User Interface - Feature #4393

add TITLE-DCOLOR attribute support

11/08/2019 04:01 PM - Greg Shah

Status:	Closed		**Start date:**	
Priority:	Normal		**Due date:**	
Assignee:	Roger Borrello		**% Done:**	100%
Category:			**Estimated time:**	0.00 hour
Target version:				
billable:	No		**version:**	
vendor_id:	GCD			

Description

Related issues:	
Related to User Interface - Bug #4604: DCOLOR on FRAME or BROWSE TITLE phrase...	**New**
Related to User Interface - Bug #4617: Number of GUI frames in default window...	**New**
Related to User Interface - Bug #4618: Default location of dialog box title i...	**New**
Related to User Interface - Bug #4619: GUI fill-in widgets show in FWD, when ...	**New**

History

#1 - 11/08/2019 04:05 PM - Greg Shah

Implement TITLE-DCOLOR support (and also TITLE-FGCOLOR and TITLE-BGCOLOR) also make sense to implement at this same time, unless it is troublesome). TITLE-DCOLOR is ChUI and the others are GUI.

The application in question only uses it for dialog-box, but we should go ahead and implement for browse, frame and popup menu as well (these are the 4 listed places it can be used).

#2 - 03/02/2020 05:30 PM - Greg Shah

The following testcases were created:

```
testcases/uast/browse_title_color_attributes.p
testcases/uast/browse_title_color_default_values.p
testcases/uast/browse_title_color_default_value_core.i
testcases/uast/browse_title_color_option_core.i
testcases/uast/browse_title_color_options.p
testcases/uast/frame_title_color_attributes.p
testcases/uast/frame_title_color_default_values.p
testcases/uast/frame_title_color_options.p
testcases/uast/menu/menu_title_color_attributes.p
testcases/uast/menu/popup_menu_title_color_attributes.p
```

Title phrase options DCOLOR, BGCOLOR, FGCOLOR and FONT are covered (the use of these inside a TITLE [options] <title_char_expr> clause). The title phrase can only be specified in a frame phrase (dialogs or regular frames) or a browse options phrase. Menus can't have a title phrase.

The attributes TITLE-DCOLOR, TITLE-BGCOLOR, TITLE-FGCOLOR and TITLE-FONT are tested for frames (dialogs or regular frames), browse and menus (regular and popup).

I believe the testcases to be quite complete.

#3 - 03/02/2020 06:43 PM - Greg Shah

Title Phrase

Summary:

- the syntax processing of options for the frame phrase and the browse options phrase is identical (same possible syntax, same compile errors in ChUI, same limitations)
- some of the syntax options are undocumented for browse but they are possible
- in GUI, no options have any impact
- in ChUI
  - in browse, no options have any behavior
  - in frame, DCOLOR and COLOR are honored

See:

```
testcases/uast/browse_title_color_option_core.i
testcases/uast/browse_title_color_options.p
testcases/uast/frame_title_color_options.p
```

Results are from 11.x.

Frame Phrase

Syntax:

```
TITLE [DCOLOR <color_num> | BGCOLOR <color_num> | FGCOLOR <color_num> | COLOR <color_phrase> | FONT <font_num>
] <title_char_expr>
```

- limits on syntax
  - COLOR
    - cannot be specified if BGCOLOR or FGCOLOR are present
    - in GUI it **can** be used with DCOLOR
    - in ChUI it **cannot** be used with DCOLOR (this is a ChUI-specific compile failure with 2 messages Conflict in COLOR options -- FGC, BGC, PFC, and COLOR. (3523) and ** frame_title_color_options.p Could not understand line 21. (196))
    - it can be present with FONT in both GUI and ChUI
  - BGCOLOR, DCOLOR and FGCOLOR and/or FONT can be intermixed with each other in any way
  - any of them can be specified by themselves
- behavior results
  - in GUI (at least in later versions of the 4GL), **none of these options have any affect**
  - in ChUI:
    - DCOLOR and COLOR <color_phrase> are both honored
    - BGCOLOR, FGCOLOR and FONT are all silently ignored (the value has no affect on the resulting title)

Browse Options Phrase

Documented Syntax:

```
TITLE [DCOLOR <color_num>] <title_char_expr>
```

Actual Syntax:

```
TITLE [DCOLOR <color_num> | BGCOLOR <color_num> | FGCOLOR <color_num> | COLOR <color_phrase> | FONT <font_num>
] <title_char_expr>
```

- limits on syntax
  - BGCOLOR, FGCOLOR, COLOR or FONT are **undocumented options** for browse (all of them are accepted)
  - COLOR
    - cannot be specified if BGCOLOR or FGCOLOR are present
    - in GUI it **can** be used with DCOLOR
    - in ChUI it **cannot** be used with DCOLOR (this is a ChUI-specific compile failure with 2 messages Conflict in COLOR options -- FGC, BGC, PFC, and COLOR. (3523) and ** browse_title_color_option_core.i Could not understand line 21. (196))
    - it can be present with FONT in both GUI and ChUI
  - BGCOLOR, DCOLOR and FGCOLOR and/or FONT can be intermixed with each other in any way
  - any of them can be specified by themselves
- behavior results

- in GUI and in ChUI, **none of these options have any affect**

#4 - 03/03/2020 06:28 PM - Greg Shah

Attributes

This documents the TITLE-BGCOLOR, TITLE-DCOLOR, TITLE-FGCOLOR and TITLE-FONT attributes for frame (both regular frames and dialogs), browse and popup menus.

Frames

See:

```
testcases/uast/frame_title_color_attributes.p
testcases/uast/frame_title_color_default_values.p
```

- no difference in behavior was seen between regular frames and dialogs, except in ChUI assignment of TITLE-DCOLOR (see below)
- initial values
  - setting options in the title phrase makes no difference on the initial values, reading the attributes shows no affect from any option being present (or missing)
  - this includes the COLOR option to the title phrase
  - in GUI:
    - TITLE-BGCOLOR and TITLE-FGCOLOR are set to 0 **before** the frame is realized and ? (unknown value) **after** the frame is realized
    - TITLE-DCOLOR and TITLE-FONT are always read as ? (unknown value)
  - in ChUI:
    - TITLE-BGCOLOR and TITLE-FGCOLOR are always read as ? (unknown value)
    - TITLE-DCOLOR and TITLE-FONT are set to 0 **before** the frame is realized and cannot be read **after** the frame is realized (see below for the 4088 warning)
- reading
  - always returns the initial values as noted above for GUI and ChUI
  - in ChUI, **after realization** there is the additional limitation:
    - trying to read TITLE-DCOLOR generates a 4088 WARNING (error-status:error is false and error-status:get-number(1) is 4088) with the text **Progress does not support the TITLE-DCOLOR attribute for the FRAME f in this display environment. (4088)
    - trying to read TITLE-FONT generates a 4088 WARNING (error-status:error is false and error-status:get-number(1) is 4088) with the text **Progress does not support the TITLE-FONT attribute for the FRAME f in this display environment. (4088)
- writing/assignment
  - in GUI, all of these attributes can be set to any value (including the unknown value literal) and there is NO affect (no visible change in the frame title AND reading the value back shows no change)
  - in ChUI:
    - **before** realization, all attributes can be assigned; this assignment has no affect on regular frames but for dialogs, the TITLE-DCOLOR will be honored
    - for both frames and dialogs, **after** realization:
      - TITLE-DCOLOR and TITLE-FGCOLOR can be assigned but it has no affect and the value cannot be read back (reading shows no change occurred)
      - trying to write TITLE-BGCOLOR generates a 4088 WARNING (error-status:error is false and error-status:get-number(1) is 4088) with the text **Progress does not support the TITLE-BGCOLOR attribute for the FRAME f in this display environment. (4088)
    - trying to write TITLE-FONT generates a 4088 WARNING (error-status:error is false and error-status:get-number(1) is 4088) with the text **Progress does not support the TITLE-FONT attribute for the FRAME f in this display environment. (4088)

Browse

See:

```
testcases/uast/browse_title_color_attributes.p
testcases/uast/browse_title_color_default_values.p
testcases/uast/browse_title_color_default_value_core.i
```

- initial values
  - setting options in the title phrase makes no difference on the initial values, reading the attributes shows no affect from any option being present (or missing)
  - this includes both the DCOLOR documented option (which is actually honored) in the title phrase
  - in GUI:
    - TITLE-BGCOLOR is always ? (unknown value) **before and after realization**
    - TITLE-DCOLOR is always ? (unknown value) **before and after realization**
    - TITLE-FGCOLOR is always ? (unknown value) **before and after realization**
    - TITLE-FONT is always 0 **before and after realization**
    - it does not matter if any of the options are specified or if no options are specified, the attributes never change
  - in ChUI:
    - TITLE-BGCOLOR is always ? (unknown value) **before and after realization**
    - TITLE-DCOLOR is ? (unknown value) before realization and it cannot be read after realization (see below)
    - TITLE-FGCOLOR is always ? (unknown value) **before and after realization**
    - TITLE-FONT is always ? (unknown value) **before and after realization**

