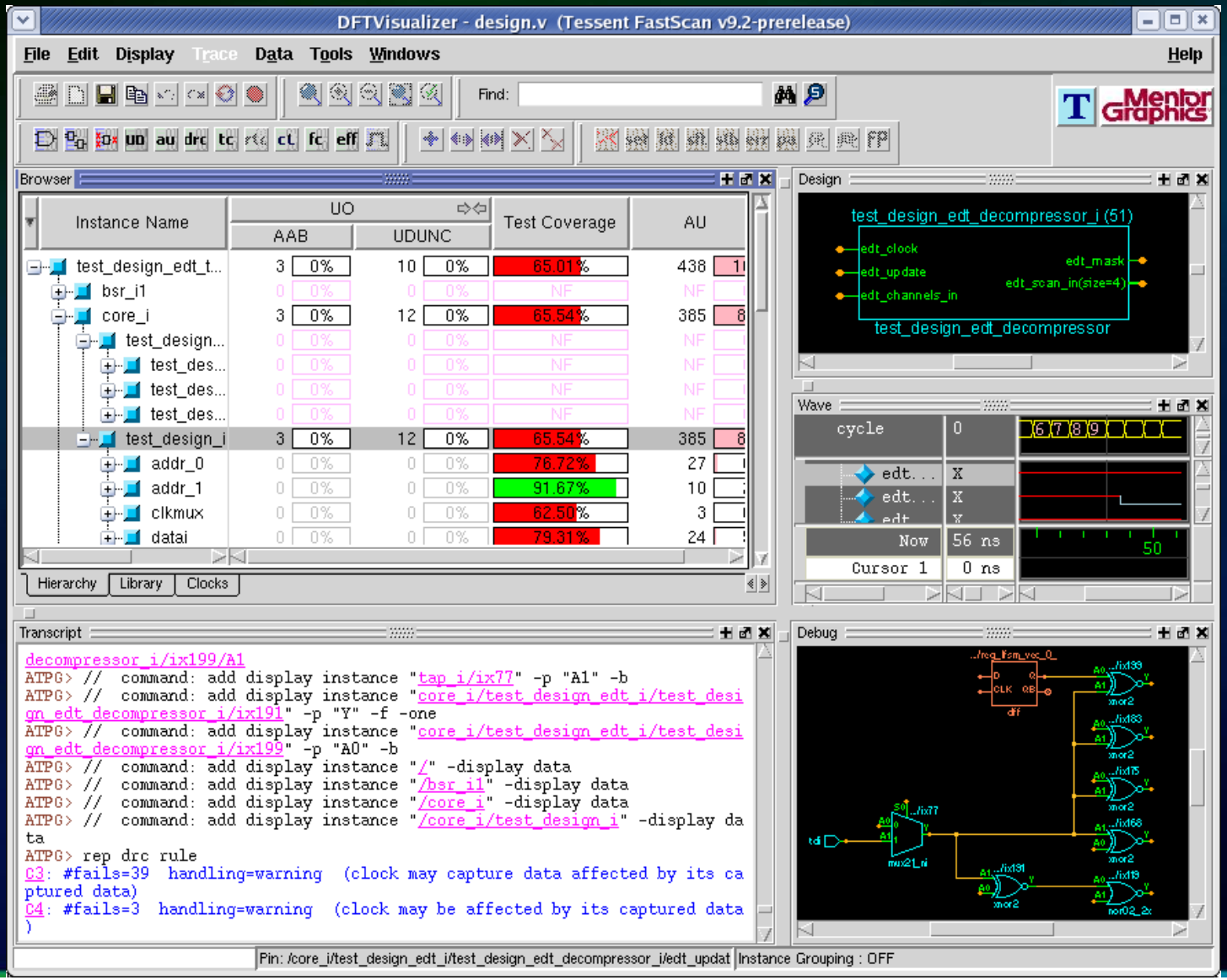
Enhanced Tcl/Tk Widgets for EDA Applications

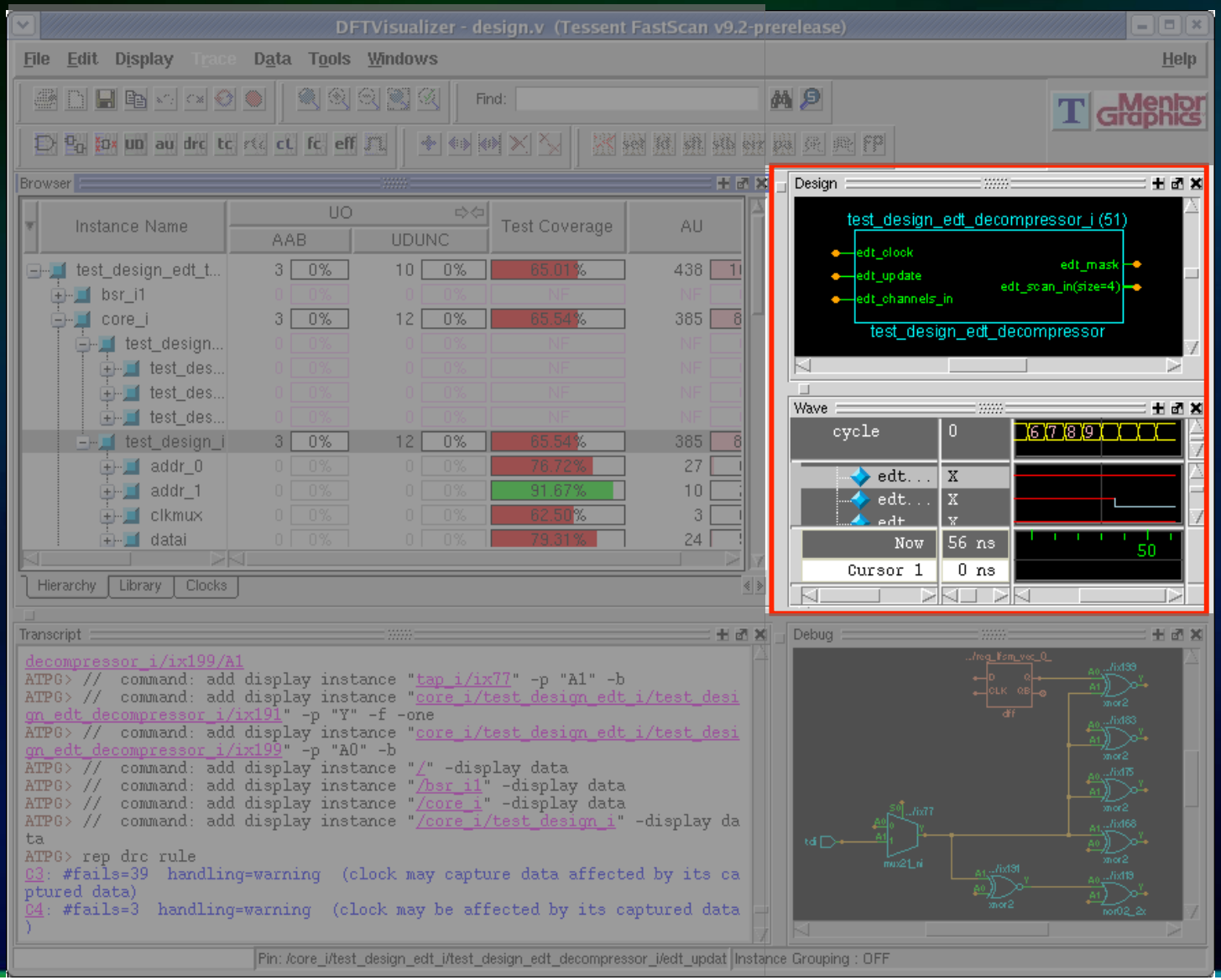
Gaurav Bansal, Roshni Lalwani
gaurav_bansal@mentor.com, roshni_lalwani@mentor.com

Gaurav Bansal
17'th Annual *Tcl/Tk* Conference (Tcl'2010)
October 11 - October 15, 2010

