Creating IOC Databases

Andrew Johnson (AES/SSG, Argonne)

Includes material from:

Nick DiMonte (AES/CTL, Argonne)

Outline

- Purpose
- Tools available
- VisualDCT
- Creating a Database
- Creating Records
- Property Inspector
- Creating links
- Macros
- Hierarchical Design

How to Create a Database

- Databases are just ASCII text files, what's wrong with just doing this?

```
$ echo 'record(bi, "anj:bi1") {}' > test.db
$ vi test.db
```

Reminder: DBD & DB Files

```
menu(aoOIF) {
  choice(aoOIF_Full,"Full")
  choice(aoOIF_Incremental,"Incremental")
}
recordtype(ao) {
  include "dbCommon.dbd"
  field(VAL,DBF_DOUBLE) {
    prompt("Desired Output")
    promptgroup(GUI_OUTPUT)
    asl(ASL0)
    pp(TRUE)
  }
  …
  field(OIF,DBF_MENU) {
    prompt("Out Full/Incremental")
    promptgroup(GUI_OUTPUT)
    interest(1)
    menu(aoOIF)
  }
  …
}
```

```
record(bo,"$(user):gunOnC") {
    field(DTYP,"Soft Channel")
    field(ZNAM,"Beam Off")
    field(ONAM,"Beam On")
}
record(calc,"$(user):flameM") {
  field(CALC,"A<32?A+1:0")
  field(SCAN,".1 second")
  field(INPA,"$(user):flameM.VAL NPP")
}
record(ao,"$(user):cathodeCurrentC") {
    field(DESC,"set cathode current")
    field(DTYP,"Raw Soft Channel")
    field(SCAN,"1 second")
    field(OIF,"Full")
    field(PREC,"2")
    field(EGU,"Amps")
    field(DRVH,"20")
    field(DRVL,"0")
    field(HOPR,"20")
    field(LOPR,"0")
}
```

The highlighted parts of the DB file must match entries in the IOC's DBD file

Problems with using a Text Editor

- Content must follow complicated rules
- User must know or look up which fields are available in each record type and the correct choice strings for each menu field
  - Each IOC may have a different set of record types and menu choices
  - Fixing typo's is not efficient use of engineers time
- A text editor doesn't let you see more than a few records at once
  - It can't provide an overview of the database
  - Doesn't show connections between records (links)

Database Configuration Tools

- Database Configuration Tools (DCTs) help avoid mistakes
  - Show what design fields a record type provides
  - Present a list of valid choices for field values
  - Enforce correctness of syntax and content
  - Hide details (fields) that don't matter
- Newer ones provide a graphical view of the database
  - assert(worth(picture) == worth(words << 10));

