# Managing Entities in Central

> ⚠️ **Warning**
>
> This is an *experimental preview*! Please do not use it for real projects unless you are comfortable with things going wrong, or features changing in future releases. It is very unlikely you will ever lose data using Entities, but you may end up with *unexpected* data.
>
> This a preview because:
>
> - You can't yet edit or delete Entities
> - Performance is poor when managing more than a few thousand Entities
> - The Form specification and API may change
>
> You should also be aware that currently, all devices will always download all Entities. This may not be acceptable to you due to privacy concerns.
>
> All of that said, if you have thoughts and feedback on Entities, please post on the [discussion forum](#).

New in Central as of version 2022.3, **Entities** are a powerful new way to automatically bring the information you are collecting right back to your devices and Forms in the field.

In ODK today, you can attach existing data during the Form creation process, often through `choices` sheets or `.csv` files. Once this data is loaded into the Form, you can use it as the source for selection choices, to prepopulate data fields in the Form, or validate new input. A list of districts, for example, can be used as choices for selection, and information about each district can then be shown to the user or checked against. If you are new to these techniques, you can learn more about them on the [Form Datasets](#) page.

This is wonderful if this data is already known when you are publishing your Form, and if it doesn't change over the course of your project. In many cases, however, the data being collected today for the project *is* the data you need tomorrow for the same project. Think about seeing a patient for the first and then second time, or registering and then revisiting a survey site for a follow-up. These kinds of workflows are called [Multiple Encounter](#) or longitudinal workflows.

If you have this kind of workflow, you can use the registered data items in a follow-up Form without using Entities, but it requires some work. You can download the Submission data from the registration Form, and then you can attach that data to a follow-up Form, as described above. However, the data available to devices in the field is only as current as your latest follow-up Form update. Some people write programs that automate this process over the API but this requires technical skills that many projects don't have.

Now there is a way to automate this process from within Central itself.

## Introducing Entities

First, some definitions: each item that gets managed by an ODK workflow is called an **Entity**. A collection of Entities is called a **Dataset**. A `districts` choices list, or a file `patients.csv`, these are Datasets. So each district would be an Entity in the `districts` Dataset, and each patient would be an Entity in the `patients` Dataset.

Next, a new feature. You can now teach Central that some of your incoming Submission data should create Entities in a Dataset created and managed for you by Central under whatever name you assign it—— `patients`, for example. This Dataset can then be attached by that name (*(1)* below) to your follow-up Forms, exactly like you would manually attach a [CSV Dataset](#).
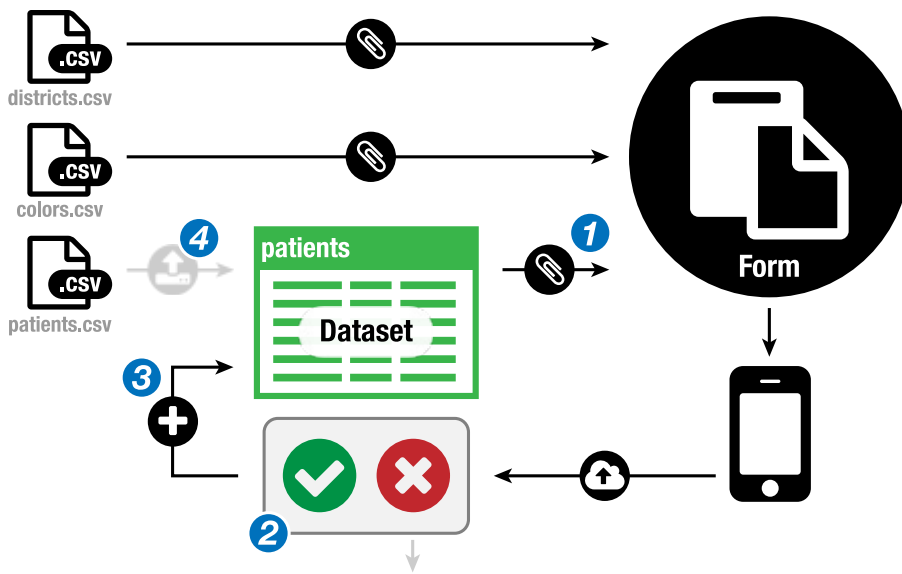
Using this functionality, you don't have to create or upload `patients.csv` yourself, nor update it when new patients are added. When Submissions arrive and are approved *(2)*, Central will create *(3)* new Entities that will appear in patient follow-up Forms as soon as those Form updates are received.

