C4G BLIS EQUIPMENT INTERFACE MANUAL

VERSION 1.0

PREFACE

PURPOSE OF THIS DOCUMENT

The goal of this document is to spell out all the necessary configurations to be made on BLIS and on any equipment that needs to be interfaced (either **uni-directional** or **bi-directional**) with BLIS. This document should be consulted at all times if there are any doubts in configurations on client sites. All supported equipment are listed in this document. Please note that only equipment listed here can be interfaced using the java based **BLIS Client Interface Tool.**

WHO SHOULD READ THIS DOCUMENT

This manual is written for BLIS (Basic Laboratory Information System) site implementers, developers and for those who want to know more about how BLIS is interfaced with equipment and other external systems in the medical facility.

For successful BLIS interface configuration and deployment at client sites, it is recommended that implementers are equipped with below skills;

Skill	Level of Competence
SQL	Intermediate
XML	Basic

Implementers also need to pay particular attention to details.

WHAT CAN YOU FIND IN THIS DOCUMENT

This configuration manual employs the step by step approach in helping you identify configuration sections, you can also find in this document pictures, sample SQL queries, explanations on configuration values and sample settings which can help you to fast track your site BLIS interface implementations. You will also find vivid clarifications on configurations for all supported equipment.

Project funded by PEPFAR

Philip Boakye   - **Team Lead**
Patricia Enninful   - **Technical Officer**
Stephen Adjei-Kyei   - **Software Developer**

CONVENTIONS USED IN THIS DOCUMENT

The chart below explains the icons used in this manual

When you see…	Then …
	The statement following the icon is a **tip** or a **reminder** of important information you are assumed to know. You can skip this information.
	The statement following the icon is **important** information and should not be ignored. It also alerts you to potential problems that you may encounter along the way. Read and remember these tidbits to avoid possible trouble.
	This icon signifies that the accompanying explanation **may be informative** but isn't crucial to BLIS interface deployment. Feel free to skip these snippets, if you like.

REVISION HISTORY

Version	Date	Author	Change
1.0	13/06/2014 -	Stephen Adjei-Kyei	Initial Draft

Project funded by PEPFAR

Philip Boakye　　　　　- **Team Lead**
Patricia Enninful　　　- **Technical Officer**
Stephen Adjei-Kyei　　- **Software Developer**

Table of Contents

Project funded by PEPFAR

Philip Boakye   **- Team Lead**
Patricia Enninful   **- Technical Officer**
Stephen Adjei-Kyei   **- Software Developer**

THE BLIS INTERFACE CLIENT

BLIS Interface client is a very lightweight tool for Interfacing BLIS with analyzers used in the medical laboratory. There are 2 main deployment modes. (Uni-directional and Bi-directional) The deployment mode actually differ per analyzer and this is purely based on the capability of the analyzer that has to be interfaced.

	Bi-directional Communication: 2 Way communication; Analyzer is able to send and receive information from BLIS
	Uni-directional Communication: 1 way communication; Analyzer only send results to BLIS

After a successful deployment;

a) **Bi-directional:**
   a. Patient demographics, specimen and test details on BLIS can be downloaded automatically into the analyzer
   b. Test results can be transferred from Analyzer to BLIS automatically
   c. There is no need to register sample and patient details on analyzer; everything is done on BLIS and transferred to analyzer.

b) **Uni-directional:**
   a. All tests performed on Analyzer with correct BLIS Specimen ID are automatically sent to BLIS.
   b. BLIS demographics cannot be transferred to analyzer

	BLIS Client Interface tool is a JAVA based application. Communication Protocols used are RS232, TCP/IP. Also BLIS Interface Client can directly connect to analyzer database or a flat file if accessible. The Messaging formats supported are HL7, ASTM, ABX Proprietary Format, Vital Proprietary format, CLSI LIS2-A, CLSI LIS2-A2 The protocol and messaging format used is purely determined by the capability of the analyzer that has to be interfaced.

HOW IT WORKS

BLIS Interface Client connects to BLIS via BLIS HTTP API and pulls details of patient, specimen and tests. It convert the details into analyzer format then transmit to the analyzer using analyzer supported

Project funded by PEPFAR

Philip Boakye   **- Team Lead**
Patricia Enninful   **- Technical Officer**
Stephen Adjei-Kyei   **- Software Developer**

protocol and message format. It is a platform independent tool which can run on Windows, Linux, UNIX and Mac.

Find Flow Diagram below for pictorial view

Diagram 1

BLIS INTERFACE FLOW DIAGRAM
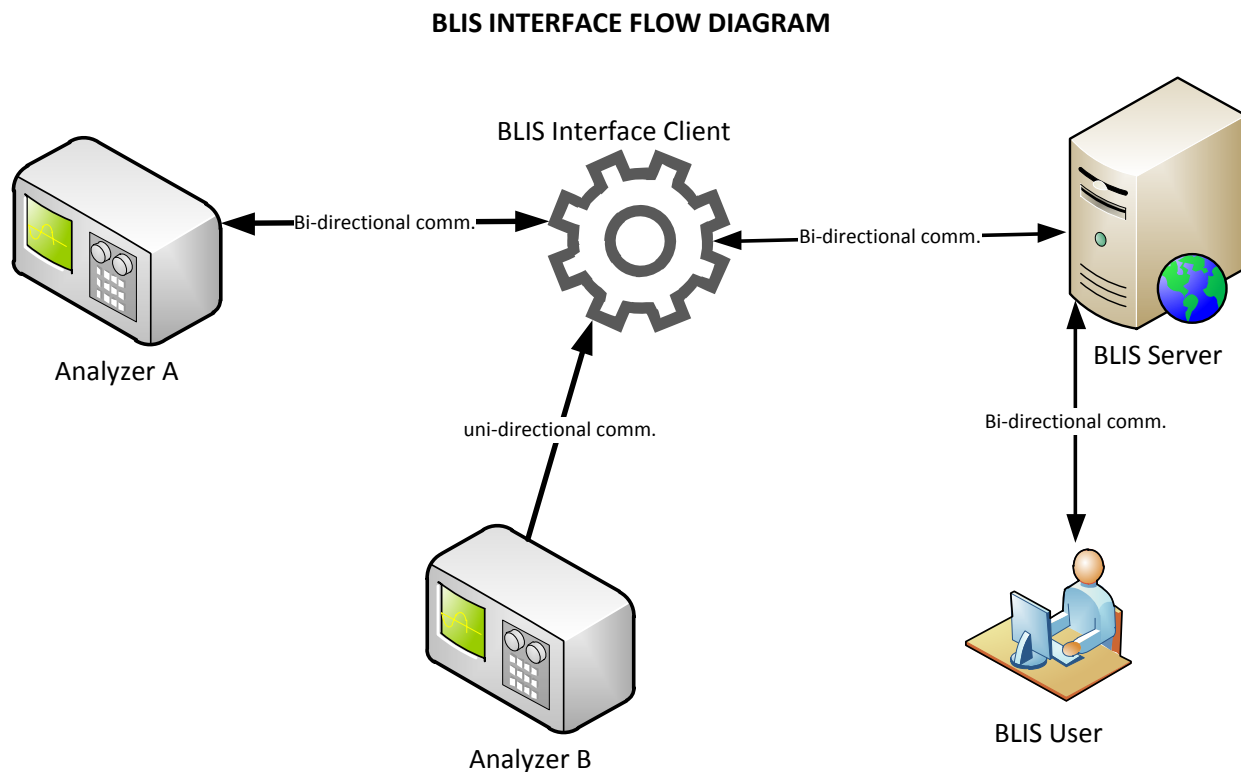
Typical bi-directional flow

A bi-directional interface with an analyzer has below steps:

1. Specimen Registration on BLIS
2. Specimen details are fetched from BLIS by BLIS Interface Client
3. Formatted details are sent to analyzer for tests to be performed
4. Tests results are sent from Analyzer to BLIS Interface Client
5. BLIS Interface Client transmits results to BLIS to be saved.

