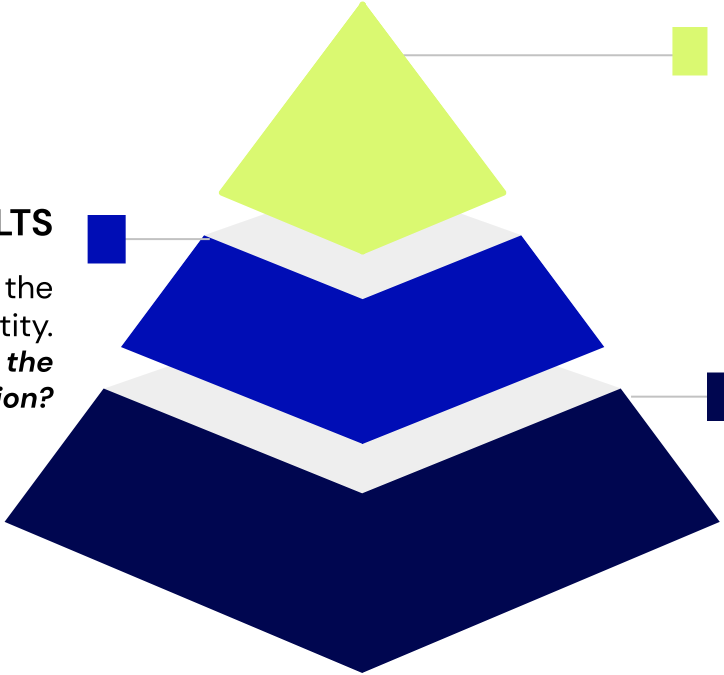
Inventory Foundations and Best Practices

Benchling

Inventory Foundations

Sample Tracking in Benchling



INVENTORY

Tracks physical entities in your fridges and freezers.
Where is it? When was it frozen?

RESULTS

Stores assay data produced after the creation of the entity.
How pure is it? What is the concentration?

REGISTRY

Catalogues a biological sample by capturing its characteristics.
What is it? Where did it come from?

Value of Benchling Inventory

Decrease time to find a sample in the freezer

Know freezer locations of samples simply by signing into your Benchling application

See **how many containers** of a particular plasmid exist

Know **how many stocks you have on hand** and when more need to be generated

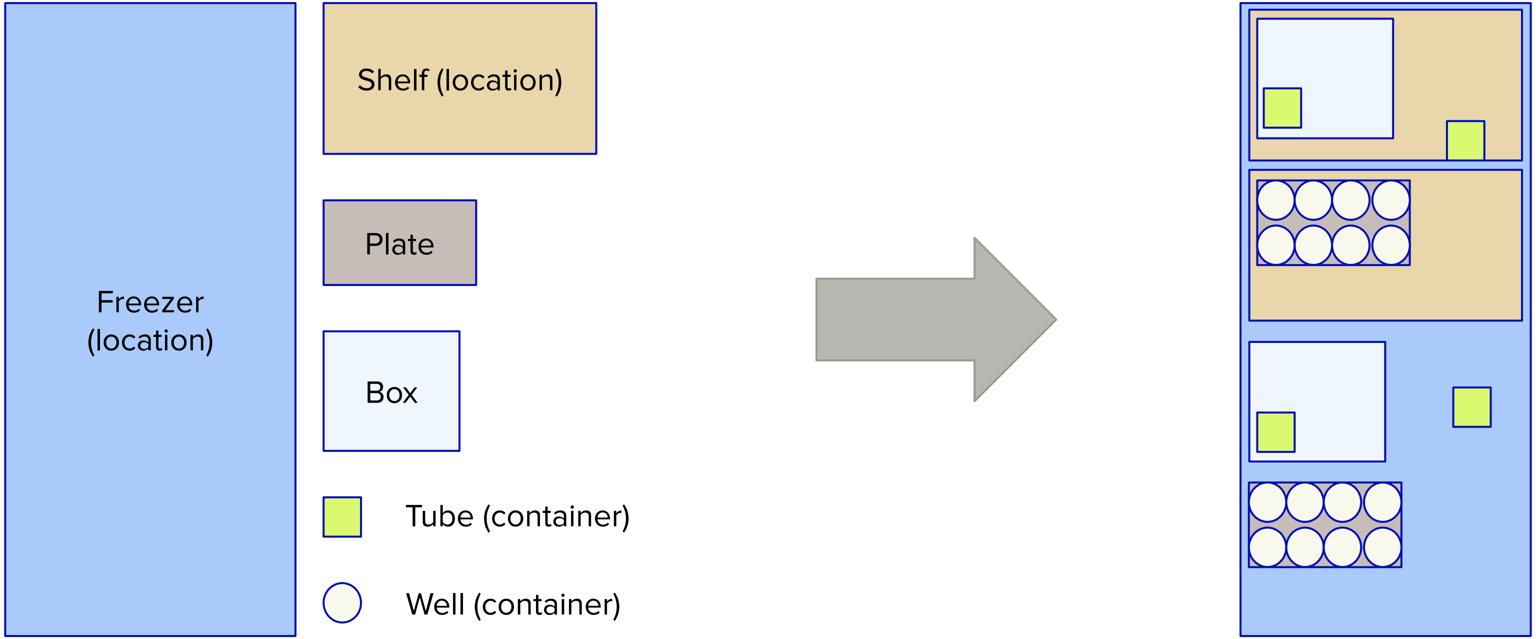
Check out/Check in containers to give others a view of **which samples are available**

