

GEANT-4 GPU Port:

Design Document: Detailed Design

Team 8

Stuart Douglas – dougls2

Matthew Pagnan – pagnanmm

Rob Gorrie – gorrierw

Victor Reginato – reginavp

Detailed Design: Version 0

January 8, 2016

Table of Contents

1	Introduction	1
1.1	Revision History	1
1.2	Note About G4 variables	1
1.3	Document Structure & Template	1
2	NeutronHPDataPoint	2
2.1	MIS (Module Interface Specification)	2
2.1.1	Interface Syntax	2
2.1.2	Exported Access Programs	2
2.1.3	Interface Semantics	2
2.1.4	State Variables	2
2.1.5	Environment Variables	2
2.1.6	Assumption	3
2.1.7	Access Program Semantics	3
2.2	Error Handling	3
2.3	Key Algorithms	3
3	NeutronHPVector	3
3.1	MIS (Module Interface Specification)	3
3.1.1	Interface Syntax	3
3.1.2	Exported Access Programs	5
3.1.3	Interface Semantics	5
3.1.4	State Variables	5
3.1.5	Environment Variables	5
3.1.6	Assumption	6
3.1.7	Access Program Semantics	6
3.2	Error Handling	6
3.3	Key Algorithms	6
4	CMake Files	6
4.1	MIS (Module Interface Specification)	6
4.1.1	Interface Syntax	6
4.1.2	Exported Access Programs	6
4.1.3	Interface Semantics	6
4.1.4	State Variables	6
4.1.5	Environment Variables	6
4.1.6	Assumption	6
4.1.7	Access Program Semantics	6
4.2	Error Handling	6
4.3	Key Algorithms	6

1 Introduction

1.1 Revision History

All major edits to this document will be recorded in the table below.

Table 1: Revision History

Description of Changes	Author	Date
Set up sections and filled out Introduction section	Matthew	2015-12-15

1.2 Note About G4 variables

Geant4 uses its own basic types for standard C++ types (G4int, G4bool, G4double, etc). These types are currently just `typedefs` to the respective type as defined in the system libraries.

1.3 Document Structure & Template

The design documentation for the project is based off of templates from WHAT TEMPLATES??????, and is broken into two main documents.

The system architecture document details the system architecture, including an overview of the modules that make up the system, analysis of aspects that are likely and unlikely to change, reasoning behind the high-level decisions, and a table showing how each requirement is addressed in the proposed design.

This detailed design document covers the specifics of several key modules in the project. For each module, an MIS is given fully detailing the interface of the module. Then, the methods for handling errors within the module are discussed, and finally the main algorithms and data structures used by the module are presented.

2 NeutronHPDataPoint

2.1 MIS (Module Interface Specification)

2.1.1 Interface Syntax

Routine Name	Input	Output	Exceptions
G4NeutronHPDataPoint			
G4NeutronHPDataPoint	G4double, G4double		
operator(REVISE THIS)	G4NeutronHPDataPoint		
GetEnergy		G4double	
GetXsection		G4double	
SetEnergy	G4double		
SetXsection	G4double		
GetX		G4double	
GetY		G4double	
SetX	G4double		
SetY	G4double		
SetData	G4double, G4double		

2.1.2 Exported Access Programs

2.1.3 Interface Semantics

2.1.4 State Variables

- energy : G4Double
- xSec : G4Double

2.1.5 Environment Variables

There are no environment variables for this Module.

2.1.6 Assumption

2.1.7 Access Program Semantics

2.2 Error Handling

2.3 Key Algorithms

3 NeutronHPVector

3.1 MIS (Module Interface Specification)

3.1.1 Interface Syntax

Routine Name	Input	Output	Exceptions
G4NeutronHPVector			
G4NeutronHPVector	G4int		
SetVerbose	G4int		
Times	G4double		
SetPoint	G4int, G4NeutronHPDataPoint		
SetData	G4int, G4double,G4double		
SetX	G4int, G4double		
SetEnergy	G4int, G4double		
SetY	G4int, G4double		
SetXsec	G4int, G4double		
GetEnergy	G4int	G4double	
GetXsec	G4int	G4double	
GetXsec	G4double	G4double	
GetXsec	G4double,G4int	G4double	
GetX	G4int	G4double	
GetY	G4double	G4double	
GetY	G4int	G4double	
GetVectorLength		G4int	
GetPoint	G4int	const G4NeutronHPDataPoint &	
Hash			
ReHash			
InitInterpolation	istream		
Init	istream,G4int, G4double, G4double		

Init	istream, G4double,G4double		
ThinOut	G4double		
SetLabel	G4double		
GetLabel		G4double	
CleanUp			
Sample		G4double	
Debug		G4double *	
Merge	G4NeutronHPVector *, G4NeutronHPVector *		
Merge	G4InterpolationScheme, G4double, G4NeutronHPVector *, G4NeutronHPVector *		
SampleLin		G4double	
IntegrateAndNormalise			
Integrate			
GetIntegral		G4double	
SetInterpolation- Manager	const G4InterpolationManager &		
SetInterpolation- Manager	G4InterpolationManager &		
G4Interpolation- Manager		const G4Interpolation- Manager &	
SetScheme	G4int,const G4InterpolationScheme &		
GetScheme	G4int	G4InterpolationScheme	
GetMeanX		G4double	
GetBlocked		vector<G4double>	
GetBuffered		vector<G4double>	
Get15percentBorder		G4double	
Get50percentBorder		G4double	
Check	G4int		G4HadronicException
IsBlocked	G4double	G4bool	

3.1.2 Exported Access Programs

3.1.3 Interface Semantics

3.1.4 State Variables

- G4NeutronHPInterPolator : theLine
- totalIntegral : G4double
- G4NeutronHPDataPoint * : theData
- G4InterpolationManager : theManager
- G4double * : theIntegral
- G4int : nEntries
- G4int : nPoints
- G4double : label
- G4NeutronInterpolator : theInt
- G4int : Verbose
- G4int : isFreed
- G4NeutronHPHash : theHash
- G4double : maxValue
- vector<G4double>: theBlocked
- vector<G4double>: theBuffered
- G4double : the15percentBorderCash
- G4double : the50percentBorderCash

3.1.5 Environment Variables

There are no environment variables for this Module.

3.1.6 Assumption

3.1.7 Access Program Semantics

3.2 Error Handling

3.3 Key Algorithms

4 CMake Files

4.1 MIS (Module Interface Specification)

4.1.1 Interface Syntax

Routine Name	Input	Output	Exceptions
--------------	-------	--------	------------

4.1.2 Exported Access Programs

4.1.3 Interface Semantics

4.1.4 State Variables

- useCuda : Bool

4.1.5 Environment Variables

- NeutronHPVectorGPU.cu : cuda file with GPU code

4.1.6 Assumption

4.1.7 Access Program Semantics

4.2 Error Handling

4.3 Key Algorithms