Motion Control at PSI

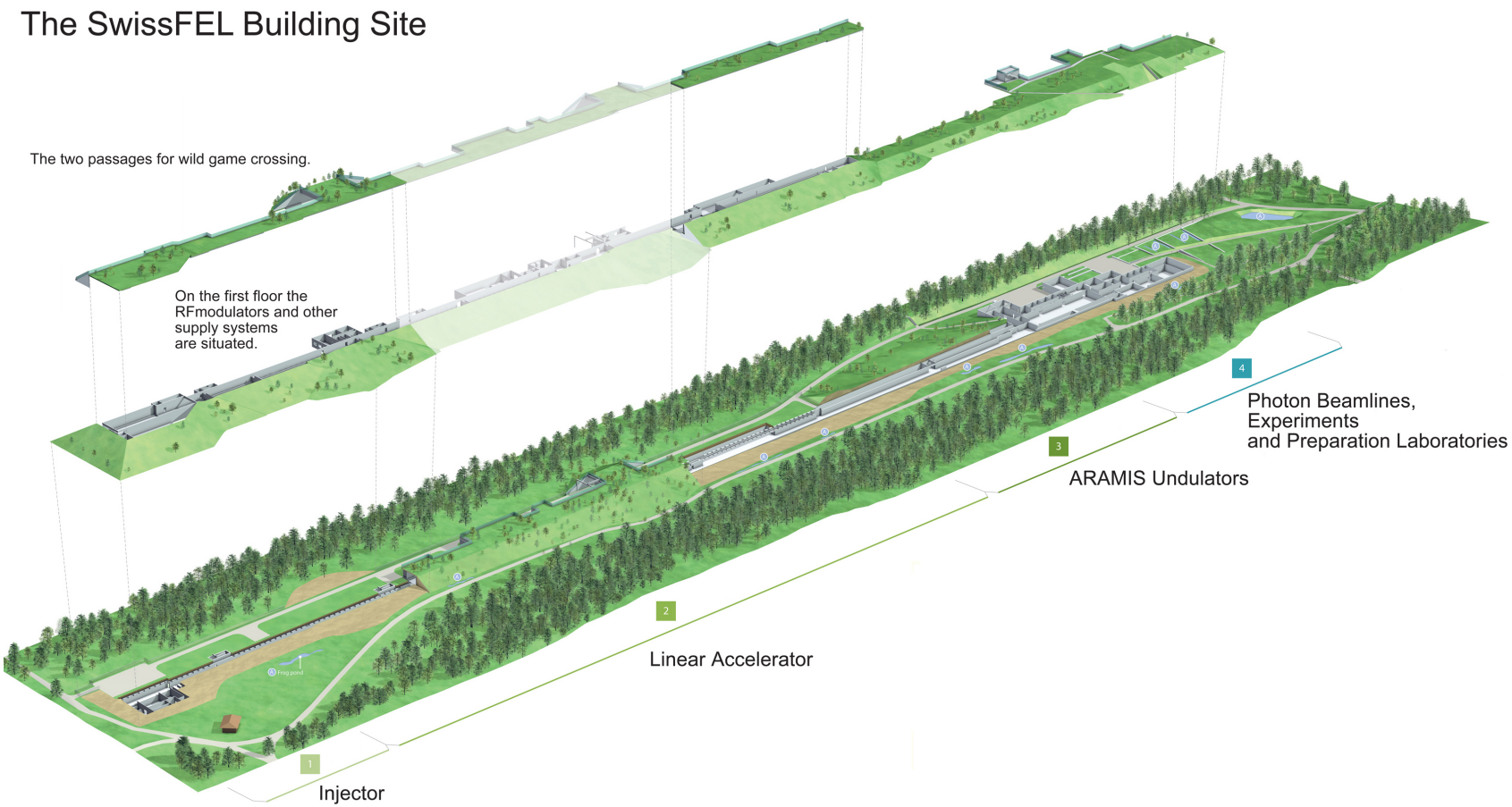
What's after the PowerBrick?