Project funded by PEPFAR

Philip Boakye   **- Team Lead**
Patricia Enninful   **- Technical Officer**
Stephen Adjei-Kyei   **- Software Developer**

Diagram 2
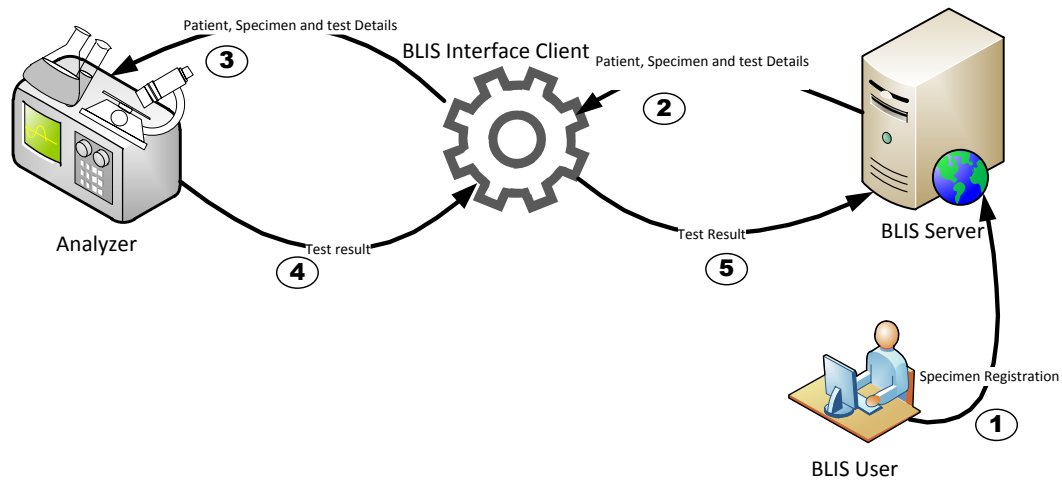
Typical uni-directional flow

A Uni-directional interface with an analyzer has below steps:

1. Specimen Registration on BLIS
2. Lab Technician register specimen with BLIS Specimen ID on Analyzer
3. Tests results are sent from Analyzer to BLIS Interface Client
4. BLIS Interface Client transmits results to BLIS to be saved.

Diagram 3

Project funded by PEPFAR

Philip Boakye   **- Team Lead**
Patricia Enninful   **- Technical Officer**
Stephen Adjei-Kyei   **- Software Developer**

Lab Technician register specimen
with BLIS Specimen ID on Analyzer
2

Analyzer

LAB Technician

Test result with BLIS Specimen ID
3

BLIS Interface Client

Test result with BLIS Specimen ID
4

BLIS Server

Specimen Registration
1

BLIS User
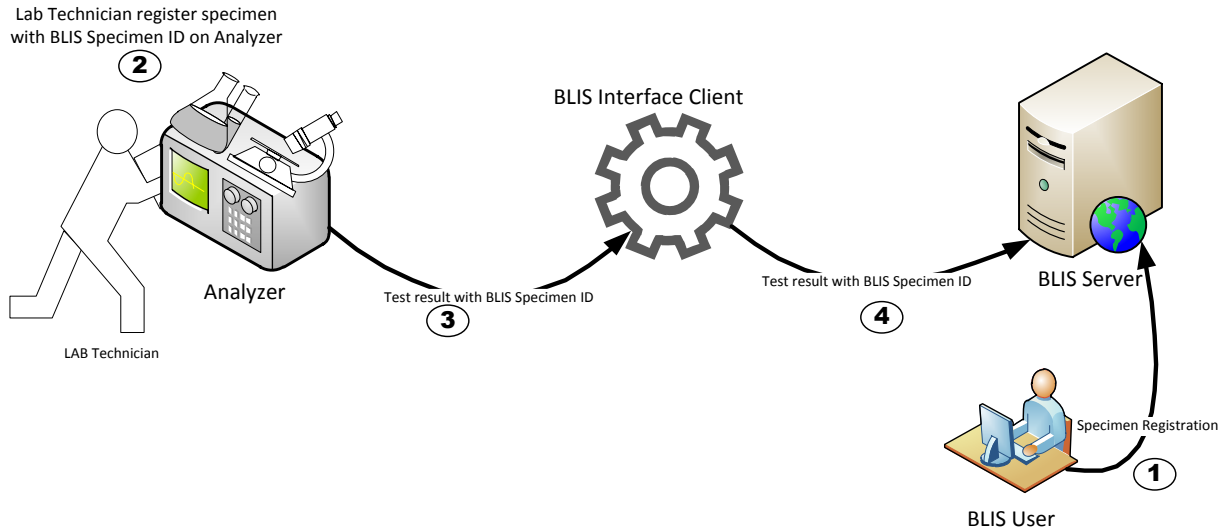
IMPORTANT CONFIGURATION FILES

Important!	Please pay particular attention to details in these files. The behavior of BLIS Interface Client depends on them. If you have any doubts about a change please refer to the section which explains that section or contact **GHSS BLIS Team** for more clarification.

Main Configuration file

The main configuration file is **BLISInterfaceClient.ini** (Text Based configuration) located at the main directory of BLIS Interface Client. The format of **BLISInterfaceClient.ini** is like most popular configuration files for php (**php.ini**), mysql (**my.ini**), etc.

i	Beginning a line with **Semi colon (;)** sets that line as a comment and will not be used by BLIS Interface Client. Example: ;** ; I am a comment. The above line is a comment. The line below is also a comment ;**

Project funded by PEPFAR

Philip Boakye   **- Team Lead**
Patricia Enninful   **- Technical Officer**
Stephen Adjei-Kyei   **- Software Developer**
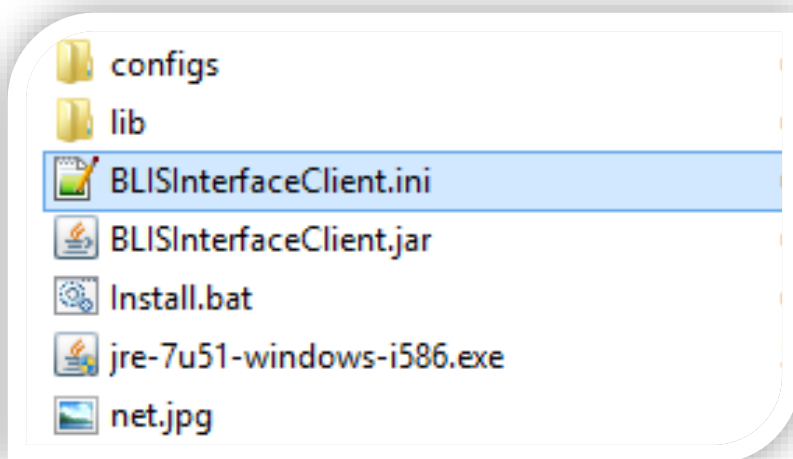
Sections

[FEED SOURCE]

This section specifies the source from which information will be retrieved. Supported options are

a) **RS232**: Connection via serial port. This option is used for equipment where TCP/IP connection is not available.

b) **TCP/IP**: Connection via TCP sockets with server IP address and port.

c) **MSACCESS**: Direct database access

d) **HTTP:** Connection via HTTP protocol
e) **TEXT:** Connection to flat files for data

[MSACCESS CONFIGURATIONS]
This portion is used to configure settings for Microsoft Access connection. Configuration items are:

a) **DATASOURCE:** The name of the ODBC data source configured at the **ODBC Data source administrator dialog.** As of this writing, Only ABX Pentra 60 C+ supports MS Access

Project funded by PEPFAR

Philip Boakye   **- Team Lead**
Patricia Enninful   **- Technical Officer**
Stephen Adjei-Kyei   **- Software Developer**

configurations. Please refer to **ABX Pentra 60 C+** configuration section for details on how to configure system data source at the ODBC Administrator dialog from control panel.

b) **DAYS:** No of days old data to pull from BLIS. Default is 0 which means current day.