> ✏️ Note
>
> For now, you can only add a new **Entity** to the Dataset for each Submission, but in the future you will be able to update existing Entities, or create multiple Entities at once. Also, there is not yet a way to directly upload data you may already have into a Central-managed Dataset *(4)*.

If you are interested in seeing how Entities can fit into your workflow right away, we recommend following the Quick Start guide in this next section, where you will upload a tree registration Form and a tree follow-up Form we have created already and see how trees are created by one Form and appear in another.

# Quick Start

We recommend watching the video below once or twice to get an overview of how Entities work in Central. Once you have that context, proceed with the steps below to try it yourself!

In this Quick Start, you will upload two Forms we have already authored for you: the Tree registration Form and the Tree follow-up Form. You will step through the registration of a new tree and the process of following up on that tree in a later Form.

First, let's prepare the Forms for use with Central.

1. Create a new Project, if you can. It can be easier to see what's going on without other Forms in the way.
2. Upload the Tree registration Form. Notice how it has seen from the Form definition that the Form creates Entities in a Dataset, and populates three Entity Properties.
3. Publish the Form. As the dialog says, this action creates the `trees` Dataset within the Project you created.
4. Now, upload the Tree follow-up Form.
5. Switch over to the Form Attachments tab before you publish this Form, and notice how the Form wants a data file called `trees.csv`, but that Central has already linked the Dataset `trees` (created when you published the registration Form in step 3) and you don't need to upload any data here.
6. Go back to the Draft Status page and publish this Form. Because this Form only *consumes* `trees`, it does not make any changes to any Entity Properties.
7. Now, create an App User within this Project, allow it to access both of these Forms, and load them up into Collect or some other ODK-compatible client.

Next, let's see these Forms working together.

1. First, fill out and submit the Tree registration Form. Be sure to choose a species and specify a Tree circumference. Use 100 if you are not sure what to fill in.
2. Go back to Central. You can download the `trees` Dataset under the Datasets tab in your Project.
3. But your tree isn't there. It does take a moment sometimes to create an Entity from a Submission, but in this case it's because we're not done yet. An Entity will not be created until you *approve* the submission.
4. Go to the Trees registration Form Submissions page, and approve your tree.
5. Open the Submission details page for that Submission by putting your mouse on its row and clicking More.
6. You should now see a record of your approval, as well as of the creation of a new Entity based on the Submission. If you don't see the Entity yet, wait a second and refresh.
7. You can try downloading the `trees` Dataset again if you want to see your tree there.
8. Now go back to Collect and update your Forms to fetch the new data.
9. Next, fill out the Tree follow-up Form.
10. Your tree is here! Choose it.
11. Report a new circumference that is smaller than the old one. This is probably not a good idea for a tree. See how the Form warns you about this problem, based on the data you'd put into the registration Form.
12. Correct the circumference to a larger number, and submit the Form.

That's it! The follow-up Form only creates normal Submissions, so you can access the data it collects like any Form.

# Entities in v2022.3

If you skipped the Quick Start above, we suggest you go back and give it a try. You will learn hands on with Central a lot of what will be described below.

In these early versions of Entities, you cannot create a Dataset directly through the Central website. To begin using Entities, you will need to author a Form which defines them. By uploading a Form that specifies the fields in a Submission that should be used to create a new Entity, and the name of the Dataset these new Entities should go to, you will prompt Central to create the Dataset. You'll be able to see the Dataset information Central recognized in your Form once you upload it.

When you publish this Form, the new Dataset and/or new Entity Properties will be created for you automatically within the Project. You can learn more about authoring these kinds of Forms in the sections below.

> ✏️ Note
>
> In this version of Entities, a Submission must be approved before an Entity will be created from it. In future versions, you will be able to choose to create the Entity immediately when the server receives the Submission.

To see this new Dataset and download data from it, visit the `Datasets` tab on the Project page. In future versions, you

will see many more controls and more helpful information than you do now.

To use data from a Dataset in another Form, you can refer to it by `NAME.csv` where `NAME` is the name of your Dataset. When you upload that Form, you should see on the Form Attachments tab that the file has been automatically linked to the Dataset. You can always override this connection by uploading your own data file to use instead. This does not affect the Dataset itself, your file is used *instead* of the Dataset for that Form only.

