

Machine Integration

- Mindray BS120
- GeneXpert
- Sysmex XQ320

Mindray BS120

This documentation covers the end-to-end process of BS120 integration, from installation and configuration of necessary software to testing and troubleshooting.

Prerequisites

Ensure the following requirements are met before proceeding with the integration:

- Node.js: Version 12 or later must be installed.
- npm: Verify that npm is installed and up to date (Run `npm -version` in the command prompt).
- Internet connection: Required for installing dependencies (Use USB tethering with Smart phone or ethernet using a Wi-Fi router).

Cloning and Setting Up the Driver

1. Clone the **BS120-CSV** driver from GitHub using this link [MLAB Drivers](#)
2. Move the BS120-CSV folder in the cloned folder to C:// on the computer running the BS120 software
3. Open Command Prompt (cmd) and navigate to the driver folder: `cd C:/BS120-CSV`
4. Install the required dependencies: `npm install`
5. Run the application to verify installation: **node app.js**. If the driver is running correctly, you will see a confirmation message.
6. Stop the application and proceed with the necessary configurations.

Configuring the Driver

1. Open the configuration file: **C:/BS120-CSV/config/settings.json**
2. Modify the following settings as needed:
 - a. IP Address
 - b. Admin Username (lisUser)
 - c. Admin Password (lisPassword)

Exporting Results from BS120 Software

1. Open the BS120 software.
2. Navigate to Results and History.
3. Click the Export button on the right side.
4. Set the export path to: C:/BS120-CSV/outputs
5. Save the exported results.
6. Open File Manager and verify that a CSV file has been generated in the outputs folder.

1. Verifying Data Logging

1. Run the driver again: `node app.js`
2. Export results again from the BS120 software.
3. Check the logs to confirm data is being processed.
4. If logs are displaying correctly, proceed with the next step.

Setting Up Automatic Execution with PM2

1. Install PM2 globally: `npm i -g pm2`
2. Start the driver with PM2: `pm2 start app.js --name="BS120"`
3. Install PM2 Windows Startup (Requires internet connection): `npm install pm2-windowsstartup -g`
4. Set up PM2 startup service: `pm2-startup install`
5. Save the PM2 process list: `pm2 save`

Mapping

The test measure IDs for Renal Function, Liver Function, and Electrolytes are mapped in the configuration file:

C:/BS120-CSV/config/mapping.json

The mapping structure follows this format:

```
mapping.json ×
BS120-CSV > config > mapping.json > ...
1  {
2  "A-AMY" : 188,
3  "ALT"   : 189,
4  "AST"   : 190,
5  "ALP"   : 191,
6  "TP"    : 192,
7  "TC"    : 193,
8  "UREA"  : 194,
9  "GLU-O" : 204,
10 "TBIL-DSA" : 196,
11 "DBIL-DSA" : 197,
12 "UA" : 198,
13 "TG" : 199,
14 "ALB" : 200,
15 "CREA-J" : 239,
16 "LDH" : 236,
17 "P" : 242,
18 "CK" : 244,
19 "A-HBDH" : 287,
20 "R-GT" : 136,
21 "CA" : 241,
22 "MG" : 243,
23 "TBIL-VOX" : 292,
24 "DBIL-VOX" : 293,
25 "GLU-HK" : 195,
26 "HDL-C" : 237,
27 "LDL-C" : 238,
28 "CRE" : 143,
29 "CRP" : 120
30 }
31
```

Each name corresponds to the test name from the BS120 software, and the number represents the measure-mapping-id in IBLIS.

How to Find the Mapping Name?

All test identifiers (names) in the mapping file must be derived directly from the BS120 software, specifically from the SAMPLE REQUEST section used by laboratory technologists (ask the ones in the laboratory if necessary) during test registration.

Critical Guidelines

1. Maintain exact nomenclature as displayed in the BS120 software
2. Preserve all formatting, capitalization, and special characters
3. Do not standardize or "correct" abbreviated test names for instance in case of

Creatine;

CREA-S, CREA-J, CREA, or CREA-SOX (as specified in BS120).

This precision ensures system integrity and prevents potential misidentification of laboratory tests throughout the workflow process.

How to Find the Mapping ID

1. Open IBLIS and navigate to the test you want to map. In the example below, it's liver function tests.
2. Click Enter Results or Edit on a relevant test (e.g., a Liver Function test).
3. Right-click to inspect the HTML element of the measure input field.
4. Hover over the input field, and the measure-mapping-id will be displayed. For this example, ALPU has an id of 309.
5. Copy the ID and update the corresponding test in mapping.json. If not found, add another entry.
6. Repeat the process for all required tests.
7. Save the changes and restart the PM2 process: `pm2 restart BS120`

Final Testing

1. Export results that include an accession number (tests where the barcode was scanned using the small label from IBLIS reception).
2. Verify that the results are correctly fetched in IBLIS by checking corresponding tests bearing the same accession number.

