

# VisualDCT

---

## User's Manual

Revision: 1.1

---

Status: Draft

---

Repository: acc/products

---

Project: ACC-EPIC-SBIR08

---

Folder: VDCT4/documentation

---

File: MAN-VDCT4.doc

---

Owner: Matej Sekoranja

---

Last modification: October 15, 2009

---

Created: October 12, 2009

---

## Document History

Revision	Date	Changed/ reviewed	Section	Modification
1.0	2009-10-01	Janez Golob	All	Created.
1.1	2009-10-12	Jernej Kamenik	All	Updated.

## Confidentiality

This document is classified as a **public document**. As such, it or parts thereof are openly accessible to anyone listed in the Audience section, either in electronic or in any other form.

## Scope

This document is a users manual oh how to use VisualDCT v3 for EPICS v4 (pvdata) and IRMIS databases. It also contains some tips and tricks.

## Audience

The audience of this document are all users of VicualDCT v3.

## Typography

This document uses the following styles:

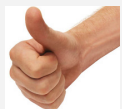


A box like this contains important information.



### Warning!

A box like this provides information, which should not be disregarded!



A box like this contains a feature, it should be pointed out.

## Glossary of Terms

VDCT ..... Visual configuration tool for EPICS. This document refers to the product version 3, developed for use with EPICSv4 (pvdata) and IRMIS databases.

---

EPICS .....	Experimental Physics and Industrial Control System. When not explicitly stated, this document refers to EPICS v3.x under this name. For EPICS v4 see PVDATA
PVDATA .....	Process Variable Data. Also referred to as EPICS v4.
IRMIS .....	Integrated Relational Model of Installed Systems. A relational database containing information about the operational EPICS IOCs installed at a particular site.
CAPFAST .....	A schematic design tool.

---

## References

---

- [1] EPICS homepage, <http://www.aps.anl.gov/epics/>
- [2] VDCT homepage, <http://visualdct.cosylab.com/>
- [3] IRMIS homepage, <http://www.aps.anl.gov/epics/irmis/>
- [4] Capfast homepage, <http://www.aps.anl.gov/epics/extensions/capfast/>

## Table of Contents

---

<b>1. INTRODUCTION .....</b>	<b>5</b>
<b>2. Basic Principles .....</b>	<b>6</b>
2.1. Running VDCT .....	6
2.2. User Interface .....	8
2.2.1. Visual tab .....	9
2.2.2. Spreadsheet view .....	10
2.2.3. Hierarchy view .....	11
2.2.4. Menu Command Reference .....	12
2.3. Drawing Graphics Primitives .....	13
<b>3. EPICS v3 (DB) with IRMIS Support .....</b>	<b>14</b>
3.1. Visual representation of objects .....	14
3.2. Morphing .....	15
3.3. Linking .....	15
3.4. IRMIS .....	16
3.5. The Module Menu .....	16
<b>4. PVDATA .....</b>	<b>17</b>
4.1. Hierarchies .....	17
4.2. Record and Structure Visualization .....	18
4.3. Database construction and linking .....	18
4.4. The Module Menu .....	19

---

## Figures

---

Figure 1: Main VisualDCT Window.....	8
Figure 2: Visual tab.....	9
Figure 3: Spreadsheet tab .....	10
Figure 4: Tree view .....	11
Figure 5: Graphical primitives.....	13
Figure 6: Input dialog for specifying remote IRMIS database location.....	16
Figure 7: pvdata example database in hierarchy view .....	17
Figure 8: pvdata example database in visual view .....	18

---

## Tables

---

**Error! No table of figures entries found.**

## 1. INTRODUCTION

---

The Visual Database Configuration Tool version 3 (VisualDCT) is a visual database creating and editing tool for EPICS v4 (pvdata) and EPICS v3 with capability to interact with IRMIS RDB that contains information about the operational EPICS IOCs installed on a particular site. It is written in Java and therefore supported in various systems.

This completely new tool has been developed to meet new needs mandated by new EPICS v4 specification, in particular:

- Support for arbitrary hierarchies in both templates and primitive types

The IRMIS Db development embeds EPICS databases into the more general context of the various control system views

- The information on encapsulation, cabling, housing is contained in a single descriptive database
- Enables linking of various control system aspects of the same component

The goal of the new VDCT development had therefore two folds:

- To build a visual database configuration tool with full EPICS v 4 compliance
- To allow for/support interfacing with, visualization and manipulation of other related IRMIS control system views (cabling, etc..)
- At the same time the tool should retain if not enhance the existing functionality of VDCT for EPICS v3. based on feedback from the community.

This manual describes the VisualDCT version 3.0-beta build 20091012. For previous versions, refer to [2].

## 2. BASIC PRINCIPLES

VisualDCT allows to visually render and manipulate data models, which can be mapped to a hierarchical graph structure. The different model support implementations are loaded into the application as Modules. The modules to be loaded are specified through the configuration and can thus be provided as 3<sup>rd</sup> party libraries. The loaded modules can be interchanged at runtime. Present VisualDCT distribution comes with two model implementations:

- *EPICS v3 Module* allows to create and manipulate existing EPICS v3 database instance files (.db). This Module comes interfaced with the IRMIS database support, so EPICS database files can be loaded from remote IRMIS locations. The default rendering engine of EPICS models is now using the standard Capfast EPICS record symbols.
- *EPICS v4 (pvdata) Module* allows to create new and manipulate existing pvdata xml database definition files. Completely generic hierarchical pvdata structures are supported. Since structures in pvdata are not defined as Capfast symbols and since they are typically also much more compact (contain less member fields) compared to records EPICS v3, the default rendering engine for pvdata uses a different visual structure representation.

