

# **mother\_batches**

A database requires everything to have a unique identifier. It can supply auto-incremented ids, but these are meaningless outside of the database itself and hard for a user (me) to visually parse and interpret.

It's preferable for the ids to have some amount of human-readable meaning, so that a human can more clearly understand the data they're working with and catch errors as they happen. Not all situations offer a way to create meaningful ids, however, and I didn't want to force it when it didn't make sense. Thus, there are 3 main types of unique identifiers used across the database:

## **Batch Numbers**

All stages have batches, and all batches must be distinguished from each other somehow. The pre-LIMS paperwork system wasn't quite as rigidly consistent as this database about the formatting, but it always broke along the lines of date and type (whether specific single microbe or combined recipe).

Before the LIMS system was implemented, the Final Product had two different batch numbering systems in place: - For most products, a simple 6-digit date in the format YYMMDD, plus a letter if multiple batches of the same type were produced in a day, so that two batches of MicroAZ-IF produced on October 14 2023 would be **231014A** and **231014B** - For TerraMax Dry, the letters **BDSC** plus a number that increased by 1 with every new batch, so that **BDSC217** would be followed by **BDSC218**.

There was apparently a plan at some point in the past to shift everything to match the TerraMax Dry pattern and create unique codes for all the products off the same pattern.

This plan broke down because not all products could be so neatly translated, but I thought the core idea was solid. The date-based code didn't differentiate between products because it didn't need to; a person could look at the package and determine what the product was. But in a database this would translate to a split key, needing both the batch number and the product together to have a fully distinct identifier, and that seemed unnecessarily messy.

Instead, I used the basic pattern from the **BDSC** batch numebrs to produce a set of product codes that I could add to as I needed. (See list of all product codes [here](#))

I then combined this code with the 6-digit date number to create a final product batch number that would be both unique and meaningful. Those MicroAZ-IF batches would now be **AZ-231014A** and **AZ-231014B**.

Then I extneded that same logic to all other stages, and built batch numbers around the format:

**type-stage-date**

Where **type** would be the code for the microbe, stabilized lquid or bomb type (again see the full list of codes [here](#)), and **stage** would be a one or two character abbreviation.

- **MS** : Mother Stock
- **WS** : Working Stock
- **1** : Primary Starter Culture
- **2** : Secondary Starter Culture
- **F** : Ferment
- **SL** : Stabilized Liquid
- **DMP** : Dry Microbial Powder
- **B** : Bomb

## Type Codes

### Product Codes

Type codes were required in order to construct meaningful unique batch numbers for batches of different products produced on the same day. I began constructing them according to an earlier attempt to create product codes according the the format: - microbial contents: **B** for Bradyrhizobium - liquid/dry: **D** for Dry - application type: **S** for Seed Treatment - conventional/green: **C** for conventional

This worked well once for **BDSC** but immediately breaks down when you consider products like Vertex that contain more than one microbe. But it was simple enough to fix by just using **V** instead, ad assuming that you can use whatever letter makes the most sense for other products.

## Integer IDs