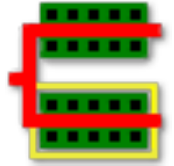


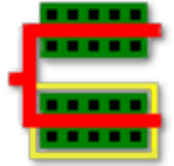
Glade

Peardrop Design Systems
23rd June 2021



Part 1 Agenda

1. About Glade
2. Starting Glade
3. The Glade GUI
4. Techfile setup
5. Importing design data
6. Viewing design data
7. Labs



1. About Glade

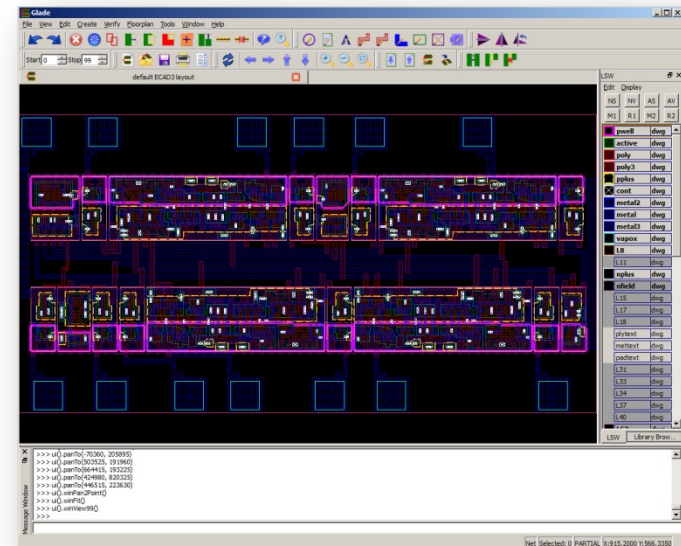
- **GDS LEF and DEF Editor**
 - Multipurpose EDA tool
 - Allows viewing and editing of design data in a many formats
 - Multi Platform: runs on Windows, Linux, Mac
 - Simple yet powerful

Programmable in Python

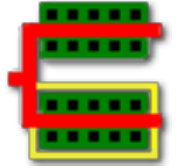
Pcells (parameterised cells) in Python

DRC, extraction, LVS

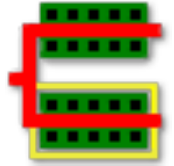
Etc...



2. Starting Glade



- Command line / icon
- Command line args
- Environment Variables



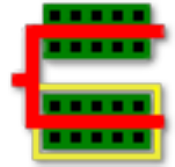
Starting Glade

- From the icon
 - Set file association if you want to double click on e.g. a GDS file and have Glade open it.
- From the command line
 - glade [options]
- On startup, Glade reads:
 - `$HOME/gladerc.xml` **and then** `./gladerc.xml`
 - Display and selection settings, window arrangement.
 - `$HOME/.glade.py` **and then** `./glade.py`
 - Python script that is run on startup. Useful for loading default techfile, project library etc. Note that this is read **before** any command line options.
- Glade checks for OpenGL support
 - Can be disabled using env var `GLADE_USE_OPENGL=no`

Command line args

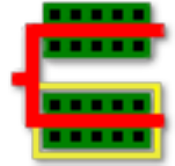
- Glade can be started from the command line with various options:
- `glade [-ng] [-library <directory>] [-libName <name> [-tech <file>] [[-map <file>] -drf <file> -tf <file>] [-gds <file>] | [-gdsout <file>] | [-oasis <file>] | [-oasisout <file>] | [-dxf <file>] | [-lef <file> -def <file>] | [-edif <file>]] [-cell <name>] [-cellview <cell> <view>] [-script <file>] [-h] [-v]`
- Multiple options can be used, e.g.
 - `Glade -tech lefdef.tch -lef lef1.lef -lef lef2.lef -def design.def`
 - `Glade -library XyceLib -library FreePDK15 -cell nand -script drc.py`

Environment Variables



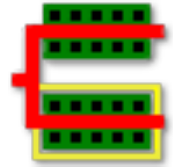
- GLADE_HOME – used to locate HTML help files
- GLADE_LOGFILE_DIR – where logfiles are created
- GLADE_USE_OPENGL – disable OpenGL
- GLADE_DRC_FILE – default DRC file location
- GLADE_EXT_FILE – default LPE file location
- GLADE_NETLIST_FILE – default reference netlist for LVS
- GLADE_DRC_VARS – Python global vars for DRC
- GLADE_EXT_VARS – Python global vars for LPE
- GLADE_THREADED_EXTRACTION – number of threads to use
- GLADE_FASTCAP_WORK_DIR – used by Fastcap
- GLADE_NO_DELETE_TMPFILES – don't delete Fastcap mesh files
- GLADE_DEBUG_SUBMASTERS – show Pcell submasters in lib browser
- PYTHONPATH – Used by Glade's embedded Python interpreter

3. The Glade GUI



- GUI components
- LSW
- Library Browser
- Hierarchy Browser
- Net Browser
- World View
- Bindkeys
- Function Keys

