



Global CyberSoft

OpenELIS Architect System

By: **GCS**

Date: Sep 21th, 2009

Duration: 3 hrs



Global CyberSoft

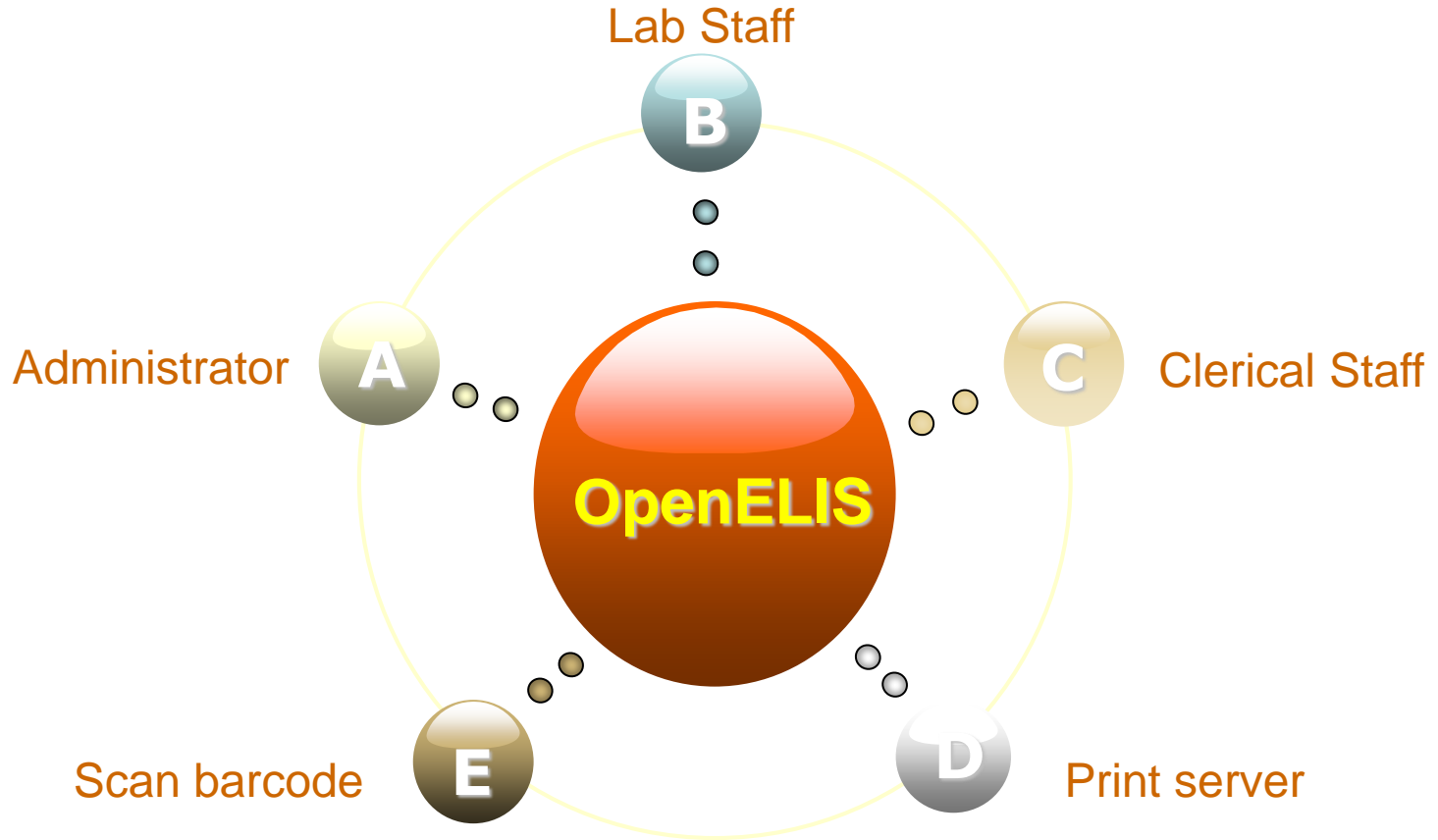
Contents

- ✚ **Constructor**
- ✚ **Module interfacing**
- ✚ **Application and Component Architecture**
- ✚ **Documents reference**



