HZB Helmholtz Zentrum Berlin

This project receives funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 654000.

SECoP and Metadata for Sample Environment

Klaus Kiefer (HZB) **Workshop on Research Data Management, Berlin, 11 June 2019**

Content

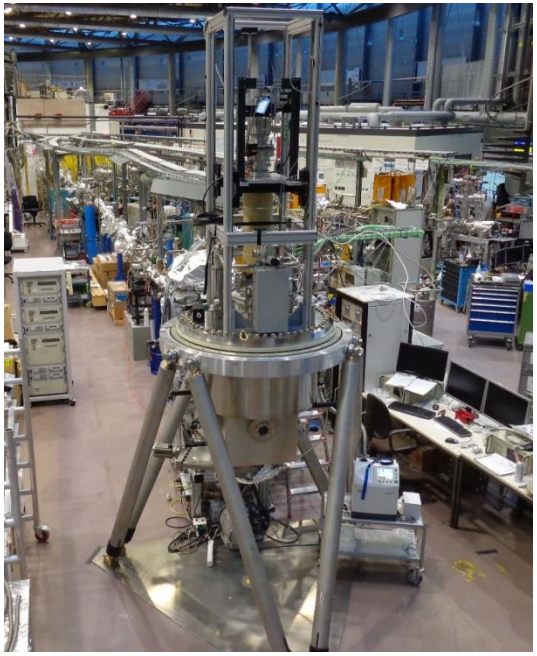
- Sample Environment
- Definition of the problem
- SECoP
- Metadata
- Implementations
- What still needs to be done…

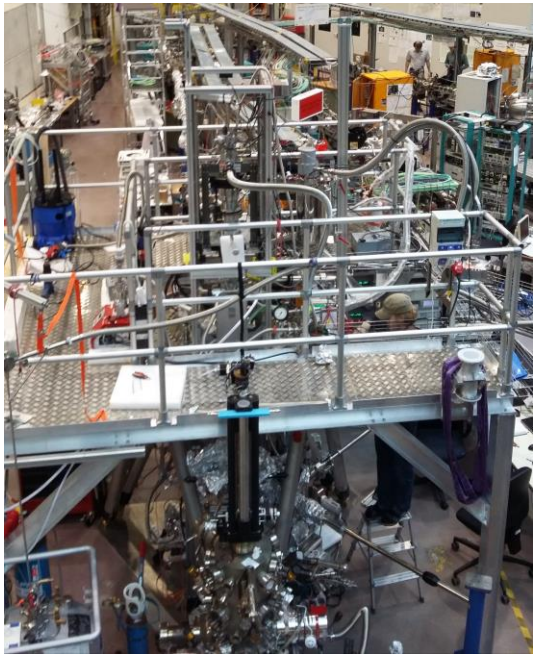
Sample Environment

HZB Helmholtz Zentrum Berlin

VEKMAG: 9T 3D-Vector Magnet und ^{4}He Cryostat @ PM2

F. Radu, H. Ryll, W. Kuch, H. Zabel, C. Back
B. Klemke, K. Kiefer





Physical properties:

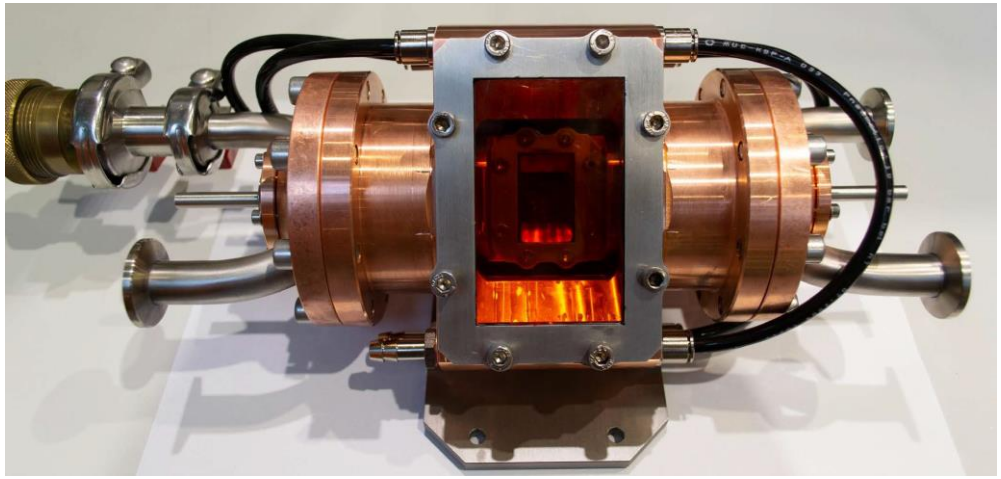
CRYOGENIC
CRYOGENIC LIMITED

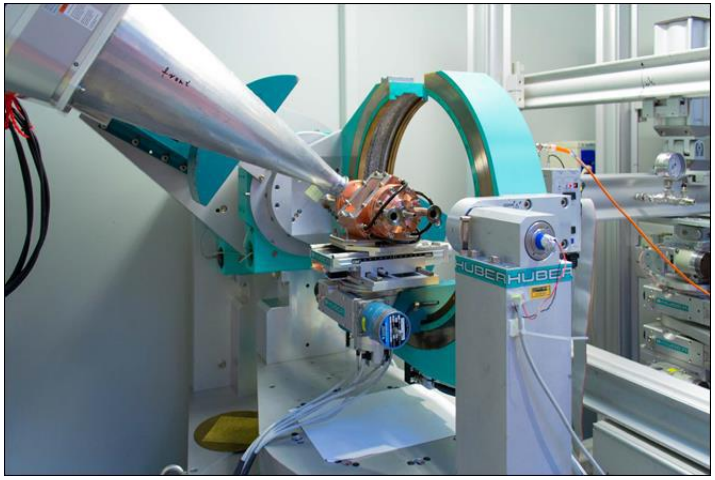
- **Temperature (2 K .. 500 K)**
- **Magnetic field (9T, 2T, 1T)**
- **Helium level**

HZB Helmholtz Zentrum Berlin

HOT VAPOR ADSORPTION CHAMBER FOR X-RAY DIFFRACTION @ KMC-2

R. Blažević, N. Grimm, D. Wallacher, D. Többens, S. Kaskel, I. Senkovska and V. Bon





Physical properties:

- **Temperature (293 K to 423 K)**
- **Pressure (up to 5 bar)**

Definition of the Problem

Outline of the Problem

HZB Helmholtz Zentrum Berlin

Problems:

- Different Experiment Control Softwares (Caress, NICOS, spec, LISE/M…) with different ways to control SE
- Complex SE software interface protocols (CORBA, TACO/TANGO, EPICS…)
- Time consuming integration of new or external SE equipment
- No easy mobility of sample environment equipment
- No standard for SE metadata

- Physicists are programming drivers

Communication to Sample Environment

Experiment Control Software

SECoP

Control Schemes

Sample Environment Control Node

Temperature

Mag. Field

Complex Measurement

SECoP Goals

HZB Helmholtz Zentrum Berlin

- **Common** Standard
- **Inclusive** (fulfilling the needs of all facilities, no hard restrictions for hardware/software)
- **Simple** (easy to use and to implement)
- **Self explaining** (possibility for plug and play operation with compatible clients)
- Integration of **Metadata** (self explaining)

simple, inclusive, self explaining, provides metadata

HZB Helmholtz Zentrum Berlin

ISSE

www.sampleenvironment.org

The International Society for Sample Environment

Committee for the Standardization of Sample Environment Communication

(Head: M. Zolliker (PSI), Members: K. Kiefer (HZB), E. Faulhaber (MLZ), A. Pettersson (ESS), Alan Ye (NIST), Facility contacs: E. Lelíèvre-Berna (ILL), K. Baker (STFC), Andrew Manning (ANSTO), Gary Lynn (SNS), Koji Kaneko (JRR-3), Seiko Kawamura (J-Parc))

SINE2020

Task 7.1

This project receives funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 654000.

