Requirements for Analysis Package Framework

At the software workshop June 4-6 1997 we had an extended discussion on the an analysis framework. The discussion centered on how we can make a choice on a package in a relatively short term. The is clearly no obvious choice. We decided to make a list of needed and desirable features which such a package should have. Below is my first attempt to summarize this, and do welcome comments both from the participants in the meeting and other interested persons. The later part of the document has considerations of how to make the decision and time scale.

REQUIREMENTS

Data access and manipulation:

- **requirement** : The persistent data format should be common from the DAQ to the final physics analysis.

- *consideration:* This is a desirable feature. It will make access to various components uniform. Different persistent formats are possible, but may require different access methods form the users point of view.

- **requirement:** It must be possible to place Data from different stages of processing (DAQ, analysis) on different physical files/volumes. Likewise access to such distributed storage must be possible.

- *consideration:* With the large amount of data which will be accumulated it is not desirable nor feasible to replicate significant parts of the data in subsequent analysis.

- **requirement:** It should be possible to combine (chain) data from many different runs in a single analysis in a transparent way.

- *consideration***:** A convenient feature which requires minimal users interaction during pass analysis.

- **requirement:** The persistent data format should be self describing and thus machine independent. It might be desirable to be able to compress the data structures at the object level.
- *consideration:* The different architectures likely to be used (Pentium Pro, Alpha) employs little endian data format while other (ibm, SGI, hp) uses big endian.
- **requirement:** Data structures should be extendable such that future versions of code is still able to read old structures/data of course with the caveat that simple extensions are done and that additional data elements most likely are undefined.
- *consideration:* This is a convenient way to not have version depending access routines to data.
- **requirement:** The Framework should run on all the platform supported by RCF.
- **requirement:** The package itself or a BRAHMS supplied top-level interface needs to be simple to encourage analysis by all physicist in the collaboration.
- **requirement:** The lower level user and detector module must fit into the framework in a modular way.

Platforms

The framework should run on most of the platforms which will be available to the collaborations. This applies to as well operating systems (UNIX, LINUX, windows-NT) as to the underlying architecture.

Support

The package should be supported by RCF, as should compilers and utilities.

DECISIONS and TIMESCALE.

In terms of decision process development of data structures, Data Flow Diagrams (DFDs) should proceed without the final decision of analysis shell. To help in the evaluation within the two possibilities of ROOT and STAF test example consisting of reading GBRAHMS output and performing some simple operations in at least two user modules should be done.

The overall time scale for establishing the analysis framework is to have a prototype analysis working by late 97, and a final package ready by end of 98.

The prototype is of course based on simulated data and could include an analysis chain which goes from digitization (calibration) tracking, and PID to an output in form of N-tuples or ROOT files of identified tracks and associations. It is envisioned to deal with TPCs Drift chambers and Time-of-flight detectors for both MRS and FS.

The Full analysis chain should in addition to tracking also include PID with Cherenkov detectors tools for cross sections , efficiency and acceptance calculations, and access to database.

TPC clustering should have been checked out on real test-beam data.