Troubleshooting

If results are still not being fetched correctly, follow these steps:

1. Check the CSV Export
 - a. Verify that the exported CSV file exists in `C:/BS120-CSV/outputs`.
 - b. Open the file to ensure it contains valid data.
 - c. If the file is missing, repeat the export process from the BS120 software.
2. Verify Node.js and npm Versions
 - a. Run the following commands: `node -v`
`npm -v`
 - b. Ensure Node.js is version 12 or later and npm is up to date.
3. Check the Logs
 - a. Run the driver manually: `node app.js`
 - b. Observe the console output for any error messages.
4. Confirm Mapping Configuration
 - a. Open `C:/BS120-CSV/config/mapping.json` and verify that the test names and IDs are correctly mapped.
 - b. Ensure that the measure-mapping-id matches the corresponding test in IBLIS.
5. Restart PM2 Process
 - a. Run: `pm2 restart BS120`
 - b. Check if the process starts successfully.

6. Inspect Network and IP Configuration

- a. Ensure the correct IP address is set in settings.json.
- b. Verify that the BS120 machine and the computer running the driver are on the same network.

7. Check PM2 Logs

- a. Run: pm2 logs BS120
- b. Look for any error messages related to connectivity or data processing.

8. Verify Network Connectivity

Make sure the computer can ping the Ip address of the server where IBLIS is running. This can be done by opening command prompt (cmd) and typing ping [IP_ADDRESS]

If the issue persists, recheck all configurations or for further assistance open an issue on the EMR

Helpdesk.

GeneXpert

Machine Integration with IBLIS -

Setup Guide

This document serves as a step-by-step guide for integrating the GeneXpert machine with the IBLIS system

using the GeneXpert driver. The driver enables automated fetching and submission of test results from

GeneXpert to IBLIS.

Pre-Configuration Requirements

- Ethernet Cable (together with/without USB to Ethernet Adapter): Required for establishing a network connection between the GeneXpert machine computer and the IBLIS server.
- Node.js v18: Required to run the GeneXpert driver.
- Access to the GeneXpert software (on the computer controlling GeneXpert) and IBLIS server.

Step 1: Connect the GeneXpert Machine to the Network

Connect the GeneXpert machine's controlling computer (laptop or desktop) to your facility's network:

- Via Ethernet port, or
- Via an external Ethernet adapter (if the Ethernet port is in use).

If the GeneXpert machine is already connected to the GxAlert network, follow these steps

A. Open Settings

B. Click on Network & Internet

C. Click on Status

D. Click "Change Adapter options. "

E. Right-click the network adapter you want to change the priority for and select its Properties:

F. Select Internet Protocol Version 4 (TCP/IPv4)

G. Click on the Properties button.

H. Click Advanced. In this section, uncheck the box that says Automatic metric and enter a number in

the Interface metric box.

The number you enter here is the priority of the network interface; the lower the number, the higher

the priority. A high number means low priority.

I. Click OK to save the current dialog and the subsequent one. These steps should be repeated for any

other network adapters that need configuration. In this case, the Wi-Fi adapter is given a higher priority than the adapter used for the Internet. This configuration ensures that Windows attempts to

use the primary connection first to reach hosts and automatically switches to the secondary connection if the primary one

NB: Set priority 1 to the GxAlert network adapter and set the network (EGPAF) adapter to 100 so that

GxAlert always has high priority.

Step 2: Verify Network Connectivity

1. Open Command Prompt (CMD) on the GeneXpert machine.
2. Ping the IBLIS server to check connectivity: ping <IBLIS_SERVER_IP>
 - If successful, proceed to the next step.
 - If unsuccessful, try another adapter or network port.

Tip: If the server is unreachable but another workstation (e.g., an All-in-One in the lab) is reachable, you can use the reachable workstation.

Step 3: Install the GeneXpert Driver

1. Check Node.js Version:

```
node --version
```

If Node.js is not installed, use nvm to install it:

```
nvm install 18
```

```
nvm use 18
```

2. Clone the Driver Repository:

```
git clone
```

```
https://github.com/HISMalawi/mlab_drivers.git
```

```
cd mlab_drivers/GeneXpert_Machine_Driver
```

3. Install Dependencies:

```
npm install
```

TIP: (Alternatively, since the servers do not have internet, install dependencies on a working machine, copy the node_modules folder into the driver folder on the server.)

4. Run the Driver:

```
node app.js
```

You should see: "Server is listening!"

