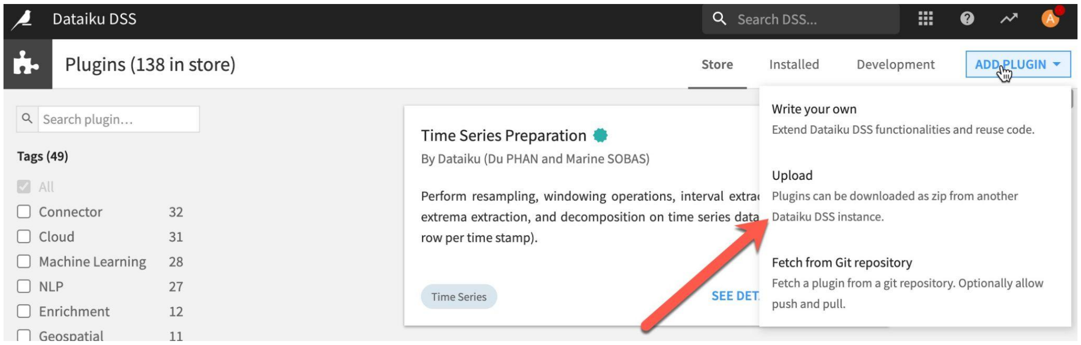
-T-H-R-E-A-D-

v1.1 Installation Instructions

1. In your Dataiku Design node, navigate to the Plugins page. From the “Add Plugin” menu, select “Upload”



Time Series Preparation

By Dataiku (Du PHAN and Marine SOBAS)

Perform resampling, windowing operations, interval extrac extrema extraction, and decomposition on time series data row per time stamp).

Write your own

Extend Dataiku DSS functionalities and reuse code.

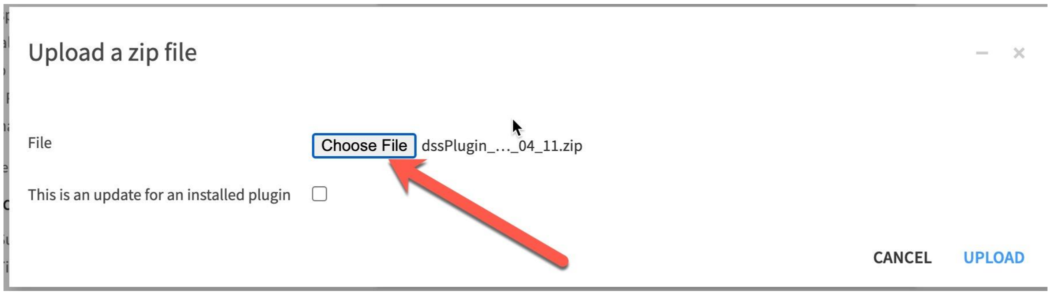
Upload

Plugins can be downloaded as zip from another Dataiku DSS instance.

Fetch from Git repository

Fetch a plugin from a git repository. Optionally allow push and pull.

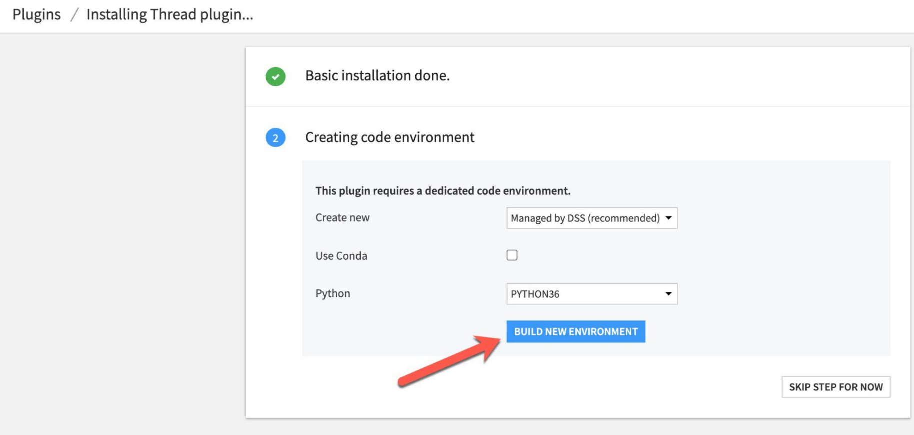
2. Select the provided .zip file



File Choose File dssPlugin_…_04_11.zip

This is an update for an installed plugin □

3. After the upload has completed, you will be prompted to build a new code environment. This environment can use Python versions 3.6 or 3.7



Plugins / Installing Thread plugin...

Basic installation done.

