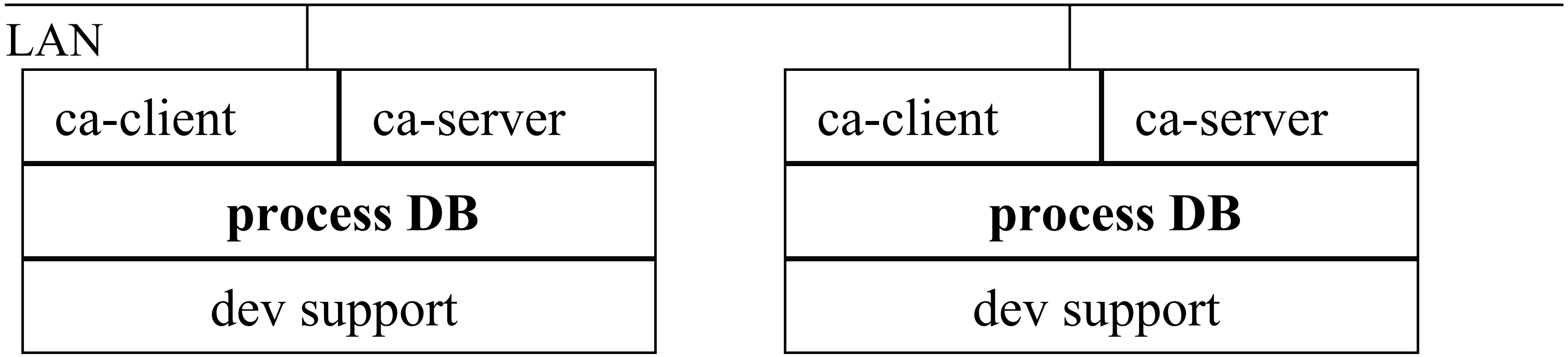
Experimental Physics and Industrial Control System (EPICS) Database

Bob Dalesio, June 27,2001

Outline

- How does the process database fit into EPICS
- Database configuration tools
- Process blocks for data acquisition and control
- Configurable functions in the process database

EPICS Process Database Is Distributed



Each IOC can contain one or more process databases

Each database contains its local directory - it is self contained

Each IOC contains its own scan tasks to support each scan period

Access to a database is given to all clients through the channel access server

References to a process variables in another IOC are done through the channel access client.

Process Databases Are Configured Off-line and Downloaded at Initialization

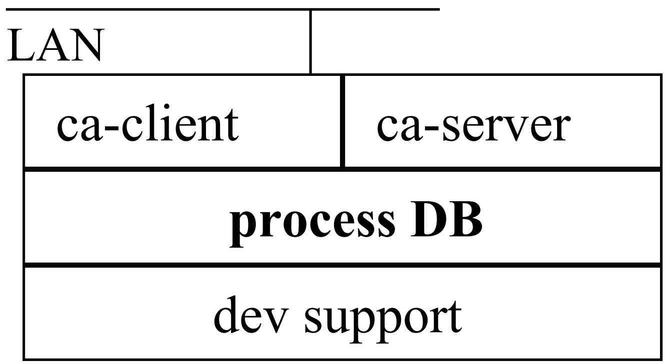
The Process Database is loaded as an ASCII file

It is configured (at least partially) using some configuration tool:

- GDCT - graphical configuration tool - nice menus - no hierarchy - no query
- Capfast - Commercial schematic package - good hierarchy - no query
- tcl/TK/DCT - A single record form
- any text editor
- Relational databases - nice query capability - no hierarchy

Downloaded at Initialization using NFS or FTP

The EPICS Process Database Is Used as a Low-level Interface to the Process



Process Blocks exist to implement various inputs:
analog inputs, binary inputs,
high-speed analog inputs, pulse counters

Process blocks exist to implement various outputs
analog outputs, binary outputs,
high speed analog outputs, position, timing

A Device layer separates the process function of interfacing I/O from the protocols required to communicate to various physical devices

The device layer is typically expected to cache the values and not delay in read/write from the process database