- reading
    - for both ChUI and GUI browse, all of these attributes are read-only
    - it does not matter if any of the options are specified or if no options are specified, the attributes never change
    - after realization, trying to read TITLE-DCOLOR generates a 4088 WARNING (error-status:error is false and error-status:get-number(1) is 4088) with the text **Progress does not support the TITLE-DCOLOR attribute for the BROWSE b in this display environment. (4088)
- writing
    - any attempt at assignment generates a runtime WARNING 4052 (error-status:error is false and error-status:get-number(1) is 4052) with the text TITLE-<whatever> is not a setable attribute for BROWSE b. (4052)
    - in both GUI and ChUI, there is no affect for setting any of these (no matter whether it is before or after realization)

Menus

See:

```
testcases/uast/menu/menu_title_color_attributes.p
testcases/uast/menu/popup_menu_title_color_attributes.p
```

Popup Menus

- initial values
    - all attributes are initially set to ? (unknown value)
    - this is the same in both GUI and ChUI
- reading
    - in GUI:
        - TITLE-BGCOLOR, TITLE-FGCOLOR and TITLE-FONT will each report the last assigned value or ? if never assigned
        - TITLE-DCOLOR is always ? even after assignment
    - in ChUI:
        - TITLE-BGCOLOR, TITLE-DCOLOR and TITLE-FONT will each report the last assigned value or ? if never assigned
        - TITLE-FGCOLOR is always ? even after assignment
- writing
    - in GUI:
        - TITLE-BGCOLOR, TITLE-FGCOLOR and TITLE-FONT can each be assigned and the value is saved (and can be read back out)
        - TITLE-DCOLOR can be assigned but the value is not saved, it is always ? even after assignment
    - in ChUI:
        - TITLE-BGCOLOR, TITLE-DCOLOR and TITLE-FONT can each be assigned and the value is saved (and can be read back out)
        - TITLE-FGCOLOR can be assigned but the value is not saved, it is always ? even after assignment
    - there is never a change in behavior
    - note that the title does not appear at all, so perhaps if the title was actually supported it would work
    - TODO: I could not find a way to open the ChUI popup menu from 4GL code or via user input; thus I could not see if there was any title present or if there was any visual difference

Non-Popup Menus

For both GUI and ChUI NON-popup menus:

- all attributes can be read and always report ? (unknown value) **before and after realization**
- assignments are allowed but:
    - no attributes are changed by any assignment
    - no errors or warnings are given
    - there is no visible affect at all

This is somewhat consistent with the 4GL documentation that states that these attributes are only valid for popup menus.

#5 - 03/24/2020 12:38 PM - Greg Shah

- Assignee set to Roger Borrello

#6 - 03/24/2020 05:06 PM - Greg Shah

- File ges_4393_title_color_and_font_attributes_and_options_support_built_on_4231b_rev_11389_20200324.zip added

Attached are my incomplete changes for this task. The intention is to **finish the implementation of all the options and attributes as documented in #4393-2 through #4393-4.** You MUST read those entries carefully. I think the conversion side is done. The runtime is partially implemented. I think the remaining changes are:

- TITLE-BGCOLOR
    - BrowseWidget: Add getTitleBgColor(), setTitleBgColor(int64 bgcolor), setTitleBgColor(long bgcolor).
    - MenuWidget: Add getTitleBgColor(), setTitleBgColor(int64 bgcolor), setTitleBgColor(long bgcolor).
    - GenericFrame: Add getTitleBgColor(), setTitleBgColor(int64 bgcolor), setTitleBgColor(long bgcolor).
- TITLE-DCOLOR
    - BrowseWidget: Add getTitleDColor(), setTitleDColor(int64 dcolor), setTitleDColor(long dcolor).
    - MenuWidget: Remove setTitleColor(ColorSpec titleColor) (I think it is dead code that can't be used). Add getTitleDColor(), setTitleDColor(int64 dcolor), setTitleDColor(long dcolor).
    - GenericFrame: Add getTitleDColor(), setTitleDColor(int64 dcolor), setTitleDColor(long dcolor). Please leave setTitleColor(ColorSpec titleColor) alone since it is used for COLOR phrase conversion and a different interface from the other options/attributes.
- TITLE-FGCOLOR
    - BrowseWidget: Add getTitleFgColor(), setTitleFgColor(int64 fgcolor), setTitleFgColor(long fgcolor).
    - MenuWidget: Add getTitleFgColor(), setTitleFgColor(int64 fgcolor), setTitleFgColor(long fgcolor).
    - GenericFrame: Add getTitleFgColor(), setTitleFgColor(int64 fgcolor), setTitleFgColor(long fgcolor).
- TITLE-FONT
    - BrowseWidget: Rework getTitleFont() and setTitleFont(int64) to match documented behavior. Add setTitleFont(long font).
    - MenuWidget: Add getTitleFont() , setTitleFont(int64) and setTitleFont(long font).
    - GenericFrame: Rework getTitleFont(), setTitleFont(int64) and setTitleFont(long font) to match documented behavior.

#7 - 03/24/2020 05:48 PM - Roger Borrello

- Assignee deleted (Roger Borrello)

Is the below caused by commented out options { generateAmbigWarnings = false; } ? Was that just for debug?

```
[ant:antlr] ANTLR Parser Generator   Version 2.7.7 (20060906)   1989-2005
[ant:antlr] /home/rfb/projects/fwd/4231b-dev/src/com/goldencode/p2j/uast/progress.g:15688: warning:nondetermin
ism upon
[ant:antlr] /home/rfb/projects/fwd/4231b-dev/src/com/goldencode/p2j/uast/progress.g:15688:      k==1:KW_ACCEL,K
W_BGCOLOR,KW_COL,KW_COL_BGC,KW_COL_DC,KW_COL_FGC,KW_COL_FONT,KW_COL_PFC,KW_CTX_H_ID,KW_DCOLOR,KW_FGCOLOR,KW_LA
B_BGC,KW_LAB_DC,KW_LAB_FGC,KW_LAB_FONT,KW_LAB_PFC,KW_MOU_PTR,KW_PFCOLOR,KW_ROW,KW_WID_ID
```

#8 - 03/24/2020 07:59 PM - Roger Borrello

- Assignee set to Roger Borrello

#9 - 03/24/2020 09:49 PM - Greg Shah

Just re-enable the warnings suppression.

#10 - 03/25/2020 10:58 AM - Roger Borrello

I'm in BrowseWidget and I see some methods with @Override annotation, and some without. What determines whether it is needed or not?

#11 - 03/25/2020 11:17 AM - Greg Shah

It is recommended anytime you are:

- implementing a method from an interface (which you are doing, right?) OR
- overriding a method in your parent class hierarchy

#12 - 03/25/2020 11:21 AM - Roger Borrello

Greg Shah wrote:

> It is recommended anytime you are:
>
> - implementing a method from an interface (which you are doing, right?) OR
> - overriding a method in your parent class hierarchy

Yes, I'm implementing. I will leave what is in there without the attribute for now. If I have time, I'll dive deeper. I will make sure my additions are correct.

#13 - 03/25/2020 11:44 AM - Roger Borrello

Should there be setters for **FG**, **BG**, and **D** which receive (ColorSpec color)? Or just int64 and long?

#14 - 03/25/2020 11:45 AM - Greg Shah

Just implement what is in TitledElement. The ColorSpec stuff is not related.

#15 - 03/25/2020 12:52 PM - Roger Borrello

GenericFrame had a method that looks a little broken:

```
/**
 * Sets the color for the frame title.
 *
 * @param    titleColor
 *           The overridden color for the title.
 */
@Override
public void setTitleColor(ColorSpec titleColor)
{
   frame.config.titleDColor = titleColor.convert();
   frame.pushWidgetAttr("titleDColor", frame.config.titleDColor);
```

```
            }
```

It's using the titleDColor attribute. Was this just a little half-cocked? It also already had setTitleFgColor but using ColorSpec color. Should that be left alone, and I just add the other 2 signatures?

#16 - 03/25/2020 12:59 PM - Greg Shah

Did you see this part in #4393-6:

> GenericFrame: Add getTitleDColor(), setTitleDColor(int64 dcolor), setTitleDColor(long dcolor). Please leave setTitleColor(ColorSpec titleColor) alone since it is used for COLOR phrase conversion and a different interface from the other options/attributes.

> It also already had setTitleFgColor but using ColorSpec color.

I don't think this one is needed.

#17 - 03/25/2020 01:34 PM - Roger Borrello

Greg Shah wrote:

> Did you see this part in #4393-6:

>> GenericFrame: Add getTitleDColor(), setTitleDColor(int64 dcolor), setTitleDColor(long dcolor). Please leave setTitleColor(ColorSpec titleColor) alone since it is used for COLOR phrase conversion and a different interface from the other options/attributes.

Yes I did, but in that method, it is setting frame.config.titleDColor instead of frame.config.titleColor. It seems implemented wrong.

> It also already had setTitleFgColor but using ColorSpec color.

> I don't think this one is needed.

OK.

#18 - 03/25/2020 01:45 PM - Greg Shah

Yes I did, but in that method, it is setting frame.config.titleDColor instead of frame.config.titleColor.

There is no such thing as frame.config.titleColor.

It seems implemented wrong.

It is OK.

#19 - 03/25/2020 02:41 PM - Roger Borrello

Are titleBGgColor and titleFgColor attributes to be kept with titleDColor and titleFont in BaseConfig?

