Using the FIB-SEM-Tomography Schema and Map for Zeiss Auriga

The acquisition data looks like this:

A white paper with green check mark and white text

Description automatically generated

The files with extension .ve-a3f and .bak contain information pertaining to the whole project in XML format. The .bak file can be selected as this is smaller and without the detailed log. The image datasets are contained in the folders called InLens and SESI. There could be more image folders, depending on the detectors used. The folders titled AutoTune and keyframes and the execution\_log.log are to be ignored.

The XML in the .bak file can be either converted to json and mapped with the schema for SEM-FIB Tomography or read directly as XML and mapped. In this case, the attributes of the XML elements are referred to in the map with an @ symbol. For example, the element ATLAS3D-Job has an attribute called “version”. It is referred to in the map as [ATLAS3D-Run-bak.ATLAS3D-Job.@version](mailto:ATLAS3D-Run-bak.ATLAS3D-Job.@version). Here, “ATLAS3D-Run-bak”, refers to the file name.

## Dataset Parameters

The TIFF images in the folders InLens and SESI have metadata (contained as XML at tag 51023)

The dataset parameters can be read from any of the images in the corresponding dataset. For example to record the common parameters of the images in the folder InLens, the path would be referred to the map as ImageFolder.Image.<parameter-path>. Here ImageFolder refers to the folder InLens and Image refers to any one of the slice images inside it.

## Splitting program name and program version

"acquisition.dataset.program.programName":"ImageFolder.Image.Fibics.Application.Version"

The value contained in **Fibics.Application.Version** has also the version number- eg : “AtlasEngine v5.0”. Therefore, the **acquisition.dataset.program.programName** has to be split again into Name and Version. Then **acquisition.dataset.program.programVersion** will get the value after the blank space which is “v5.0” and **acquisition.dataset.program.**programName will get the value before the blank space which is “AtlasEngine”.

## Split aperture size and unit

"acquisition.dataset.instrument.eBeam.apertureSetting.size":"ATLAS3D-Run-bak.ATLAS3D-Job.ATLAS3D-Setup.SEM\_System\_State.ApName"

**"ATLAS3D-Run-bak.ATLAS3D-Job.ATLAS3D-Setup.SEM\_System\_State.ApName"** is a string which contains the aperture size. We read the entire string into **"acquisition.dataset.instrument.eBeam.apertureSetting.size”.**

For eg., The string would be “[5] 120 µm [HC] (5.0 kV)”. From this text, we need to take the aperture size of “120 µm” and split it into **"acquisition.dataset.instrument.eBeam.apertureSetting.size.value“** (120) and **"acquisition.dataset.instrument.eBeam.apertureSetting.size.unit”** (µm).

## Split Tilt Correction Angle into value and unit

"acquisition.dataset.instrument.eBeam.tiltCorrectionAngle":"Images.SEM Image.SliceImage.ScanInfo.item"

**"Images.SEM Image.SliceImage.ScanInfo.item"** contains a string like “-36.0°”. This string must be split into unit and value. Then **"acquisition.dataset.instrument.eBeam.tiltCorrectionAngle.value“** is to be assigned “-36.0” and **"acquisition.dataset.instrument.eBeam.tiltCorrectionAngle.unit“** is to be assigned “degree”.

## Assign units manually for the following:

**"acquisition.dataset.instrument.scan.dwellTime.unit":“µs”**

**"acquisition.dataset.instrument.scan.pixelWidth.unit":“µm”**

**"acquisition.dataset.instrument.scan.pixelHeight.unit":“µm”**

**"acquisition.genericMetadata.zCutSpacing.unit": “µm”**

**"acquisition.dataset.instrument.iBeam.millingCurrent.unit":”pA”**