Asynchronous completion mechanisms are available when a delay is required

Process Blocks Are Connected to Perform Complex Functions

Process Blocks containing control / general algorithms:
- general purpose calculation: algebraic, logical, trigonometric, relational, if/else
- PID, select, ramp, C-subroutine interface

Process blocks can be used to control complex devices:
- beam position monitor, power supply

Process blocks can be used for data collection
- scan, histogram, compression

New process block types can be easily added - useful for repetitious functions

Each Process Block is Comprised of a Set of Fields That Determine the Operation of Each Instance

I/O fields are used for data flow

SCAN fields determine scheduling

CONVERSION fields are used to convert I/O

ALARM fields specify alarm conditions and severity

CHANNEL ACCESS fields control server notification and permission

CONTROL fields are used to configure closed loop control behavior

OPERATOR parameters are configured for display

Algorithmic specific parameters are used to configure individual functions

Minimum Configuration - Analog Input

ValveRB	
SCAN	*.1 Second*
INP	*#C0 S0*
DTYP	*XY566*
LINR	*Linear*
EGUF	*100*
EGUL	*0*
EGU	*%*

Analog Input

Reads the value for logical card 0 logical signal 0 Every .1 second from the Xycom566 device and scales the raw input between 0 and 100 percent. The raw value read from the hardware is put into the RVAL field and the converted value is put into the VAL field.

NOTE: device support sets the engineering units slope given EGUF and EGUL. The device support layer knows the number of bits of resolution of the hardware and the format

Minimum Configuration - Discrete Input

TwoBitValveRB	
SCAN	*I/O Intr*
INP	*#C0 S4*
DTYP	*UniDig24*
NOBT	*2*
ZRVL	*0*
ONVL	*1*
TWVL	*2*
THVL	*3*
ZRST	*Travelling*
ONST	*Open*
TWST	*Closed*
THST	*Error*
THSV	*Major*

Mulibit Binary Input

Reads the value for logical card 0 bits 4 and 5 (counting from 0) every time there is any change on the UniDig24 card. The result to the four possible states. Both limits closed is an error condition.

Minimum Configuration – Binary Output

OneBitValveCtrl	
SCAN	*Passive*
OUT	*#C0 S4*
DTYP	*UniDig24*
ZRST	*Closed*
ONST	*Open*
HIGH	1

Binary Output

Write the VAL field to the UniDig24 logical card 0, bit 4 (5^{th} bit) every time the operator makes a change – and then set it to 0 after 1 second.

I/O Fields are Used for Data Flow

DTYP

- Software type - constant or database address
- Name of a hardware device

INP (OUT)

- Database Address - local or remote
  - Process Passive (PP/NPP)
  - CA Links (CA/CPP-In Only)
  - Maximize Severity (MS/NMS)
- Hardware address - logical or physical

SIM

- Simulation fields include value and location

Scan Fields Determine Scheduling

SCAN - Scan Mechanism

- Periodic - .1, .2, .5, 1, 2, 5, 10 (menu driven)
- Hardware Event - originates from the driver layer
- Software Event - from an event record or program

PHAS - Order within a scan mechanism

EVNT - Event number for a software event

FLNK - Forward Processing Link

- Process this record next - if and only if it is Passive
- Forward links to other IOCs must link to .PROC

SDIS - Scan Disable Location

DVAL - Scan Disable Value

Process and Data Flow

Temp1	
SCAN	*.1 Second*
PHAS	*0*
INP	*#C0 S0*
DTYP	*XY566*
FLNK	*0*

Analog Input

Reads the value for logical card 0 logical signal 0
Every .1 second from the Xycom566 device

Temp1	
SCAN	*I/O Event*
PHAS	*0*
INP	*#C0 S0*
DTYP	*DVX2502*
FLNK	*0*

Analog Input

Reads the value for logical card 0 logical signal 0
Whenever the DVX2502 driver posts an event
(Drivers can post events on interrupt - or have independent scan threads that post the events)