[RS232 CONFIGURATIONS]

This portion is used to configure settings for serial port connection. Items are:

a. **COMPORT:** The port on which to communicate with the analyzer. If the analyzer you want to interface uses RS232 communication, refer to the section on how to find the comport on your system.

b. **BAUD_RATE:** The speed of connection.eg (4800,9600,38400,57600,115200)

c. **PARITY:** How is data checked for errors? Available options are (Odd, Even, None). Default is "None".

d. **STOP_BITS:** Choose the time between character transmissions for your serial device. Available options are (1, 2). Default is 1

e. **DATA_BITS:** Choose the number of data bits used by your serial device to represent characters. Available options are (7, 8). Default is 8

f. **APPEND_NEWLINE:** Should the data transfer be appended with the **Newline (/n)** character (Yes/No). Default is No

g. **APPEND_CARRIAGE_RETURN:** Should the data transfer be appended with the **Enter (/r)** character (Yes/No). Default is No

[TCP/IP CONFIGURATIONS]

This portion is used for the configuration of TCP/IP connection. Items are:

a. **PORT:** Port on which to listen for data

b. **EQUIPMENT_IP:** The IP address of the analyzer system if applicable

c. **MODE:** How should the BLIS Interface Client connect to analyzer? Should it connect as a server or client? Options (server/client). Default is server

d. **CLIENT_RECONNECT:** This setting is used only if **MODE=client.** Should the application reset client connection when server sends **"End of Transmission (EOT)"?** Options (yes/no). Default is yes

[TEXT]

This portion is used to configure reading of text files for test results. Items are:

a. **BASE_DIRECTORY:** The parent directory where all files will be located

b. **USE_SUB_DIRECTORIES:** Should sub directories be opened for files? (Options are yes/no)

Project funded by PEPFAR

Philip Boakye   - Team Lead
Patricia Enninful   - Technical Officer
Stephen Adjei-Kyei   - Software Developer

c. **SUB_DIRECTORY_FORMAT:** Sub directories naming format. Example **ddMMyy**. Only used when USE_SUB_DIRECTORIES = yes

d. **FILE_NAME_FORMAT:** Files naming format

e. **FILE_EXTENSION:** File extension used. Example txt,csv,exp,evl

f. **FILE_SEPERATOR:** File value separator used. Supported Options (TAB,COMMA,COLON,SEMI-COLON,SPACE)

[BLIS CONFIGURATIONS]

This portion is used to set the BLIS HTTP server configurations that will be used by this tool to communicate with BLIS. Items are:

a. **BLIS_URL:** The URL on which BLIS is accessed from the web browser. Example is http://192.168.56.1:4001/

b. **BLIS_USERNAME:** BLIS account username

c. **BLIS_PASSWORD:** BLIS account password

[EQUIPMENT]

This section contains the list of all supported equipment this version of BLIS Interface Client can communicate with. This portion is pre-filled with all supported analyzers.

Important!	Please note that if your equipment is not listed here you cannot connect with this version of BLIS Interface Client. To use any of the equipment listed here just uncomment that line. It is extremely important that you uncomment only **ONE (1)** equipment at a time.

[MISCELLANEOUS]

This portion is used to set miscellaneous application configurations. Items are:

a. **ENABLE_LOG:** Set whether you want logs to be generated by application. Options are (yes/no). Default is yes

b. **WRITE_TO_FILE:** Set to enable or disable writing messages received from analyzer to file. Options are (yes/no). Default is no

c. **POOL_DAY:** How many days old data to pull from BLIS. 0: is for current day, 1: for yesterday, 2: for 2 days ago, etc. Default is 0

d. **POOL_INTERVAL:** Time interval in seconds to wait and fetch for data in BLIS. Default is 30 seconds

Project funded by PEPFAR

Philip Boakye	**- Team Lead**
Patricia Enninful	**- Technical Officer**
Stephen Adjei-Kyei	**- Software Developer**

e. **ENABLE_AUTO_POOL:** Set whether automatic data retrieval is enabled or not. Options are (yes/no). Default is yes

Analyzer specific configuration files

Every analyzer has its own **xml** configuration file. All specific settings are made in their respective files.

They are located at **mainfolder\configs\AnalyzerName***

The purpose of the analyzer specific configuration file is to match BLIS specimen types, test types and measures to that of the analyzer so that tests can be sent seamlessly from the analyzer to BLIS and test can be saved into BLIS from the analyzer as though they were entered manually by technicians.

Please refer to the configuration section of the particular analyzer you want to interface with BLIS for specific details.

Sections

supportedtestTypes

This portion is used to configure BLIS tests that are supported by the analyzer. BLIS Interface Client will only send BLIS tests configured here to the analyzer. The most important field to fill here is the **testtypeid**

```
<supportedtestTypes>
    <testtype>
        <testtypeid>306</testtypeid>
        <testtypename>FBC Day</testtypename>
    </testtype>
</supportedtestTypes>
```

> **test_type_id** from BLIS Database

You just have to repeat the **testtype** block for each BLIS test type you want to configure.

<texttypeid>**BLIS TesttypeID**</testtypeid>

You must select the BLIS **test_type_ids** from the database using SQL.

Project funded by PEPFAR

Philip Boakye   **- Team Lead**
Patricia Enninful   **- Technical Officer**
Stephen Adjei-Kyei   **- Software Developer**

(i)	Find example SQL query below to check and identify the test_type_id for **Potassium**. Query below might yield more than 1 row; look through and pick the exact test_type you want.
	`SELECT test_type_id, name`
	`FROM ` `` `test_type` ``
	`WHERE name LIKE '%Potassium%'`
	Please note that above query is just an example. Depending on the implementation you are performing, you might write other queries to find the BLIS test_type_id.
	Please refer to the database section for more details

<testtypename>**BLIS TestType name**</testtypename>

Type in the name of the test for which you have setup the ID.

Important!	Please note that tests not configured here will not be sent to analyzer. All supported tests must be setup here.

Supportedsamples

This portion is used to configure BLIS specimen types that are supported by the analyzer. BLIS Interface Client will only send BLIS specimens configured here to the analyzer. The most important element to fill here is the **lissampleid**

> Specimen_type_id from BLIS Database

```
<supportedsamples>
    <sample>
        <equipmentsampleid>1</equipmentsampleid>
        <lissampleid>1</lissampleid>
        <samplename>Serum</samplename>
    </sample>
    <sample>
        <equipmentsampleid>3</equipmentsampleid>
        <lissampleid>3</lissampleid>
        <samplename>Whole Blood</samplename>
    </sample>
</supportedsamples>
```

You just have to repeat the **sample** block for each BLIS specimen type you want to configure.