VisualDCT follows common editing principles for all model implementations. In the following sections therefore, separate features will be described for the following aspects:

- *Engineer drawing style* for designing new and editing existing control system designs. Records can be moved, edited, linked on the graphical canvas.
- *Spreadsheet editing* represents the non-visual counterpart of the previous item. It supports the same model designs, modulo visual information. It is useful when manipulating large databases.
- *Hierarchical Tree view* of the model structure allows for seamless navigation of complex hierarchical models.
- *Persistence layer* is responsible for converting the edited and created model designs into (and from) well defined formats (EPICS v3,4, IRMIS) and storing (reading) them into (from) local files or remote databases.

### 2.1. RUNNING VDCT

In order to run VisualDCT, Java Runtime Environment 1.6 is required. VisualDCT is distributed as a Java Archive package (.jar file), so there is only one file in the binary distribution. Convenience execution scripts are provided for UNIX-like operating systems and Microsoft Windows. This file has to be added to the java classpath variable (search path for application classes and resources) to help JVM find com.cosylab.vdct.VisualDCT class, which is the main class of the VisualDCT.

Usage of VisualDCT:

```
java -cp VisualDCT.jar com.cosylab.vdct.VisualDCT
```

Listing 1: Basic run command.

VisualDCT Java ARchive package (.jar file) is so called executable JAR file, which means it can be run as:

```
java -jar VisualDCT.jar
```

Listing 2: Running executable JAR.

If your GUI has this feature, a double-click on the VisualDCT.jar file icon will also do it. VisualDCT accepts two parameters which are not obligatory: configuration folder and model file.

An example of running VisualDCT, using ~/db configuration folder and test.db model file:

```
java -jar VisualDCT.jar -config=~/db test.db
```

Listing 3: An example of running VisualDCT.

Two predefined configuration folders are supplied with the VisualDCT distribution:

- config/db/ - loads EPICS v3 module and its default configuration. Using this configuration, database instance (.db) files can be opened directly at VisualDCT startup. Preloaded database definition files (.dbd) and other relevant properties can be specified in the EPICS module configuration file.
- config/pvdata/ - loads EPICS v4 (pvdata) module and its default configuration. Using this option, database definition xml files of pvdata can be opened directly at VisualDCT startup. Preloaded structure definition files and other relevant properties can be specified in the pvdata module configuration file.



Always make sure to load a compatible configuration when opening model definition files at VisualDCT startup via command line arguments. Trying to load an incompatible model with a specified configuration will cause the application to fail in unpredictable ways.

## 2.2. USER INTERFACE

As every powerful IDE also VisualDCT provides indispensable facilities as clipboard and undo support. A great effort was given to synchronization between the record instance database and its visualization. Every change done visually is immediately reflected in the database and vice versa; all actions like moving, renaming and deletion of records which affect links are automatically fixed by the VisualDCT.

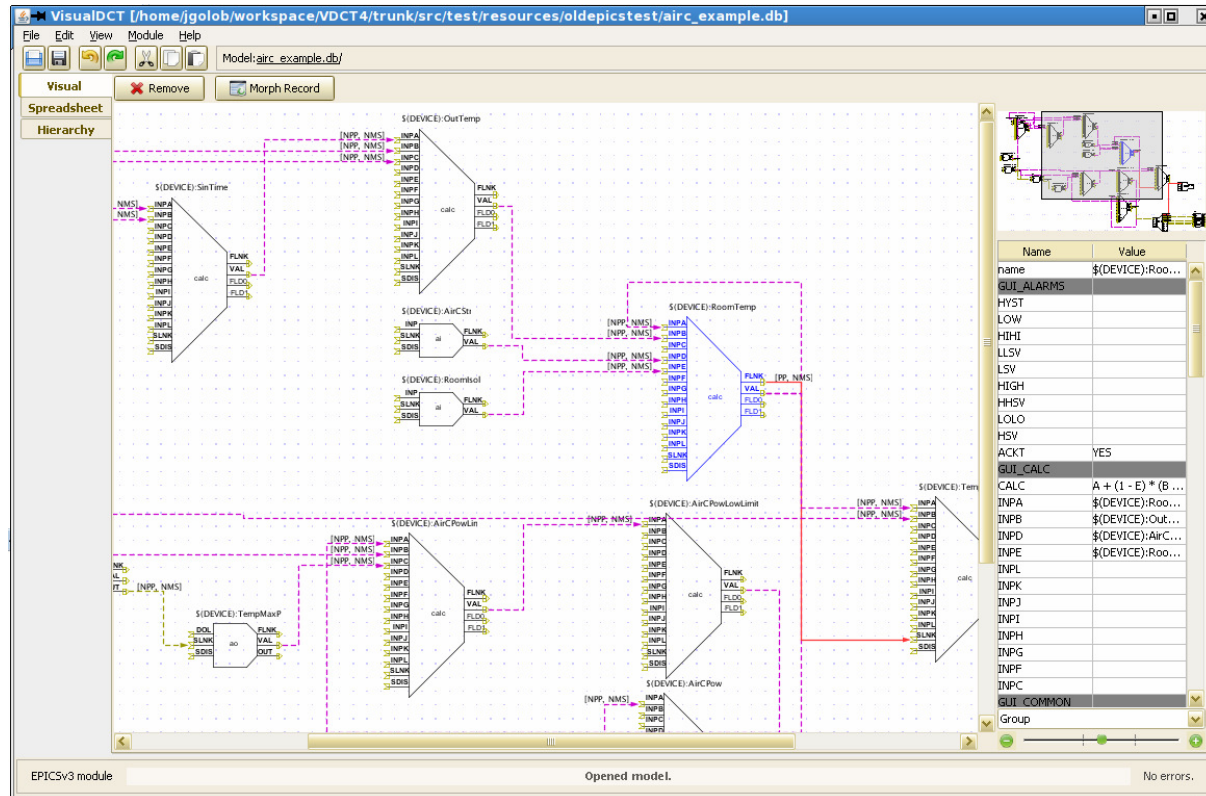


Figure 1: Main VisualDCT Window

The main window of VisualDCT consists of :