Niko Kivel

Paul Scherrer Institute, 5232 Villigen PSI, Switzerland. Tel: +41 56 310 2226; E-mail: niko.kivel@psi.ch

EPICS Collaboration Meeting, Motion Control Workshop,
2019-06-03, ITER France

- Aramis beam line complete
- Athos beam line, first light end of 2019
- Two endstations (Alvra and Bernina) operational, third (Cristallina) will follow in 2020
- First Athos endstation (AMO) will receive first light in Q2/2020



Motion components overview

controllers

- DeltaTau PowerBrick LV IMS PSI (65 Aramis + 55 Athos)
- Schneider Electronics (MForce, MDrive)
- SmartAct MCS (18 ch Rack version and stand-alone versions)
- various others (Newport, PI, ...)

Motion components overview

controllers

- DeltaTau PowerBrick LV IMS PSI (65 Aramis + 55 Athos)
- Schneider Electronics (MForce, MDrive)
- SmartAct MCS (18 ch Rack version and stand-alone versions)
- various others (Newport, PI, ...)

SwissFEL in service (> 800)

- $\simeq 350$ controlled by PB
- $\simeq 120$ Schneider Steppers
- $\simeq 80$ SmarAct actuators
- Beckhoff EtherCAT for insertion devices

Motion components overview

controllers

- DeltaTau PowerBrick LV IMS PSI (65 Aramis + 55 Athos)
- Schneider Electronics (MForce, MDrive)
- SmartAct MCS (18 ch Rack version and stand-alone versions)
- various others (Newport, PI, ...)

SwissFEL in service ($>$ 800)

- $\simeq$ 350 controlled by PB
- $\simeq$ 120 Schneider Steppers
- $\simeq$ 80 SmarAct actuators
- Beckhoff EtherCAT for insertion devices

SLS in service ($>$ 2000)

- for SLS 2.0 upgrade $>$ 1600 axes will need new driver
- need motorRecord
- mainly low demands on motion control

Future / Progess

- Does the PB have a future at PSI?

Future / Progess

- Does the PB have a future at PSI?

→ Yes and No. It'll stay in SwissFEL for now, but will not be used for SLS 2.0

Future / Progess

- Does the PB have a future at PSI?

→ Yes and No. It'll stay in SwissFEL for now, but will not be used for SLS 2.0

- When do we abandon it?

Future / Progess

- Does the PB have a future at PSI?

→ Yes and No. It'll stay in SwissFEL for now, but will not be used for SLS 2.0

- When do we abandon it?

→ as soon as possible → controled phase out

Future / Progess

- Does the PB have a future at PSI?

→ Yes and No. It'll stay in SwissFEL for now, but will not be used for SLS 2.0

- When do we abandon it?

→ as soon as possible → controled phase out

- What will replace it?

Future / Progess

- Does the PB have a future at PSI?

→ Yes and No. It'll stay in SwissFEL for now, but will not be used for SLS 2.0

- When do we abandon it?

→ as soon as possible → controled phase out

- What will replace it?

→ We don't know yet ... it most likely → certainly will involve EtherCAT.

it's most likely not going to be a vendor-bound solution (Omron, Beckhoff, ...)

Evaluation

ECMC

- direct EPICS implementation
- some limitations (s-curve, constant veloctity, ...)