Leave the driver running and proceed to the GeneXpert computer.

Step 4: Configure the GeneXpert Software

1. Log in to the GeneXpert software using:

- Username: genexpert
- Password: cepheid

2. Navigate to:

Setup → System Configuration → Host Communication Settings

3. In the Host Communication Settings:

- Enable Host Communication
- Enable Automatic Result Upload
- Under Communication Settings:

Protocol: ASTM

Run Host As: Server

Server IP Address: IP of the machine running the driver (either the server or All-inOne)

4. Ensure Check Status shows a green checkmark (✓).

Step 5: Configure Host Test Codes

For each enabled assay under Host Test Code, click Edit and update as follows:

- EID Assays (e.g., "HIV-1 Qual..."):

Set Result Test Code: EID

- Viral Load Assays:

Set Host Test Code: Viral Load or Viral Load 2 (if more than one exists)

- COVID-19 Assays:

Set Host Test Code: COV-2 or COV-2 2

- TB Assays (MTB-RIF):

Result Test Codes: MTB, Rif Resistance

Add suffix (e.g., MTB-RIF-1) if more than one is enabled

- HPV Assays:

Result Test Codes: HPV 16, HPV_18_45, OTHER HR HPV

After updating all assays, click OK to save changes.

Step 6: Edit Driver Configuration Files

Navigate to: cd

config/

1. settings.json

Update the following:

```
{  
  "lisPath": "http://<IBLIS_SERVER_IP>:8005/api/v1/interfacer",  
  "lisUser": "<IBLIS_USERNAME>",  
  "lisPassword": "<IBLIS_PASSWORD>",  
  "port": 3031  
}
```

2. mapping.json

Match the measure names to their corresponding IDs from the IBLIS database: Inspect the page by right-clicking and clicking select inspect, then click on the selector

Navigate to the GeneXpert result field and get the ID.

Get the ID and add it to the corresponding values in the mapping.json

Step 7: Run and Test Integration

Start the driver:

```
node app.js
```

From the GeneXpert machine, click on View Results, which brings up the most recent generated result,

then click on Upload test

Select a test that was entered from the IBLIS reception, whose barcode was scanned when creating a test in

the genexpert machine, then click Upload

Check the process for the GeneXpert driver. Then type pm2 log [Gene-Xpert id] obtained for the first

command

Example:

```
http://192.168.200.100:8005/api/v1/interfacer?specimen_id=123&measure_id=456&result=Detected&dec=0
```

Ensure that the correct test type and result appear in IBLIS.

Step 8: Set Up Auto-Start with PM2

To ensure the driver starts automatically on reboot:

```
npm install -g pm2
```

```
pm2 start app.js --name genexpert-driver
```

pm2 save

pm2 startup

Run the generated command from the pm2 startup output in the terminal.

User Orientation for Successful Integration

- Users must scan the IBLIS specimen label (Sample ID) during test registration in the GeneXpert software.
- Ensure this is done at the point of creating a new test, where both the cartridge and sample ID are scanned.

GeneXpert Integration with CIHEB EID/VL System via IBLIS

To enable bidirectional integration between IBLIS and the CIHEB EID/VL system through the GeneXpert machine, follow the steps below.

1. Configure application.yml in mlab_api

Copy the ciheb_gex_subscription block from the application.yml.example to the application.yml and update the values as required:

The ciheb_gex_subscription block is like this;

```
ciheb_gex_subscription:  
ciheb_eid_vl_endpoint: "http://<eid_vl_server_ip>:<genexpert_port>/api/v1/subscribe/genexpert"  
app_name: "IBLIS"  
organization_name: "EGPAF"  
iblis_result_endpoint: "/api/v1/interfacer/genexpert"  
iblis_ip_address: "192.168.1.1"  
iblis_backend_port: 8005  
iblis_username: "YourUsername"  
iblis_password: "YourPassword"  
client_type: "other"
```

3. Update the following values:

A. ciheb_eid_vl_endpoint: Replace eid_vl_server_ip with the actual IP address of the EID/VL server, and the port number, should be requested from CIHEB (Gift Malolo)

B. iblis_ip_address: Enter the IP address of the IBLIS server.

C. iblis_backend_port: Confirm that the backend (mlab_api) is running on port 8005. If different, update accordingly.

D. iblis_username and iblis_password: Use valid IBLIS credentials (the same used for logging into the system).

4. Subscribe IBLIS to the CIHEB EID/VL System

From within the mlab_api project directory, run the subscribe command provided, and it should not have

any errors: rake gex_sub:subscribe

This registers IBLIS with the CIHEB EID/VL system, enabling result synchronization.