HZB Helmholtz Zentrum Berlin

SECoP Basics

SECoP - what has to be defined

1) the **philosopy**

2) the general **structure** of the SECoP information

3) the **syntax** for the SECoP messages

4) the set of **rules** for the SECoP logic

5) the **meaning** of predefined SECoP components

HZB Helmholtz Zentrum Berlin

SECoP Philosopy

SECoP Philosopy

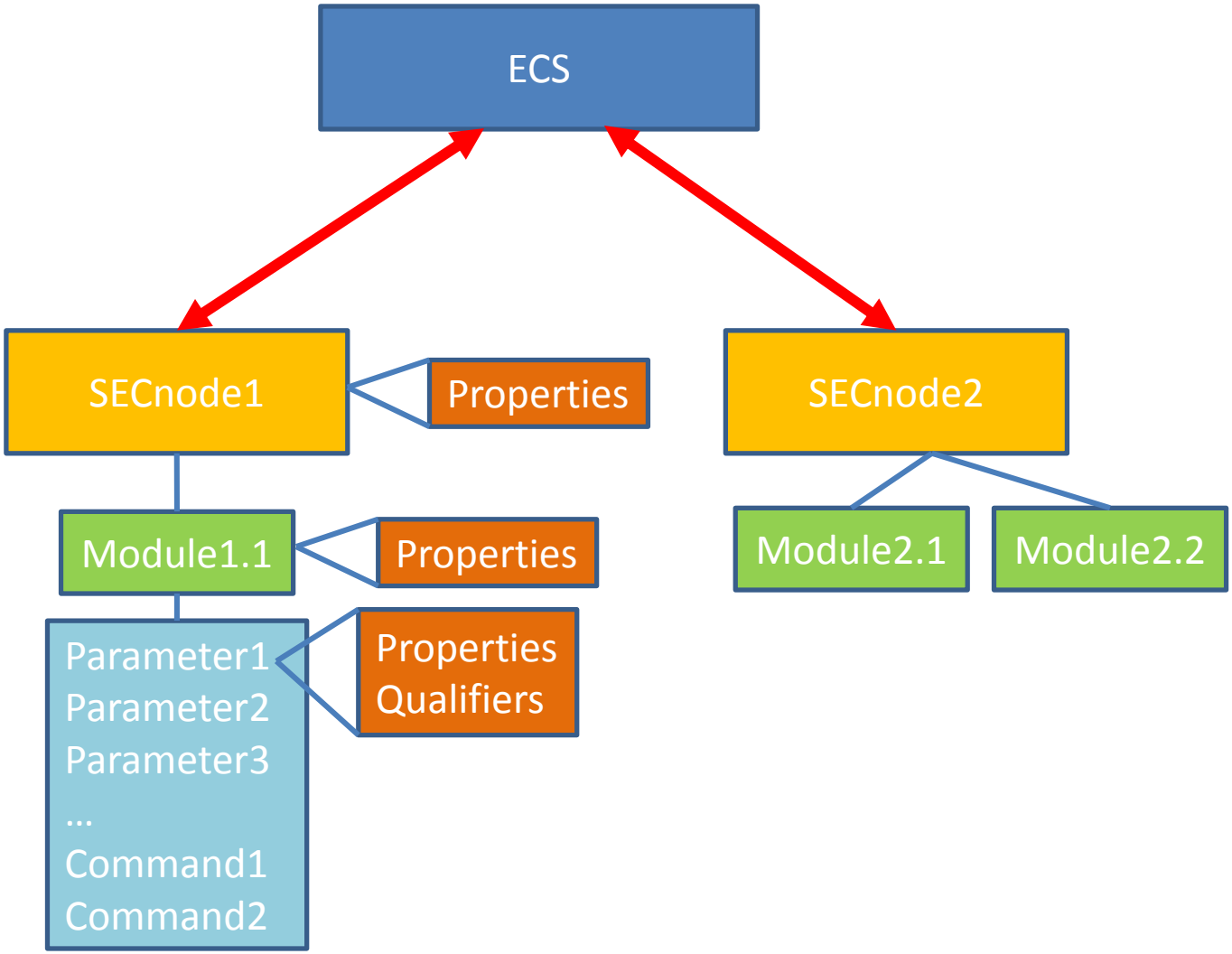
HZB Helmholtz Zentrum Berlin

- **Self-explaining**: the description of a SEC node must contain all necessary information for a) operating the SEC node by the ECS without further documentation in at least a basic mode, b) providing all relevant metadata information
- **Integrative**: enable the use of SECoP by ECSs with a great variety of design concepts (e.g. synchronous vs. asynchronous communication)
- **Simple**: all parts of SECoP (transport layer, syntax) should be as simple as possible (but as complex as needed)
- **Necessary** - **sufficient** - **unambiguous**
- Transport layer: **message oriented** (presently TCP/IP), ASCII
- Protocol is **independent from specific transport layer**
- **Wrap complex functionality** of sample environment equipment on the SEC node side / simplify (standardize) the use of SE equipment by the ECS
- Keep the **overhead** for the SECoP protocol **on SEC node** (server) side **small**
- **Avoid unnecessary traffic**
- Better be **explicit**
- All protocol messages must be **human readable** (with only exception: blob)
- Use **JSON**
- **Impose best-practices to the programmer** of the SEC node by making important features mandatory
- **Must ignore policy**
- Allow for **multiple clients;** if you have more than one client, SECoP does not handle any resulting problems (the SECnode might)
- There should be a **general way of doing things**.

HZB Helmholtz Zentrum Berlin

SECoP Structure

SECoP Structure



SECnode **Module:Parameter**
Module:Command

SECoP Structure

SEC-Node

SE-equipment
e.g. cryostat

Module

Contains all information needed for one physical parameter
e.g. temperature, mag. field