GUI



Menus

Toolbars

Tab/MDI windows

Bindkeys

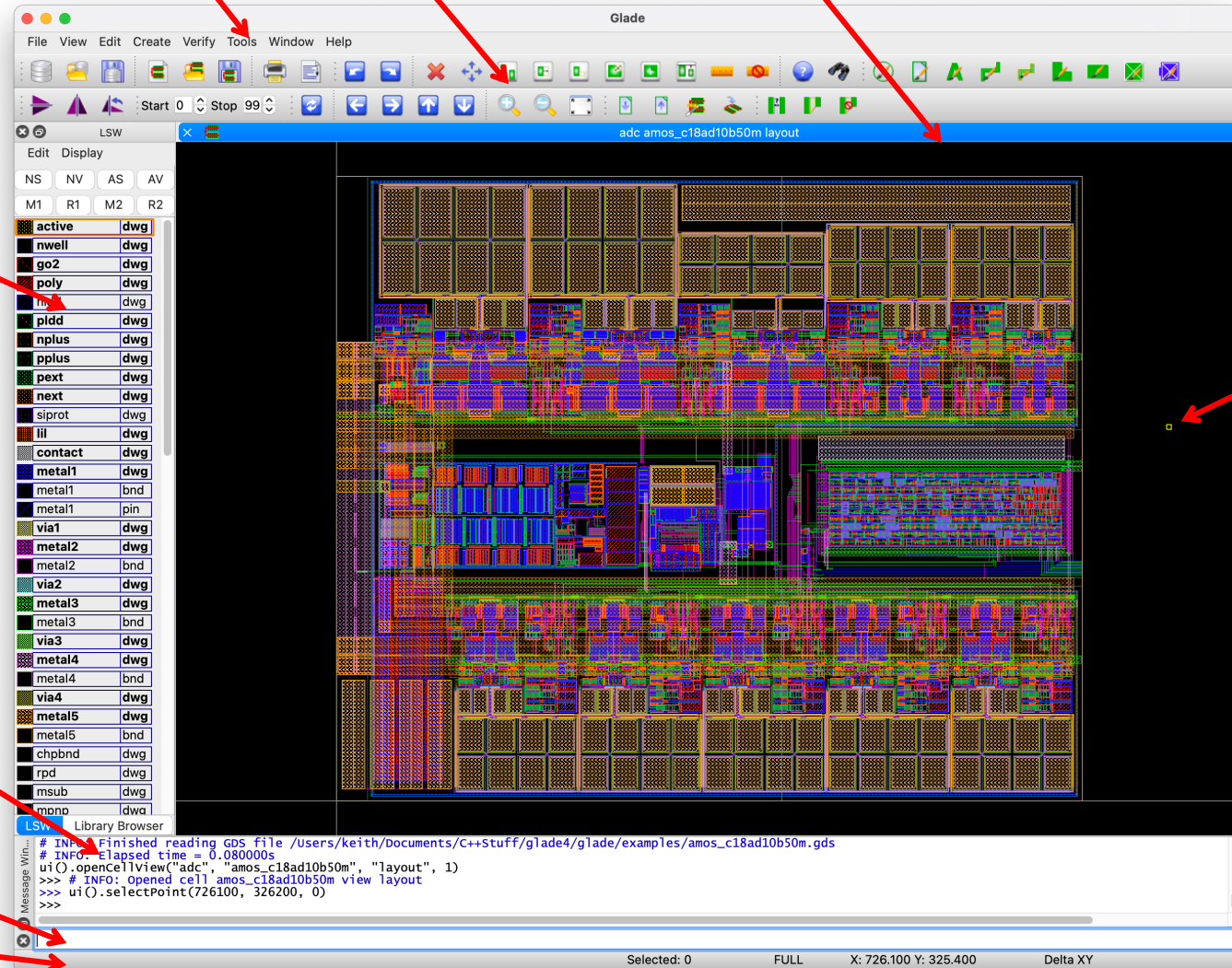
Dock windows

Cursor

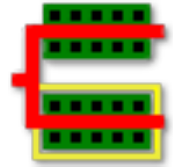
Message dock window

Command line

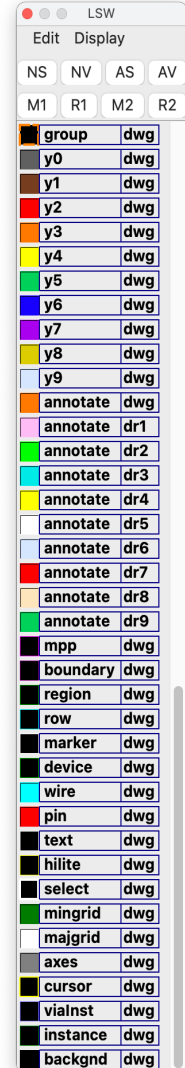
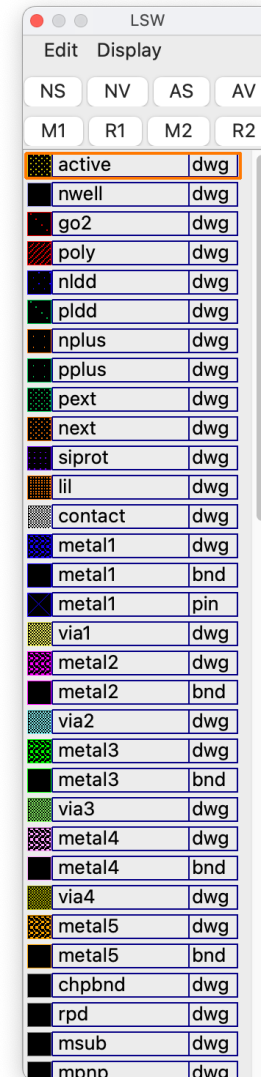
Status bar

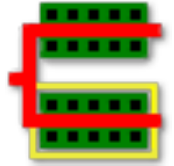


The LSW



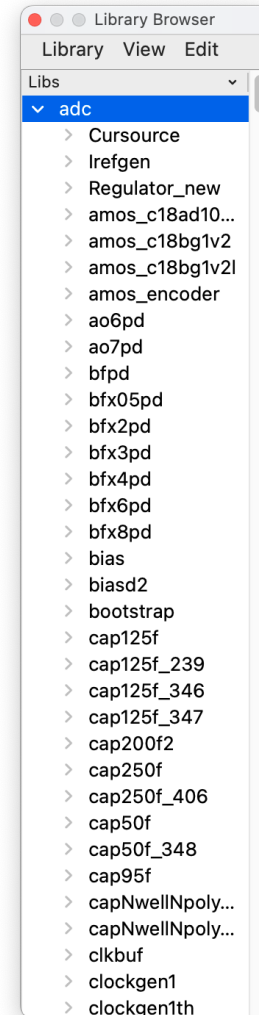
- Layer Select Window:
 - Controls layer visibility and selectability
 - Allows easy changing of layer color and/or fill pattern
 - Default is set by the techfile
 - 2 memories for frequent changes
 - User layers and System layers
 - Generally can't draw on system layers



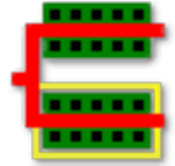


The Library Browser

- Library Browser:
 - Displays cells and their views in the library
 - Create, open, copy, rename, delete cells and/or cellViews
 - Multiple libraries supported
 - Library A's cellViews can have instances from library B



