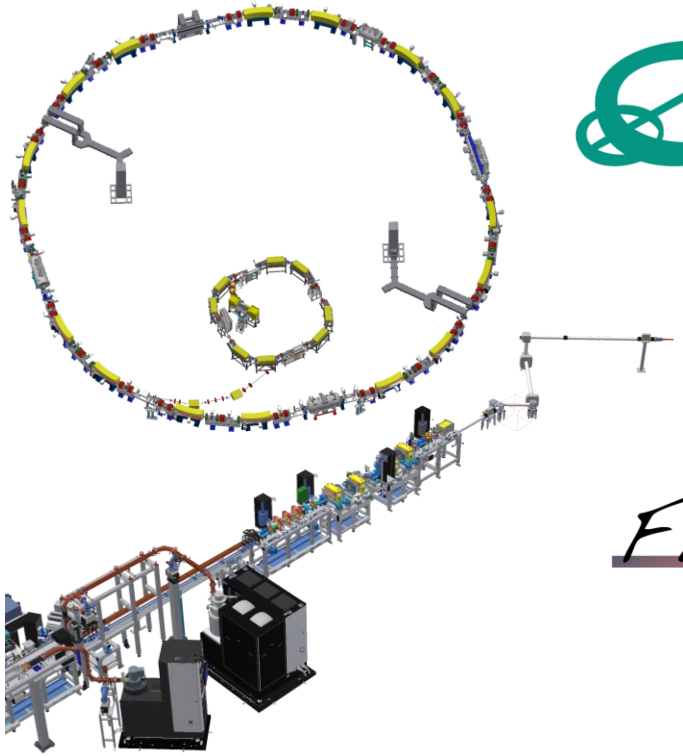
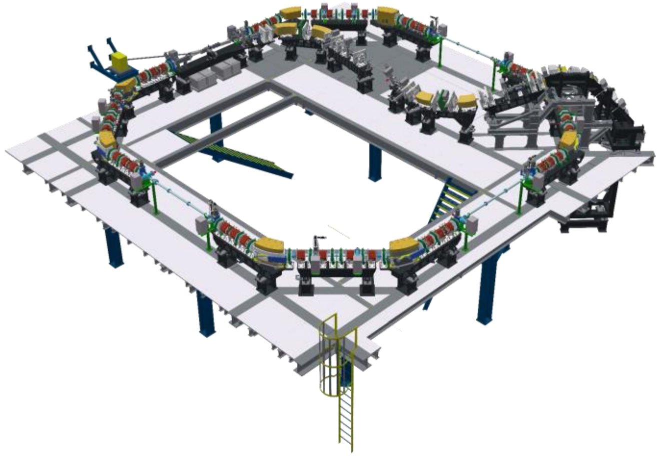
Concepts for Gating EPICS Alarm

E. Blomley & J. Gethmann

KIT Accelerators

- KARA (Karlsruhe Research Accelerator)
  - Mix of user operation and accelerator research
- FLUTE (Ferninfrarot Linac- und Test-Experiment)
  - Experiment driven without fixed beam schedule
- cSTART
  - Storage ring for non-equilibrium physics
  - FLUTE and laser plasma accelerator as injector
  - Installation starts ~ Mid 2026, commissioning ~ 2027
- Software stack
  - EPICS 7 (still Channel Access only)
  - Ubuntu (24.04) for terminals & servers
  - Most servers run inside VM





Motivation: Gating Alarms?

```
record(bi, Kicker:Status:On) {
    field(ZSV,  “MAJOR“)
    field(OSV, “NO_ALARM“)
}
```

- Make “alarms” (dis)appear based on certain conditions
  - Kicker magnet **OFF** → **Alarm,** **ON** → **No alarm**
  - But only in injection mode. Otherwise:
  - Kicker magnet **OFF** → **No alarm,** **ON** → **Alarm,**

- “Alarms” in EPICS and this talk:
  - The alarm **severity** and alarm **threshold fields** of records
  - “Alarms” in EPICS can also be seen as notifications:
  - Help operators to identify & fix potential issues

```
record(ai, Magnet:Current) {
    field(LOLO,  “5“)
    field(LLSV, “MAJOR“)
    field(LOW,  “8“)
    field(LSV, “MINOR“)
    field(HIGH,  “14“)
    field(HSV, “NO_ALARM“)
}
```

- Not talking about actual (hard-wired) alarms & interlocks: machine protection, personal safety systems, required by regulations, …