Parameter

e.g. value, status, target

Command

e.g. go, hold, stop

SECoP Structure

SEC-Node

SE-equipment
e.g. cryostat

Module

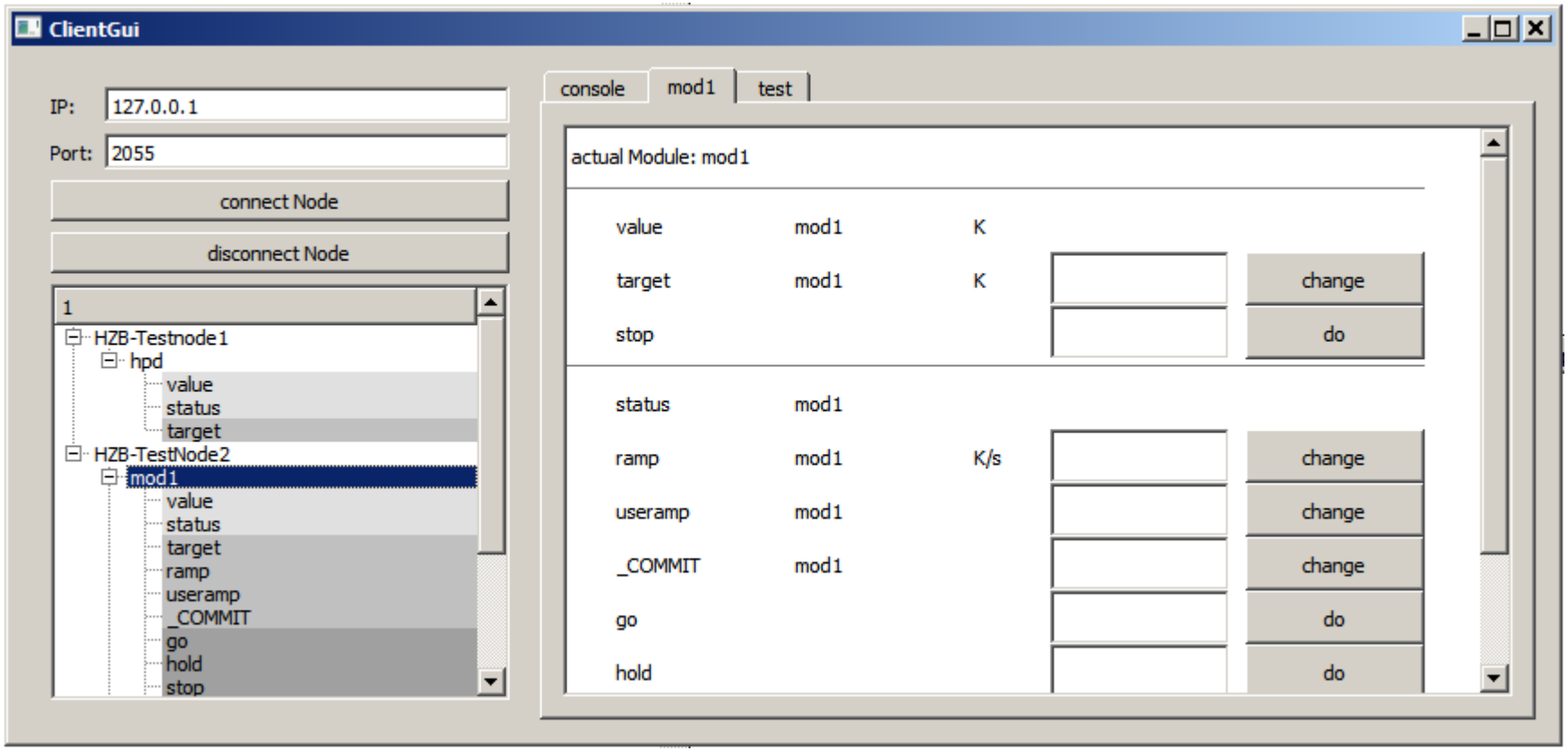
Contains all information needed for one physical parameter
e.g. temperature, mag. field

Parameter

e.g. value, status, target

Command

e.g. go, hold, stop



SECoP rules (handshake)

SECoP rules: status / handshake

HZB Helmholtz Zentrum Berlin

how is the ECS informed about acceptance of requests:

receiving request → do/initiate all things to be done (incl. side effects) → set status to "busy" → send a reply

how is the ECS informed about completion of tasks:

idle/busy → (request) busy → (completion) idle

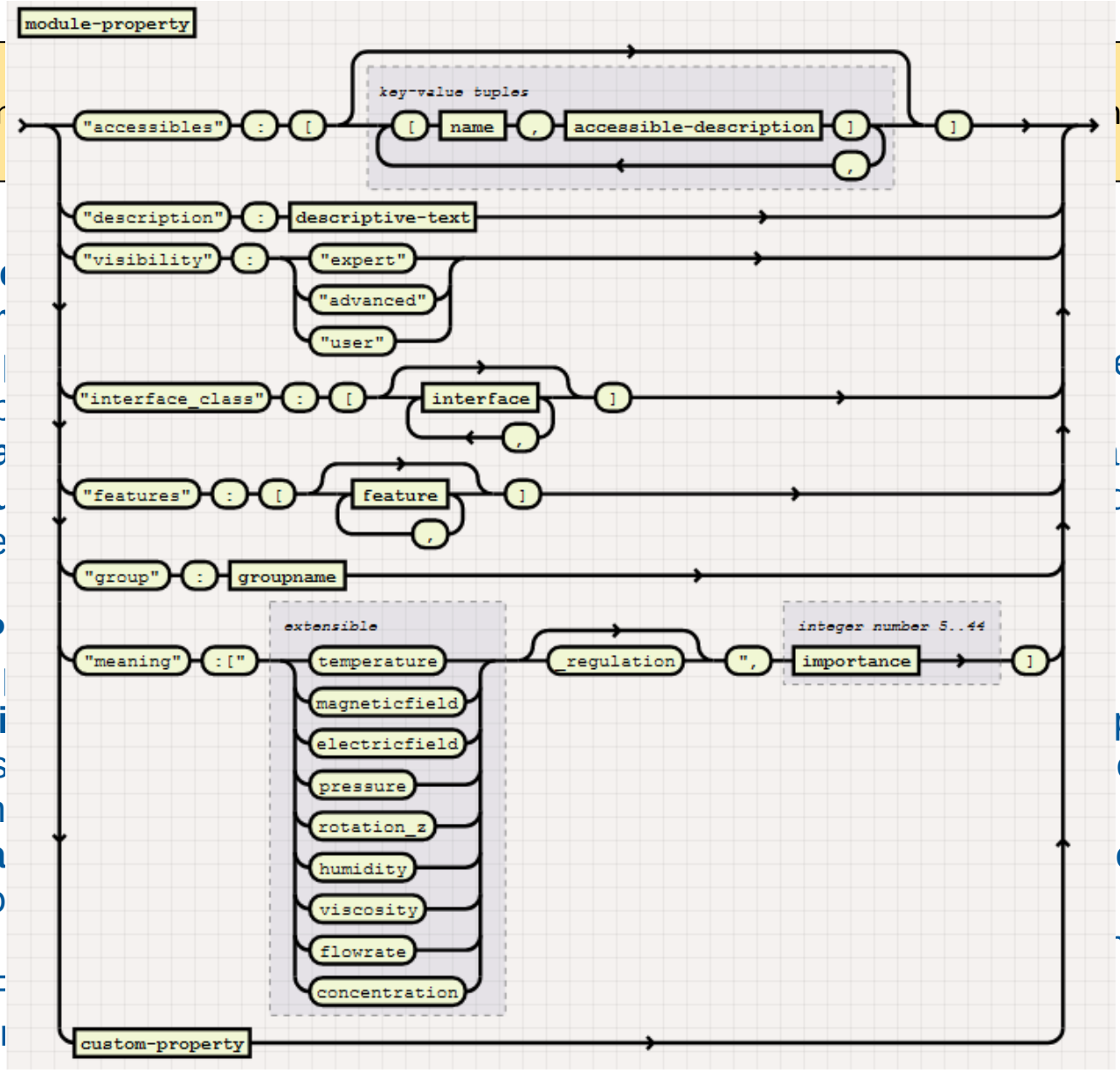
- No message but "change target" and commands may take considerable time. Only these two requests can change the status to busy.
- SEC node may go busy for other reasons (e.g. internal recalibration, other client issuing a command).
- With two or more actions in parallel there is no possibility to differentiate which actions are still running. When all actions are finished: go to "idle"
- SEC node is free to reject any action (but "stop" command). Send an error message if the action is rejected.
- "Stop" command must not be rejected. Stop is ending all running actions.
- SEC node must reject an action (error message send) which can't be performed immediately (no queueing!) (to be discussed)