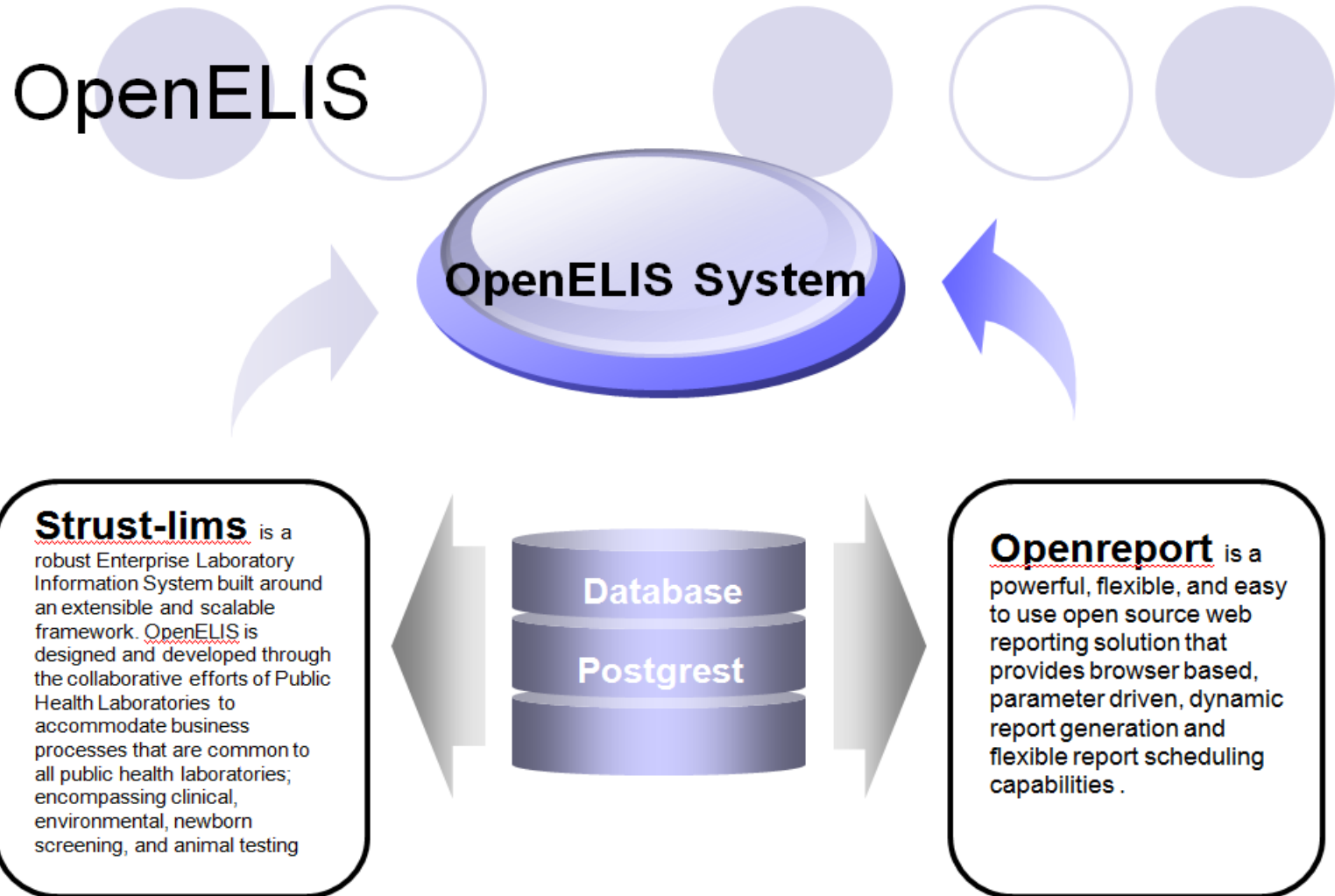
Global CyberSoft

OpenELIS viewer



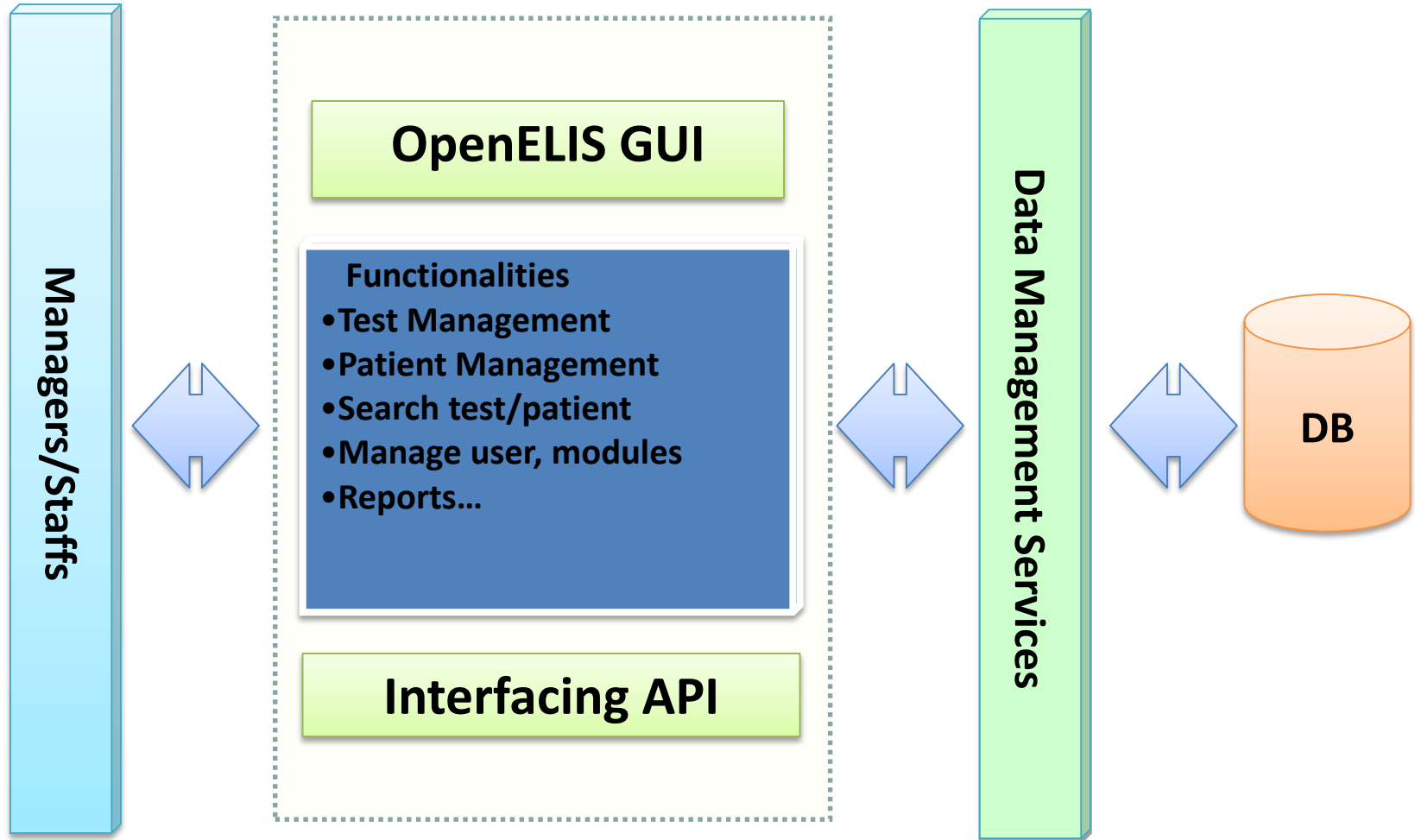


Module interfacing



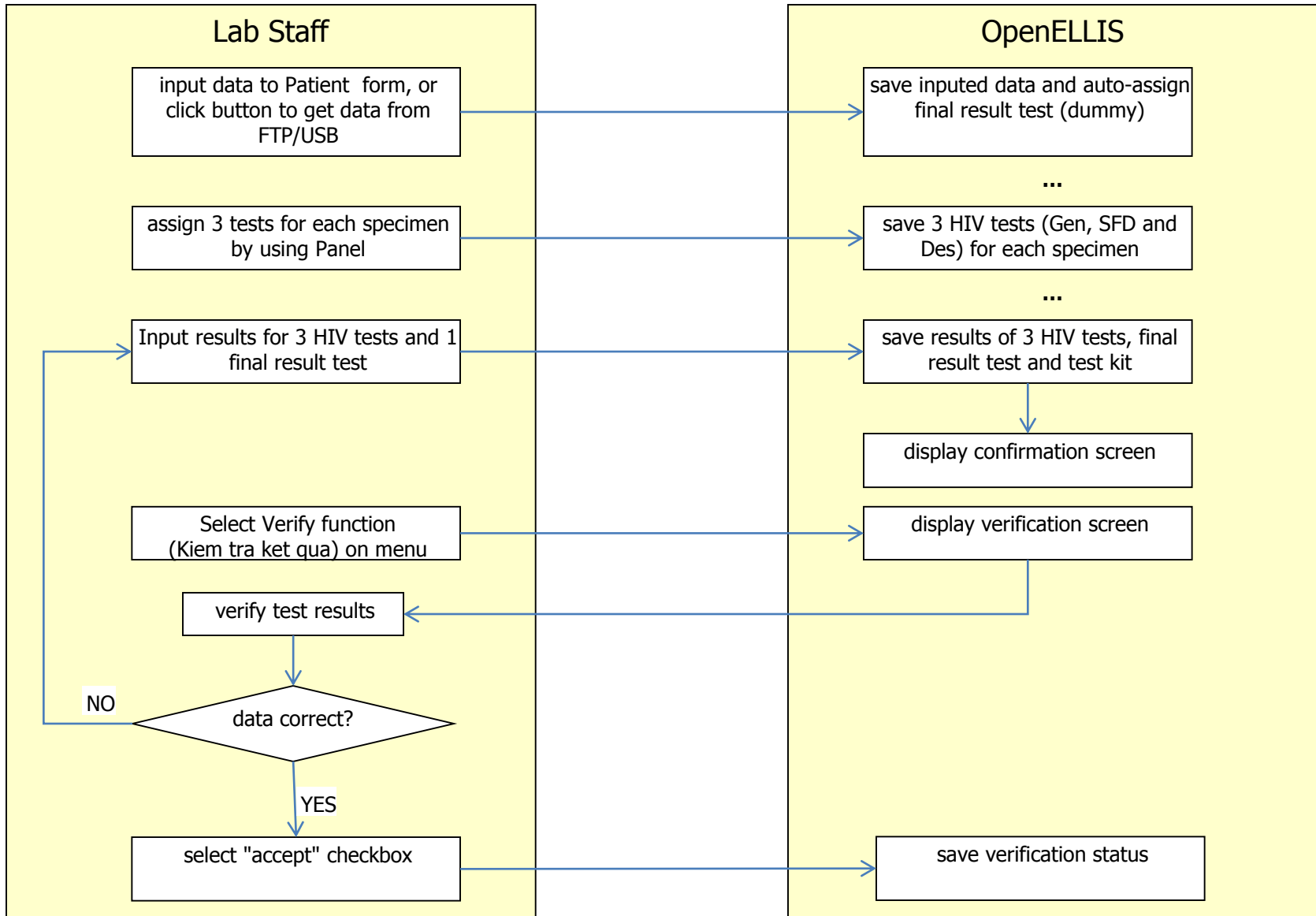


Timesheet Management System – Overview



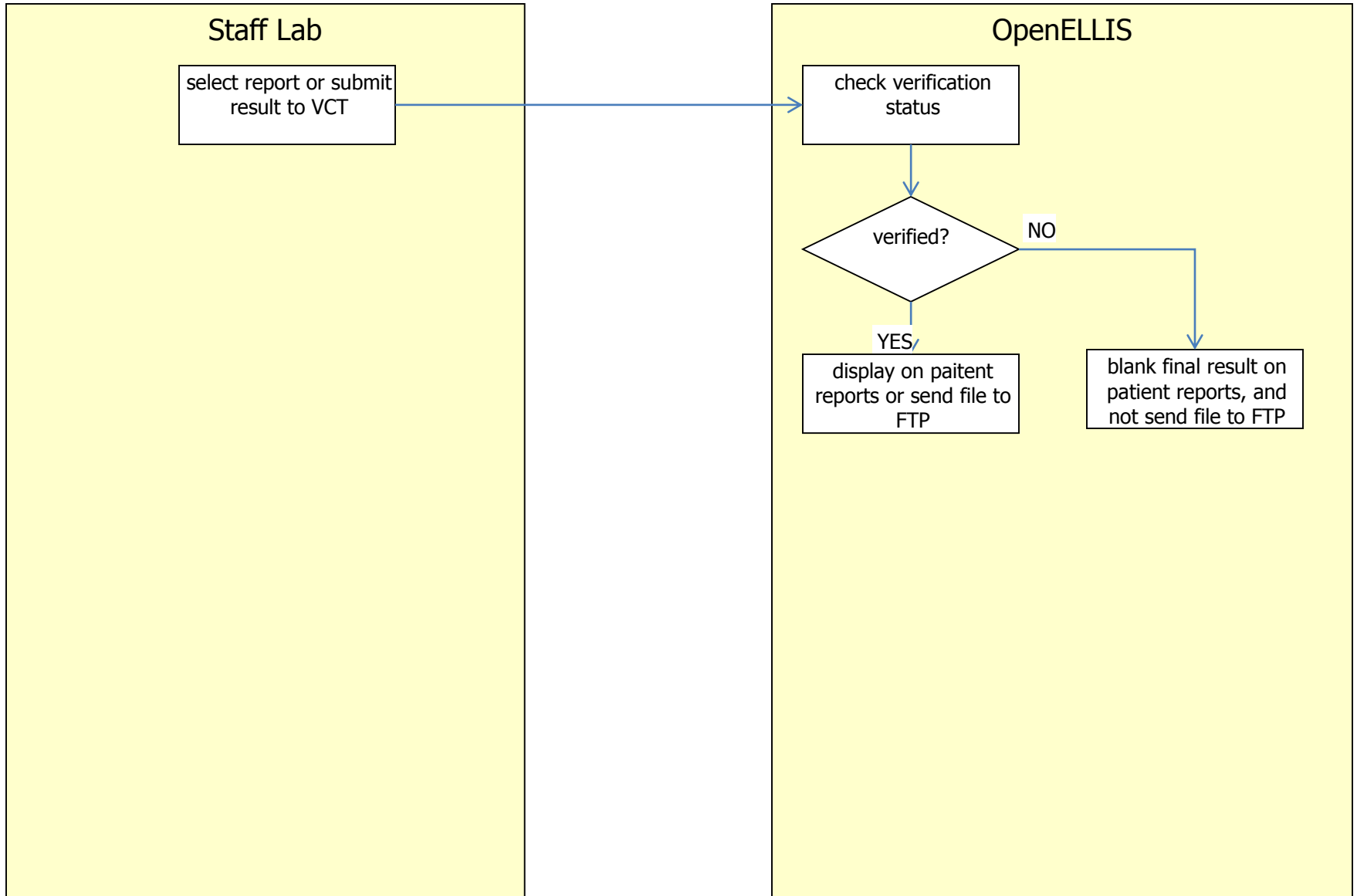


Module interfacing





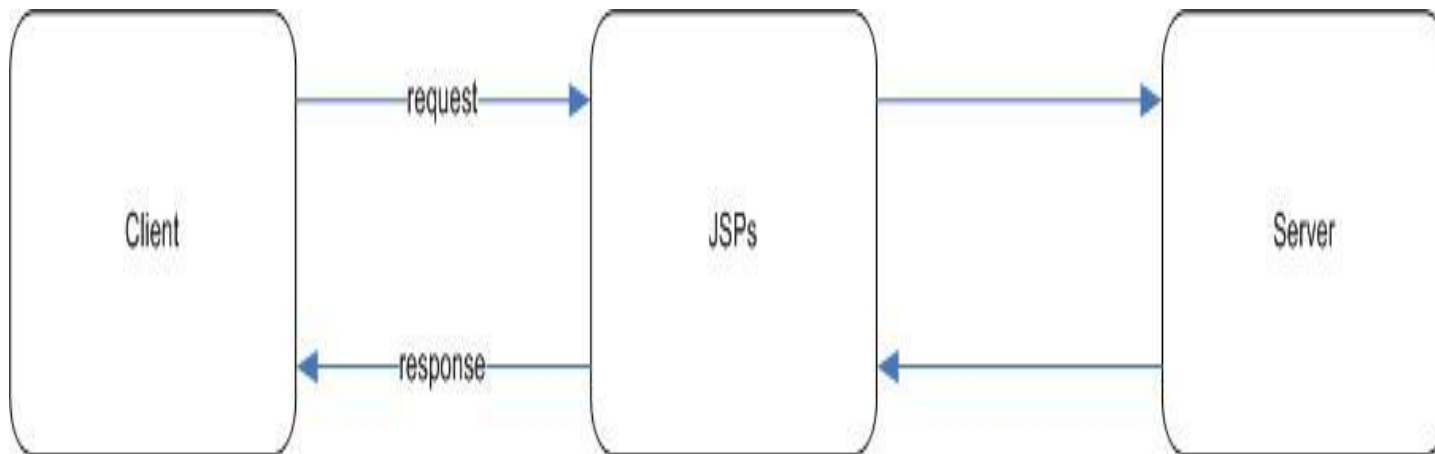
Module interfacing





