Parallel 8B
Towards Reengineering Geant4

Kenel improvements	*KURASHIGE, Hisaya*	
Redwood meeting Room, Building 48	16:00 - 16:20	
CLHEP in Geant4 9.5	*COSMO, Gabriele*	
Redwood meeting Room, Building 48	16:20 - 16:40	
Ideas for G4 re-enginnering	*ELVIRA, V. Daniel et al.*	
Redwood meeting Room, Building 48	16:40 - 17:00	
A new general framework	*MATO, Pere*	
Redwood meeting Room, Building 48	17:00 - 17:20	

Improvements of Physics Vector : ref-08

- Cached variables are confined in a class of G4PhysicsVectorCache
  - G4PhysicsVector has a pointer to the object
- Remove creation of physics vector in event loop
  - New scheme of calculation of velocity
- Review and Clean up source code of G4PhysicsVector and derived class to have same object size
  - No variable is added in derived classes
  - Use G4Allocator for PhysicsVector and PhysicsVectorCache

Other developments : Physics List

- Physics List : → See my talk in plenary 7
  - OrderingParameterTable
    - Default ordering parameters used by G4PhysicsListHelper
    - New Method of ***RegisterProcess()***
  - Default implemantation of SetCuts
    - G4VUserPhysicsList::SetCuts method becomes ***Non-Virtual***
  - Changing PhysicsConstructor
    - Replace a PhysicsConstructor in an existing Physics List
    - New Method of ***ReplacePhysics ()***

      in G4VModularPhysicsList

These modification improve code maintenance and readabilities of Physics Lists

Other developments : Tracking Flag

- Review of ForcedCondition in tracking
  - NotForced : default
  - StronglyForced : G4Scintillation
  - Forced : Event Biasing
  - ExclusivelyForced : Fast Simulation
  - Conditionally → will be removed

Usage status in Geant4

- CLHEP 2.1.0.1 supported
  - Since 9.4 release series
  - New release 2.1.1.0 foreseen for release 9.5 (minor fixes, no migration required)
- Limited set of CLHEP classes used in Geant4
  - Physics vector (3-vectors, 4-vectors, simple rotations)
  - Geometrical vectors and transformations (3D-vectors, 3D-points, etc…)
  - Random numbers and evaluator
  - System of units and constants
  - Other minor uses of CLHEP are restricted to examples

CLHEP classes in Geant4 …

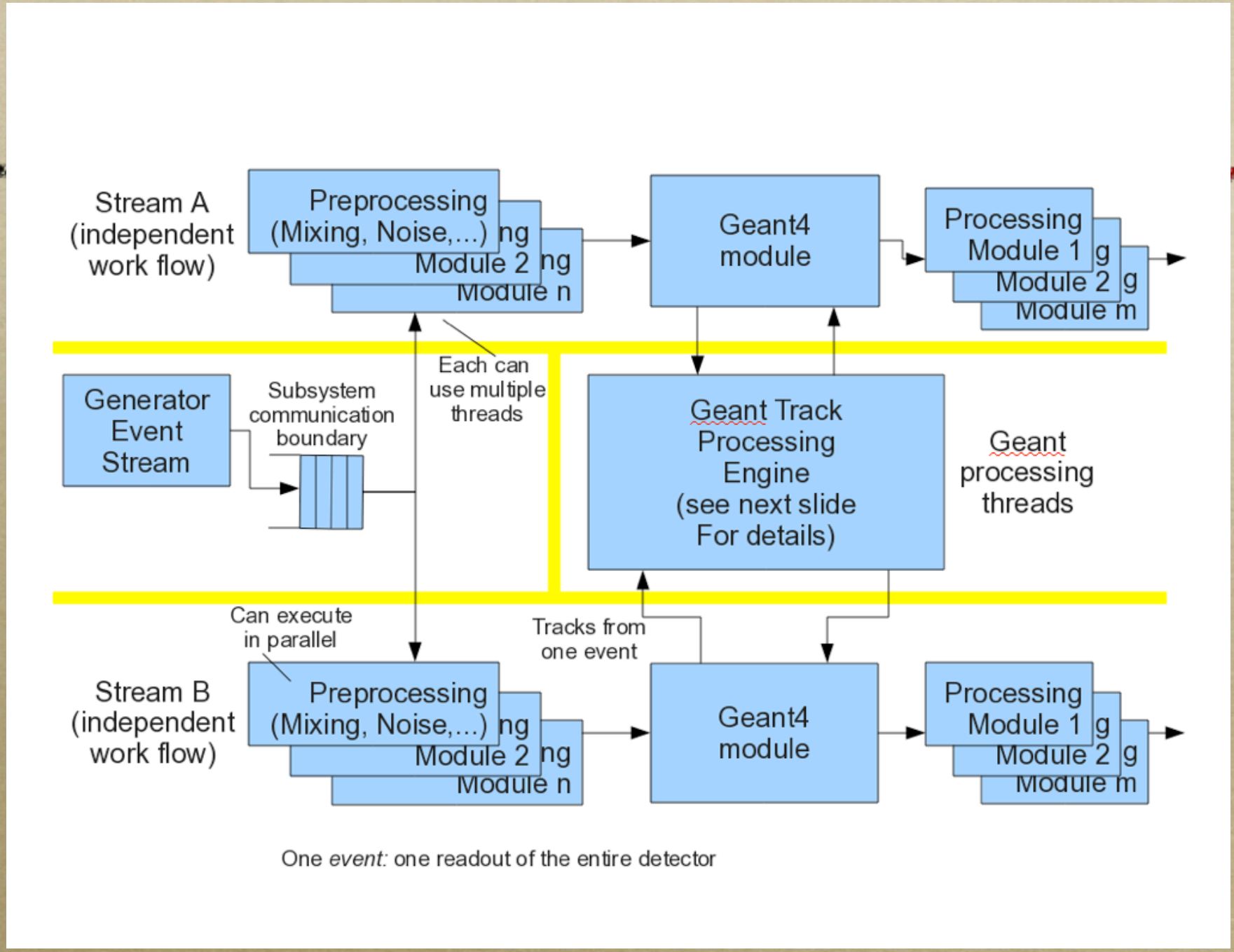
- Integration strategy proposed and discussed since Geant4 Workshop in Lisbon, 2006
  - Now realised since release 9.5-Beta
- New module "externals" introduced
  - Aimed to include any embedded external package
  - Currently including: clhep, expat
  - Planning to move therein also: zlib, gl2ps (from visualization)
- Limited set of CLHEP packages included
  - Evaluator, Geometry, Random, Units, Utility, Vector
  - Excluded obsolete random engines
  - Removed ZOOM exceptions from code
  - Coherent adoption of std namespace
  - Added support for DLL library build (identified relevant symbols to be exported)
  - Integrated with standard Geant4 build system

