**Integrator**

In order to allow the download of Test Bench documents and Station Acceptance documents, we need to define a new T2 request in **rrps.dt.events.t2defs** to retrieve documents using the SDOK API

([https://mtuds.visualstudio.com/4b4684cc-e6da-47b7-9b46-c0f79a99c274/\_apis/git/repositories/8a9f04fc-8b85-4537-995e-855ec5d337bb/Items?path=%2F.attachments%2FDTWIN\_SDOK\_ODATA%20Documentation1-00dfb2e8-3cc3-443a-a490-126aea27d1f9.docx&download=false&resolveLfs=true&%24format=octetStream&api-version=5.0-preview.1](https://mtuds.visualstudio.com/4b4684cc-e6da-47b7-9b46-c0f79a99c274/_apis/git/repositories/8a9f04fc-8b85-4537-995e-855ec5d337bb/Items?path=/.attachments/DTWIN_SDOK_ODATA%20Documentation1-00dfb2e8-3cc3-443a-a490-126aea27d1f9.docx&download=false&resolveLfs=true&$format=octetStream&api-version=5.0-preview.1))

The T2 request might be defined as follows:

T2\_REQUEST\_SAP\_SDOCK = T2\_PROVIDER\_SAP\_DOCUMENTSINGLE + 'SDOK'

""" Request for SAP SDOK document

UTF8 encoded JSON.

{

    "equnr" : "Equipment number",

    "document\_type": "PPM or MOA"

}

"""

This request can be handled by the existing **rrps.dt.integrator.sapequimenthistory** integrator

in a new method which would work similarly to **\_\_t2\_do\_document\_req** ;

this new method would retrieve the equipment number and doc\_type from the T2 request parameters

and pass them to the SDOK endpoint, then stream the document in the SAP response.

---

**Follower**

In order to serve PPM and MOA documents,

**the follower needs to expose a new API to download test documents at the**

**endpoint /test-document/<asset id>/<document type>,**

**where document type can be either PPM or MOA** ;

**the equipment number for an asset identifier can be retrieved from the basic\_data of the asset.**

***What will be the “equnr”??? 🡺 Engine ???***

The implementation of the method for this endpoint will be very similar to the **get\_document\_by\_id**method in **default\_controller.py** .