

# pvaSrv – the IOC Side Bridge from pvAccess to an IOC

R. Lange, HZB / BESSY II

# Goals

- Seamless fit into a V4 (pvAccess) system
- Minimal footprint
- Minimal configuration

# pvaSrv on a 3.14 IOC

- Client can connect to any <record>.<field>
- Data is presented as an NTType
- pvAccess request options are supported in a best-effort approach

# Server Side

```
# st.cmd
```

```
[... load databases ...]
```

```
iocInit()
```

```
Starting iocInit
```

```
#####
```

```
## EPICS R3.14.12.3 $Date: Mon 2012-12-17 14:11:47 -0600$
```

```
## EPICS Base built Mar 13 2013
```

```
#####
```

```
iocRun: All initialization complete
```

```
[...]
```

```
pvaSrv
```

```
[...]
```

# Client Side

```
> caget int01
int01          0
> cainfo int01
int01
  State:          connected
  Host:           172.23.31.54:5064
  Access:         read, write
  Native data type: DBF_LONG
  Request type:   DBR_LONG
  Element count:  1

> pvget int01
int01          0
> pvinfo int01
CHANNEL   : int01
STATE     : CONNECTED
ADDRESS   : 172.23.31.54:5075

uri:ev4:nt/2012/pwd:NTScalar
  int value
  alarm_t alarm
    int severity
    int status
    string message
  time_t timeStamp
    long secondsPastEpoch
    int nanoSeconds
    int userTag
  display_t display
    double limitLow
    double limitHigh
    string description
    string format
    string units
  control_t control
    double limitLow
    double limitHigh
    double minStep
```

# pvaSrv on a 3.15 IOC (Planned)

- All the above, plus...
- dbGroup: set of PVs that can be accessed under a new name as a group
- Atomic operations on the records backing that group
- Definition of dbGroups statically or on-the-fly
- pvAccess request options implemented via server-side plug-ins