SECoP syntax

SECoP basics: definitions

HZB Helmholtz Zentrum Berlin

Definition:
The static ir … properties with predefined …



SEC Node
- **equipr…**
- **descri…** … is a short descrip…
- **firmwa…** …re)
- **timeou…** …d within a time well be…

Module P…
- **descri…**
- **visibili…** …ply that the access… …ed modules. (option…
- **interfa…** …e_control", "drivab…
- **group** … …separator) (*see SECoP…*
- **meani…**

SECoP syntax: examples

description	request	**describe**
	reply	**describing** *SEC-node-id description*
change value	request	**change** *module*:*parameter value*
	reply	**changed** *module*:*parameter value*
read request	request	**read** *module*:*parameter*
	reply	**update** *module*:*parameter value*

HZB Helmholtz Zentrum Berlin

www.github.com/SampleEnvironment/SECoP

SECoP 1.0 Beta

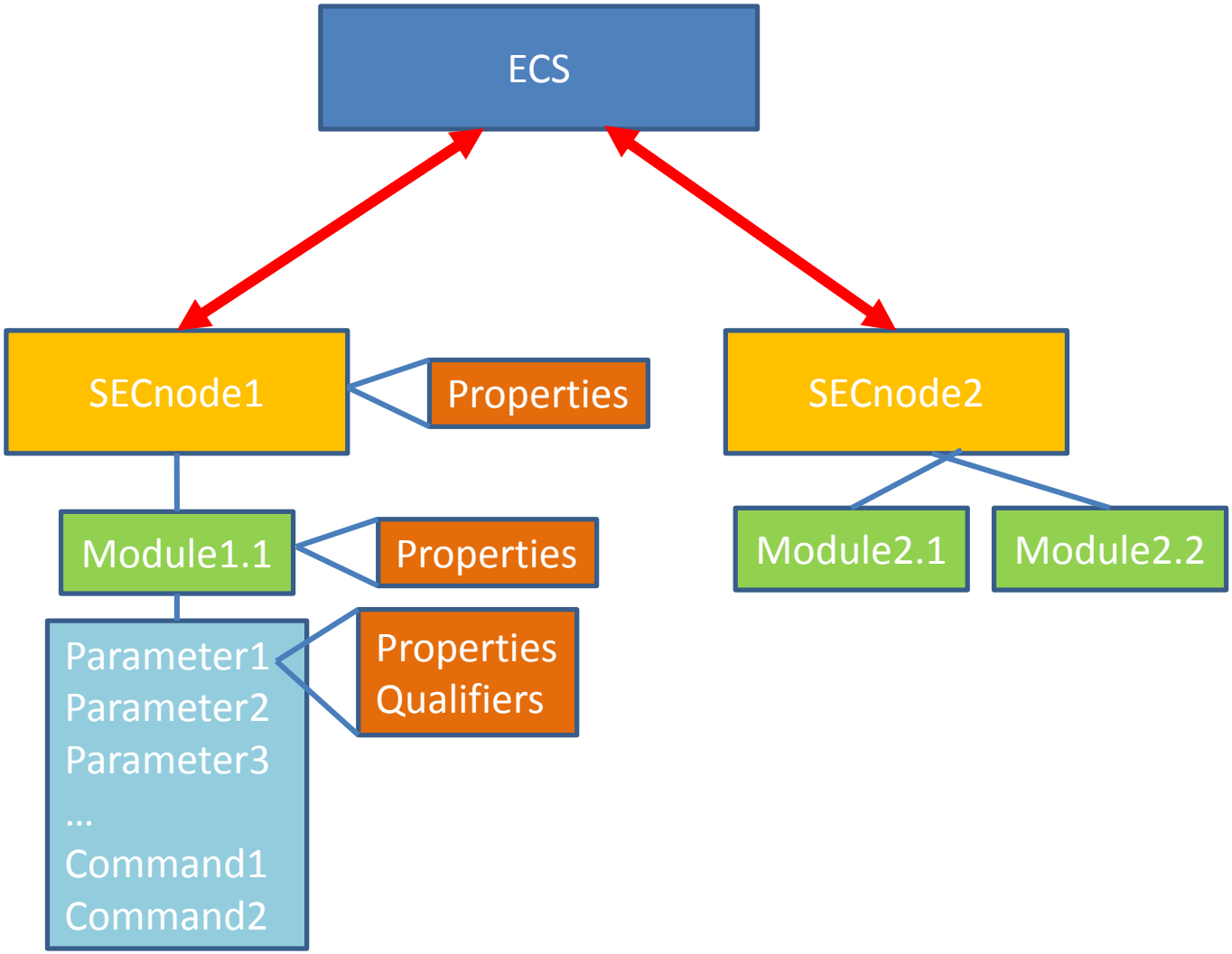
HZB Helmholtz Zentrum Berlin

Metadata in SECoP

Metadata

- **Structure** (module, grouping)
- **Information** (live metadata, static metadata)
- Predefined **meaning** (predefined properties, property „meaning“, interface classes)
- Translation to Nexus

Metadata: Structure



SECnode **Module:Parameter**
Module:Command

Metadata

- **Structure** (module, grouping)
- **Information** (live metadata, static metadata)
- Predefined **meaning** (predefined properties, property „meaning“, interface classes)
- Translation to Nexus

