



DataPLANT's QuickStart on ARCs

V1.1, 2022

We are very happy that you chose our tools and infrastructure to create and share your own ARCs. In this QuickStart we focus on how to use the ARC Commander and SWATE to store your data and enrich it with metadata.

This document is work in progress: If you experience any inconsistencies, have questions or would like to suggest additions to it, please feel free to send a message to: info@nfdi4plants.org.

- [Setup and environment](#)
 - [Required softwares](#)
 - [The command line](#)
- [ARC initialization](#)
- [Adding metadata](#)
 - [ISA investigation](#)
 - [ISA studies and assays](#)
- [Sharing your ARC](#)
 - [DataPLANT registration and access](#)
 - [ARC synchronization](#)
 - [Invite collaborators](#)
- [Data annotation](#)
 - [SWATE](#)
- [DataPLANT Support](#)

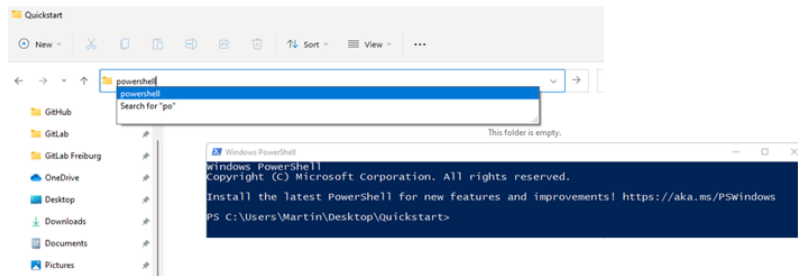
Setup and environment

Required softwares

- ☐ Please download the latest version of the [ARC Commander](#) for your operating system and install it according to [these instructions](#).
- ☐ Prerequisites for using the ARC Commander are [git](#) and [git LFS](#)

The command line

- Most of this quickstart (especially the section [ARC initialization](#)) is based on the command line (Windows: cmd, powershell; Linux and Mac: terminal). The following picture shows exemplarily how to open a powershell on windows by entering *powershell* into the explorer path:



- Text formatted as code blocks represents commands to copy/paste into the command line:

```
echo "hello - I am a code block"
```

ARC initialization

1. Create and navigate to a local folder, which you want to initialize as an ARC.

```
mkdir ~/QuickStartARC;
cd ~/QuickStartARC
```

2. Test if the ARC Commander is functional by displaying the ARC commander version and help menu:

```
arc --version
arc --help
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Martin\Desktop\Quickstart> arc
no valid subcommand has been specified.

SUBCOMMANDS:
  init <init args>      Initialize basic folder structure
  export <export args>  Exports the full arc to a json object
  sync <sync args>      Synchronize the ARC with its upstream repository. Commits changes made in the ARC. If a
                        remote is set or is given, also pulls from there and pushes all previously made commits.
                        Download an arc from a remote repository (e.g. from gitlab)
  get <get args>        Get the ArcCommander's current version
  update                Update items in the arc against each other
  --version             Get the ArcCommander's current version
  investigation, i <verb and args>  Investigation file functions
  study, s <verb and args>  Study functions
  assay, a <verb and args>  Assay functions
  configuration, config <verb and args>  Configuration editing
  remoteaccess, remote <verb and args>  Subcommands for handling access functionality to remote repositories
  Use 'arc.exe <subcommand> --help' for additional information.

OPTIONS:
  --workingdir, -p <working directory>  Set the base directory of your ARC
  --verbosity, -v <verbosity>          Set the amount of additional printed information: 0 -> No information, 1 (Default) -> Basic
                                        Information, 2 -> Additional information
  --help                                display this list of options.

PS C:\Users\Martin\Desktop\Quickstart>
```

3. Initialize your ARC by executing

```
arc init
```

4. This will create the general ARC folder structure:

<input type="checkbox"/> Name	Date modified	Type
.arc	22.03.2022 08:27	File folder
assays	22.03.2022 08:29	File folder
runs	22.03.2022 08:27	File folder
studies	22.03.2022 08:29	File folder
workflows	22.03.2022 08:36	File folder

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Martin\Desktop\Quickstart> arc init
Start processing command with the arguments:
  EditorPath:Field ""
  GitLFSByteThreshold:Field ""
  Owner:Field ""
  RepositoryAddress:Field ""

