iSIMLab Design/Impl Notes

I. Utilize experiences
   a. Navy surface system simulation CSM and development
      i. Navy CPU design team
   b. eTrade, eStore
      i. DevOps design and implementations
      ii. Scalability techniques
         1. Eg. NFS
   c. Intepreted computer languages [Ruby, Python, JavaScript]
      i. Aka, Run-Time compiled languages

II. Design
   a. Step 1 Cfg file – builds project cfg'ation and message list
   b. Step 2 Cfg file – builds communication module hierarchy
      i. Via templates builds 2 comm levels 1. Infrastructure 2. Application
      ii. Plugins
         1. Infrastructure built in plugins, no user action required
         2. Application user defines custom msg/module data & algorithms
   c. Step 3 Cfg file - app custom msg/module data & algorithm plugin
      i. iSIMLab v 4.0 moves Step 3 to Step 0
      ii. Purpose . Allow Step 2b. App build to inject Step 3 app cfg'd data into src module file(s)

III. Modules
   a. Infrastructure modules:
      i. SubjectInfo – Reads EMR and via greyscale creates module Good and Bad[1] cell counts
      ii. SessionMgr – reads in the main loop msgs
         1. Eg. Tick_chain
            a. Tick_GBC – calculate GoodBadCellCnts
            b. Tick_ENV – calculates BadCell cnts by environment
            c. Tick_IV – calculates IV GoodBadCellCnts
   b. Application modules:
      i. Cluster Brain is the root of the human body sim modules

IV. Implementation
   a. iSIMlab team builds initial demo prjs w/Step0 , 1 ,& 2 Cfg files
   b. New users copy demo projects and modify for customized usage
   c. The copy feature auto builds the src prj to a users sandbox and
      i. Runs initial cfg'ation test code verifying baseline
      ii. User can modify Step 0,1 and.or 2 Cfg files and or modules

[1] GBC – is the initial plugin incorporated into iSIMLab infrastructure along with the Step 3 app plugin