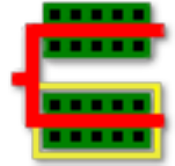
The Hierarchy Browser



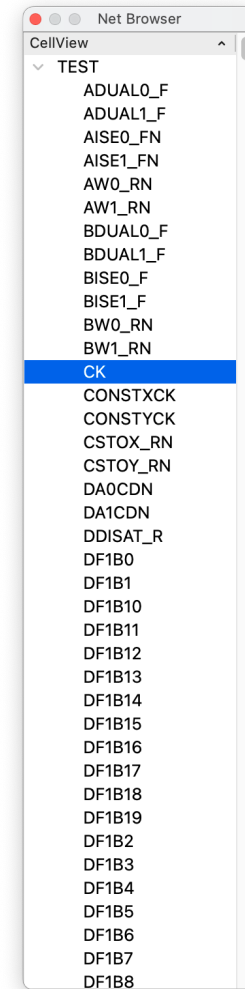
- Hierarchy Browser:
 - Displays the hierarchy of the design
 - Descend instances in the hierarchy
 - Click on cell/instance in the browser to select it in the design



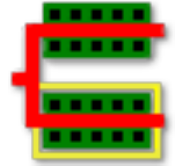
The Net Browser



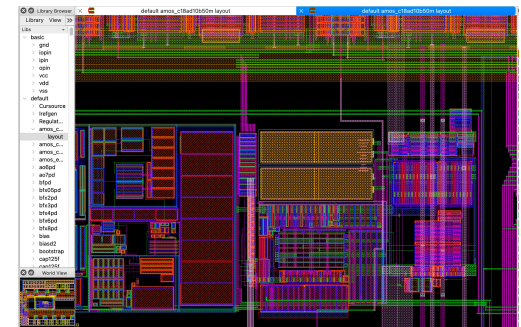
- Net Browser:
 - Displays the nets in a design
 - Can select instances connected to the net (drivers/loads)
 - Can select shapes of the net
 - Can delete a net

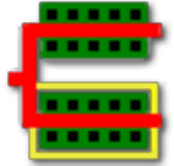


The World View



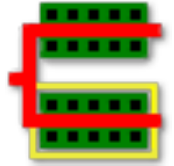
- World View:
 - Displays the complete design in a small window
 - Yellow box indicates the canvas viewport
 - Move the viewport by dragging the yellow box
 - Right mouse or wheel zooms viewport





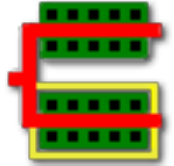
Bindkeys

- 4 ways of issuing commands
 - Menu entry
 - Toolbar icon
 - Command line (python)
 - Bindkey
- Bindkeys may be changed (Edit->Bindkeys...)
 - Combination of keys & modifiers (shift, ctrl, alt)
- Bindkeys are saved to gladerc.xml file



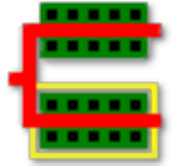
Function Keys

- F1 – Help
- F2 – Toggle Gravity
- F3 – Show/Hide option dialogs
- F4 – Toggle Full/Partial select
- F5 – Show Coord entry dialog
- F6 – Toggle connectivity display
- F7 – Toggle Object/Net selection
- F8 – Toggle Immediate Move
- F9 – Cycle through snap modes



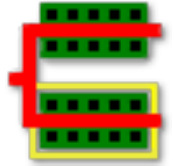
4. Techfile Setup

- Design data can be imported without a techfile for quick viewing
 - E.g. GDS2, OASIS, LEF/DEF
 - Generated layer names (or real layer names for LEF/DEF)
 - Random colors, no fill
- Cadence/Silicon Canvas techfile can be imported
- Techfile can be saved and/or manually edited



Techfile Setup

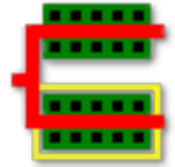
- The techfile is an ascii file containing information about the process you are working on.
- It defines
 - Layer names, colors, fill patterns, linestyles
 - Layer functions (routing / via etc)
 - Layer connections (used by net tracer)
 - Width/spacing rules
 - Via definitions
 - MPP (MultiPartPath) rules
- Glade can also read Cadence display.drf/ .tf or Laker .dsp/ .tf techfiles.



LAYER section

//	Name	Purpose	gds_num	gds_dtyp	RGBA	sel?	vis?	fillstyle	linestyle	valid?
LAYER	psub	drawing	0	0	(217,150,150,128)	t	t	empty	dashed2	t ;
LAYER	nwell	drawing	1	0	(150,150,217,128)	t	t	empty	dashed2	t ;
LAYER	od	drawing	2	0	(217,204,0,128)	t	t	dots_rare	plain	t ;
LAYER	poly	drawing	3	0	(255,0,0,128)	t	t	zagr1	plain	t ;

- Layer order in LSW matches order in techfile
- Layers have GDS number / datatype mappings
- Layer colour as RGBA (alpha)
- Visibility and Selectability
- Fill style (name of stipple pattern)
- Line style (name of line style pattern)
- Valid (whether displayed in LSW or not)
- Note all techfile lines end with ‘;’

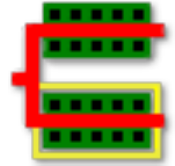


FUNCTION section

```
// Layer Function.  
//  
FUNCTION poly      drawing ROUTING ;  
FUNCTION cont      drawing CUT ;  
FUNCTION metal1    drawing ROUTING ;  
FUNCTION via12     drawing CUT ;
```

- **FUNCTION** of the layer/purpose pair can be one of:
 - CUT (a via or contact layer)
 - ROUTING (a layer that can have connectivity)
 - BLOCKAGE (as in LEF/DEF)
 - MASTERSLICE (as in LEF/DEF)
 - PIN (as in LEF/DEF)
 - OVERLAP (as in LEF/DEF)
 - WELL
 - DIFFUSION
 - POLY
 - IMPLANT
 - NONE
- Mainly used in LEF/DEF interface

CONNECTIONS section



```
// Layer Connections.
```

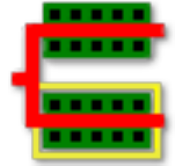
```
//
```

```
CONNECT poly    drawing BY cont  drawing TO metal1 drawing ;
```

```
CONNECT metal1 drawing BY via12 drawing TO metal2 drawing ;
```

```
CONNECT metal2 drawing BY via23 drawing TO metal3 drawing ;
```

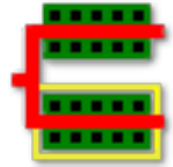
- Used to define connectivity for e.g. the Net Tracer
- Connect layer1 to layer2 by a via layer
- Can also directly connect two layers
 - CONNECT li drawing TO poly drawing ;



Spacing Rules section

```
// Spacing rules.  
//  
MINWIDTH nwell drawing 1.80 ;  
MINSPEACE nwell drawing 1.20 ;  
MINAREA active drawing 0.45;  
MINENC nwell drawing active drawing 0.20 ;  
MINOVLP poly drawing active drawing 0.18 ;
```

- Defines simple layer rules,
 - used by Verify->Check... command
 - Used by DRD (Design Rule Driven) layout
- Can also be used e.g. in pCells

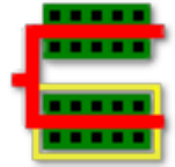


Via rules section

```
// Via rules.  
//  
VIA od_m1  
    metal1 drawing -0.130 -0.130 0.130 0.130  
    cont drawing -0.080 -0.080 0.080 0.080  
    od drawing -0.150 -0.150 0.150 0.150
```

- Defines via definitions for Create->Via command
- Vias have an upper and lower layer normally of function ROUTING
- Vias have a layer of function CUT
- Can have multiple cut shapes in vias e.g. to create a 2x1 or 1x2 via.

