Stanford Linear Accelerator Center

Index Panel

Stanford Linear Accelerator Center

Infrastructure: The (Now Standard) "Bound Check" SCP

by Ed Miller

Users of the SCP in the MCC control room have, until a couple of months ago, been presented with an invitation when starting a SCP to run either the "Bounds Check" SCP or the "Standard" SCP. This choice had been offered for over a year, but is no longer. What was this all about?

"Bound Check" refers to code (mostly FORTRAN code and C code, in our case) which has been compiled in such a way that if the running program makes an array reference outside the declared bounds of an array, then the program will report the problem and abort, rather than obliviously continuing. All the code in the Bound Check SCP had these run-time checks, whereas none of the code in the old standard SCP had the checks.

This conversion project started well over a year ago with the identification and repair of coding practices that would 'obviously' fail with run-time bound checking enabled. For the FORTRAN-literate, these 'obvious' problems were mostly dummy array arguments dimensioned (1) that needed to be changed to (*). A similar problem also existed with array elements of structures which were declared dimension (1), which require a different fix. There were over 300 routines (of about 6500 total) requiring these pre-emptive fixes.

The remaining bound violations were found by running the SCP, exercising it, and having it abort. The first 50 or so such problems were found by a cadre of software 'monkeys' pressing buttons, and the remaining 150 or so (to date) by the 'real' users, as the Bound Check SCP was made available for more general use. (And now, we celebrate the year of the Monkey.)

By November 2003, after about a year of this in-the-field testing, the rate of newly-revealed problems finally dropped to less than one per fortnight, and nearly all previously identified problems had been fixed. So, at that time we declared "Mission Accomplished" [whatever happened to that banner?], and the Bound Check SCP replaced the standard SCP.

Of these hundreds of coding errors (or sloppiness) uncovered in this 'already-tested' software, only a handful were causing 'real' problems of significance. But now we have this useful checking working for us forever, and it will certainly reveal more such real problems sooner than would the alternative (oblivious) approach. The cost (in execution time) is very small -- probably no more than 1%.

If _your_ SCP (on node MCC) crashes, it now almost always will write a dump file. Please don't sabotage this dump operation, which may take a couple of minutes to complete. And, especially if you find such crashes annoying, please submit an ARTEMIS report. This report is most useful if you can reproduce the problem, but it is even useful if you can supply only a vague indication as to what you were doing to 'cause' the crash. In many cases the dump by itself is sufficient to identify and correct the problem, but not in all cases.

Volume 1, Issue 1

30 January 2004

Inside this issue:

Infrastructure: Realm Split Progress and Meaning

by Terri Lahey

The realm split is a project to separate PEP/LINAC controls from NLCTA/8Pack controls. The main goal is to support the accelerator programs on their own computers. This will remove possible impact by one program on the others, and will allow us to maintain computer applications, operating systems and hardware when there is a downtime for a given accelerator program.

We are dedicating the OPI00GTW0* computers to either PEP/LINAC or NLCTA/8Pack. Multiple people from ESD software contributed in moving NLCTA/8Pack applications to NLCTA/8Pack computers. Work remaining includes moving NLCTA/8Pack IOC boot, cmlog (error logging) and Matlab from PEP to NLCTA OPI000GTW* computers.

The recently commissioned NFS file server is an integral piece for supporting the realm split. Now that the local NFS server is implemented, we can proceed with the remaining work when time is available for intrusive systems work.

Special points of interest:

- *Briefly highlight your point of interest here.*
- *Briefly highlight your point of interest here.*
- *Briefly highlight your point of interest here.*
- *Briefly highlight your point of interest here.*

Infrastructure: Configs (DDRF, 8-Pack) to Back Up Epics PVs

by Nancy Spencer

Infrastructure: DB Install Procedural Changes

by Ken Underwood

The control system database installation (DBINSTALL) procedure that was originally developed for the SLAC Linear Collider (SLC) requires modifications to maintain PEP colliding beam operation throughout the DBINSTALL process. This article describes some of the challenges and opportunities to minimize the impact of a DBINSTALL on PEP colliding beam operation.

The PEP beams have important dependencies upon the control system which the SLC beams did not have. The Summary Information Process (SIP) monitors many control system components and can initiate machine protection actions based upon state changes. The tune feed forward process monitors stored beam currents and adjusts the ring betatron tunes accordingly.

Since all control system processes must be stopped prior to taking down the control system to install the new database, these colliding beam dependencies are not functional until after the micros are IPLed. Fast feedback processes that keep the beams in collision are also not functional when the micros are IPLed until they have received their new database, timing, BPM calibrations, and feedback setups.