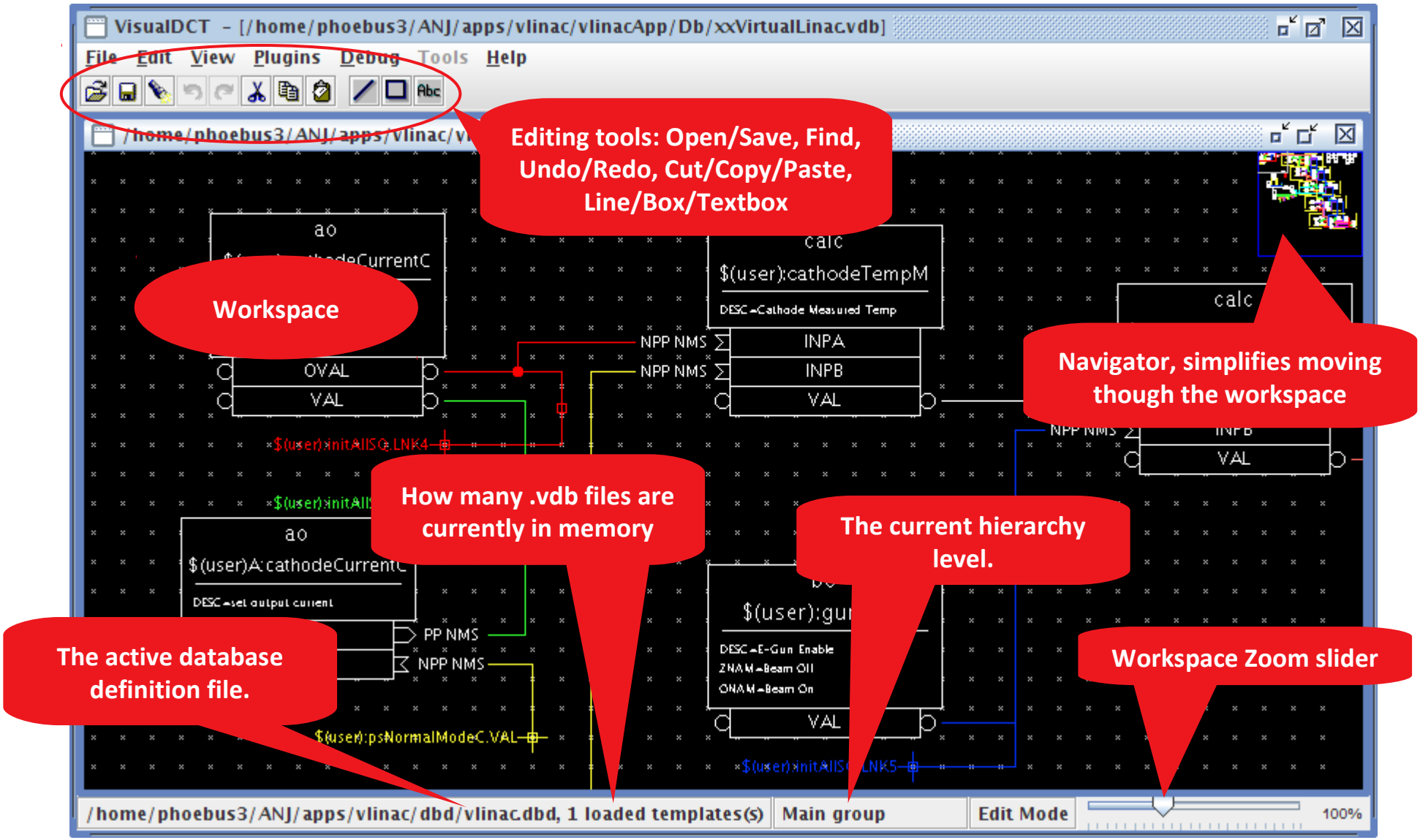
History of EPICS DCTs

- DCT — Database Configuration Tool
  - Text-mode (curses) interface, provided with very old versions of EPICS Base
  - Solves the record types/fields/choices/file syntax problems
  - No graphical view of the database
- DCT in Tcl/Tk, JCDT — Java Database Configuration Tool
  - GUI Forms views of individual records, like the original DCT
  - Still no graphical view of the database
- GDCT — Graphical Database Configuration Tool
  - Flat graphical view of database records and field settings
  - No hierarchy, so not suitable for large databases
- Cap*Fast*
  - Electronic circuit design tool from Phase 3 Logic
    - Hierarchical editor, use wires for database links
  - Program from LANL converted schematic diagrams into database files
  - Not database-aware, editor doesn't offer or check choices for menu fields

Current EPICS DCTs

- VisualDCT — Visual Database Configuration Tool
  - Written under contract by CosyLab (GPLv3)
  - Hierarchical design, graphical layout stored in .db file comments
  - Database-aware, reads DBD files
  - Creates database files directly
  - Can import non-graphical databases
- TDCT — TRIUMF Database Configuration Tool
  - Can read & write Cap*Fast* schematic and symbol files
  - Hierarchical design, graphical layout stored in separate .sch and .sym files
  - Database-aware, reads DBD files
  - Creates database files directly
  - Can import VDCT databases