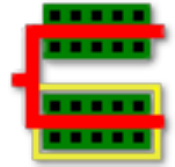
MultiPartPath rules section



```
// MultiPartPath rules
//
MPP nguard LAYER nwell drawing WIDTH 1.80 BEGEXT 0.90 ENDEXT 0.9 OFFSET 0.3 ;
MPP nguard LAYER od drawing WIDTH 1.18 BEGEXT 0.59 ENDEXT 0.59 ;
MPP nguard LAYER nimp drawing WIDTH 1.54 BEGEXT 0.77 ENDEXT 0.77 ;
MPP nguard LAYER cont drawing WIDTH 0.16 BEGEXT -0.08 ENDEXT -0.08 SPACE 0.18 LENGTH 0.16 ;
MPP nguard LAYER metal1 drawing WIDTH 0.60 BEGEXT 0.30 ENDEXT 0.30 ;
```

- MPP rules define how a MPP object is created
 - A MPP is like a normal path but with multiple layers and (optionally) contacts between the layers
 - Typically used for guard rings.
 - Can be used in Pcell code.
 - Can be edited and moved/stretched just like a normal path.

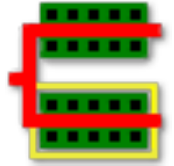
Stipple pattern section



```
// Name Type Fill pattern
STIPPLE     zagl     STIPPLE
```

```
0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1
0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0
0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0
1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0
0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1
0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0
0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0
1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0
0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1
0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0
0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0
1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0
0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1
0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0
0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0
1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0
;
```

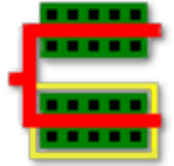
- Stipple patterns can be 8x8, 16x16, 32x32
- Can create/edit with the stipple pattern editor



Line style section

```
// Name Width Style
LINE      plain      0      SOLID ;
LINE      thicksolid  4      SOLID ;
LINE      thick       2      SOLID ;
LINE      dashed2     2      DASH ;
LINE      dotted      0      DOT ;
LINE      dashdot     0      DASHDOT ;
LINE      dashdotdot  0      DASHDOTDOT ;
```

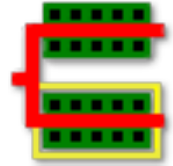
- Line styles used to define the line surround of a shape
- Defines the width of the lines (0 is default width)
- Also defines any pattern
 - – – – – DASH
 - DOT
 - – . – . – . DASHDOT etc.
- Can be set in the stipple pattern editor



5. Import/Export Data

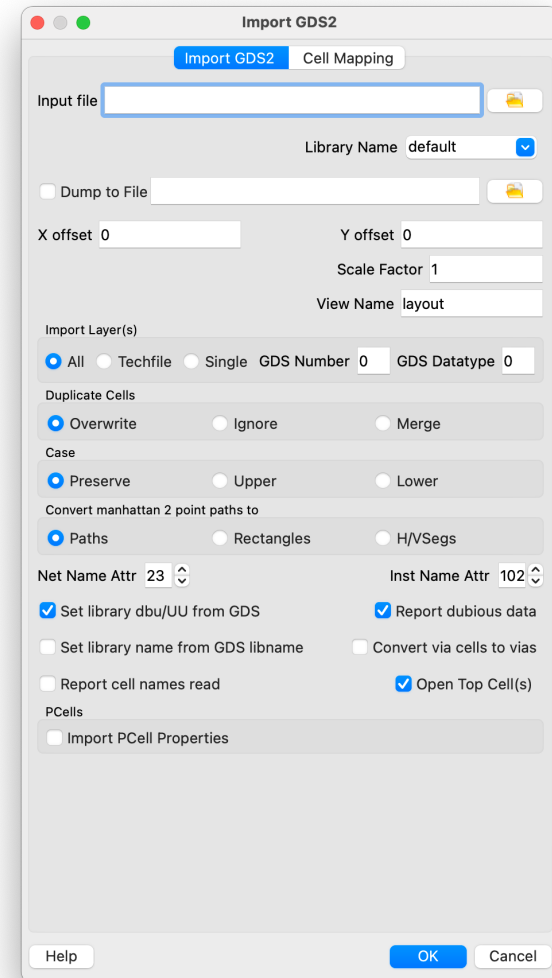
- Import
 - Glade techfile
 - Cadence techfile
 - GDS2
 - OASIS
 - LEF
 - DEF
 - EDIF
 - DXF
 - Verilog
 - CDL
- Export
 - Glade techfile
 - GDS2
 - OASIS
 - LEF
 - DEF
 - EDIF
 - DXF
 - Verilog
 - CDL

Import GDS2

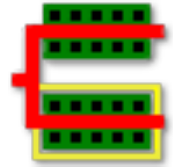


Many options for GDS2 import:

- Offset Coords
- Scale data
- Choose layer(s) to import
- Handle duplicate data
- Read attributes
- Reads gzip/compressed data automatically