1. *Main menu*
2. *Toolbar with address field* - makes access to the frequently used actions easier and also displays the current location (address) of view within a hierarchical model. It is possible to quickly transverse hierarchical models by clicking on any displayed hierarchy level name in the address field.
3. *Workspace tabs* - are the main components of the VisualDCT. They provide different visualizations and editing capabilities of supported models. The workspace contains three tabs which are described in more detail below: *Visual*, *Spreadsheet* and *Hierarchy* tab.
4. *Status bar* – displays basic information and status of currently loaded Module and Model as well as progress of long-lasting (usually persistence) tasks. Debug and log consoles can be inspected by double clicking on the status label and progress bar respectively.



### 2.2.1. Visual tab

The Visual tab shows visual representation of a model (a single level of a hierarchical model) on a graphical canvas. It consists of nodes (records in EPICS v3, records or structures in EPICS v4), pins (fields in EPICS v3, scalars or scalar arrays in EPICS v4) and links between them. In addition primitive graphical objects (lines, rectangles, ovals, text fields) can be drawn on the canvas. Objects can be selected by clicking on them with the left mouse button. The nodes, primitives and link control points (marked with small rectangles on link junctions) can be moved around the canvas via mouse drag and drop.

The Visual tab also contains an **action bar** which contains all actions that can be done on selected object(s). The same actions are also available through a pop-up menu, which can be brought up over a selected object by clicking on it with the right mouse button. Primitives, nodes and pins can be added to the model using these actions. On the other hand, links are created by left-clicking on a pin and dragging the link into another pin. In this way links are always defined between source and target pins so it is not possible to draw a link with undefined source or target.

The selected object's properties can be edited through the **property editor** table on the right side of the canvas. Name-value pair table allows the user to inspect or modify fields. Rows can be grouped according to model specification or displayed alphabetically. This is controlled with the combo-box below the table.

In the top-right corner of the visual tab there is the **satellite view**, where the complete canvas is displayed and the present view is marked with a rectangle. It is possible to move the present view by clicking inside the satellite view.

In the bottom right corner of the visual tab, there is the **zoom slider**. The view can be zoomed in or out using the slider controls (the + and – buttons and the slider knob).

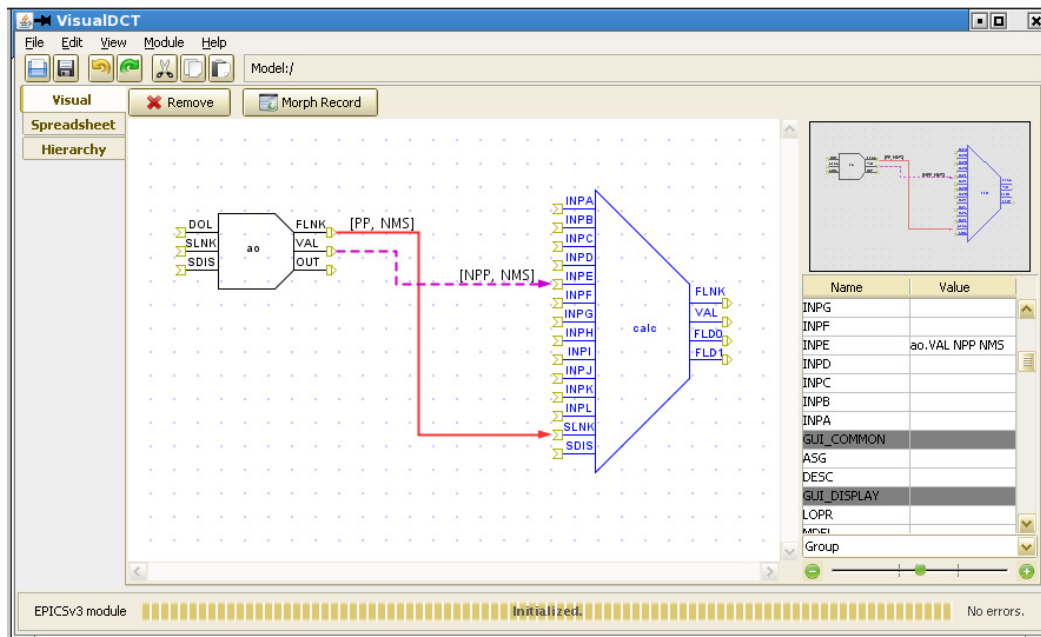


Figure 2: Visual tab

### 2.2.2. Spreadsheet view

The Spreadsheet view represents the non-visual counterpart of the Visual view. It supports the same functionality as Visual view (without support for drawing graphical primitives). Nodes are grouped into separate tabs depending on their type, as determined by the model implementation (e.g. record types in EPICS v3). Each tab contains a table of node rows and an appropriate action bar.

