OAK RIDGE National Laboratory

EPICS Automation

Kay Kasemir

Feb. 2022

ORNL is managed by UT-Battelle, LLC for the US Department of Energy

U.S. DEPARTMENT OF ENERGY

Control System

.. should support automated control.

How can EPICS do this?

Monitoring, Supervisory Control

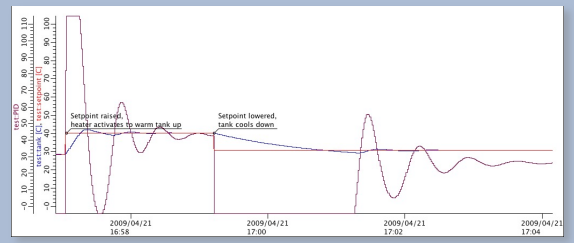
IOC

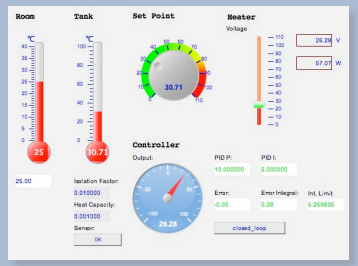
```
record(ai, "sensor")
{
  field(DTYP, "SensorXYZ")
  field(INP,  "@bus1 signal2")
  field(SCAN, "1 second")
  ..

record(ao, "voltage")
{
  field(DTYP, "PowerSupplyABC")
  field(OUT,  "@bus2 signal4")
  field(SCAN, "Passive")
  ..
```

Channel Access 'monitor'

Channel Access 'put'

User Interface





Automation via Records on IOC

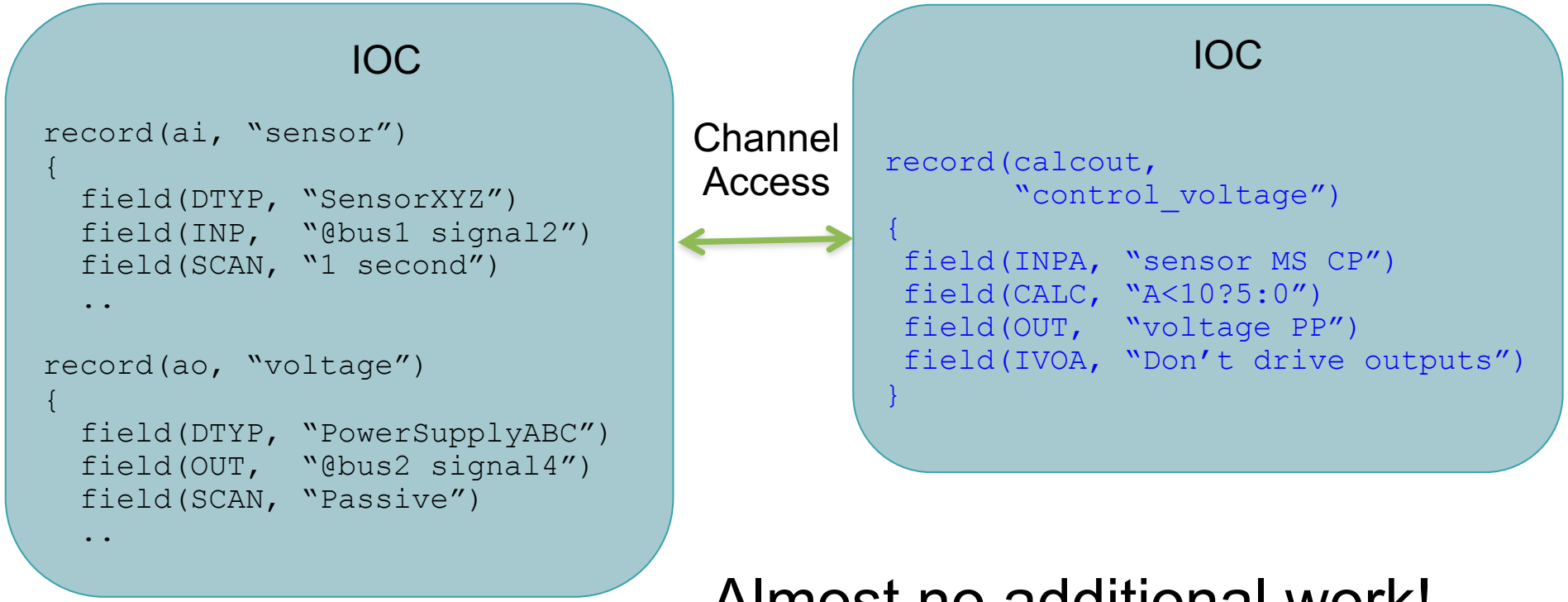
IOC

```
record(ai, "sensor")
{
  field(DTYP, "SensorXYZ")
  field(INP,  "@bus1 signal2")
  field(SCAN, "1 second")
  ..

record(calcout, "control_voltage")
{
  field(INPA, "sensor CP")
  field(CALC, "A<10?5:0")
  field(OUT,  "voltage PP")
}

record(ao, "voltage")
{
  field(DTYP, "PowerSupplyABC")
  field(OUT,  "@bus2 signal4")
  field(SCAN, "Passive")
  ..
```