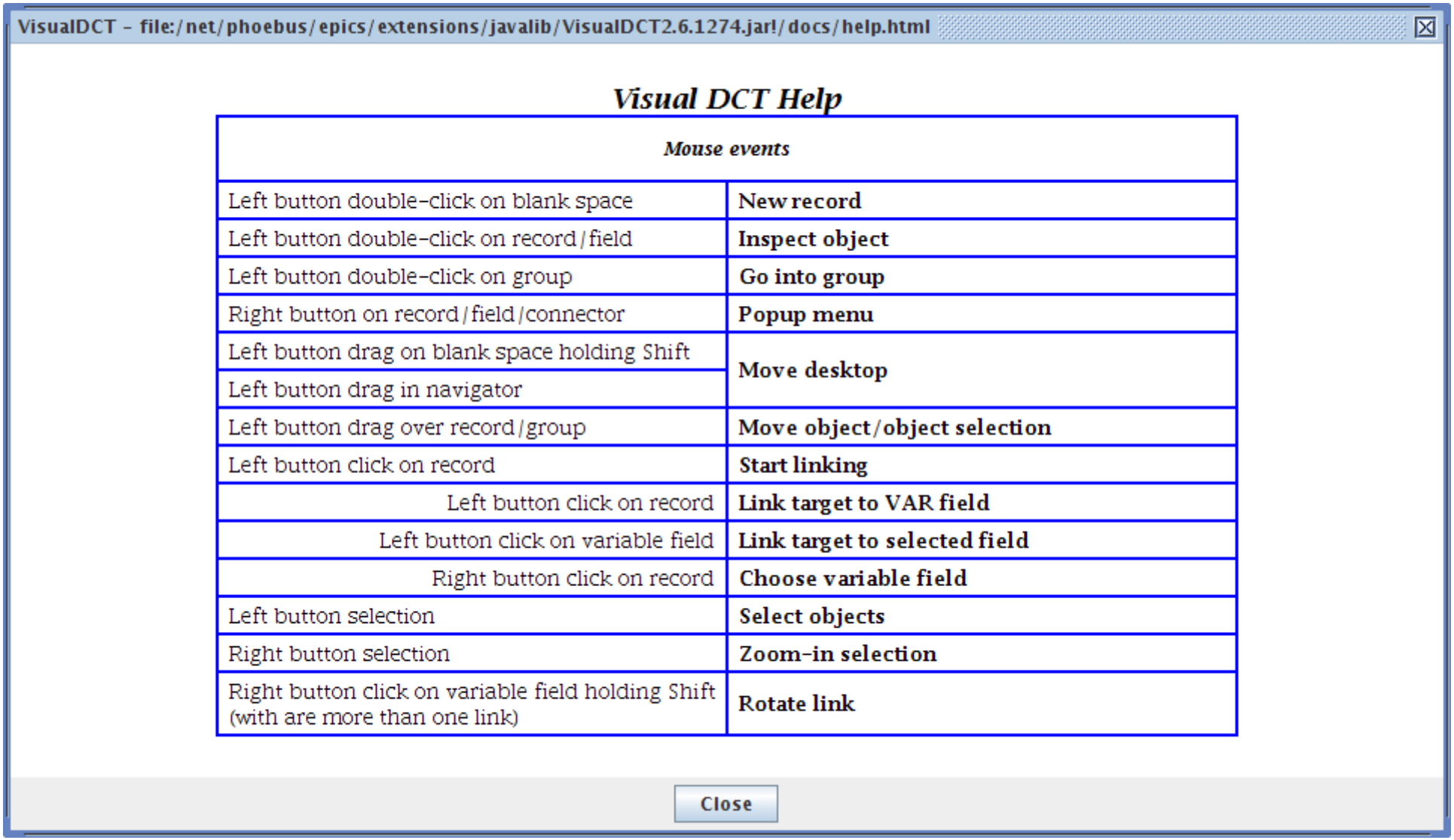
VisualDCT Main Window



Using VisualDCT

- Type **`vdct`** to launch it
  - Actually a script that sets the Java Classpath and runs program
- Optional parameters for paths to a .dbd file and/or a .vdb file
  - If not given you'll be asked to select a .dbd file to use
  - VisualDCT's .vdb files usually contain the path to their .dbd file
- Editing diagrams uses a combination of left & right mouse-button clicks, drags and double-clicks, possibly with Shift pressed, and does different things depending on what you're pointing at and whether it's currently selected or not
  - Help tells you what mouse events it recognizes
- A .vdb file that doesn't use hierarchy features can be loaded directly by an IOC
  - Use 'File :: Generate' to create a flat .db file from a hierarchical design