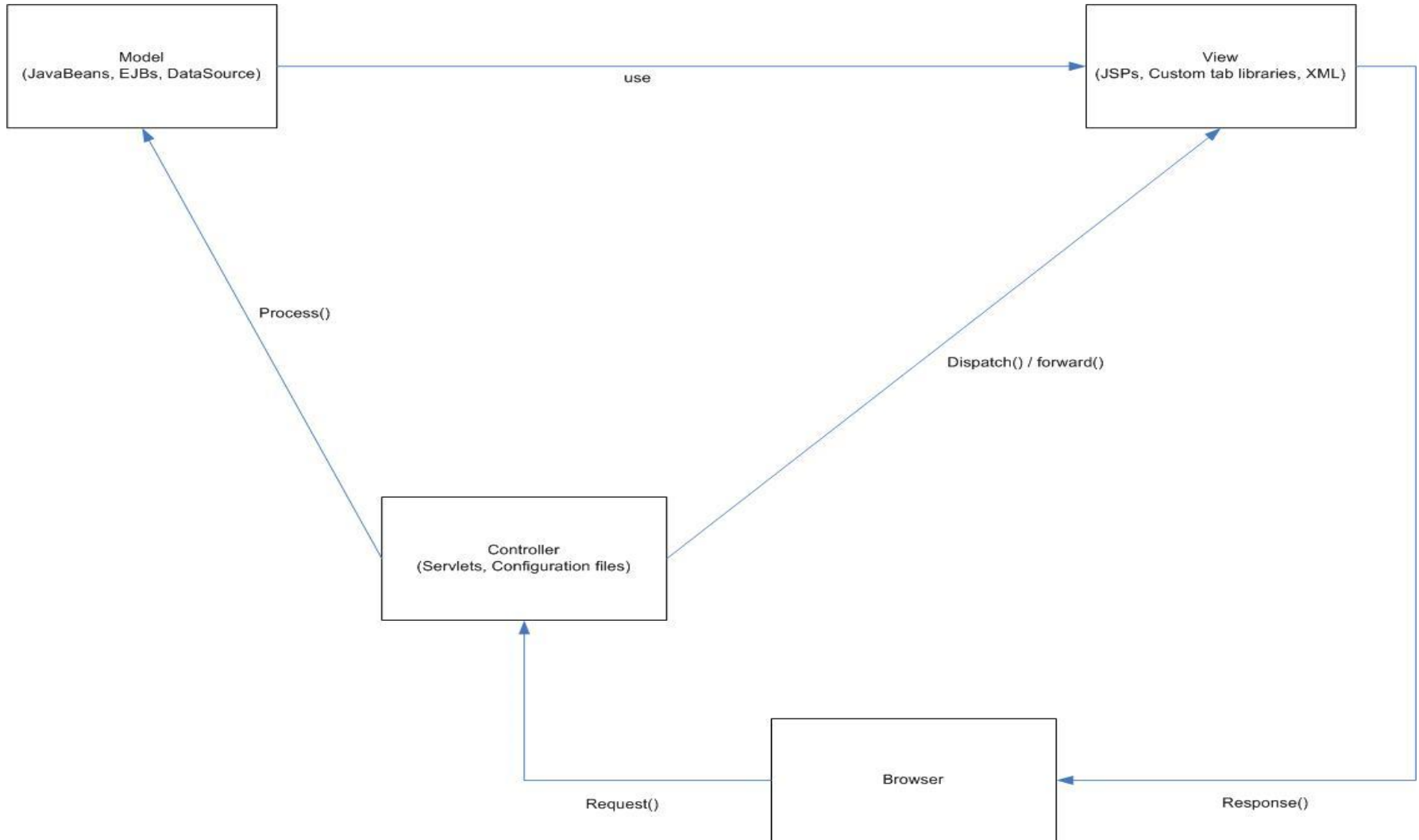
Application and Component Architecture

- ✦ **This is a classic client-server approach. The client sends a request to the server, typically a Web browser and a Web application server. The server then accepts the request and interprets the request. After computed the request, the server sends response back to the client**
- ✦ **This approach is the simplest method of requesting and sending data between client and server but it reduces modularity and minimizes reusability. As the application gets more complex, developers found that they have embedded a significant amount of Java code into their JSPs that would be very difficult to maintenance and error prone**





