



# Meeting Minutes

<i>date:</i> September 28, 2006	<i>subject:</i> <b>LCLS Controls Software Meeting</b>
<i>from:</i> Doug Murray x2235	
<i>department:</i> LCLS Controls	

---

<i>Linac Coherent Light Source</i>	•	<i>Stanford Linear Accelerator Center</i>
------------------------------------	---	---

---

## Attendees:

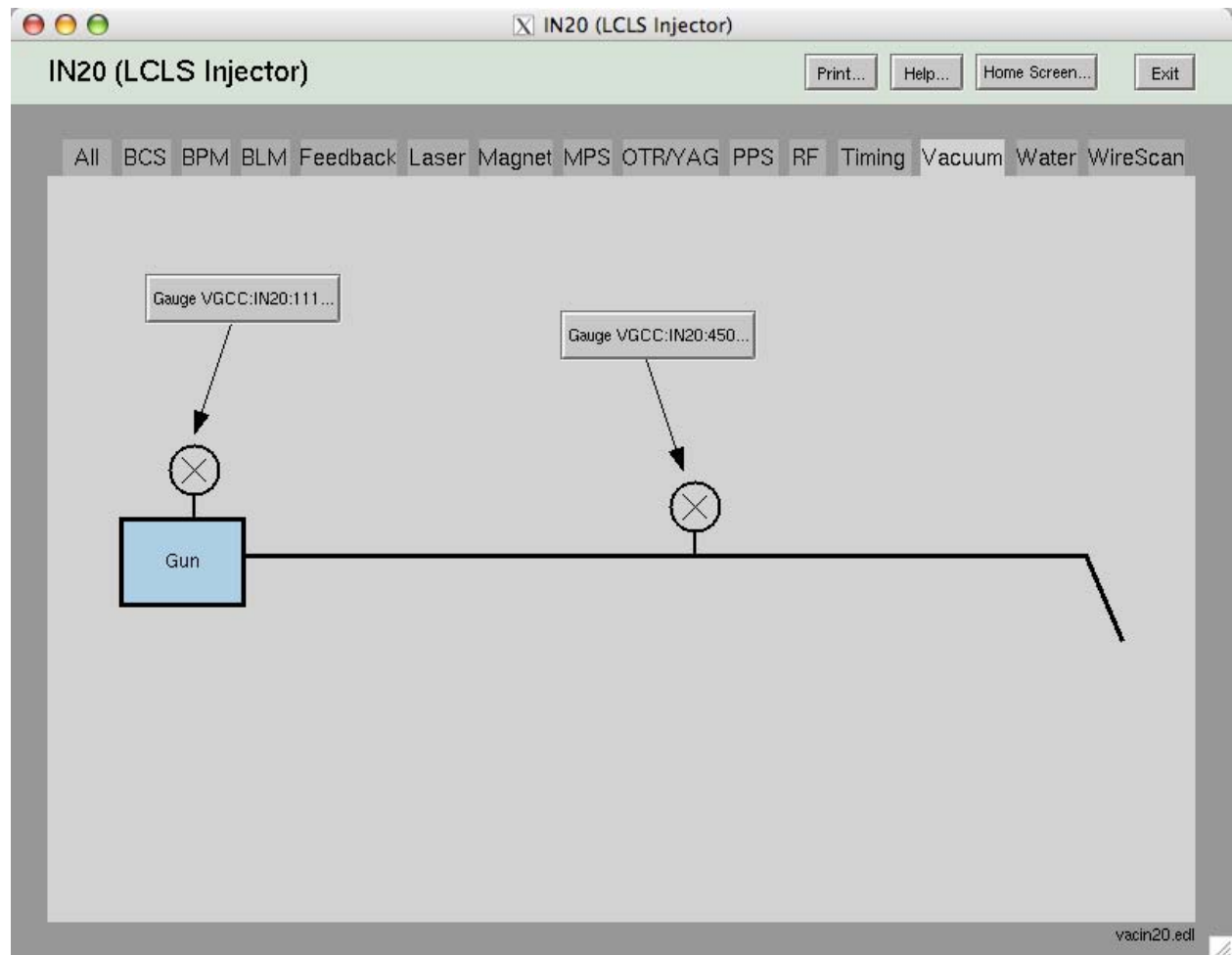
Arturo Alarcon	Dayle Kotturi ( <i>absent</i> )	Debbie Rogind
Diane Fairley	Doug Murray	Hamid Shoaee
Kristi Luchini ( <i>absent</i> )	Mike Zelazny	Sergei Chevtsov
Sheng Peng	Stephanie Allison ( <i>came late</i> )	Stephen Norum
Stephen Schuh	Steve Lewis	Till Straumann
Ron Chestnut	Terri Lahey	Jingchen Zhou
Judy Rock		

## Agenda:

1. EPICS Display Screen discussion. Stephen Schuh and Stephen Norum will show some recent work and template for common OPI screens.
2. Proposed EPICS installation changes for LCLS.
3. A Discussion of the Software Build Environment.

## Minutes:

1. Stephen Schuh described recent work on some display templates done by himself and Stephen Norum.
  1. He showed a simple start up screen, containing one list of accelerator regions and another list of subsystems.
    1. The discussion started with comments relating to that screen. It was suggested that other lists be added, for high-level applications and other support tools.
    2. Stephen said that wouldn't be a problem, but this particular screen was created only to navigate to the templates.
  2. He went on to show an example of their template screen.



