Nescent

Feeding Experiments End-User Database

Editing Help Text

October 2014

Prepared by Squishymedia, Inc
116 NE 6th Ave, Ste 300
Portland OR, 97232

For more information, please contact

Kelsey Friedman
503 488 5951
kelsey@squishymedia.com

Updating Field Text in FEED2

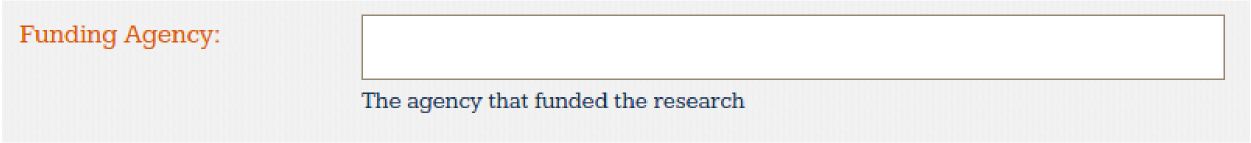
Brief instructions for technical users

What kind of text can be updated?

This document describes how to update **labels** and **help text** for most fields. These are displayed in two different ways in FEED2:

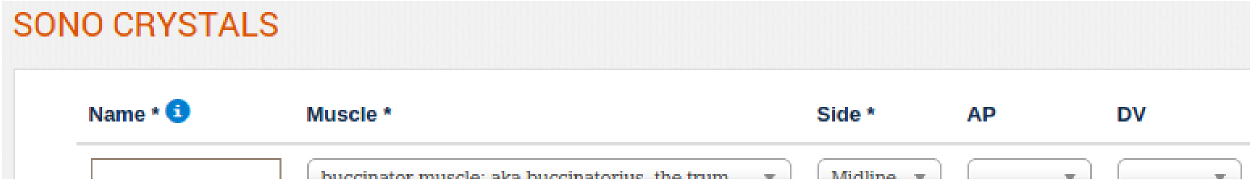
- **Regular layout**: The **label** is in orange to the left of the field; the **help text** is displayed below the field.

  For example: (“Funding Agency” is the **label**, “The agency that funded the research” is the **help text**)



- **Tabular layout**: The **label** is displayed in bold in the column header; **help text** is displayed in a tooltip (with an “i” icon) if specified.

  For example: (the “Name” **label** has a **help text**; other fields do not)



No matter how they are displayed, **labels** and **help texts** are edited in the same way.

Where does the text come from? Which files do I edit?

There are currently two primary files which contain labels and help texts.

- **src/feeddb/feed/models.py** contains most texts
- **src/feeddb/feed/extension/forms.py** contains overrides for a few fields; as a side effect, the labels and help texts for these fields are defined here.

As of this writing, these are the only fields which are overridden in **extension/forms.py**:

- “Notes” on strain sensors

- “Sensor Types” on experiments

All other fields use labels and help text from **models.py**

How do I make the changes in **models.py**?

This file contains a block for each entity type (known as “model” in Django) that starts with **class *<type>***. Each block defines the fields for that type, some metadata, and optionally some custom behaviors. For example, the block that begins with **class Experiment** looks like this:

```
class Experiment(FeedBaseModel):
   title = models.CharField(max_length=255)
   bookkeeping = models.CharField("Bookkeeping", max_length=255,blank = True, null=True, help_text =
BOOKKEEPING_HELP_TEXT)
   study = models.ForeignKey(Study)
   subject = models.ForeignKey(Subject)
   start = models.DateField("Start Date", null=True, help_text=DATETIME_HELP_TEXT)
   end = models.DateField("End Date", blank = True, null=True, help_text=DATETIME_HELP_TEXT)
   ...
```

Below the fields are some functions (blocks beginning with “def”) and other python code; they can be safely ignored.

Each line consists of three parts:

machine_name = ***FieldType***(***field_arguments***)

The **field_arguments** consist of a label (the first item, unnamed), the help text (named as “help_text=...”), and other misc details (also named in the list). If the label is omitted, Django will generate a label based on the machine name. If the help text is omitted, no help text will be displayed.

All texts must be enclosed in quotes; either double quotes or single quotes work. If you need to use a quote mark within the text, you can precede it with a backslash.

A couple examples:

- This field will be labeled “Title” and has no help text:

  ```
  title = models.CharField(max_length=255)
  ```
- This field will be labeled “Experiment Title”:

  ```
  title = models.CharField(“Experiment Title”, max_length=255)
  ```

- This field will be labeled “Experiment Title” with a help text:

  ```
  title = models.CharField(“Experiment Title”, max_length=255, help_text=”Please provide a descriptive
  name for your experiment.”)
  ```

- This field will be labeled “Bookkeeping” and has a help text defined earlier in the file:

  ```
  bookkeeping = models.CharField(“Bookkeeping”, max_length=255,blank = True, null=True, help_text =
  BOOKKEEPING_HELP_TEXT)
  ```

  The help text is defined on line 15:

  ```
  BOOKKEEPING_HELP_TEXT = 'Enter any text required for lab bookkeeping concerning the Study here'
  ```

How do I get my changes to show up?

In order for changes to any *.py file to show up on the site, you need to force the server to recompile the python files. If you are deploying with mod_wsgi on Apache, this means you need to run `touch src/feeddb/wsgi.py`. Alternatively, you can restart the HTTP server service. If you have trouble, consult your local sysadmin for site-specific help with this step.

If you make a syntax error when editing models.py, the site will stop working. You should always make changes like this in a staging or testing environment and use version control. Failing that, keep a backup of the last know working **models.py** so you can restore it if needed.

What should I *not* do in models.py?

This document does not describe how to:

- Remove a field
- Add a field
- Change the type of a field
- Make a field required or not required
- Make any other changes to the way a field is stored
- Make any other changes to the way a field is displayed

Each of these tasks requires additional steps not described here.