2 Creating code environment

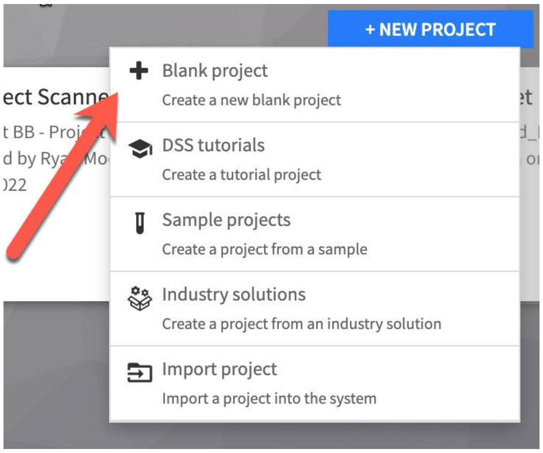
This plugin requires a dedicated code environment.

Create new: Managed by DSS (recommended)

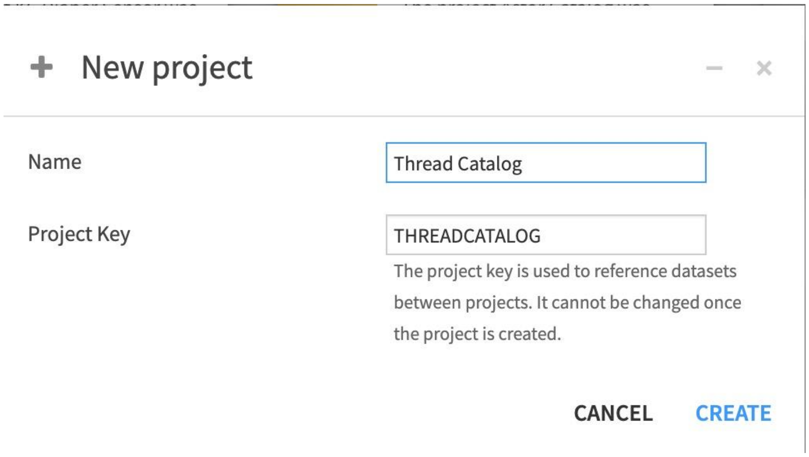
Use Conda: □

Python: PYTHON36

4. Create a new blank project in Dataiku. Suggested name is “Thread Catalog”



+ NEW PROJECT
Blank project
Create a new blank project
DSS tutorials
Create a tutorial project
Sample projects
Create a project from a sample
Industry solutions
Create a project from an industry solution
Import project
Import a project into the system



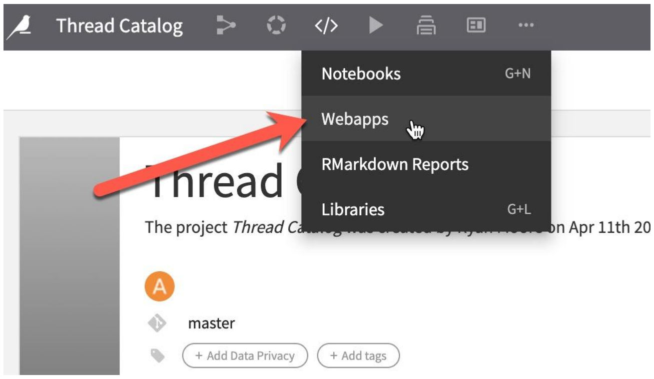
New project

Name: Thread Catalog

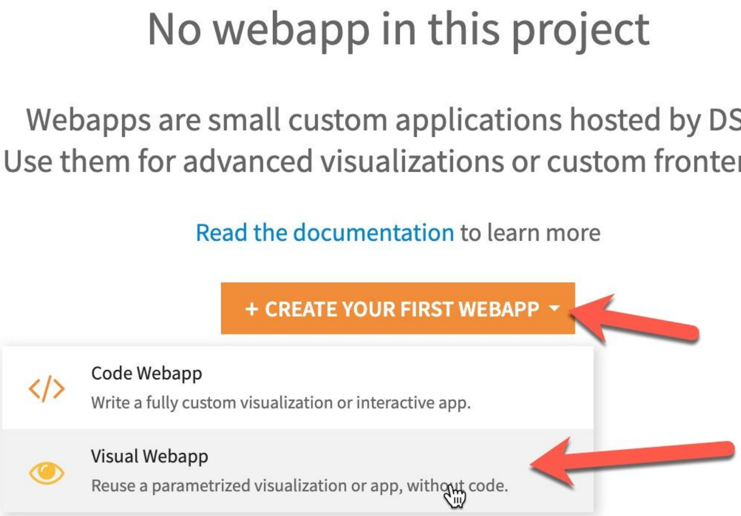
Project Key: THREADCATALOG

The project key is used to reference datasets between projects. It cannot be changed once the project is created.