Simulation And Many Cores

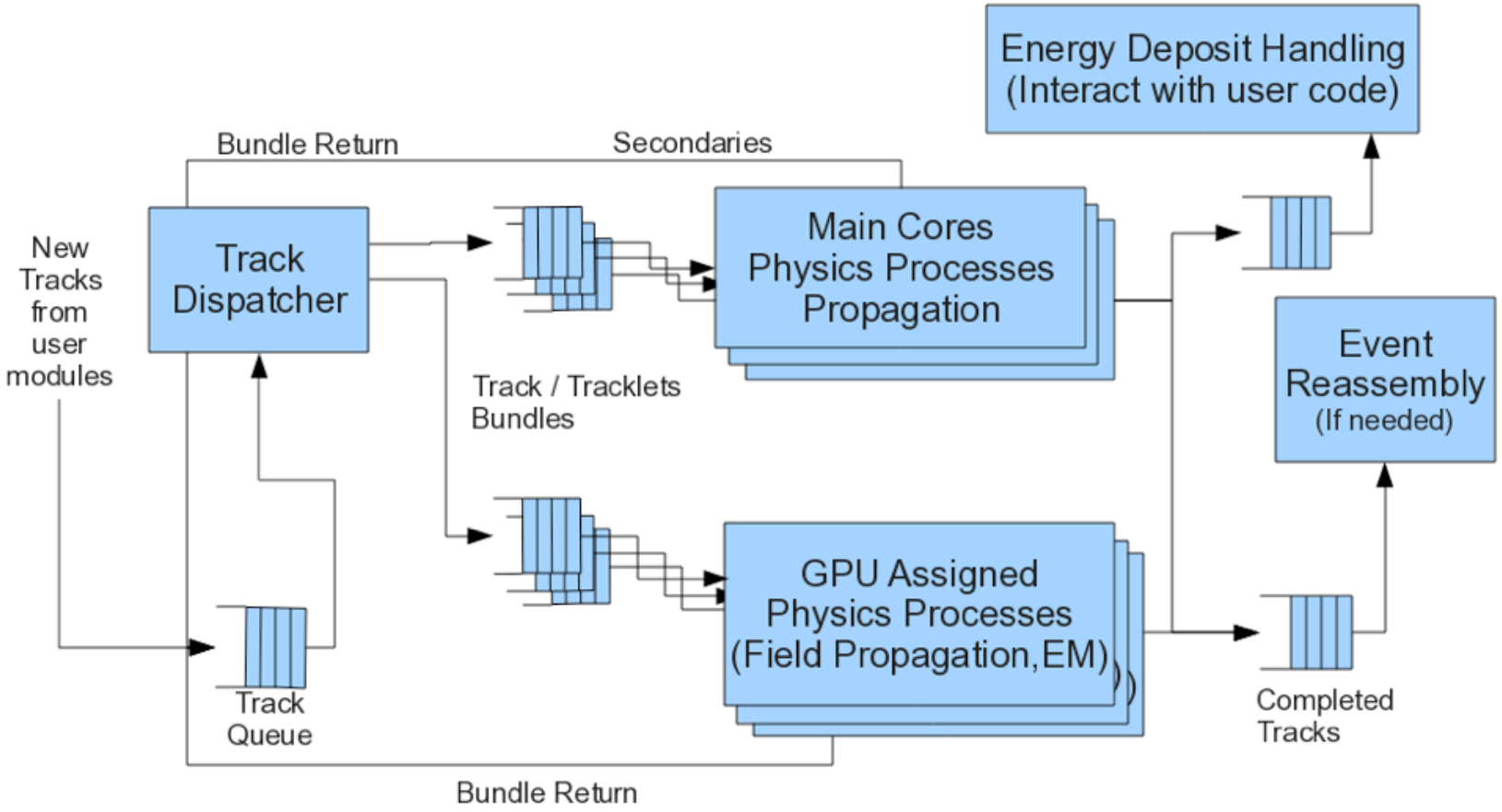
Geant4 Collaboration Meeting
September 22nd, 2011
Philippe Canal, FNAL