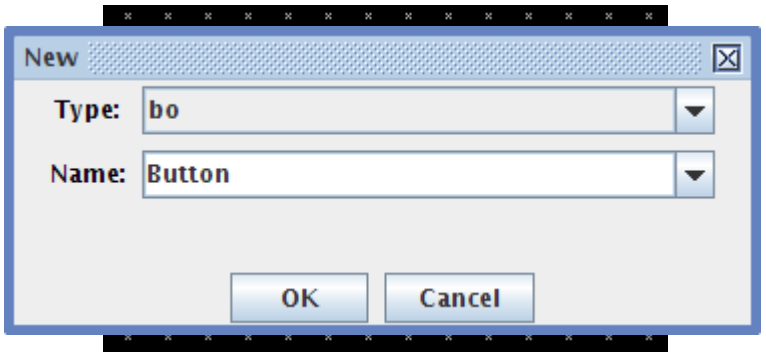
Help :: Help Topics



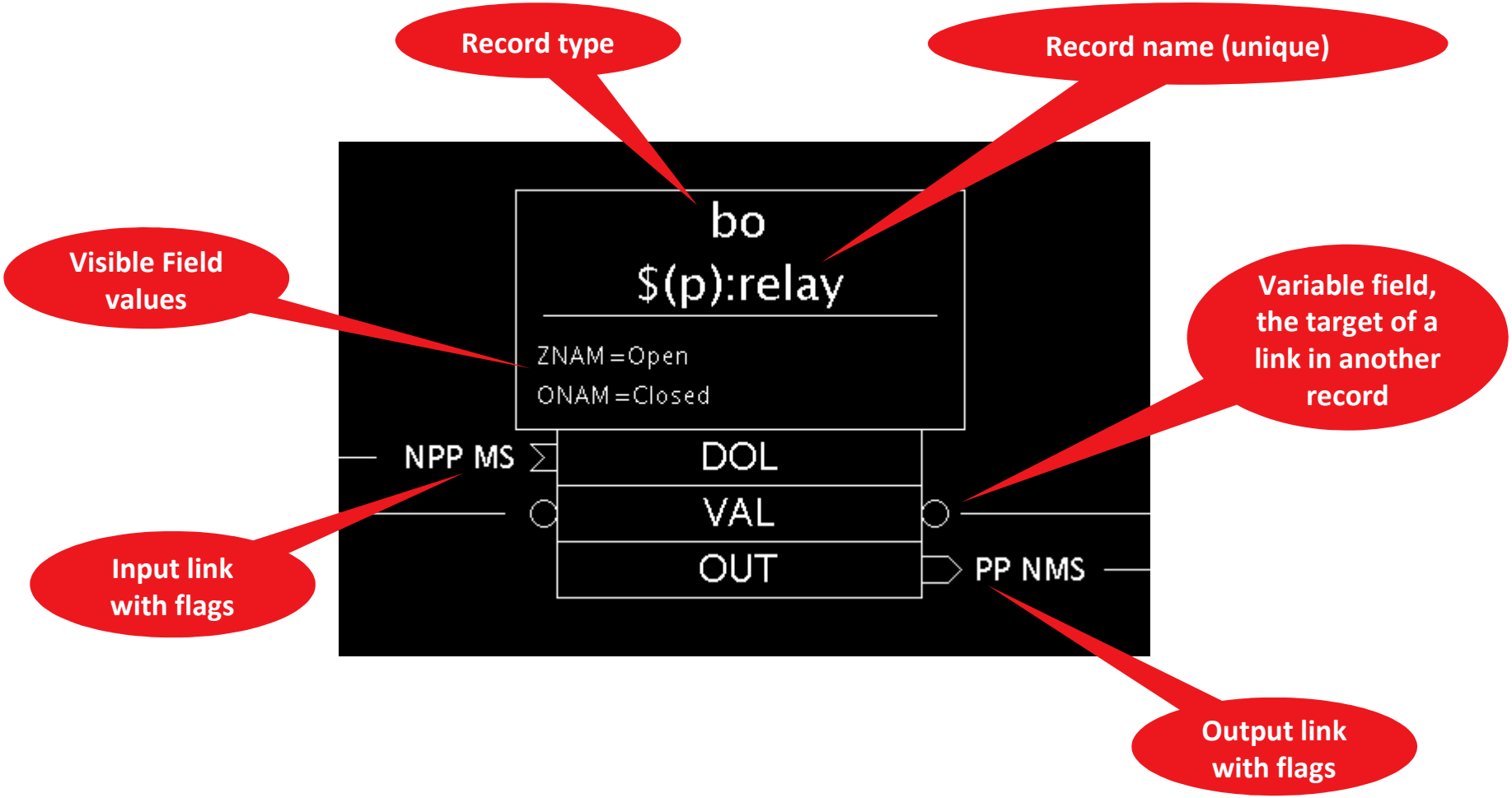
Creating a Database

- If you don't have an empty work-space, use 'File :: New'
- The file-name is set when you use 'File :: Save' or 'File :: Save as...'
  - Use the extension .vdb for VisualDCT databases

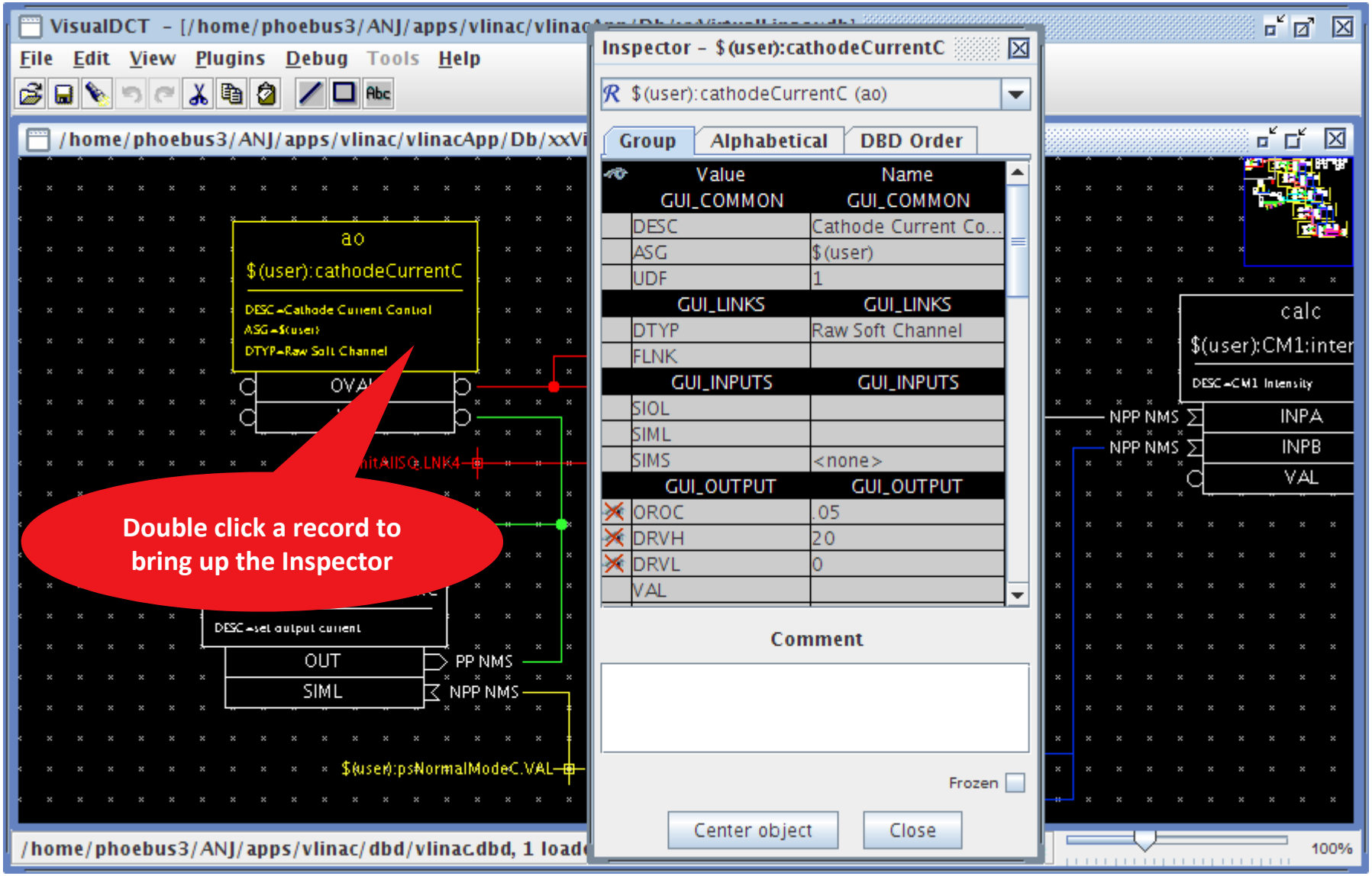
- Double-click in an empty area of the work-space to create a new record
  - Select the record type to create
  - Type in a record name
  - Press Return or click OK



