physics Working Group Chair: (PRINT NAME) _____

Signature: _____ Date _____

Software Group Representative (PRINT NAME)

Signature: _____ Date _____

Acknowledged by:

CLAS12 Coordinating Committee
Software Coordinator [R. De Vita]_____ ☐ Date _____
Computing manager [N. Baltzell]_____ ☐ Date _____
Simulation lead [M. Ungaro]_____ ☐ Date _____

A request according to this form would need to be approved:

- by the relevant PWGs for what concerns the physics and analysis aspects (event generator, kinematics, statistics, …)
- by the Software Group for what concerns the computing resources (CPI hours, disk space, …)

The CCC would be notified as well to make sure all WG chairs are informed and could address possible conflicts arising from multiple requests.

- by default, all accounts have priority P=1
- P = 10 means 10x more running jobs than P=1
- ideally: **one or two** high priority accounts at once

GEMC 3

Complete rewrite. Modern C++(17, 20)

Multithreaded mode

Plugins for generators

Plugins for outputs

Plugins for digitization routines

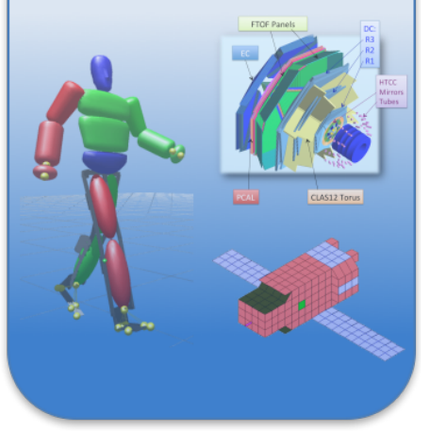
Built-in Streaming Readout

Built-in Hooks for AI

GEMC 3

New generator mechanism uses plugins.
One input file distributed to threads.

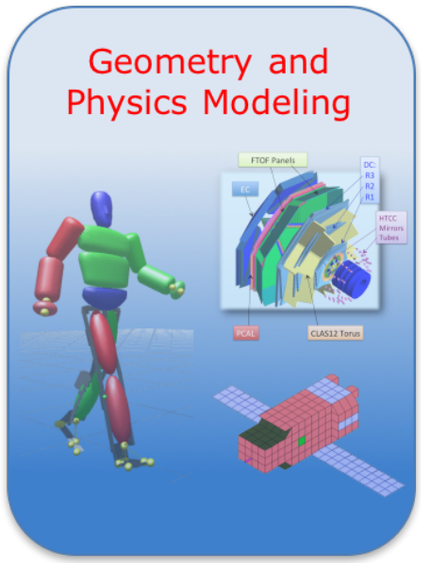
Geometry and Physics Modeling



Python API for geometry. TEXT, JSON, MYSQL Databases.
Possible Additions: DD4HEP, TGEO

GEMC 3

New generator mechanism uses plugins.
One input file distributed to threads.



Python API for geometry. TEXT, JSON, MYSQL Databases.
Possible Additions: DD4HEP, TGEO

Run options: tilts, displacement, calibration, inefficiencies

GTouchable library

Replicas and Divisions

Option library: JSON (jcards, not gcards)

Transport Calculation
- Energy loss
- Secondaries

Digitization
- Hit definition
- Sensitivity

Output
- Bank defs
- File format

Modern Run Action

Dynamic Digitizations: specialized on-demand routines like calibration constants.

Nuclear Physics Cross Sections

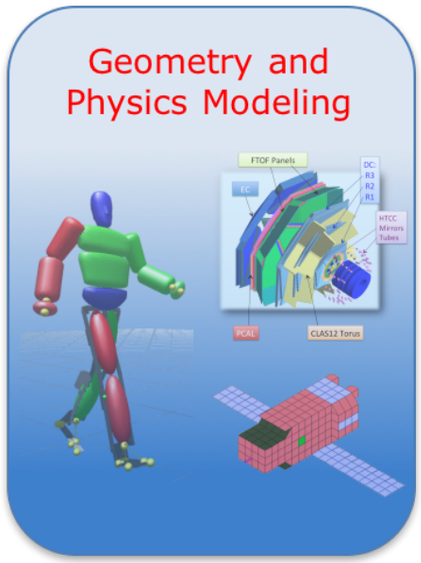
Electromagnetic Fields

GEMC 3

Improved API for digitization, “pulse” output, calibration constants mechanism, output defined in plugins.