Process and Data Flow - Passive Scanning

Temp1	
SCAN	*Passive*
PHAS	*0*
OUT	*#C0 S0*
DTYP	*XY220*
FLNK	*0*

Binary Output

Write the value to logical card 0 signal 0 whenever
- a channel access put is done to this record
- a write is done from another record within this IOC
- a forward link from any other record points here
- a read is done from another record that specifies PP

Using Process Passive Links (PP/NPP)

AI_1	
SCAN	*Passive*
INP	*#C0 S0*

CALC_2	
SCAN	*1 second*
INPA	*AI_1 NPP*
INPB	*AI_1 PP*
CALC	*A-B*

Using Channel Access Links (CPP/CA)

Temp1	
SCAN	*1 second*
INP	*#C0 S0*
EGU	*DgC*
LINR	*TypeJdgc*
MDEL	*1*

Temp2	
SCAN	*I/O Intr*
INP	*#C0 S1*
EGU	*DgF*
LINR	*TypeJdgf*
MDEL	*5*

TEMP_ILOCK	
SCAN	*Passive*
INPA	*Temp1 CPP MS*
INPB	*Temp2 CPP MS*
CALC	*(A>100) || (B>100)*
FLNK	*Valve_1*

Valve_1	
SCAN	*Passive*
OUT	*#C0 S0*
OMSL	*Closed_Loop*
DOL	*Temp_ILOCK MS*
ZNAM	*Closed*
ONAM	*Open*

Roughly Equivalent Processing Chains

Ex. A

AI_1	
SCAN	*.1 Second*
PHAS	*0*
INP	*#C0 S0*

CALC_1	
SCAN	*.1 Second*
PHAS	*1*
INPA	*AI_1*

AO_1	
SCAN	*.1 Second*
PHAS	*2*
DOL	*CALC_1*
OMSL	*Closed-loop*

Ex. B

AI_2	
SCAN	*.1 Second*
INP	*#C0 S0*
FLNK	*CALC_2*

CALC_2	
SCAN	*Passive*
INPA	*AI_2*
FLNK	*AO_2*

AO_2	
SCAN	*Passive*
DOL	*CALC_2*
OMSL	*Closed-loop*

Ex. C

AI_3	
SCAN	*Passive*
INP	*#C0 S0*

CALC_3	
SCAN	*Passive*
INPA	*AI_3 PP*

AO_3	
SCAN	*.1 Second*
DOL	*CALC_3 PP*
OMSL	*Closed-loop*

Record Execution

		Ex. A	Ex. B	Ex. C
AI	PACT = 1	1	1	5
	Read	2	2	6
	Conversion	3	3	7
	Alarm	4	4	8
	Monitor	5	5	9
	Forward Link	6	6	10
	PACT = 0	7	22	11
CALC	PACT = 1	8	7	3
	Read Inputs	9	8	4
	Do CALC	10	9	12
	Alarms	11	10	13
	Monitors	12	11	14
	Forward Link	13	12	15
	PACT = 0	14	21	16
AO	PACT = 1	15	13	1
	Read Setpoint (DOL)	16	14	2
	Limits	17	15	17
	Output	18	16	18
	Alarm	19	17	19
	Monitor	20	18	20
	Forward Link	21	19	21
	PACT = 0	22	20	22

Which Record Is NEVER Processed ?

AI_1	
SCAN	*Passive*
PHAS	*0*
INP	*#C0 S0*

CALC_1	
SCAN	*Passive*
INPA	*AI_1 PP*

CALC_2	
SCAN	*Passive*
INPA	*AI_1 PP*

AO_1	
SCAN	*.1 Second*
DOL	*CALC_1 PP*
OMSL	*Closed-loop*

Which Record Is Processed Twice ?

```
AI_1
SCAN   Passive
PHAS   0
INP    #C0 S0
```

```
CALC_1
SCAN   Passive
INPA   AI_1 PP
```

```
CALC_2
SCAN   .1 Second
INPA   AI_1 PP
```

```
AO_1
SCAN   .1 Second
DOL    CALC_1 PP
OMSL   Closed-loop
```