#20 - 03/25/2020 03:20 PM - Greg Shah

Consider the following facts:

- The reason we store state in a separate config class (e.g. BaseConfig) is to have a serializable container which can be transferred between server and client (both directions). This enables client side state to be set and read on the server while being actually honored/implemented in the client.
- Any state that is not needed on the client, does not need to be placed inside one of these classes.
- If you read the #4393-4 details carefully, you will find that only the TITLE-DCOLOR **frame** attribute (and TITLE DCOLOR <color> title **frame** option are ever honored, and this is only for ChUI. Thus, only config.titleDColor ever matters to the client.

Clear?

#21 - 03/25/2020 09:34 PM - Roger Borrello

The documentation for Attributes:Frames for writing/assignment says...
in ChUI:
 before realization, all attributes can be assigned; this assignment has no affect on regular frames but for dialogs, the TITLE-DCOLOR will be honored

Does this mean **titleDColor** is stored outside GenericFrame?

#22 - 03/25/2020 09:45 PM - Roger Borrello

Updates are buildable in revision 11395. Right now GenericFrame is the only update without stubs. Please review... there's plenty left to do, like store

titleDColor properly.

#23 - 03/26/2020 06:53 AM - Greg Shah

Does this mean titleDColor is stored outside GenericFrame?

Yes.

As I said in [#4393-20](#):

Thus, only config.titleDColor ever matters to the client.

Do not mess with the config.titleDColor.

#24 - 03/26/2020 07:29 AM - Greg Shah

Code Review Task Branch 4231b Revision 11395

The changes are moving the right direction.

1. You've messed with config.titleDColor. No bueno.

2. The state management in your GenericFrame changes is not 100% correct. Please compare the details above with your implementation. For example, TITLE-FONT always returns unknown in ChUI but in GUI there is different behavior based on the realized flag.

#25 - 03/26/2020 08:55 AM - Roger Borrello

Greg Shah wrote:

Code Review Task Branch 4231b Revision 11395

The changes are moving the right direction.

1. You've messed with config.titleDColor. No bueno.

Where? In setTitleColor(ColorSpec titleColor) on line 6860? That wasn't me... I mentioned how odd that method looked. Did you mean to *whack* it in #note-16? Happy to do so.

2. The state management in your GenericFrame changes is not 100% correct. Please compare the details above with your implementation. For example, TITLE-FONT always returns unknown in ChUI but in GUI there is different behavior based on the realized flag.

That's a stub... I'll resume and let you know. BTW, I left some odd looking logic in some of the methods, since it helped to match up with the spec. Would you prefer I simplify the logic, once we agree it matches, or just document the snot out of it so it's somewhat readable?

#26 - 03/26/2020 09:05 AM - Greg Shah

In setTitleColor(ColorSpec titleColor) on line 6860? That wasn't me... I mentioned how odd that method looked. Did you mean to whack it in #note-16? Happy to do so.

No. Don't touch that.

To be mre clear: there must not be a local member variable for titleDColor. In ChUI, changes to that value MUST be in frame.config.titleDColor in order for the client to honor it.

#27 - 03/26/2020 10:38 AM - Roger Borrello

Using this to ef with titleDColor:

```
    frame.config.titleDColor = new Color(0, (int) dcolor);
    frame.pushWidgetAttr("titleDColor", frame.config.titleDColor);
```

OK?

Moved on with the ./uast/frame_title_color_attributes.p testcase, and received a compilation error. So there are supposed to be implementations in there? Or is something else wrong?

```
    [javac] /home/rfb/projects/VirtualBox-VMs/shared/projects/testcases/src/com/goldencode/testcases/FrameTitl
eColorAttributes.java:58: error: cannot find symbol
    [javac]           b4Bgc.assign(fFrame.asWidget().getTitleBgColor());
    [javac]                                         ^
    [javac]   symbol:   method getTitleBgColor()
    [javac]   location: class FrameWidget
```

#28 - 03/26/2020 10:48 AM - Greg Shah

I forgot that we emit attributes using the FrameWidget instead of direct access to the GenericFrame instance. The TitledElement needs to be implemented in FrameWidget instead of GenericFrame. The frame options will map to GenericFrame in the frame definition, so those cannot move. The dichotomy is caused by the lack of multiple inheritance in Java. The 4GL sometimes treats frames like a widget (i.e. something that is a sub-class of GenericWidget) and sometimes treats frames as frames (a very special visual container that implements layout and other features not present in widgets). The non-widget frame cases are in GenericFrame and implement CommonFrame as the interface. The stuff that normally comes from a handle usage is thus put into FrameWidget as it is considered widget stuff.

#29 - 03/26/2020 11:01 AM - Roger Borrello

Greg Shah wrote:

> I forgot that we emit attributes using the FrameWidget instead of direct access to the GenericFrame instance. The TitledElement needs to be implemented in FrameWidget instead of GenericFrame. The frame options will map to GenericFrame in the frame definition, so those cannot move. The dichotomy is caused by the lack of multiple inheritance in Java. The 4GL sometimes treats frames like a widget (i.e. something that is a sub-class of GenericWidget) and sometimes treats frames as frames (a very special visual container that implements layout and other features not present in widgets). The non-widget frame cases are in GenericFrame and implement CommonFrame as the interface. The stuff that normally comes from a handle usage is thus put into FrameWidget as it is considered widget stuff.

I will wholesale move set/get Title BgColor/DColor/FgColor/Font to FrameWidget. Where are the frame-options methods?

#30 - 03/26/2020 11:11 AM - Greg Shah

> Where are the frame-options methods?

That is the setTitleColor(ColorSpec) (which you keep complaining about) which handles the COLOR <color_phrase>.

And you will need to LEAVE the setters in GenericFrame (setTitleBgColor(int64 bgcolor), setTitleBgColor(long bgcolor), setTitleDColor(int64 dcolor), setTitleDColor(long dcolor), setTitleFgColor(int64 fgcolor), setTitleFgColor(long fgcolor), setTitleFont(int64 font), setTitleFont(long font)). The getters are not needed but these setters can be used for the equivalent options in the title phrase.

#31 - 03/26/2020 11:23 AM - Roger Borrello

Greg Shah wrote:

> Where are the frame-options methods?

That is the setTitleColor(ColorSpec) (which you keep complaining about) which handles the COLOR <color_phrase>.

And you will need to LEAVE the setters in GenericFrame (setTitleBgColor(int64 bgcolor), setTitleBgColor(long bgcolor), setTitleDColor(int64 dcolor), setTitleDColor(long dcolor), setTitleFgColor(int64 fgcolor), setTitleFgColor(long fgcolor), setTitleFont(int64 font), setTitleFont(long font)). The getters are not needed but these setters can be used for the equivalent options in the title phrase.

Without the getters in GenericFrame I get GenericFrame is not abstract and does not override abstract method ... in TitledElement

#32 - 03/26/2020 12:04 PM - Greg Shah

GenericFrame should not implement TitledElement, only implement in FrameWidget. You can still leave the setter methods in GenericFrame without the interface being implemented.

#33 - 03/26/2020 12:46 PM - Roger Borrello

- File title-bgcolor_not_setable.png added

Greg Shah wrote:

> GenericFrame should not implement TitledElement, only implement in FrameWidget. You can still leave the setter methods in GenericFrame without the interface being implemented.

GenericFrame doesn't implement TitledElement, but it does implement CommonFrame, which implements TitledElement. How to rectify?

Proceeding with the testcase, with all getters/setters in both GenericFrame and FrameWidget for now... I get the error **TITLE-BGCOLOR is not a setable attribute for FRAME widget. (4052) when I launch the testcase, before the window is rendered.

#34 - 03/26/2020 01:06 PM - Greg Shah

Remove it from CommonFrame.

> Proceeding with the testcase, with all getters/setters in both GenericFrame and FrameWidget for now... I get the error **TITLE-BGCOLOR is not a setable attribute for FRAME widget. (4052) when I launch the testcase, before the window is rendered.

Feel free to fix your bugs.

#35 - 03/26/2020 01:47 PM - Roger Borrello

Greg Shah wrote:

> Feel free to fix your bugs.

LOL... nowhere did I add a 4052 warning... I'll find and fix, whoever's it is.

#36 - 03/26/2020 02:16 PM - Roger Borrello

Greg Shah wrote:

> Remove it from CommonFrame.

When extends TitledElement is removed from CommonFrame, the testcase's frame doesn't get the methods:

```
    [javac] /home/rfb/projects/VirtualBox-VMs/shared/projects/testcases/src/com/goldencode/testcases/FrameTitl
eColorAttributes.java:61: error: cannot find symbol
    [javac]              b4Font.assign(fFrame.getTitleFont());
    [javac]                                  ^
.
.
.
```

#37 - 03/26/2020 02:21 PM - Greg Shah

Did you add it to FrameWidget?

#38 - 03/26/2020 02:23 PM - Roger Borrello

Greg Shah wrote:

> Did you add it to FrameWidget?

ahh... TitledElement is in there as implements, but not extends.

#39 - 03/26/2020 02:38 PM - Greg Shah

Why would we want it to be extends?

#40 - 03/26/2020 02:56 PM - Roger Borrello

Greg Shah wrote:

> Why would we want it to be extends?

I don't want to do that. I believe I need to add our getters/setters to the CommonFrame interface, don't I?

#41 - 03/26/2020 02:57 PM - Greg Shah

No, I don't think so.

#42 - 03/26/2020 02:59 PM - Greg Shah

Figure out why the getTitleFont() is converting differently than getTitleBgColor(). The font is not converting to FrameWidget. Why?