Project funded by PEPFAR

Philip Boakye   **- Team Lead**
Patricia Enninful   **- Technical Officer**
Stephen Adjei-Kyei   **- Software Developer**

< lissampleid >**BLIS specimen_type_id** </ lissampleid >

You must select the BLIS **specimen_type_ids** from the database using SQL.

<table>
<tr>
<td>(i)</td>
<td>Find example SQL query below to check and identify the specimen_type_id for Urine.
<code>SELECT specimen_type_id,name
FROM `specimen_type`
WHERE name LIKE '%Urine%'</code>
Please note that above query is just an example. Depending on the implementation you are performing, you might write other queries to find the BLIS specimen_type_id.
Please refer to the database section for more details</td>
</tr>
</table>

<table>
<tr>
<td>Important!</td>
<td>Please note that specimen types not configured here will not be sent to analyzer. All supported specimen must be setup here.</td>
</tr>
</table>

supportedtests

This portion is used to configure BLIS measures that are supported by the analyzer. BLIS Interface Client will only send BLIS measures configured here to the analyzer. The most important element to fill here is the **listestid**

```
<listest>
    <equipmenttestid>1</equipmenttestid>
    <listestid>632</listestid>
    <listestname>WBC</listestname>
</listest>
<listest>
    <equipmenttestid>2</equipmenttestid>
    <listestid>623</listestid>
    <listestname>RBC</listestname>
</listest>
<listest>
    <equipmenttestid>3</equipmenttestid>
    <listestid>624</listestid>
    <listestname>HGB</listestname>
</listest>
```

This ID is dependent on the analyzer and is configured by developer. **Please do not modify any equipmenttesttypeid. Leave it as it is.**

measure_id from BLIS Database

Project funded by PEPFAR

Philip Boakye	**- Team Lead**
Patricia Enninful	**- Technical Officer**
Stephen Adjei-Kyei	**- Software Developer**

	Please note that Measures in BLIS are the actual tests that are performed on the analyzer and not the Test Types. For Instance a BLIS test type like FBC could have measures like WBC, RBC, HGB, etc. These measures are what you need to configure here.

You just have to repeat the **listest** block for each BLIS measure you want to configure.

< listestid>**BLIS measure_id** </ listestid>

You must select the BLIS **measure _ids** from the database using SQL.

	Find example SQL query below to check and identify the **measure _id** for **Potassium**. SELECT measure_id, name FROM `measure` WHERE name LIKE '%Potassium%' Please note that above query is just an example. Depending on the implementation you are performing, you might write other queries to find the BLIS measure_id. **Please refer to the database section for more details**
	Every measure is related to a test_type. Find example SQL query below to find all measures that belong to **Electrolyte** test type: SELECT tm.test_type_id, tm.measure_id, t.name AS testname, m.name AS measurename FROM `test_type_measure` tm INNER JOIN test_type t ON t.test_type_id = tm.test_type_id INNER JOIN measure m ON m.measure_id = tm.measure_id WHERE t.name LIKE 'Electrolyte' **Example output of query above** | test_type_id	measure_id	testname	measurename | |---|---|---|---| | 4 | 1 | Electrolyte | Sodium | | 4 | 2 | Electrolyte | Potassium | | 4 | 3 | Electrolyte | Phosporus | You need to configure all 3 measures here plus all other measures from all supported BLIS test_types. **Please refer to the database section for more details**

|	Please note that measures not configured here will not be sent to analyzer. Tests performed on analyzer will not be sent to BLIS if they have not been configured here.

SETTING UP BLIS INTERFACE CLIENT ON WINDOWS
1. Copy the BLIS Interface Client Folder to **Drive C:**

Project funded by PEPFAR

Philip Boakye - Team Lead
Patricia Enninful - Technical Officer
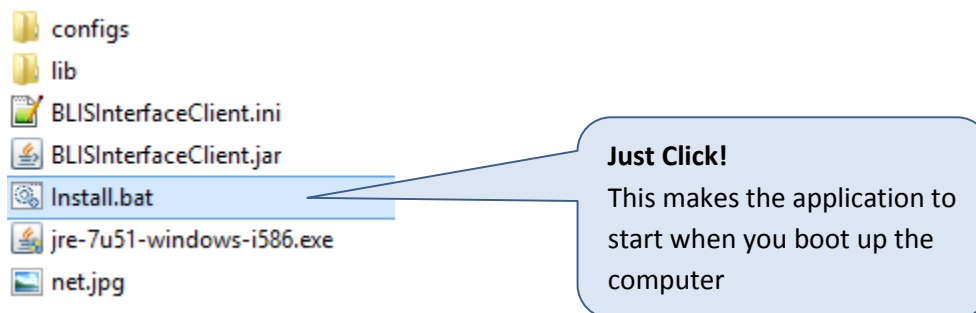**Stephen Adjei-Kyei** - Software Developer

2. The tool needs Java runtime to run. The main folder contains Java runtime environment 7 update 51. Install this file first.
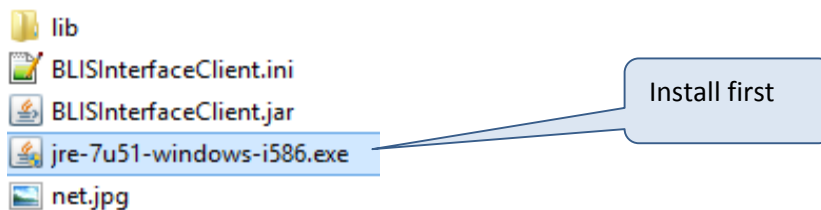
You can get the current version of java runtime environment from here:
http://java.com/en/download/

3. Make configurations in the **BLISInterfaceClient.ini** file based on the equipment you want to interface. Please refer to the section which explains required and correct configuration for your analyzer.

4. Make supported BLIS specimen types, tests types and measure configurations in the appropriate **xml** configuration file for your analyzer. Please refer to the section which explains required and correct configuration for your analyzer. For example **Mindray BS-200E** configuration can be found at **mainfolder\configs\hl7\MindrayInterface.xml**

5. Click on **Install.bat** to setup BLIS Interface Client to start when Windows starts

6. Click on **BLISInterfaceClient.jar** to start tool

Project funded by PEPFAR

Philip Boakye   **- Team Lead**
Patricia Enninful   **- Technical Officer**
Stephen Adjei-Kyei   **- Software Developer**

SETTING UP BLIS INTERFACE CLIENT ON LINUX

SETTING UP BLIS INTERFACE CLIENT ON MAC OS

SPECIFIC CONFIGURATIONS FOR INTERFACING ANALYZERS

Below sections are required regardless of the type of analyzer used. These section must be set correctly. Find example settings below:

[BLIS CONFIGURATIONS]

> 1. Put correct BLIS URL here
> 2. Create a new user account for the analyzer and put details here.

```
[BLIS CONFIGURATIONS]
    ;BLIS URL local
    BLIS_URL = http://192.168.56.1:4001/
    ;username
    BLIS_USERNAME = equipment
    BLIS_PASSWORD = equipment
```

Project funded by PEPFAR

Philip Boakye   **- Team Lead**
Patricia Enninful   **- Technical Officer**
Stephen Adjei-Kyei   **- Software Developer**

[EQUIPMENT]

```
[EQUIPMENT]
    ;ABX Pentra 60C+
    ;ABX MICROS 60
    ;ABX Pentra 60
    ;Pentra ES 60
    ;Mindray BS-120
    ;Mindray BS-130
    ;Mindray BS-180
    ;Mindray BS-190
    ;Mindray BS-200
    ;Mindray BS-220
    Mindray BS-200E
```

