



/ MeasurementInstrument / Properties / Manufacturer

Information are located in the DCC under the DOM branch „dcc:item“->“dcc:manufacturer“

/ MeasurementInstrument / Properties / InstrumentDescription

This information can be taken from the DCC DOM branch „dcc:item“

/ MeasurementInstrument / Properties / InstrumentProperties

This information is provided by the manufacturer and may or may not be obtained from the DCC XML file.

/ MeasurementInstrument / Properties / InstrumentProperties / OutputQuantityType

What is the output type of the sensor (e.g. voltage)

/ MeasurementInstrument / Properties / InstrumentProperties / OutputQuantityDimension

Univariate or N-dimensional multivariate. Default is 1.

/ MeasurementInstrument / Properties / InstrumentProperties / InstrumentRange

In what ranges the measurand is measured according to the manufacturer (e.g. force in 50 - 500 N)
Is of type dict with elements like „upper_limit“, „lower_limit“

/ MeasurementInstrument / Properties / InstrumentProperties / InstrumentConditions

Could be a dict containing any kind of information about the conditions during the measurement according to the manufacturer (e.g. temperature and frequency ranges)

/ MeasurementInstrument / Properties / InstrumentProperties / MeasurandType

Type of the measurement instrument input (e.g. acceleration, force, pressure, ...)

/ MeasurementInstrument / Properties / CalibrationInformation

This information can be obtained from the DCC DOM branch „dcc:calibrationlaboratory“

/ MeasurementInstrument / Properties / CalibrationInformation / CalibrationID

For example a unique identifier to the calibration certificate.

/ MeasurementInstrument / Properties / CalibrationInformation / CalibrationRangeCovered

Measurement range covered by the calibration. Ideally this is identical to MeasurementRange.

/ MeasurementInstrument / Properties / CalibrationInformation / CalibrationConditionsCovered

Measurement conditions covered by the calibration. Ideally this is identical to MeasurementConditions.

/ MeasurementInstrument / Properties / CalibrationInformation / CalibrationResult

Is a dict element. Specific entries of this dictionary depend on the type of MeasurementInstrument (e.g. LTI_System may use „frequency_response“)

/ MeasurementInstrument / Methods / Constructor (init)

Possible Python code:

Function **init**(manufacturer=[], InstrumentDescription=[],...):
if isinstance(manufacturer, dict):

Alternatively, the method DCC_Reader could be called.

/ MeasurementInstrument / Methods / PrintInstrumentInformation

Gives a structured output to the command line of the information contained in the dictionaries „Manufacturer“, „InstrumentDescription“, „InstrumentProperties“ and „CalibrationInformation“. Optionally, further dictionaries could be provided as a list in the function call.

/ MeasurementInstrument / Methods / DCC_Reader

Expects the location of valid DCC XML file. It sets the basic class properties by reading out the XML file DOM tree.

CalibrationResult is imported as a dictionary build from the individual DCC XML file entries.

/ MeasurementInstrument / Methods / DCC_Writer

Outputs the information contained in the properties to a DCC XML file.

/ MeasurementInstrument / Methods / save_object

Serialize all information to a single Python file

/ MeasurementInstrument / Methods / read_object

read serialized object

/ LTI_System

Is a subclass of MeasurementInstrument for instruments which are used in a dynamic measurement.

/ LTI_System / Properties / LTI_model

In the long term requires a separate class. For a start can be a second order model with parameters given as Python dict.

/ LTI_System / Methods / Constructor (init)

Could be an init function specifically for the intended type of LTI system

/ LTI_System / Methods / frequency_response

For given frequencies (ndarray) return complex frequency response values of the system

/ LTI_System / Methods / apply_as_filter

input time series (as ndarray) output filtered time series with filter coefficients from system response