Reserve containers for experiment use so others do not take them / spend time looking for them

Trace the **location history** of any sample

Comprehensive Audit Logs answer where has a container been stored

Inventory schemas determine hierarchy in Benchling



Freezer (location)

Shelf (location)

Plate

Box

🟩 Tube (container)

○ Well (container)

Boxes and plates can specify a particular type of container to be held within, such as wells in a plate, and tubes in a box. This is set in the schema of the containing storage item (box and plate)

Storage Schemas in Benchling

Location

- Highest tier of Storage
- Can hold other locations, boxes, plates, and containers

Box

- Can be held in locations
- Can hold containers

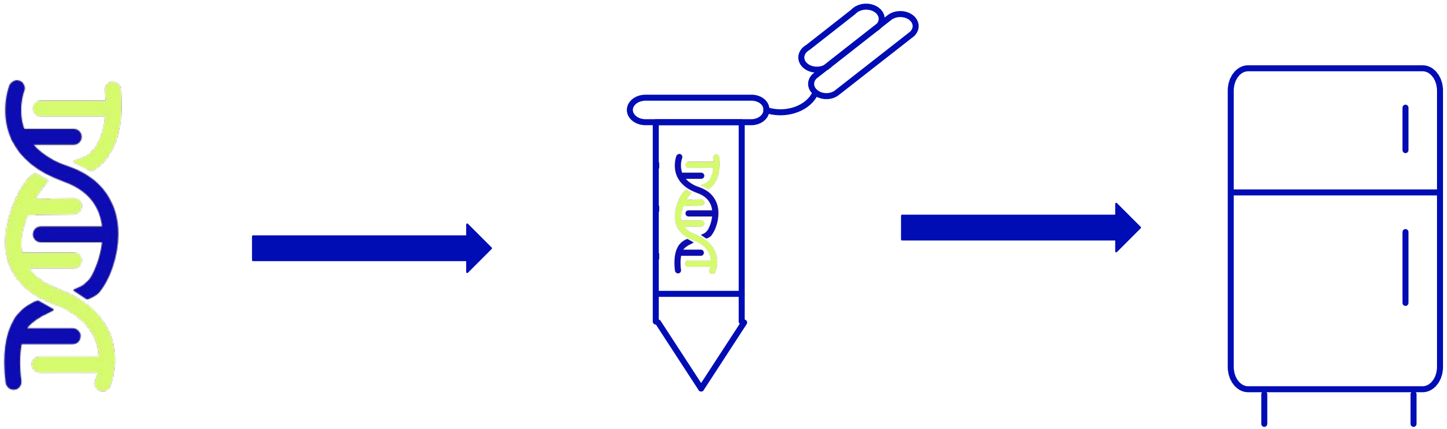
Plate

- Can be held in locations
- Can hold wells

Container

- Can be held in boxes and locations
- Can hold samples/entities

Inventory in Practice



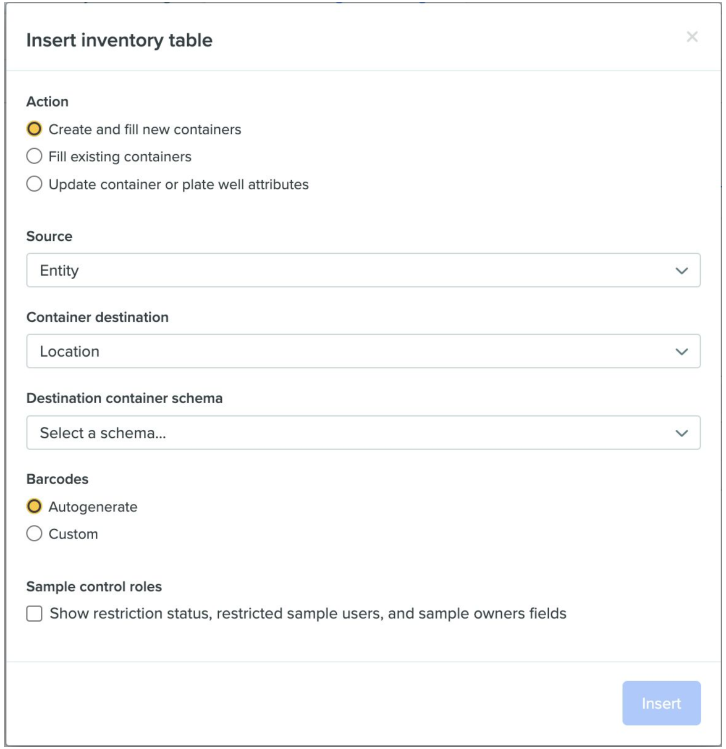
Register a DNA Sample in the Registry to capture its characteristics (sequence, manufacturer)