All supported equipment has already been listed and commented here:
Just uncomment your analyzer name

[MISCELLANEOUS]

```
[MISCELLANEOUS]
    ; options are yes/no
    ENABLE_LOG = yes
    ;Option to enable writing messages received from analyzer to file. Options are yes/no
    WRITE_TO_FILE = yes
    ;How many days old data to pull from BLIS. 0: is for current day, 1: for yesterday, 2: for 2 days ago, etc
    POOL_DAY = 0
    ;Time interval in Seconds to wait and fetch for data. Default is 30 seconds
    POOL_INTERVAL = 30
    ; Set whether automatic data retrieval is enabled or not. Options are yes/no
    ENABLE_AUTO_POOL = yes
```

Make changes as required

The various specific equipment configuration xml files are very similar. They have the same sections and so you do no need any special training in setting them up.
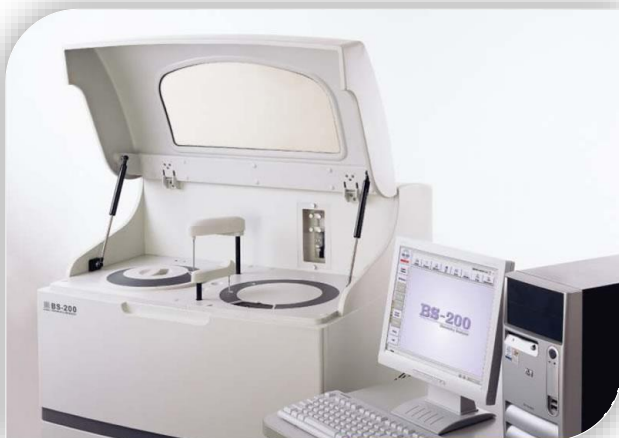
Project funded by PEPFAR

Philip Boakye   **- Team Lead**
Patricia Enninful   **- Technical Officer**
Stephen Adjei-Kyei   **- Software Developer**

All the files are located at **Mainfolder/configs/;** as at the time of this writing, there are 10 configurations ready for deployment.
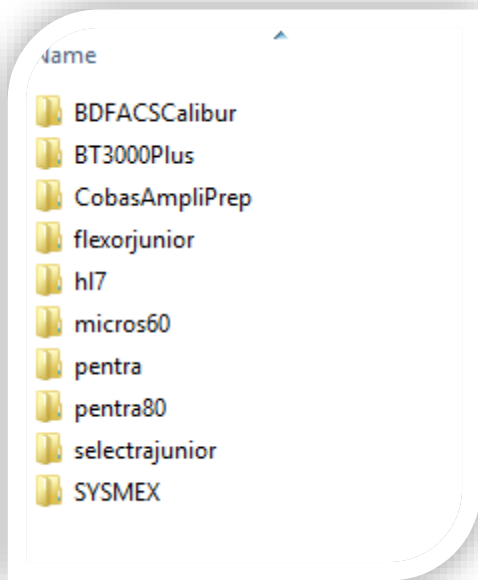
This section will explain the specific configurations that needs to be made in the main application configuration file **(BLISInterfaceClient.ini).**

Mindray BS-200E

Project funded by PEPFAR

Philip Boakye   **- Team Lead**
Patricia Enninful   **- Technical Officer**
Stephen Adjei-Kyei   **- Software Developer**

The Mindray BS -200E is a chemistry analyzer. This analyzer uses TCP/IP connection.

i	Please note that: The computer attached to the analyzer needs to be configured to reach the BLIS network. Depending on the facility you want to implement this; you might require a network cable or a wireless adapter.

BLISInterfaceClient.ini

Configuration sections that needs to be modified are as follows:

[FEED SOURCE]

```
[FEED SOURCE]
     TCP/IP
```

Set it as it is here

[TCP/IP CONFIGURATIONS]

```
[TCP/IP CONFIGURATIONS]
    ;Port on which to listen for data
    PORT = 5150
    ;equipment IP address if applicable
    EQUIPMENT_IP = 192.168.10.2
    ;How should the application connect to analyzer. options (server/client)
    MODE = server
    ;This setting is used only if MODE=client
    ;Reset client Connection when server sends "End of Transmission (EOT)". Options (yes/no)
    CLIENT_RECONNECT = yes
```

Correct MODE for this analyzer is **server**

MindrayInterface.xml

This is the main configuration file for BS-200E and can be located at
\BLISInterfaceClient\configs\hl7\ MindrayInterface.xml

Project funded by PEPFAR

Philip Boakye   **- Team Lead**
Patricia Enninful   **- Technical Officer**
Stephen Adjei-Kyei   **- Software Developer**

Important!	**The <Message></Message> Section should be ignored. These configurations are used by the system and was configured by developer.**

The rest of the sections are same as other xml files.

Important!	The Mindray BS-200E analyzer allows you to configure all the BLIS measure IDs in the analyzer. This is to map the analyzer test to BLIS ID so that the analyzer will send the BLIS ID to BLISInterfaceClient. **Please ask for more details from the LAB Technicians at the facility**

ABX Pentra 60 C+

The ABX Pentra 60 C+ is a hematology analyzer. The BLISInterfaceClient connects to its MS Access Database.

Below are the configuration that needs to be setup in **BLISInterfaceClient.ini**

FEED SOURCE]

```
[FEED SOURCE]
     MSACCESS
```
Set it as it is here

```
;****************************************************
;
;Configuration settings for Microsoft Access database connection
;
;****************************************************
[MSACCESS CONFIGURATIONS]
     ;The ODBC Datasouce name
     DATASOURCE = PentraCon
     ;How many days old test data should be imported into msacess database.
     ;Default is 1 day old test data
     DAYS = 0
```
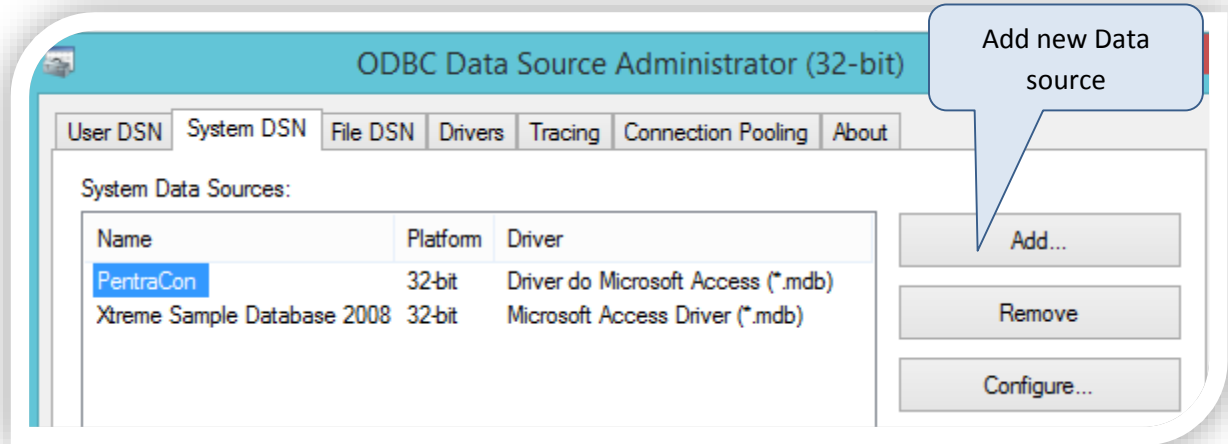Data source name

Project funded by PEPFAR

Philip Boakye   - **Team Lead**
Patricia Enninful   - **Technical Officer**
Stephen Adjei-Kyei   - **Software Developer**