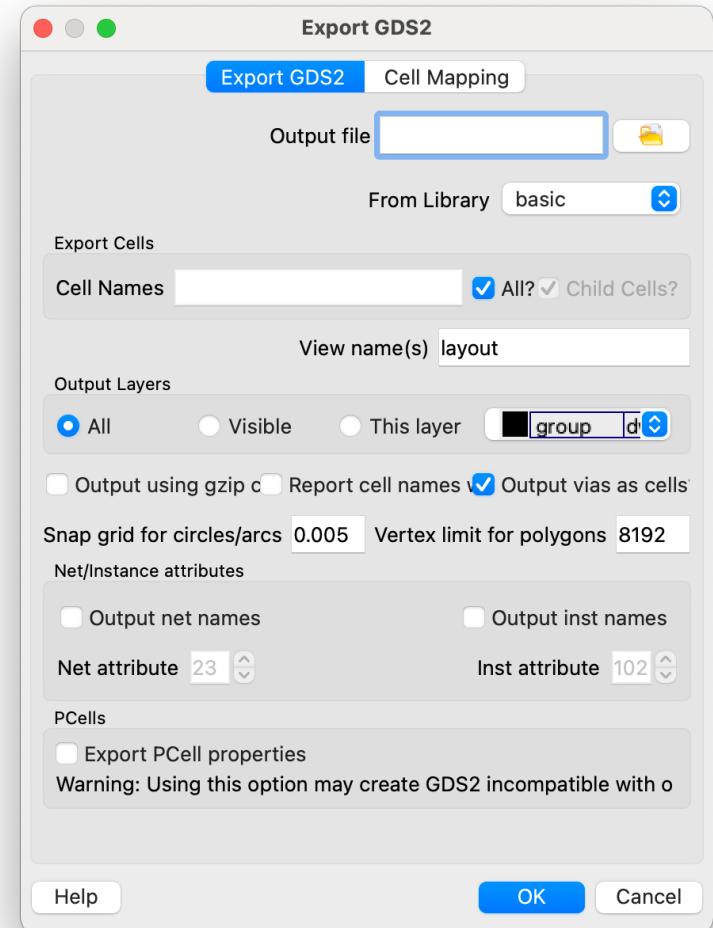


Export GDS2

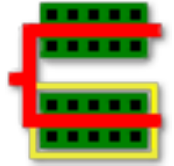


Many options for GDS2 export:

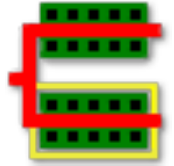
- Cell(s) to export
- Choose layer(s) to export
- Write attributes for net names and inst names
- Pcell property info can be written so design data containing Pcells can be transferred to other Glade users.



6. Viewing Design Data

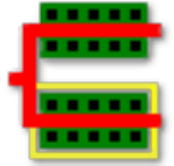


- Navigating a design
 - Zooming and Panning
 - Control visible hierarchy displayed
 - Rulers
 - Using the LSW
 - Querying
 - Searching
 - Setting Options



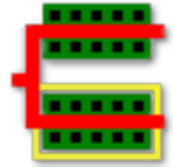
Window Fit & Zoom

- Fit and Fit+ set canvas viewport to design extent
 - Fit+ allows a margin, set by Fit% in Pan/Zoom Options
- Zoom In / out:
 - Bindkeys (Z, shift+Z)
 - Zoom in or out by amount set in Pan/Zoom Options
 - Mouse Wheel / trackpad
 - Right Mouse drag
 - Drag left-to-right to zoom in by area
 - Drag right to left to zoom out by area
 - Display Options RMB Mode controls this behaviour
- Zoom Selected fits selected set to canvas viewport
- Zoom using World View dock window



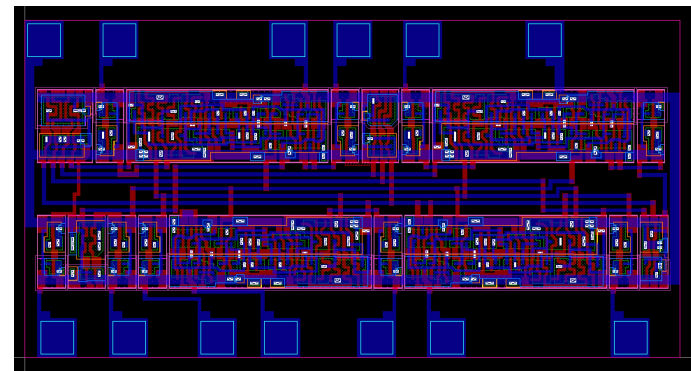
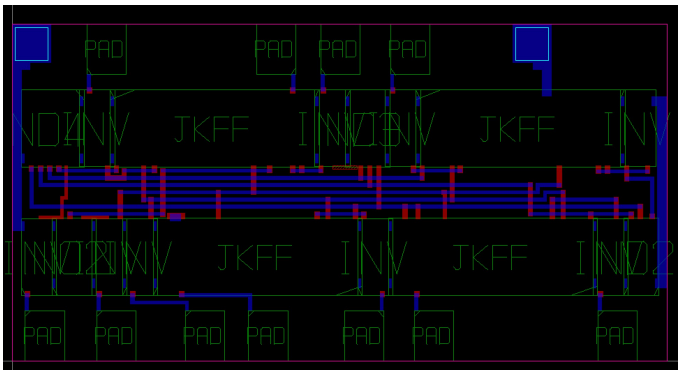
Window Panning

- Pan using Bindkeys
 - Arrow keys by default
- Pan using tab key
 - Pans such that canvas viewport center is moved to cursor position
- Pan using Middle Mouse button
 - Realtime panning
- Pan to Point
 - Can use F5 bindkey to enter a coordinate
- Pan using World View dock window

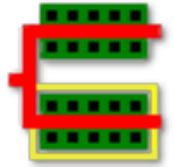


Display Levels

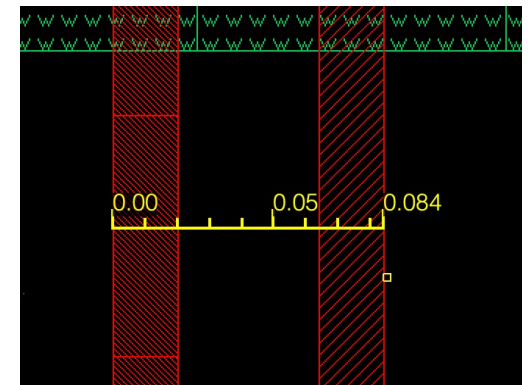
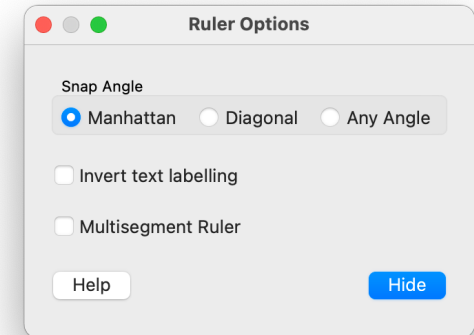
- Glade displays hierarchical designs
 - instances or arrays of cells placed within other cells
 - 0 is the top level
 - No limit to the number of levels (99 by gui)
 - Start level is the first level of hierarchy to display
 - Normally 0
 - Stop level is the last level
 - Instances are shown as outlines if they are on a level greater than stop level