Add sample to container to keep track of its physical properties (i.e. volume, concentration)

Specify the location of container within boxes, shelves, or fridges so scientists can locate it easily

Inventory Best Practices

Best Practice for Users – Using Inventory Tables to Create and Fill Containers



Insert inventory table ×

Action

● Create and fill new containers

○ Fill existing containers

○ Update container or plate well attributes

Source

Entity

Container destination

Location

Destination container schema

Select a schema...

Barcodes

● Autogenerate

○ Custom

Sample control roles

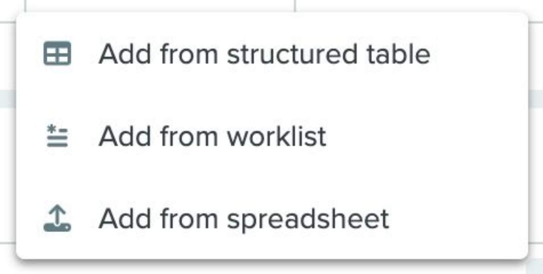
□ Show restriction status, restricted sample users, and sample owners fields

Insert

When Inserting an Inventory Table into a Notebook Entry or Template, you designate the inventory action you want to perform.

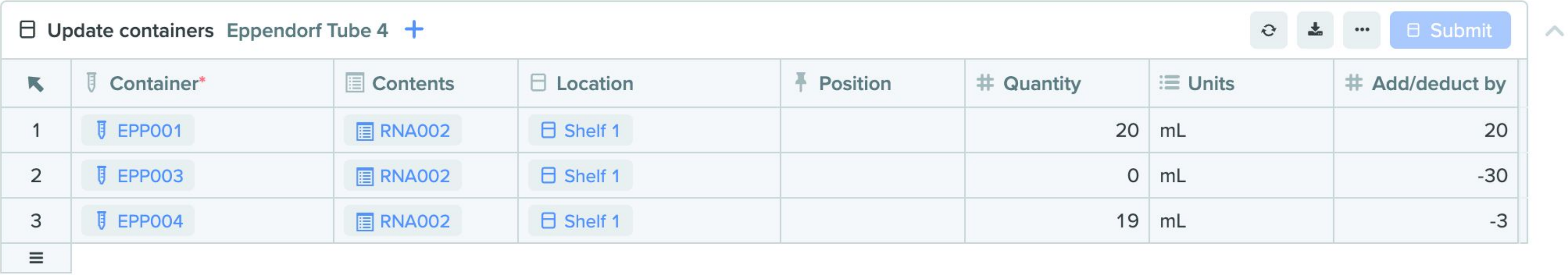
Opting to **“Create and fill new containers”** allows scientists to easily document samples created as part of a scientific workflow in one easy step.