#43 - 03/26/2020 03:43 PM - Roger Borrello

Greg Shah wrote:

> Figure out why the getTitleFont() is converting differently than getTitleBgColor(). The font is not converting to FrameWidget. Why?

Well the trees for b4-fgc = frame f:title-fgcolor and b4-font = frame f:title-font look almost exactly the same. Both using accessor=asWidget and widgetType=FrameWidget

But the conversion is definitely different:

```
        b4Fgc.assign(fFrame.asWidget().getTitleFgColor());
        b4Font.assign(fFrame.getTitleFont());
```

kw_titl_ keywords are in methods_attributes.rules, but only kw_titl_fon is in widget_references.rules.

Fixed this... but where should readOnlyError come from?

```
    [javac] /home/rfb/projects/VirtualBox-VMs/shared/projects/testcases/src/com/goldencode/testcases/FrameTitl
eColorAttributes.java:359: error: cannot find symbol
    [javac]            dfFrame.readOnlyError("title-bgcolor");
    [javac]                   ^
    [javac]   symbol:   method readOnlyError(String)
    [javac]   location: variable dfFrame of type FrameTitleColorAttributesDf
```

#44 - 03/26/2020 03:52 PM - Roger Borrello

Nevermind. Found the sources to our readOnlyError.

#45 - 03/26/2020 04:34 PM - Roger Borrello

In common-progress.rules:read_only_attribute is included prog.kw_titl_bgc. I am going to remove this, since we can set it. Does that sound correct? That was the source of the above error.

#46 - 03/26/2020 04:48 PM - Greg Shah

Correct.

#47 - 03/26/2020 05:08 PM - Roger Borrello

I have the frame_title_color_attributes.p testcase working on GUI. There are some differences, between FWD and 4GL, but unrelated. For example, FWD i widgets are filled with 0's, but 4GL widgets are blank. Another, the title to the dialogs are left-justified in FWD, and centered in 4GL.

Now to test ChUI.

#48 - 03/26/2020 06:43 PM - Roger Borrello

For the ChUI, what is the key to building it as-if using ChUI development, so that "{&window-system}" eq "tty" is true?

Right now, the error message is being displayed, but the client is cycling in a loop, and not ever displaying the dialog. Sorry... haven't spent much time in ChUI-land.

#49 - 03/26/2020 07:06 PM - Greg Shah

Look at the Hotel ChUI p2j.cfg.xml for clues. You can also add things to that project to test, though the regular testcases/uast/ will also work.

#50 - 03/26/2020 07:34 PM - Roger Borrello

Roger Borrello wrote:

> Right now, the error message is being displayed, but the client is cycling in a loop, and not ever displaying the dialog. Sorry... haven't spent much time in ChUI-land.

My bad... when we have the 4088 condition in a getter, I was returning null. Now I'm returning new integer() and it's much better.

Updates checked into rev 11399. Please look over. There's just one thing going on with ChUI, and it will help when I build it properly. I feel more confident with Menu and Browse now.

#51 - 03/27/2020 09:12 AM - Greg Shah

Code Review Task Branch 4231b Revision 11399

OK, you are close here.

Looking deeper at the changes in widget_references.rules, I think the reason we **previously** needed the title-font stuff there was because we have it defined directly in CommonWidget. In fact, we can remove the ?etTitleFont() from CommonWidget and use the normal conversion. This means removing the title-* stuff from widget_references.rules which will result in a cleaner implementation. In other words, there will be less of the "this frame is a widget" dichotomy which is so confusing.

Constantin: Do you see any flaw in my reasoning?

#52 - 03/27/2020 09:12 AM - Greg Shah

Roger: If I'm right, then you would remove all the changes from FrameWidget and just put them in GenericFrame.

#53 - 03/27/2020 09:25 AM - Constantin Asofiei

Greg Shah wrote:

> Roger: If I'm right, then you would remove all the changes from FrameWidget and just put them in GenericFrame.

Actually, I think you forgot about the 'dual behavior' of a frame - as a handle and as a standalone widget (h:title-font and frame f1:title-font access).

Roger: for frames, the place for the implementation needs to be in FrameWidget; GenericFrame APIs need to call the appropriate API from FrameWidget via GenericFrame.frame. This is the current approach to avoid code duplication.

#54 - 03/27/2020 09:39 AM - Greg Shah

I was thinking about the dual nature, but I guess I got part of it backwards. If not present in widget_references.rules can't we just implement in FrameWidget so that both forms of attribute/method usage are processed the same way. In other words, I don't see why we need the attributes to be present in CommonFrame.

I agree that the frame options (setters only, drop the getters) need to be in CommonFrame and implemented in GenericFrame. Also agree that they should call the FrameWidget implementation instead of duplicating code. Am I misunderstanding things?

#55 - 03/27/2020 09:46 AM - Constantin Asofiei

Greg Shah wrote:

> I was thinking about the dual nature, but I guess I got part of it backwards. If not present in widget_references.rules can't we just implement in FrameWidget so that both forms of attribute/method usage are processed the same way.

As long as an attribute/method can be accessed via both a handle and a frame reference (i.e. frame f1:title-font), then these need to be available in both CommonFrame and FrameWidget. At least this is how is working now.

Or, do you mean to convert frame f1:<widget/method> differently, and access the asWidget() method directly?

In other words, I don't see why we need the attributes to be present in CommonFrame.

There is a TitledElement interface being used at this time.

#56 - 03/27/2020 09:48 AM - Greg Shah

Or, do you mean to convert frame f1:<widget/method> differently, and access the asWidget() method directly?

Yes, this one. I don't see a reason that we need CommonFrame here.

#57 - 03/27/2020 09:52 AM - Constantin Asofiei

Greg Shah wrote:

Or, do you mean to convert frame f1:<widget/method> differently, and access the asWidget() method directly?

Yes, this one. I don't see a reason that we need CommonFrame here.

OK, with this one I agree, with some concerns - there might be cases where we need to access GenericFrame (i.e. during frame setup). So we might not be able to move everything 'at once'.

We should start with some attributes (like this title-related set) and slowly move the others, too.

#58 - 03/27/2020 09:56 AM - Greg Shah

Yes, I'm thinking that we leave the frame options in CommonFrame/GenericFrame. These are just the setters (setTitle*()) and NOT the getters. The getters can only be used as an attribute. The GenericFrame setters can just redirect to FrameWidget.

#59 - 03/27/2020 01:51 PM - Roger Borrello

Greg Shah wrote:

> Yes, I'm thinking that we leave the frame options in CommonFrame/GenericFrame. These are just the setters (setTitle*()) and NOT the getters. The getters can only be used as an attribute. The GenericFrame setters can just redirect to FrameWidget.

Can I get a recap for frames...

- TitiledElement - leave alone
- CommonFrame - currently an inteface that extends TitledElement. No getters or setters within at the present time
- GenericFrame - class that implements CommonFrame. Currently has all getters and setters
- FrameWidget - class that implements TitledElement. Currently has all getters and setters (duplicates)
- methods_attributes.rules - connects getters and setters to TitledElement
- widget_references.rules - Allows attributes to be directed to CommonFrame instance instead of allowing the WID_FRAME to be converted with an asWidget() accessor

#60 - 03/27/2020 02:17 PM - Greg Shah

> TitiledElement - leave alone

Yes

> CommonFrame - currently an inteface that extends TitledElement. No getters or setters within at the present time

No. It should not have TitledElement.

> GenericFrame - class that implements CommonFrame. Currently has all getters and setters

No. It should only have the setters as these can be used from frame options at conversion. The implementation should call the same method in FrameWidget which can be gotten like this: asWidget().setTitle*().

> FrameWidget - class that implements TitledElement. Currently has all getters and setters (duplicates)

Yes. But there will not be any duplication of code.

> methods_attributes.rules - connects getters and setters to TitledElement

Yes

widget_references.rules - Allows attributes to be directed to CommonFrame instance instead of allowing the WID_FRAME to be converted with an asWidget() accessor

Remove all kw_tit* items from here.

#61 - 03/27/2020 03:11 PM - Roger Borrello

- File ChUI_frame_title_color_attributes1.png added

- File ChUI_frame_title_color_attributes2.png added

My previous note was stating the "current" state, so thank you for clarifying.