OpenPLC Motion

- OPC UA $\rightarrow$ EPICS
- needs license for run time library and OPC UA

Beckhoff TwinCAT

- s7 $\rightarrow$ EPICS
- Windows in the field

ECMC: Architecture

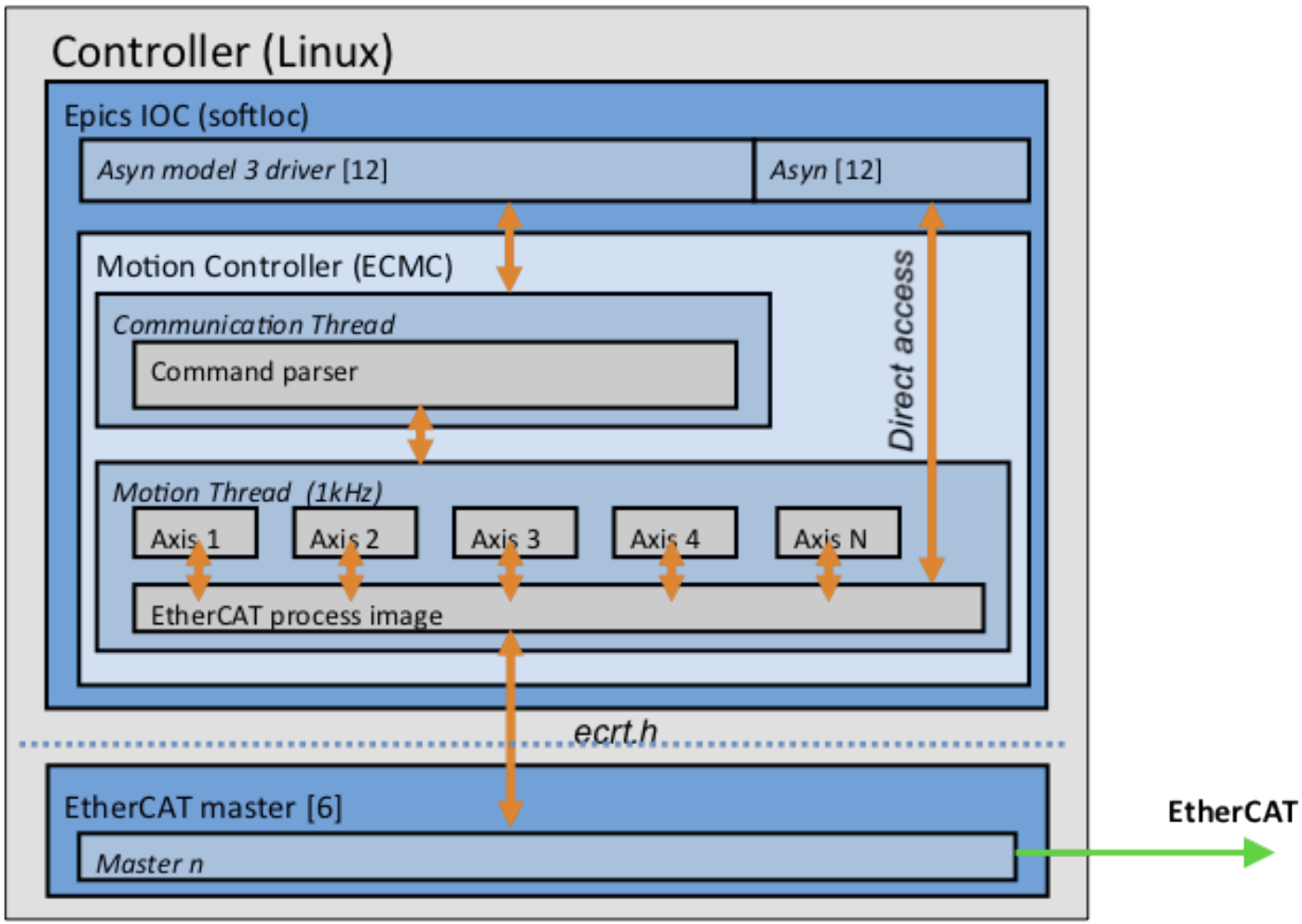


figure courtesy of ESS

ECMC at PSI

Software

- supported kernels:
  - RHEL 7: 3.10 PSI default
  - 3.10 PREEMPT_RT (Cern)
- tried to compile IgH master with lastest community patches and dedicated NIC-drives → failed!
- PACKAGING, checkout ▸ github

ECMC at PSI

Software

- supported kernels:
  - RHEL 7: 3.10 PSI default
  - 3.10 PREEMPT_RT (Cern)
- tried to compile IgH master with lastest community patches and dedicated NIC-drives → failed!
- PACKAGING, checkout ▸ github