~~Gating Alarms~~ → Dynamic EPICS notifications to support operators

Examples

Kicker On

If in injection mode

```
record(bi, Kicker:Status:On) {
    field(ZSV,  “MAJOR“)
    field(OSV, “NO_ALARM“)
}
```

In all other operation modes

```
record(bi, Kicker:Status:On) {
    field(ZSV,  “NO_ALARM“)
    field(OSV, “MAJOR“)
}
```

Magnet Strength

If beam energy is low

```
record(ai, Magnet:Current) {
    field(LOLO,  “5“)
    field(LLSV, “MAJOR“)
    field(LOW,  “8“)
    field(LSV, “MINOR“)
    field(HIGH,  “14“)
    field(HSV, “MAJOR“)
    field(HIHI,  “20“)
    field(HHSV, “MAJOR“)
}
```

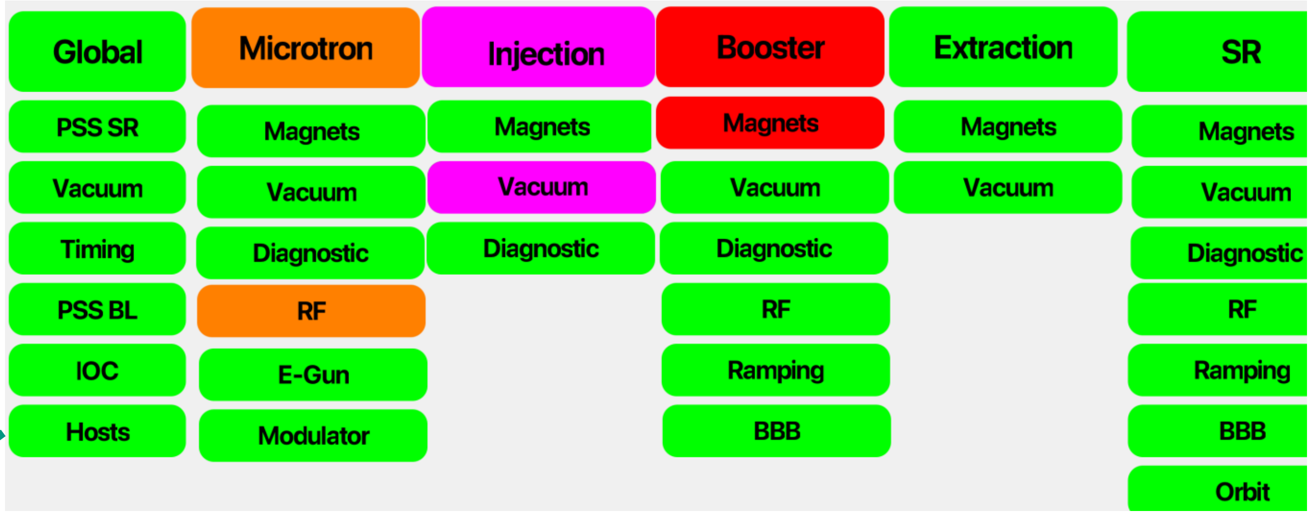
If beam energy is high

```
record(ai, Magnet:Current) {
    field(LOLO,  “5“)
    field(LLSV, “MAJOR“)
    field(LOW,  “8“)
    field(LSV, “MAJOR“)
    field(HIGH,  “14“)
    field(HSV, “NO_ALARM“)
    field(HIHI,  “20“)
    field(HHSV, “MINOR“)
}
```

If in machine development mode

```
record(ai, Magnet:Current) {
    field(LOLO,  “5“)
    field(LLSV, “NO_ALARM“)
    field(LOW,  “8“)
    field(LSV, “NO_ALARM“)
    field(HIGH,  “14“)
    field(HSV, “NO_ALARM“)
    field(HIHI,  “20“)
    field(HHSV, “NO_ALARM“)
}
```