With respect to the ChUI testing, I wanted to make sure I had the states correct before I moved onto correcting the implementations. In doing so, I fail to see how TITLE-DCOLOR is honored in ChUI after realization of the Dialog. See my 2 screen grabs, with side-by-side. 1st is pre-F1, 2nd is post-F1.
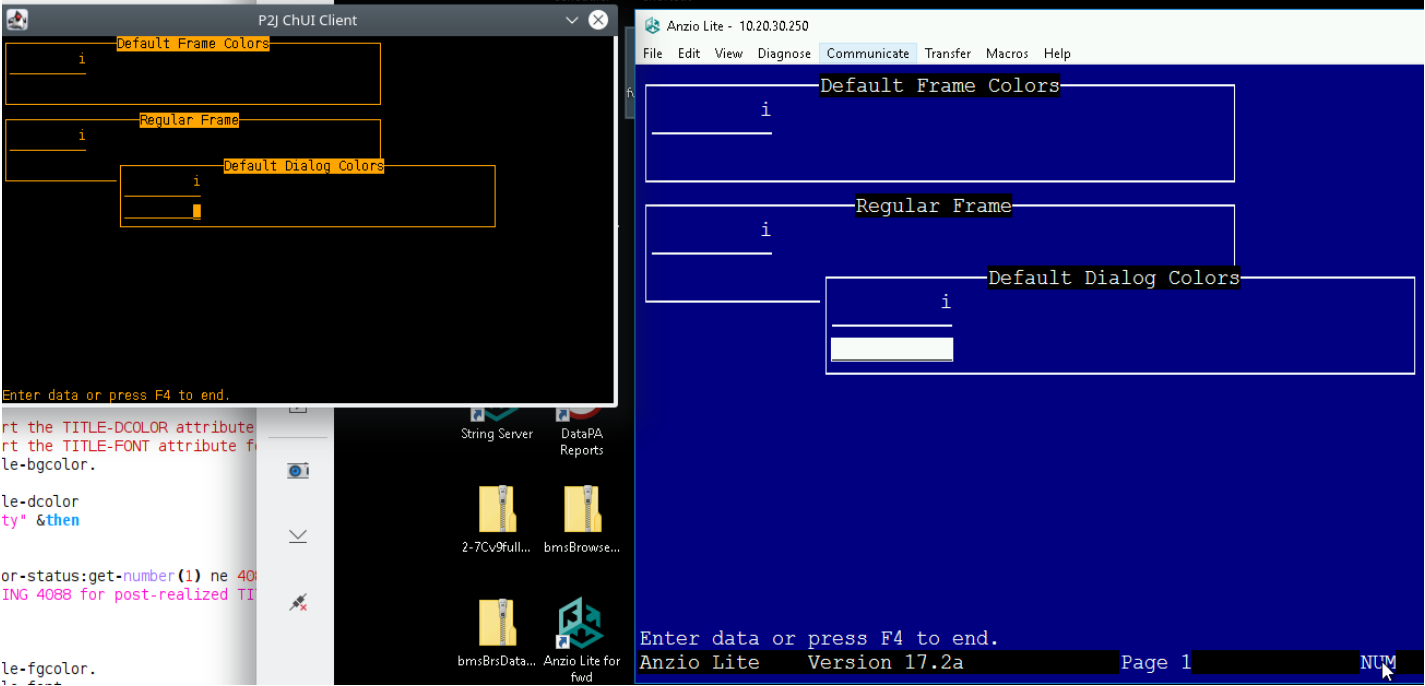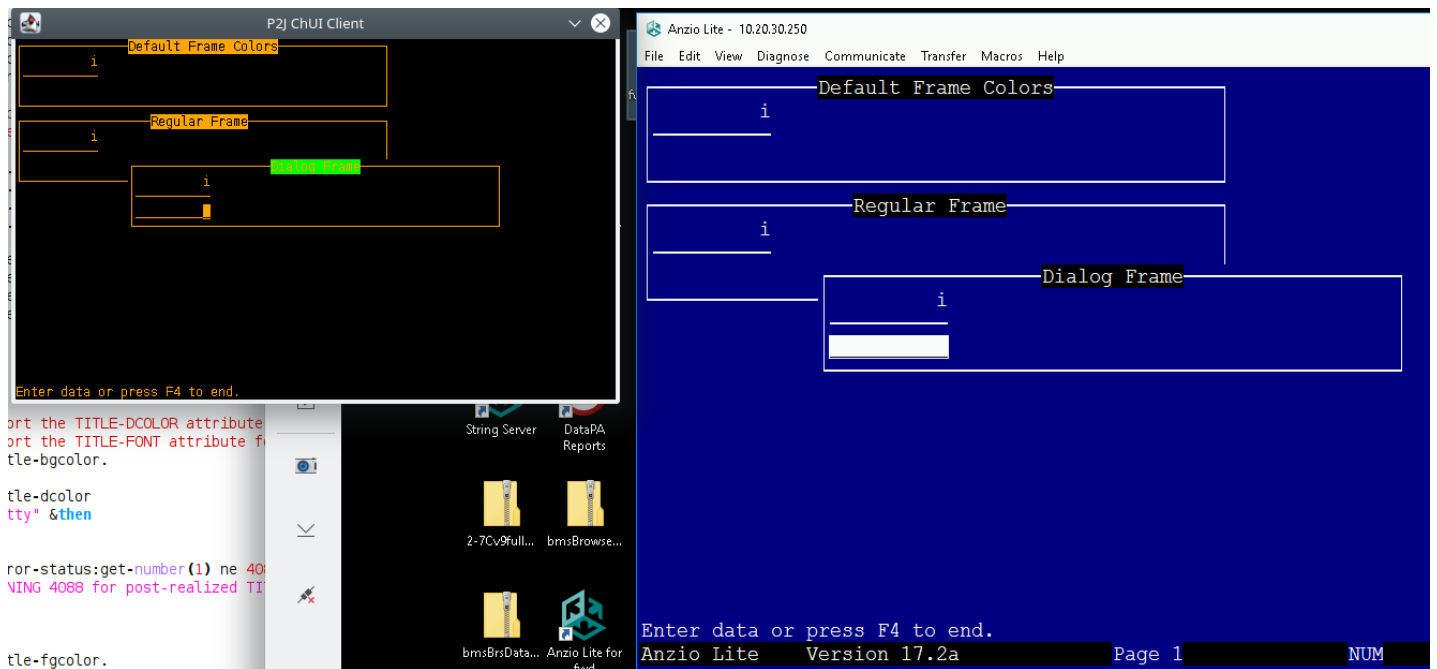
#62 - 03/27/2020 03:22 PM - Greg Shah

Try frame df:title-dcolor = 1 instead of 3 on line 208 of testcases/uast/frame_title_color_attributes.p. You should see an underline in the title of the 2nd dialog.

#63 - 03/27/2020 03:26 PM - Roger Borrello

Greg Shah wrote:

> Try frame df:title-dcolor = 1 instead of 3 on line 208 of testcases/uast/frame_title_color_attributes.p. You should see an underline in the title of the 2nd dialog.

OK... that's clearer!

Changes made for frame, and testcase still works. Menus and Browse don't have the same complexity of the class structure, do they?

#64 - 03/27/2020 03:41 PM - Roger Borrello

Updates for frames in **4231b-11400**.

#65 - 03/27/2020 03:45 PM - Roger Borrello

Roger Borrello wrote:

> Updates for frames in **4231b-11400**.

Hold on... a couple more frame testcases to validate.

#66 - 03/27/2020 03:57 PM - Roger Borrello

The frames are CommonFrame, and trying to access the methods which aren't in that interface.

```
    [javac] /home/rfb/projects/VirtualBox-VMs/shared/projects/testcases/src/com/goldencode/testcases/ui/FrameT
itleColorDefaultValuesF5.java:30: error: cannot find symbol
    [javac]            frame.setTitleFont(4);
    [javac]                 ^
    [javac]   symbol:   method setTitleFont(int)
    [javac]   location: variable frame of type CommonFrame
```

#67 - 03/27/2020 04:39 PM - Roger Borrello

Roger Borrello wrote:

> The frames are CommonFrame, and trying to access the methods which aren't in that interface.
>
> [...]

All the frame initialization set calls were unresolved.

I was able to resolve this, but it seems to go against what we laid out today.

- Change CommonFrame to extend TitledElement.
- Add getters **back** to GenericFrame which use @asWidget().@_getter_call in their implementations

#68 - 03/27/2020 05:19 PM - Greg Shah

> Menus and Browse don't have the same complexity of the class structure, do they?

No, they don't.

> All the frame initialization set calls were unresolved.