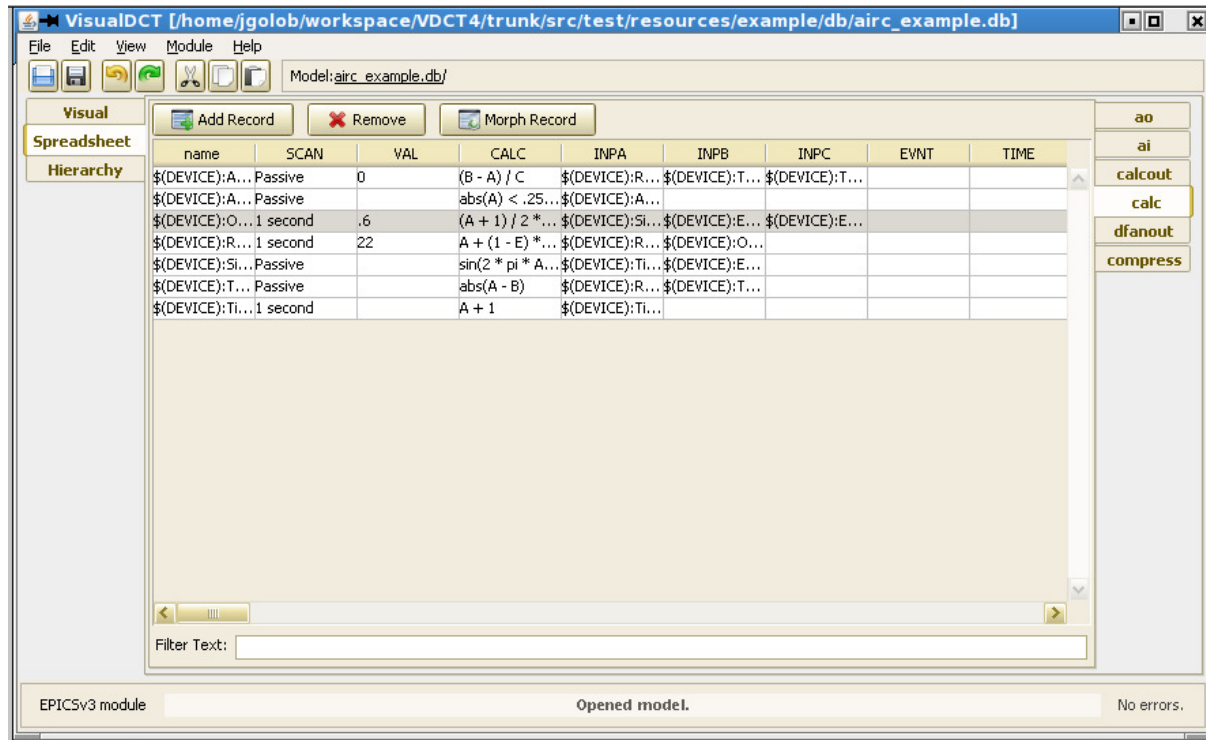


Figure 3: Spreadsheet tab

Again the **action bar** contains all actions that can be done on the selected table or row. Alternatively, the same actions are available through a mouse pop-up menu.

Each table row contains one node (record in EPICS v3, record or structure in EPICS v4). The displayed table rows can be filtered using the **filter text field** at the bottom of the table. To apply a filter, enter the chosen matching string into the field and press the **Enter** key.



Hint: You can add multiple clone nodes (same type and numbered name) easily by copying an existing node and pasting it multiple times. This functionality replaces the **Add row** action in the VisualDCT 2.x Spreadsheet view and now also works in the Visual view.

### 2.2.3. Hierarchy view

In Visual view the only way to navigate the structured data is via the **Explore** action. This is not a very convenient way to access hierarchically structured data. Therefore, the hierarchy view is intended to be used for fast navigation through hierarchies. The Hierarchy view is especially useful with EPICS v4 (pvdata) due to its inherently hierarchical structure. It represents model nodes (records in EPICS v3, records or structures in EPICS v4) in a tree view which can thus be easily transversed by simply clicking on tree nodes with the left mouse button. Selection of a node also updates the Visual and the Spreadsheet views to the selected level, which is also displayed in the address field on top of the VisualDCT GUI. The hierarchy view also contains an **action bar** on top containing all available actions allowing for model manipulation as well as a **property editor** table on the right, where node properties can be edited as in the Visual view.

The following tree node states are possible:

- – Node is not focused nor explored
- – Node is focused and its properties can be edited in the Property editor table
- – Node is explored and its contents are shown in the Visual and Spreadsheet views