What Happens Here ? Intro. to the PACT Field

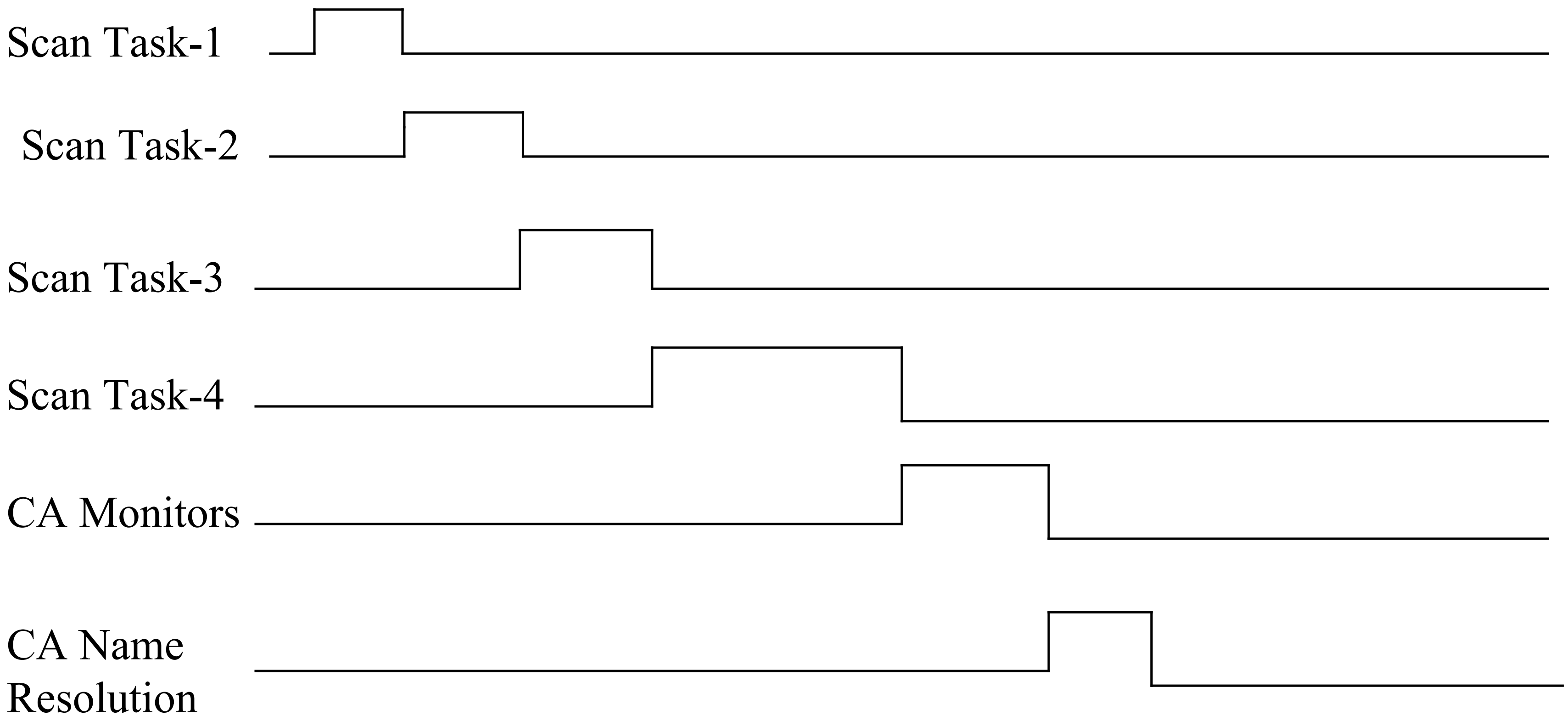
AI_1	
SCAN	*Passive*
INP	*#C0 S0*
FLNK	*CALC_2*