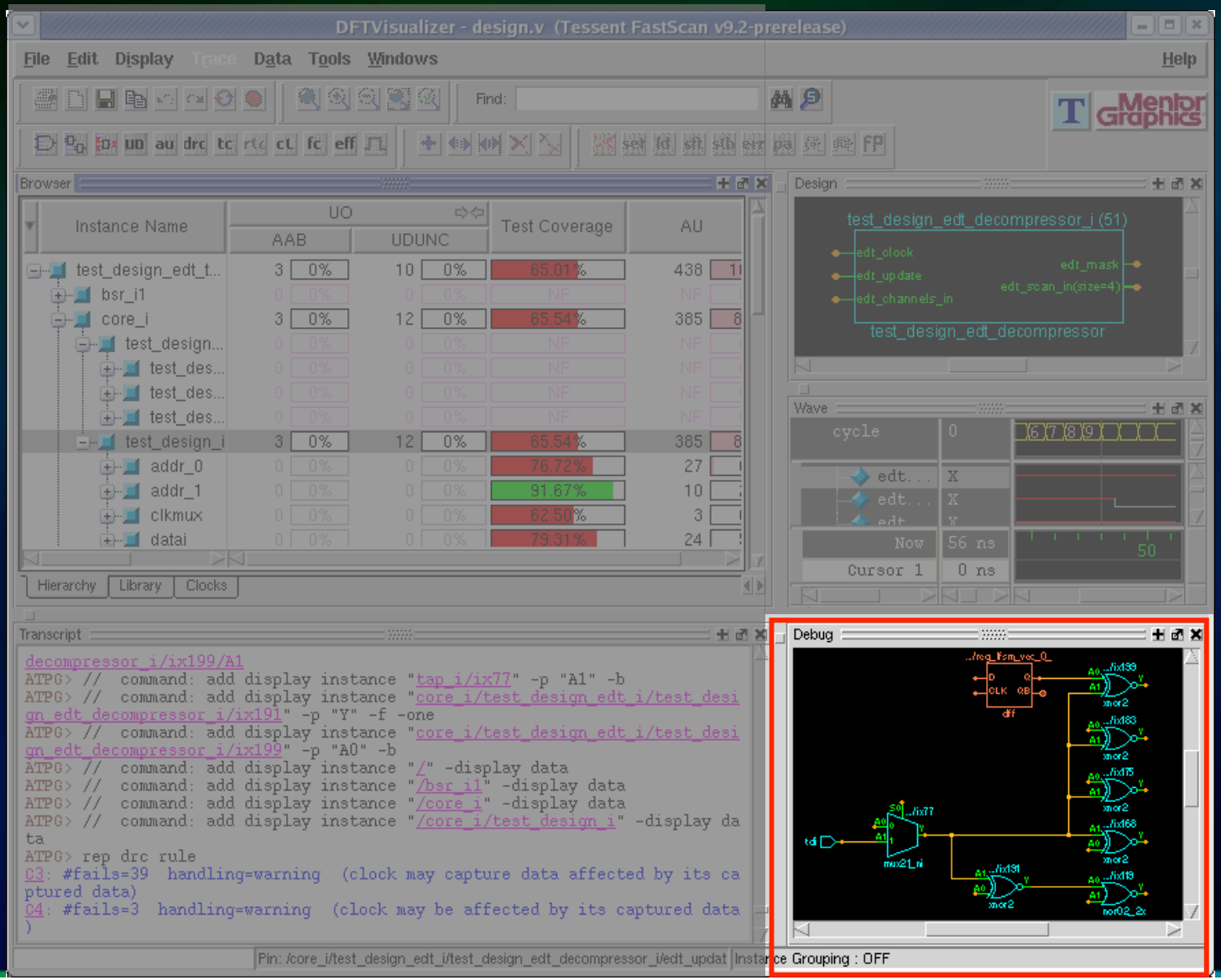
DFTVisualizer



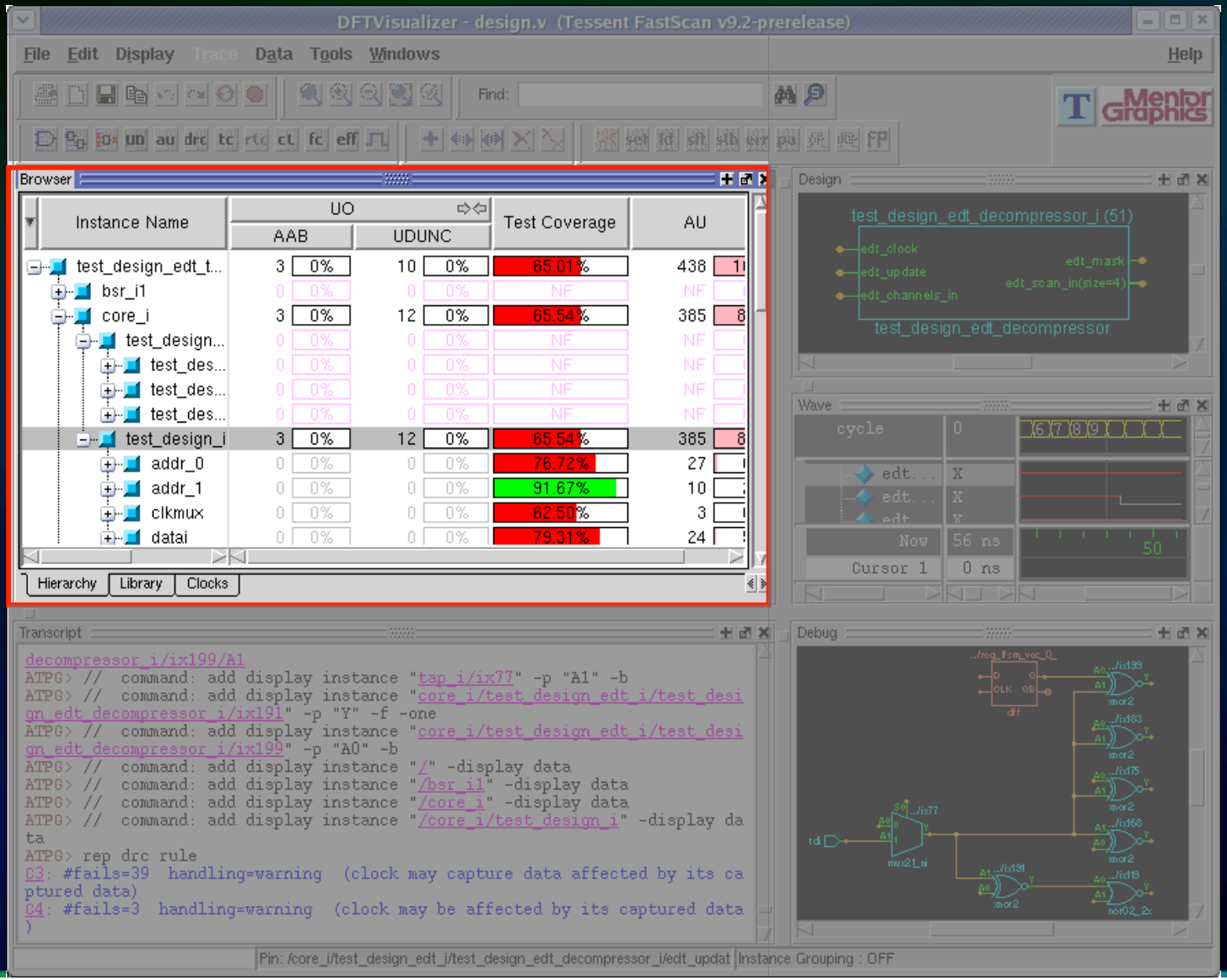
DFTVisualizer – Design and Wave Window



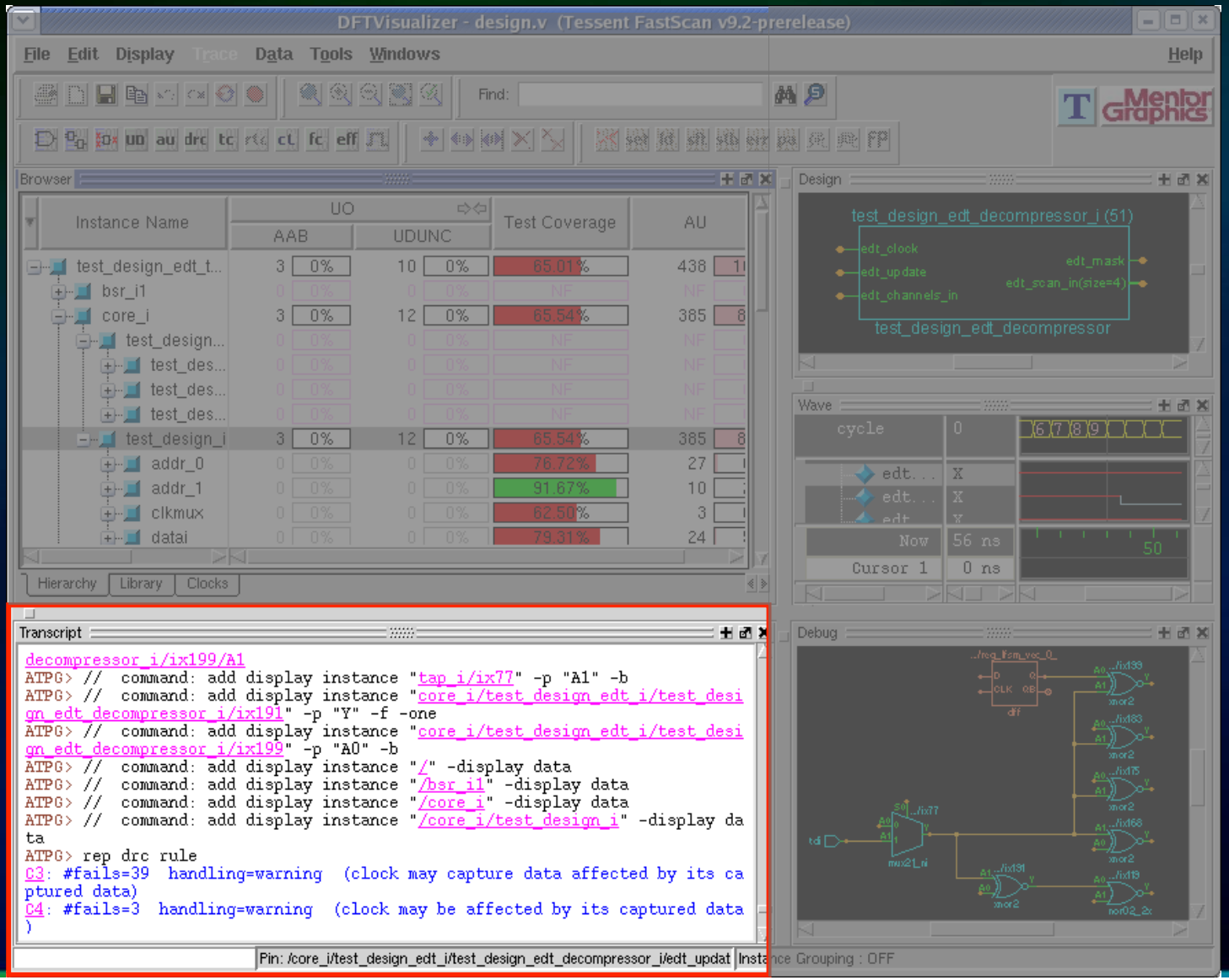
DFTVisualizer – Debug Window



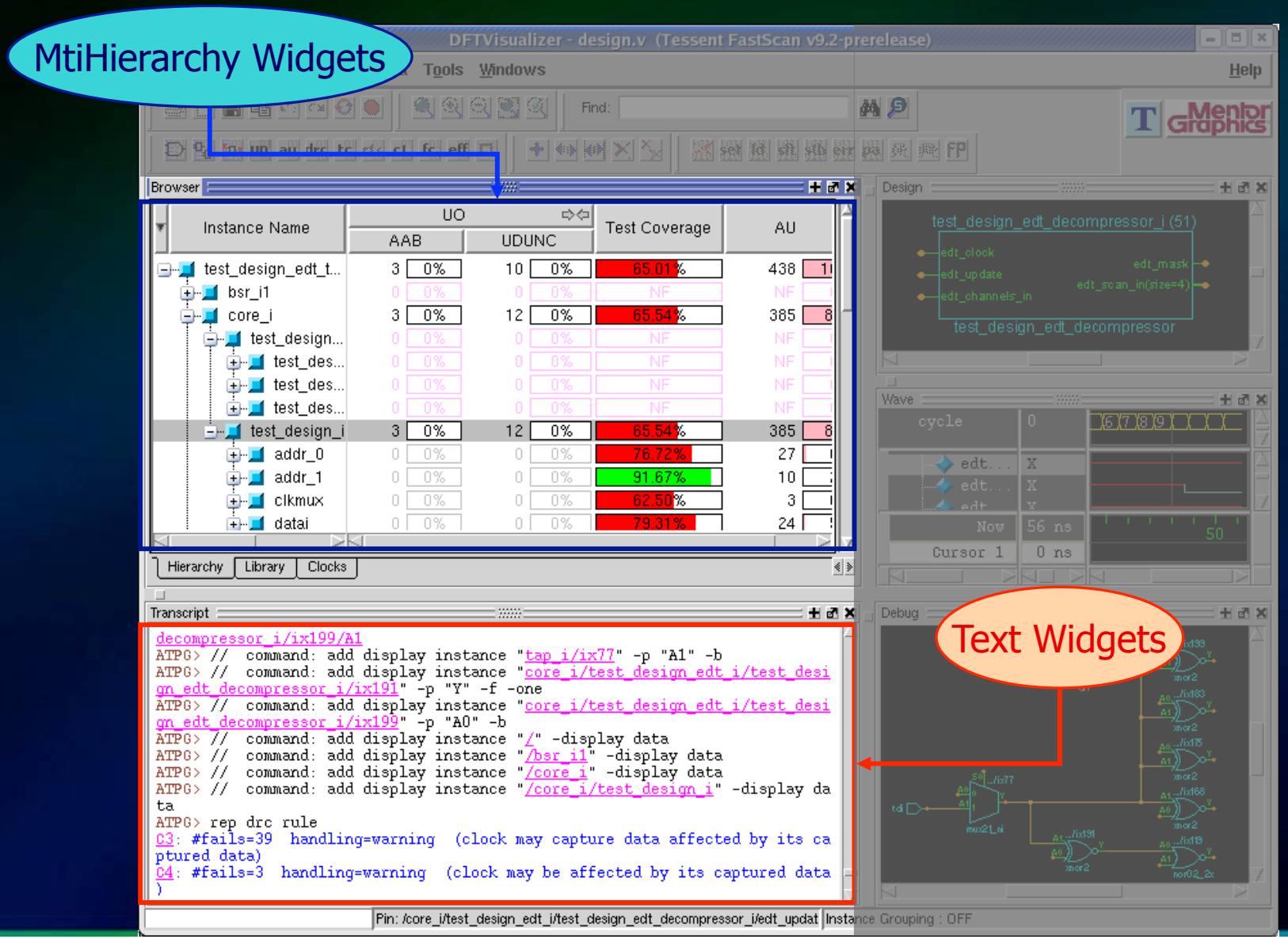
DFTVisualizer – Browser Window



DFTVisualizer – Transcript Window



DFTVisualizer



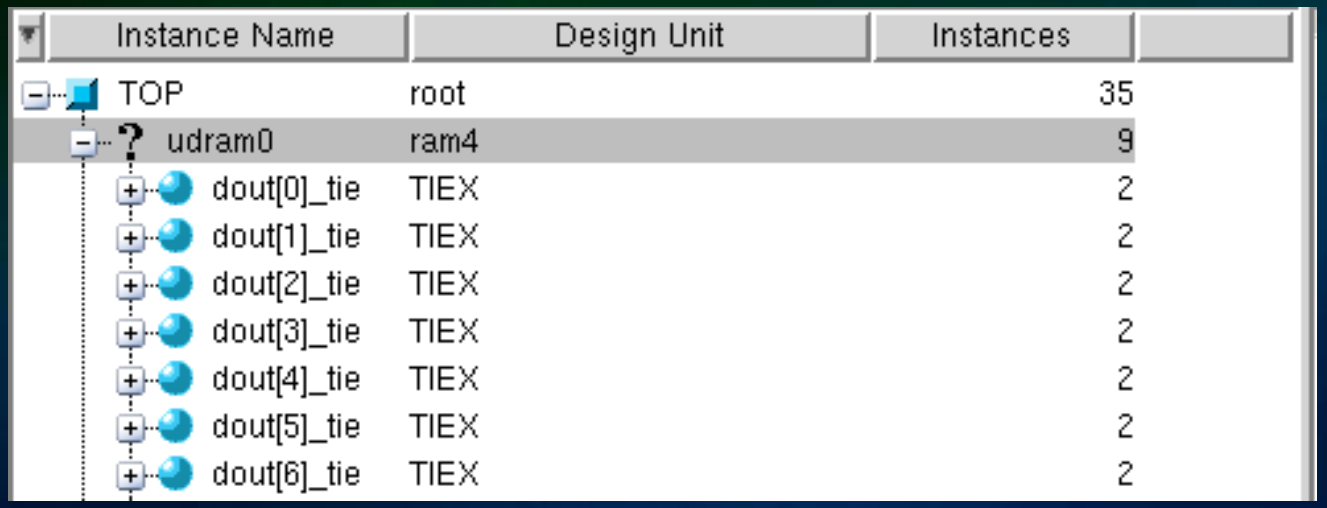
Agenda

- Enhancement in MtiHierarchy widget
  - Callback support for improved performance
  - Sub column support
  - Making tree column frozen in place (non-scrollable)
- Enhancement in Text widget
  - Hyperlinks
  - Incremental Parsing

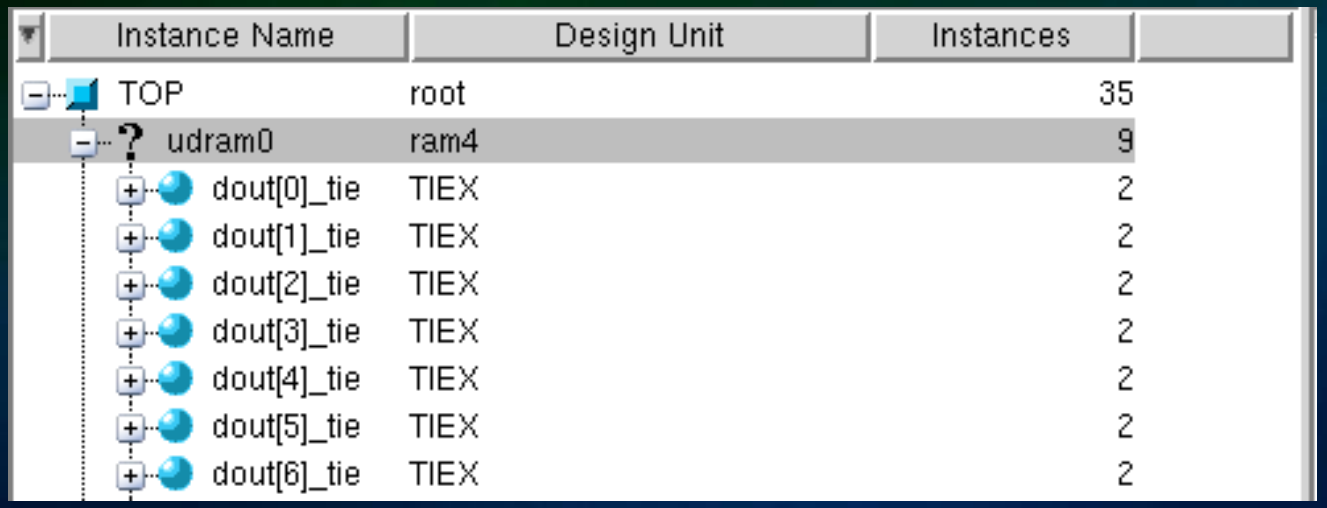
Agenda

- Enhancement in MtiHierarchy widget
  - Callback support for improved performance
  - Sub column support
  - Making tree column frozen in place (non-scrollable)
- Enhancement in Text widget
  - Hyperlinks
  - Incremental Parsing

Browser Window – MtiHierarchy Widget

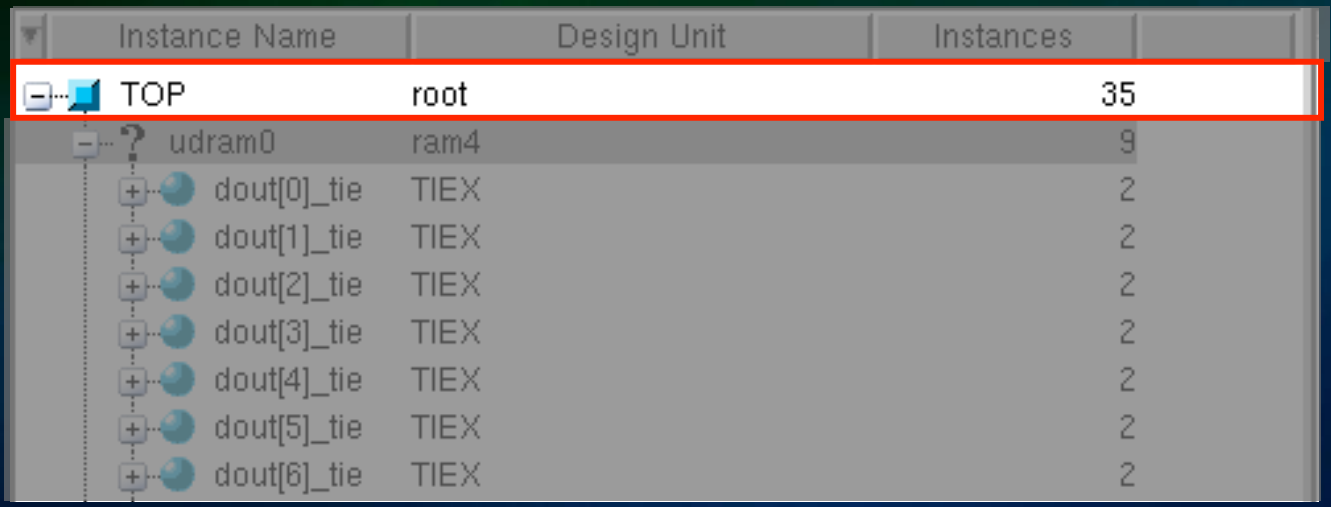


Browser Window – MtiHierarchy Widget



Format used to pass information to MtiHierarchy widget for each row is
<name> <app data> <label> <tag> <icon> <label> <tag> <icon>…

