iSIMLab Technical Document

v 1.0 March 2025

This iSIMLab Tech Intro Document is available at https://isimlab.cloud/ted-0/

History

In the mid 1980's an effort was started to design a **human body** computer **simulator** for use in cancer research.

Decades later the ideal technology became available to implement its 350 man year effort in a reasonable timeframe.

iSIMLab – General Description

The resulting product is called iSIMLab, short for Internet Simulation Laboratory.

iSIMLab leverages eCommerce technologies to join the Internet community into collaborative teams to build large complex projects initially in the area of cancer research.

Basics building blocks

iSIMLab is composed of 3 basic components: a Portal, an SDK and a Library.

Portals

iSIMLab's portals include: A **Research** Portal, A **Doctor** Portal and A **Data** Portal

SDK

An SDK short for Software Development Kit is the support software tools to implement iSIMLab features.

Library

iSIMLab's library is where projects, drug protocols, and simulator algorithms are stored.

The Macro Details are

Portals

Research Portal

The **Research** Portal is where

- many different human body cancer simulators are built

Dr Portal

The **Doctor** Portal is where

- patient health records are HIPAA filter to become subjectId'd EHRs
- subjectId simulator runs are scheduled and
- results are disseminated

Data Portal

The **Data** Portal is where

- simulation run results can be data mined

Portal Features

iSIMLab portal features include

- Researcher Portal :
  - a library of preBuilt and tested projects, drug protocols and algorithms
    - normally researchers will copy and customize a project for their specific needs
    - customization can include adding/removing/editing the Library
      - drug protocols, algorithm and/or module source components
- Dr Portal : Doctors will
  - search and select a cancer project
  - enter in a patient EHR and be assigned an anonymous subjectId for that patient
  - proxy eSign a patient to subjectId HIPAA form
  - schedule simulator clone runs by matching a subjectId to a project where
    - each clone runs a different drug protocol
  - view run results
- Data Portal : Researchers can data mine simulator run results

SDK

Plugin architecture

Plugins allow applications to easily add custom features.

iSIMLab's SDK is built upon a plugin architecture. The initial plugin created is the Human Body Simulator Good Bad Cell algorithm.

This plugin provides the following infrastructure library modules: SubjectInfo, SessionMgr, Console and Prognosis.

The SubjectInfo module reads EHR and creates each module tissues good and bad cell counts.

The SessionMgr modules reads and executes the session command file which contains the drug protocol/algorithm instructions.

The Console module is created dynamically and executes the POST for all module command combinations, initializes all modules and then runs the session drug protocol command.

The Prognosis module reads each clone's Good Bad Cell tissue counts and creates a mathematical predicted 5 year survival and QOL comparison matrix.

Library

iSIMLab's Libray is sorted globally via a 'fully qualified name' format is as follows:

`UserId`_`componentTypeAbbreviation`.`Descriptor`_`Version #`

Global FQN DB name format:

Component	gName	Description
projects	cci2_prj.sHBS_0.0.0	Heart organ
module	cci2_mods.Heart_0.0.0	Heart organ
Command /message	-	Unused
plugins	cci2_pisDP.GBC.F5FU.sHBS_0.0.0	Each module command/message may have a custom plugin algorithm
session	cci2_scs. sHBS_ 0.0.0	Sessions contain main loop drug protocol commands
doctor	drwelby_dr_0.0.0	DoctorId. With patient to SubId tags
subject	drwelby_subId.00001_0.0.0	SubjectID: EHR records
batch	drwelby_batId.00001.subId.00001_0.0.0	Batch Run Request Data
EBLibrary	cci2_EBLib..sHBS_0.0.0	Project list of all FQN components

cci2 is iSIMLab's administrator UserId. 'cci' is short for Code Collaborator Interface which was iSIMLab's original name

Simulator

iSIMLab users create a unique UserId.