How to setup the data source name

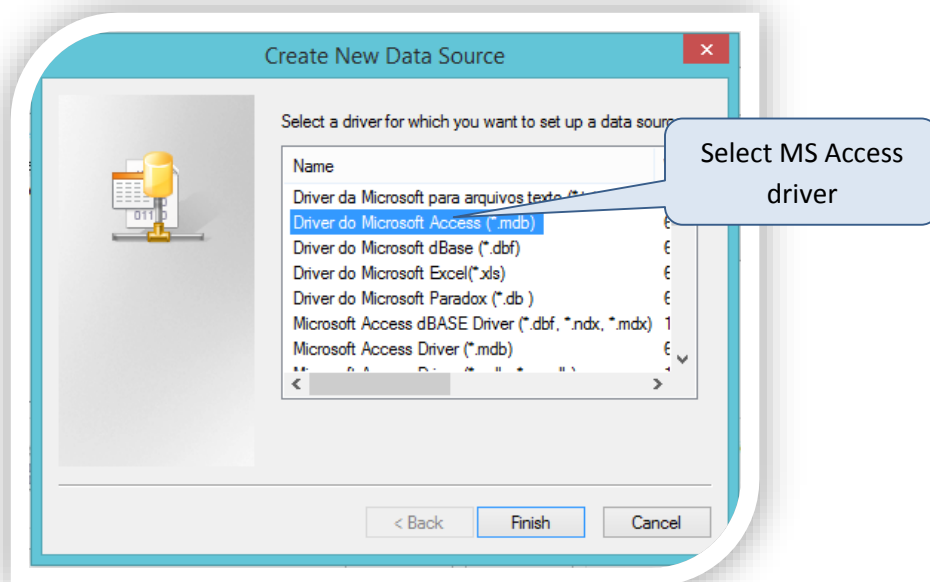The ABX Pentra 60 C+ is setup on Windows XP

1. Open Control Panel
2. Click on Administrative tools
3. Click on ODBC Data sources
4. Select the System DNS Tab

5. Click on Add
6. Select **"Driver do Microsoft Access (*.mdb)"**

Project funded by PEPFAR

Philip Boakye   **- Team Lead**
Patricia Enninful   **- Technical Officer**
Stephen Adjei-Kyei   **- Software Developer**

Select MS Access driver

7. Click on finish
8. Enter a name for your data souce
9. Click on select button to select the MS access database
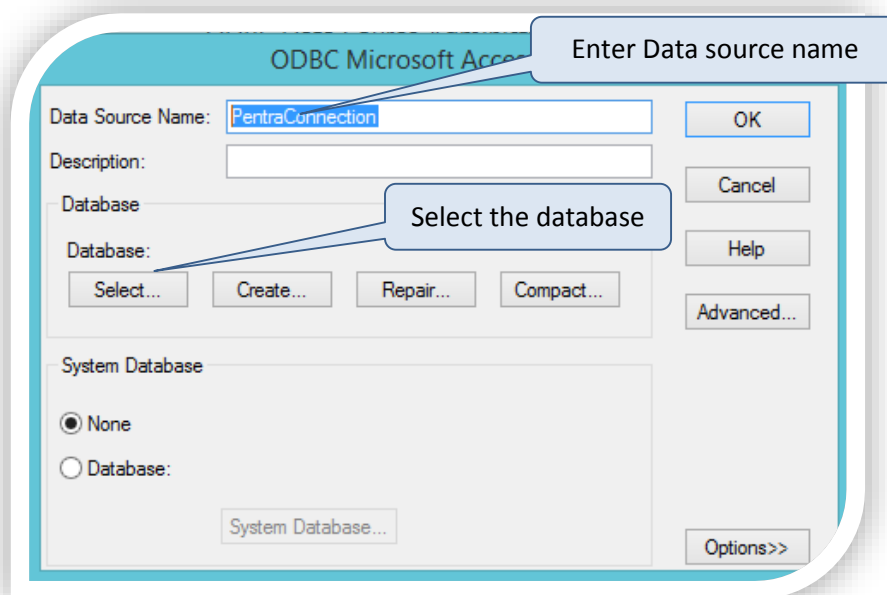
Enter Data source name
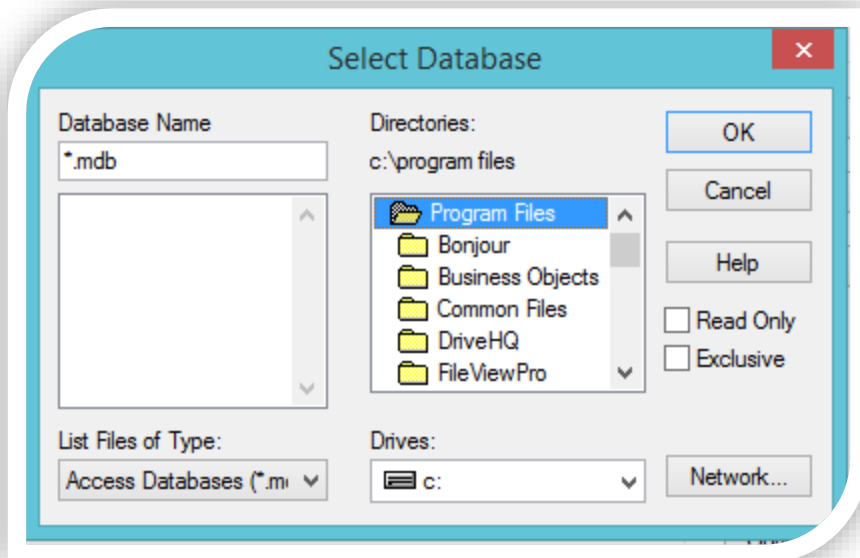
Select the database

Project funded by PEPFAR

Philip Boakye   **- Team Lead**
Patricia Enninful   **- Technical Officer**
Stephen Adjei-Kyei   **- Software Developer**

10. Select the database file
    a. By default the database name is **ASP01DB.MDB**
    b. It is usually located at C:\program files\abx\data\
    c. Select the file and click on OK
11. Enter the data source name you created at the data source section in the configuration file as indicated above.

ABX MACROS 60

The ABX Macros 60 analyzer has no computer attached to it. BLISInterfaceClient connects to it using RS232 protocol.

Below are the configuration that needs to be setup in **BLISInterfaceClient.ini**

```
;^^^^^^^^^^^^^^^^^
[FEED SOURCE]
    RS232
```

Project funded by PEPFAR

Philip Boakye   **- Team Lead**
Patricia Enninful   **- Technical Officer**
Stephen Adjei-Kyei   **- Software Developer**

```
.S232 CONFIGURATIONS]
    ;The comm port on which to connect
    COMPORT = COM11
    ;The speed of connection.eg (4800,9600,38400,57600,115200)
    BAUD_RATE = 38400
    ;How should data be checked for errors. Available options are (Odd,Even,None)
    PARITY = None
    ;Choose the time between character transmission for your serial device. Available o
    STOP_BITS = 1
    ;Choose the number of data bits used by your serial device to represent characters.
    DATA_BITS = 8
    ;Should the data transfer be appended with the Newline(/n) character (Yes/No)
    APPEND_NEWLINE = No
    ;Should the data transfer be appended with the Enter(/r) character (Yes/No)
    APPEND_CARRIAGE_RETURN = No
```

SYSMEX XS 500i

BT 3000 PLUS

BD FACSCalibur

ABX Pentra 80

SELECTRA JUNIOR

Database Section

BLIS uses MySQL as its backend database. You can connect to the database using any MySQL client; like MySQL workbench, MySQL Query Browser, MySQL front or phpMyAdmin.

