GEANT4 GPU Port:

Software Requirements Specification
Volere Template, Edition 16

Stuart Douglas – 1214422
Matthew Pagnan – 1208693
Rob Gorrie – 1222547
Victor Reginato – 1209975

Version 0
April 25, 2016

Contents

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
Initial draft of document	Stuart, Matthew, Rob, Victor	2015-10-07
Update for Rev0 presentation	Matt	2016-02-08
Changes document based on Dan's comments	Victor	2016-04-23

2 Project Drivers

2.1 Purpose of Project

2.1.a Project Background

Physics researchers use software simulations to model how particles interact with environments, and to determine the effects of these interactions. Members of McMaster's Engineering Physics department use Geant4 – a simulation toolkit developed by CERN. McMaster's researchers have developed their own fork of the software, G4-STORK, designed to study McMaster's nuclear reactor. Currently, running G4-STORK simulations (as well as other Geant4 simulations) that require many particles takes a long time to compute when run on the CPU. This limits researchers to smaller number of particles than are realistic, and prevents them from seeing the effects of longer time periods.

2.1.b Goal of the project

The goal of this project is to significantly increase the computation speed of Geant4 simulations, which should transfer to all projects written in Geant4. Increasing the speed will allow researchers to use more accurate models, to see the effects of longer time periods on the particles, and to generally increase their productivity. This will be achieved by porting the algorithms that currently run on the CPU to the GPU, taking advantage of the parallel power of the GPU's many cores.

2.2 Stakeholders

The stakeholders that are currently involved with the project are:

- the project group

- the supervisors of the project (Dr. Adriaan Buijs and Dr. Emil Sekerenski)

- McMaster Engineering Physics Department, specifically grad student Wesley Ford

2.2.a   The Client

Clients for the project are [Unnecessary "the" —DS][Fixed —MP] Dr. Buijs and his grad student Wesley Ford, representing the McMaster Engineering Physics Department. The clients proposed the project because they have invested interest in running G4-STORK simulations more efficiently. They will be using the parallelized code to run and study nuclear simulations, and require that the code executes much more quickly to obtain useful data.

2.2.b   The Customer

The customer in this case also includes the clients. As such they will be the part of the end-user group that we will cater to. The customer is also other members of the Engineering Physics department who wish to run simulations using GEANT-4 as they also have use for the end product. The users will want to run simulations with many particles and particle collisions, the optimization of the code will allow for them to do this in a timely fashion.

2.2.c   Other Stakeholders

[removed this section as suggested by peer review —MP]
Collaborators and GEANT-4 users external to McMaster could be potential stakeholders. If the software is successfully completed and the resulting performance improvements are significant, the changes will be submitted to the GEANT-4 repository. It is then up to the GEANT-4 group to decide if those contributions are accepted or not, and if they are then all users of GEANT-4 stand to benefit from them. However, they are not considered stakeholders throughout the development of the project, the focus is instead on those mentioned in 2.2.

2.2.d   The Hands-On Users of the Product

Graduate students and professors from the McMaster Engineering Physics Department will be hands-on users of the product. They will all have familiarity with the existing product.

2.2.e   Personas

Consider Tanner Thomson, a graduate student in the Engineering Physics Department. Tanner wants to study how particles in McMaster's nuclear reactor affect each other, given some specific starting conditions, and a very large number of particles. Tanner knows how to use the existing G4-STORK program, but it will take weeks

to run the computations. Tanner inputs his desired number of particles and specific starting conditions into a G4-STORK simulation. He enables the GPU features of the new product, and successfully runs the simulation in a reasonable amount of time. He observes interesting results that would not have been feasible to obtain with the old product.

2.2.f   Priorities Assigned to Users

Priority will be given to Wesley Ford, the graduate ["graduate" —DS] [fixed —MP] student that proposed the project to Dr. Buijs. He will be the primary user of G4-STORK as it is integral to his current research.

