PrimEx Note # 29
January 12, 2005

Implementing a PrimEx 'Book Keeping' Database for Tracking Run Information

Dustin McNulty

1. Introduction

This document details the design and use of a MySQL database for cataloging PrimEx run and intra-run information (commonly referred to as PrimEx book keeping). This database is named *book_keeping* and can be accessed in the usual way (prompt>mysql -hprimexdb -uprimex_user book_keeping). It is an offline database designed to collect information from various sources and reorganize it into a compact set of tables; these tables can then be used to select runs and regions within runs to include or exclude in ones analysis in a consistent, highly flexible manner.

2. Database Structure

The database is named *book_keeping* and currently contains two tables: *run_list*, and *epics_events*. The run_list table possesses a unique entry for each *run* during the experiment. This entry contains various parameters which describe the run as a whole (e.g. radiator B, Carbon target, etc.). The epics_events table is a collection of all EPICS events acquired during the experiment; each enrty in this table is associated with a unique run and event number. The epics_events table will be detailed at the end of this section. The following will describe the run_list table.

2.1. run_list Table

A description of this table is given in Table 1. (Note that this table can be reproduced by executing the following command: prompt>mysql -hprimexdb -uprimex_user book_keeping −−execute="describe run_list"). Currently, the *run* field ranges from 4100 to 5448. Table 2 shows the distinct values for the remaining fields with the exception of Mod_time which is simply the time in year/month/day/hour/minutes/seconds (e.g. 20050105152255 for January 5, 2005 15:22:55) in which the entry was last modified.

The only values in Table 2 that require additional explaination are the *good*, *bad*, and *unknown* values associated with the *ped*, *lms*, and *production* fields. *good* indicates that a skim file was produced and analyzed to yield reasonable and usable data, *bad* indicates that a skim was produced but that it was found to possess unusable or incomplete data, and *unknown* values indicate there was no skim produced for *ped* and *lms* fields and for *production* it means no information is known about that particular *run*. The default value for all fields in Table 2 is *unknown*.

Table 1: Description of the run_list table.

Field	Type	Null	Key	Default	Extra
run	int(11)		PRI	0	
radiator	varchar(32)	YES		NULL	
target	varchar(32)	YES		NULL	
type	varchar(32)	YES		NULL	
ped	varchar(32)	YES		NULL	
lms	varchar(32)	YES		NULL	
production	varchar(32)	YES		NULL	
Mod_time	timestamp(14)	YES		NULL	

Table 2: List of distinct values for most fields in run_list table.

radiator	target	type	ped	lms	production
D	unknown	CALIBRATION	good	good	good
OUT	LEAD	OTHER	bad	bad	bad
B	EMPTY	TAC	unknown	unknown	unknown
C	CARBON	PI0			
A	BERYLLIUM	COMPTON			
unknown	TIN	unknown			
		COSMIC			
		PAIR			

2.2. epics_events Table

The fields in this table are too numerous to list. There are 170 in total, 160 from the epics bank (e.g. beam currents, hall temperatures, etc.) and 10 from the eventid bank (e.g. event number and time). The epics bank variables are all floats and the eventid variables are integers. These events occurred, on average, every ~ 15 seconds, thus there were ~ 250 epics events per hour of data-taking. There are approximately $425,000$ entries in epics_events table spread over 1100 runs. For any given event, some of the epics-bank variables will have NULL value which means they were not read-out for that particular epics event.

3. Book Keeping Data Sources

3.1. run_list table

The *run* values in run_list were obtained from the list of stub-files located in /mss/hallb/primex/data/october_2004/; a unique *run* value was associated with each

set of corresponding stub-files found in this directory. The *radiator* values were obtained directly from the primex_online database RunID table using a simple query such as "select Radiator from RunID where Run_Number=<specific #>". The *target* values were not successfully stored in the RunID table and thus had to be determined by an alternate method involving a brute force search through the Run_Comment field of the RunID table. For now, the values of this field are the least accurate and are also mostly marked *unknown*. The final values for the *target* field will be determined from the stepper motor EPICS x and y values acquired during the run.

The *type* field describes whether the run was the following: $CALIBRATION$–specifically snake scans, $OTHER$–currently unknown, TAC, $PI0$–π^0 production, $COMPTON$, *unknown*–default, $COSMIC$, and $PAIR$–e^+e^- production. These values were obtained from a combination of searching through the Run_Comment field and the paper run summary sheets filled out by shift workers during the run.

The *ped* and *lms* field values were obtained from run lists created by the analysis of the raw data associated with the corresponding phase of the run; For most runs, pedestal data was acquired during phase 1, Light Monitoring System (LMS) data was acquired during phase 2, and phase 3 was for production data. The *production* field in run_list is meant not only for phase 3 data but is a more general field used to indicate whether the type of data given in the *type* field for a particular run is deemed good, bad, or unknown. The source for this information is from the analyses of the raw production data of the specific *type*.

3.2. epics_events table

Of course the sources of data for this table are the raw data files. The EPICS events were filtered from the raw data files and stored on skim files along with the EVENTID bank values. These skim files were then used to produce each entry in this table.

4. Database Use

As mentioned in the introduction, you can connect to the *book_keeping* database in the usual way. To get started using the new database, some fundamental MYSQL commands will be shown here. For example, one can reproduce individual columns of Table 2 by executing the following command at the Jlab CUE unix/linux prompt 'mysql -hprimexdb -uprimex_user book_keeping –execute="select distinct type from run_list;"' or to obtain this reult without the tabular divide lines, simply insert '-B' after mysql in the above command. Similarily, one could connect to the database and then execute at the mysql prompt 'select distinct type from run_list;' to achieve the same result. To reproduce the *radiator* column in Table 2, one could execute the command 'select distinct radiator from run_list;'. And the following final example will produce your own run list (stored in a file named 'my_run_list.txt') which satisfies

your analysis criteria: From the unix/linux prompt, execute 'mysql -B -hprimexdb -uprimex_user book_keeping –execute="select run from run_list where target='LEAD' and radiator='B"' > my_run_list.txt'.

5. Future Development

There are two main subjects for this section: The run_list table design and data-entry are not complete, and a new table for intra-run information needs to be developed. In time, after more scrutiny is applied to the run_list details, the table will become more accurate. There are still, however, some design issues with the table: A *converter* field should be added, and an additional *target* value distinguishing the two different empty targets (big-blank, and empty-frame). In addition, there may be a need to track other information in this table; for example, pair spectrometer magnet current and polarity, and electron beam energy.

For the completetion of this database, at least two more tables need to be developed: one for tracking information on an event-by-event basis (e.g. beam trips, current, and position, as well as livetime etc.), and another for tracking bad hardware (e.g. dead ROC, dead TAGE crate, TAGGER_MAC, TS, etc.) and bad counters (e.g. no ADC, no LMS, no dynode, dead, or strange gain or yield).