Initial descriptions for events for EntryPoints. For discussion before we more formally document...

EntryPoints {

~~(**On End** *execute_stmt*)   // finished the test program execution of the DUT~~
~~   // started with On Start.   This event is triggered from~~
~~   // some higher level controlling software~~

Above no longer needed. On End occurs at the completion of the "execute_stmt" for an On Start event. Will remove this event.

    (**On Load** *execute_stmt*)   // event triggered when the test program has been loaded but
                                            // but before other things such as vectors have been loaded.

    (**On LotEnd** *execute_stmt*)   // ended a specific LOT started with OnLotStart

    (**On LotStart** *execute_stmt*)   // started a new LOT to be tested

    (**On Reset** *execute_stmt*)   // starting to reset the tester, & testprogram??
                                                  test program should not care about resetting the tester
                                                  same as OnUnload

For On Reset above, Ajay will investigate usage in his environment and come back to group to help decide if phase 1 or 2

    (**On Start** *execute_stmt*)   // started the test execution of a DUT

    (**On WaferEnd** *execute_stmt*)   // at the end of a wafer under test
    (**On WaferStart** *execute_stmt*)   // at the start of a wafer under test

    (**On <*UserKeyword*>** *execute_stmt*) // a special user specified event has
                                            // occurred   Contact Test, Partial Test,...

   } *// End Entry Points*

Doug to add a sequence diagram here to capture the order with which events are expected to occur.
For now, here is a simple one to discuss this order, before I put something prettier in place. Interested
in your feedback...   Note: On <UserKeyword> could happen anywhere after On Load.

On Load
On LotStart
 On WaferStart
  On Start
 On WaferEnd
On LotEnd

Phase 2 Entry points... put here until we find a better place for them. Note the following paragraph describes out criteria for which entry points were in phase 1 and which were moved to phase 2...

Discussed using the following criteria for which of these events are in Phase 1, and which get moved to Phase 2. The idea is for phase 1, just provide the events to be consumed by the test program to attend to, coming from tester O/S or tester controller, or s/w sitting on top of the O/S which does higher level control of execution of the test program. Also, don't deal with events being thrown from within the test program or flow.   For that kinda stuff, plan to address it in Phase 2, and bank on the fact that any type of event/exception handling at the test program/flow level will be implemented using existing facilities of the ATE test programming language itself.

(**On Abort** *execute_stmt*)   // something bad happened at tester
                                          // high power, current, voltage, smoke,... need to abort

(**On Calibration** *execute_stmt*)   // tester needs to be calibrated, so starting that, typically
                                          // externally initiated, may change timings

(**On Exception** *execute_stmt*)   // some exception has been thrown
                                          // test pgm crash, I/O exception, ... deal with it

(**On MultiSiteDisable** *execute_stmt*) // event to disable Multi-Site Testing??
                                                // which sites enabled?, convey this info?
(**On MultiSiteEnable** *execute_stmt*)   // event to enable Multi-Site Testing??

(**On PatternLoad** *execute_stmt*)   // started loading the test patterns to be run, distinquish
                                              // from OnLoad... needed by the testprogram?
                                              // Patterns loaded on demand as needed

(**On PowerDown** *execute_stmt*)   // starting to power down the DUT... triggered
                                          // from within?
(**On PowerUp** *execute_stmt*)   // starting to power up the DUT

(**On Retest** *execute_stmt*)   // Started retest of the DUT...
                                          // Does an On Start get triggered after this, or does this
                                          // this take the place of an On Start trigger, for when a
                                          // DUT needs to be retested

(**On SiteEnd** *execute_stmt*)   // at the end of a Site being tested, for sequential multi-site
(**On SiteStart** *execute_stmt*)   // at the start of a Site being tested, for sequential
                                           // multi-site

(**On Unload** *execute_stmt*)   // start unloading the test program and patterns

(**On Validation** *execute_stmt*)   // start validating the test program, ultraflex does this
                                          // maybe use UserKeyword since tester specific