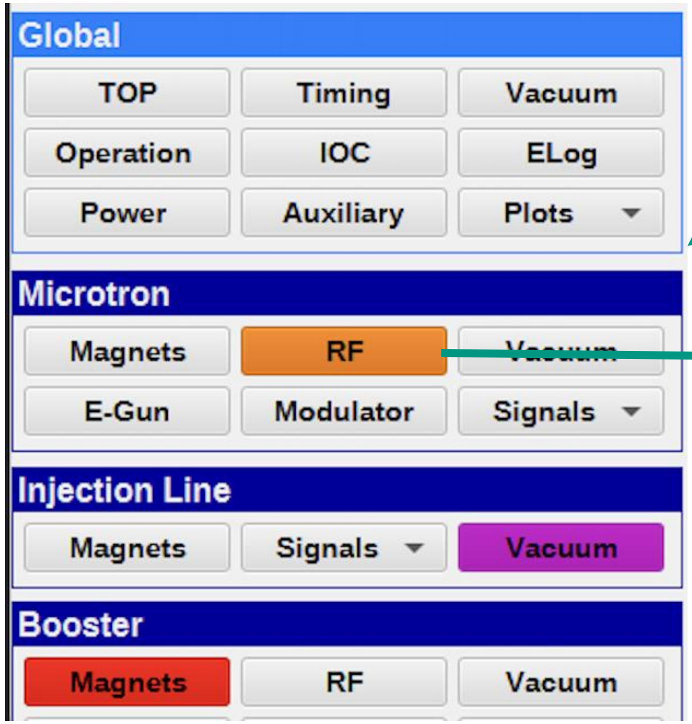
Use Case: Guide the operator



Global	Microtron	Injection	Booster	Extraction	SR
PSS SR	Magnets	Magnets	Magnets	Magnets	Magnets
Vacuum	Vacuum	Vacuum	Vacuum	Vacuum	Vacuum
Timing	Diagnostic	Diagnostic	Diagnostic		Diagnostic
PSS BL	RF		RF		RF
IOC	E-Gun		Ramping		Ramping
Hosts	Modulator		BBB		BBB
					Orbit

- Support operator to quickly identify potential issues
- Alarm summaries mapped to navigation menu
- Lead to panel where the issue can be fixed