Start Arc Init
Initialized empty shared Git repository in C:/Users/Martin/Desktop/Quickstart/.git/
Done processing command.
PS C:\Users\Martin\Desktop\Quickstart>
```

Adding metadata

ISA investigation

The ISA investigation workbook allows you to record administrative metadata of your project. Add the isa.investigation.xlsx workbook including an identifier to your ARC with

```
arc i create -i "QuickStartInvestigation"
```

ISA studies and assays

The ISA study and ISA assay workbooks allow you to annotate your experimental data.

1. Add an isa.study.xlsx workbook including an identifier to your ARC with

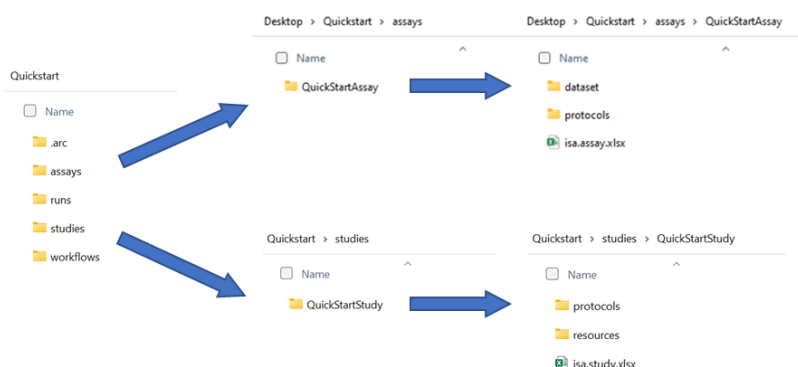
```
arc s create -s "QuickStartStudy"
```

2. Add an isa.assay.xlsx workbook including an identifier to your ARC with