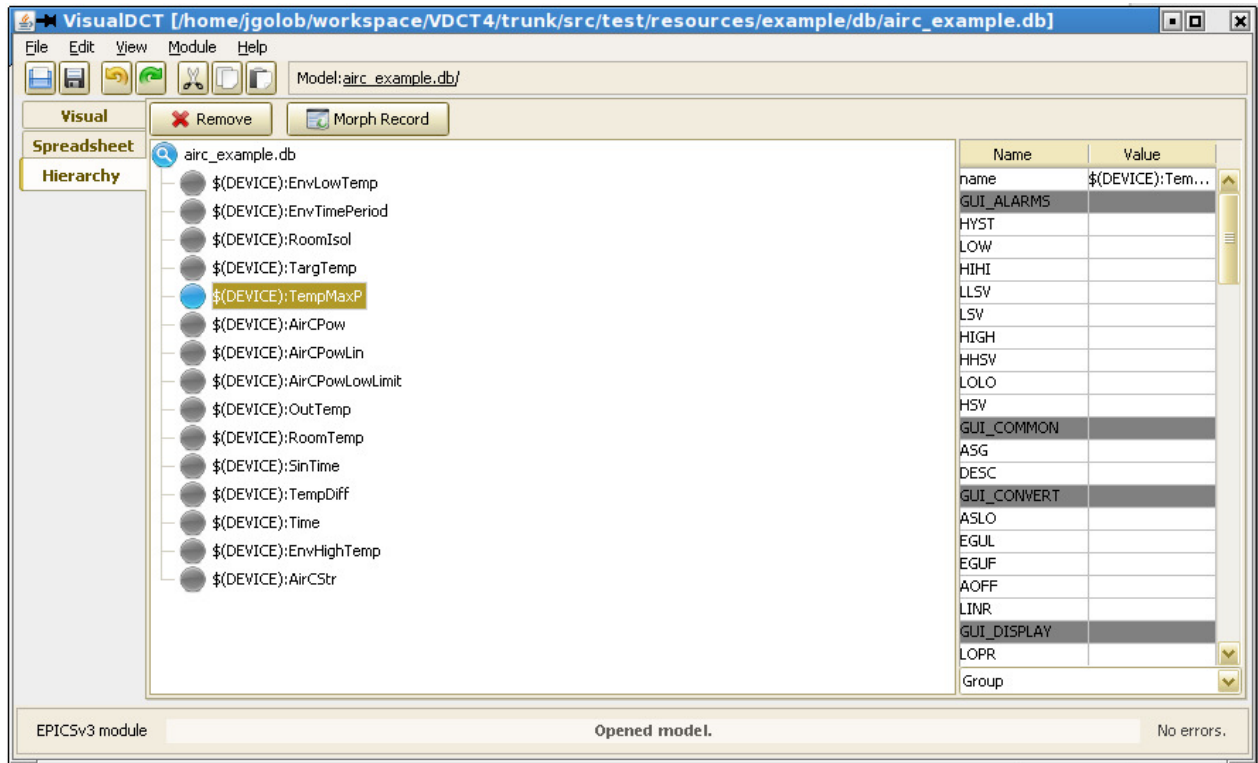


Figure 4: Tree view

### 2.2.4. Menu Command Reference

This section describes menu commands available by the VisualDCT. Convenient keyboard shortcuts are given in parentheses.

#### File Menu:

- *New Model* (Ctrl-N) - close the currently active model, and allow the user to create a new model.
- *Open* (Ctrl-O) - close the currently active model, and provide a file selection window which will allow the user to open a new existing model. The model will be checked for consistency.
- *Import Model* - provides a file selection window which will allow the user to specify a new existing model which will be added (appended) to the existing active models.
- *Save Model* (Ctrl-S) - save the currently active model.
- *Save Model As* - save the currently active model, and allow the user to specify a name of the file into which the model will be saved.
- *Print* (Ctrl-P) - print the canvas.
- *Exit* (Ctrl-Q) - exit the VisualDCT.

#### Edit Menu:

- *Undo* (Ctrl-Z)- undo the last action.
- *Redo* (Ctrl-Y)- redo the previously undone action.
- *Remove* – remove the selection.
- *Copy* (Ctrl-C)- copy the selection and put it on the clipboard.
- *Cut* (Ctrl-X)- cut the selection and put it on the clipboard.
- *Paste* (Ctrl-V)- insert the clipboard contents to the workspace.
- *Select All* (Ctrl-A) - select all objects on the canvas.
- *Show Grid* - toggle grid visibility on the workspace.
- *Snap to Grid* - snap objects to the predefined grid.

#### View Menu:

- *Debug Log* – display the debug log dialog.
- *Progress Details* – display a dialog showing the history of executed actions in the VisualDCT session.
- *Zoom In* (Ctrl=) - increase zoom scale by 10%.
- *Zoom Out* (Ctrl-) - decrease zoom scale by 10%.
- *Zoom Selection* - zoom the selection to fit the screen.
- *Zoom All* – zoom all to fit the screen.
- *Compactify Canvas* – pack the objects on the canvas so that no blank space is left around them
- *Configure Canvas* – display the canvas configuration dialog.
- *Toggle Bird View* (Ctrl-B) – toggle a magnifying glass rectangle over the cursor.

**Module menu:**

- *Switch module* - switch between modules.

Other model dependent actions may be put into the Module menu by the currently loaded module. The actions for EPICS v3 and pvdata modules are described in subsequent sections.

**Help Menu:**

- *Commands Quick Guide* - list of available mouse commands and keyboard shortcuts.
- *About* - display program information, version number and copyright.

## 2.3. DRAWING GRAPHICS PRIMITIVES

VisualDCT currently supports the following graphical primitives:

- *lines* (support dashed and arrows)
- *ovals*
- *rectangles*
- *multi-line labels* (support automatic wrapping and html)

All primitives can be added on the canvas in the Visual view trough the action bar or the popup menu. They can be moved around by mouse dragging, and resized though their control points. Their properties can be edited in the property editor.