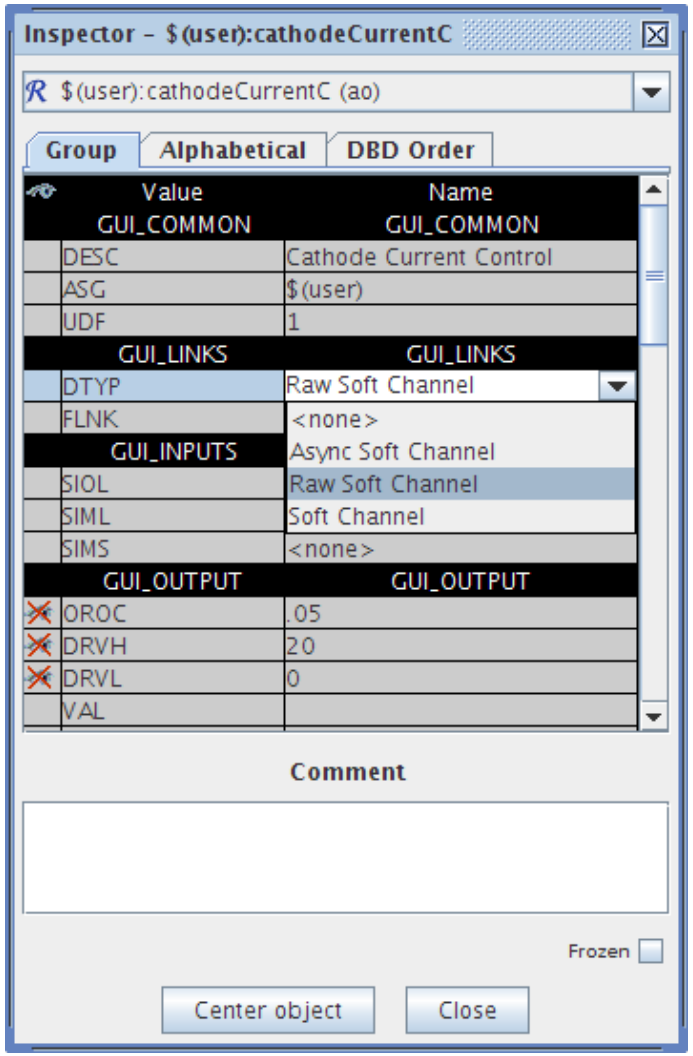
Record Symbol



The Inspector: Examining fields



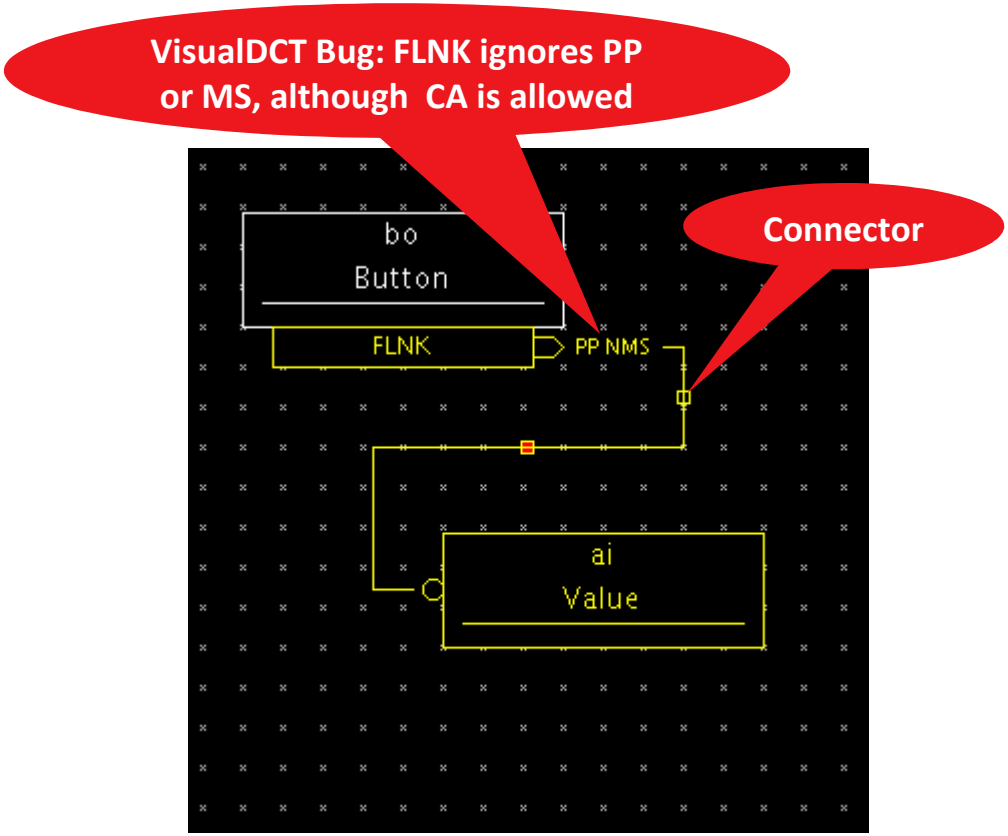
Editing field values



- Tabs alter the field order shown
- Click in the right column of a row to edit that field value
  - Menu fields will offer a list of valid choices for selection
- The left column controls visibility of fields in the work-space
  - Normally only fields with non-default values are shown
  - Click to change: Show always, Never
- Switch to a different record by clicking the record name pull-down, or just double-click on it in the work-space

Setting link fields

