High Energy & Nuclear Physics: Software

Computer Science and Physical Science Collide for the NuSteam Program
About me
Dr. Gene Van Buren

- PhD in nuclear physics
- Participated in a variety of physics experiments, but most of my career has been on the STAR Experiment at RHIC
- Calibrations & Productions Leader for STAR
- Co-Leader of STAR's Software & Computing Team for 10 years
• Programming: Fortran [almost dead], C, C++, Java, Python
• Scripting: csh (or other shells), Python, Perl (and PHP), xml, HTML, Javascript, Jupyter Notebook
• Frameworks [not a complete list]:
  • Community-wide: PAW [Fortran: dead], Root
  • Smaller: Clara, Fun4All, Root-spinoffs (e.g. AliRoot, root4star, FAIRRoo)
Root
Framework example

• Write one, or a few lines of code directly at the prompt and see the results immediately

• Write a macro with up to a modest number of code lines that may be worth executing repeatedly, and test immediately

• Load compiled shared object libraries and execute C functions and use shared objects and their member functions
  • Use at the Root prompt requires a dictionary and a few standard rules

• Let's go to the command line and try it out...
Data Flow

From experiment to publication

- Raw data (see Jeff Landgraf's presentation tomorrow)
- Reconstruction
  - Calibration
  - Production
- Analyses
- Presentations (talks, posters)
- Publication
- Archives
Data Formats
From experiment to publication

- Raw data (see Jeff Landgraf's presentation tomorrow)
  - raw data files
- Reconstruction
  - Calibration
    - databases or constants files
  - Production
    - binary data summaries ("DSTs")
- Analyses
  - user data summaries or text files
- Presentations (talks, posters)
  - tables & plots
- Publication
  - selected tables & plots
- Archives
  - all of the above go to long term storage == tape
Reconstruction
Converting raw into physics

- Calibrations are applied to raw data to get physical measurements
- Physical measurements are quantitatively combined to get particle measurements
- Analysis: particle measurements are combined to get statistical measures of actual physics
Reconstruction

Example images from STAR

Simulations: event generators
Understanding the science

• Execute the physics of a particle or nuclear collision (e.g. Pythia, Hijing, URQMD)
  • No description of detectors
• Pre-experiment:
  • What are signatures we can look for experimentally?
• Post-experiment:
  • What can explain what we see experimentally?

Animation example:  https://www.sdcc.bnl.gov/phobos/Animations/zx_jan122008.mpg
Simulations: detectors
Understanding the experiment

• Frameworks for describing the detectors, and for executing the physical process that occur when particles traverse those detectors (GEANT)

• Design stage:
  • Are detectors capable of achieving performance needs? (e.g. how spatial resolution of multiple measurements becomes a single momentum resolution)?
  • Are detectors susceptible to impeding factors (e.g. background particles, occupancies, blocking materials or fields)?
  • How does reconstruction software perform?
Simulations: detectors
Understanding the experiment

Various GEANT versions of STAR detector components
Simulations: embedding
Understanding the experiment

- Extensions to detector simulators allowing for overlay of simulated data (particles or whole events) into real data (real backgrounds, occupancies, statuses)

- Analysis stage:
  - How well did the reconstruction software perform in a real (non-ideal) data environment?
  - What can be learned about systematic uncertainties? (quantify to some degree what it is we don't know)
Simulations: Monte Carlos
Understanding complex math

• Simulations often involve a complex system of materials and possibilities
• The possibilities are often factorizable as random probabilities
• Even so, determining expectation values for averages or ranges of any sort can involve daunting ... usually impossible ... integrals of these various probabilities
• In a Monte Carlo simulation, a statistical approximation is found by repeatedly running the experiment and throwing the dice for each random probability
• Repetition (often in the millions) builds up statistically strong answers
"It can all be done in software."

Thanks for your time and attention! Good luck!