```
arc a create -s "QuickStartStudy" -a "QuickStartAssay"
```

Note: An assay must be linked to a study. If a study does not exist, it will be created automatically in this step.

- The ARC Commander will add a subdirectories to the *studies* and *assays* folder. Your ARC should look similar to this now:



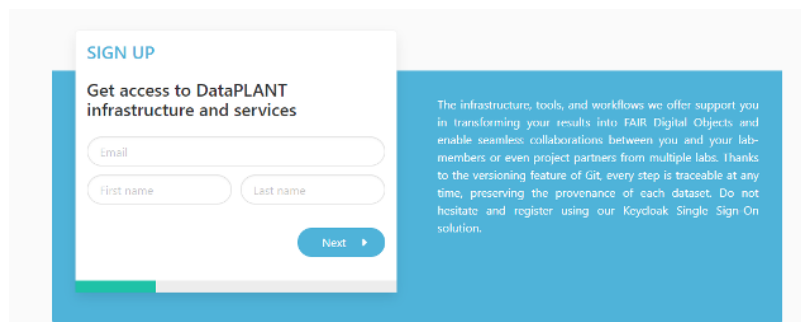
- These steps can be repeated to add as many studies and assays as needed. Accordingly, more subdirectories will be added. Multiple assays can be grouped in a study when the same StudyIdentifier in the text editor window is used.

3. Place the data for each assay in the respective dataset folder.

Sharing your ARC

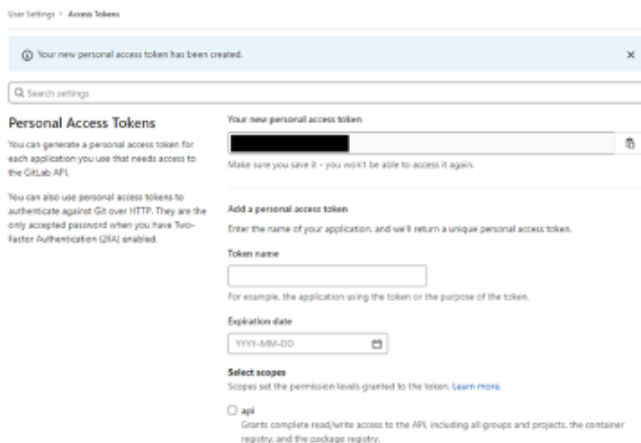
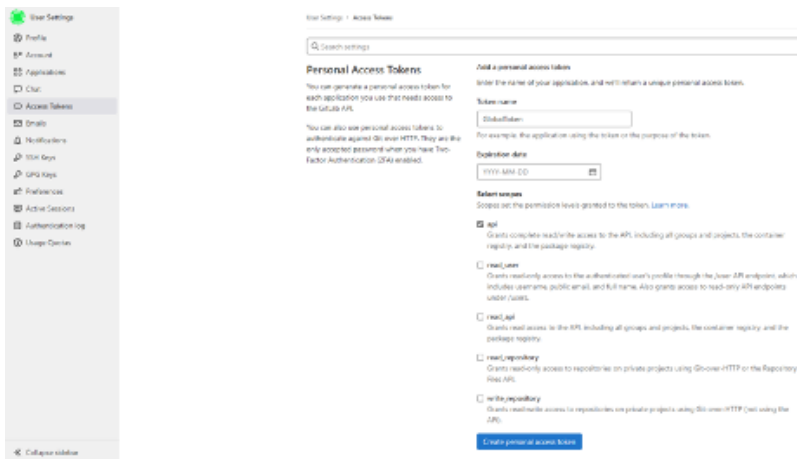
DataPLANT registration and access

In case you are not a member of DataPLANT yet, please visit <https://register.nfdi4plants.org> to register. Afterwards, you will be granted access to DataPLANT's DataHUB, available under <https://git.nfdi4plants.org>. The DataHUB allows you to share your ARCs with registered lab or project partners.

A screenshot of the DataPLANT registration form. The form is titled "SIGN UP" and "Get access to DataPLANT infrastructure and services". It contains input fields for "Email", "First name", and "Last name", followed by a "Next" button. To the right of the form, there is a blue box with white text explaining the benefits of the infrastructure, tools, and workflows, and mentioning the versioning feature of Git and the Keycloak Single Sign On solution.

After successful registration, please visit the [DataHUB](#) to set an access token for ARC Commander synchronization:

1. Sign-in in the top right corner. Click on your profile picture in the top right corner and go to *Preferences -> Access Tokens*.
2. Create an api access token with a name of your choice. These tokens grant read and write access to all of your groups and projects. Make sure you save your access token upon successful creation, as this is the only time you will have access to the token (in case you lose the token, you can simply create a new one).



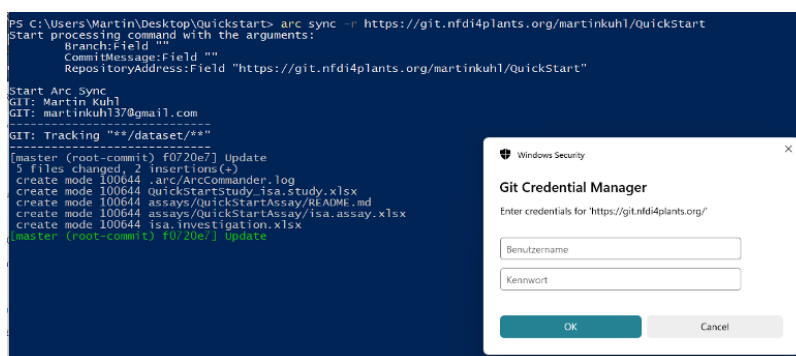
ARC synchronization

1. Open the ARC Commander within your ARC as described above.

Synchronize your ARC with the DataHUB using the command `arc sync`. The ARC Commander will ask for your credentials, where you need to enter your DataHUB handle (displayed on the DataHUB when clicking on your profile picture) and the newly generated access token.

1. If you did not connect your local ARC with a remote so far, you can specify the remote address with the flag `-r` in combination with a URL, e.g.,

`arc sync -r git@git.nfdi4plants.org:brilator/quickstart.git`



2. In case you want to create a new remote repository at this URL, it needs to be assembled like the following example:

<https://git.nfdi4plants.org/YourUserName/NameOfLocalFolder>

If no repository exists under the given URL, the ARC Commander will produce an error ensuring that you spelled the URL correctly*** Use `arc sync -f` to force synchronization to the specified URL.