Browser Window – MtiHierarchy Widget



Format used to pass information to MtiHierarchy widget for each row is
<name> <app data> <label> <tag> <icon> <label> <tag> <icon>…

MtiHierarchy Widget – Callback Support

0xAAAAAA {HINST IN 4 -1} TOP branch vlog_inst

<name>

MtiHierarchy Widget – Callback Support

0xAAAAAA {HINST IN 4 -1} TOP branch vlog_inst

<name>

<app data>

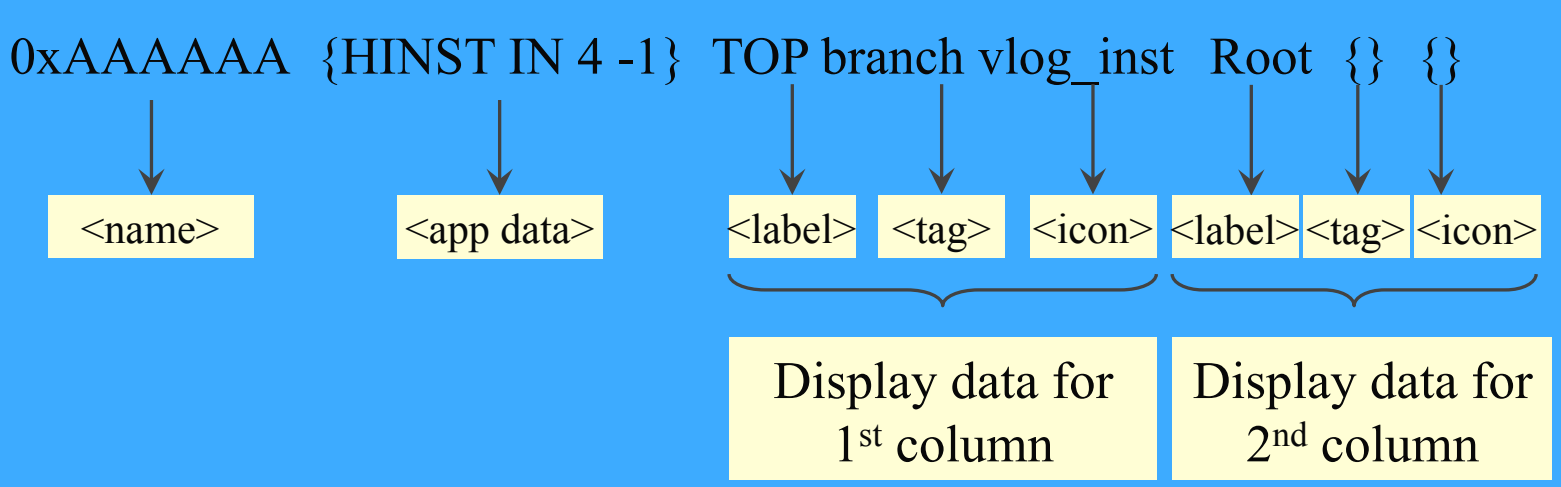
MtiHierarchy Widget – Callback Support

0xAAAAAA {HINST IN 4 -1} TOP branch vlog_inst

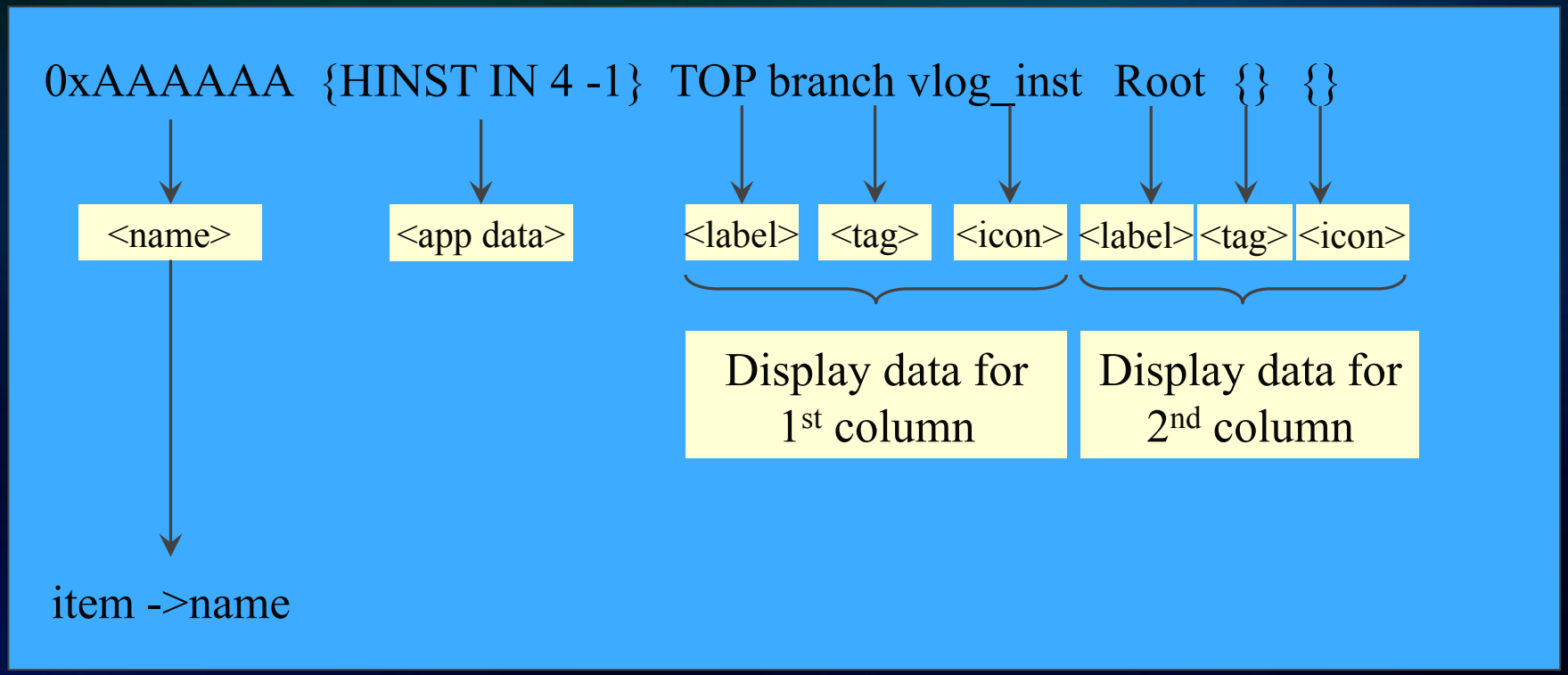
<name> <app data> <label> <tag> <icon>

Display data for 1st column

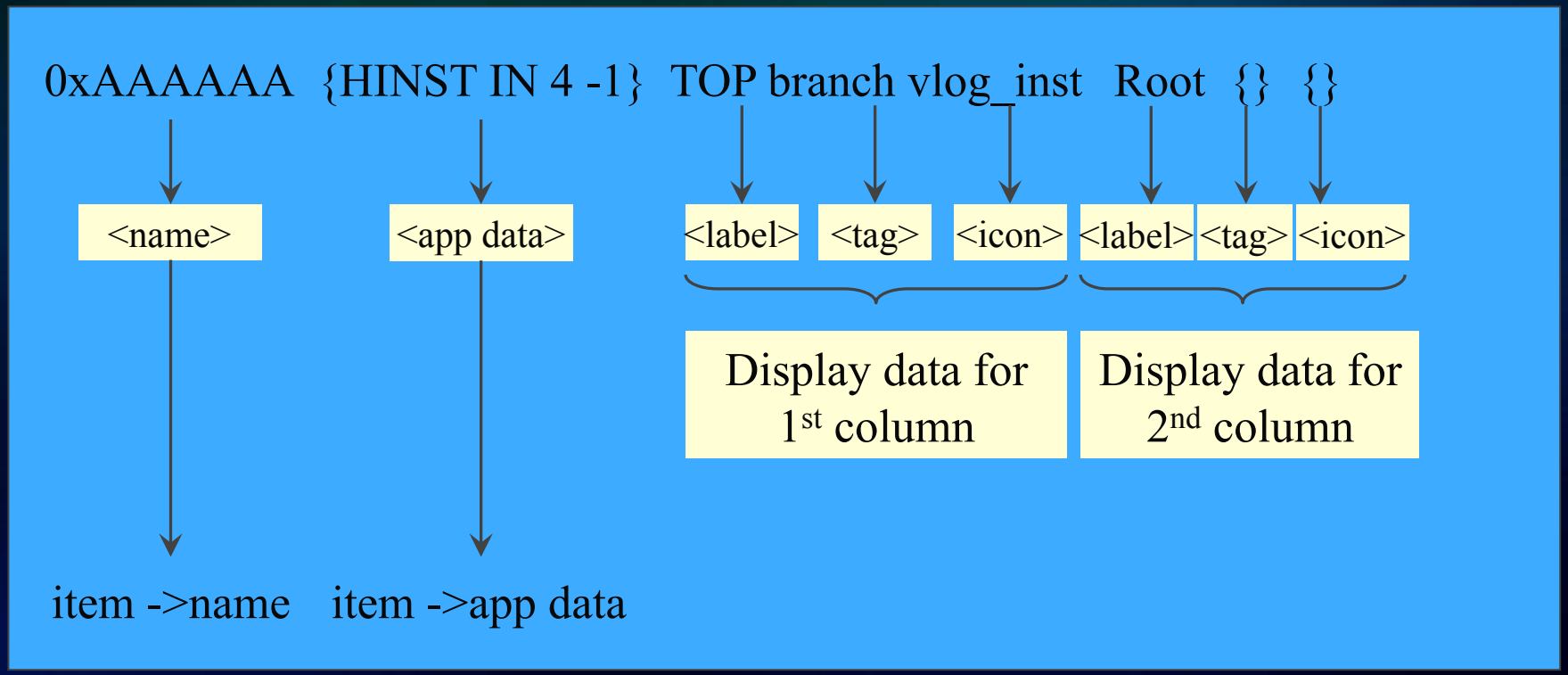
MtiHierarchy Widget – Callback Support



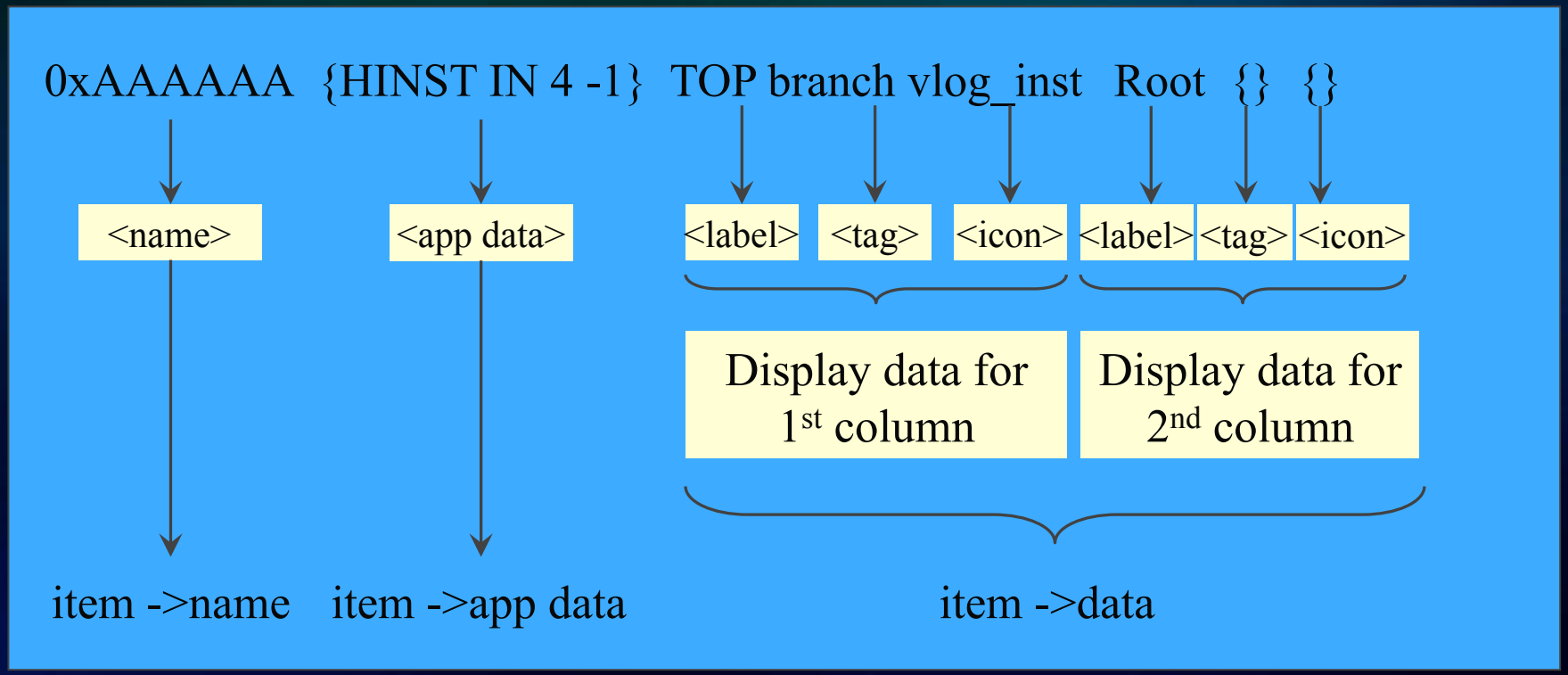
MtiHierarchy Widget – Callback Support



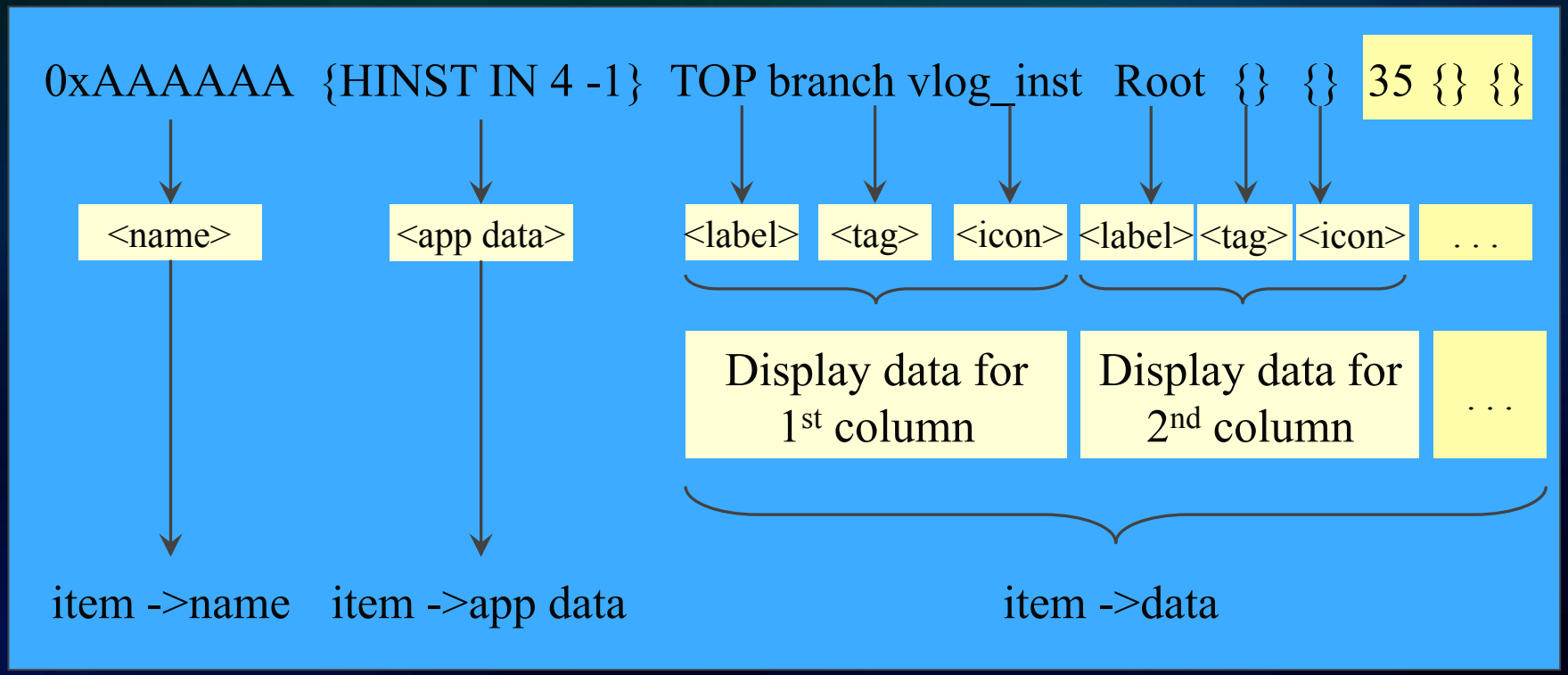