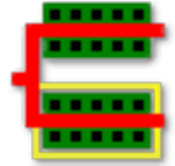
Rulers



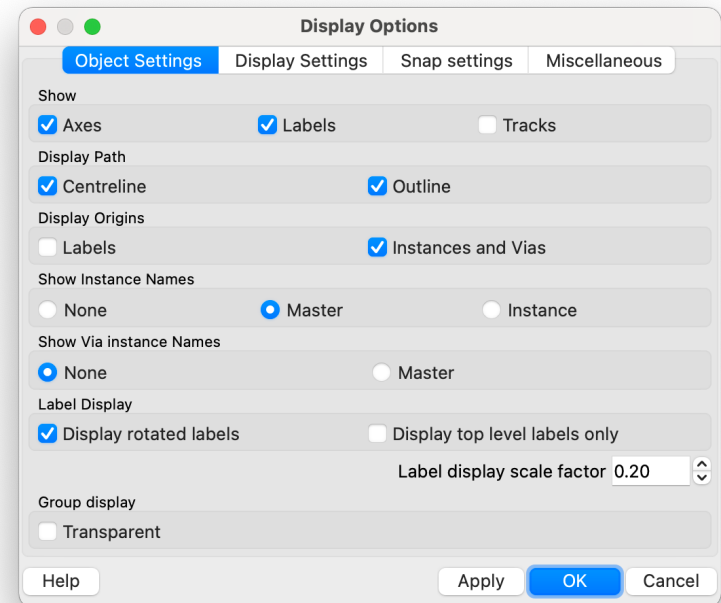
- Use icon, menu or bindkey (K) to create rulers
- F3 bindkey shows/hides options
 - Snap angle
 - Manhattan
 - Diagonal
 - Any angle
 - Multisegment ruler
- Rulers can snap to edges/vertices
 - Controlled by Display Options Snap Settings
- Rulers are not permanent
- Delete all rulers (Shift+K bindkey)



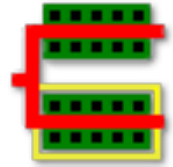
Display Options 1



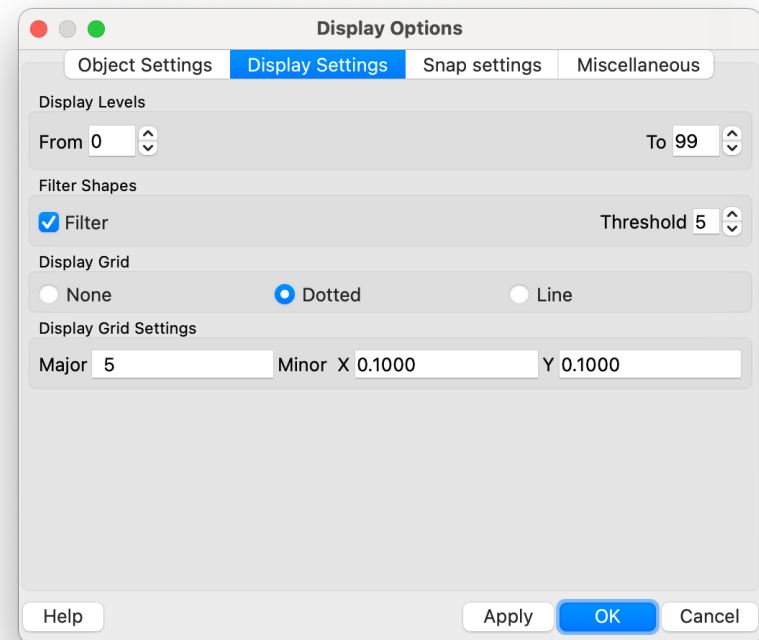
- Object settings control what is displayed
 - Axes, text labels
 - Path centreline/outline
 - Origins
 - Instance names (when display level does not show contents)
 - Label display
 - Grouped object display mode



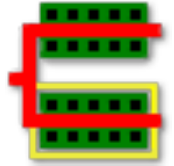
Display Options 2



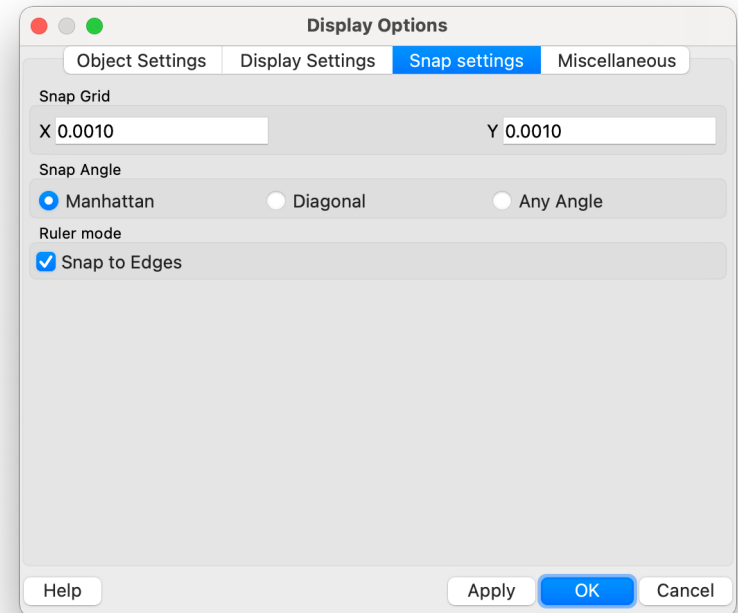
- Display settings
 - Display start/stop level
 - Shape filtering
 - Small shapes can be filtered to speed drawing of large designs
 - Display grid