HZB Helmholtz Zentrum Berlin

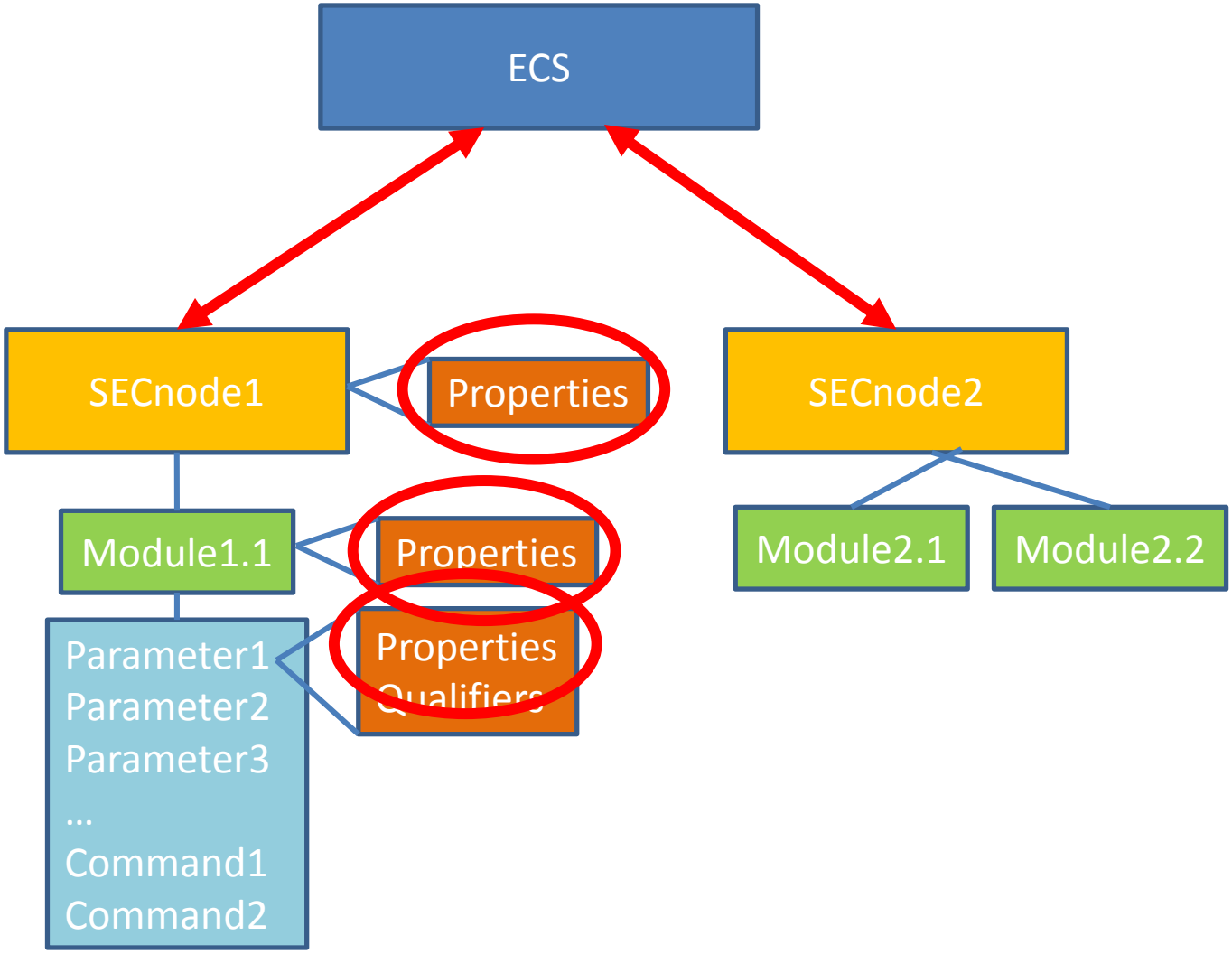
Static Metadata:

must not and cannot change during a running experiment

Basic information: **properties**

- the *equipment_id* and the *description* of a *SEC-node* (both mandatory),
- the *description* (mandatory) and the *meaning* (optional) of a *module,*
- the *description* and the *datatype* of an *accessible* (both mandatory). In general, the *datatype* has a complex structure that can contain among other things the physical unit of a *parameter.*

Static Metadata



HZB Helmholtz Zentrum Berlin

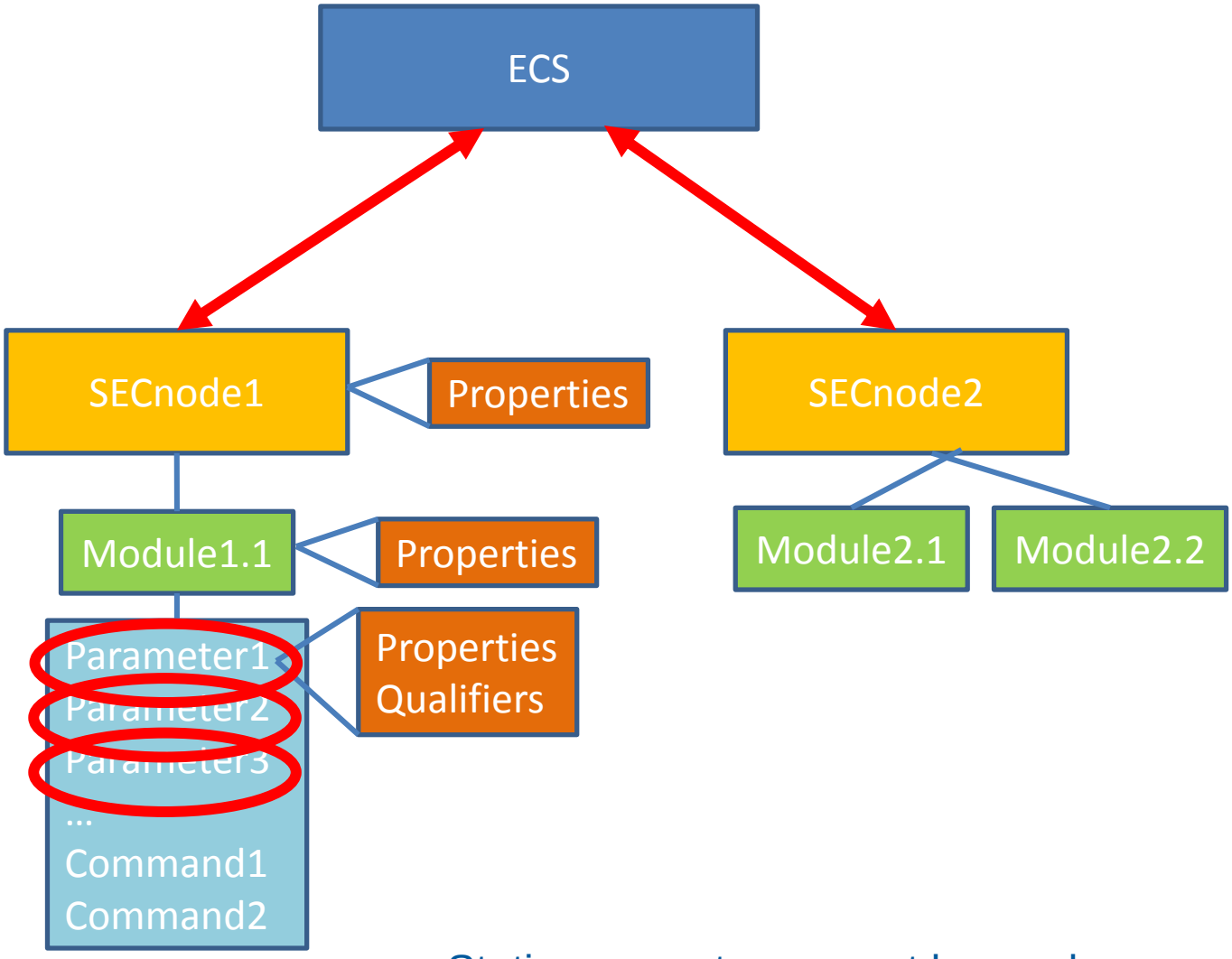
Static Metadata:

must not and cannot change during a running experiment

Numerical and structured metadata: **static parameters**

- Example: Calibration curve of a sensor
- General meaning attributed by interface classes

Static Metadata: static parameters



Static parameter: cannot be read
(only describe message)