Hardware

- various Beckhoff terminals (BiSS needed asap!)
- MicroEpsilon capaNCDT6500

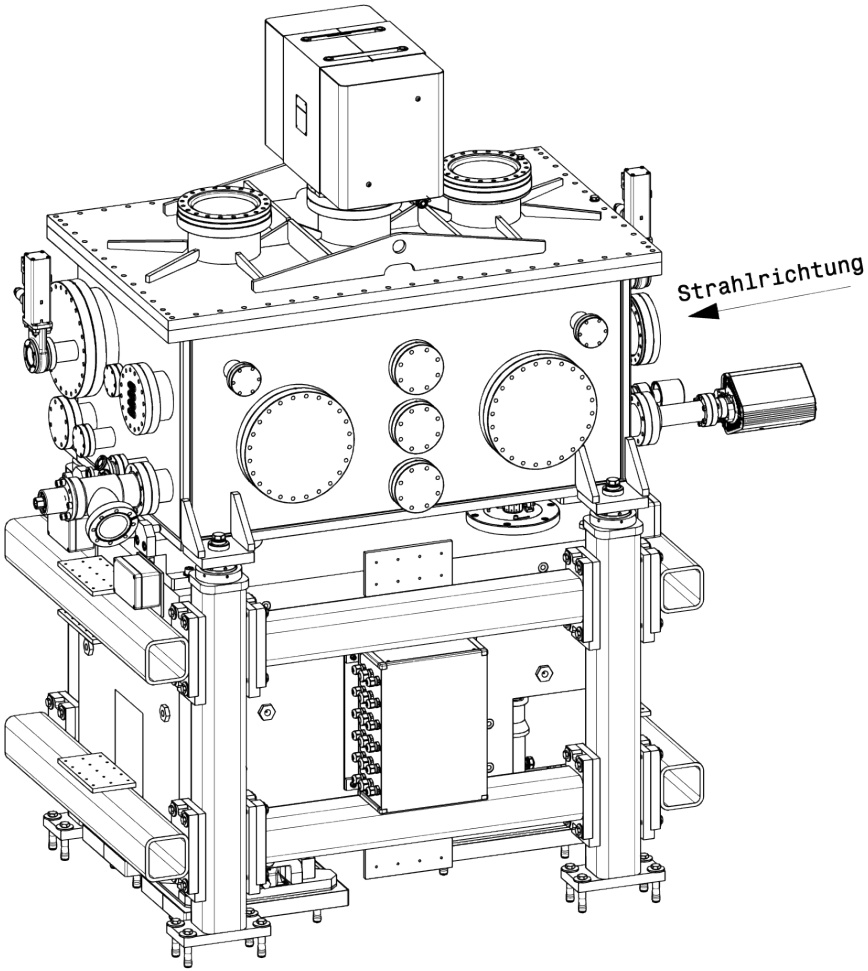
PAUL SCHERRER INSTITUT PSI

Example: IOC with ECMC_config

```
require ECMC_config kivel

# slave 0 (ecmcEK1100-EtherCatCoupler)
$(SCRIPTEXEC) $(ECMC_config_DIR)addSlave.cmd,
        "SLAVE_ID=0, HW_DESC=ecmcEK1100-EtherCatCoupler"
# slave 1 (ecmcEL1018-digitalInput)
$(SCRIPTEXEC) $(ECMC_config_DIR)addSlave.cmd,
        "SLAVE_ID=1, HW_DESC=ecmcEL1018-digitalInput"
# slave 2 (ecmcEL2808-digitalOutput)
$(SCRIPTEXEC) $(ECMC_config_DIR)addSlave.cmd,
        "SLAVE_ID=2, HW_DESC=ecmcEL2808-digitalOutput"
# slave 4 (ecmcEL5101-32bit-IncEncoder)
$(SCRIPTEXEC) $(ECMC_config_DIR)addSlave.cmd,
        "SLAVE_ID=4, HW_DESC=ecmcEL5101-32bit-IncEncoder"
# slave 9 (ecmcEL7037-StepperDrive)
$(SCRIPTEXEC) $(ECMC_config_DIR)configureSlave.cmd,
        "SLAVE_ID=9, HW_DESC=ecmcEL7037-StepperDrive, CONFIG=-Motor-Nanotec-ST4118L1804-

# Apply hardware configuration
$(SCRIPTEXEC) ($(ECMC_config_DIR)applyConfig.cmd)

# go active
$(SCRIPTEXEC) ($(ECMC_config_DIR)setAppMode.cmd)
```

