

---

# Limes-Inventory

*Release 0.5*

**Tony Liu**

**May 20, 2022**



# CONTENTS

<b>1</b>	<b>Contents</b>	<b>3</b>
1.1	Start Here! . . . . .	3
1.2	Example Use Cases . . . . .	4
1.3	Limes Portal . . . . .	5
1.4	Using the Python Client . . . . .	7
1.5	On the Network . . . . .	8
1.6	Contributing . . . . .	9



**Limes** (Lab Inventory Management EcoSystem) is a networked system of data providers with both a *pip installable client* and *web portal*. Linked data providers include eLab and Shamwow. Limes' purpose is to provide a single point of access for all LIMS-related information at Hallam Lab.



## CONTENTS

### 1.1 Start Here!

#### 1.1.1 Workflows Diagram

**Diagram elements:**

- **Items** represent starting conditions, desired outcomes, or required intermediates.
- **Actions** show the steps necessary to obtain indicated items.
  - **Standard** actions can be performed through the *portal* with no installations.
  - **Advanced** actions require the use of the *python client*.
  - **Development** actions require changes to the *code base*.

The above shows the possible high-level actions for working with LIMS data, what each action achieves, and how to chain actions together. For example, to get *barcoded labels* from from *physical samples*, the action *Add Samples* must first be done to get *ELab Records*, followed by the action *Print Labels*.

*Example Use Cases*

---

#### 1.1.2 List of Actions

**Standard Actions**

- Adding Samples
- Bulk Add Samples
- Update Samples
- Printing Labels
- Search Samples
- *Search By Storage Location*

---

**Tip:** View eLab's full documentation [here](#)

---

## Advanced Actions with Python

### Developer Actions

## 1.2 Example Use Cases

- *Processing Environmental Samples*
  - From the Overview
  - *Detailed Breakdown*
  - *Prerequisites*
  - *Steps*

---

### 1.2.1 Processing Environmental Samples

#### From the Overview

---

#### Detailed Breakdown

---

#### Prerequisites

1. A *sample type* must exist in eLab for each type of sample encountered.

A *type* refers to a set of unique descriptors. For example, all Saanich Inlet samples require *location, depth and date of collection*, while all fosmid pools may require details on the *source sample, screening outcomes or gene content information*. Since fosmid pools and Saanich Inlet samples each require a different set of descriptors from each other, they are of different types.
2. A *label template* must exist for each style of label to be printed.

#### Steps

1. **Review the prerequisites above and ensure they are satisfied.**
2. Record environmental samples in an eLab staging area
3. *Get the barcodes of the newly created sample entries* from step 2.
4. *Print labels for the original samples by using their barcodes.* Label the original samples.
5. Filter water sample(s) through Sterivex filters [*small volume, large volume*]



6. Record filters in an eLab staging area. Specify each filter's parent to be the respective sample from step 2.
7. *Get the barcodes of the newly created filter entries* from step 6.
8. *Print labels for the Sterivex filters*. Label the filters.
9. Move all samples digitally from the staging area to their final locations
10. Transfer all samples physically to their final storage locations.

*Back to top*

---

## 1.3 Limes Portal

link: <https://limes-inventory.github.io/>

Log in using your eLab credentials.

---

### Note:

- A connection to the UBC Secure network is required, via VPN, campus wifi, or ethernet cable.
- 

### 1.3.1 Search by Storage Location

*Log in to the portal* and click Search By Storage Location.

Search for a storage location by entering the names of the storage levels to that location, seperated by commas. This need not be too precise. For example,

11, 2, 3

would yield

-80C Freezer #11, shelf 2, rack 3

**Warning:** Be aware of ambiguities. 1, for instance, will match terms like Freezer #01 or Freezer #15, and 8 will match -80C Freezer! *The search is simply matching the name of each storage level.*

The **barcode** and full address of the specified location will appear and all samples in that specific location will be displayed in a table.

---

**Note:** If a storage is not found, pressing the Reload Storages button will have Limes synchronize its remembered storage locations with eLab.

---

Pressing Print will display the barcodes of all samples found.

### 1.3.2 Scanning Labels

*Log in to the portal* and click **Scanner**.

Barcodes can be entered directly into the box: **Enter Barcode Here**. After the specified delay indicated by the box **Delay (ms)**, the barcode will be entered into the table. This barcode will be automatically checked against elab and a checkmark (✓) will appear in the **Added?** column if the barcode was found in eLab.