Only critical control system processes can be started until most of the micros have been IPLed to prevent unessential processes from trying to talk to micros that are not yet functional. The length of time that the control system processes are not running must be minimized. In addition the PEP micros should have IPL preference over other micros to get the fast feedback loops functioning as soon as practical.

The following changes to the DBINSTALL procedure are currently being evaluated:

1. Prior to the DBINSTALL, the tune feed forward loops should be turned OFF to prevent sudden betatron tune jumps.
2. When the control system is back up to minimum functionality, IPL micros MP00, PR00, and PR02.
3. When these micros are functioning, IPL micros PR03, PR04, PR06, PR08, PR10, and PR12.
4. When all of the PEP micros are functional, IPL the rest of the control system micros.
5. When most of the remaining micros are functional, start the rest of the processes that comprise the control system.

The micro IPLs are currently managed using “Private IPL Lists” that are available off of the “Micro Diagnostic” touch panel, but will eventually be implemented in software.

Using Old Public BPM Calibrations

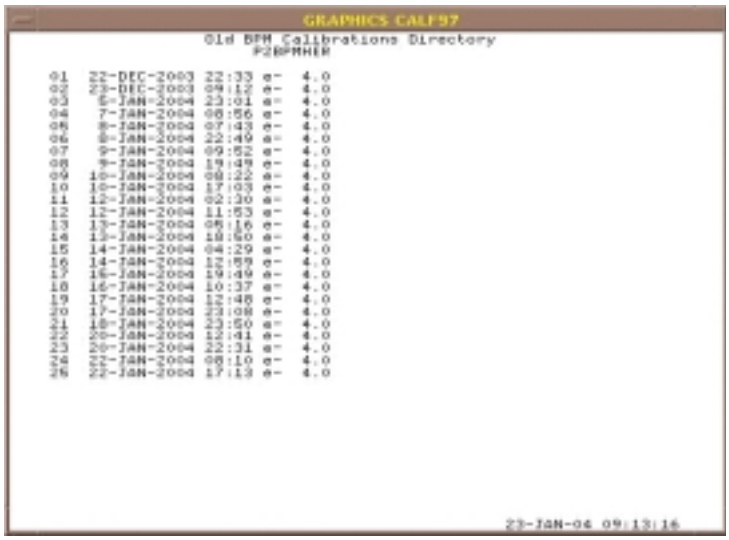
by Mike Zelazny and Ron MacKenzie

Two new buttons have been added to the BPM Calibration SCP panel to:

1.) list the available previous public BPM calibrations; and
2.) load any one of the previous public BPM calibrations as a private BPM calibration.

After selecting a BPM definition, pressing the

button you will get a list of available previously public BPM calibrations that you may use. The display looks as follows:



Previously public BPM calibrations available for use

Listed on this display are the dates of the BPM calibration files, the particle type this BPM definition uses, in this case electrons, and the beam strength used when the original calibration was made public. Public BPM calibrations are kept for roughly one month, but there will always be at least three available. If you would like to try reading BPMs using any one of the above saved calibrations, make your own private BPM definition by pressing the button

LIST OLDPUB CALIBS

then any one of the five blank buttons just to the right

CREATE MEAS DEF					

so that one previously blank buttons looks like this example:

PP= 10
P2BPMHER

PR10 *
PR10
e- B1

Then press the LOAD OLDPUB CALIBS button to load the calibration as a private calibration for your SCP only. You will get

Enter index of file to load as new private BPM Calibration [default = 24 which is the next to last]

The index is the number just to the left of the date on the above SCP graphics display. After entering an index you will notice that the SCP touch panel will go from indicating

Currnt Cal is PUBLIC to Currnt Cal is PUBLIC

You may then read BPMS in the usual way. If you don't like the current public BPM calibration you may make your now private BPM calibration the public BPM calibration in the usual way by pressing the SCP button. If you choose to do so you will see a new entry in the SCP graphics display above for the new public BPM calibration. Online help is available for these two new SCP buttons.

SCP Button	Online Help
LIST OLDPUB CALIBS	After selecting a BPM definition and after putting the BPM definition on a CREATE MEAS DEF button on the bottom row of this panel, this button displays a list of available previous public BPM Calibrations that you may want to reuse.
LOAD OLDPUB CALIBS	Load an old public BPM calibration as a private BPM calibration. After selecting a BPM definition and after putting the BPM definition on a CREATE MEAS DEF button on the bottom row of this panel, this button loads any one of the public BPM calibrations listed when pressing the LIST OLDPUB CALIBS button as a private BPM calibration for your SCP. After trying the loaded private BPM calibration you may choose to make it the current public BPM calibration by pressing the MAKE CALIB PUBLIC button.