OOMH

- tripod (3 motors):
  - translation (Y)
  - pitch
  - roll
- x-translation (1 motor)
- yaw (1 motor)
- bender (2 motors)
- 2 piezo LEGS (PUL/DIR)
- absolute encoders (BiSS)
- incremental encoder
- capaNCDT6500 capacitive displacement sensor



Timing and Event system

What works already

- bsread (EPICS)
- EVR trigger → EtherCAT terminal (EL5101 incr. enconder)

Timing and Event system

What works already

- bsread (EPICS)
- EVR trigger → EtherCAT terminal (EL5101 incr. enconder)

What should work

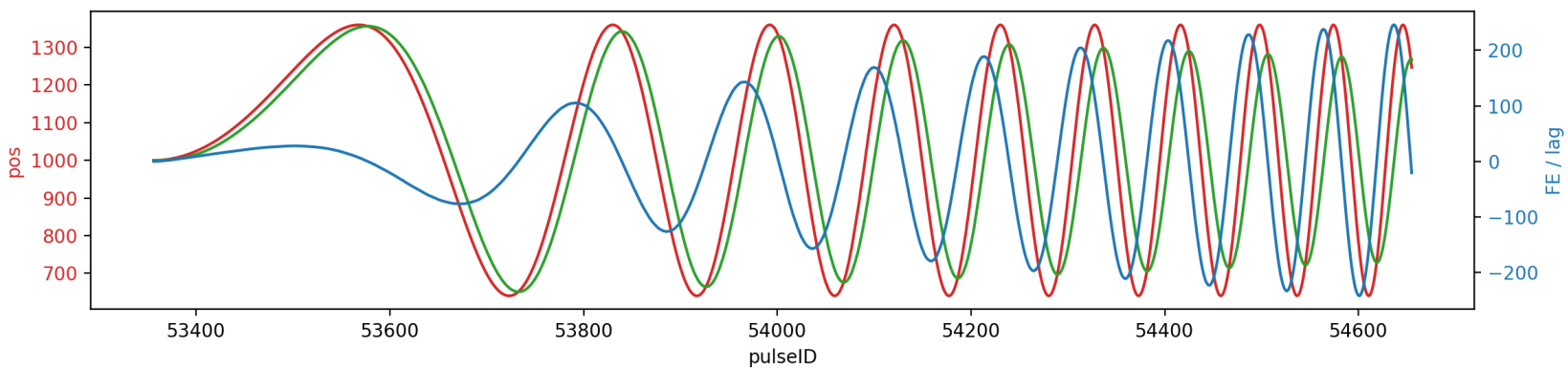
- sync EtherCAT clock to event system (distributed clock)

What doesn't work (yet)

- bsread (C++/API)
- Event handling in ECMC (PLC)

bsRead: Example

- PC-Hardware
  - standard PSI-PC with i5 processor
  - PCIe EVR 300
  - HP 4-Port server grade NIC
- EtherCAT-Hardware
  - EK1100: coupler
  - EL7041-1000: stepper driver
  - EL5001: SSI-encoder terminal
- ECMC: 1 kHz, closed-loop, single axis
- bsread: 100 Hz, local HDF5 storage (no data buffer)



