Pavilion Tutorial Guide

Ver. 1.0

* Overview of Pavilion
  + Tool to run and analyze the results from an assortment of different tests under one umbrella or framework. Successor to Gazebo.
  + No actual tests, just the framework (python 2.7 based)
  + Open source, available via Git or tar ball
    - <https://github.com/losalamos/Pavilion>
  + Documentation in docs directory in installed location
  + State of various pieces (Project under development)
  + Build – see Build-Notes.txt in docs directory (only needed to create a new tar ball)
* Install procedures
  + Copy tar file to install location and extract
  + User ENV setup
    - setenv PVINSTALL <install location>/pavilion
    - setenv PATH \ ${PVINSTALL}/PAV:${PVINSTALL}/PAV/scripts:${PATH}
  + link pavilion to install location
    - ln –s ${PVINSTALL} pavilion
* Setting up test and the default configuration files
  + Philosophy, yaml files, inheritance and where to place files
  + Example files found in ${PVINSTALL}/docs directory
  + Edit the default Config file
* Attaching tests/jobs to Pavilion
  + Startup (editing the test config file)
  + The ENV variables available (pav show\_env)
  + Output
    - Trend data, TD (see the README.txt file in the docs directory)
    - Review test log directory components
  + Single and multi-test examples
* CLI
  + Using pav and the help levels
  + Running a job – “pav run\_test\_suite ./<test\_suite>”
  + pvjobs – check job status script
  + View the integrated test suite
  + Looking at results
    - What defines a test (name x nodes x PE’s x test parameters)
    - Summary - “pav get\_results –ts ./<test\_suite>”
      * Basic (selected dates or default of 15 days prior)
      * “-T” showing trend data
      * Base-lines (selected dates or default)
      * Box plots (selected dates or default)
      * Line charts (by calendar day)
    - test\_results.csv – TD result synopses in result root
    - splunkdata.log, also in result root, but Splunk format
    - mk-tpl - last 24 hr. using summary, base-lines, failure analyzer
    - jobFailAnal - knowledge base failure analysis tool (\*)
    - blDiff - show delta’s between two baseline files
* Debugging Tips
  + Main pavilion log (/tmp/<user>/pav.log)
  + Job logs (using “-i”, “-f”, “-p” with get\_results sub-command)
  + Moab output files (err and out files in the result directory)
  + User requested saved files
* The “working space”
* Software Architecture
  + Adding new commands
  + DRM support