

APSU High-Level Engineering Applications

This is a page to start listing, conceptualizing, and describing the set of engineering tools that will be required for the APS Upgrade MBA.

Related Material:

- [MBA Diagnostics and Controls Conceptual Design Review 20-21 March 2014 Presentation](#)

Machine Design/Development/Installation:

- Naming Convention (components and PVs) and tools to promote compliance
- DAQ Services (acquisition, data management, analysis toolkit)
 - How to accommodate "slow data" needed with fast acquisition
- [Component Database \(CDB\)](#)
 - component catalogue, component tracking, machine design, installed equipment, travellers ...
 - Tablet app for relocating components, uploading travellers, etc
- [Technical Systems Data Management](#)
 - Magnet measurement data, calibration data (power supplies, BPMs, etc), BPM configuration, ...
 - Data catalogue
 - APIs to other toolkits (e.g. Matlab, Octave, python)
- Engineering Electronic Logbook(s)
- Simulation of technical subsystems for IOC, display, script development
 - Look into pyDevSup (<http://mdavidsaver.github.io/pyDevSup/>): a python API for writing EPICS device support.
 - Start with RTFB simulation, then move to MBA
 - Need a Virtual MBA (compared with the Virtual LINAC)
- Tools to auto-generate EPICS databases (from the CDB) (and engineering displays?)
 - Includes Naming Derivation Tool that names PVs appropriately
- Process Variable/Channel Finder
- Simulation of accelerator (elagant driven) for physics application development
- Easy to use datalogger for teststands and laboratory tests
 - Originally a CLI client for Channel Archiver data. In the past year it has grown to include a Python API, and support for the SLAC Archiver Appliance.
It also includes a2aproxy, which acts as a transparent proxy between Channel Archiver clients and the Archiver Appliance server. This allows existing Channel Archiver clients to be used with Archiver Appliance.
The source is hosted on github: <https://github.com/epicsdeb/carchivetools> and I've written some brief notes on setup and usage: <https://github.com/epicsdeb/carchivetools/wiki>
SLAC Archiver Appliance: <http://epicsarchiverap.sourceforge.net/>
- Configuration management of all deployed software (update of our current procedures)
 - Includes embedded software and FPGA source
- IOC administration guidelines using procServ (tech-talk message of 3/16/15 from ulrik.pedersen@diamond.ac.uk)
- scripts for expedient work (e.g. daqapps)

Not-so High-Level:

- real-time linux distribution for microTCA, soft IOCs, ...

Machine Checkout/Startup:

- Technical Equipment Checkout/Test Scripts/Tools
- Injection Orchestration (bunch replacement)
- BPM/Orbit server
 - knows about BPM configuration, returns arrays accordingly
- PS Synchronous Setpoints (conditioning, synchronous adjustments)
- Fault Analysis (includes identifying suspicious components, responsible group)
- Background performance monitoring and process/services/IOC monitoring (look for abnormal conditions before they cause a fault)
- caput log (to tell who did what). See <http://mdavidsaver.github.io/pyDevSup/>