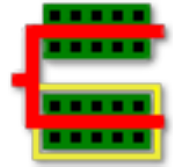
Display Options 3



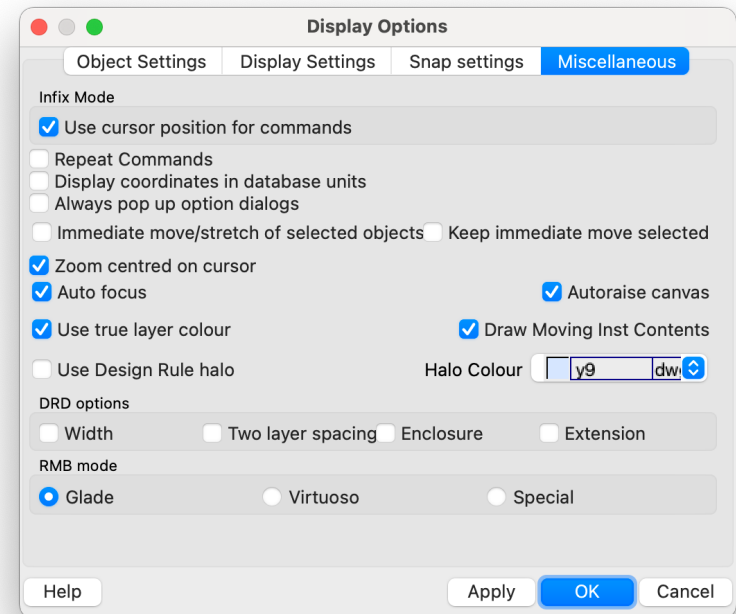
- Snap settings
 - Snap grid for cursor
 - Snap angle for many commands
 - Ruler entry mode
 - Snap to edges highlights and snaps ruler to edges of shapes



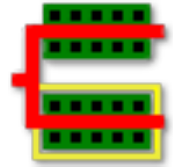
Display Options 4



- Miscellaneous settings
 - Many options
 - Immediate Move F8 key



Selection Options



- Selection Mode (F4 key)
- Selection Type (F7 key)
- Connectivity flightlines
- Dimming
- Gravity (snapping cursor to nearest objects) (F2 key)
- Dynamic highlight
- Dynamic Net Highlight

Selection Options

Selection Mode: ☒ Full ☐ Partial

Selection Type: ☒ Object ☐ Net

Max pins: 10

☐ Show connectivity

☐ Show Inst Pins ☐ Show Selected item connectivity

☒ Limit instPin query for net: 1000

Dimming

☐ Dim unselected objects ☐ Dim unhilit objects

Dim factor (%): 70

Gravity

☒ Gravity Depth: 0 Range: 0.1000

Gravity snap to: ☒ Edge ☐ Vertex

Gravity to Path: ☒ Centre ☒ Outline

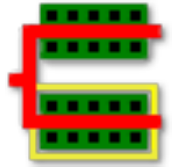
☒ Display Cursor Box ☐ Cross cursor

☐ Dynamic Highlight

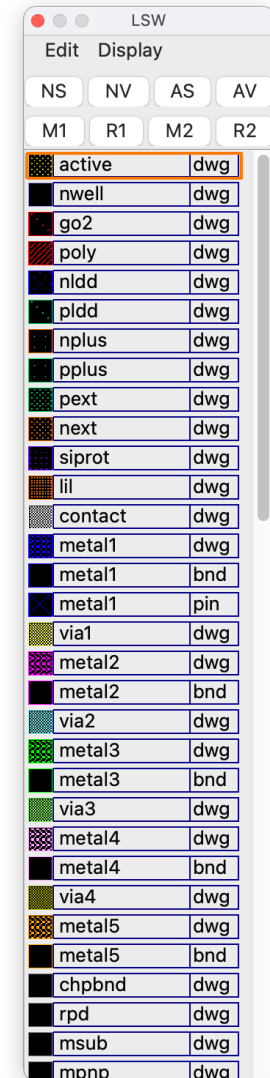
☐ Dynamic Net Highlight

Help OK Cancel

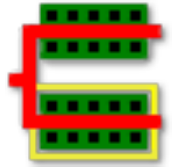
LSW usage



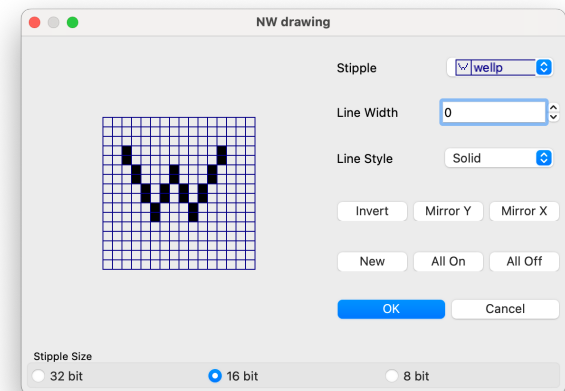
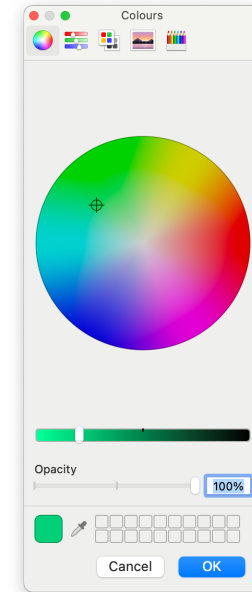
- Toggle visibility of layers by RMB on layer name
 - NV/AV buttons toggle all off/on
- Toggle selectability of layers by MMB on layer name
 - NS/AS buttons toggle all off/on
- Set the 'Current Layer' by LMB on layer name
 - This is the layer used in 'Create' cmds
- Edit layer color by LMB on color box
- Edit stipple pattern/line style by RMB on color box
- 2 memory saves M1/M2 and memory recalls R1/R2



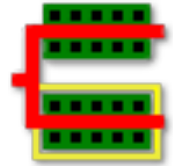
Editing layer colors



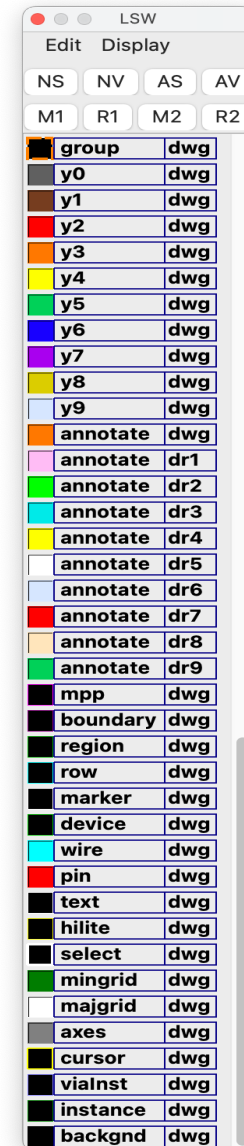
- Layer color chooser uses platform-specific dialog
 - Note opacity (alpha blending) is supported
- Stipple pattern editor
 - Create or modify stipple patterns