Project funded by PEPFAR

Philip Boakye	**- Team Lead**
Patricia Enninful	**- Technical Officer**
Stephen Adjei-Kyei	**- Software Developer**

phpMyAdmin has been included in the BLIS package. You can connect to the database after you have started BLIS.exe. Use example URL as follows:

http://192.168.56.1:4001/**phpMyAdmin**

Username: **root**

Password: **blis123**

How to know the Working Database

You need to know the current working database for the site you are interfacing before you can select accurate data. To get the site database follow steps below:

1. Log into BLIS as **administrator**
2. Click on **Lab Configuration**
3. Look at the **Address bar** for the database name
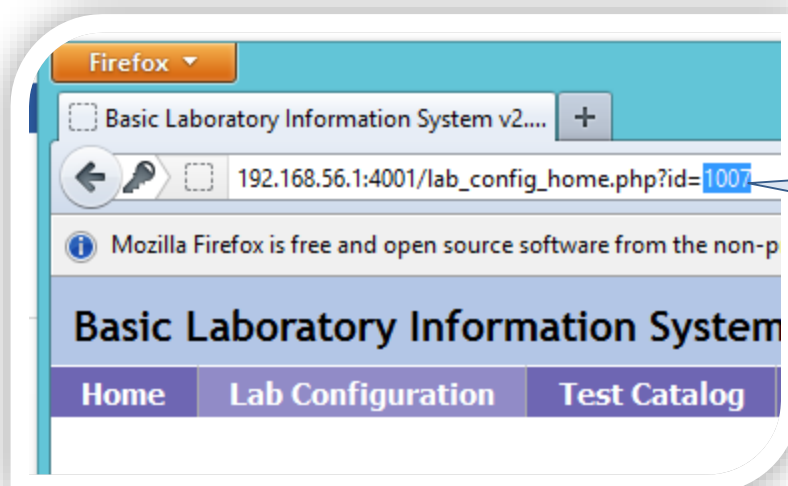
4. Open phpmyAdmin using previous section details.
5. Look at the database list on the left hand side and select correct database

Project funded by PEPFAR

Philip Boakye	**- Team Lead**
Patricia Enninful	**- Technical Officer**
Stephen Adjei-Kyei	**- Software Developer**

Select database **blis_1007**
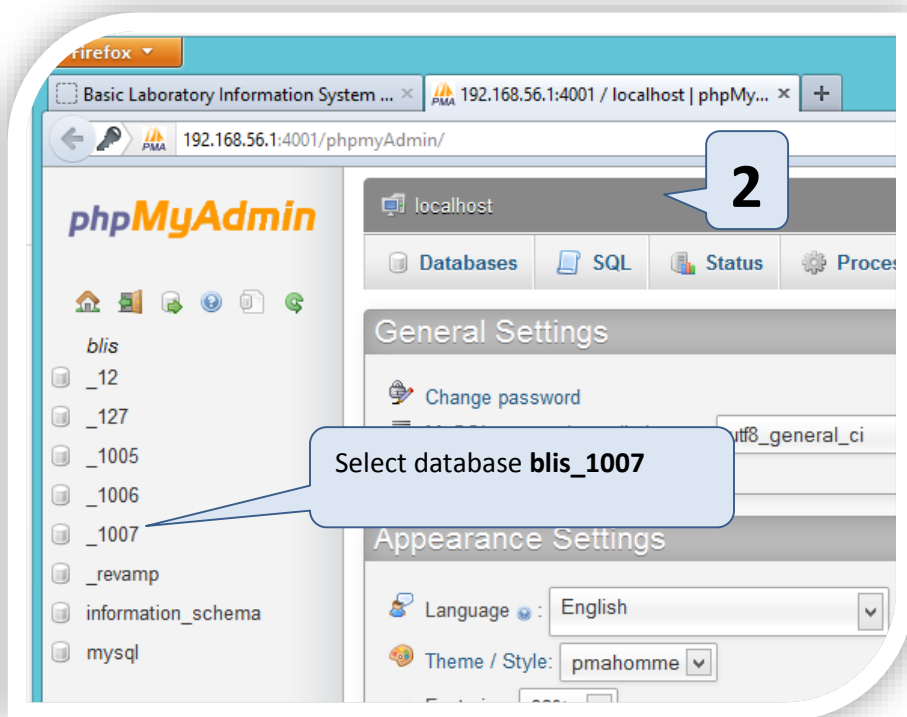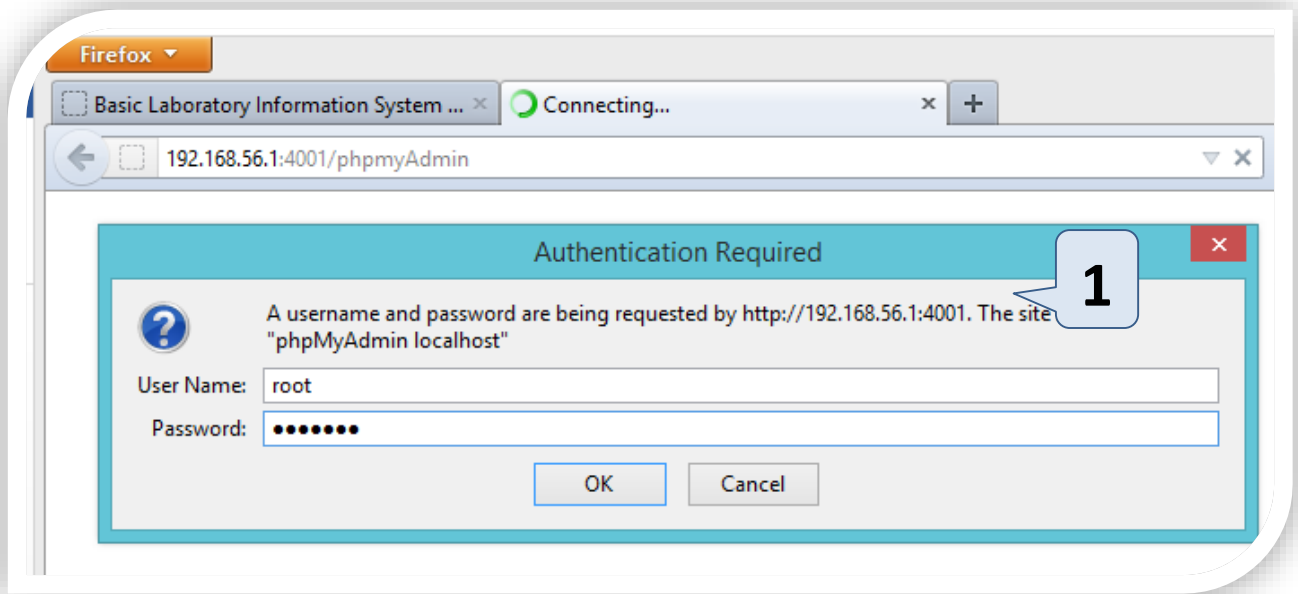
Project funded by PEPFAR