1. and 2. level alarm hierarchy reflects our navigation

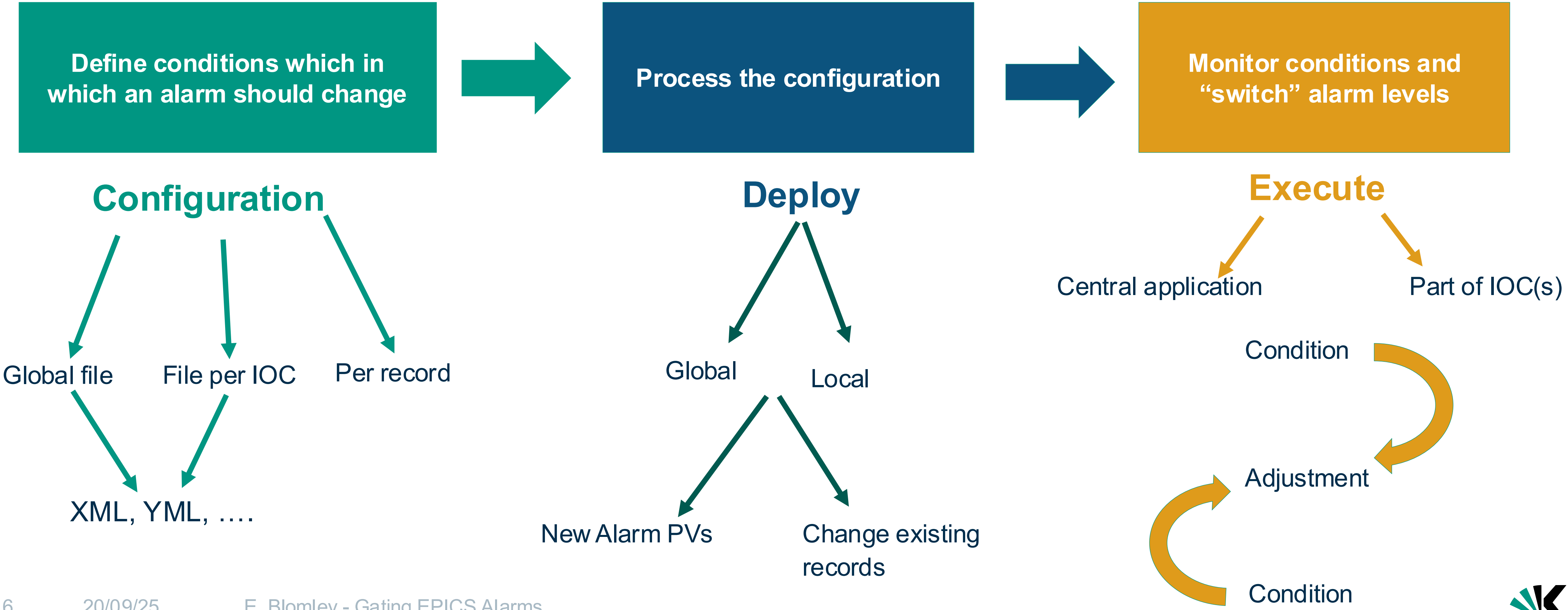


Opening the panel should clearly indicate the alarm (or lead to another panel): Would expect to directly see minor alarm on RF panel

Often not necessary to switch to alarm view!

System Design

The task: Design a system which provides such a functionality



Define conditions which in which an alarm should change

Process the configuration

Monitor conditions and “switch” alarm levels

Configuration – On Record Level

If in injection mode

```
record(bi, Kicker:Status:On) {
    field(ZSV,  “MAJOR“)
    field(OSV, “NO_ALARM“)
}
```

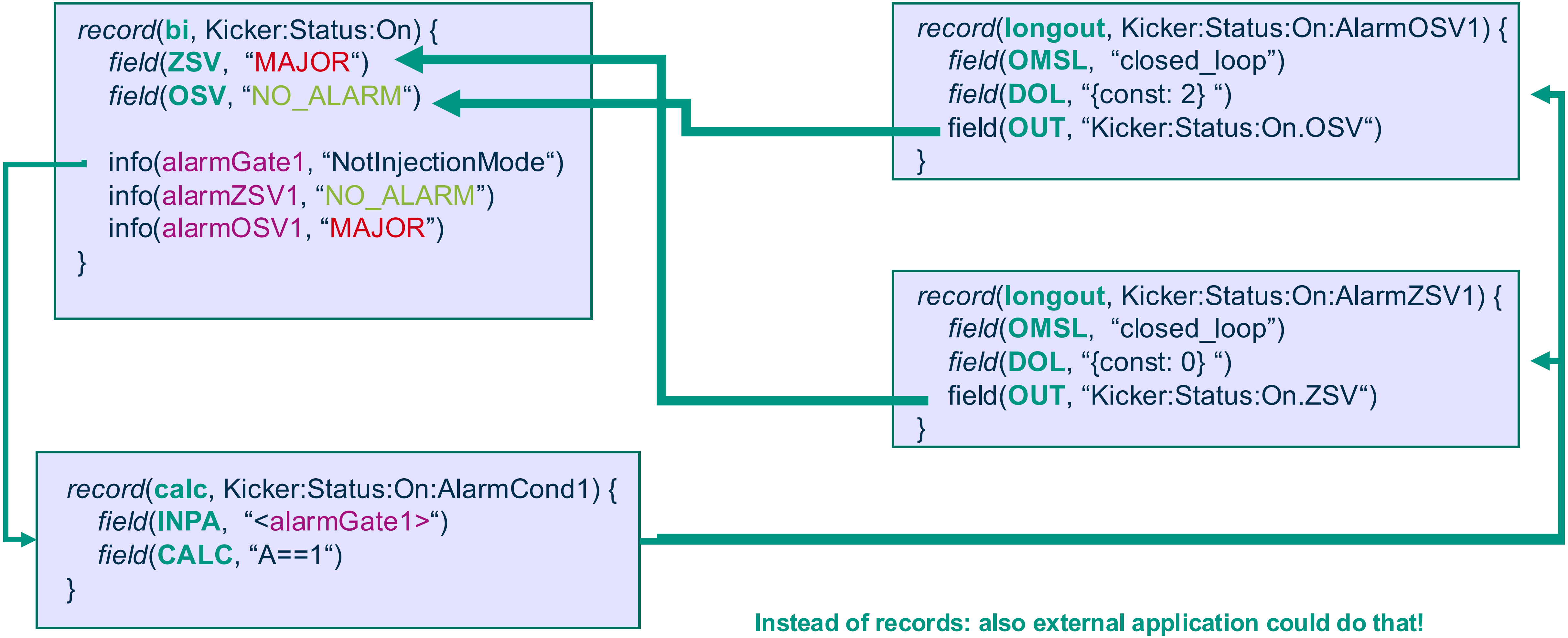
In all other operation modes

```
record(bi, Kicker:Status:On) {
    field(ZSV,  “NO_ALARM“)
    field(OSV, “MAJOR“)
}
```

```
record(bi, Kicker:Status:On) {
    field(ZSV,  “MAJOR“)
    field(OSV, “NO_ALARM“)

    info(alarmGate1, “NotInjectionMode“)
    info(alarmZSV1, “NO_ALARM”)
    info(alarmOSV1, “MAJOR”)
}
```

```
record(ai, Magnet:Current) {
    field(LOLO,  “5“)
    field(LLSV, “MAJOR“)
    field(LOW,  “8“)
    field(LSV, “MINOR“)
    field(HIGH,  “14“)
    field(HSV, “MAJOR“)
    field(HIHI,  “20“)
    field(HHSV, “MAJOR“)

    info(alarmGate1, „HighEnergyStatus“
    info(alarmLSV1, “MAJOR“)
    info(alarmHSV1, “NO_ALARM“)
    info(alarmHHSV1, “MINOR“)

    info(alarmGate2, „MachineOperation“)
    field(alarmLLSV2, “NO_ALARM“)
    info(alarmLSV2, “NO_ALARM“)
    info(alarmHSV2, “NO_ALARM“)
    info(alarmHHSV2, “NO_ALARM“)
}
```