Metadata: information

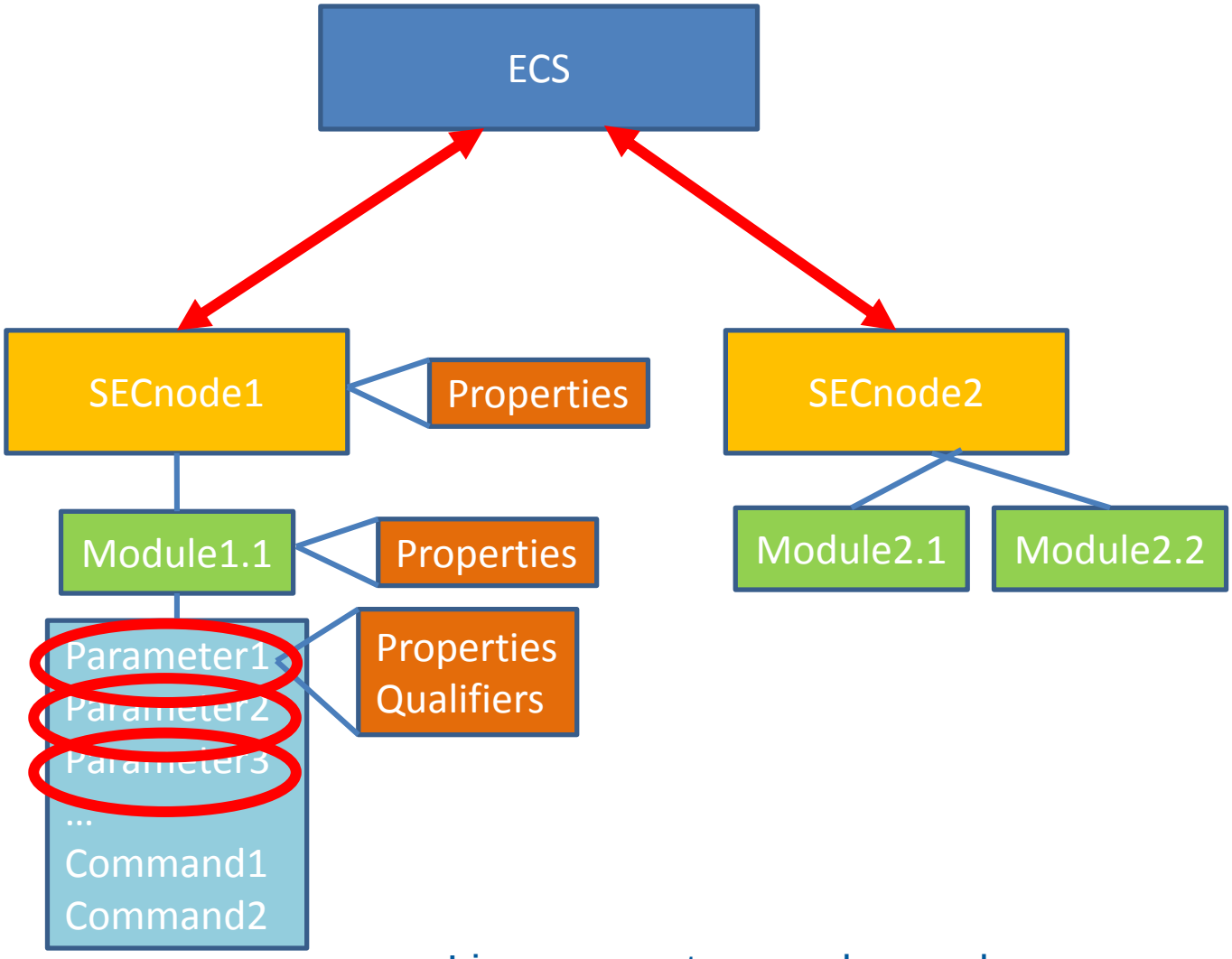
HZB Helmholtz Zentrum Berlin

Live Metadata:

all metadata information that can be altered during the measurement

- (Parameters at sample position (sample temperature, humidity,...))
- Additional information (Helium level, temperature of waterbath (humidity cell): trasported as SECoP parameters

Live Metadata



Live parameter: can be read

Metadata

HZB Helmholtz Zentrum Berlin

- **Structure** (module, grouping)
- **Information** (live metadata, static metadata)
- Predefined **meaning** (predefined properties, property „meaning“, interface classes)
- Translation to Nexus

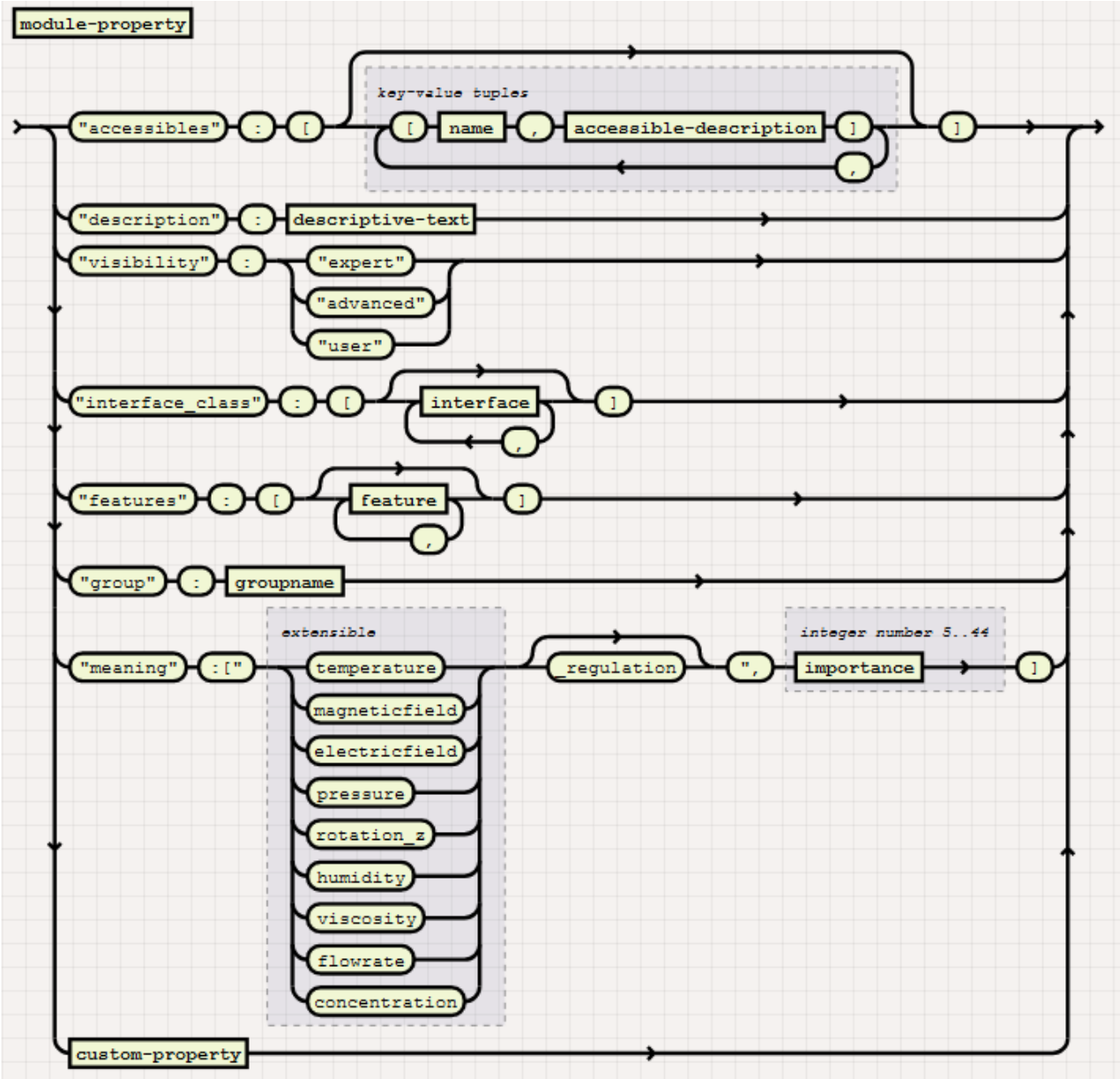
Metadata: meaning

HZB Helmholtz Zentrum Berlin

Definition:
The static ir … properties with predefined …

SEC Node
- **equipr…**
- **descri…** … e is a short descrip…
- **firmwa…** … re)
- **timeou…** … d within a time well be…