New generator mechanism uses plugins.
One input file distributed to threads.

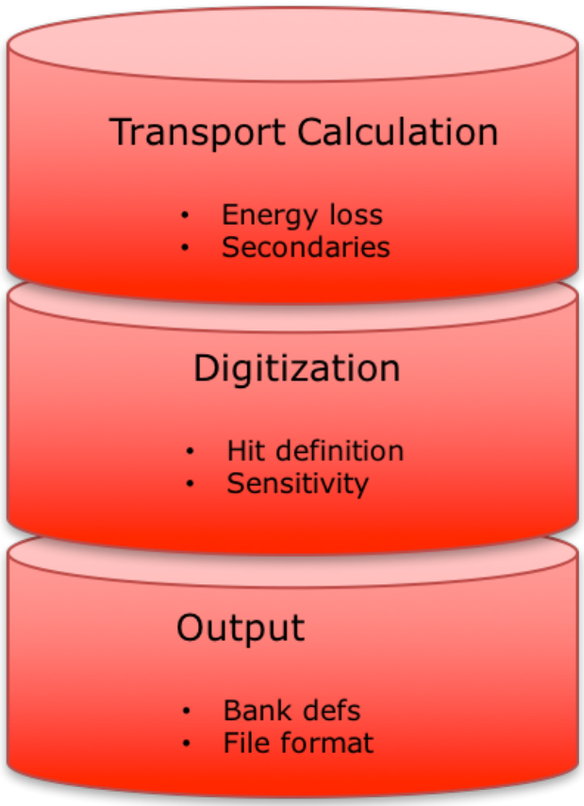
Run options: tilts, displacement, calibration, inefficiencies



GTouchable library

Replicas and Divisions

Option library: JSON (jcards, not gcards)



Modern Run Action

Dynamic Digitizations: specialized on-demand routines like calibration constants.

Python API for geometry. TEXT, JSON, MYSQL Databases.
Possible Additions: DD4HEP, TGEO

Nuclear Physics Cross Sections

Electromagnetic Fields

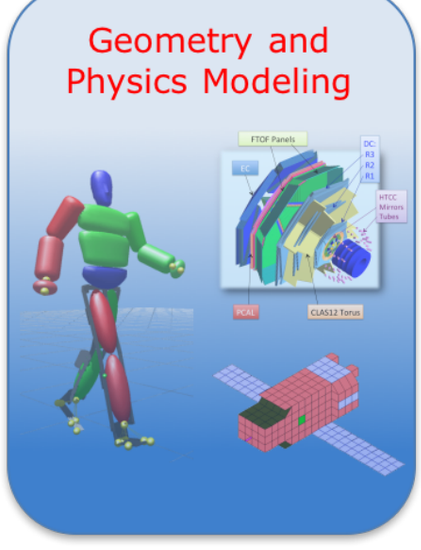
Various Digitization Routines are now “real” plugins: shared objects loaded at run time.

GEMC 3

New generator mechanism uses plugins.
One input file distributed to threads.

Improved API for digitization, “pulse” output, calibration constants mechanism, output defined in plugins.

1 event per thread MT Run Manager



Geometry and Physics Modeling

Python API for geometry. TEXT, JSON, MYSQL Databases.
Possible Additions: DD4HEP, TGEO

Various Digitization Routines are now “real” plugins: shared objects loaded at run time.

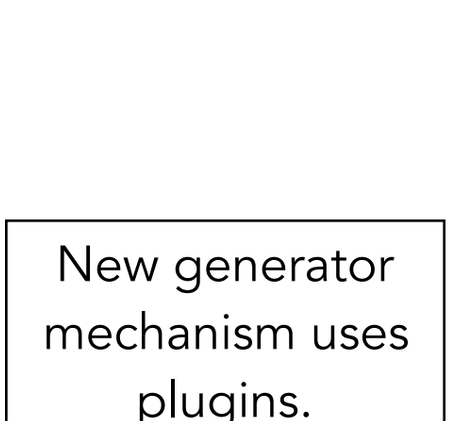
GEMC 3

New generator mechanism uses plugins.
One input file distributed to threads.