We need the setTitle*(long) methods in CommonFrame. That is it. Don't add the full TitledElement and remove the extra (unused) getters and setters (the setTitle*(int64) cases).
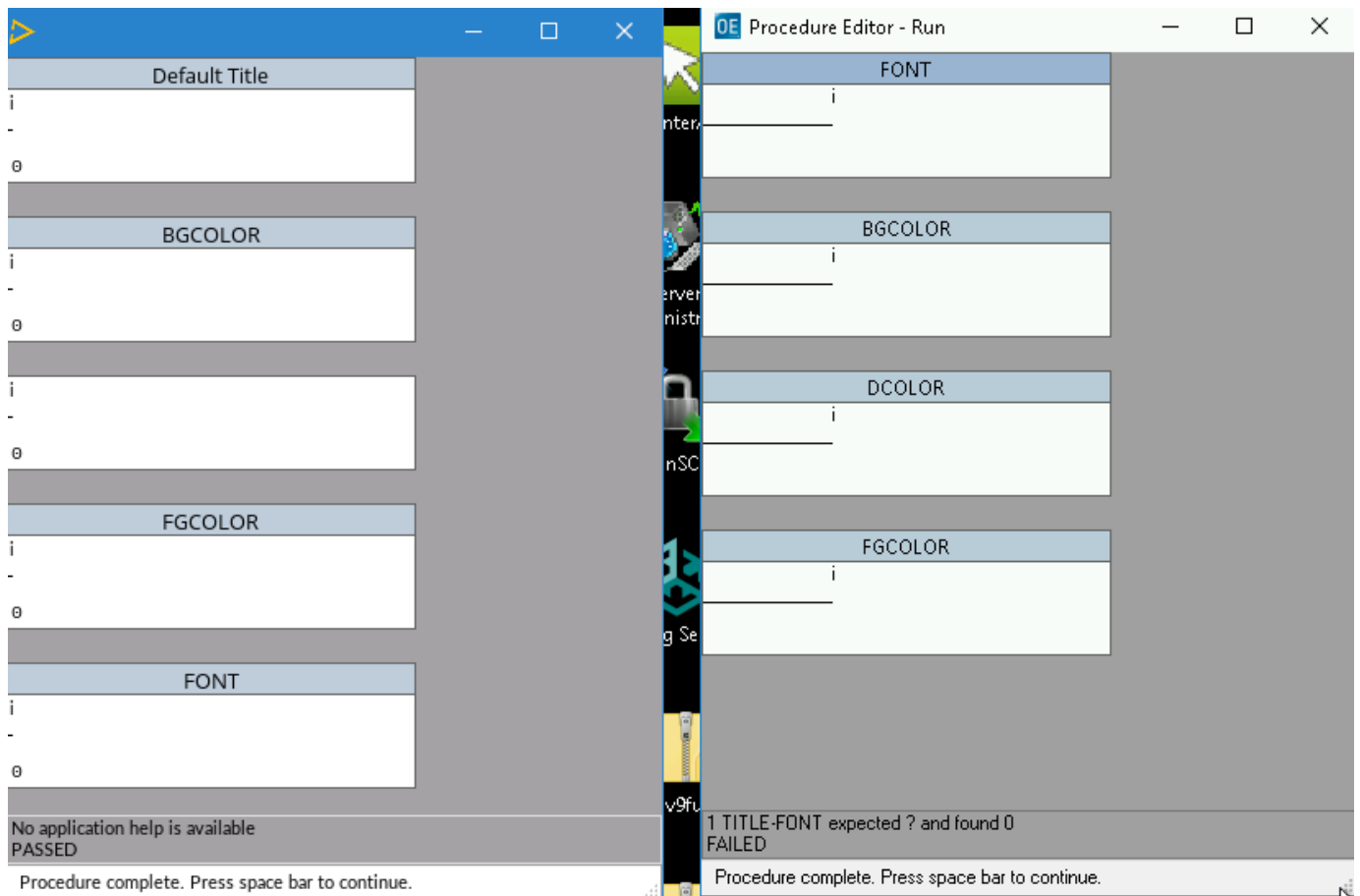
#69 - 03/27/2020 05:22 PM - Roger Borrello

- File GUI_frame_title_color_default_values2.png added

I will go back and update per your last note. I was writing the below while you made your update:

All the ChUI tests passed side-by-side, except for the fact that the color schemes seemed different. There were often times the 4GL would have a highlighting background color constrasting the foreground color, so you could read the text, while on FWD, they would be the same, and hence you couldn't read the title.

In GUI, the frame_title_color_default_values.p test had unexpected results, as the 4GL didn't get the TITLE-FONT it was expecting from initialization.

Procedure Editor - Run

Default Title

i
-
0

BGCOLOR

i
-
0

i
-
0

FGCOLOR

i
-
0

FONT

i
-
0

No application help is available
PASSED

Procedure complete. Press space bar to continue.

FONT

i

BGCOLOR

i

DCOLOR

i

FGCOLOR

i

1 TITLE-FONT expected ? and found 0
FAILED

Procedure complete. Press space bar to continue.

#70 - 03/27/2020 05:42 PM - Roger Borrello

Greg Shah wrote:

> We need the setTitle*(long) methods in CommonFrame. That is it. Don't add the full TitledElement and remove the extra (unused) getters and setters (the setTitle*(int64) cases).

Remove the unused int64 getters **and** setters from where? The GenericFrame?

#71 - 03/27/2020 05:43 PM - Greg Shah

Roger Borrello wrote:

> Greg Shah wrote:
>
> > We need the setTitle*(long) methods in CommonFrame. That is it. Don't add the full TitledElement and remove the extra (unused) getters and setters (the setTitle*(int64) cases).
>
> Remove the unused int64 getters **and** setters from where? The GenericFrame?

Yes.

#72 - 03/27/2020 05:44 PM - Greg Shah

Remove all getters. Remove the int64 setters. Only the long setters should be in either CommonFrame or GenericFrame.

#73 - 03/27/2020 05:48 PM - Greg Shah

> All the ChUI tests passed side-by-side, except for the fact that the color schemes seemed different. There were often times the 4GL would have a highlighting background color constrasting the foreground color, so you could read the text, while on FWD, they would be the same, and hence you couldn't read the title.

This is not a concern right now. The 4GL ChUI environment you are testing in probably has a customized protermcap file which would account for the difference. We know that we may need to offer a similar facility at some point.

> In GUI, the frame_title_color_default_values.p test had unexpected results, as the 4GL didn't get the TITLE-FONT it was expecting from initialization.

I wonder if this is also due to a customized default GUI ini file present in that installation.

In both of these cases, I have previously run these same programs in the 4GL and have gotten them to work. I don't think we need to look further at this right now.

#74 - 03/27/2020 06:53 PM - Roger Borrello

Cleaned up and checked in:

```
modified src/com/goldencode/p2j/ui/CommonFrame.java
modified src/com/goldencode/p2j/ui/GenericFrame.java
Committed revision 11401.
```

#75 - 03/30/2020 09:20 AM - Roger Borrello

In MenuWidget, is it adequate to utilize config.popupOnly to determine between popup and non-popup menus for the purpose of these TITLE- rules?

#76 - 03/30/2020 09:20 AM - Greg Shah

Yes.

#77 - 03/30/2020 10:07 AM - Roger Borrello

For the cases where menu TITLE- attributes are actually handled (TITLE-BGCOLOR, TITLE-FGCOLOR and TITLE-FONT), but not acted upon, this is were simple local storage would work, correct? However, determining if it is **not** set is not clear to me. Would it suffice to initialize the local storage to an invalid value, such as -999 and return integer() if that is the current value, otherwise return the current value?

#78 - 03/30/2020 10:24 AM - Greg Shah

> For the cases where menu TITLE- attributes are actually handled (TITLE-BGCOLOR, TITLE-FGCOLOR and TITLE-FONT), but not acted upon, this is were simple local storage would work, correct?

Yes.

> However, determining if it is not set is not clear to me. Would it suffice to initialize the local storage to an invalid value, such as -999 and return integer() if that is the current value, otherwise return the current value?

Using a real number is usually a bad idea. Just use null.

private integer someLocalFlag = null;

When set, you assign it a non-null value. Before using it or returning it, you check for null.

#79 - 03/30/2020 10:49 AM - Roger Borrello

The previous convention for the setters had the int64 versions calling the long versions:

```
setTitleBgColor(bgcolor.isUnknown() ? -1 : bgcolor.intValue());
```

I just reused those because they seemed pretty standard. However, if -1 is pass to the long version of the setters that need to handle actual (though unused) updates, wouldn't I need to "filter" a -1 so that the value remains null? That obfuscates any opportunity to actually set the value to -1.

#80 - 03/30/2020 11:19 AM - Greg Shah

Just reverse the logic. The long version calls the wrapper version.

#81 - 03/30/2020 11:23 AM - Roger Borrello

Greg Shah wrote:

> Just reverse the logic. The long version calls the wrapper version.

Can you review this implementation of TITLE-FGCOLOR?

```
/** Private storage for attributes that are not acted upon by the client */
private integer titleFgColor = null;

/**
 * Gets the TITLE-FGCOLOR writable attribute.
 *
 * @return   The current value of the TITLE-FGCOLOR attribute.
 */
@Override
public integer getTitleFgColor()
{
   /* Mimick unusual 4GL behavior */
   if (!config.popupOnly)                 // non-popup, always ? (unknown)
   {
      return new integer();
   }
   else                                   // popup GUI or ChUI
   {
      if (!LogicalTerminal.isChui())  // GUI, last assigned value or ? (unknown)
      {
         return (titleFgColor != null) ? titleFgColor : new integer();
      }
      else                                // ChUI, always ? (unknown)
      {
         return new integer();
      }
   }
}

/**
 * Sets the TITLE-FGCOLOR writable attribute.
 *
 * @param    fgcolor
 *           The new value for the TITLE-FGCOLOR attribute.
 */
@Override
public void setTitleFgColor(int64 fgcolor)
```

```
    {
        setTitleFgColor(fgcolor.isUnknown() ? -1 : fgcolor.intValue());
    }


    /**
     * Sets the TITLE-FGCOLOR writable attribute.
     *
     * @param    fgcolor
     *              The new value for the TITLE-FGCOLOR attribute.
     */
    @Override
    public void setTitleFgColor(long fgcolor)
    {
        /* Mimick unusual 4GL behavior */
        if (!config.popupOnly)               // non-popup, NOP
        {
            return;
        }
        else                                 // popup GUI or ChUI
        {
            if (!LogicalTerminal.isChui())   // GUI make non-actionable (local) assignment
            {
                titleFgColor = (fgcolor != -1) ? new integer(fgcolor) : null;
                return;
            }
            else                             // ChUI, NOP
            {
                return;
            }
        }
    }
```

#82 - 03/30/2020 11:37 AM - Greg Shah

You did not reverse the logic.

#83 - 03/30/2020 12:20 PM - Roger Borrello

Greg Shah wrote:

   You did not reverse the logic.

