Research Management Plan for the Hall C 12 GeV Detector Software

3/19/2012

1. Introduction

This document describes plans for the development and implementation of the data analysis software for Hall C experiments for the initial period of 12 GeV running. The data analysis software will be based on the set of Fortran code (Hall C ENGINE) developed for experiments using the High Momentum Spectrometer (HMS) and the Short Orbit Spectrometer (SOS) in the 6 GeV era. The SOS will not be used for the 12 GeV experiments. The Fortran code relies on the CERNLIB histogramming and data storage libraries which are no longer supported by CERN. The Super High Momentum Spectrometer (SHMS), which is being built, and the HMS will be used for the initial set of 12 GeV experiments. The detectors used in the SHMS are similar to those in the HMS and therefore the software needs are also similar. The simplest approach would be to copy and modify the existing HMS Fortran code to be used for the SHMS detectors. This will be the first part of a two part strategy for having the data analysis software ready and well tested before the 12 GeV experiments start. The second part of the strategy is to develop a C++ ROOT analysis software. Hall A has already developed a C++ ROOT analysis software which has been used for many years and Hall C would build upon the Hall A software. With the downtime between the end of 6 GeV running and the start of the 12 GeV experiments, the time and manpower for changing to a different programming style is available. The decision to develop a C++ ROOT analysis software is motivated by the need:

- To have a modern object oriented language. New students will be more familiar with C++.
- To have histogramming and data storage in the ROOT libraries which are support by CERN and the world-wide community.
- To have similar style codes in Halls A and C which both use spectrometers so that users can minimize the cross Hall learning curve.
- To share code development and documentation with Hall A and to take advantage of ROOT and C++ software developed elsewhere in the world.
- To have a straight forward mechanism for adding third arm detector setups to the code. Hall A has had great success with adding BigBite and other third arms to their software package during the 6 GeV era.

Monte Carlo simulation is a necessary part of the analysis to produce final results from the experimental data, but a full blown GEANT based Monte Carlo is not needed. The Hall C Simulation Monte Carlo, SIMC, was used for HMS-SOS coincidence experiments in Hall C during the 6 GeV era. SIMC has been modified to include code for the optics, apertures and detectors of the SHMS.

2. Goals

The main goal is to have data analysis software ready when the experimental program starts so that data can be analyzed online and offline and quickly have publishable results. In addition, the Hall C Simulation Monte Carlo has been expanded to include the SHMS. A meeting of interested users and staff has been held. A set of milestones have been established (see Table 1) and volunteers are been

assigned to manage different aspects (see Table 2). For the development of the C++ Analyzer , the initial goal is to be able analyze the scintillator detectors from old HMS 6 GeV data to compare with the original Hall C Fortran ENGINE Analyzer. This would be accomplished by the end of 2012. Then by mid 2013 , the C++ code would be ready to begin a full comparison to the ENGINE analyzer.

In parallel to the C++ code effort, the Hall C Fortran ENGINE Analyzer will be updated to include the SHMS. First the ENGINE code will be documented to aid in the implementation in the new C++ Analyzer. Then code will be added for the decoding of the new ADC and TDC modules. The final step will be to add the code for the SHMS detectors.

3. Milestones

Milestones are listed in Table 1.

Table 1: Milestones

Month/Year	Milestone
5/2012	Define management structure. MOUs setup.
6/2012	Official Simulation Monte Carlo code ready
7/2012	Code management system deployed
7/2012	Define set of reference HMS data for testing ROOT analyzer
9/2012	Preliminary definition of DAQ hardware/Data format
10/2012	DAQ decoding software more “object oriented” in Hall A ROOT analyzer
11/2012	Ability to decoding old Hall C data in ROOT analyzer
12/2012	ROOT analysis of HMS hodoscopes verified with old data set
1/2013	Complete definition of software functionality
4/2013	Decoding of new DAQ hardware ready in Fortran Analyzer
6/2013	SHMS added to Fortran Analyzer
7/2013	Full analysis of HMS from old data set
9/2013	SHMS Code (C++) ready for shower counter tests
12/2013	Full ROOT analysis of legacy HMS data verified
1/2014	Decode and basic analysis of BCM/BPM data in ROOT Analyzer
2/2014	Calibration codes ready
4/2014	Both ROOT and Fortran analyzers ready for cosmic tests of individual detectors
7/2014	Demonstrate focal plane analysis with cosmics (HMS)
9/2014	Basic Coincidence analysis, First beam
12/2014	Data driven bug fixing/code improvements

4. List of Tasks

5. Statement by Hall C Leader

6. Institutional Responsibilities

Table 2

	Task
Mark Jones, Jefferson Lab	Software Manager
Gabriel Niculescu, James Madison University	C++ Root Analyzer Coordinator
Ed Brash, Christopher Newport University	Fortran Analyzer Coordinator
John Arrington, Argonne National Lab	Calibrations and Workflow
Pete Markowitz, Florida International University	Online Histograms
Dave Mack, Jefferson Lab	Physics Requirements
Dave Gaskell, Jefferson Lab	Monte Carlo Simulation