Automatic PEP2 BPM Calibrations

by Mike Zelazny and
Ed Miller

Summary of New Feedbacks for the Last Year

by Karey Krauter

New E-Log/Automatic Entries Now from Software Releases

by Spencer Clark and Bob Hall

There is a new Electronic Logbook in the wings at MCC. The new E-Log is an extension of one implemented at Jefferson Laboratory, and provides multiple Logs, each with its own constituency. It is a distributed network Log that can be accessed across the network.

The E-Log has been extended by its original developer Theo Larrieu and our Bob Hall to allow multiple logs to have their own authorized authors, but allow readers to examine any log or combination of logs across the network. There is a facility for automatic script driven data entry for logging common events and important periodic data. Soon there will be a facility for insertion of common text form information, such as the shift change forms.

A number of Groups such as the PEP Low Level RF Team, the 8-Pack Development Group, Test Accelerator, and Software Engineering have commenced using this Logging System. Tools are available for converting log entries from the earlier prototype E-log for those who would like to move to a supported and backed-up facility.

An important advantage of the new E-Log is the ability to attach multiple graphics attachments to an E-Log entry. Graphics attachments may be generated from SLAC VAX/VMS MCC systems, UNIX systems, and Windows systems. This is accomplished through the "print queue" mechanism where graphics images are "printed" to special E-Log print queues. The images then appear in a "thumbnail" list of graphics files that may be attached to a log entry. Also Windows graphics files may be attached directly to a log entry. Log entries and associated graphics files are stored in an Oracle database.

Contact Bob Hall (3409) in SWE for detailed questions, help in getting started, or converting text and data from other log systems.

Bunch Injection Controller IOC Enhancements

by Ron Chestnut

The BIC and Bunch Current Monitor hardware are now in an environmentally controlled (and clean) rack. Alan Fisher and Mark Petree planned and carried out that work, in co-operation with the ESD and SCS networking people. The prime motivation was phase stability, which has been achieved. The other hardware enhancements are the addition of a terminal server to provide remote access, IDOM-driven resets for the BIC IOC and the Kiethly DVMs used for reading out the total ring currents (DCCT), and an upgrade of the BIC from a 68040 (MV167) to a 68060 (MV177) processor. We expect soon to move to a Power PC processor with substantially more memory and faster CPU.

Alan and Mark also added RF circuitry to the LBL-supplied downconverter to allow the collection of data at four phases in the RF, separated by 90 degrees. This "4-Phase" setting has contributed further to the stability of the Bunch Current Monitor phase and amplitude data. This hardware mode is supported by new software which reads out the combined "cosine" and "sine" phases. In addition to having a much more stable per-bucket amplitude, the per-bucket phase is also available now for setting the optimal overall phase. The "PHASES" display available off the EXPERT dispatch panel gives an overview of this information. The (very) small "More" buttons on the far right of the PHASES panel give even more detailed information for selectable buckets.

The single largest project in the BIC area has been the support of continuous injection, or "Trickle Charge". This involved some major surgery to the injection logic.

- Several additional states were added to the BaBar/PEP injection handshake mechanism to allow this hybrid state, a mixture of Stable Running and Injection. Most recently the BaBar Trickle Management status was added to the lower left of the main BIC panel, indicating ALLOW or BLOCK. Software takes no action (this blockage is a hardware mechanism), but operations now knows why trickle injection is blocked.
- A new display to provide some diagnostics on trickle charging is the per-bucket array of injected charge, available from the EXPERT dispatch panel for LER or HER as "All Requested Charge" or "Expanded Requested Charge".
- Allow new total ring current (part of goal setting) to take effect immedi-

ESD Software Engineering Group

Stanford Linear Accelerator Center

2575 Sand Hill Road, MS 46
Menlo Park, CA
94025-7015

We’re on the Web!
http://www.slac.stanford.edu/grp/cd/soft/indexpanel/index.html

Bunch Injection Controller IOC Enhancements (cont.)

to take effect immediately while trickling.

- Allow more per-bucket overfill than originally specified for PEP-II injection.
- Spread out low trickle rate evenly over each second (add NO-Ops).

Several small software enhancements have been made in response to operations needs:

- Stop Injection on Beam Loss
- Automatic BPM Calibration support (suppress injection while calibrating)
- Smooth injection transitions over intermediate goals