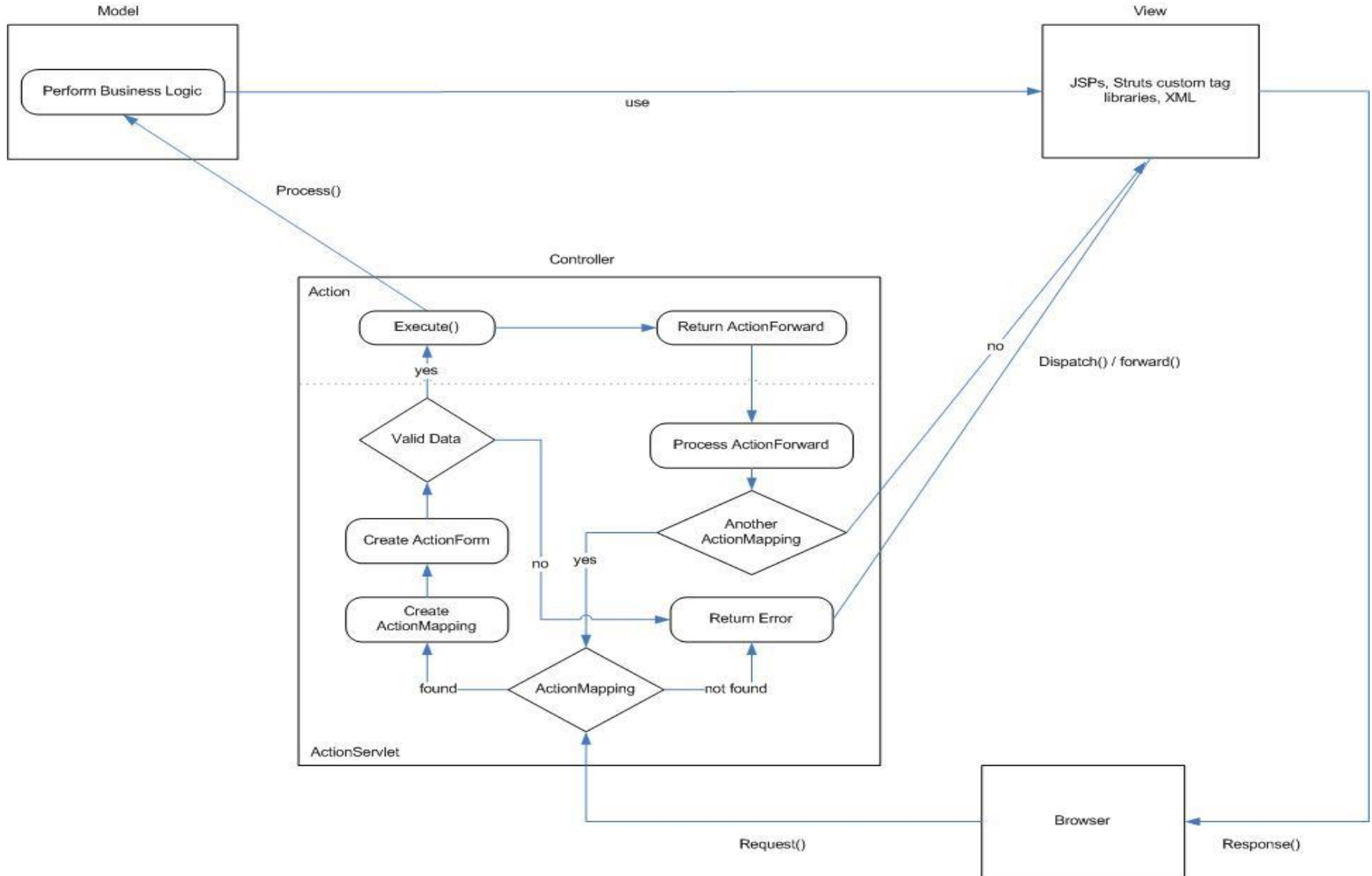
Application and Component Architecture





Application and Component Architecture

Global CyberSoft





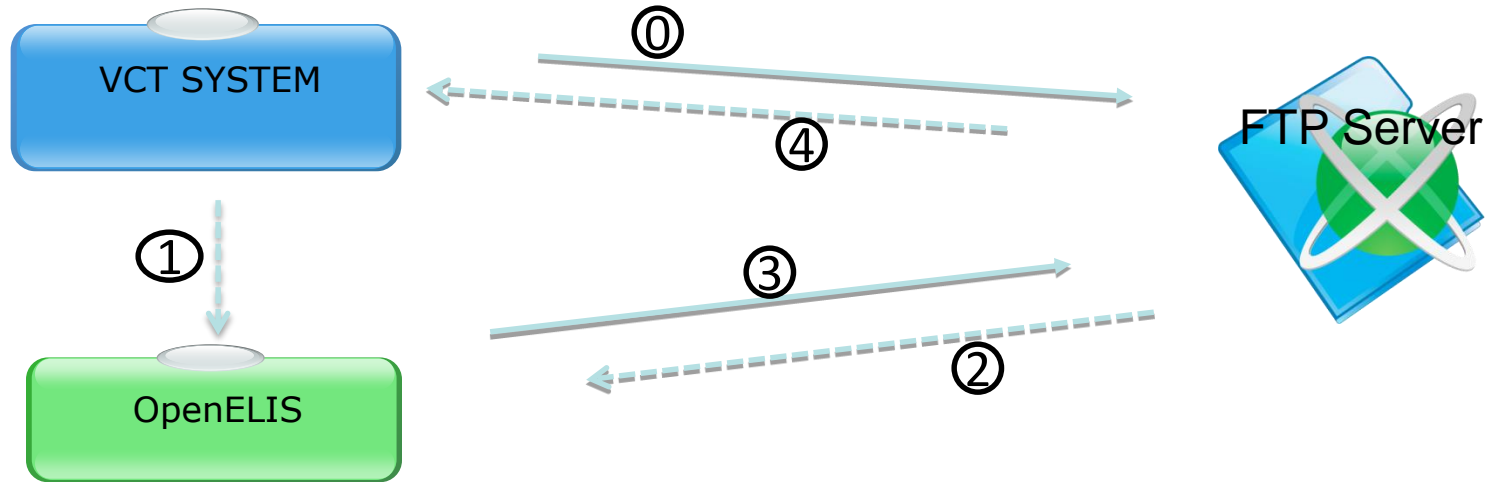
Application and Component Architecture

NON-WEB CLIENTS		WEB CLIENTS	
ADAPTERS (Instrument interfaces)	WEB SERVICES (SOAP objects, XML)	WEB APPLICATION FRAMEWORK OBJECTS (Struts ActionServlet, MVC-Controller)	
APPLICATION SPECIFIC & CORE BUSINESS OBJECTS (EJBs, Servlets, Java Beans)			
DB OBJECTS (EJBs, Hibernate, Java Beans)			
DATA SOURCES (Oracle, MS-SQL, mySQL, Postgres, XML, txt)			

- ✦ Data source layer: This is storage media where data resides, either structural databases or files such as XML and text files.
- ✦ DB objects layer: This layer provides an interface to the business layer to access data that stored in the data source layer. The business layer does not know how data would be saved and retrieved by the DB object layer. These objects are usually implemented as EJBs, Servlets, and Java beans. Third party object relational mappings such as Hibernate are increasing in popularity over enterprise entity beans and direct JDBC calls.
- ✦ Business objects layer: This layer includes both core objects and application specific objects. Core objects in this layer serve general business purposes commonly used in most applications. For example, data validations, message encryption/decryption, emails, etc. should be encapsulated in these objects. These objects work with other objects such as Web Services objects or application specific objects to fulfill clients' requests. Application specific objects should encapsulate business logic for a particular application and provide public interface to other applications via the controller layer to utilize their functionalities.
- ✦ Controller objects layer: This layer includes the adapter objects, web services objects, and Web applications framework objects. The controllers act upon receiving requests from clients and trigger actions accordingly. They act as gatekeepers between the clients and application resources.
- ✦ Client layer: Any applications or systems that send data to the data sources and/or present requested data to the end users or other systems.



Application and Component Architecture



Step	Description
0	VCT upload xml file (information patient) in FTP server
1	VCT request to LIS file upload success. Staff VCT call to Staff Lap OpenELIS about file upload success.
2	LIS download xml file (information) from FTP server then update result for new patient
3	LIS upload result test about 1 month in FTP server
4	VCT download result from FTP server and search new result for update into VCT system
note	LIS and VCT support import xml file from USB if disconnect network



Naming convention

✦ FTP storage folder

- ✦ VCT uploads to /VCT: Readonly by LIS, full control by VCT, None by others.
- ✦ LIS uploads to /LIS: Full control by LIS, readonly by others.
- ✦ If other programs added, ie OPC, the storage folder will be /OPC etc.
- ✦ All xml will be composed into a zip file and set password protected. Password will be set separately for each pair of a LIS and a VCT to ensure security

✦ File uploaded by VCT in PAC Ha Noi lab would be: **LIS-HNO-02.zip**, which contains LIS-HNO-02.xml. This file is uploaded to /VCT directory

✦ File uploaded by LIS in PAC Ha Noi lab would be: **LIS-VCT-HNO-PAC.zip**, which contains LIS-VCT-HNO-PAC.xml. This file is uploaded to /LIS directory.