```

[Warning] (Local-Commit) The local update
[Error] Remote repo was set, but does not exist.
[Check] Whether it was spelled correctly. If not, you can run "arc sync" again using the --repositoryAddress argument.
If you want to create a new remote repository instead, you can run "arc sync -f" to force push the local repository to a new upstream.
Done processing command.
PS C:\Users\Martin\Desktop\Quickstart> arc sync -f
Start processing command with the arguments:
  Branch:Field ""
  CommitMessage:Field ""
  Force:Flag
  RepositoryAddress:Field ""

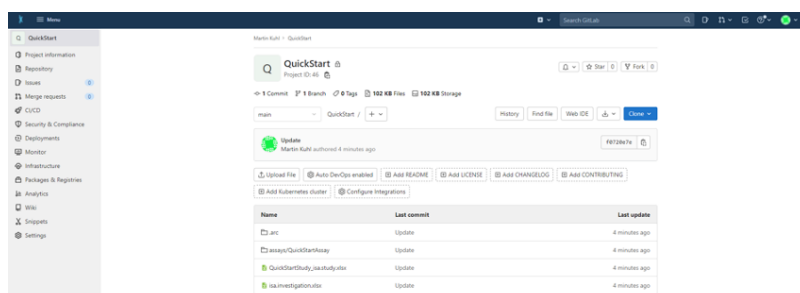
Start Arc Sync
git: Martin Kuhl
git: muelikuhl37@gmail.com
GIT: Tracking "**dataset/**"
-----
On branch main
Untracked files:
  (use "git add <files>..." to include in what will be committed)
  gitattributes

```

Note: In case you did not set your git user name and email address you might get a warning to do so. These are needed for displaying them on the git commits. You can update the settings with the following commands:

```
git config --global user.name <your_name>
git config --global user.email <your_email>
```

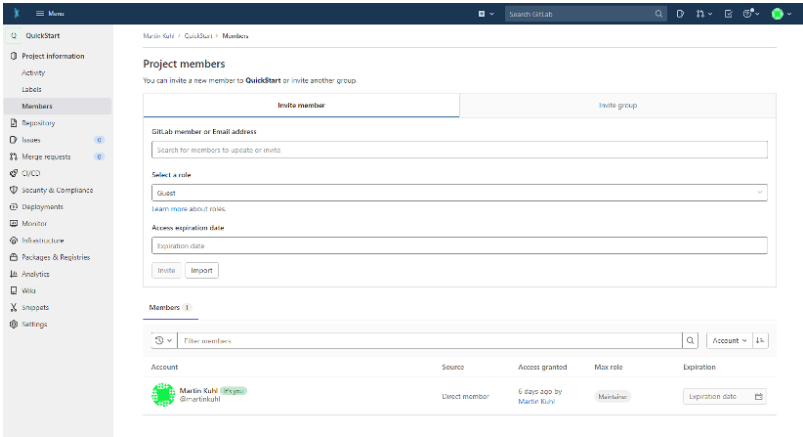
2. Check if the upload was successful by visiting the respective URL.



Invite collaborators

1. You can invite lab-colleagues or project partners to join your ARC for collaborative work. While inside your ARC on the DataHUB, click on *Project information* -> *Members* in the left navigation panel. Search for registered researchers and select a role for each individually. These roles come along with different rights. In short:

- *Guest*: Have the least rights. This is recommended for people you ask for consultancy.
- *Developer*: The choice for most people you want to invite to your ARC. Developers have read and write access, but cannot maintain the project on the DataHUB, e.g. inviting others.
- *Maintainer*: Gives the person the same rights as you have (except of removing you from your own project). This is recommended for inviting PIs or group leaders allowing them to add their group members for data upload or analysis to the project as well.



Note: A detailed usage instruction for the ARC Commander can be found [here](#).

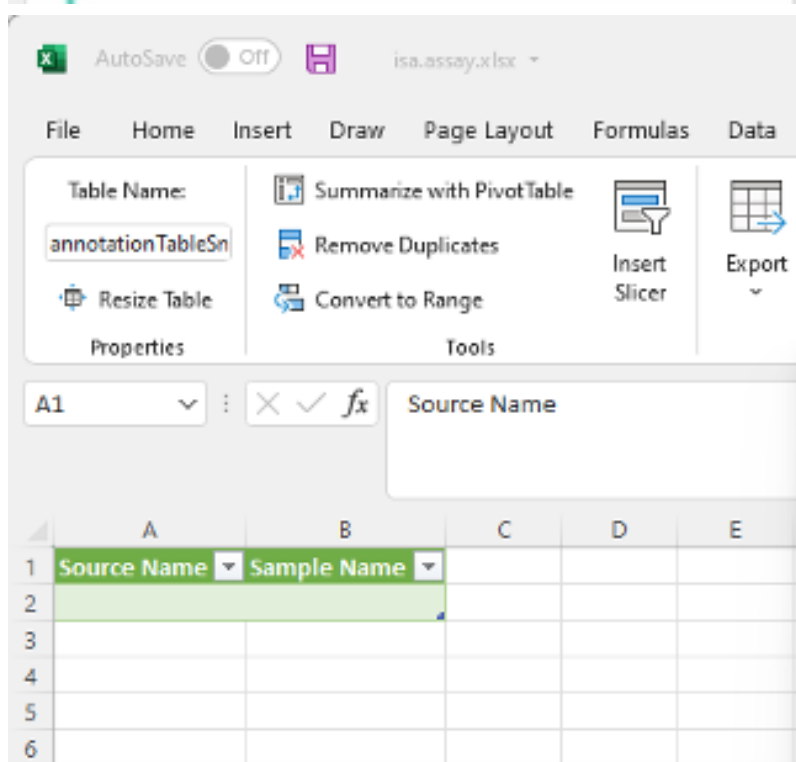
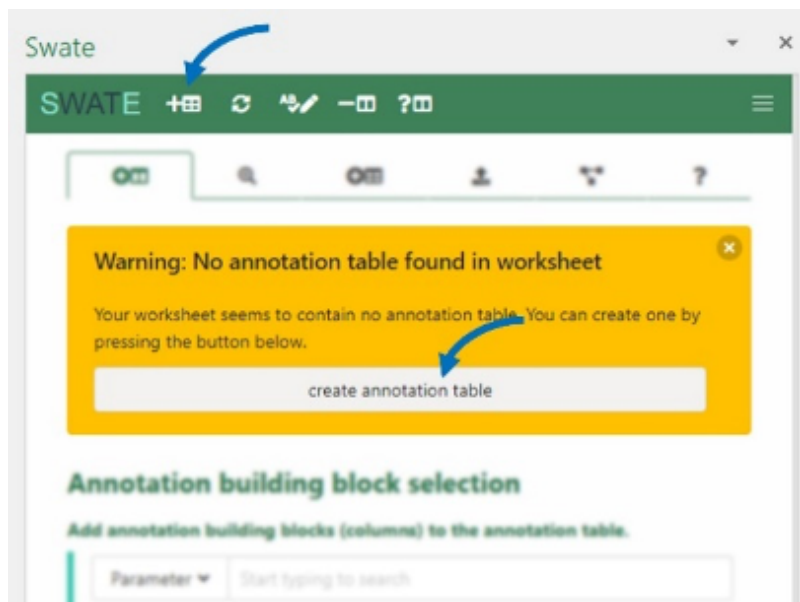
Data annotation

Your ARC should now contain one `isa.investigation.xlsx` and one or more `isa.study.xlsx` and `isa.assay.xlsx` file(s), respectively. Use the `isa.study.xlsx` to describe the characteristics of your samples, e.g. how you grew your plant, and `isa.assay.xlsx` to annotate the experimental analyses.

SWATE

DataPLANT provides the Excel Add-In SWATE to support you in data annotation.

- ☐ Download and install the newest SWATE version according to [these instructions](#).
- ☐ In case you use an Excel version older than Excel 2019, please install [SWATE for Excel online](#).
- Use the *create annotation table* button in the yellow pop-up box (this only appears if you start SWATE on an Excel worksheet without an existing annotation table). An annotation table with the building blocks *Source Name* and *Sample Name* will be generated.



- Annotate your table with help of the [annotation principles](#). In short, *Characteristics* are used for study descriptions and describe inherent properties of the source material (e.g. a certain strain). *Parameters* describe steps in your experimental workflow (e.g. an instrument model or a growth chamber), and *Factors* describe independent variables that result in a specific output (e.g. the light intensity).
- The combination of ISA (Characteristics, Parameter, Factor) and a biological or technological ontology (e.g. temperature, strain, instrument model) gives the flexibility to display an ontology term, e.g. temperature, as a regular process parameter or as the factor your study is based on (Parameter [temperature] or Factor [temperature]).

Customize your table by adding building blocks:

1. Choose the type of building block you want to add (A).
2. If you chose a descriptive building block type (building blocks besides Sample Name, Source Name, and Data File Name), use search field (B) to search for an Ontology Term. SWATE accesses the SwateDB with a list of established external ontologies designated suitable for use in plant science. In addition, we feature our own ontology NFDI4PSO to extend the DB with missing, but necessary terms.