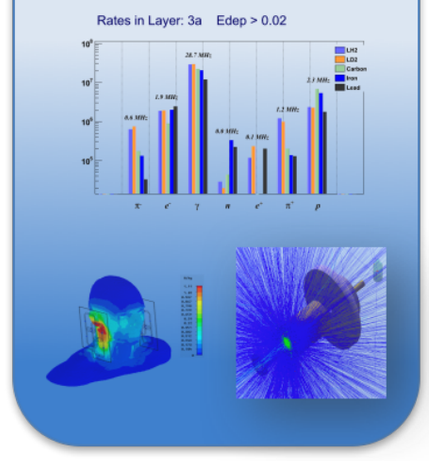
Improved API for digitization, “pulse” output, calibration constants mechanism, output defined in plugins.

Collecting hits from threads: transparent to users.

Geometry and Physics Modeling



Results



Python API for geometry. TEXT, JSON, MYSQL Databases.
Possible Additions: DD4HEP, TGEO

Various Digitization Routines are now “real” plugins: shared objects loaded at run time.

Output plugins: text, hipo4, json, root, streaming

GEMC 3

- Option library: JSON, grouping items (for example particles momentum, vertexes, etc)
- Modern Run Action
- Multithreading verbosity, logging
- GTouchable library
- Dynamic Digitizations: specialized on-demand routines like calibration constants.
- GUI Refurbish
- Event generation library / factories / plugin
- GUI Generator
- Output factories / plugins
- ROOT Output
- Replicas and Divisions
- Multihit TDC
- Read Gzipped field maps
- API to sqlite / mysql
- Streaming readout supports
- Hooks for AI

detectors/clas12/ctof/plugin/hitDigitization.cc

```
GObservables* gdata = new GObservables();

                    double epmt = 0.22*edep;

gdata->addObservable(epmt,        //   value
                  "attEdep",      //    name
                     "MeV");      //    unit
```

(no more external definitions in "banks.pl")

these routines are not compiled with GEMC anymore: shared objects loaded at run time with geometry

detectors/clas12/ctof/plugin/loadConstants.cc

```
bool ctofPlugin::loadConstants(int runno, string variation){

                                    Constant1 = 2.198;

                                       if(runno == 11)
Doubles                             Constant1 = 11.198;
                                       if(runno == 12)
                                    Constant1 = 12.198;

Vectors                     Constant3.push_back(3.0);
                             Constant3.push_back(4.0)

Strings                          Constant4 = "hello";

                                    return true;
}
```