## Testing Forms with Entities

Checking that your Forms are working together the way you expect is challenging with Entities. You can create a Draft of a Form and use it as a safe space to try out Form definitions and see resulting Submissions. But Datasets reach *across* Forms. They live alongside Forms within the Project. How do Form Drafts related to the same Entities connect together?

It's very confusing, and Drafts currently don't handle this question very well. A future version of Central will provide better answers.

For now, *Datasets work on published Forms and Submissions only*. The creation of Datasets or new Entity Properties only occurs at the moment you publish the Form. Only real form Submissions create Entities. This means that you can't test the usage of Datasets in follow-up Forms until real Entities have been created.

To try the end-to-end workflow across multiple Forms, we recommend creating a temporary project just for testing. You can publish all the Forms, create Entities for testing, and make sure that all the Forms work well together.

You can also try a follow-up Form Draft by manually creating a CSV of sample Entities and then attaching it to your Draft, as described in [Forms With Attachments](#). When you have verified the logic of the follow-up Form and are ready to publish it, you can change the link from the CSV to the desired Dataset.

# Creating Forms to use Entities

Central does a lot of work to help you manage Entities, but at least for now the only way to ask it to do so is to create a Form that describes how.

In the following section, we describe how to author Forms that create new Entities, also known as registration Forms. After that, we discuss Forms that use Entities, also known as follow-up Forms.

## Build a Form that creates Entities

You'll start by building a Form that creates new Entities in a Dataset called `trees`. When you publish this Form, a `trees` Dataset will be created for you. When a Submission to this Form is approved, an Entity will be created in the `trees` Dataset from data in the Submission. These types of Forms are often referred to as registration, enrollment, intake or discovery Forms.

### Define the structure of the Entity-creating Form

Start by defining the structure of your Form in [XLSForm](#) using the standard Form building blocks. For example, [this form](#) is used to register trees:

**XLSForm**

<div align="center">survey</div>

| type | name | label |
|------|------|-------|
| geopoint | location | Tree location |
| select_one species | species | Tree species |
| integer | circumference | Tree circumference in cm |
| text | intake_notes | Intake notes |

Test your Form to make sure it works and collects the data that you need.

### Specify the Dataset the Form should save Entities to

Add a new `entities` sheet to your XLSForm. This is where you will specify your Dataset's name, under `dataset`.

Each Entity will be automatically assigned a unique name based on one or more Properties from the Entity data, just like the `instanceName` on submissions. The `label` field here is where you provide the expression to use as a label for each Entity in the Dataset.

The Dataset name will be used by Central to uniquely identify that Dataset. If a Dataset with the name you specify already exists in Central, this Form will create Entities in that existing Dataset. If Central doesn't yet have a Dataset with the specified name, it will be created.

The label expression can use any field in the Form.

**XLSForm**

| entities | |
|---|---|
| **dataset** | **label** |
| trees | concat(${circumference}, "cm ", ${species}) |

## Specify the Form fields that are saved to Entities

If you think of your Dataset as a spreadsheet, each row represents an individual Entity and each column specifies an Entity Property.

You define Entity Properties by adding a `save_to` column to your XLSForm. You then put an Entity Property name in the `save_to` column for each Form field that you would like to save for use in follow-up Forms.

**XLSForm**

| survey | | | |
|---|---|---|---|
| **type** | **name** | **label** | **save_to** |
| geopoint | current_location | Tree location | geometry |
| select_one species | species | Tree species | species |
| integer | circumference | Tree circumference in cm | circumference_cm |
| text | intake_notes | Intake notes | |

If you'd like to check your work, you can compare with this example form, with the `entities` sheet and `save_to` information.

When you publish this Form on Central, the `trees` Dataset will be created for you.

# Build a Form that uses Entities

Your `trees` Dataset can now be attached to any Form using `select_one_from_file` or `csv-external`.

**XLSForm**

| survey | | | |
|---|---|---|---|
| **type** | **name** | **label** | **calculation** |
| select_one_from_file trees.csv | tree | Please select a tree | |
| calculate | prior_circumference | | instance('trees')/root/item[name=${tree}]/ci |
| integer | circumference | The circumference was previously measured as ${prior_circumference}cm. Please enter the current circumference in cm. | |

You can see the full XLSForm here.

The same Dataset can be used in many different Forms. The concepts and patterns described in the data collector workflows and the Form Datasets sections apply to server-managed Datasets as well.