When ready to confirm the reception of barcodes (and their samples by proxy), select the desired barcodes from the scanned barcodes table and click **Confirm Recieve**. **The MMAP API will then be notified of these barcodes**. The **Info** column will display identifiers for the barcode if the MMAP API was succesfully updated, or continue to display **unknown** on failure.

**Warning:** The MMAP API will only accept confirmation once, so there is no way to tell the difference between a succesful confirmation and a repeat confirmation. Attempts to improve this was met with logistical resistance.

Selected barcodes with retrieved metadata (ex. names or other identifiers) can be copied to the clipboard as a table to be later pasted into a spreadsheet for bulk entry into eLab.

---

**Tip:** Double clicking on any row will open the eLab page for that barcode if found in eLab.

---

---

**Tip:** Use the **Toggle Camera** button to scan barcodes directly using the built-in camera of the device.

---

---

**Note:** The scanned list of barcodes will persist between login sessions and devices for each user, *as long as the server has not restarted*.

---

### 1.3.3 Printing Labels

Ensure that a template for the style of label you wish to print has been created.

*How do I make a template?*

*Log in to the portal* and click **Print**.

In the **Items** field, enter each item to print by pasting its' **barcode** on a new line like so:

```
005000009764829
005000009764830
005000009764831
```

---

**Tip:** Consider using the *Search By Storage Locaton* feature to acquire barcodes of many samples of storage locations like boxes / racks. *Have a look at this example for a walkthrough*

---

---

**Note:** If a barcode is matched to a sample on eLab, the sample's name will be automatically sent to the label, otherwise, only the custom text will be sent.

---

Add additional text by separating it with a comma or the tab character. *Pasting entries directly from a spreadsheet is possible here.*

```
005000009764829
005000009764830, custom text
005000009764831, text for "o.text2", text for "o.text3"
005000009764831 tab separated & from a spreadsheet
```

---

**Note:** It is possible to print *unlinked* labels, though this is **highly NOT recommended**. Simply enter your own barcode or start with a comma to indicate a blank barcode, for example:

```
12345, a lonely label not connected to eLab
, a sad label with no barcode
```

---

The information to be printed for each label will be shown in the table at the bottom.

Pressing the Copy to Clipboard button will allow you to paste the label information into a spreadsheet for editing. It should then be possible to directly paste the updated contents back into the Items box.

A Label Template and Printer must be selected before printing.

---

**Note:** Ensure that the selected template matches the physical labels loaded in the printer.

---

When ready, press Print All and wait for confirmation.

## 1.3.4 Changing Labels

[View the protocol on eLab](#)

## 1.3.5 Troubleshooting

[View Troubleshooting on eLab](#)

# 1.4 Using the Python Client

## 1.4.1 Install

```
(env)$ pip install limes-inventory
```

---

**Tip:** We recommend performing the installation within a virtual environment which can be created by [venv](#) or [conda](#).

---

## 1.4.2 Import

```
from limes import Limes
```

## 1.4.3 Login

```
Limes.Login('username', 'password')
```

## 1.4.4 Search

```
result = Limes.Search('guag')  
result.ToDict()
```

# 1.5 On the Network

## 1.5.1 ELab

A wrapper inheriting `Connections` was created to provide typing for `eLab's API` and integrate it as a `Provider`.

[Click here to go to eLab](#)

## 1.5.2 Server

The server runs on a VM where access is only possible via SSH. It implements the `Provider` interface (specified by the `Connections` module) with additional endpoints for the *portal* and printers. This is meant to be the persistent source of truth for the network, though communications with other network elements need not always go through the server.

The server utilizes both REST and `SocketIO` protocols.

*What's a "Limes Portal" and how do I use it?*

## 1.5.3 Shamwow

An `SshConnection` module allows for Shamwow to be connected as a `Provider` by emulating a bash console and connecting via ssh. While not explicitly implemented, the `SshConnection` module allows Limes to access Shamwow on behalf of a user (using their ssh credentials) and run commands.

## 1.5.4 Printers

A `python driver` handles print requests from the server via `SocketIO`. For each request, it generates a barcode, QR code, and label from an existing template using `Apache OpenOffice Writer` before sending a print command over USB.

*How to print labels*

## 1.6 Contributing

More details coming soon. In the meantime, take a peek at the [repo on Git](#).

### 1.6.1 Architecture Diagrams

**Models**

**Connections**

**Core Server**

**Providers**

---

*Back to top*