Data flow driven,
periodic,
steady-state control:

1. Read inputs
2. Compute desired outputs
   a) calcout to write ao.VAL
   b) calc, then use DOL and OMSL=closed_loop in ao
3. Write outputs

Distribute Records onto different IOCs



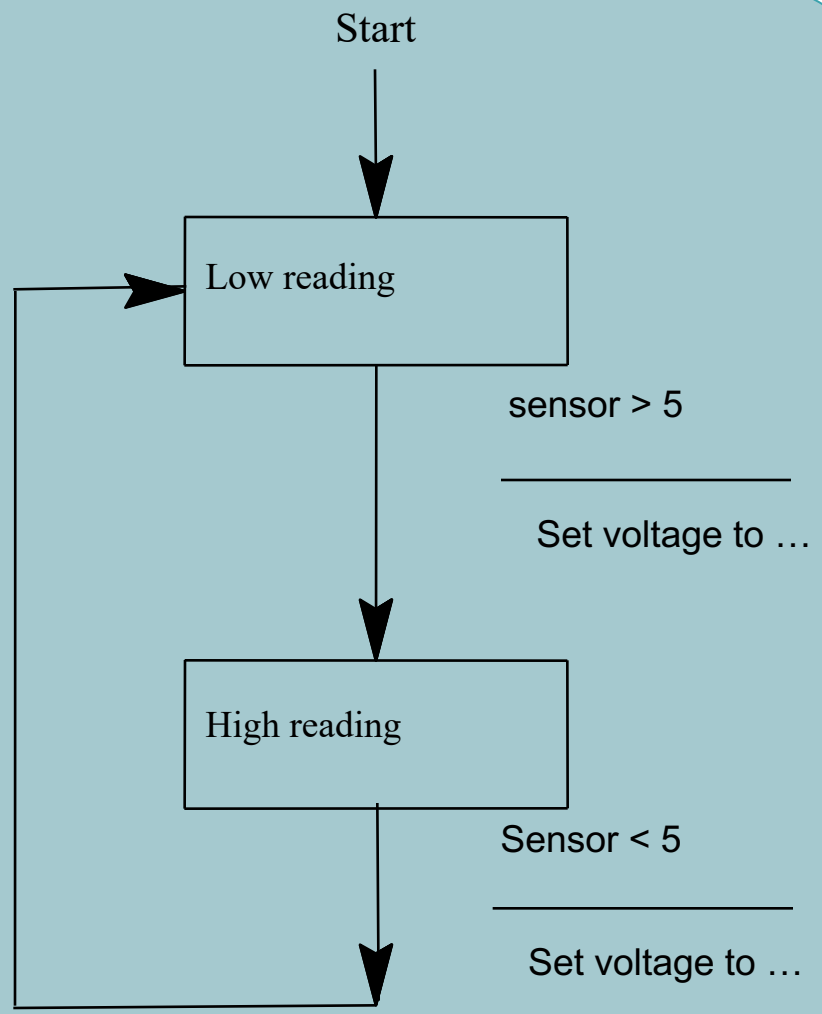
Almost no additional work!

Anticipate network issues;
see 'MS', 'IVOA'

Automation via State Machine

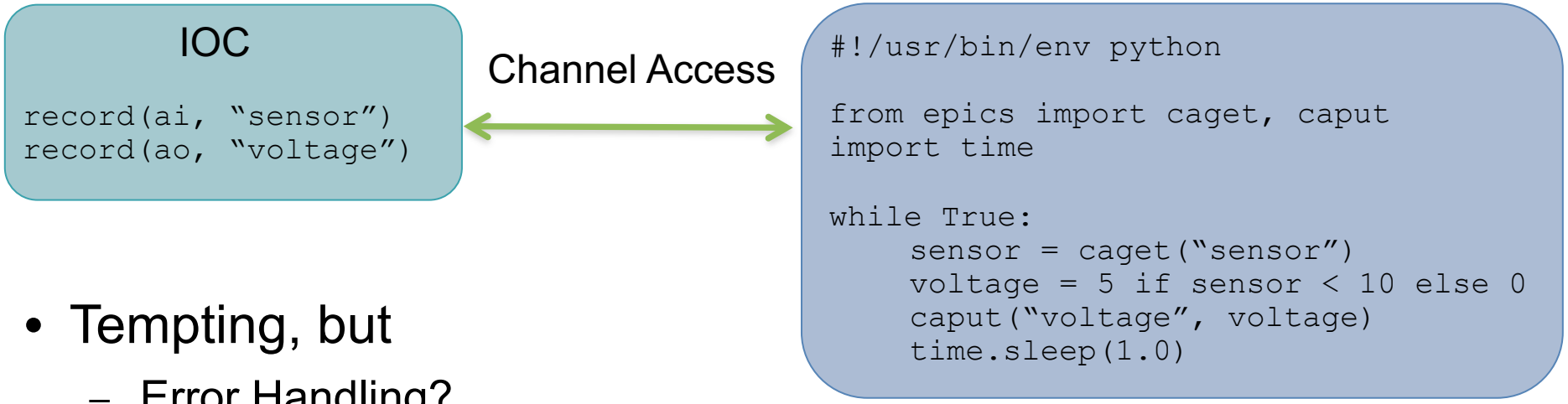
IOC

```
record(ai, "sensor")
{
  field(DTYP, "SensorXYZ")
  field(INP,  "@bus1 signal2")
  field(SCAN, "1 second")
  ..

record(ao, "voltage")
{
  field(DTYP, "PowerSupplyABC")
  field(OUT,  "@bus2 signal4")
  field(SCAN, "Passive")
  ..
```