3. He pointed out that the header at the top of the window and the border surrounding the area in the center belonged to a common startup file for EDM.
  1. Subsystem-specific EDM files would fill the area in the center. The tabs, help buttons and other navigation aids are drawn by the initial file.
  2. Developers need merely set up their EDM files in a standard directory using the filename used by the startup file. He pointed out the expected filename is displayed on the lower right corner of the EDM window.
  3. He also suggested that we used the idea of including an ellipsis (...) on each button or widget that brings up another screen.
4. There was some discussion about the practicality of graphic displays in the control room.
  1. Ron pointed out that operators currently get things done very quickly because of the consistent position of buttons on the SCP screens.
  2. Hamid agreed, and said their actions were similar to touch-typing. He suggested the nice graphics might "get old" quickly.
  3. Doug said that a previous meeting's discussion had indicated new operators could take up to a year to learn to use the existing control system. He said this would help that situation, and nothing was stopping other displays from containing more complex non-graphic screens for experienced users.
  4. Stephen Schuh also suggested that he, as a recent operator, could imagine a mix of graphic and textual screens to be useful, especially for specific tasks that aren't performed often.
5. Steve Lewis said the idea of Tabs was an important addition to EDM that he thought would be very useful.

1. There was a brief discussion about how tabs were quite common in most modern GUI environments.
2. Steve insisted that a user-programmable tool like EDM was preferable to a general programming language.
3. Doug suggested that a modern programming environment with EPICS support could provide a better environment for the user if done properly; tabs and other modern widgets would be inherently available.
4. Ron suggested there was no merit in talking about developing a new EDM-type tool.
5. Hamid agreed, and indicated there have been many versions of display managers for EPICS over the years, which have failed.
6. The discussion then turned to some EDM-specific topics.
  1. We discussed aspects of colors. Stephen Norum said these colors were chosen to keep the screens simple, and make colors more pronounced. Few colors were used; most of the major elements are grey.
  2. There were questions about new colors being put in the **colors.list** file.
  3. Doug mentioned that updating and managing the **colors.list** file was cumbersome because of the file format. Also, current colors are specified as 16-bit values for each of the RGB channels, and were currently written as decimal values rather than hex.
    1. Steve Lewis said that when the colors were originally created, he used a separate tool to create them. Doug said that would probably be the best idea for creating our own color scheme, if we wanted to do so.
  4. Ron mentioned that some forethought and care must be taken in setting up alarm and severity colors for EDM calc records and color rules.
    1. Judy said that she had recently been working on adjusting some of the custom color rules for PEP, and agreed that it requires more overall design work up front.
    2. Ron said the PEP rules might be too complex for our needs.
    3. Steve suggested that these rules might be better designed into the database. All clients would then benefit, not just the local EDM clients using those rules.
    4. Ron said it was a design decision that would have to be made.
    5. Sheng said he has previously used **gencalc** or **calcout** records and made those decisions in the database.
    6. We agreed to try and limit the determination of severity and alarm rules to the database whenever possible.
2. We then discussed some proposed changes to the EPICS installation layout for our development work.
  1. Doug presented 3 issues of concern in the current build environment.
    1. One cannot access individual packages under *site* if one has a local copy of a different *site* package.
    2. Updates to new versions of EPICS can be cumbersome.
    3. We don't keep adequate version information on-line.
  2. Doug suggested a change to the RELEASE file in each package would fix the first problem.
    1. Jingchen said he had developed a workaround to the problem.
    2. Doug recommended the underlying problem be fixed, but was interested to see the workaround.
  3. He said that Stephanie had suggested (and started to implement) a common include file that would define the basic environment variables.
    1. Sheng said he was concerned that he wouldn't be able to override those variables or do other custom changes.
    2. Doug said that the environment variables would be defined in a single location, but could be done only if they haven't already been set.
  4. There was discussion regarding the version information. It was suggested that one or two versions of each package be kept on-line.

1. Under the package area would be another subdirectory showing the version number. These would be checked out of the repository using the appropriate tags for CVS.
2. Till argued that this would present problems when a package changed; one application might require the update, but another application destined for the same IOC might not be able to upgrade.
3. Sheng said that the issue could be resolved by having the second application upgrade to the new version.
4. Sheng also asked why only 1 or 2 previous versions would be maintained. It was suggested that it would be easier to maintain.
3. The discussion then turned to the build strategy for the operations network.
  1. A recent exchange of e-mail had prompted a discussion of static versus dynamic linking of applications.
  2. Jingchen suggested it would be easier to maintain application servers on the production network if the applications were statically linked.
  3. It was pointed out that these specific production servers should be kept at the same version of OS and development tools as the development server.
  4. Steve mentioned that many other EPICS sites have a dedicated build server. An export of the repository would be done to a local disk on that machine, and applications built there.
    1. Jingchen pointed out that this was not how builds are currently done, and that builds are not typically allowed on production servers.
    2. It was pointed out that the build server, although on a production network, would not be running any controls application software directly. It could be used exclusively for building and as a boot server.
  5. Ron pointed out that keeping proper OS and tool versions on a build server and using dynamically linked libraries represents a large maintenance workload.
  6. Terri mentioned that we could piggyback off SCCS, which uses external vendors to build and maintain software packages.
    1. It was pointed out that there are a relatively small number of packages used by the EPICS software.
    2. Jingchen mentioned that we would need to keep a very large number of packages up-to-date. There is a very large number of packages that would be added to the Perl tool alone.
  7. Jingchen also mentioned that sharing everything via NFS would be preferable, to reduce the amount of maintenance work.
    1. Steve suggested that using **rdist** is a good option; it will synchronize files on different machines to keep them up to date. It's reliable, and is used at a lot of Labs.
    2. There was a question about having all libraries and tools on a single file server, which could represent a single point of failure in a running environment.
    3. Jingchen pointed out that the NFS services couldn't fail because we have redundant servers.
  8. It was suggested that a build server be investigated further, although there was not a clear consensus that this is the correct solution.