MtiHierarchy Widget – Callback Support



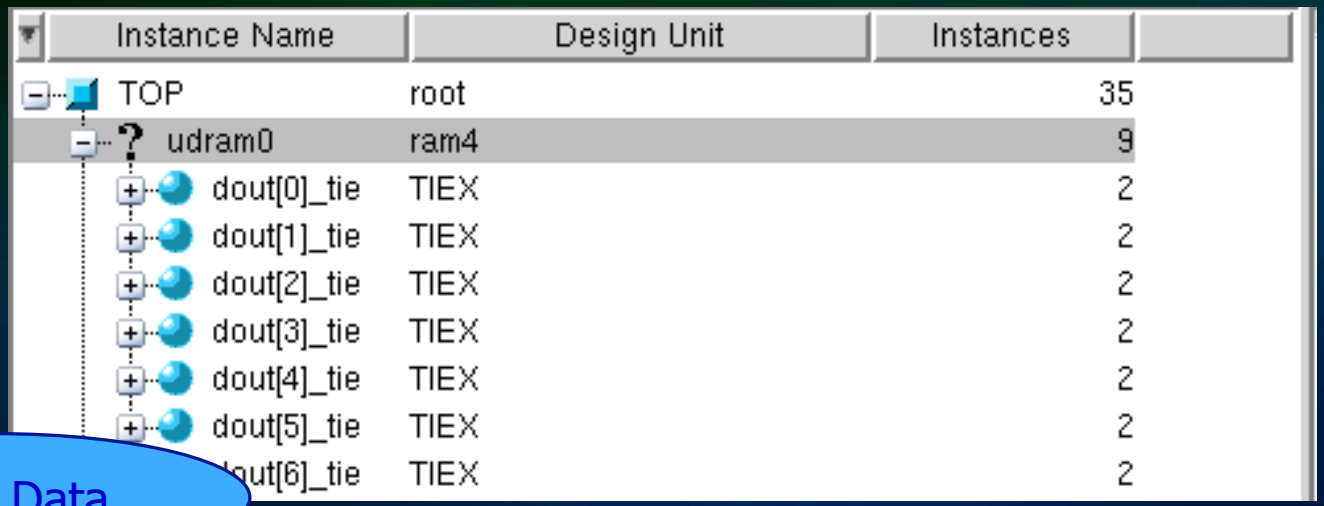
MtiHierarchy Widget – Callback Support



MtiHierarchy Widget – Callback Support



MtiHierarchy Widget – Callback Support



Static Data

<name1> <app data1> <label11> <tag11> <icon11> <label12> <tag12> <icon12>…

<name2> <app data2> <label21> <tag21> <icon21> <label22> <tag22> <icon22>…

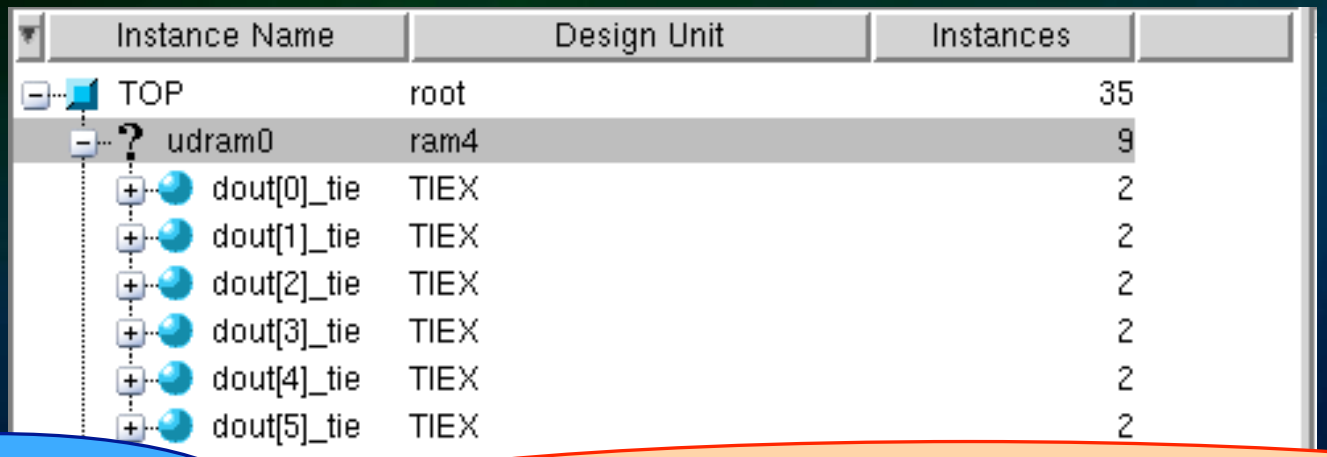
.

.

.

<nameN> <app dataN> <labelN1> <tagN1> <iconN1> <labelN2> <tagN2>

MtiHierarchy Widget – Callback Support



Static Data

Dynamic Data

<name1> <app data1> <label11> <tag11> <icon11> <label12> <tag12> <icon12>…

<name2> <app data2> <label21> <tag21> <icon21> <label22> <tag22> <icon22>…

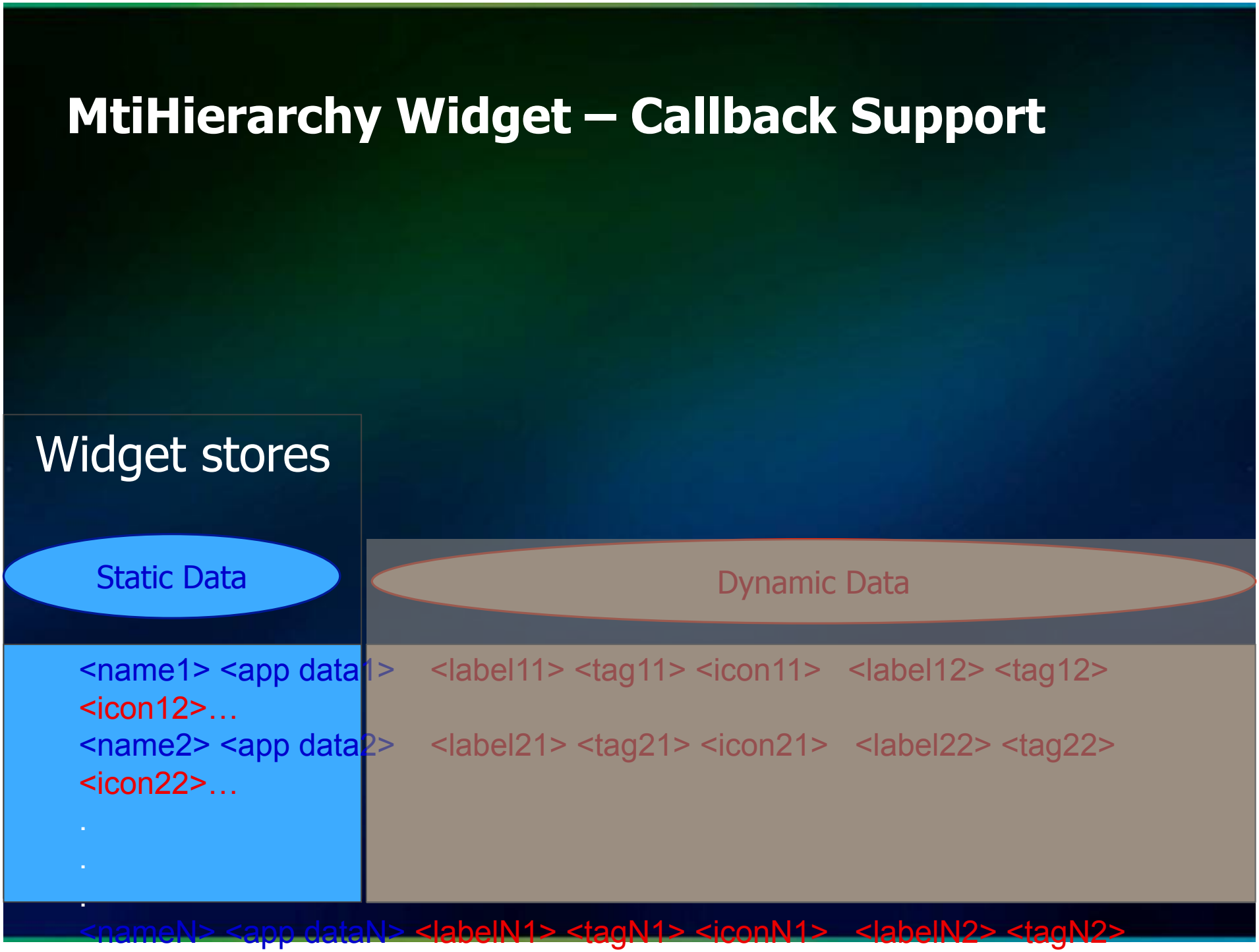
.

.

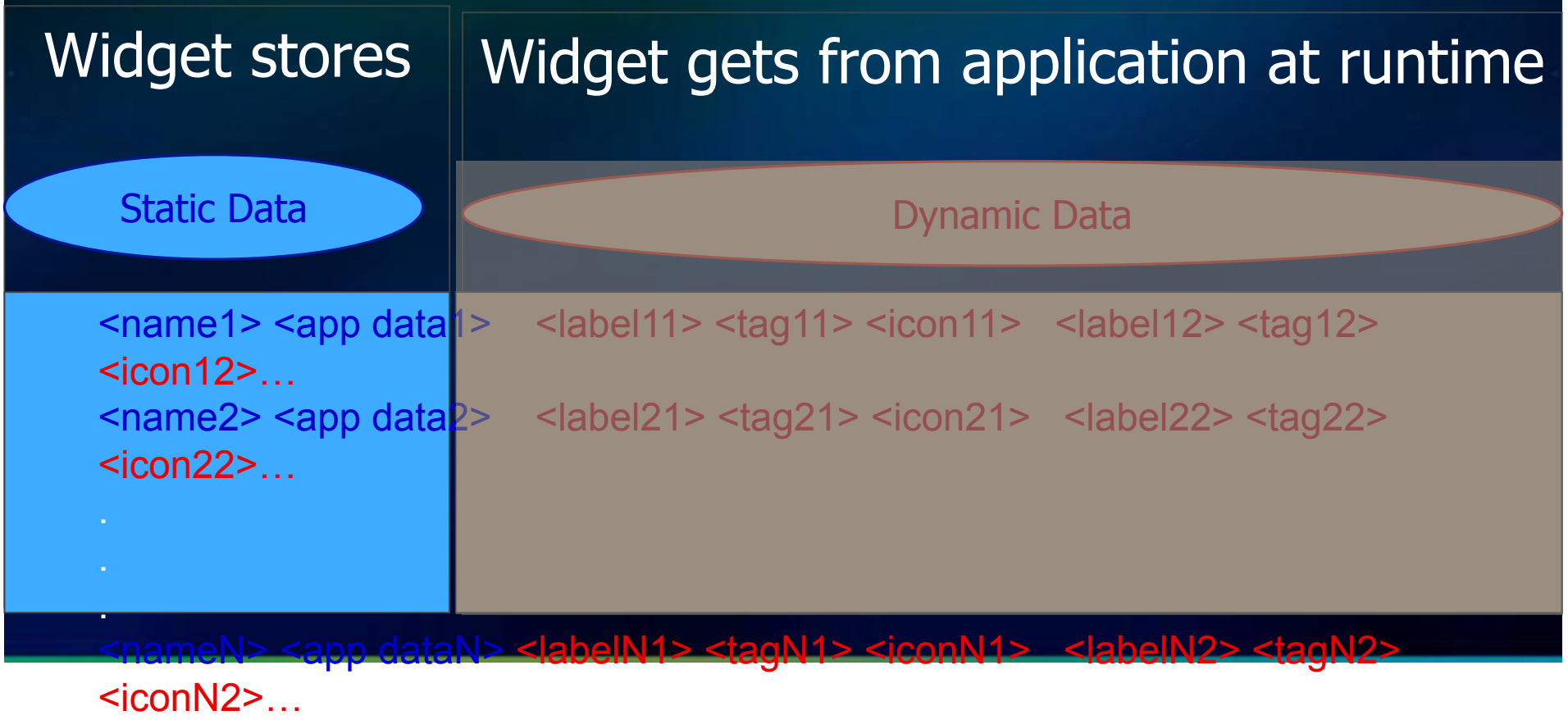
.