✦ LIS data directory in FTP server

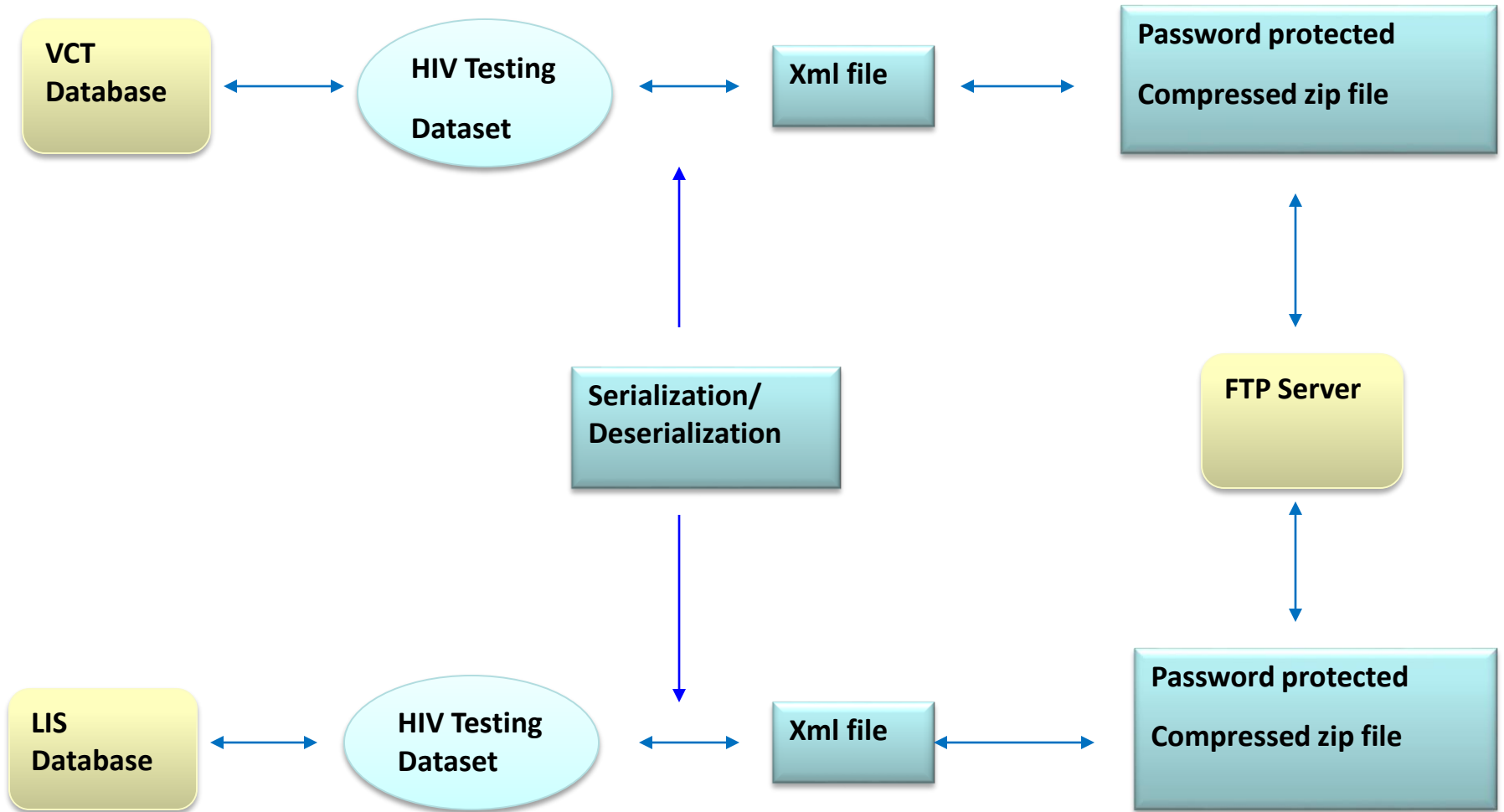
- ✦ /LIS
 - LIS-VCT-HNO-PAC.zip is VCT data file uploaded by LIS at Hanoi PAC
 - LIS-OPC-HPG-VTH.zip is VCT data file uploaded by LIS at Hanoi VTH

✦ VCT data directory in FTP server

- ✦ /VCT
 - LIS-HNO-01.zip is Data file uploaded by VCT site code 01 in Hanoi
 - LIS-HNO-02.zip is Data file uploaded by VCT site code 02 in Hanoi