Adjust alarm fields of existing record – using records



```
record(bi, Kicker:Status:On) {
    field(ZSV,  “MAJOR“)
    field(OSV, “NO_ALARM“)

    info(alarmGate1, “NotInjectionMode“)
    info(alarmZSV1, “NO_ALARM”)
    info(alarmOSV1, “MAJOR”)
}
```

```
record(longout, Kicker:Status:On:AlarmOSV1) {
    field(OMSL,  “closed_loop”)
    field(DOL, “{const: 2} “)
    field(OUT, “Kicker:Status:On.OSV“)
}
```

```
record(longout, Kicker:Status:On:AlarmZSV1) {
    field(OMSL,  “closed_loop”)
    field(DOL, “{const: 0} “)
    field(OUT, “Kicker:Status:On.ZSV“)
}
```

```
record(calc, Kicker:Status:On:AlarmCond1) {
    field(INPA,  “<alarmGate1>“)
    field(CALC, “A==1“)
}
```

Instead of records: also external application could do that!

Or an EPICS module?

Possible side-effects of in-place adjustments?

In-place adjustments

- **EPICS record processing**: Look at how alarm propagation behaves
- **Phoebus widgets**: Do (all) widgets pick up changes and adjust alarm borders etc?
- Does the **archiver** pick this up and logs correctly?
- How will this look like in the **alarm server** & logger?
- How about disconnects and **restarts**?

Additional alarm records

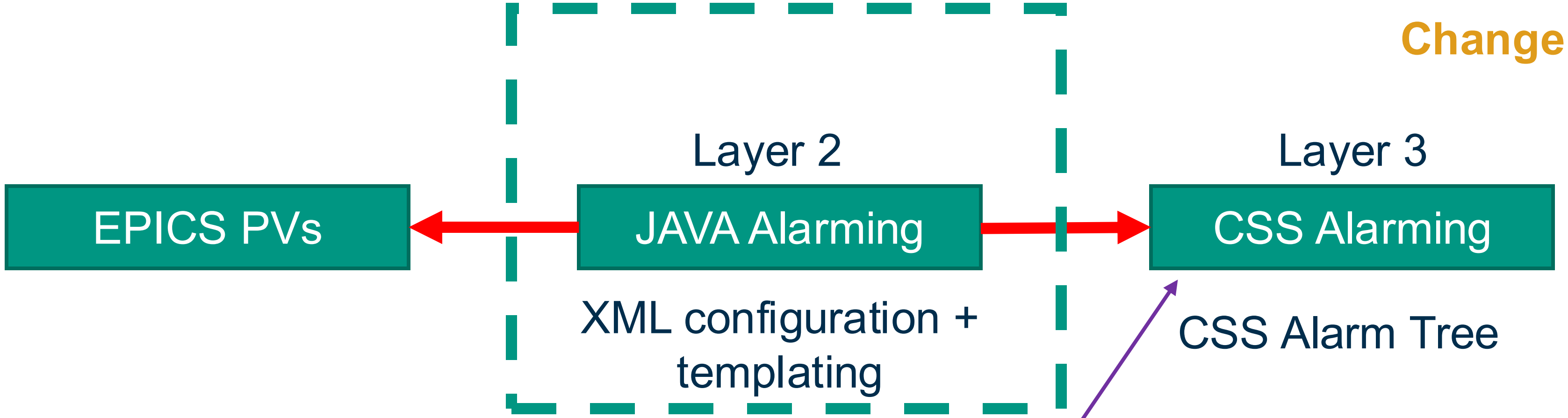
- **EPICS level:** value and alarm state now in separate records
- **Phoebus**: Add all PVs to panels? Use JavaScript??
- **Archiver:** Again, add alarm PVs also to archive?
- **Alarm server** & logger?
- How about disconnects and **restarts**?