5. In this new project, navigate to the “Webapps” landing page.



6. Click “Create your first webapp” and select “Visual Webapp”



No webapp in this project

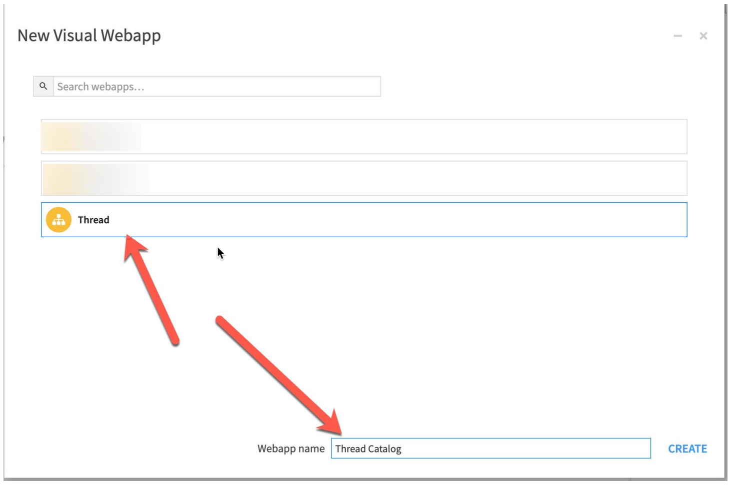
Webapps are small custom applications hosted by DSS.
Use them for advanced visualizations or custom frontends.

Read the documentation to learn more

Code Webapp
Write a fully custom visualization or interactive app.

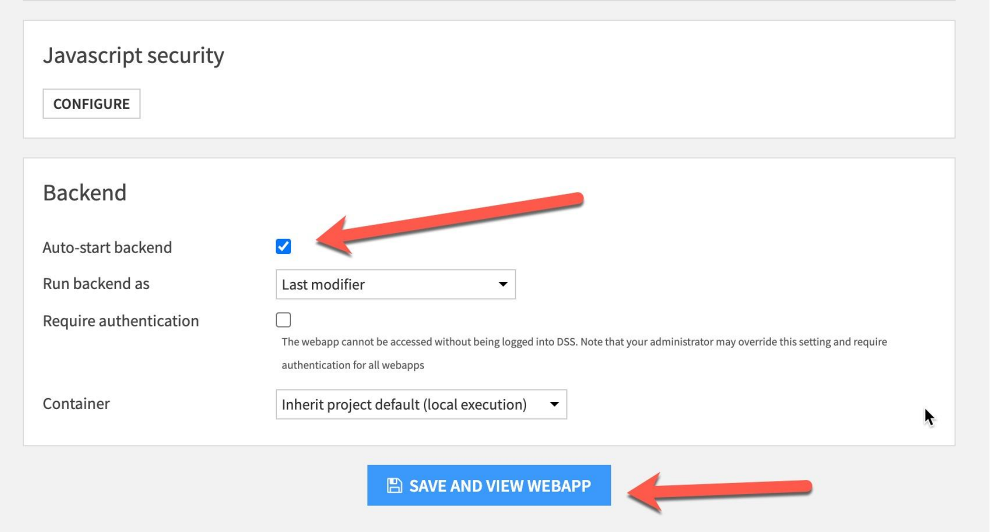
Visual Webapp
Reuse a parametrized visualization or app, without code.

7. Select the “Thread” plugin option and name this new Webapp “Thread Catalog”



Webapp name: Thread Catalog

8. In the following page, select “Auto-start backend”, then “Save and view webapp”



Javascript security

Backend

Auto-start backend ☑

Run backend as: Last modifier

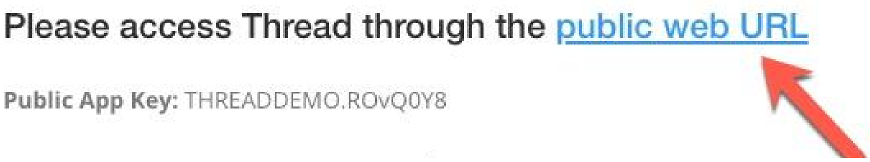
Require authentication ☐

The webapp cannot be accessed without being logged into DSS. Note that your administrator may override this setting and require authentication for all webapps

Container: Inherit project default (local execution)

For “Run backend as”, select a user with proper privileges. Your Thread installation will be able to index all projects accessible by the selected backend user.