<nameN> <app dataN> <labelN1> <tagN1> <iconN1> <labelN2> <tagN2> <iconN2>…

MtiHierarchy Widget – Callback Support

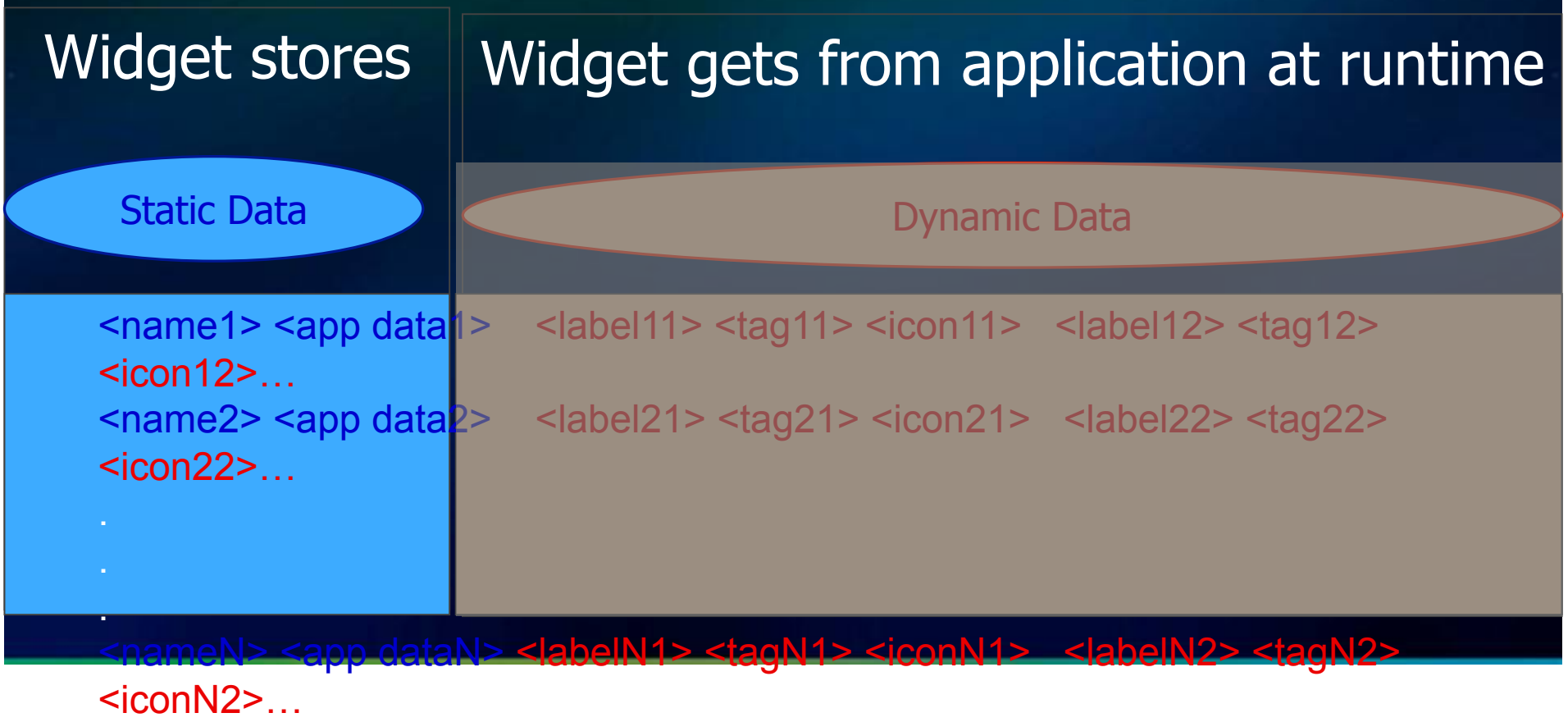


MtiHierarchy Widget – Callback Support



MtiHierarchy Widget – Callback Support

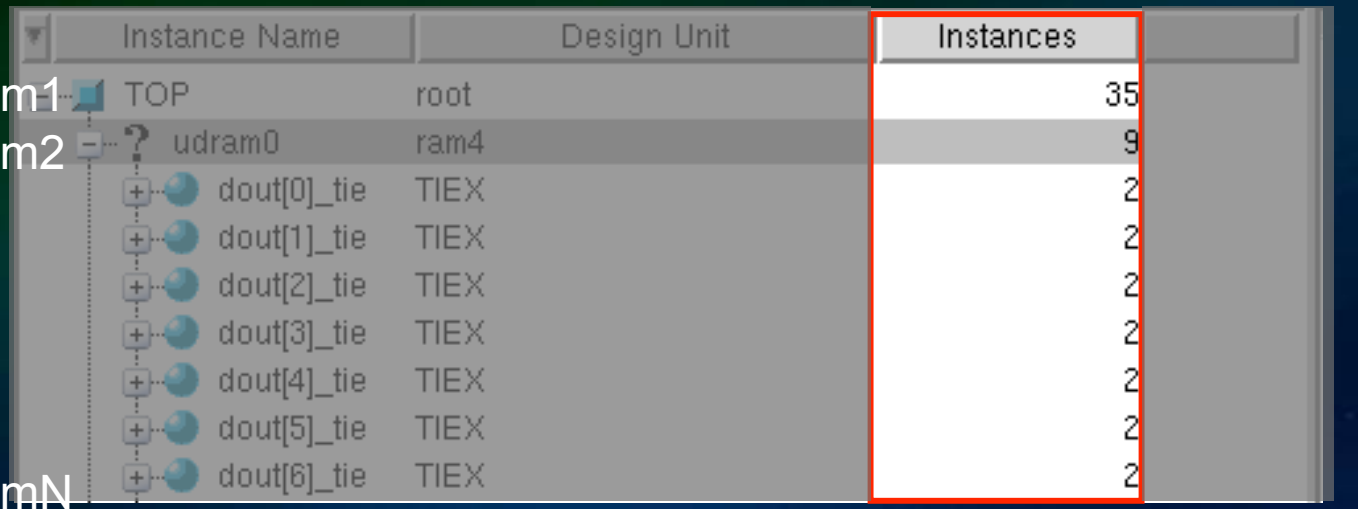
- Changes like addition/deletion of a column does not require any iterations by the widget
- Display data is provided by the application - takes fraction of a second



Side Effects

Side Effects – Sorting

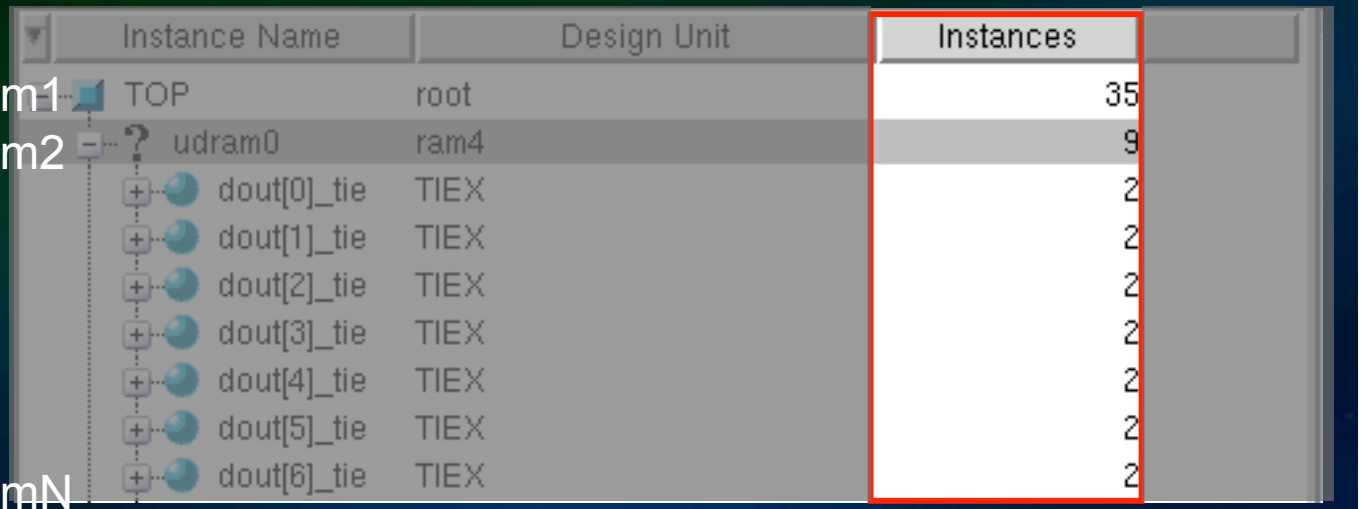
Item1
Item2
.
.
.
.
.
ItemN



Instance Name	Design Unit	Instances
TOP	root	35
udram0	ram4	9
dout[0]_tie	TIEX	2
dout[1]_tie	TIEX	2
dout[2]_tie	TIEX	2
dout[3]_tie	TIEX	2
dout[4]_tie	TIEX	2
dout[5]_tie	TIEX	2
dout[6]_tie	TIEX	2

Side Effects – Sorting

Item1
Item2
.
.
.
.
.
ItemN



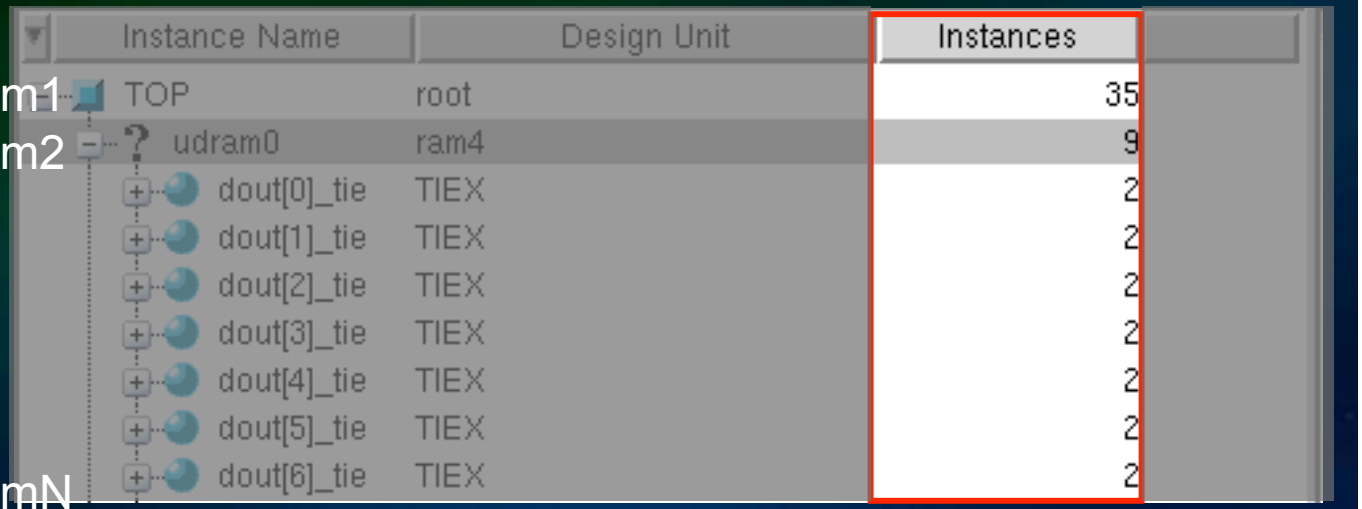
Static Data

Dynamic Data

<name1> <app data1> <label11> <tag11> <icon11> <label12> <tag12> <icon12>…
<name2> <app data2> <label21> <tag21> <icon21> <label22> <tag22> <icon22>…
.
.
.
<nameN> <app dataN> <labelN1> <tagN1> <iconN1> <labelN2> <tagN2> <iconN2>

Side Effects – Sorting

Item1
Item2
.
.
.
.
.
ItemN



Static Data

Dynamic Data

<name1> <app data1> <label11> <tag11> <icon11> <label12> <tag12> <icon12>…
<name2> <app data2> <label21> <tag21> <icon21> <label22> <tag22> <icon22>…
.
.

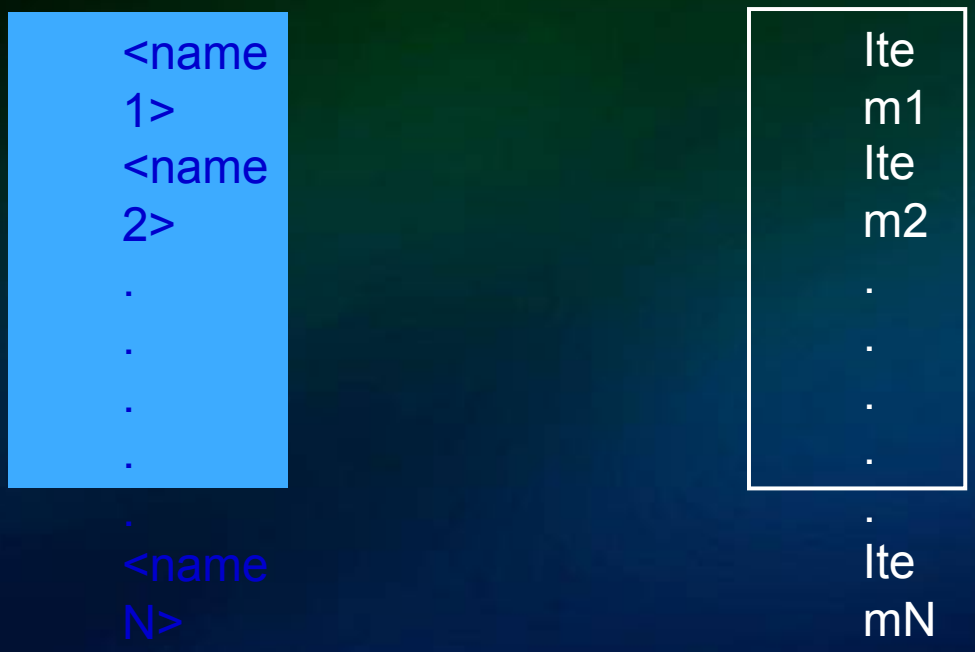
<nameN> <app dataN> <labelN1> <tagN1> <iconN1> <labelN2> <tagN2> <iconN2>

Side Effects – Sorting

<name 1>
<name 2>
.
.
.
.
.
<name N>

- Application gives us sorted list of "<name>" (item->name pointers)

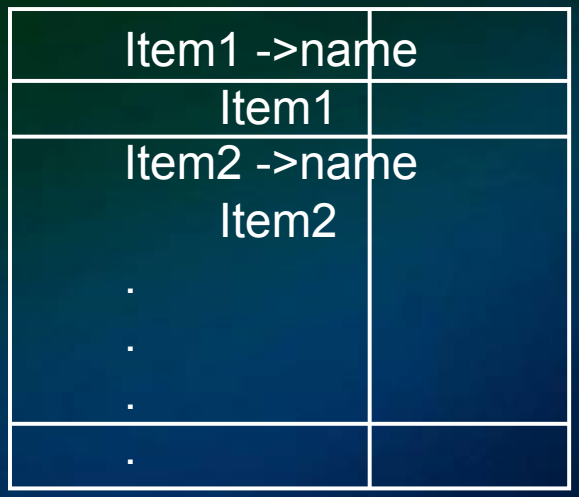
Side Effects – Sorting



- Application gives us sorted list of "<name>" (item->name pointers)
- We need the list of sorted "item" pointers.