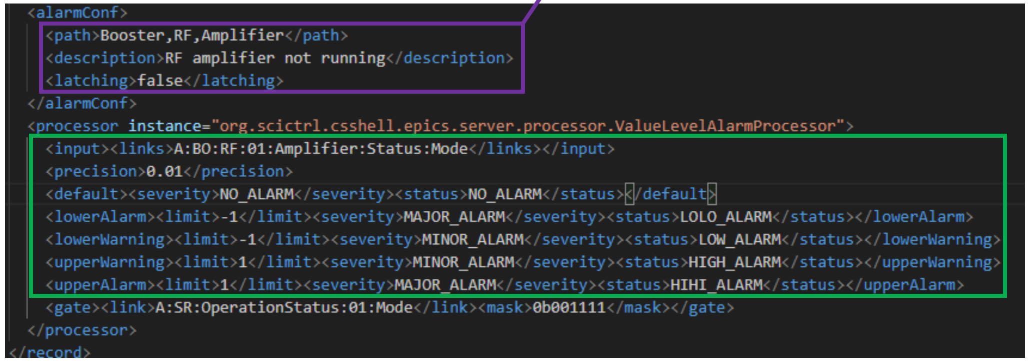
Putting it Together – Our current system

Central XML configuration

Central custom Java application creating additional alarm records

Change severity of alarm records





```
  <alarmConf>
    <path>Booster,RF,Amplifier</path>
    <description>RF amplifier not running</description>
    <latching>false</latching>
  </alarmConf>
  <processor instance="org.scictrl.csshell.epics.server.processor.ValueLevelAlarmProcessor">
    <input><links>A:BO:RF:01:Amplifier:Status:Mode</links></input>
    <precision>0.01</precision>
    <default><severity>NO_ALARM</severity><status>NO_ALARM</status></default>
    <lowerAlarm><limit>-1</limit><severity>MAJOR_ALARM</severity><status>LOLO_ALARM</status></lowerAlarm>
    <lowerWarning><limit>-1</limit><severity>MINOR_ALARM</severity><status>LOW_ALARM</status></lowerWarning>
    <upperWarning><limit>1</limit><severity>MINOR_ALARM</severity><status>HIGH_ALARM</status></upperWarning>
    <upperAlarm><limit>1</limit><severity>MAJOR_ALARM</severity><status>HIHI_ALARM</status></upperAlarm>
    <gate><link>A:SR:OperationStatus:01:Mode</link><mask>0b001111</mask></gate>
  </processor>
</record>
```

Alternative Approaches

- **YML file for each IOC → Python script to parse this → create additional alarm records during/after IOC init**
- **In-record configuration → process during IOC init (similar to autosave) → generate internal EPICS records** which modify the alarm fields of the actual record
- **In-record configuration → ChannelFinder picks up INFO fields → ChannelFinder PostProcessor used for central application**
- …

Technology Stack

- Our custom system is in operation for 10+ years, but we cannot **maintain** or develop the **custom** Java **code** anymore

- **Scaling** to potentially millions of PVs relevant? (not for us)

- Use **technology** which best **fits to us**? (No Java nor C++ developer)

- Make use of **complex features** such as ChannelFinder?

- Try to do it only with EPICS base and EPICS modules to avoid ANY **additional dependency**?

- Is there **interest** in the **EPICS community**?
  - If not, we might just go for implementing what is easiest and makes most sense for us

Quo Vadis?

- We are using gated alarms for a long while.. and plan to continue…
- Need to replace our current system and addition of new accelerator good opportunity
- But before committing to design, resources & time: **Looking for feedback**!

Feedback, opinions, discussions?

Still other approach?
Maybe you do something similar?

Community Interest?

edmund.blomley@kit.edu