5. Map Genexpert results sent through the subscription to CIHEB with IBLIS measures

- From mlab_api, tail this log: tail -f log/genexpert.log
- Try sending or uploading results from GenExpert, and the genexpert log should show this below:

Edit the mapping file: vim config/gx_mapping.json

```
{  
  "cov-2":{
```

```
“id”: 190,  
“aliases”:  
}  
}
```

- From the Genexpert log, locate the measure name under the processing data from CIHEB entry.

Add

this measure name to the aliases field in the mapping file. Next, navigate to the Enter Results page

in IBLIS to identify the corresponding measure mapping ID and assign this value under the id field in the mapping file (similar to the standard mapping process used with the machine driver).

Repeat this process for all possible test results generated by the Genexpert.

- Once the mapping updates are complete, restart the mlab_api service by running `sudo systemctl restart mlab_api.service`

- For troubleshooting and validation, refer to the Genexpert log to monitor the processing of incoming data.

Outcome

- Orders placed in IBLIS and tested on GeneXpert will have results automatically transmitted back to

IBLIS via the subscribe endpoint (`/api/v1/interfacer/genexpert`).

- Orders placed in the EID/VL system and tested on GeneXpert will likewise receive results automatically, ensuring seamless integration and synchronization between both systems.

Testing the Integration

- Register a test in IBLIS
- Scan the Sample ID in GeneXpert during test setup
- Run the test and allow it to complete
- Check IBLIS to verify if the result has been correctly fetched and matched

Sysmex XQ320

1. Clone/Copy the xq320 driver from: https://github.com/HISMalawi/mlab_drivers onto the server
2. Navigate to xq320 directory and do: `npm install` .Since servers do not have internet, you can do this on your local machine, make sure node version is the same as that on the server via `node -version` then copy `node_modules` to this directory on the server.
3. Navigate to `config/settings.json` file of xq320 and change `serverIp` to your server ip address and `localhost` on `lispath` to actual server IP. `LisUser` and `lisPassword` to credentials used to log into iBLIS and `serverPort` to 5555
4. Open the file `xq320.js` via `vim xq320.js` and the last line as follows: where there is `settings.serverPort`, it should be 5555 and where there is `settings.serverIP` should be server ip address just the same as those you have set in `config/settings.json`
5. Do run this command to allow firewall on the server: `sudo ufw allow 5555/tcp`
6. In the xq320 folder, run this to start the driver: `node xq320.js` This should start the service and a successful start will show a log that it has started on this ip address and port which are set in the `config/settings.json`.
7. Go to the machine and give the machine network by connecting to it at the back via ethernet cable. Make sure the network is capable of DHCP as the machine automatically picks the IP address, there is no manual setting of network.
8. On the machine software interface, go to settings then TCP/IP connection and set the Host IP as the Server Ip which is the same as that in `config/settings.json` and the port to the port same as `serverPort` in `config/settings.json`
9. Save the TCP/IP connections. Upon saving and going back to main screen and clicking browser, bottom left of the screen has to show a green bar as shown in the picture below. This indicates that there is connection between the service that was run in step 4 and the machine. Without this not communication can happen. This also shows that network between machine and LAN is Ok via the ethernet.
10. Once this is ok, Register one FBC test in IBLIS, take the accession number without the facility code i.e starting from the number, go to the machine, let the lab tech do one dummy test but the test should have the sample id as the accession number of the just registered test(achieved by entered the accession number on the sample ID input field when recording the detail of the sample in the machine before testing it).
11. Once it has finished testing, click on explore on the top navigation on the machine, you will be presented with a list as below image. Select the test with the accession number you registered(should be highlighted blue), then click browserr.
12. Browser will have the details of the test(results and the sample id i.e accession number) as shown in the picture in step 7. Click validate and once its validated, click on Output the HC Outpus. This should send results to the service and you can observe the log of the service running that was done in step 4.

13. If all is good, fetch the result in IBLIS for the test you registered and compare the results if the machine what is in the browser of step 9.
14. Should some results be missing, check the mapping in config/xnseries.json.
15. If every this is OK, make the service to auto start by doing: `pm2 start xq320.js` then `pm2 save`.
16. To make the machine automatically sent the results once the machine finishes testing, do step 15 and 16.
17. Making sure the test is auto validated once testing finishes by going to the machine interface Menu, then settings and choose the settings for auto validation (can be found in operational) as below image. Then click modify settings and set auto validation to be for all samples and save.
18. Making sure the test results are automatically sent to iblis once testing finishes by going to the machine interface Menu, then settings and choose the settings for auto output (can be found in operational) as below image. Then click modify settings and make sure HC is select to as well to be an auto output destination as well and save. That's it. You can try testing an actual sample and verify if its sent to iblis automatically without you clicking on output as was done in step 10.