Module P…
- **descri…**
- **visibili…** … ply that the access… … ed modules. (option…
- **interfa…** … e_control", "drivab…
- **group** … … separator) (*see SECoP…*
- **meani…**



Metadata

- **Structure** (module, grouping)
- **Information** (live metadata, static metadata)
- Predefined **meaning** (predefined properties, property „meaning“, interface classes)
- Translation to Nexus

4.1 NXenvironment

The NeXus class NXenvironment is intended for "parameters for controlling external conditions" [2]. In the combination with SECoP, NXenvironment can be best represented by a *SEC-node*.

The fields in NXenvironment can be directly attributed to the SECoP information coming from the *SEC-node*. Table 1 shows a possible mapping scheme.

NeXus (NXenvironment)	**SECoP (*SEC-node*)**
name (NXchar)	*equipment_id*
description (NXchar)	*description*
program (NXchar)	*firmware*
(note) (NXnote)	complete answer to "*describe*" message

Table 1: Mapping of NXenvironment

4.2 NXSensor

The NeXus class NXSensor is "used to monitor an external condition. The condition itself is described in NXenvironment." [2]. Compared to NeXus, SECoP offers here an additional structure level with the possibility to combine several *parameters* in a single *module*. In order to keep all information from both *module* and *parameter*, the entries for NXsensor have to be a combination of *module* and *parameter* information. A possible mapping of the most relevant entries is presented in table 2.

NeXus (NXsensor)	**SECoP (*parameter*)**
name (NXchar)	modulename_parametername suggestion: use only "modulename" if *parameter* is *main value*
model (NXchar)	*description* of both the *module* and *parameter*
measurement (NXchar)	if *main value* of a *module*: map to *module property* "*meaning*"
value_log (NXlog)	*value* and timestamp

Table 2: Mapping of NXsensor

Note: The possible keywords for the NeXus measurement are not completely covering all physical sample environment parameters that are typically used in beamline experiments. It would be beneficial to extend this list.

SECoP implementations

SECoP implementations

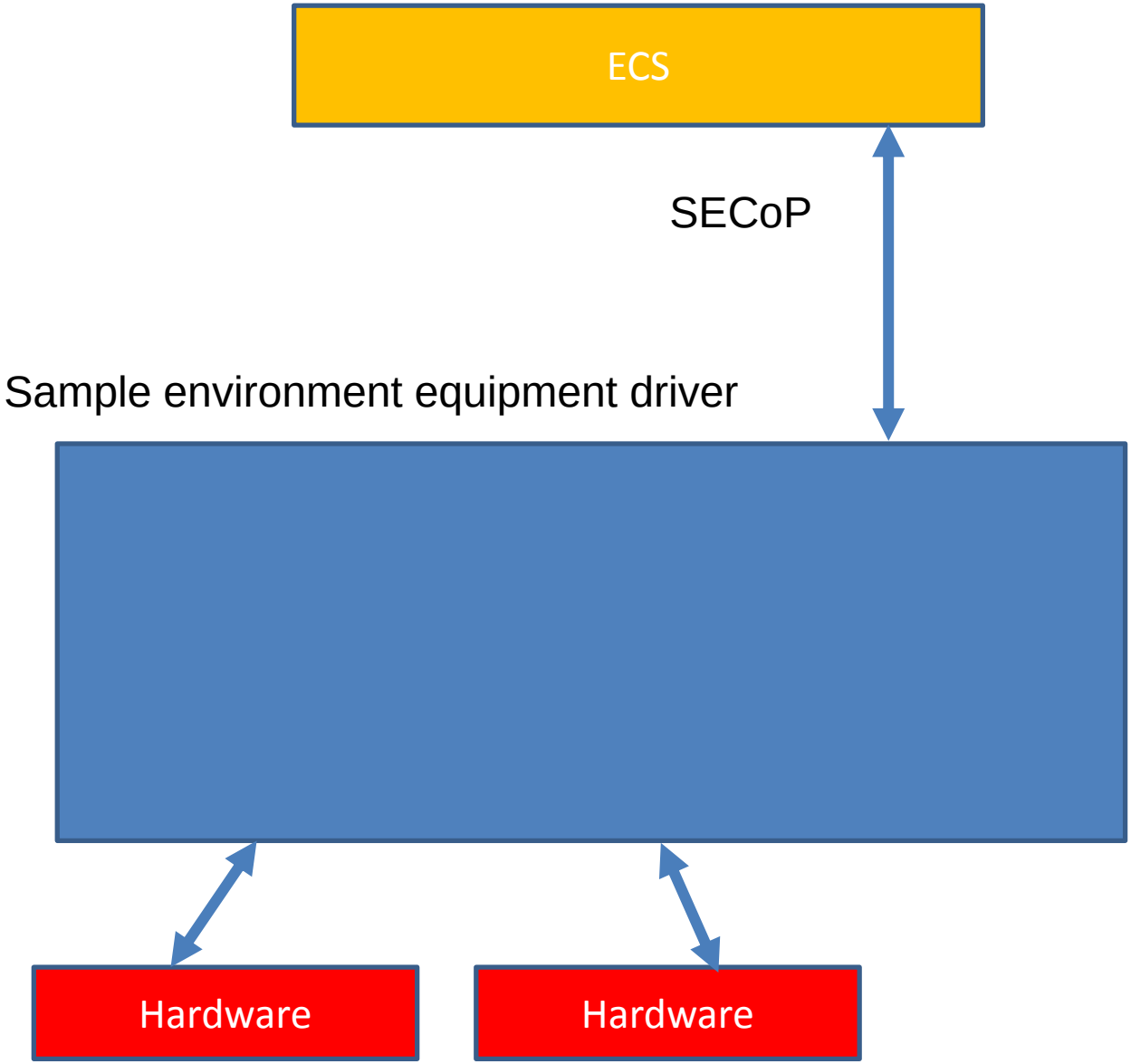
Two solutions, already usable for tests:

1) **SECoP library** (DLL, c-library) from Frank Wutzler and Lutz Rossa (HZB)
   with an API for most compiled languages and an adapter to LabView