Research Portal

Each iSIMLab researcher logs in with this unique UserId and normally copys an existing project which creates a unique sandbox area that will be customized for specific areas of study.

Simulator Build Input

The copy project will contain the following key data elements:

- a project name eg. sHBS –is an iSIMlab demo project
- a module build list eg. Brain, Heart, Lungs, ...
- a command/message list eg. InitComm, InitSubjectInfo, ...
- optional list
    - session command/message/drugProtocol list

Simulator Build

iSIMLab's copy feature builds a unique UserId sandboxed project by

1. Checking out Library components
2. Dynamically building the 'Console' main loop program which
    a. Spawns simulator clones each running a unique drug protocol
    b. Each clone program
        i. Runs a communication infrastructure POST
        ii. Calls the 'Init' prefixed commands initializing the system and
        iii. Calls the SessionMgr which invokes the sessionCmd drug protocol main loop commands

iv. At the end of the all clone sessions the Prognosis modules is invoked to
    1. Create a comparison matrix of the mathematically predicted 5 year survival rate and quality of life of each clone

Simulator Build Output

The iSIMLab simulator build creates

- a unique UserId/projectName file system sandbox
- a log file
- a prognosis report

iSIMLab Architecture Description

iSIMLab Infrastructure Architecture

Simulator aka aSCC, Automated Software Conference Call

Conference Call

iSIMLab' simulator build and run is designed along the lines of a telephone conference call where

1. iSIMLab is the calls host and schedules
    a. time of the call
    b. invitees
    c. subject matters
2. the iSIMLab copy project feature is the host who
    a. uses the copied programs configuration: UserId, projectName, module list, command list, and clone configurations to
        i. create a unique file system directory sandbox
        ii. check out cfg'd components from the Library building a copy of the original program and then
        iii. invokes each clones 'Console' module where
            1. a POST, power on self test, for all module command combinations completes a 100% communication test coverage
                a. Like a telephone conference roll call
            2. All 'Init' prefixed commands are invoked initializating the application
            3. The main loop session drug protocol commands are called
3. the build and run establishes a successful operational copy into the new users sandbox
4. The user may now update the project for their specific needs

iSIMLab Application Architecture

Plugins

An iSIMLab application plugin allows the application to easily configure customized algorithm functionality based upon each modules command.

iSIMLab Plugins

iSIMLab's plugin architecture facilitates the following features.

1. At the SDK infrastructure level 1 the iSIMLab team creates system functionality
    a. Currently 1 plugin called sHBS GBC exists
        i. sHBS.GBC short for human body simulator good/bad cell algorithm
2. Level 2 is at the project level where
    a. Library module, sessionCmd and algorithm components can be
        i. created, swapped, edited or deleted

Business IP protection

iSIMLab's is designed to run on separate public and private infrastructures.

The private infrastructure protects IP, Intellectual Property, by utilizing an ACL, Access Control List, that restricts access to only authorized parties.

Evolution

iSIMLab as a product will be developed in stages.

iSIMLab stage 1 shall be developed by the iSIMLab team & include the following

- a public internet infrastructure
- a library with the following components
    - a set of demo projects
    - a set of session command drug protocols
    - a small set of plugin algorithm components
        - Human Body Simulator, GoodBadCell algorithm implementation
        - Clinical InVitro
    - A set of programs that creates a transform interface
        - between non-native computer languages to iSIMLab's computer language
        - between MATHML and iSIMLab's computer language

Stage 2 shall be where the iSIMLab team will search public medical repositories and import the following into the iSIMLab Library

1. Source code modules
    a. Using the above transform interface program to convert to iSIMLab's native language
2. Data modelers
    a. Custom algorithm creation is required here to port functionality creating a new iSIMLab plugin

Stage 3 shall be where the public shall be able to import components into the iSIMLab Library

Project evolution

As the iSIMLab's Library becomes populated projects will branch and mature by choosing newer and newer modules, drugProtocols and/or algorithms components.