3   Project Constraints

3.1   Mandated Constraints

There are global constraints put in place by the existing software, the stakeholders, and the structure of 4ZP6. The project must be built upon the existing GEANT-4 code, which is used by G4-STORK to run the simulation. The final product must be able to run any code/simulation that ran on the existing software.

The GPU code being used to speed up the simulation is only able to be run on a NVIDIA GPU. Therefore the software must run in parallel on a NVIDIA GPU. The GPU's must also have been made in 2008 or later since GPU's older than 2008 do not support CUDA. Newer GPU's are preferable since they will have better data transfer times. The NVIDIA GPU used must have a compute capability (version) of at least 1.3 because it needs to support double-precision floating-point operations.

Additionally, the final product needs to be completed by the end of April, 2016. If these global constraints are not met the final product is not acceptable.

[Why an NVIDIA GPU? Was that constrained by your client? —DS][Added an explanation for why we are constrained to using an NVIDIA GPU —MP]

3.2   Naming Conventions & Terminology

Throughout the document, "the project", "the product", and/or "the software" all refer to the modified GEANT-4 code that will include the capability to be run on a GPU. The "existing software" refers to the current GEANT-4 simulation toolkit.

Table 2: Glossary

Term	Description
GEANT-4	open-source software toolkit used to simulate the passage of particles through matter
G4-STORK	(Geant-4 STOchastic Reactor Kinetics), fork of GEANT-4 developed by McMaster's Engineering Physics department to simulate McMaster's nuclear reactor
GPU	graphics processing unit, well-suited to parallel computing tasks
CPU	computer processing unit, general computer processor well-suited to serial tasks
GP-GPU	concept of running "general-purpose" computations on the GPU
CUDA	parallel computing architecture for general purpose programming on GPU, developed by NVIDIA
PUC	Product Use case

3.3 Relevant Facts and Assumptions

3.3.a Facts

- GEANT-4 are programmed in C++

- GEANT-4 run simulations on the CPU

- Simulations run calculations on each particle independently

- Calculations on each particle are relatively simple probabilities

3.3.b Assumptions

- The user will have a strong understanding of particle physics

- The user will know how to use G4-STORK

4 Functional Requirements

4.1 The Scope of the Work

4.1.a The Current Situation

The project, Geant4, is currently at a point where it is able to run all necessary simulations, on the CPU. Currently, when the Geant4 simulation is run with a large number of particles or over a period of several seconds it takes a lot of computation time.

4.1.b   The Context of The Work

Figure 1: Work Context Diagram
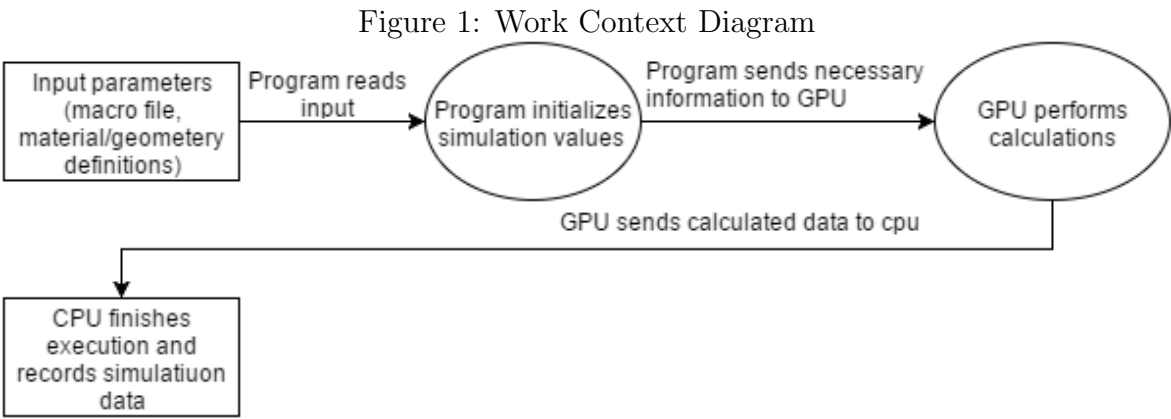
[Figure caption? —DS][Added caption and revised diagram —VR]