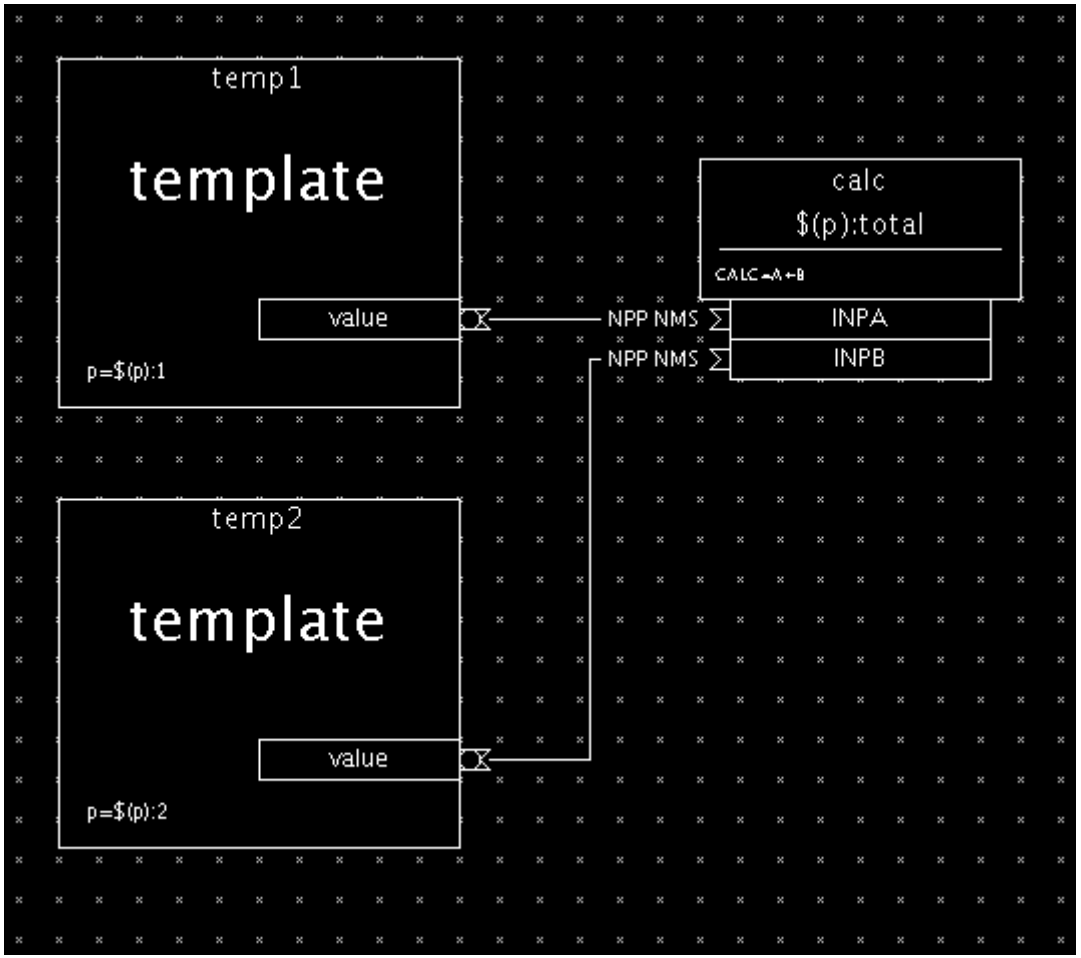
- Right-clicking on a record brings up a menu with various options
- Select the link field you want to set under INLINK, OUTLINK or FWDLINK
  - The record starts flashing red
- Now find and click on the target record
  - Left-click selects the default (VAL) field
  - Right-click lets you select any other field (pick the VARIABLE menu item, then use MORE as needed to reach the field)
- The link appears with a line to its target
  - Set PP/MS by right-clicking on the link
- Use the ‘Add Connector’ link menu item as needed to move the line around; this link needed 2 connectors