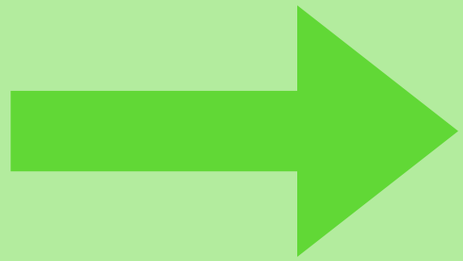
GEMC 3: streaming readout scope

Data Source - Data Streaming

Continuous DATA Stream

TCP

10/40GbE



Data Subscribers, Analyzers

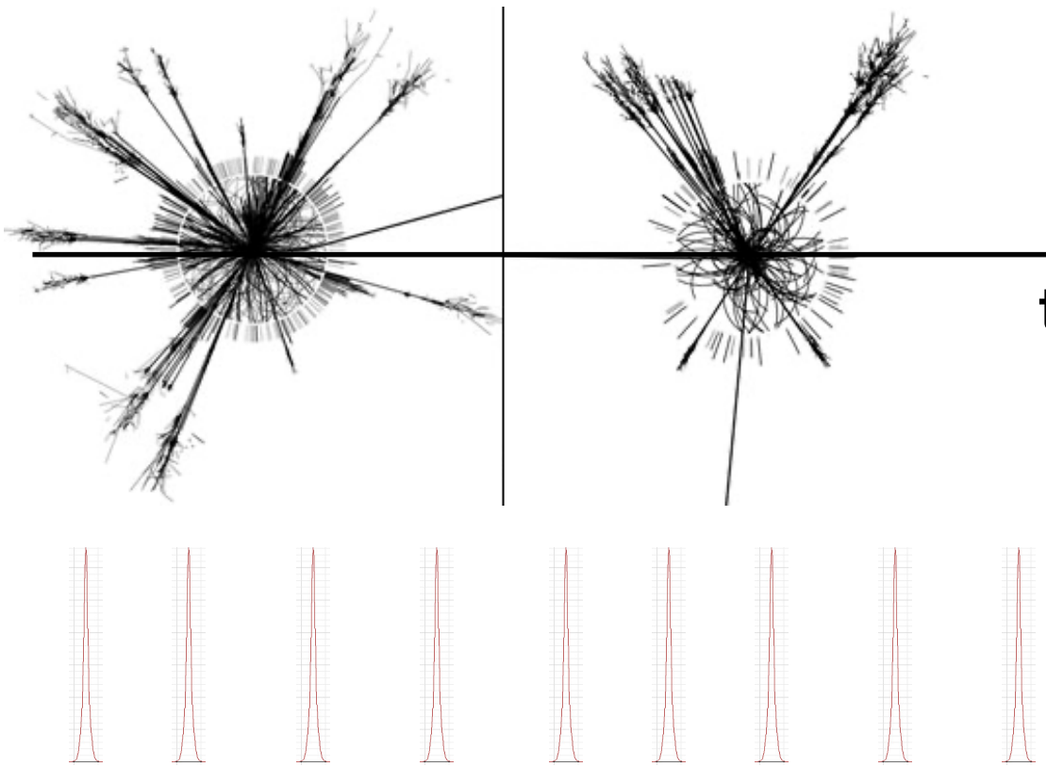
GOAL: Having simulated data that can entirely replace the data source

Streaming protocols / analysis systems should be transparent to the data source: experiment or simulation

This will help addressing challenges on hardware, communications and software issues.

GEMC 3: streaming readout scope

generator: events are at absolute times t_i



beam structure (CEBAF: 4 ns)
for beam physics background
OR
actual data background merged