4.1.c   Work Partitioning

Table 3: Work Partitioning

Event Name	Input/Output	Summary
Input parameters	overloaded classes and macro file are read (in) overloaded classes are compiled and linked to obj files(out)	This step is just checking to see what in the program needs to be compiled and compiling it before the run is executed
Program initializes simulation values	overloaded classes object files and macro file are read (in) geometry is drawn in simulation world (out)	This event sets up the simulation environment and starts execution of the main run as defined in the macro file
GPU performs calculations	data required for calculations is transferred to GPU memory space(in) calculations are performed and useful work is returned (out)	This step calculates specific methods in parallel on the GPU and returns the result to the CPU.

[This section is supposed to show a breakdown of the business use cases —DS][revised this section —VR]

4.2 Business Data Model & Data Dictionary

NA – The product does not store any data, and requires no data model.

4.3 The Scope of the Product

The following table outlines the use cases for the product.

Table 4: Product Use Cases Summary

PUC #	PUC Name	Actor(s)	Input/Output
1	Simulating Particles	Researcher	Simulation parameters (in), Distribution of particle's locations (out)

Descriptions of each PUC, referenced by PUC # are as follows. [PUC is not in your glossary —DS]. [added PUC to glossary —MP]

1. The software will be used by researchers wishing to simulate large numbers of particles interactions with materials. The researcher sets simulation parameters, including the number of particles, their lifetime, and the material properties before running the simulation. On completion, the program gives back a map of where each particle traveled, so researchers can study where the particles are most likely to end up.

4.4 Functional Requirements

Req. #: 1	**Req. Type**: 4.4	**Use Case #**: 1

Description: Particle computations run on the GPU

Rationale: Design requirement, will allow particle simulations to run faster (requirement 4)

Fit Criterion: Running the product with GPU computation enabled will result in all computations on particles being offloaded from the CPU (existing product) to the GPU (new product)

Priority: Very High	**History**: Created September 29, 2015

Req. #: 2	Req. Type: 4.4	Use Case #: 1

Description: Existing projects should not be affected by the new code.

Rationale: Design Requirement, need to ensure that users can continue to use GEANT-4 as before.

Fit Criterion: Running an existing simulation should produce identical results.

Priority: High   **History**: Created September 29, 2015

[This requirement should be split. The default case should have its own requirement. —DS][split requirement into two —MP]

Req. #: 3	Req. Type: ??	Use Case #: 1

Description: If the user does not specify Geant4 to run on the GPU then it shall run on the CPU by default.

Rationale: User should be able to use Geant4 with the GPU acceleration the exact same way they used it before

Fit Criterion: Running Geant4 without specifying to use the GPU will not run any code on the GPU

Priority: Low   **History**: Created January 8, 2016

[Removed this requirement. Anyone using this software will know if they have a compatible GPU —MP]
[Removed for same reason —MP]

4.5 Look and Feel Requirements

4.5.a Appearance Requirements

[Then why does your 4.1.b diagram show "Visualizations rendered and stored?" —DS][Revised Figure 1 to remove visualization. —VR]

4.5.b Style Requirements

NA - No Style Requirements [This is not a requirement for your end-product. —DS][commented out the CodeOrganization requirement —MP]

4.6 Usability and Humanity Requirements

4.6.a Ease of Use Requirements

NA – There are no Usability and Humanity Requirements
[Removed this requirement since it is done automatically by the code —MP] [Your description shouldn't include "should be easy". It should be detailing *what* the product is supposed to do. Ease of use is a non-functional requirement. —DS][Moved to Ease of Use Requirements —MP]

4.6.b Personalization and Internationalization Requirements

NA – There will be no adjustments for personal and international preferences offered.

4.6.c Learning Requirements

NA – The software will not have any specific learning requirements beyond knowledge of using existing Geant4

4.6.d Understandability and Politeness Requirements

NA

4.6.e Accessibility Requirements

NA – The software will not provide any specific support for accessibility.