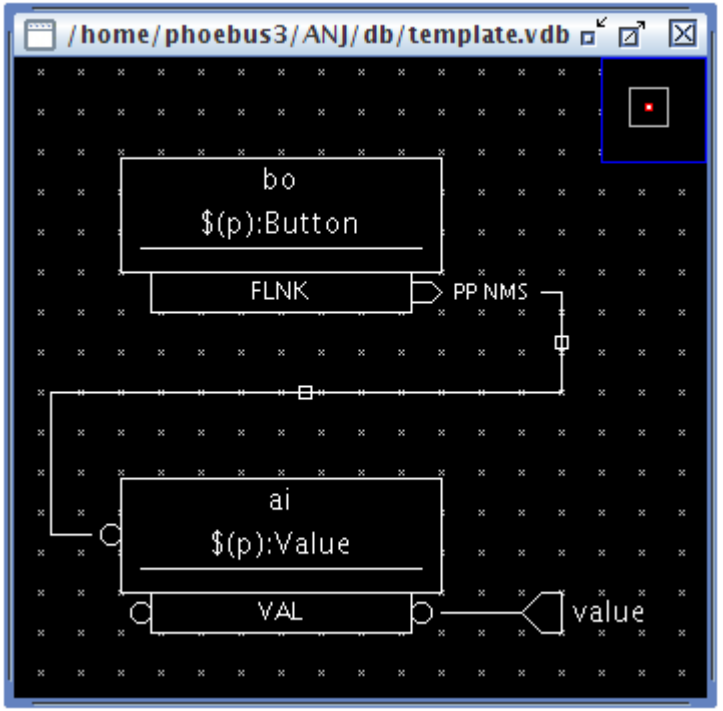


Using Macros

- Some screen-shots on previous slides used macros in the PV name
  - **`$(user):cathodeCurrentC`**
- Macros can also be used in field values
  - Even in menu fields, just type into the box (expect vdct warnings though)
- Macros must be given values before/as the IOC loads the database
  - `$ msi -olinac-anj.db -Muser=anj linac.db`
  - `epics> dbLoadRecords linac.db user=anj`
  - `$ msi -olinac-anj.db -Slinac.substitutions`
  - `epics> dbLoadTemplates linac.substitutions`
- Macros are also used in hierarchical database designs
  - A database file that instantiates lower-level templates can provide values for macros used in those templates
  - A macro with a . in the name like **`$(a.b)`** is a reference to port **`b`** of an instance of the lower-level template named **`a`**