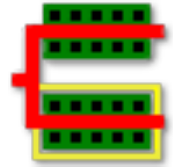


System Layers



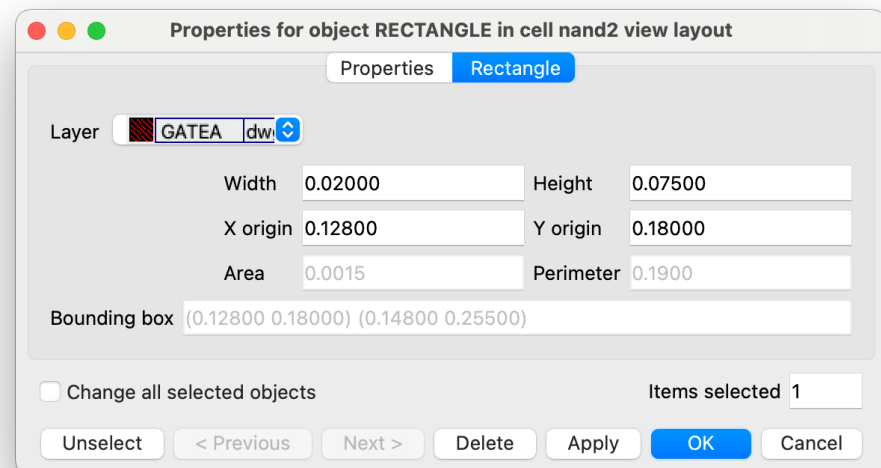
- background – canvas background color
 - Some people prefer white
- instance – instance box/names
- vialnst – as instance
- cursor – for cursor box, stretching, dynamic highlight etc.
- axes
- min/mag grid
- select – outline color for selection
- device/wire/pin – for schematics/symbols
- marker – for DRC errors
- boundary/row/region – for LEF/DEF usage
- Group – for grouped objects
- y0—y9 scratch layers
- annotate for e.g. symbols



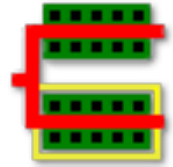


Querying Objects

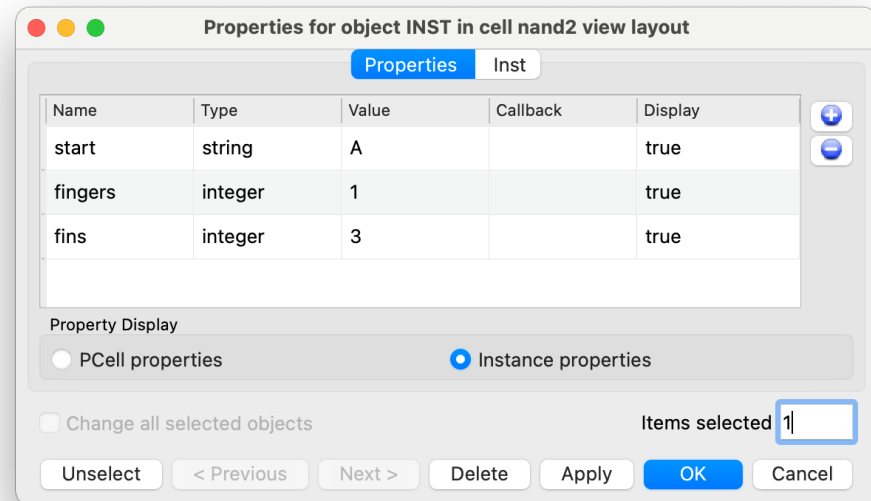
- Query Dialog (bindkey Q)
 - Allows changing object attributes and properties
 - Change attributes of all selected objects
 - Unselect objects
 - Delete objects



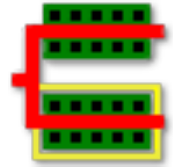
Querying Objects



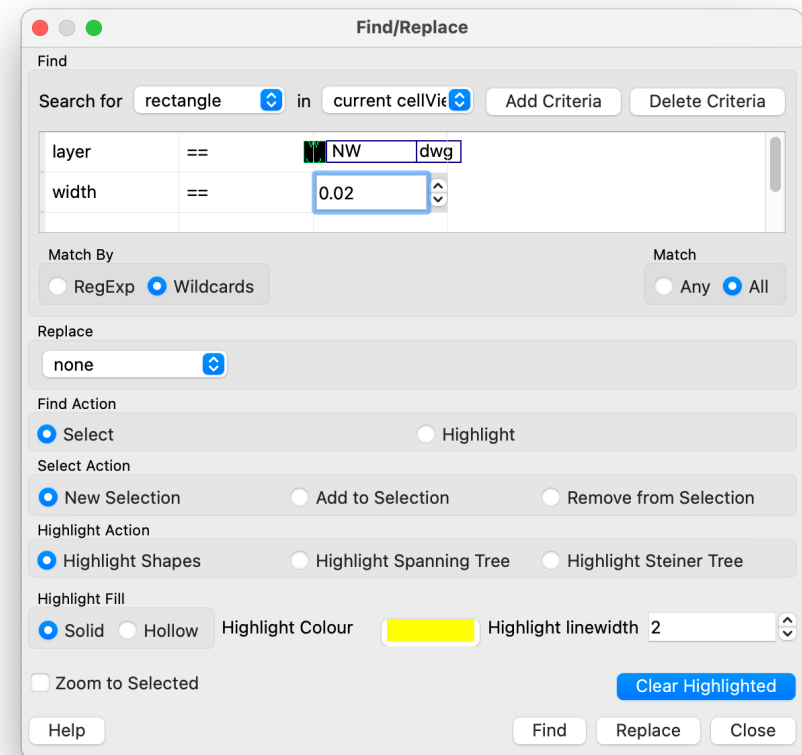
- Query properties
 - Some object have properties (e.g. Pcell instances)
 - Name, type, value, optional callback and display

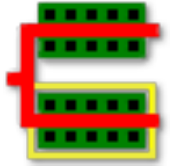


Searching



- Search for objects
- Set multiple search criteria
 - Add/remove from selection
- Highlight net shapes
 - Or steiner/spanning tree for unrouted nets
- Replace objects





7. Labs

- Start Glade
- Import design
- View design data