Interfacing with VTC system

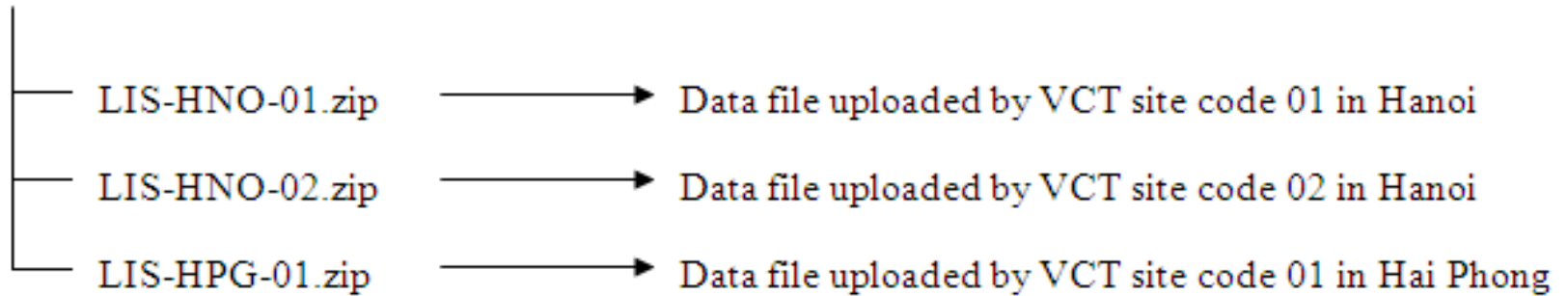




Interfacing with VTC system

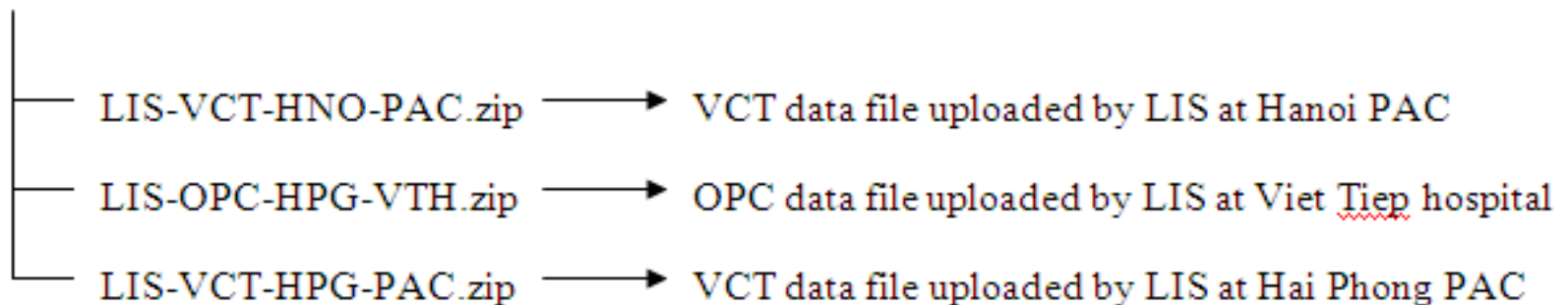
VCT data directory in FTP server

/VCT



LIS data directory in FTP server

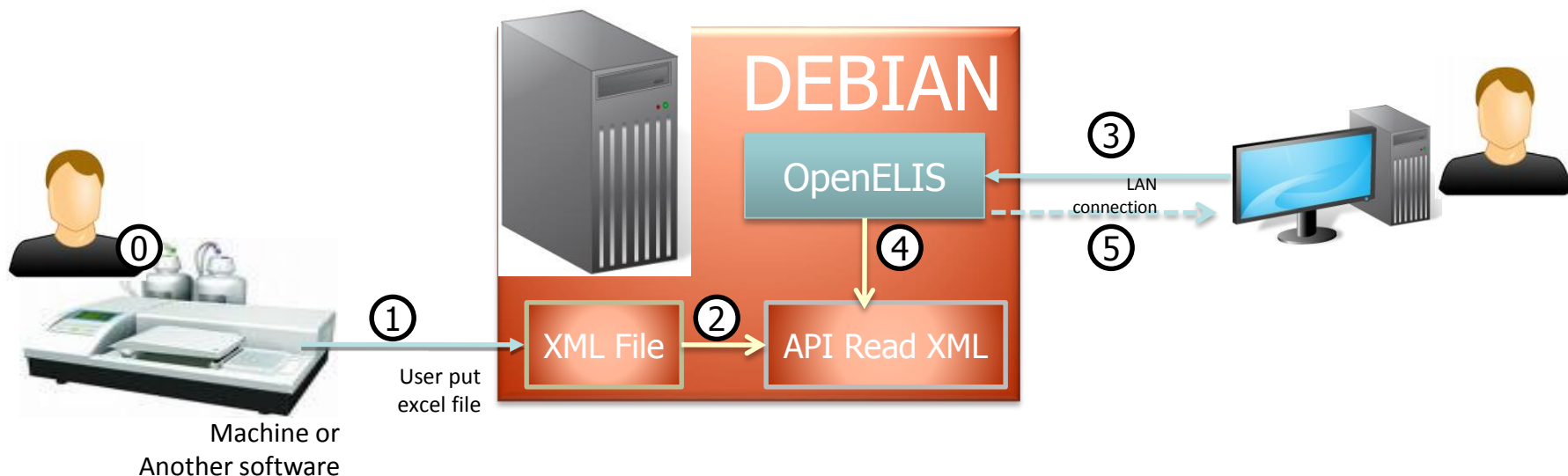
/LIS



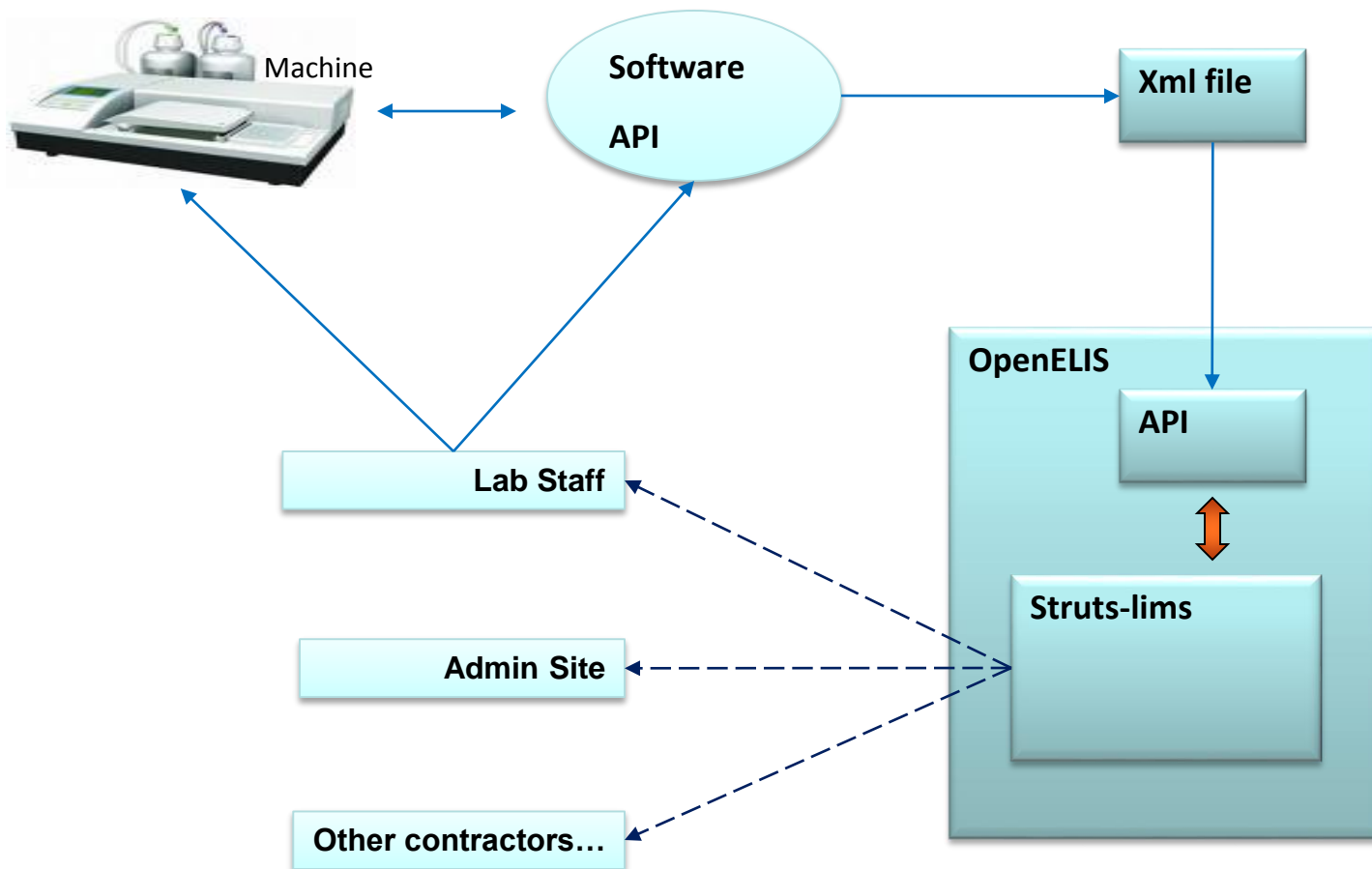


Global CyberSoft

Interfacing workflow with ELISA



Step	Description
0	A sample is inserted into the instrument for testing
1	ELISA export excel file and user put excel file into Debian
2	Function Read excel file get data from excel file an input into Database.
3	User accesses "Test Result Screen" of OpenELIS and inputs accession number (new screens for HIV, Toxoplasma... tests)
4	OpenELIS queries the result corresponding to the accession number from database
5	The result page is returned and displayed to user





Global CyberSoft

