G4Stork Tasks

What You Need to Learn

* Advanced c++
* C#
* Java
* Geant4
* G4STORK
  + Read the code
  + Read liams and Guilluames reports
* Git
* Visual studio
  + Creating projects and applications
  + Using debugging, visualization, performance and sharing tools
  + Create touch applications
  + Learn how to use kinetic
  + Creating webpages
* Linux
  + OS structure
  + Bash Scripting
  + Make files
* Windows structure
  + OS structure
  + DOS scripting
  + Nmake files
* Networks
  + How they work
* Advanced reactor physics
* Advanced mathematics
  + Differentials
  + Calculus
    - contour integration
  + Linear algebra
  + Group theory
  + sequences and series
* Advanced particle physics
* Advanced thermodynamics
* Advanced fluid flow
* Advanced CFD
  + Learn finite element and volume algorithms

Setup Version Control for G4STORK

1. Finish reading Git Manual and summarizing it
2. Create repo on Github containing G4STORK, up to date documentation and useful scripts
3. Get Andrew and Salma to merge their code with the repo
4. Create branches for each of us to develop on and merge them periodically to create new releases
5. Add everything related to the project in the repo
6. Create a webpage for the online repo to make it visually pleasing and easy to learn what the project is and how to help develop it

Document the G4STORK Code

1. Read through the classes in descending order
2. Add comments in the code for each function
3. Record information about what the class does and the major mechanisms/functions used in a word document
4. Combine the class descriptions into a manual

Improve G4STORK

* Improve the performance of G4STORK
  + Improve the Doppler broadening algorithm
    - Only Doppler broaden when the change will be significant
    - Use pregenerated cross section data at higher temps
    - Pregenerate entire cross-section spectrum data for materials that will be used a lot and do not change
    - Improve the Doppler broadening algorithm
    - Store cross-section data in the ram
    - Allow for continuous energy Doppler broadening
    - Improve search algorithm when going through cross section data files
  + Improve the geometry tracking algorithm
    - Finish creating STORKUnionSolid and STORKUnionBinaryTree
* Improve the capabilities of G4STORK
  + Add thermal hydraulics physics
    - Couple to an existing thermal hydraulics software
    - Or Create a finite element thermal hydraulics module in G4STORK
  + Improve delayed neutrons generator
    - Based on precursors from previous source distribution file
  + Calculate the adjoint flux
  + Add the ability to assign spatial temperature profiles of geometries
    - this will be needed for the thermal hydraulics
  + Verify G4STORK against MCNP
  + Get software that converts ENDF into G4NDL

Get G4STORK Working on Windows

1. Learn how to use the command prompt environment
2. Learn how unix make files work and remake them in nmake style so that they can run from the visual studio command prompt
   1. Create windows equivalent of unix commands if necessary
3. Figure out how to install all the necessary dependencies using windows tools
4. Figure out how to install GEANT4 and G4STORK using windows tools
5. Write down the procedure you used to install the software in a word doc and upload to git repo
6. Learn how to use Visual studio along with its debugging, class diagram drawer, and performance analysis features
7. Make sure that G4STORK works in windows and use visual studio to try and improve code performance

Create Website for G4STORK

* Learn how to create a website
* Donate old computer to be the server computer
* Put the G4STORK project, all the sub projects, all of the documentation and all of the thesis’ on the website
* Give contact information for all of the current and past developers
* Have links to GEANT4 and the G4STORK team on github
* Have the hierarchy map on here
* Have a list of what needs to be done (G4STORKTasks)

Create Interactive Hierarchy Map for G4STORK

1. Determine which visual software is the best for the project
   1. Gephi
   2. Visual studio
   3. Others
   4. Create my own with java or c#
2. Create the map
   1. Each node is a class
   2. Clicking a node should highlight the node and all the connected nodes
   3. Clicking a link should highlight the two nodes it connects
   4. Double Clicking on the node should provide a brief description of the class and a link to a more detailed description in the manual
   5. Double Clicking on the links between classes should bring up a short description about how the classes interact and a link to the manual for more information
   6. There should be a search bar to look for class names, function names, and other keywords related to a class
   7. The map should be able to be navigated by mouse or touch
   8. The map should be automatically updated when new classes and functions are committed to the project
      1. By reading a file that contains the names of the new classes, functions and other keywords, their relative positions in the hierarchy, pointers to the new documentation (ie the short descript, and the link to the detailed description in the manual)
   9. The map should work on every platform
3. Publish the map along with the necessary tools to run it and a user guide to the G4STORK github website

Create Installers for G4STORK

1. Learn more about creating installation scripts in Linux and in Windows
2. Create installation scripts for both platforms
   1. Use the install manuals you have created for Linux and windows
   2. The install script must all the necessary dependencies, G4STORK and scripts for the user to run g4stork with intuitive commands
   3. The installers should get the dependencies and geant4 from their relative websites and g4stork from the git website
3. Create gui versions of the installers if you have time

Integrate G4STORK into GEANT4

1. Create a separate branch in the G4STORK package
2. Modify G4STORK so that it is in line with the geant4 format
   1. Integrate the STORKParseInput into the geant4 .mac file input system
   2. Edit classes so that they can run independently of each other
   3. Organize the code into sections, geometry, physics, …
   4. Create similar documentation style

Create Better Visualization Tools

1. Create a 3D logo for G4STORK and a 3D animation
2. Using Root, MatLab or AutoDesk create scripts for viewing:
   1. a video of the time evolution of the neutron distribution in the geometry
   2. videos of the time evolution of the temperature, pressure and density distribution in the geometry
   3. videos of the time evolution of the fission, inelastic, elastic and capture process event site distribution
   4. still shots of the neutron, temperature, pressure, density, fission, inelastic, elastic and capture process event site distribution
3. Create an interactive, touch compatible program that show the geometry and allows the user to easily overlay the distribution type they are interested in and change the viewing parameters (angle, position, zoom, color map, quality)

Create an interactive map of the G4NDL file system

1. Clicking on a data file should bring up the data format for the specific file (ie using the explanations below except with out the need of if statements since it is specific to the file) as well as the relevant physics used by the file
2. Click on a directory should explain the general file format of the data files it contains as well as the relevant physics
3. Files should be color coded as to what their format is
4. This whole map should be generated by a program from a hierarchy of text files and the program should allow for easy modification and extension.