time

Geant4, Digitization

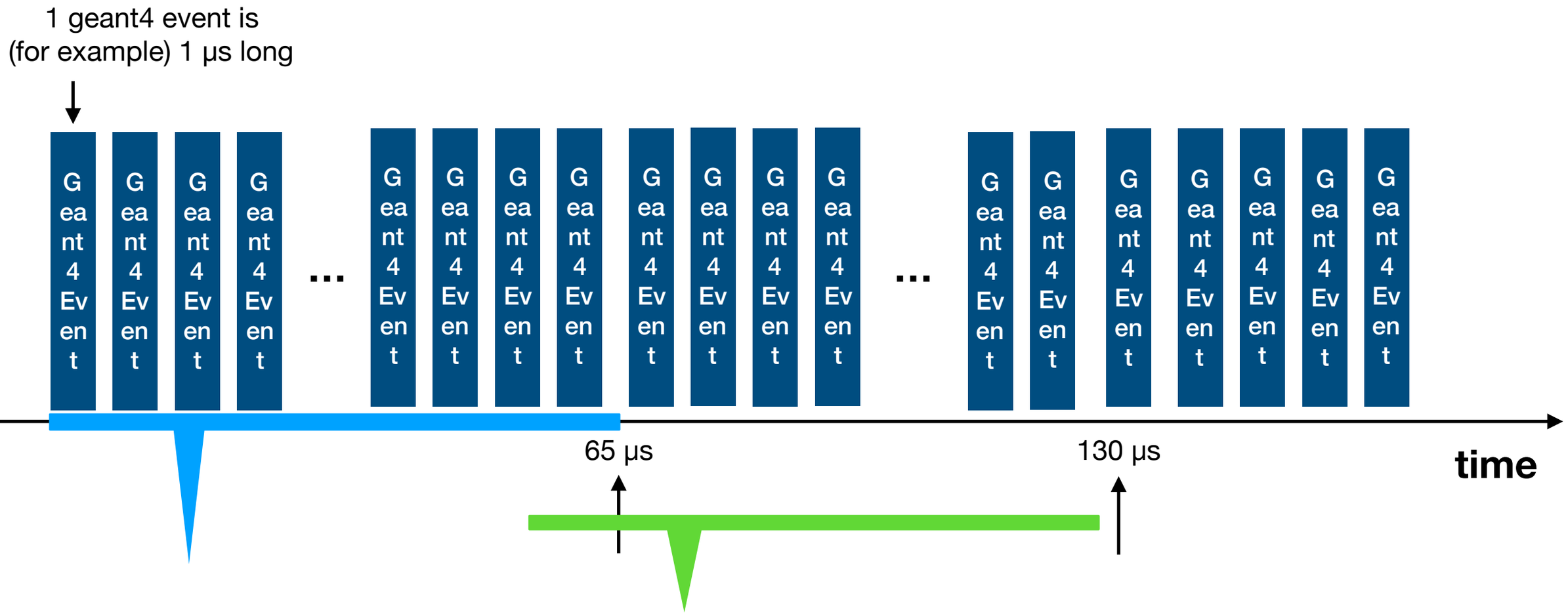
Data Buffers

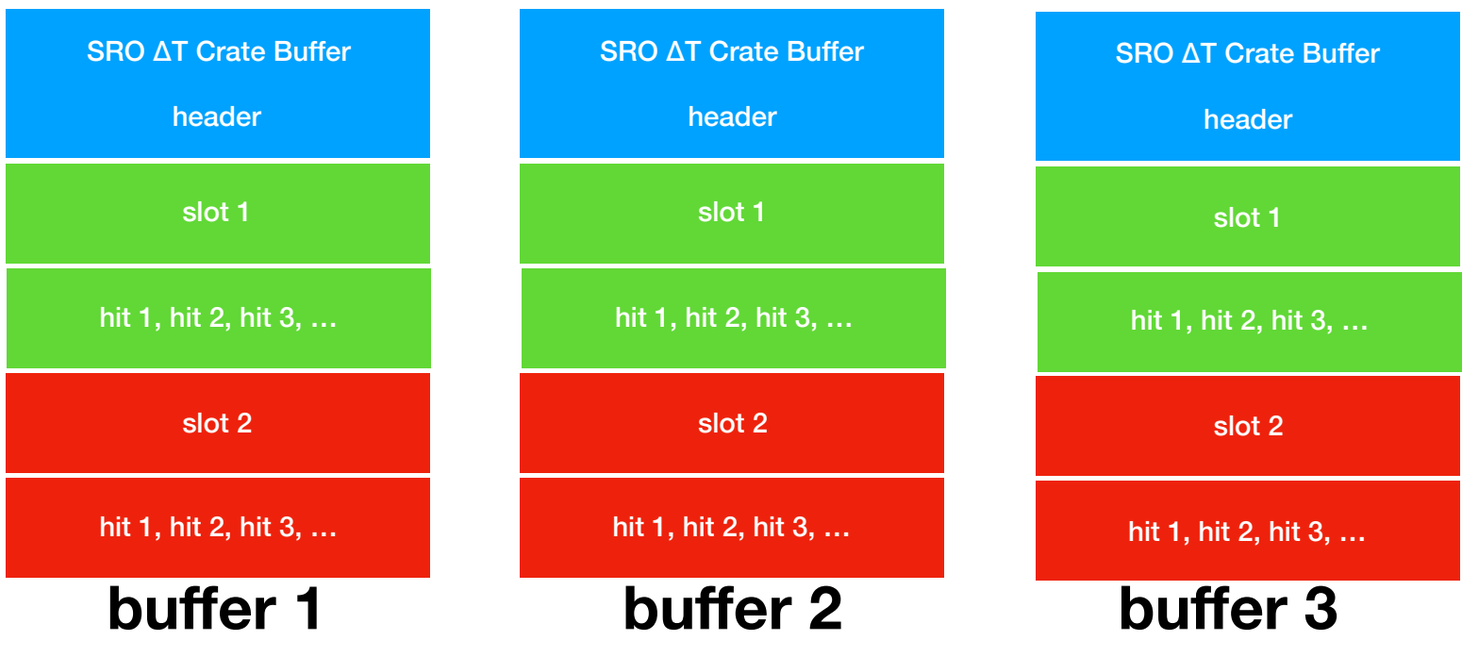
Simulated data that can replace the continuous data stream

No concept of "event"

GEMC 3: streaming readout data buffers

getting around event-centrism





- Hits from a geant4 event can end on different buffers (propagation time, etc)
- Pile-ups are intrinsic in this scenario

Summary

CLAS12Tags 4.3.2
Interactive, OSG containers being tested
Priority Mechanism in place for OSG
Priority form for high priority usage of OSG
GEMC3 work resumed
Streaming Readout

Discourse support:

https://clas12.discourse.group/