3. If you want to add a building block with a unit, check box (C) and use search field (D) to look for a fitting unit term, e.g. degree Celsius as unit for Parameter [temperature].
4. If you could not find a fitting term, you can use the Advanced Term Search with the blue links above the *Add building block* button. If you still could not find a fitting term, use free text input.

Annotation building block selection

Add annotation building blocks (columns) to the annotation table.

(A) Parameter (B) Start typing to search

(C) This Parameter has a unit: (D) Start typing to search

Use advanced search building block Use advanced search unit

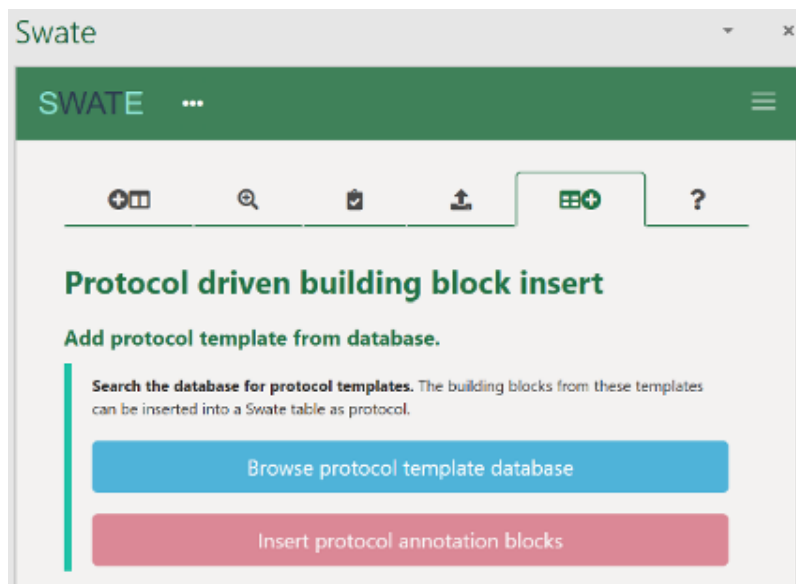
Add building block

Add/Update unit reference to existing building block.

5. For more information on customizing your annotation table click [here](#).

Use templates

1. Alternatively, you can also use one of DataPLANT's [SWATE templates](#). You can find them under the *Protocol Insert* tab in SWATE.



Annotate your samples and data

Fill the cells beneath each building block with ontology terms to note the respective *Characteristics*, *Parameter*, and *Factor* values of your experiment. Using the ontology term search function, you can fill multiple cells at once.

1. When *Use related term directed search* (A) is enabled, SWATE will suggest a selection of suitable terms within the ontology for the column header, e.g. *TripleTOF 5600* for *instrument model*.
2. When term directed search (A) is disabled, SWATE will still suggest ontology terms, but without relation to the column header.
3. If you could not find a fitting term, use free text input.

The screenshot shows the SWATE application's ontology term search interface. At the top is a green header bar with the SWATE logo and several icons. Below the header is a navigation bar with icons for home, search, settings, and help. The main section is titled "Ontology term search" and includes the instruction "Search for an ontology term to fill into the selected field(s)". There is a large text input field with the placeholder "Start typing to search for terms". Below the input field are two options: "Use related term directed search." (which is selected with a green checkbox) and "Use advanced search" (a blue link). At the bottom of the search section is a red button labeled "Fill selected cells with this term".

This screenshot shows the SWATE application with the ontology term search interface open on the right side. The left side displays a spreadsheet with columns for "Parameter (instrument model)" and "Sample Name". The search interface on the right has the "instrument model" field selected, and the search term "trip" is entered. The results show a list of instrument models with their corresponding MS numbers:

Instrument Model	MS Number
TripleTOF 5600	MS1000332
TripleTOF 6600	MS1002533
TripleTOF 4600	MS1002583
TripleTOF 5600+	MS1002584
Triple Quad 3500	MS1002591

Below the results, there is a link "Can't find the Term you are looking for? Use Advanced Search" and a note "You can also request a term by opening an Issue".

Note: More information on how to use SWATE can be found [here](#).

DataPLANT Support

For further assistance, feel free to reach out via our [helpdesk](#) or by contacting us [directly](#).