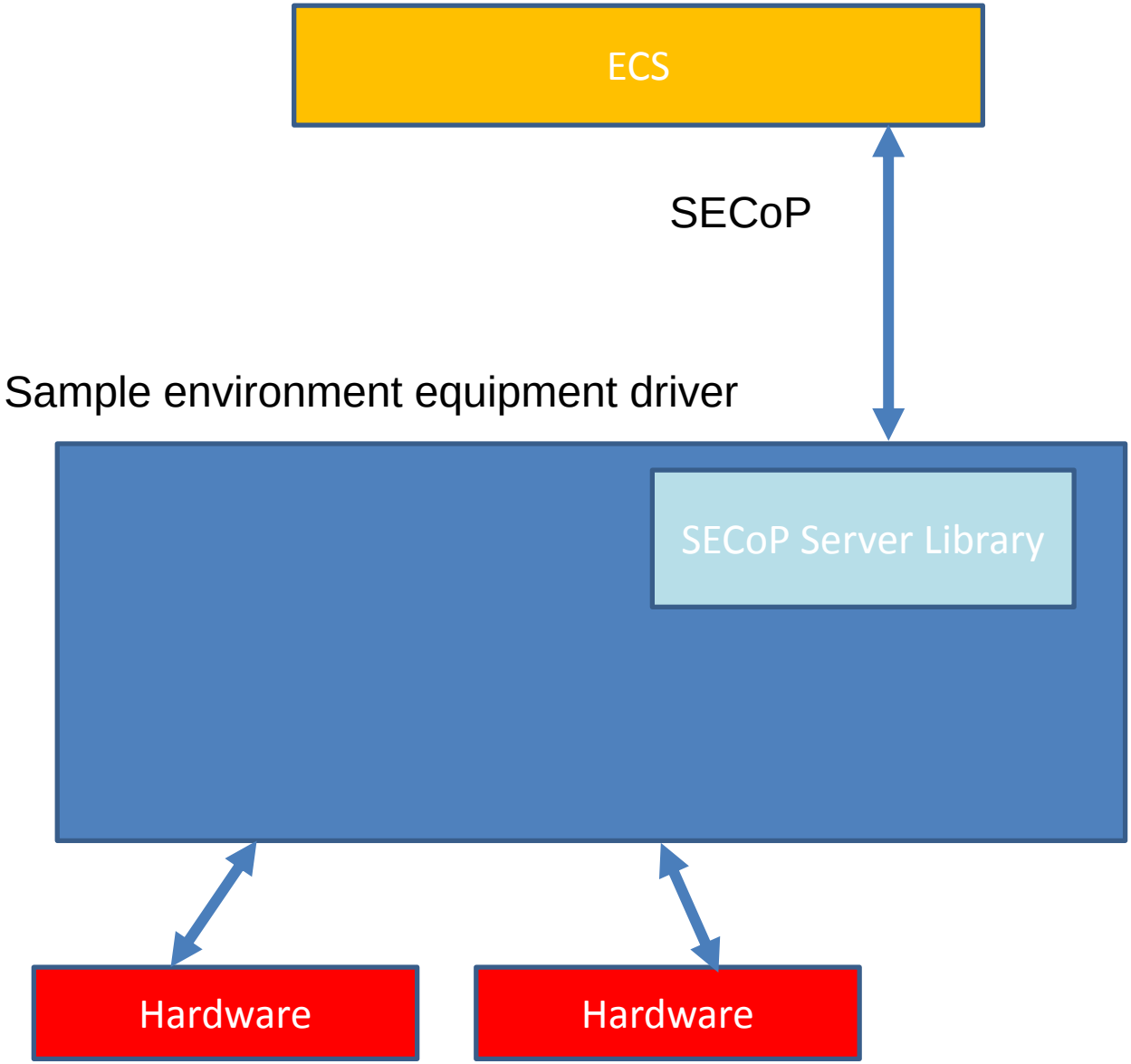
2) **FRAPPY,** a Python framework from Enrico Faulhaber (MLZ), allowing to program a complete SEC node including drivers. Successfully used (in a test setup) for equipment from MLZ to PSI and ILL, and for one smaller lab project at PSI.

   a collaborative effort (PSI jumped in – somebody else wants to join?)

SECoP Server Library



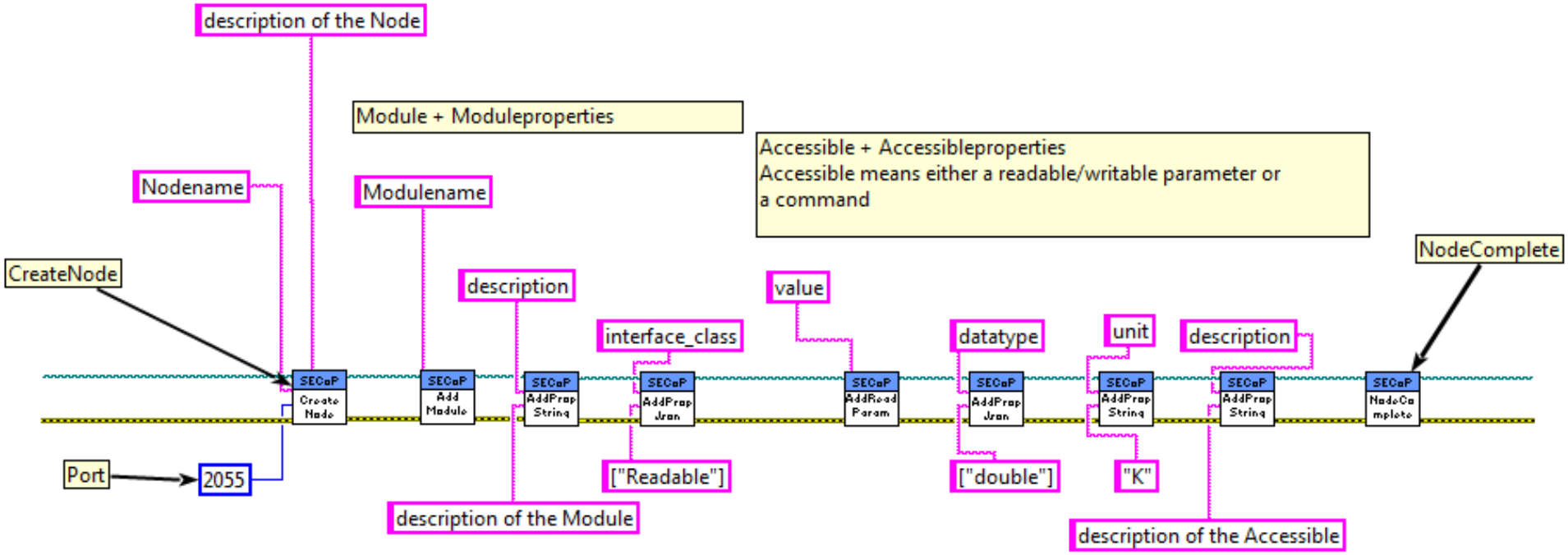
SECoP Server Library



HZB Helmholtz Zentrum Berlin

LabVIEW

Creating the SECnode



What still needs to be done?

Next steps

- Finalize next SECoP Version
- Write documentation / report

- Adapt (again) server library for C/C++
- Finish client library for C/C++
- Integration in ECS

- More tests with real equipment during real experiments
- Contact to industry
- Define complex interface classes

HZB Helmholtz Zentrum Berlin

SECoP

simple, inclusive, self explaining provides metadata

www.github.com/SampleEnvironment/SECoP

ISSE

SINE2020

Thank You!

Meeting in Grenoble March 2019

HZB Helmholtz Zentrum Berlin