Scientists can also **“Add from structured Table”** to pull samples registered in a previous table from that same entry, further saving them time and hassle.



Best Practice for Users – Using Update Tables

Update Tables allow users to quickly document the amount of material consumed, all in the context of an experiment. Updates created in this table are synced to the rest of your inventory, allowing other users to quickly gauge how much of a particular resource is left.



	Container*	Contents	Location	Position	Quantity	Units	Add/deduct by
1	EPP001	RNA002	Shelf 1		20	mL	20
2	EPP003	RNA002	Shelf 1		0	mL	-30
3	EPP004	RNA002	Shelf 1		19	mL	-3

Like all other tables, they can be inserted into Notebook Entries, Templates, and Subtemplates so users don't have to ever leave their entries or create them from scratch each time.

Storage Best Practices for Users and Admins

Users

What type of items belong in Inventory?

- Inventory is best used for locations that store long-term stocks (>24 hrs)
- Do not use Inventory for locations with transient items (stored for <24 hours) and/or undefined partition

When should items be logged into inventory?

- Enter your entities into Inventory immediately before or immediately after you transfer the sample into its storage location

What should be done to represent removed samples?

- When removing samples, immediately archive the container so other users will know that you've used up the aliquot

Admins

How should you configuration Inventory?

- Fewer location and container schema types makes inventory easier to configure
- Limit the number of sub-locations in your hierarchy to avoid repetitive clicking steps

How should you import backlogged data?

- Locations & Sublocations (one schema at a time) -> use UI / spreadsheet
- Boxes (multiple at a time) -> use spreadsheet
- Containers (with and without entities) -> use spreadsheet or Transfer table

Best Practices for Sample Intake

Action	Value Add
Bulk Container Creation (via UI)	Expedite bulk data import of information stored in spreadsheets into Benchling – useful for legacy data import and large purchases of inventory
Use Inventory Tables in Notebook Entries and Templates to Create Containers	Create, fill, and specify location of new containers for the lab while documenting the process in a notebook entry
Create or update containers using API Endpoints (Developer Platform)	Use API Endpoints alongside custom programming to perform tasks outside of Benchling notebooks. Opens avenues for automation and higher throughput processes

Best Practices for Sample Management

Action	Value Add
Print labels using Zebra or BarTender	Improved sample tracking by being able to more quickly identify samples, and scan them into search, tables, or worklists
Use the Check-in, Check-out, Reserve Features	Improve lab coordination of sample usage by designating which samples are available for use
Update Container Volume & Archival Status	Specify if a container's contents were used, expired, contaminated, or were retired using Inventory Tables or the UI to keep track of exactly how much of a sample remains
Leverage Audit Logs for detailed sample records	Trace lifecycle of a sample (container creation and updates made to the sample) and utilize them for regulatory purposes where necessary
Utilize Sample Ownership features	Designate clear owners of samples, and create permissions around who can perform specific actions on samples

Growing with Inventory

Best Practices for Different Stages

Basic Focus on Adoption	Set and communicate usage goals and expectations Configure Inventory hierarchy Train new scientists/users on Inventory fundamentals
Intermediate Focus on Optimization	Create new and update existing Inventory schemas Create and maintain Inventory hierarchy Set permissions of Inventory Communicate new Benchling Inventory features to users
Advanced Focus on Technical Maturity	Create and maintain third-party integrations Develop/evolve data model and Inventory compliance to reflect scientific process Change management for implementation of new functionality or processes Build analytical dashboards to visualize data and support decision making

Resources for Inventory

Create Inventory for Benchling Entities via UI

Add an entity to a container Help Article

Bulk Upload Samples & Containers

Upload inventory information using spreadsheets Help Article

Create Inventory using Inventory Tables in the Notebook

Help Article on Inventory Tables

Create Containers & Add Samples to the Inventory using the API

API Overview & Tutorial