Side Effects – Sorting

<name 1>
<name 2>
.
.
.
.
.
<name N>



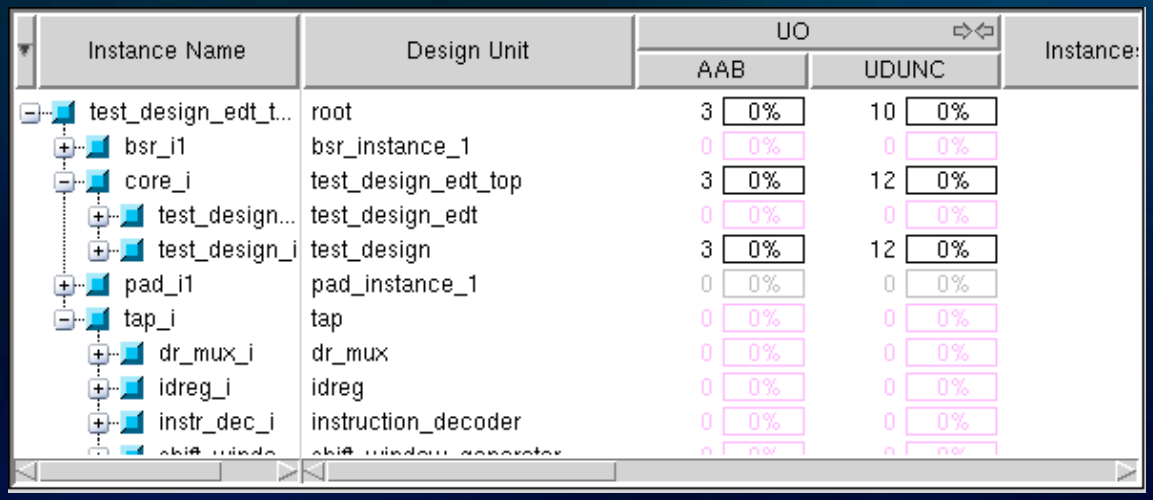
- Application gives us sorted list of "<name>" (item->name pointers)
- We need the list of sorted "item" pointers.
- So we create a hash table for item->name and item pointer

Agenda

- Enhancement in MtiHierarchy widget
  - Callback support for improved performance
  - Sub column support
  - Making tree column frozen in place (non-scrollable)
- Enhancement in Text widget
  - Hyperlinks
  - Incremental Parsing

Sub-columns

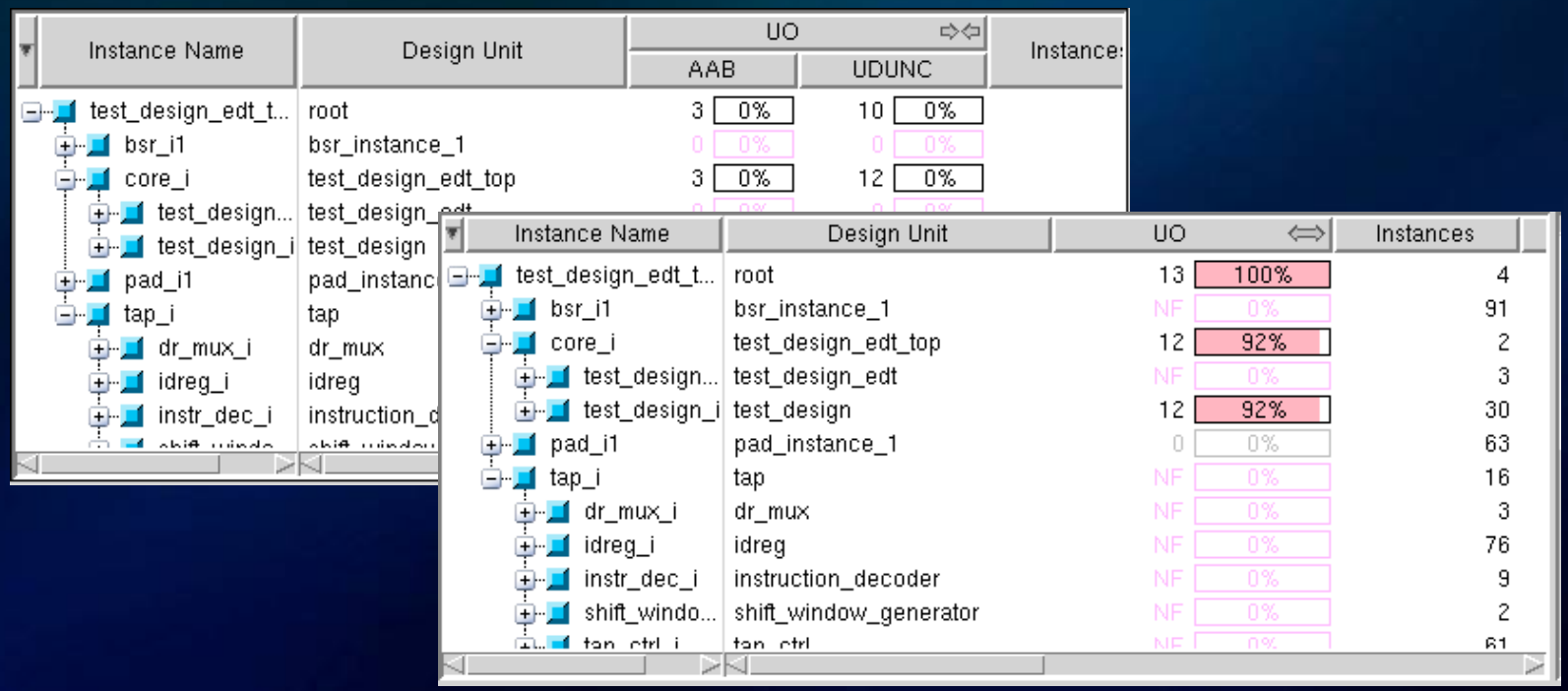
- Show the fault distribution of instances in various categories and sub-categories



Instance Name	Design Unit	UO				Instance
		AAB		UDUNC		
test_design_edt_t...	root	3	0%	10	0%	
bsr_i1	bsr_instance_1	0	0%	0	0%	
core_i	test_design_edt_top	3	0%	12	0%	
test_design...	test_design_edt	0	0%	0	0%	
test_design_i	test_design	3	0%	12	0%	
pad_i1	pad_instance_1	0	0%	0	0%	
tap_i	tap	0	0%	0	0%	
dr_mux_i	dr_mux	0	0%	0	0%	
idreg_i	idreg	0	0%	0	0%	
instr_dec_i	instruction_decoder	0	0%	0	0%	

Sub-columns

- Show the fault distribution of instances in various categories and sub-categories
- Categories and sub-categories should be displayed as in-place expandable/collapsible column.



Sub-columns

- Overview of changes
  - Enhanced MtiHeirarchy widget protocol to support the sub-column labels as a list of labels instead of column label

    <name> <appdata> <sub-label1 sub-label2 sub-label3> <tag> <icon>

Sub-columns

- Overview of changes
  - Enhanced MtiHeirarchy widget protocol to support the sub-column labels as a list of labels instead of column label

    <name> <appdata> <sub-label1 sub-label2 sub-label3> <tag> <icon>

  - Provide APIs for sub-column operations like add, delete, configure, move, etc

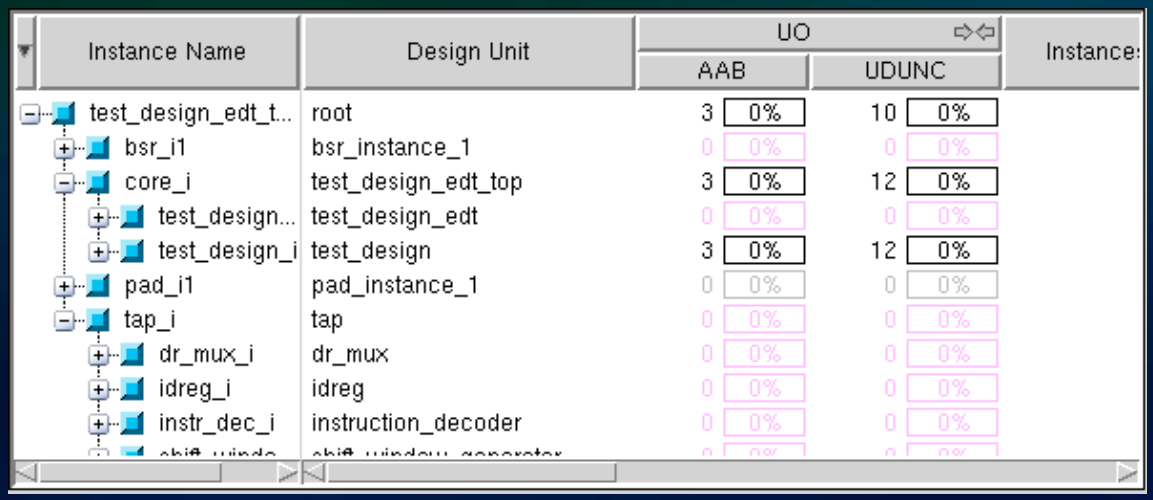
Sub-columns

- Overview of changes
  - Enhanced MtiHeirarchy widget protocol to support the sub-column labels as a list of labels instead of column label

    <name> <appdata> <sub-label1 sub-label2 sub-label3> <tag> <icon>

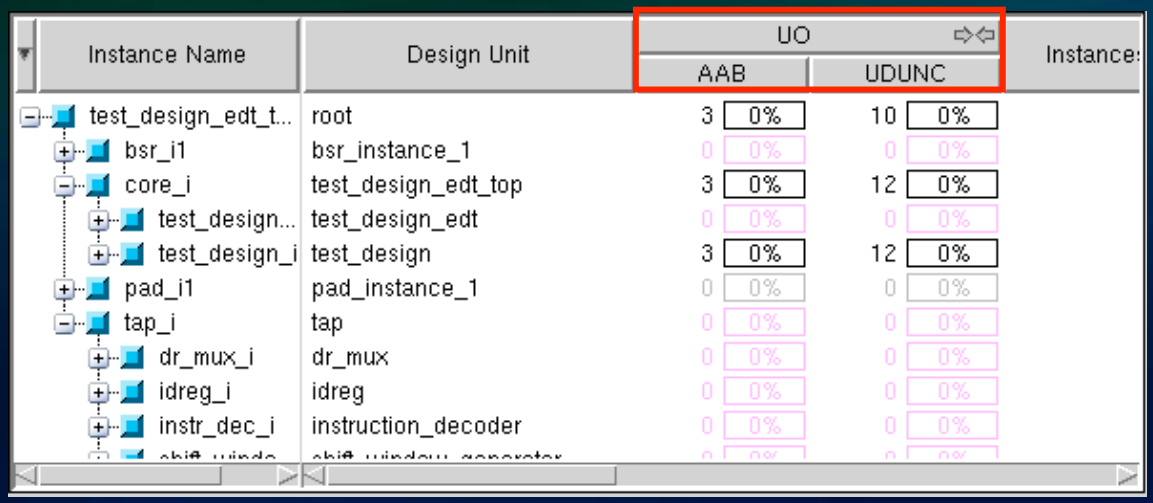
  - Provide APIs for sub-column operations like add, delete, configure, move, etc
  - Expandable/ collapsible we need to provide a clickable icon

Sub-columns – Approach



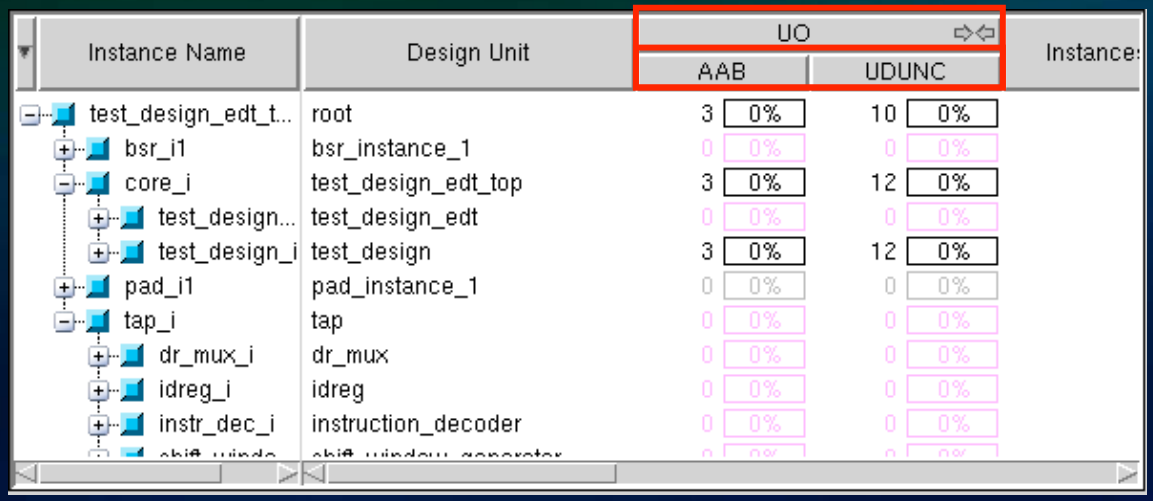
Instance Name	Design Unit	UO				Instance
		AAB		UDUNC		
test_design_edt_t...	root	3	0%	10	0%	
bsr_i1	bsr_instance_1	0	0%	0	0%	
core_i	test_design_edt_top	3	0%	12	0%	
test_design...	test_design_edt	0	0%	0	0%	
test_design_i	test_design	3	0%	12	0%	
pad_i1	pad_instance_1	0	0%	0	0%	
tap_i	tap	0	0%	0	0%	
dr_mux_i	dr_mux	0	0%	0	0%	
idreg_i	idreg	0	0%	0	0%	
instr_dec_i	instruction_decoder	0	0%	0	0%	

Sub-columns – Approach



Instance Name	Design Unit	UO				Instance
		AAB		UDUNC		
test_design_edt_t...	root	3	0%	10	0%	
bsr_i1	bsr_instance_1	0	0%	0	0%	
core_i	test_design_edt_top	3	0%	12	0%	
test_design...	test_design_edt	0	0%	0	0%	
test_design_i	test_design	3	0%	12	0%	
pad_i1	pad_instance_1	0	0%	0	0%	
tap_i	tap	0	0%	0	0%	
dr_mux_i	dr_mux	0	0%	0	0%	
idreg_i	idreg	0	0%	0	0%	
instr_dec_i	instruction_decoder	0	0%	0	0%	