Philip Boakye   **- Team Lead**
Patricia Enninful   **- Technical Officer**
Stephen Adjei-Kyei   **- Software Developer**
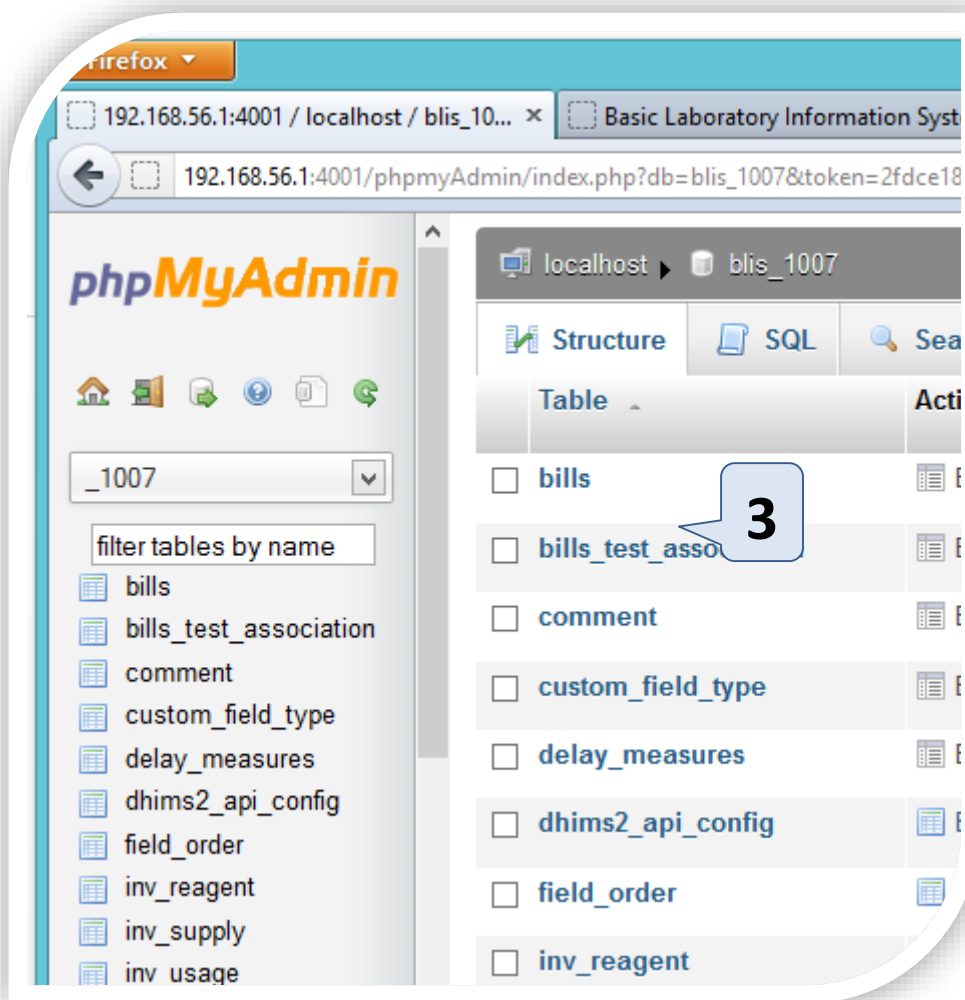
Tables to consider

There are 4 main tables to consider for BLIS equipment interfacing:

1. **specimen_type :** Contains the details of all configured BLIS specimen types
2. **test_type :** Contains the details of all configured BLIS tests types
3. **measure :** Contains the details of all measures in BLIS
4. **test_type_measure:** Contains the association of all BLIS measures to their respective test types

specimen_type

Table comments: InnoDB free: 0 kB

Column	Type	Null
specimen_type_id	int(11)	No
name	varchar(45)	No
description	varchar(100)	Yes
ts	timestamp	No
disabled	int(10)	No

Indexes: ☻

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null
PRIMARY	BTREE	Yes	No	specimen_type_id	4	A	

measure

Table comments: InnoDB free: 0 kB

Column	Type	Null	
measure_id	int(11)	No	
name	varchar(45)	No	
unit_id	int(10)	Yes	N
range	varchar(500)	Yes	N
description	varchar(500)	Yes	N
ts	timestamp	No	C
unit	varchar(30)	Yes	N

Indexes: ☻

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	
PRIMARY	BTREE	Yes	No	measure_id	12	A		
unit_id	BTREE	No	No	unit_id	2	A	YES	

Project funded by PEPFAR

Philip Boakye   **- Team Lead**
Patricia Enninful   **- Technical Officer**
Stephen Adjei-Kyei   **- Software Developer**

test_type

Table comments: InnoDB free: 0 kB

Column	Type	Null
test_type_id	int(11)	No
name	varchar(45)	No
description	varchar(100)	Yes
test_category_id	int(11)	No
ts	timestamp	No
is_panel	int(10)	Yes
disabled	int(10)	No
clinical_data	longtext	Yes
hide_patient_name	int(1)	Yes
prevalence_threshold	int(3)	Yes
target_tat	int(3)	Yes

Indexes:

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Nul
PRIMARY	BTREE	Yes	No	test_type_id	9	A	
test_category_id	BTREE	No	No	test_category_id	9	A	

Project funded by PEPFAR

Philip Boakye	**- Team Lead**
Patricia Enninful	**- Technical Officer**
Stephen Adjei-Kyei	**- Software Developer**

test_type_measure

Table comments: InnoDB free: 0 kB

Column	Type	Null	
test_type_id	int(11)	No	
measure_id	int(11)	No	
ts	timestamp	No	

Indexes: ⊙

Keyname	Type	Unique	Packed	Column	Cardinality	Collation
test_type_id	BTREE	No	No	test_type_id	15	A
measure_id	BTREE	No	No	measure_id	15	A

Database Search scenarios

As a BLIS equipment interface site implementer you will have to write SQL queries in the database to search for IDs to configure your equipment. Here are some example scenarios you may encounter.

Important!	Please note that examples below are just for demonstration purpose. You will need to write your own queries and more complex queried as desired based on your need.

Specimen Types

a) Get the ID of **Whole Blood** specimen

```sql
SELECT specimen_type_id,name
FROM `specimen_type`
WHERE name LIKE 'whole%'
```

> Any word beginning with **whole**

b) Get the ID of urine

```sql
SELECT specimen_type_id, name
FROM `specimen_type`
WHERE name LIKE '%Urine%'
```

> Any word that contains Urine

Project funded by PEPFAR

Philip Boakye	**- Team Lead**
Patricia Enninful	**- Technical Officer**
Stephen Adjei-Kyei	**- Software Developer**

Test Type

a) Get all test types assigned to **urine** specimen type

```sql
SELECT t.test_type_id, t.name
FROM `test_type` t
INNER JOIN specimen_type s ON s.specimen_type_id =
t.test_category_id
WHERE s.name LIKE 'urine'
```

b) Get all test types assigned to `Chemestry` department

```sql
SELECT t.test_type_id, t.name
FROM `test_type` t
INNER JOIN test_category c ON c.test_category_id = t.test_category_id
WHERE c.name LIKE '%Chemestry%'
```

Measures

a) Get ID for measure potassium

```sql
SELECT measure_id, name
FROM `measure`
WHERE name LIKE '%Potassium%'
```

Any word that contains **Potassium**

Test Type Measure

a) Get all measures for test type **Electrolyte**

```sql
SELECT tm.test_type_id, tm.measure_id, t.name AS testname, m.name AS measurename
FROM `test_type_measure` tm
INNER JOIN test_type t ON t.test_type_id = tm.test_type_id
INNER JOIN measure m ON m.measure_id = tm.measure_id
WHERE t.name LIKE 'Electrolyte'
```

Project funded by PEPFAR

Philip Boakye   **- Team Lead**
Patricia Enninful   **- Technical Officer**
Stephen Adjei-Kyei   **- Software Developer**