Hierarchical Example



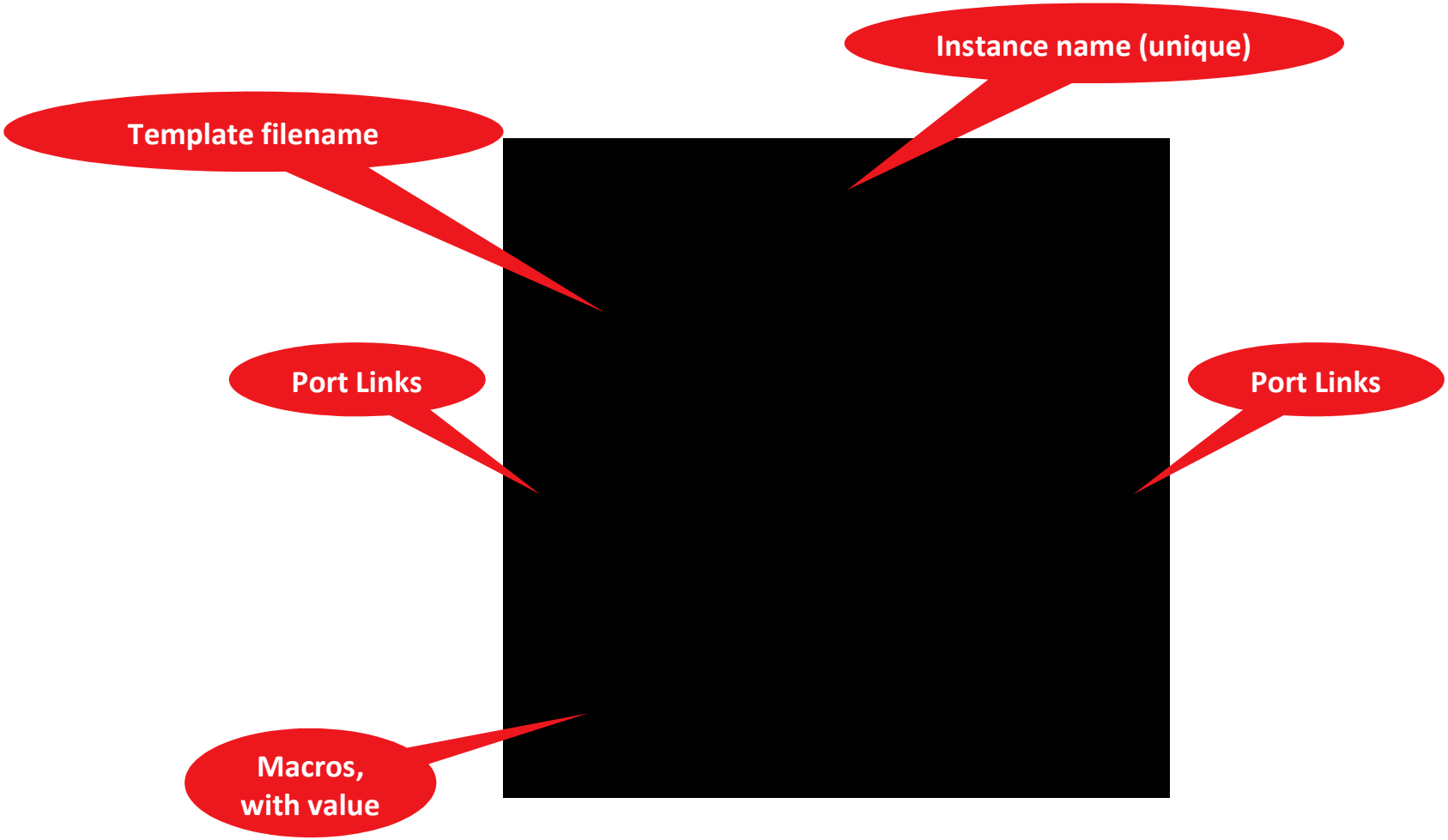


The template shown above has been instantiated twice, with different values for the macro `p`

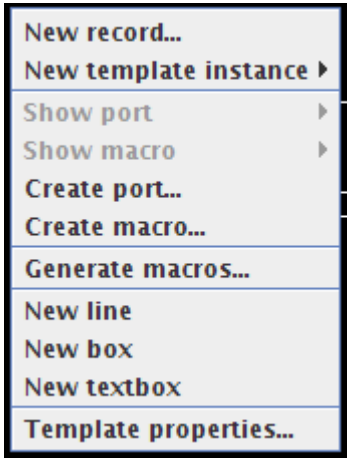
Generated Database File

```
#! Generated by VisualDCT v2.6

# expand("/home/phoebus3/ANJ/db/template.vdb", temp1)
record(bo, "$(p):1:Button") {
  field(FLNK, "$(p):1:Value")
}

record(ai, "$(p):1:Value") {
}
# end(temp1)

# expand("/home/phoebus3/ANJ/db/template.vdb", temp2)
record(bo, "$(p):2:Button") {
  field(FLNK, "$(p):2:Value")
}

record(ai, "$(p):2:Value") {
}
# end(temp2)

record(calc, "$(p):total") {
  field(CALC, "A+B")
  field(INPA, "$(p):1:Value")
  field(INPB, "$(p):2:Value")
}
```

Template Symbol



Working with Templates



- Create the template database and save it (important!)
- After saving, the work-space right-click menu shows new choices
  - Use ‘Create port...’ to add any ports to the template
  - Attach a port to any field that a record outside the template must link to
  - Use ‘Create macro...’ to add macro links
  - When you instantiate the template, port and macro links appear in the template symbol
- To instantiate template, first load it using ‘File :: Import template...’
  - The imported template is now available on the work-space right-click menu under ‘New template instance’
  - Double-click on the template box to bring up its inspector
    - To add new macros to the instance, right-click on the words ‘Add macro...’
  - When you re-save a modified template file, the parent diagram updates automatically
  - Templates may contain instances of other templates
- Use ‘File :: Generate...’ to convert the top-level diagram to a flat-file .db file

More Information

- EPICS Website :: Extensions web-page
  - VisualDCT Wiki page is linked under ‘IOC Database Management Tools’
  - Links to downloads and documentation under "Where to find it"
- User’s Manual covers most things
  - Templates and hierarchy’s have a short separate manual
- Worth considering TDCT as well, although not as widely used
  - Lets you design your own template and record-type symbols