Sub-columns – Approach

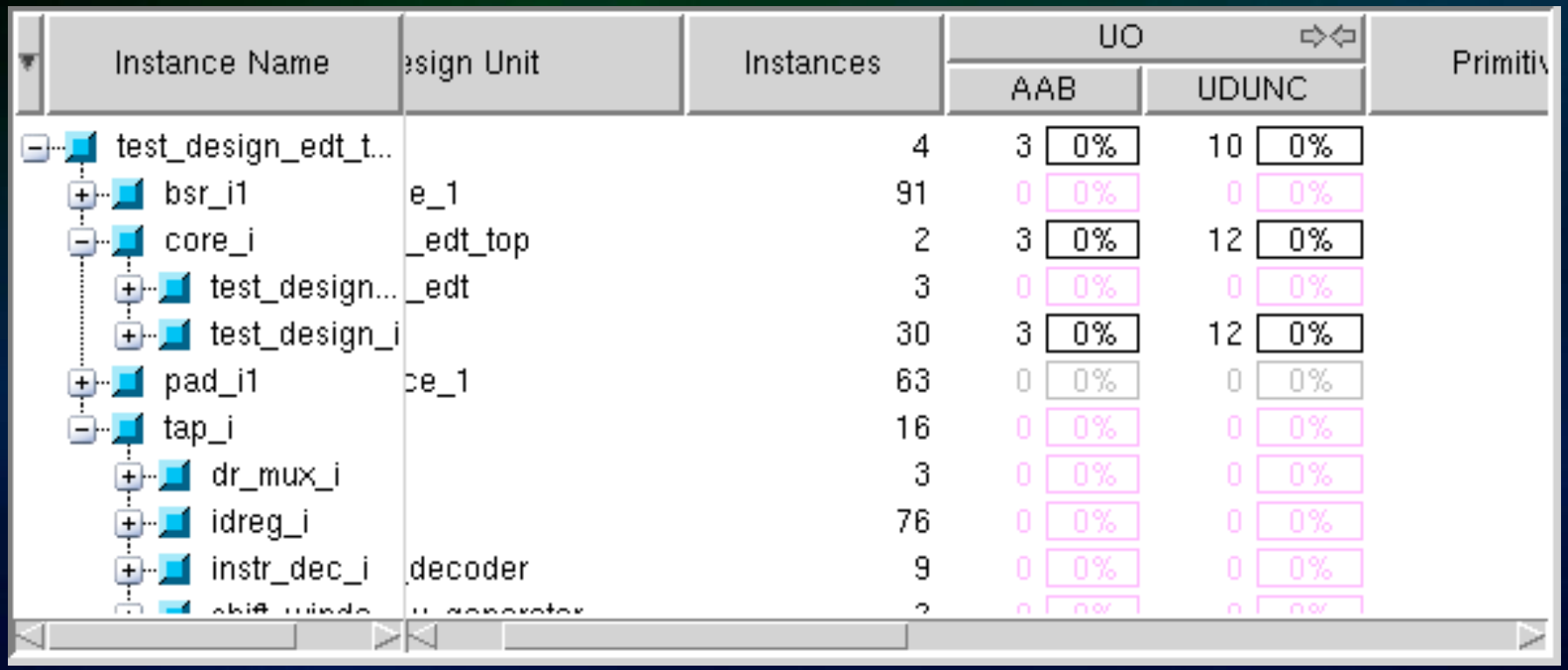


Instance Name	Design Unit	UO				Instance
		AAB		UDUNC		
test_design_edt_t...	root	3	0%	10	0%	
bsr_i1	bsr_instance_1	0	0%	0	0%	
core_i	test_design_edt_top	3	0%	12	0%	
test_design...	test_design_edt	0	0%	0	0%	
test_design_i	test_design	3	0%	12	0%	
pad_i1	pad_instance_1	0	0%	0	0%	
tap_i	tap	0	0%	0	0%	
dr_mux_i	dr_mux	0	0%	0	0%	
idreg_i	idreg	0	0%	0	0%	
instr_dec_i	instruction_decoder	0	0%	0	0%	

Agenda

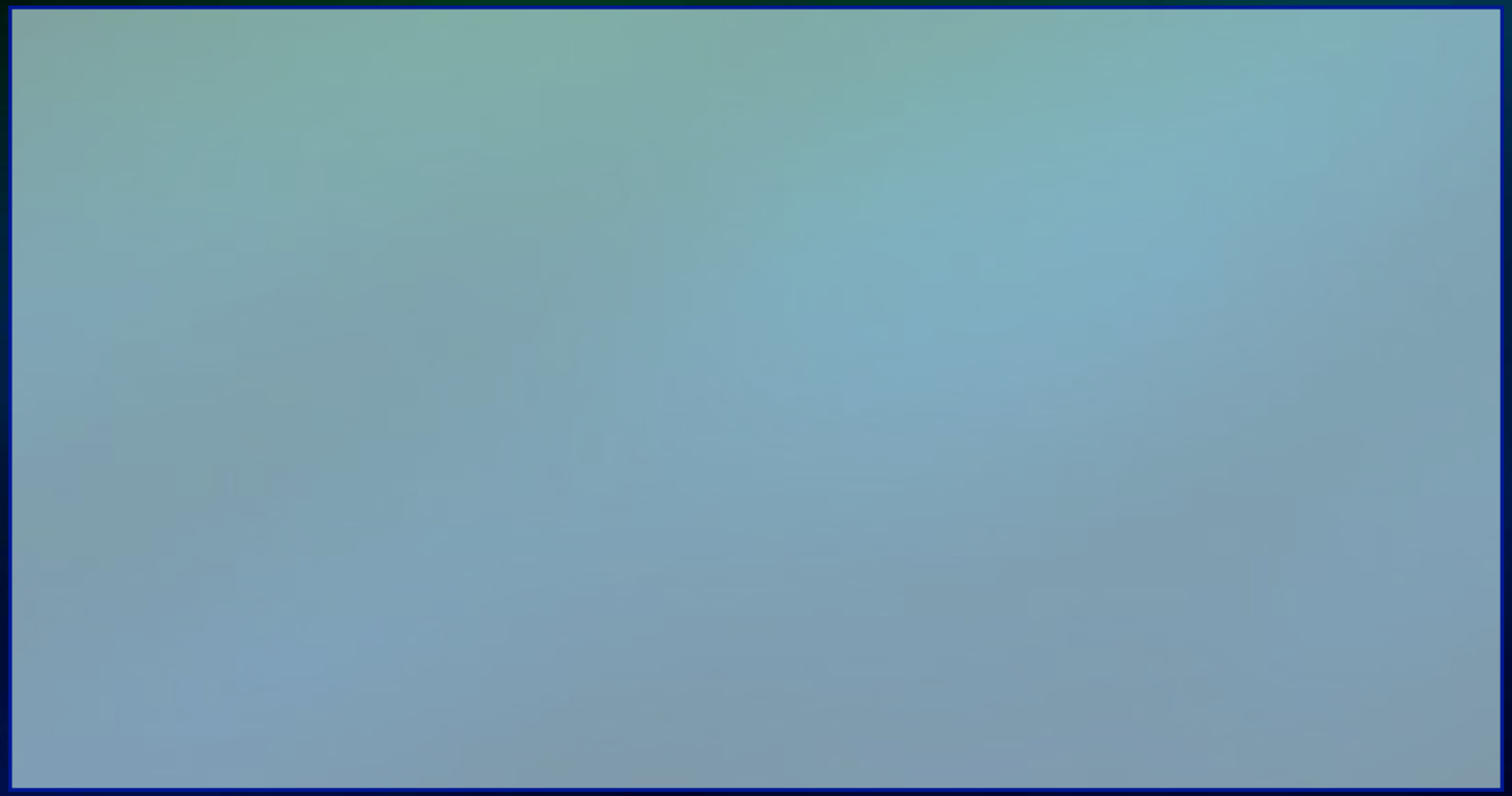
- Enhancement in MtiHierarchy widget
  - Callback support for improved performance
  - Sub column support
  - Making tree column frozen in place (non-scrollable)
- Enhancement in Text widget
  - Hyperlinks
  - Incremental Parsing

Frozen Tree Column



Frozen Tree Column

- Clipper (a frame)

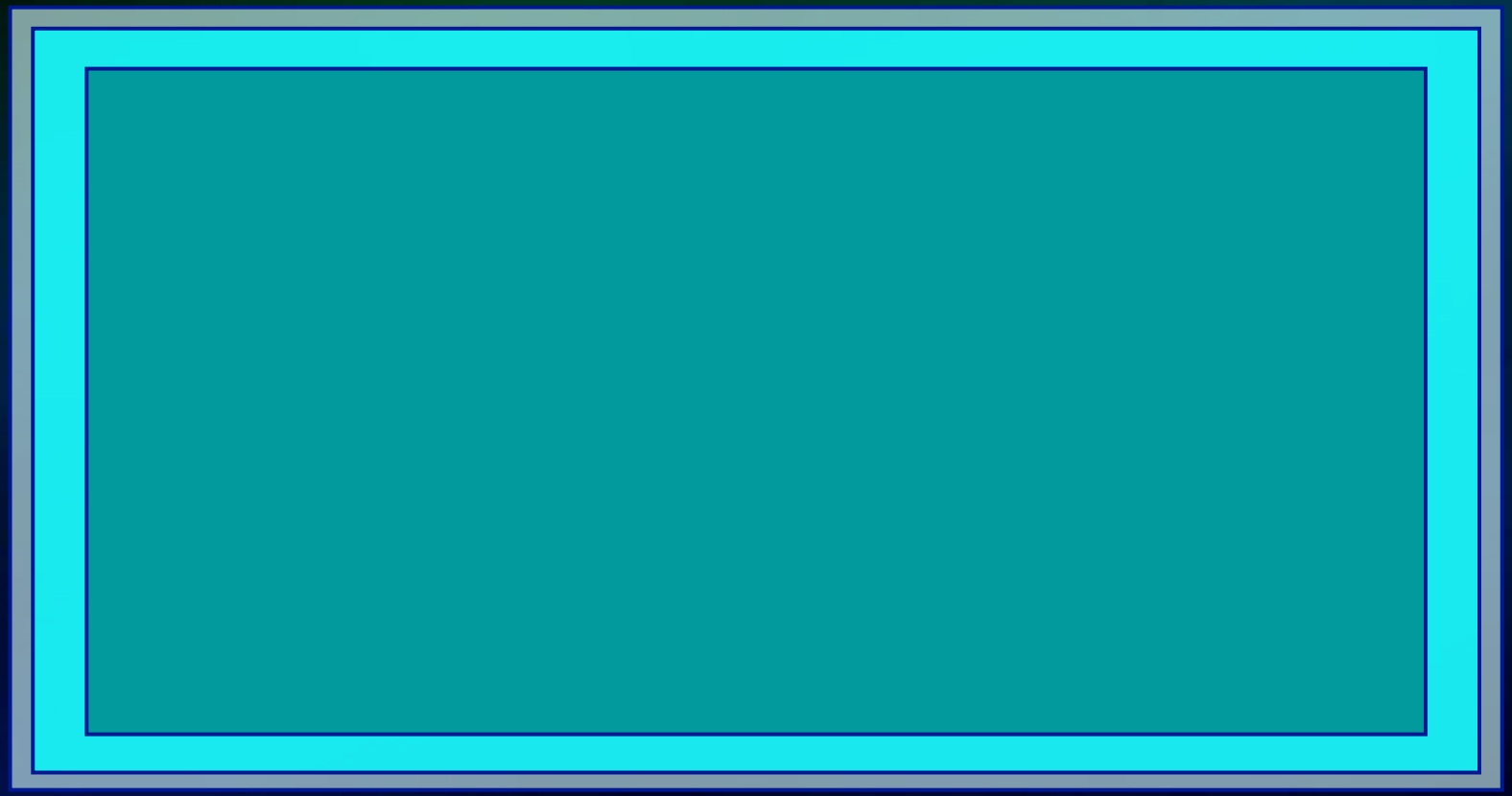


Frozen Tree Column

- Clipper (a frame) → Canvas

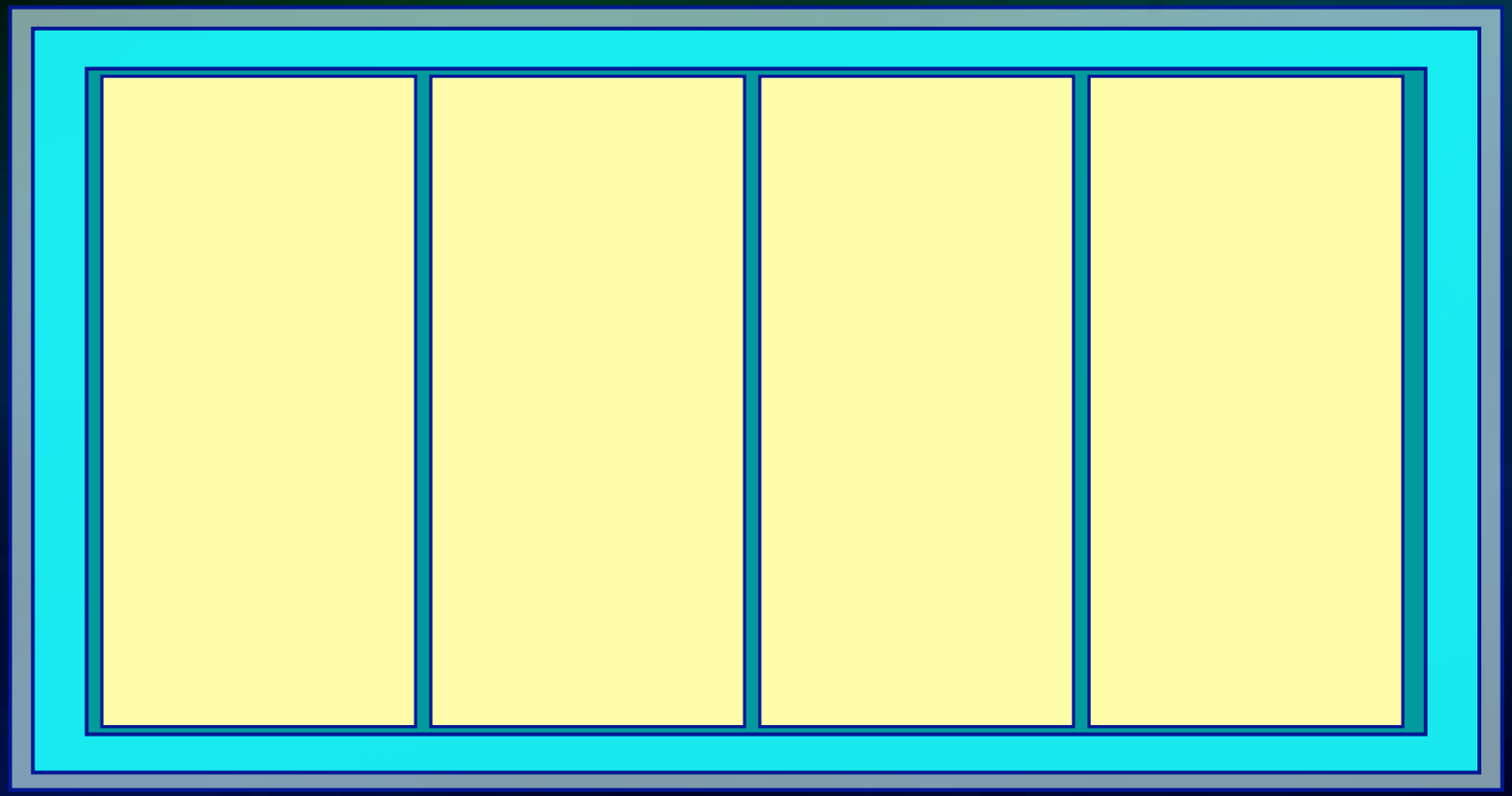
Frozen Tree Column

- Clipper (a frame) → Canvas → Sfchildsite (a frame)