4.7 Performance Requirements

4.7.a Speed and Latency Requirements

Req. #: 4	**Req. Type**: 4.7	**Use Case #**: 1

Description: Decrease the time it takes to run a particle simulation on the GPU while maintaining output that is determined to be precise according to (requirement 5)

Rationale: The entire purpose of the project is to improve the speed of the simulation.

Fit Criterion: Running a simulation with a given set of input parameters should complete significantly faster on the product as compared to the existing software. Both should have identical outputs.

Priority: Very High	**History**: Created September 27, 2015

4.7.b   Safety Critical Requirements

NA – The software is not used in safety critical environments. It is a tool for researchers to study particle interactions.

4.7.c   Precision of Accuracy Requirements

Req. #: 5	**Req. Type**: 4.7	**Use Case #**: 1

Description: Results should be produced accurately, within a certain margin of error, whether simulation is run on CPU or GPU.

Rationale: If results are not as accurate as with the existing product then researchers will not be able to draw as strong conclusions.

Fit Criterion: The results of a GPU run simulation should be close to the results of the simulation run on the CPU given that the inputs are the same. How close the results of the simulations are is influenced by the difference in the random numbers generated on the CPU and the GPU.

Priority: High	**History**: Created September 27, 2015

4.7.d   Reliability and Availability Requirements

Req. #: 6	**Req. Type**: 4.7	**Use Case #**: 1

Description: The product should be at least as stable as the existing product.

Rationale: Researchers require an extremely stable product, we do not want to introduce any new crashes or bugs.

Fit Criterion: Testing the product with a variety of simulations should never result in a crash.

Priority: High	**History**: Created September 27, 2015

4.7.e Robustness or Fault-Tolerance Requirements

Req. #: 7	**Req. Type**: 4.7	**Use Case #**: ??

Description: If an error occurs, the product will throw an exception with a description why the exception happened the same way GEANT4 would throw an exception

Rationale: Throwing an exception as a safeguard in case of an error

Fit Criterion: When an error occurs an exception is thrown and the user it notified about it.

Priority: Medium   **History**: Created January 8 2016

[Shouldn't there be some safeguards in case of an error? —DS] [added a requirement for safeguarding in case of an error —MP]

4.7.f Capacity Requirements

NA – see requirement 4.

4.7.g Scalability Requirements

NA – The product will be used by single researchers, and is run locally on their machines. This project does not encompass the functionality for distributing GEANT4-GPU to multiple GPU's.
[What if the researchers wanted to run it on a server with multiple GPUs (or hundreds of CPU cores?) —DS][added a statement indicating this project does not encompass distributed system gpu functionality. —MP]

4.7.h Longevity Requirements

NA – The product is to be completed by April 2016, and there are no plans to extend that lifetime.

4.8 Operational and Environmental Requirements

4.8.a Expected Physical Environment

The product shall be used by an Engineering Physics professor, researcher or student who will be sitting down in a temperature controlled environment.

4.8.b Requirements for interfacing with adjacent Systems

Req. #: 8	**Req. Type**: 4.8	**Use Case #**: 1

Description: The product shall work with GEANT-4 10.00.p02 and later (barring breaking changes in Geant4 introduced after the project's completion date).

Rationale: Support for most recent versions is desireable to users.

Fit Criterion: At least the last four versions of GEANT-4 will be able to run this product.

Priority: Low   **History**: Created October 4, 2015

4.8.c Productization Requirements

Req. #: 9	**Req. Type**: 4.8	**Use Case #**: 1

Description: The product will be available on a public repository along with clear instructions for installation

Rationale: Want to make the product easily available

Fit Criterion: 90% of users should be able to acquire the product with out much trouble

Priority: Low   **History**: Created October 4, 2015

4.8.d Release Requirements

Req. #: 10	**Req. Type**: 4.8	**Use Case #**: 1

Description: New versions of the product that have been patched will be available on the public repository and shall not cause previous features to fail.