"Sequencer", "State Notation Language":
Event driven, on-demand, stateful

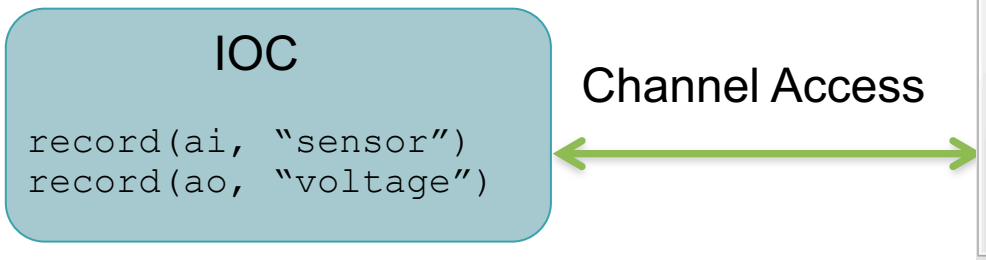
Automation via Scripts



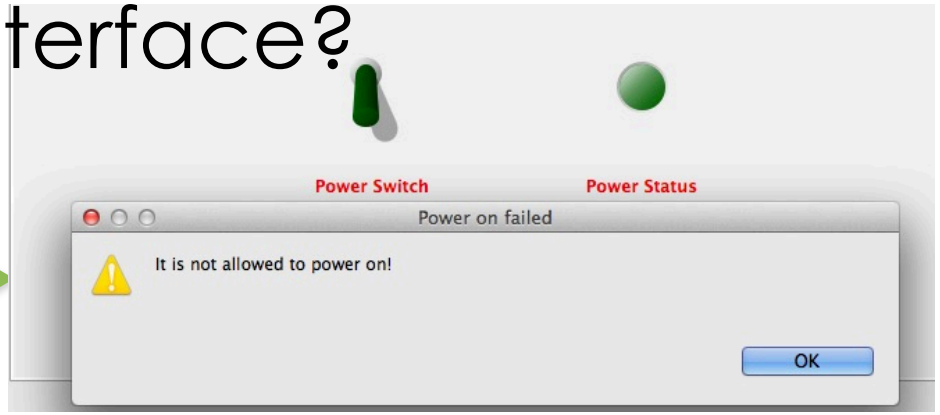
- Tempting, but
  - Error Handling?
  - caget? caput?

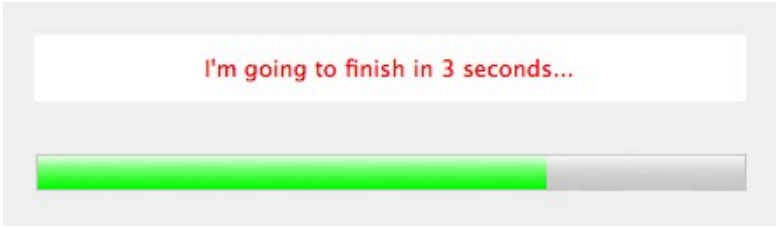
    Monitor; Connect once, then re-use the connection (PyEpics actually does this)
  - Handle disconnects, re-connects
- Should have 'console', run under procServ
  - IOC has shell
- Long-term maintenance of "Fred's script"?
  - Calc record has CALC, SCAN, INPA, ..

Automation via User Interface?



Channel Access





Check allowed values?

- What if other CA client writes to PV?
  Use DRVH, DRVL, calc records, .. to perform check on IOC, not in each user interface

Automation scripts?

- What if users open multiple user interfaces?
- What if GUI crashes (which is more likely than IOC)?

Keep user interface as just that!

Automation with EPICS

✓ Records

- Steady-data, data flow driven operations
- Continuous: Read input, compute, write output
- Limited conditional processing: calcout.OOPT

✓ State Notation Language

- On-demand, event driven
- Stateful: In State X, if Y happens, ..

? Scripts

- Useful for “I need this just once, but I need it now”
- Permanent “Python IOCs” require effort similar to IOCs

Automation via Operator Interface

- UI should never do anything