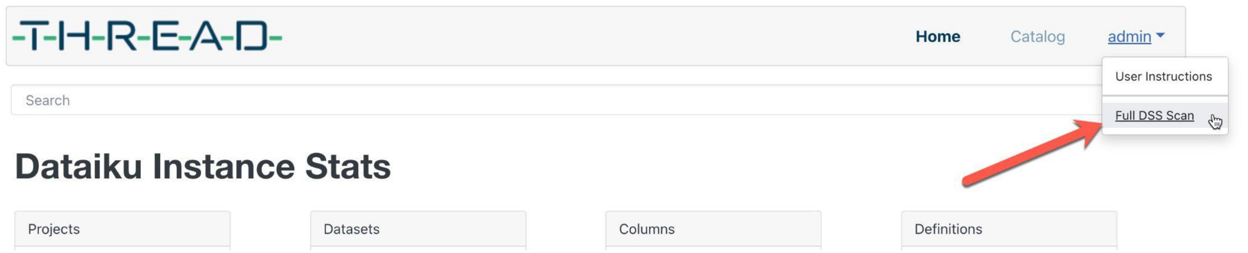
9. Navigate to the link on the Thread plugin page from Step 8.



Please access Thread through the public web URL

Public App Key: THREADDEMO.ROvQ0Y8

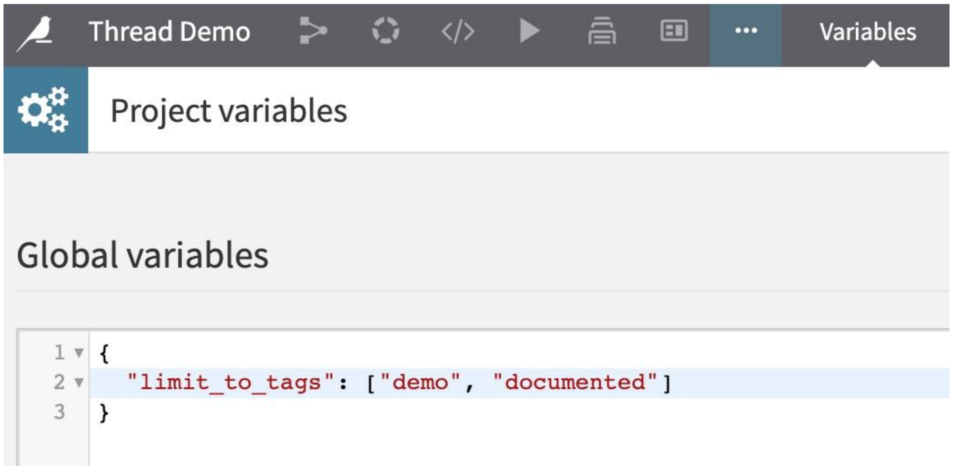
10. Click the “Scan” icon in the Thread UI to start the Dataiku server scan



Dataiku Instance Stats

NOTE:

Indexing can be limited to specific projects using project tags. Specify the tags to be included by adding a “limit_to_tags” project variable to the Thread project as shown in the following example:

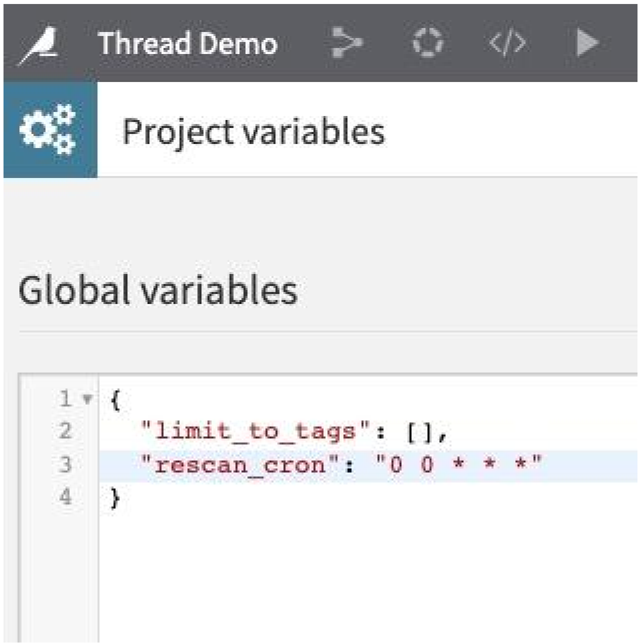


```
{
  "limit_to_tags": ["demo", "documented"]
}
```

Automating Thread Full Rescan

Thread can be configured to automatically rescan on a scheduled basis by simply adding a new project variable “rescan_cron” to the installed Thread project. This variable defines the cron schedule that a full automated scan will execute on. This scan WILL NOT delete any definitions.

Cron reference: https://crontab.guru/



```
{
  "limit_to_tags": [],
  "rescan_cron": "0 0 * * *"
}
```