Rationale: Users should be encouraged to use the latest patch, and should be confident that updating won't negatively impact their work

Fit Criterion: Each new patch of the product will not cause any previous features to fail.

Priority: Medium   **History**: Created October 4, 2015

4.9 Maintainability and Support Requirements

4.9.a Maintenance Requirements

Because G4-STORK is open source, the code written can be maintained by anyone. The original developers will not be maintaining the code, it will have to be done by end users.

4.9.b Supportability Requirements

There will be a setup.txt file describing briefly how to add a function and set inital parameters.

4.9.c Adaptability Requirements

G4STORK is only able to be run on Fedora and OS X. Therefore the project is constrained to only being able to run on Fedora and OS X. Thus the completed project is expected to run on Fedora and OS X. [Why only those? —DS] [added explanation to why project will run only on Fedora and OSx —MP]

4.10 Security Requirements

4.10.a Access Requirements

Req. #: 11	Req. Type: 4.10	Use Case #: 1
Description: All users have access to all aspects of the product.		
Rationale: The product will be open-source.		
Fit Criterion: Any user should never fail to access some functionality of the product.		
Priority: High		**History**: Created October 6, 2015

4.10.b Integrity Requirements

NA – There will be no compromise on the integrity of data outside of physical hardware malfunctions.

4.10.c Privacy Requirements

NA – The privacy of the data will be managed entirely by and at the discretion of the user.

4.10.d Audit Requirements

NA – The software does not need to meet any specific audits.

4.10.e Immunity Requirements

NA – There will be no specific features to ensure immunity.

4.11 Cultural Requirements

NA – The product is intended for a small number of stakeholders. All content is objective and scientific.

4.12 Legal Requirements

4.12.a Compliance Requirements

NA – The product does not store or access any user information. It does not need to comply with any legal standards.

4.12.b Standards Requirements

NA – There are no internal standards that are required to be met by the product.

5 Project Issues

5.1 Open Issues

- The amount of time required to speedup all classes and libraries involved in G4-STORK is unknown, feasibility of project is in question

- The actual speedup of G4-STORK achievable is unknown it could be far less than desired

5.2 Off-the-Shelf Solutions

5.2.a Ready-Made Products

No fully-fledged solution to the given problem has been implemented and distributed. There have been attempts, however, to solving certain portions of the problem. At the Helsinki Institute of Physics (HIP) a team of three attempted to test the potential efficiency increase of porting Geant4 code to a GPU. They ported some small portions of Geant4, but determined that taking on the entire project would be difficult. Their conclusions supported the conjecture that running the code on a GPU would offer significant increases in performance.

5.2.b Reusable Components

Although there is not any hard-copy code that can be reused, the team of three from HIP have published a research paper that may be used as a starting point for the current project.

5.2.c Products That Can Be Copied

NA – There are no existing products that can be copied.

5.3 New Problems

5.3.a Effects on the Current Environment

GEANT4-GPU is an extension of the current environment. The goal of GEANT4-GPU is to decrease the time that a simulation will take by running sections of the code on the GPU. The output produced by a simulation run with the extension enabled will produce the same results, more quickly, as the same simulation run with the extension disabled. In the sense that the output of any given simulation is not changed, the current environment should be the same to the user when running simulations. The user should only notice changes in the current environment when building a simulation with the GEANT4-GPU extension enabled, not when running the simulation. The difference that the user will notice when building their simulation (using the cmake command) will be the addition of a command line flag to indicate that the extension will be used. [You are describing design decisions, not effects on the current environment —DS][Changed section to describe how design decisions effect the current environment —VR]

5.3.b Effects on the Installed Systems

The GEANT4-GPU extension interfaces with the system by modifying cmake, the software used to build Geant4 on a given system. In order for a currently installed system to use the extension, a re-installation Geant4 with the modified source code is necessary. Projects already using Geant4 simply need to be re [Again you are talking about design. Here you should be detailing what your how your product interfaces with the existing system. Are there any potential conflicts that would need to be solved (i.e. by your design decision)? —DS]