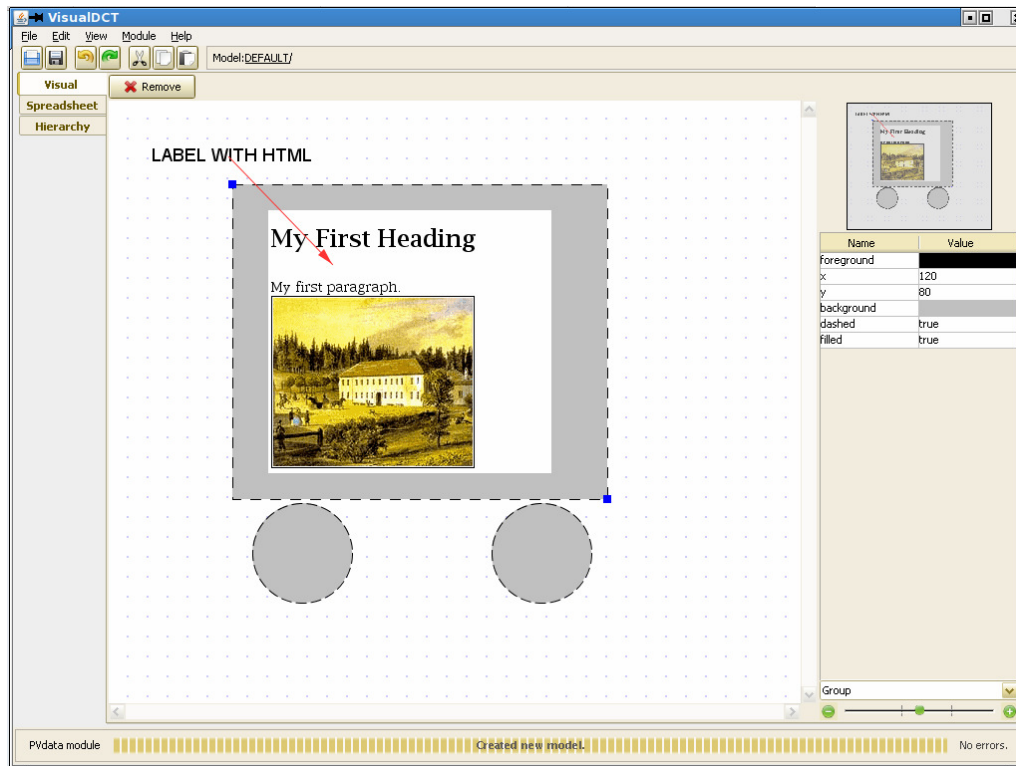


Figure 5: Graphical primitives

### 3. EPICS v3 (DB) WITH IRMIS SUPPORT

VisualDCT is designed to create and maintain EPICS record instance database (.db) files. In order for VisualDCT to execute properly, database definition (.dbd) files have to be provided which contain the specifications for the various record and device types that they intend to reference in any record instance database (.db) file to be created by VisualDCT. Once database definition (.dbd) file/s have been specified, records can be created, copied, renamed, etc. using the various facilities provided by the VisualDCT. As the user interacts with the various VisualDCT windows, selections, and data entry fields, the results of these interactions are displayed on the screen. Revisions and data entry updates of record instance data displayed on the screen do not replace previously stored record instance data until the user saves currently modified record instance database (.db) file. As VisualDCT executes, it attempts to trap and display the most common situations that might lead to diminishing the integrity of the user supplied information.



Locations of the loaded database definition files are specified through the Module → Configure DBDs action dialog. The settings are saved to the EPICS module configuration file (saved.properties.com.cosylab.vdct.model.db.DBModule) in the loaded configuration folder as:

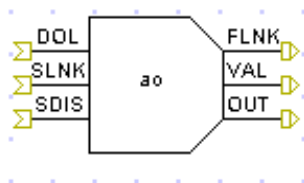
```
dbdFiles=./example/db/airc_example.dbd
```

VisualDCT supports loading database definitions and record instance databases from local file system as well as from a remote IRMIS database location.

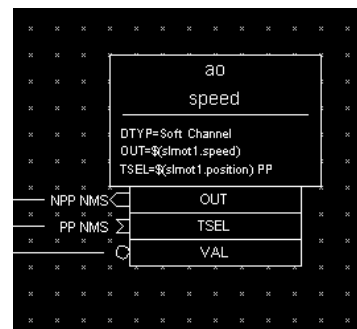
The major changes with respect to VisualDCT v2.x functionality are explained below.

#### 3.1. VISUAL REPRESENTATION OF OBJECTS

Record representation is drawn based on CapFast configuration files. Each record type should have at least one CapFast symbol file. In the opposite case, an alternative backup representation is used for the corresponding records.



New VisualDCT v3 Capfast record symbol example



VisualDCT v2.x record symbol example



A library of CapFast EPICS record symbols is included in the VisualDCT distribution. The relative or absolute paths to symbol files or folders containing symbol files must be specified in the capfast widget factory configuration file (saved.properties.com.cosylab.vdct.visual.capfast.RealCapfastWidgetFactory) in the loaded VisualDCT configuration folder. Example:

```
symbolPaths=./capfastlib/Epics ./capfastlib/Default
```

### 3.2. MORHPING

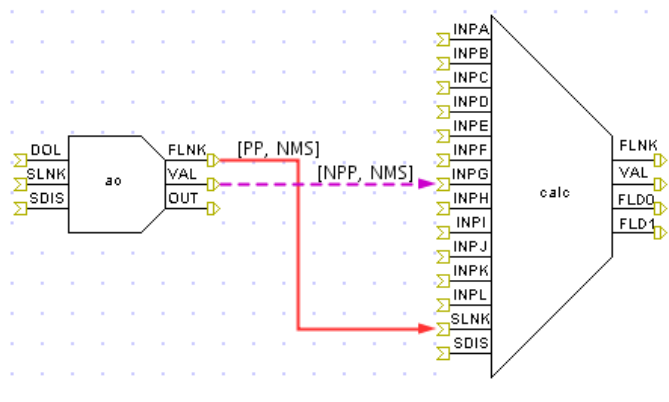
Record fields can be edited in the property editor. VisualDCT has the capability to morph (change) record types, i.e. to change type and preserving all common fields. The command is accessible via the pop-up (context) menu or the action menu bar.

### 3.3. LINKING

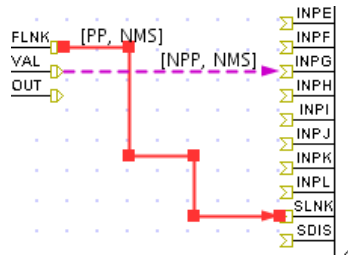
Most changes have been introduced in the way how records are connected. Now, it is no longer necessary to choose a link type and record. Instead a link can be created by simply dragging the mouse from one to another pin. Currently the VisualDCT only supports normal links.

In general, there are two ways of linking:

- value of the INPUT, OUTPUT or FORWARD link field is entered using the property editor
- using only mouse: Left click on the pin and then drag the mouse to the other desired pin.



The link position/path can be changed with control points. Control points can be added/removed with a double click on the link-control point.



### 3.4. IRMIS

The new VisualDCT also supports loading database definitions and record instance databases from a remote IRMIS database location. This can be done through the **Module → Load IOC from IRMIS** action. It brings up a input dialog, where the address of a IOC inside a remote IRMIS database can be specified.