Like this?

```
/**
 * Sets the TITLE-FGCOLOR writable attribute.
 *
 * @param    fgcolor
 *           The new value for the TITLE-FGCOLOR attribute.
 */
@Override
public void setTitleFgColor(int64 fgcolor)
{
   /* Mimick unusual 4GL behavior */
   if (!config.popupOnly)              // non-popup, NOP
   {
      return;
   }
   else                               // popup GUI or ChUI
   {
      if (!LogicalTerminal.isChui())  // GUI make non-actionable (local) assignment
      {
         setTitleFgColor(fgcolor.isUnknown() ? -1 : fgcolor.intValue());
         return;
      }
      else                           // ChUI, NOP
      {
         return;
      }
   }
}


/**
 * Sets the TITLE-FGCOLOR writable attribute.
 *
 * @param    fgcolor
 *           The new value for the TITLE-FGCOLOR attribute.
 */
@Override
public void setTitleFgColor(long fgcolor)
{
   titleFgColor = (fgcolor != -1) ? integer(fgcolor) : null;
}
```

#84 - 03/30/2020 12:25 PM - Roger Borrello

I see the missing "new" in there.

#85 - 03/30/2020 12:52 PM - Greg Shah

No.

Just reverse the logic. The long version calls the wrapper version.

This means that setTitleFgColor(long fgcolor) should call setTitleFgColor(int64 fgcolor):

```
public void setTitleFgColor(long fgcolor)
{
    setTitleFgColor(new int64(fgcolor));
}
```

And setTitleFgColor(int64 fgcolor) should **not** call setTitleFgColor(long fgcolor). Also setTitleFgColor(int64 fgcolor) should **not** treat -1 as anything special.

#86 - 03/30/2020 01:51 PM - Roger Borrello

Doesn't the int64 interface to the converted 4GL? Perhaps that's what is confusing me. Is there a long interface needed anymore if I store the attribute locally as int64??

#87 - 03/30/2020 02:00 PM - Greg Shah

The frame definitions will use setTitle*(long). It is possible for the attribute usage to convert this way too, unless we have forced wrapping somewhere. Anyway, there are ways to use it so we put it there.

#88 - 03/30/2020 02:03 PM - Roger Borrello

Suddenly that implementation seems secondary to an issue that came up when I tried to build ./uast/browse_title_color_attributes.p. Parsing tried to pull in p2j_test.book for each browse b:... reference. Sounds similar to #4564, but the testcases specific to that issues parse fine.

I changes all references of browse b1:... by naming the browse b1, and the issue moved to ./uast/browse_title_color_attributes.p:128:21: unexpected token: browse at this line: test-dc = browse b1:title-dcolor

Had you been able to build that testcase in FWD?

#89 - 03/30/2020 02:04 PM - Greg Shah

What do you mean by "build"? It parsed fine.

#90 - 03/30/2020 02:10 PM - Roger Borrello

Greg Shah wrote:

> What do you mean by "build"? It parsed fine.

When I tried to convert the ./uast/browse_title_color_attributes.p testcase:

```
[java] ./uast/browse_title_color_attributes.p
[java] WARNING: Null annotation (name) for p2j_test.book.on-hand-qty [FIELD_INT] @0:0
[java] WARNING: Null annotation (name) for p2j_test.book.cost [FIELD_DEC] @0:0
[java] WARNING: Null annotation (name) for p2j_test.book.author-id [FIELD_INT] @0:0
[java] WARNING: Null annotation (name) for p2j_test.book.book-title [FIELD_CHAR] @0:0
[java] WARNING: Null annotation (name) for p2j_test.book.sold-qty [FIELD_INT] @0:0
[java] WARNING: Null annotation (name) for p2j_test.book.price [FIELD_DEC] @0:0
[java] WARNING: Null annotation (name) for p2j_test.book.isbn [FIELD_CHAR] @0:0
[java] WARNING: Null annotation (name) for p2j_test.book.pub-date [FIELD_DATE] @0:0
[java] WARNING: Null annotation (name) for p2j_test.book.publisher [FIELD_CHAR] @0:0
[java] WARNING: Null annotation (name) for p2j_test.book.book-id [FIELD_INT] @0:0
[java] Failure in file './uast/browse_title_color_attributes.p':
[java] com.goldencode.ast.AstException: Error processing ./uast/browse_title_color_attributes.p
[java]     at com.goldencode.p2j.uast.AstGenerator.processFile(AstGenerator.java:992)
[java]     at com.goldencode.p2j.uast.ScanDriver.lambda$scan$0(ScanDriver.java:375)
[java]     at com.goldencode.p2j.uast.ScanDriver.scan(ScanDriver.java:410)
[java]     at com.goldencode.p2j.uast.ScanDriver.scan(ScanDriver.java:248)
[java]     at com.goldencode.p2j.convert.TransformDriver.runScanDriver(TransformDriver.java:366)
[java]     at com.goldencode.p2j.convert.TransformDriver.front(TransformDriver.java:231)
[java]     at com.goldencode.p2j.convert.TransformDriver.executeJob(TransformDriver.java:894)
[java]     at com.goldencode.p2j.convert.ConversionDriver.main(ConversionDriver.java:1022)
[java] Caused by: java.lang.NullPointerException: null value for annotation 'name':p2j_test.book.cost [FI
ELD_DEC]:12884902732 @0:0 HIDDEN
```


I modified the testcase to use b1 instead of b for the browse and when I tried to convert:

```
[java] ./uast/browse_title_color_attributes.p:128:21: unexpected token: browse
[java]     at com.goldencode.p2j.uast.ProgressParser.lvalue(ProgressParser.java:17450)
[java]     at com.goldencode.p2j.uast.ProgressParser.primary_expr(ProgressParser.java:58200)
[java]     at com.goldencode.p2j.uast.ProgressParser.chained_object_members(ProgressParser.java:22128)
[java]     at com.goldencode.p2j.uast.ProgressParser.un_type(ProgressParser.java:57830)
[java]     at com.goldencode.p2j.uast.ProgressParser.prod_expr(ProgressParser.java:57697)
[java]     at com.goldencode.p2j.uast.ProgressParser.sum_expr(ProgressParser.java:41604)
[java]     at com.goldencode.p2j.uast.ProgressParser.compare_expr(ProgressParser.java:57567)
[java]     at com.goldencode.p2j.uast.ProgressParser.log_not_expr(ProgressParser.java:57135)
[java]     at com.goldencode.p2j.uast.ProgressParser.bitwise_xor_expr(ProgressParser.java:57066)
[java]     at com.goldencode.p2j.uast.ProgressParser.log_and_expr(ProgressParser.java:57005)
[java]     at com.goldencode.p2j.uast.ProgressParser.expr(ProgressParser.java:12150)
[java]     at com.goldencode.p2j.uast.ProgressParser.if_stmt(ProgressParser.java:32429)
[java]     at com.goldencode.p2j.uast.ProgressParser.stmt_list(ProgressParser.java:26362)
[java]     at com.goldencode.p2j.uast.ProgressParser.statement(ProgressParser.java:8744)
[java]     at com.goldencode.p2j.uast.ProgressParser.single_block(ProgressParser.java:7306)
[java]     at com.goldencode.p2j.uast.ProgressParser.block(ProgressParser.java:6993)
[java]     at com.goldencode.p2j.uast.ProgressParser.external_proc(ProgressParser.java:6920)
[java]     at com.goldencode.p2j.uast.AstGenerator.parse(AstGenerator.java:1524)
[java]     at com.goldencode.p2j.uast.AstGenerator.processFile(AstGenerator.java:987)
[java]     at com.goldencode.p2j.uast.ScanDriver.lambda$scan$0(ScanDriver.java:375)
[java]     at com.goldencode.p2j.uast.ScanDriver.scan(ScanDriver.java:410)
[java]     at com.goldencode.p2j.uast.ScanDriver.scan(ScanDriver.java:248)
[java]     at com.goldencode.p2j.convert.TransformDriver.runScanDriver(TransformDriver.java:366)
[java]     at com.goldencode.p2j.convert.TransformDriver.front(TransformDriver.java:231)
[java]     at com.goldencode.p2j.convert.TransformDriver.executeJob(TransformDriver.java:894)
[java]     at com.goldencode.p2j.convert.ConversionDriver.main(ConversionDriver.java:1022)
[java] Failure in file './uast/browse_title_color_attributes.p':
```

I know I have to get the implementation correct, but I wanted to report this situation, and determine what might have introduced it, if you had not received the same errors with the same testcase.

#91 - 03/30/2020 02:17 PM - Roger Borrello

Same issues occur with trunk. I'll focus on implementation, but these will prevent validation of the implementation for the time being.

#92 - 03/30/2020 03:21 PM - Roger Borrello

Currently looking at why the ./uast/menu/menu_title_color_attributes.p testcase fails in **Core Code Conversion**:

```
[java] EXPRESSION EXECUTION ERROR:
[java] -------------------------
[java] throwException(errmsg)
[java] ^  { Unsupported method or attribute KW_TITL_BGC. [COLON id <12884902126> 38:19] }
[java] -------------------------
[java] ERROR:
[java] com.goldencode.p2j.pattern.TreeWalkException: ERROR!  Active Rule:
[java] ----------------------
[java]        RULE REPORT
[java] ----------------------
[java] Rule Type :   DESCENT
[java] Source AST:  [ : ] BLOCK/ASSIGNMENT/ASSIGN/EXPRESSION/COLON/ @38:19 {12884902126}
[java] Copy AST  :  [ : ] BLOCK/ASSIGNMENT/ASSIGN/EXPRESSION/COLON/ @38:19 {12884902126}
[java] Condition :   throwException(errmsg)
[java] Loop      :   false
[java] --- END RULE REPORT ---
```

Of course a handle is not preceding the COLON, so the error is thrown in methods_attributes.rules.