Design Directions

- *Replace the looping mechanism from handling one single element at a time to handling multiple elements (vectors)*
  - *Reduce the number of decisions and thus the number of incorrect branch predictions*
  - *Reduce the number of overall functions calls*
  - *Reduce the number of calculations*
    - *For example if several tracks are in the same volume, lookup/calculate/use parametrization only once*
  - *Improve memory locality for example by having collections of light weight objects*


Stream A
(independent
work flow)
Preprocessing
(Mixing, Noise,...)
Module 2
Module n
Geant4
module
Processing
Module 1
Module 2
Module m
Each can
use multiple
threads
Generator
Event
Stream
Subsystem
communication
boundary
Geant Track
Processing
Engine
(see next slide
For details)
Geant
processing
threads
Can execute
in parallel
Tracks from
one event
Stream B
(independent
work flow)
Preprocessing
(Mixing, Noise,...)
Module 2
Module n
Geant4
module
Processing
Module 1
Module 2
Module m
One event: one readout of the entire detector

Track Processing



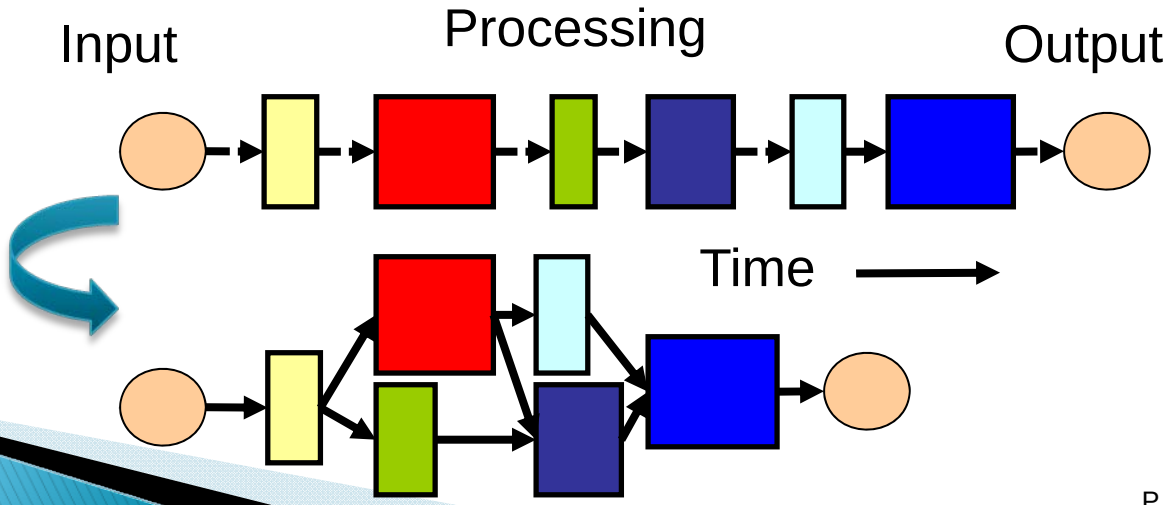
Next Generation of Event Data Processing Frameworks

16th Geant4 Collaboration Meeting
SLAC, 19-23 September 2011

P. Mato, CERN

Concurrent 'chunk' processing

- Framework with the ability to schedule concurrent tasks
  - Full data dependency analysis would be required (no global data or hidden dependencies)
  - Need to resolve the DAGs (Direct Acyclic Graphs)
- Not much gain expected with today's designed 'chunks'
  - See CMS estimates at CHEP'10 (*)
  - Algorithm decomposition can be influenced by the framework capabilities
- 'Chunks' could be processed by different hardware/software
  - CPU, GPU, threads, process, etc.



Re-engineering Geant4

- Geant4 [core] is a toolkit and should continue to be
  - Facilitates the integration in existing applications/frameworks
- However Geant4 applications should be based on a new and more modern framework
  - Configuration, proper scripting, interactivity, I/O, analysis, etc.
  - Plugins based (physics process/models, visualization drivers, etc.)
  - Ability to run full and fast MC together using common infrastructure (e.g. geometry, conditions, etc.)
    - E.g. today's frameworks allow to run different 'tacking algorithms' in the same program
    - Defining clearly the input and output types
- Make use of the common set of foundation packages (math, vectors, utility classes, etc.)

Straw man Project Timeline