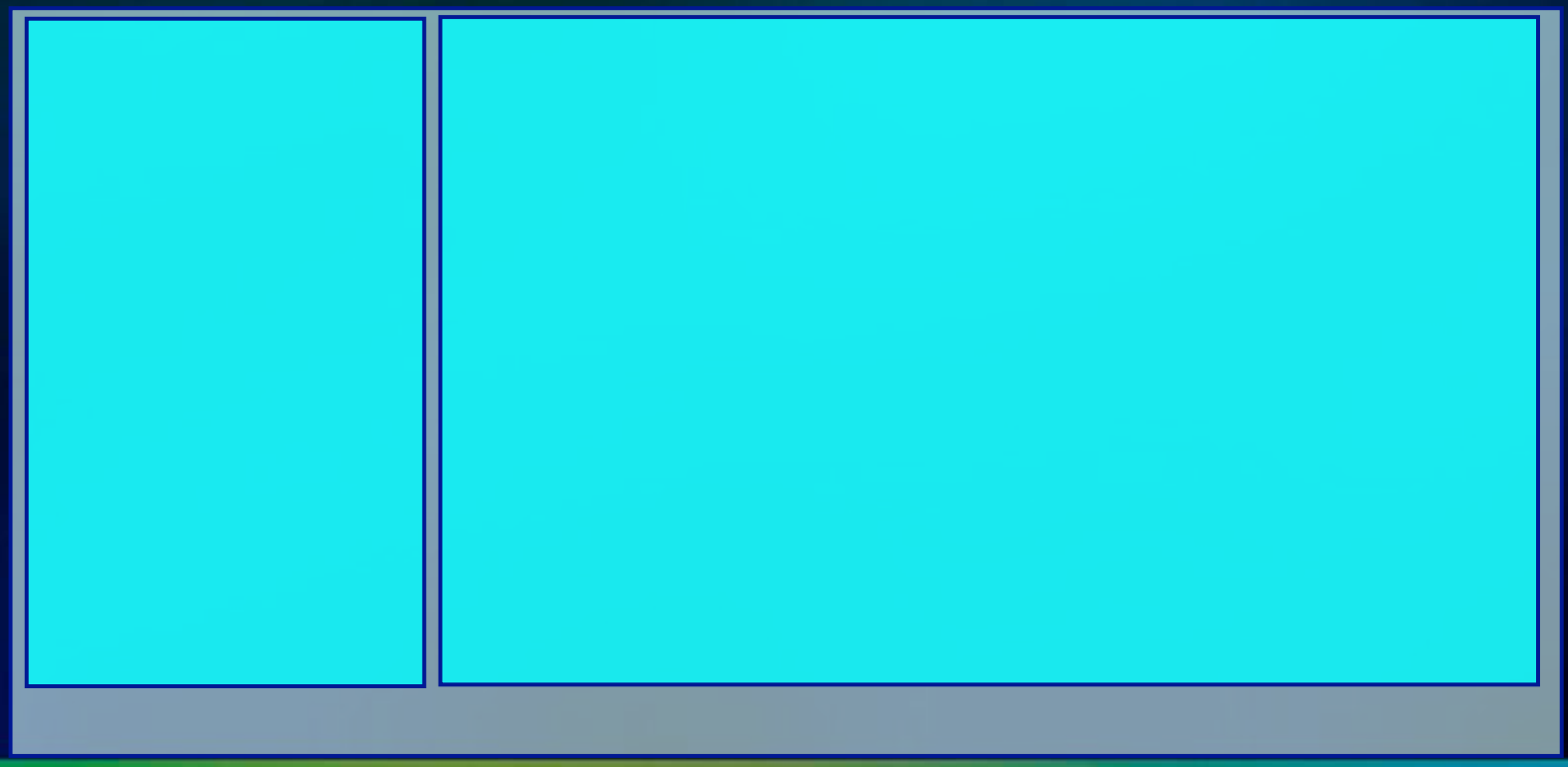
Frozen Tree Column

- Clipper (a frame) → Canvas → Sfchildsite (a frame) → Columns



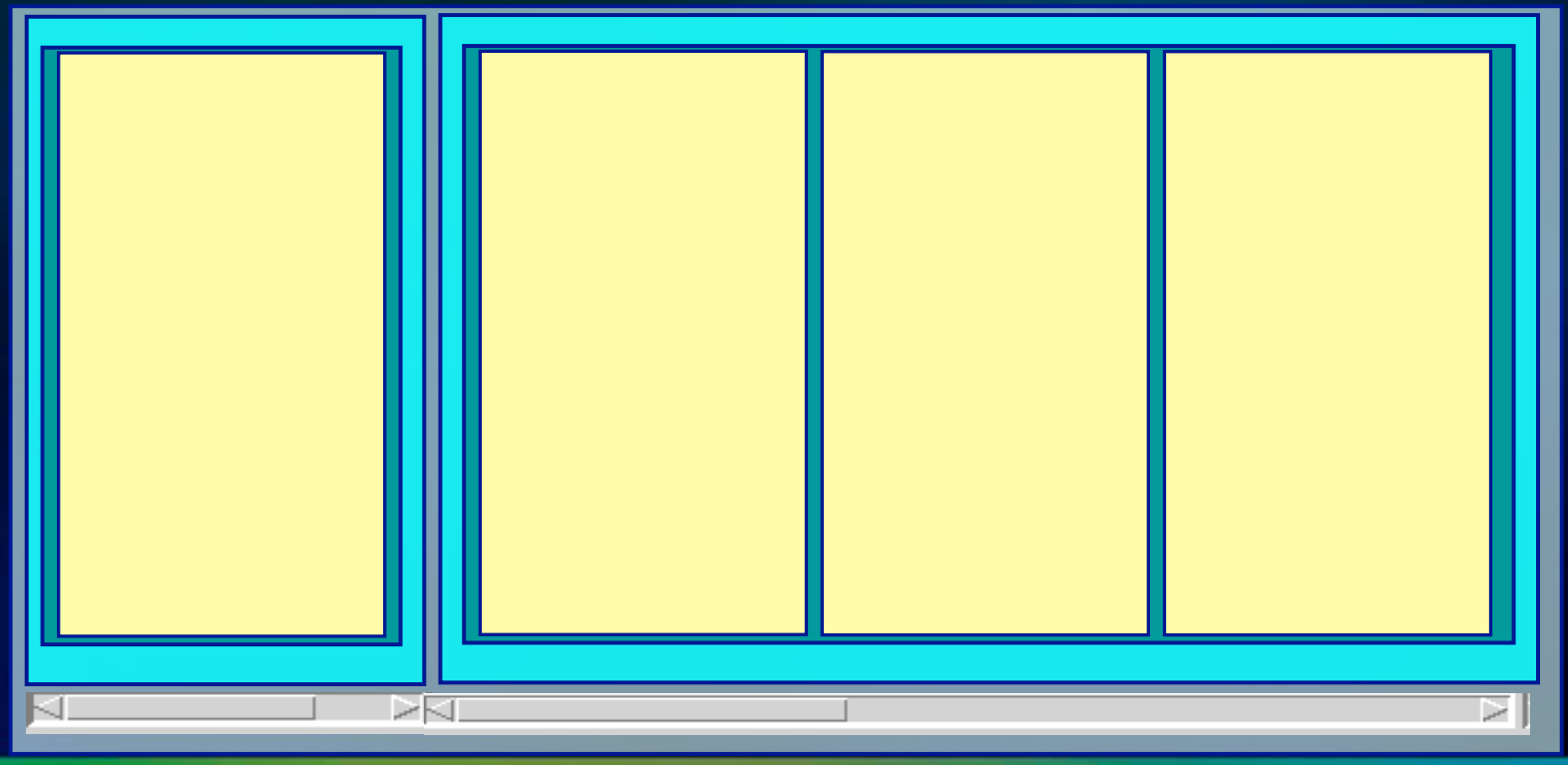
Frozen Tree Column

- Create two canvases and grid them inside the clipper as column 0 and column 1 of row 0



Frozen Tree Column

- 1st scrollbar alters the xview of the column 0 such that it scrolls the instance names of the tree column and not the column itself.
- 2nd scrollbar alters the xview of the 2nd canvas so that the rest of the columns scroll past the tree column.



Agenda

- Enhancement in MtiHierarchy widget
  - Callback support for improved performance
  - Sub column support
  - Making tree column frozen in place (non-scrollable)
- Enhancement in Text widget
  - Hyperlinks
  - Incremental Parsing

Enhanced Text Widget

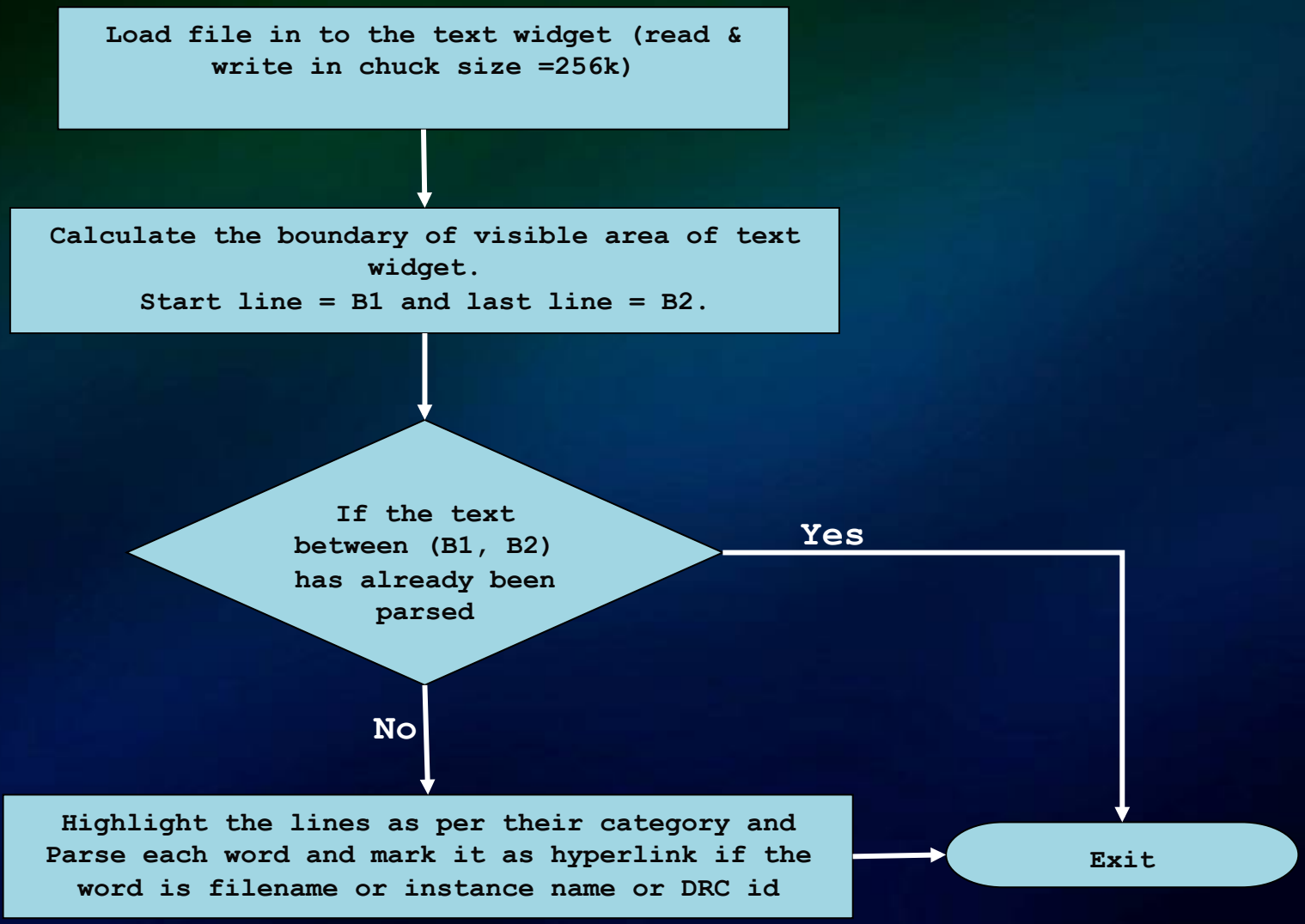
- Text Highlighting
  - Text highlighting helps to point out the issues easily
- Hyperlinks
  - Get to the relevant file or information in other windows of the tool to debug the issues very quickly.
- **Incremental parsing**

Enhanced Text Widget

Transcript

```
//  Reading group test procedure file stimulus/clock_c3_c4.g1.
//  Simulating load/unload procedure in g1 test procedure file.
//  Chain = c1 successfully traced with scan_cells = 2.
//  2 scan cells have been identified in 1 scan chain.
//  Scan group g1 successfully passed master_observe procedure audit.
//  ------------------------------------------------------------------------
//  Begin transparent latch checking for 51 latches.
//  ------------------------------------------------------------------------
//  Warning: 22 latches not transparent due to all clocks off.  (D6)
//  1 TLAs are involved in feedback networks.
//  Number transparent latches = 29.
//  1 feedback networks were identified.
//  ------------------------------------------------------------------------
//  Begin scan clock rules checking.
//  ------------------------------------------------------------------------
//  3 scan clock/set/reset lines have been identified.
//  All scan clocks successfully passed off-state check.
//  22 sequential cells passed clock stability checking.
//  All scan clocks successfully passed capture ability check.
//  Error: Clock /PH1 failed rule C3 on input 3 of /I_8516__I_582 (560).  (C3-1)
//           Source of violation: input 2 of /I_9415___1 (583).
//  Error: Rules checking unsuccessful, cannot exit SETUP mode.
//  command: set gate level primitive
//  command: analyze drc violation c3-1 -display
//  Note: Gate report now set to clock_cone (clock=/PH1).
SETUP> //         'DOFile ./stimulus/test.do' aborted at line 17
```

Enhanced Text Widget – Incremental Parsing



Thank you.