5.3.c Potential User Problems

Due to the separation of the changes and the existing product, users will not negatively respond to the changes, indeed they won't even notice them unless actively looking. To use the new product's features, users will enable them and then execute the program in an identical manner to how the existing software works (requirement ??). [You are

again making design decisions instead of detailing potential problems to be solved by those decisions —DS]

5.3.d   Limitations of the Anticipated Implementation Environment That May Inhibit the New Product

To run the GP-GPU computations, specific hardware is required [You should stop here, unless you explained (in the constraints section) why you need an NVIDIA card. Also, what is "recent"? —DS][I explained why we need an NVIDIA card. did not explain recent. TODO —MP][Expanded on "recent" in the constraints section —VR] (NVIDIA graphics card).

5.3.e   Follow-Up Problems

There are a number of potential situations that could lead to the product failing. We are confident that we will be able to succeed, however we realize that there is a possibility of failure, and have outlined the potential causes below.

- Learning curve for existing G4-STORK codebase is too steep, cannot gain adequate understanding to implement changes in time constraints

- Porting existing algorithms to CUDA requires too much work, and we are not able to run the algorithms on a GPU within time constraints

- The current product's interface for the specific algorithms that will be ported is not well-enough defined

- The large models used in the simulation exceed memory limitations on the GPU, and cannot be run on existing hardware

- Performance gains from the GPU are negligible, due to the structure of the computations

- Numerical accuracy problems lead to different results from simulations run on the existing product vs. the new product

5.4   Tasks

Record of Proposed Project	September 18
Problem Statement	September 25
Requirements Document Revision 0	October 9
Proof of Concept Plan	October 23
Test Plan Revision 0	October 30
Proof of Concept Demonstration	November 16 - 27
Design Document Revision 0	January 1

Revision 0 Demonstration	February 1 - 27
User's Guide Revision 0	February 29
Test Report Revision 0	March 21
Final Demonstration (Revision 1)	Exam period
Final Documentation (Revision 1)	April 1

5.5 Migration to the New Product

5.5.a Requirements for Migration to the New Product

Once the end product has been implemented, all new projects using G4-STORK running on GPU's will use the new code.

5.5.b Data That Has to Be Modified or Translated for the New System

In order to take advantage of the GPU speedup, old projects will need to be modified to use the new classes written for GPU acceleration.

5.6 Risks

Because our software has research based aspirations and goals, some traditional risks are less catastrophic to the success of the project, but are still risks nonetheless. The following may pose to be potential risks:

- Excessive schedule pressure

- Technical/Resource limitations [What do you mean by this? —DS][commented out the inadequate measurements item —MP]

- Optimistic time constraints

5.7 Costs

All software used in the project is open-source and/or available for free. Existing hardware will be used for development, so there are no associated monetary costs.

We have very clear and well-defined deadlines for each deliverable, and are in full confidence that we will meet each one. The time it takes for each deliverable will be variable, but the date of completion for each is concrete, as outlined in 5.4.

5.8   User Documentation and Training

- Function descriptions shall be provided for every new function in the code

- There shall be a thorough Readme file accompanying the project that will explain to the user the changes as well as how to enable GPU computation

- Users who know how to use G4-STORK should be able to easily run it on the GPU

5.9   Waiting Room

There are currently no plans for extended requirements for future releases.

5.10   Ideas for Solutions

Some solutions to a few potential problems have been breifly mentioned in the above section (see 5.3). Outside of those, we can resolve usability issues with a well-written user manual, which will explain how to download the software (or for previous GEANT4 users, how to upgrade). We can also provide documentation which can distinguish if a certain user's machine is eligible for the software.

Additionally, in terms of organizational and production problems that may arise, we will schedule meetings with both our group and our supervisors. We can utilize well known software development procedures (such as an Agile strategy) to ease the managerial work required for development of the product to run smoothly.