Conclusions

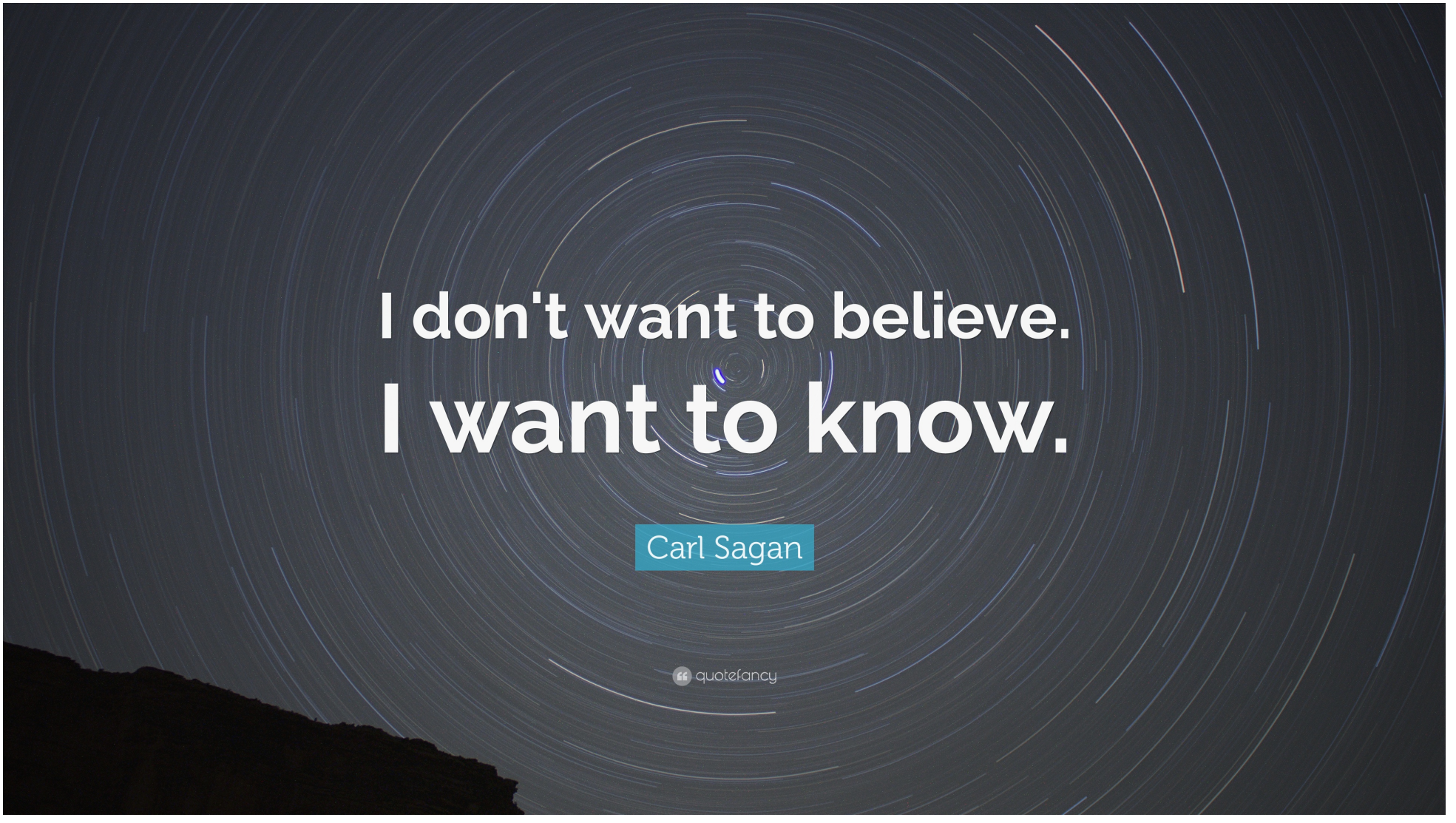
- ECMC configuration is now extremely straight forward with ECMC_config
- auto discover with basic setup script will follow soon.
- PLCs will solve present short commings in terms of homing of complex systems
- The code base is cluttered and hardly possible to work on for outsiders → refactor (Anders started this already).
- make it EPICS 7 compliant ...

Conclusions

- ECMC configuration is now extremely straight forward with ECMC_config
- auto discover with basic setup script will follow soon.
- PLCs will solve present short commings in terms of homing of complex systems
- The code base is cluttered and hardly possible to work on for outsiders → refactor (Anders started this already).
- make it EPICS 7 compliant ...

verdict

I'd start deploying as soon as possible, but management is not fully convinced yet.



Thank you