CALC_1	
SCAN	*Passive*
INPA	*AI_1 PP*

CALC_2	
SCAN	*Passive*
INPA	*AI_1 PP*
FLNK	*AO_1*

AO_1	
SCAN	*.1 Second*
DOL	*CALC_1 PP*
OMSL	*Closed-loop*

IOC Timing Chart



Data Conversions

Analog - LINR

- None - make the RAW count the engineering units Value
- Linear - Scale between EGUF and EGUL for the raw value range
- Piece-wise linear - table lookup for line segment - then linearize
- AI – has smoothing field (SMOO) for a wieghte average

Discrete

- Matches bit pattern to string

Multi-channel and Complex conversions

- Subroutine Records
- Calculation records
- Higher Level Records (like the Beam Position Monitor)

Alarm Conditions

Alarm Components

SEVR - Alarm Severity   STAT - Alarm Status

I/O Failure results

SEVR = INVALID   STAT = READ/WRITE

Configurable Analog Alarms

SEVR = NONE/MINOR/MAJOR   STAT = LIMIT

Limits are configured - 2 high and 2 low limits - with hysterisis

Alarm severity for each limit is configured

Configurable Discrete Alarms

SEVR = NONE/MINOR/MAJOR   STAT = STATE or COS

Channel Access Fields

Analog process blocks have three monitor conditions

- Value deadband - MDEL
- Archive deadband - ADEL
- Alarm status change

Discrete process blocks post value on change of state

Monitor queue and monitor count exist for each process block

Access Security - ASG Access Security Group for this record

Control Fields

Desired Output Location (DOL)- where to fetch the value that will be output

- Database Address - (Maximize Alarm Severity recommended)

Output Mode Select (OMSL) - should we use the value from DOL

- Supervisory - use the VAL field
- Closed-loop - fetch the VAL field from the DOL

INVALID Alarm Action (IACT) - what to do if the record SEVR is INVALID

- Hold the current value
- Set to the safe value specified

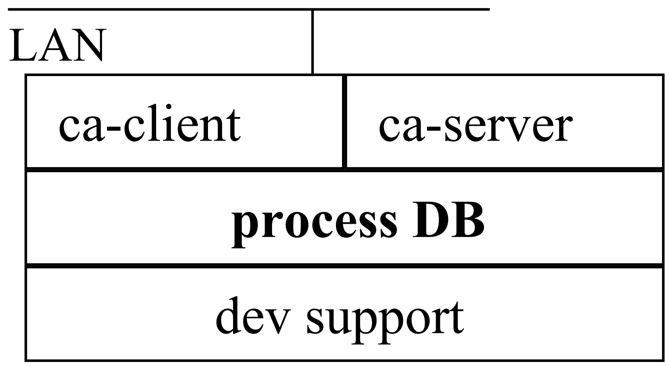
Output Address (OUT) - where to write the value

- Constant - do not write the value
- Database address - write the value to this address and Process if Passive
- Hardware address - call device support to convert and write the value

Algorithmic Specific Fields

CALC record	36 character expression, 16 input links
STEPPERMOTOR	initialization algorithm, retry deadband, retry count, velocity, acceleration, direction, cw limit, ccw limit, moving, done retries, first move error
CPID	proportional, integral and derivative gains, error, mode
SELECT	16 inputs, selection criteria
SUBROUTINE	initialization routine, routine, 16 input links

EPICS Process Database Performance Parameters



Process Block execution time varies from block type to block type

AI on a PowerPC ~ \100,000/second (50% idle)
AI on a 68060 is ~18,000/second (50% idle)
AI on a 68040 is ~6,000/second (50% idle)

Fastest periodic scan rate is dependent on vxWorks clock tick - 60 Hz

Interrupt scanning is limited by the CPU bandwidth (interrupt delay ~33usec)

Name resolution - 10,000/second - runs at the lowest priority

2,500 Process blocks use around 1 Megabyte of memory

New Developments in the Process Database

- Port the database so that it runs on operating systems other than VxWorks. (ANL) – in beta test on LINUX, Windows, and RTEMS.
- Support periodic monitoring capability.
- Develop low cost I/O solutions for the Pentium platform – complete.
- Create a new link mechanism to make checking possible at configuration time for new address types. ANL