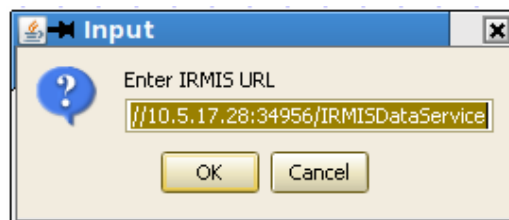


Figure 6: Input dialog for specifying remote IRMIS database location



Note that at present VisualDCT is unable to browse the IRMIS database for available IOCs. Therefore the IRMIS address should include the exact name of an IOC residing in the remote IRMIS location.

### 3.5. THE MODULE MENU

When EPICS v3 module is loaded, the **Module** menu is populated with the following new menu actions:

- *Configure DBDs* brings up a dialog, where the locations of the database definition (.dbd) file/s can be specified.
- *Load IOC from IRMIS* brings up a input dialog, where the address of a IOC inside a remote IRMIS database can be specified
- *Configure Node Widgets* allows for the record node L&F to be edited. Different record types can have different foreground colors.
- *Configure Connection Widgets* allows for the link L&F to be edited. Different link types can have different visualizations (color, dashing, arrows, line width).



## 4. PVDATA

The new EPICS v4 is based on xml encoded structure, record and database definitions. The major differences relevant to database editing with VisualDCT are outlined in the next sections.

### 4.1. HIERARCHIES

Pvdata supports arbitrary nesting hierarchies of structures and thus requires substantially different database design approach compared to EPICS v3, whose databases are essentially flat. Moving deeper into the model hierarchy can be done with the Hierarchy view by expanding the model tree structure and double clicking on nodes, or by using the **Explore** action. Moving back to higher levels within the hierarchy can be done either again through the tree representation of the model in the Hierarchy view or using the address field on top of the VisualDCT GUI.

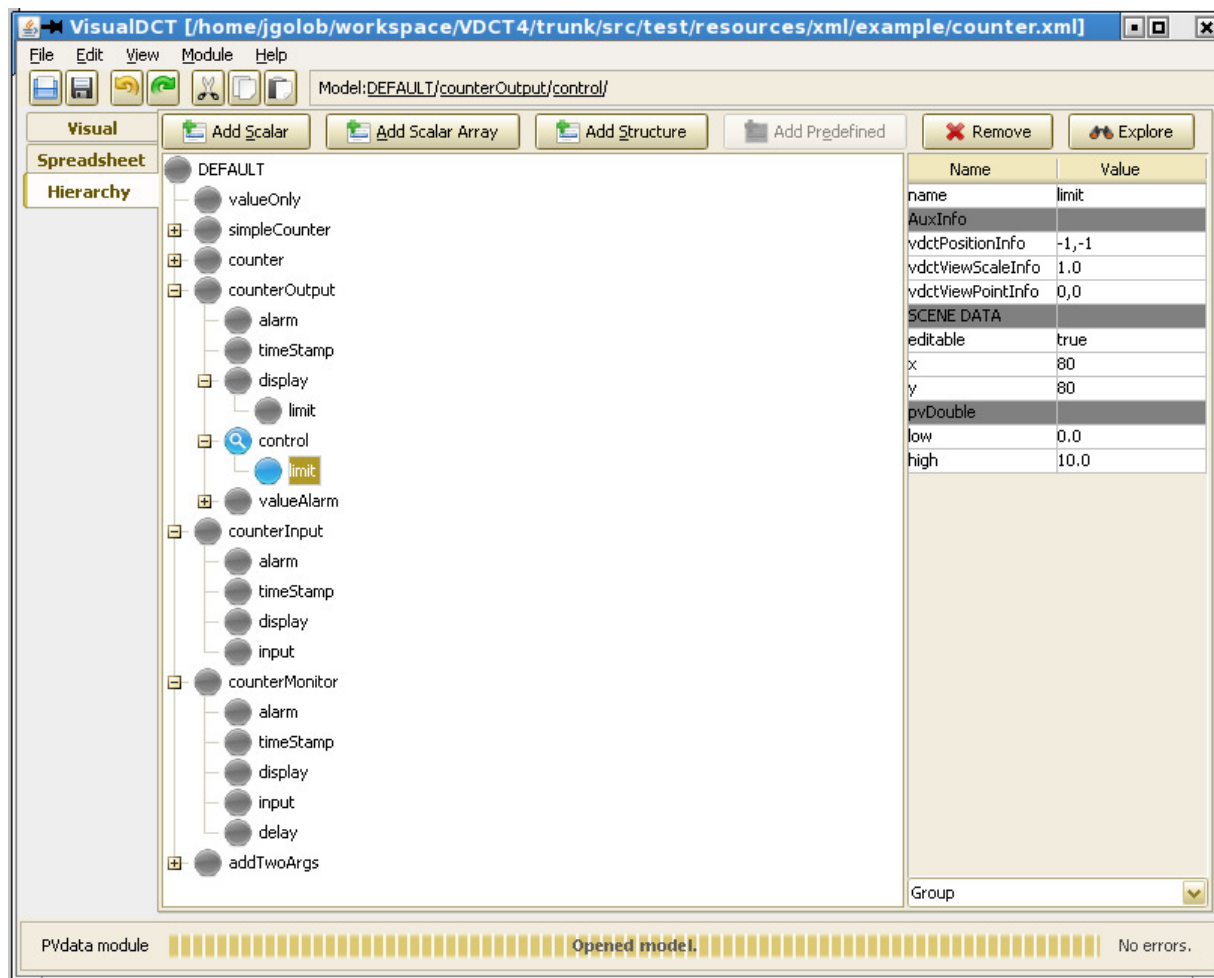


Figure 7: pvdata example database in hierarchy view

## 4.2. RECORD AND STRUCTURE VISUALIZATION

Due to the richer hierarchical structure, records and structures typically have fewer fields compared to EPICS v3 records. In addition, no CapFast symbols are available at the moment even for the basic pvdata structures and records. Thus, the visualization of pvdata records and structures in VisualDCT is different from those in the EPICS v3 module. Both Records as well as nested Structures at deeper hierarchy levels are represented as nodes in the VisualDCT visual view.

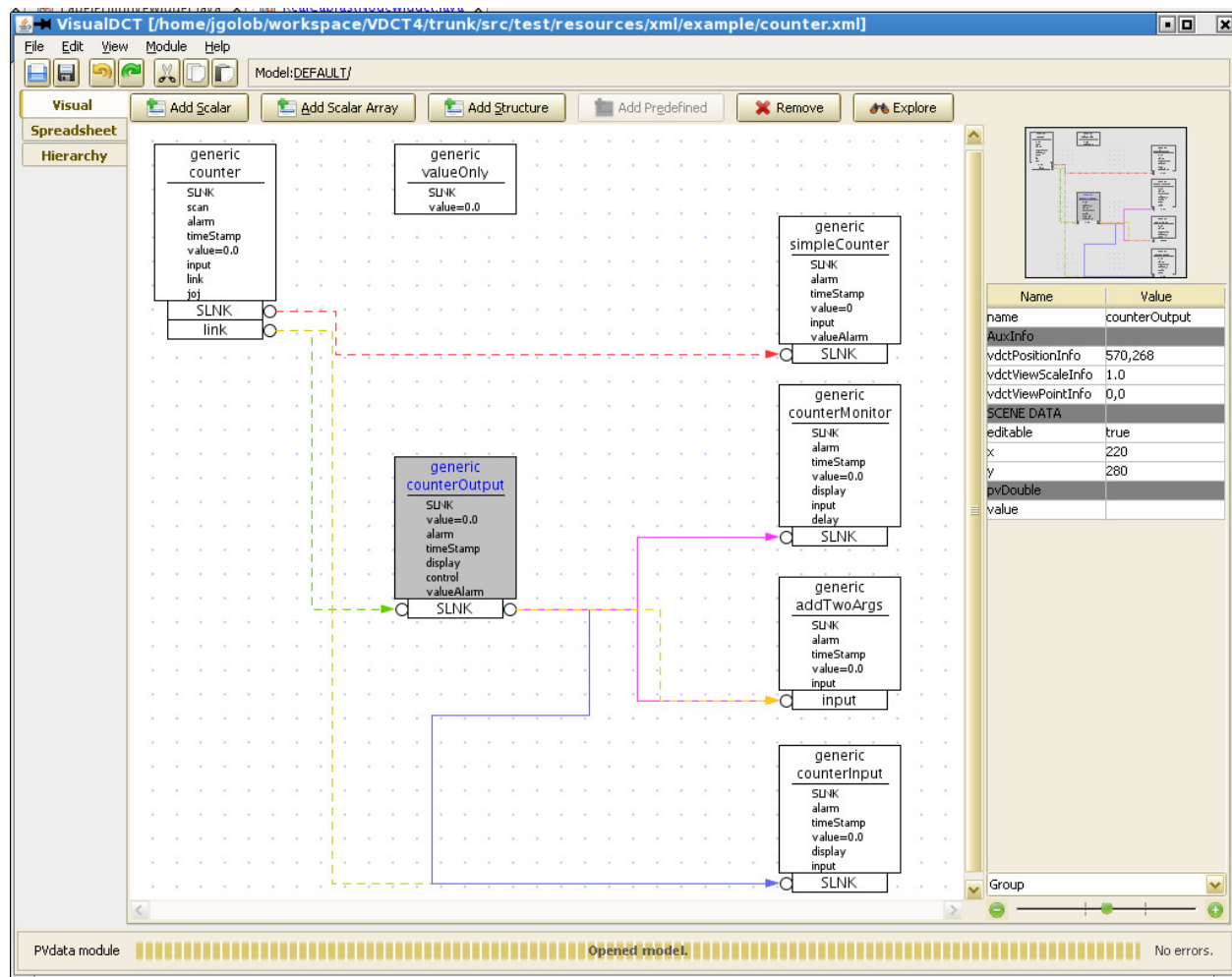


Figure 8: pvdata example database in visual view

## 4.3. DATABASE CONSTRUCTION AND LINKING

Pvdata supports including dependencies of xml database definition files. VisualDCT supports this functionality in two levels.

Locations of the included xml files inside a particular opened file can be specified using the **Module Includes** action which allows to add any number of dependencies into the file itself.

Static includes can be specified, which are not written into the model file, but are loaded on VisualDCT startup and made available to all loaded pvdata models. They are specified through the **Module -> Configure Static Includes** action dialog. The settings are saved to the pvdata module configuration file (saved.properties.com.cosylab.vdct.model.pvdata.PVModule) in the loaded configuration folder as:

```
staticIncludes=./example/pvdata/include/structures.xml
```

Contrary to EPICS v3, PVDATA databases allow to define arbitrary new structure or scalar fields within existing records or structures. VisualDCT supports this functionality, since it allows to instantiate existing as well as add new fields to record or structure nodes.

Known Pvdata links are supported and can be created by dragging the mouse between two pins. The link type and other properties are specified subsequently through wizard-type dialogs.

## 4.4. THE MODULE MENU

---

When pvdata module is loaded, the **Module** menu is populated with the following new menu actions:

- *Configure Static Includes* brings up a dialog, where the locations of statically included pvdata xml file/s can be specified.
- *Configure Node Widgets* allows for the record node L&F to be edited. Different node types can have different foreground, background and border colors.
- *Edit Connection Widgets* allows for the link L&F to be edited. Different link types can have different visualizations.