#93 - 03/30/2020 03:59 PM - Roger Borrello

Roger Borrello wrote:

> Currently looking at why the ./uast/menu/menu_title_color_attributes.p testcase fails in **Core Code Conversion**:
> [...]

Of course a handle is not preceding the COLON, so the error is thrown in methods_attributes.rules.

False alarm... I was still using trunk :-(

#94 - 03/30/2020 04:14 PM - Roger Borrello

Committed updates:

```
modified rules/gaps/expressions.rules
modified src/com/goldencode/p2j/ui/BrowseWidget.java
modified src/com/goldencode/p2j/ui/MenuWidget.java
Committed revision 11407.
```


Still working on testing, but I believe these implementations are good for review.

#95 - 03/31/2020 08:39 AM - Greg Shah

Code Review Task Branch 4231b Revision 11407

1. In gaps/expressions.rules, the kw_tit* comments should be removed. There is one of the in particular (<!-- title-font, exists for frame/dialog, exists but untested for browse, missing in menu -->) which is no longer correct at all. But even the other comments are no longer useful.

2. BrowseWidget should not have a history entry 140 when history entry 139 is for this same branch.

3. Coding standards say not to use /* */ for code comments, except in specific cases. Adding stuff like /* Mimick unusual 4GL behavior */ is not one of those cases. Also, almost everything about the 4GL is unusual. Just say: // in the 4GL, GUI or ChUI, realized or not, always returns ? (unknown). Instead of /* Readonly */, say // in the 4GL, this is always read-only.

4. In BrowseWidget, you would have much less code to write if you made a common worker method for raising the 4052 error.

5. In MenuWidget each member variable needs its own javadoc.

#96 - 03/31/2020 09:03 AM - Roger Borrello

Greg Shah wrote:

> Code Review Task Branch 4231b Revision 11407

1. In gaps/expressions.rules, the kw_tit* comments should be removed. There is one of the in particular (<!-- title-font, exists for frame/dialog, exists but untested for browse, missing in menu -->) which is no longer correct at all. But even the other comments are no longer useful.

Just to be sure... you want comments removed? Isn't there any usefulness to tieing the keyword to the 4GL symbol?

#97 - 03/31/2020 09:06 AM - Greg Shah

Just to be sure... you want comments removed?

Yes.

Isn't there any usefulness to tieing the keyword to the 4GL symbol?

Only for the ones that are not supported or which have limitations. Things that are fully supported/compatible without restrictions need no comments.

#98 - 03/31/2020 09:45 AM - Roger Borrello

Greg Shah wrote:

Code Review Task Branch 4231b Revision 11407
4. In BrowseWidget, you would have much less code to write if you made a common worker method for raising the 4052 error.

I had thought about that. There seems to be quite the variety of signatures for ErrorManager.recordOrShowError (as well as recordOrThrowError). I am not sure if maybe I should be using recordOrShowWarning.

I can make a helper in there, also for the 4088 errors, but didn't want to do that without some direction.

#99 - 03/31/2020 09:52 AM - Greg Shah

Anytime you are writing the same code over and over, you can make a helper. It is ALWAYS better than copying the code everywhere.

When you have to generate a WARNING use recordOrShowError. When you have an ERROR, use recordOrThrowError. If I recall correctly, recordOrShowWarning is for an obscure special case.

#100 - 03/31/2020 10:12 AM - Roger Borrello

Greg Shah wrote:

> Anytime you are writing the same code over and over, you can make a helper. It is ALWAYS better than copying the code everywhere.
>
> When you have to generate a WARNING use recordOrShowError. When you have an ERROR, use recordOrThrowError. If I recall correctly, recordOrShowWarning is for an obscure special case.

This need to be pretty precise, in that if we miss the extra space 4GL puts between the word "the" and the widget type, testcases that may be looking for exact positions of characters on the screen would fail. In BaseEntity, there are 2 instances of 4088 which are very different from each other. One uses displayWarning for **PFCOLOR**, the other uses recordOrShowError for **FONT**, neither using a helper. With the benefit of the helper, you'd have more consistency.

If I create a helper, it should be utilized for all applicable cases, but the risk is changing something that has no other need for change.

#101 - 03/31/2020 10:14 AM - Greg Shah

For now, I'm just suggesting that you create a helper for BrowseWidget, so make common code instead of the duplicated code. We will rationalize the larger set of error processing at a later time.

We need to move on from this task.

#102 - 03/31/2020 10:55 AM - Roger Borrello

Greg Shah wrote:

> For now, I'm just suggesting that you create a helper for BrowseWidget, so make common code instead of the duplicated code. We will rationalize the larger set of error processing at a later time.
>
> We need to move on from this task.

In GenericWidget there is already a helper called notQueryable that does the trick. You must have been through this already.

#103 - 03/31/2020 11:31 AM - Roger Borrello

Code Review updates made:

```
modified rules/gaps/expressions.rules
modified src/com/goldencode/p2j/ui/BrowseWidget.java
modified src/com/goldencode/p2j/ui/FrameWidget.java
modified src/com/goldencode/p2j/ui/MenuWidget.java
Committed revision 11410.
```

#104 - 03/31/2020 11:43 AM - Roger Borrello

Converting the browse_title_color_options.p testcase, setTitleColor is unresolved. Did that method ne get overlooked in the Title Phrase udpates?

#105 - 03/31/2020 11:52 AM - Greg Shah

Roger Borrello wrote:

> Converting the browse_title_color_options.p testcase, setTitleColor is unresolved. Did that method ne get overlooked in the Title Phrase udpates?

Compile error? What is the failure?

#106 - 03/31/2020 11:55 AM - Roger Borrello

Compilation of the testcase:

```
compile:
    [javac] Compiling 63 source files to /home/rfb/projects/VirtualBox-VMs/shared/projects/testcases/build/cla
sses
    [javac] /home/rfb/projects/VirtualBox-VMs/shared/projects/testcases/src/com/goldencode/testcases/ui/Browse
TitleColorOptionsF18.java:35: error: cannot find symbol
    [javac]           b18.setTitleColor(new ColorSpec(ColorSpec.COLOR_INPUT));
    [javac]              ^
    [javac]   symbol:   method setTitleColor(ColorSpec)
    [javac]   location: variable b18 of type BrowseWidget
    [javac] /home/rfb/projects/VirtualBox-VMs/shared/projects/testcases/src/com/goldencode/testcases/ui/Browse
TitleColorOptionsF9.java:35: error: cannot find symbol
    [javac]           b9.setTitleColor(new ColorSpec(ColorSpec.COLOR_INPUT));
    [javac]             ^
    [javac]   symbol:   method setTitleColor(ColorSpec)
    [javac]   location: variable b9 of type BrowseWidget
    [javac] 2 errors
```

It looks like setTitleColor needs to be implemented. I have added it to BrowseWidget and I believe the interface is BrowseInterface. Just need to get the annotation correct. Sound like the right places?

#107 - 03/31/2020 12:07 PM - Greg Shah

It looks like setTitleColor needs to be implemented. I have added it to BrowseWidget and I believe the interface is BrowseInterface. Just need to get the annotation correct. Sound like the right places?

Yes.

What happens in trunk?

#108 - 03/31/2020 12:14 PM - Roger Borrello

Greg Shah wrote:

What happens in trunk?

```
compile:
    [javac] Compiling 63 source files to /home/rfb/projects/VirtualBox-VMs/shared/projects/testcases/build/cla
sses
    [javac] /home/rfb/projects/VirtualBox-VMs/shared/projects/testcases/src/com/goldencode/testcases/ui/Browse
TitleColorOptionsF18.java:35: error: no suitable method found for setDColor(ColorSpec)
    [javac]             b18.setDColor(new ColorSpec(ColorSpec.COLOR_INPUT));
    [javac]                ^
    [javac]     method CommonWidget.setDColor(Color) is not applicable
    [javac]       (argument mismatch; ColorSpec cannot be converted to Color)
    [javac]     method CommonWidget.setDColor(NumberType) is not applicable
    [javac]       (argument mismatch; ColorSpec cannot be converted to NumberType)
    [javac]     method GenericWidget.setDColor(NumberType) is not applicable
    [javac]       (argument mismatch; ColorSpec cannot be converted to NumberType)
    [javac]     method GenericWidget.setDColor(Color) is not applicable
    [javac]       (argument mismatch; ColorSpec cannot be converted to Color)
    [javac]     method BaseEntity.setDColor(NumberType) is not applicable
    [javac]       (argument mismatch; ColorSpec cannot be converted to NumberType)
    [javac]     method BaseEntity.setDColor(Color) is not applicable
    [javac]       (argument mismatch; ColorSpec cannot be converted to Color)
    [javac]     method BaseEntity.setDColor(int) is not applicable
    [javac]       (argument mismatch; ColorSpec cannot be converted to int)
    [javac]     method BrowseWidget.setDColor(int64) is not applicable
    [javac]       (argument mismatch; ColorSpec cannot be converted to int64)
    [javac] /home/rfb/projects/VirtualBox-VMs/shared/projects/testcases/src/com/goldencode/testcases/ui/Browse
TitleColorOptionsF9.java:35: error: no suitable method found for setDColor(ColorSpec)
    [javac]             b9.setDColor(new ColorSpec(ColorSpec.COLOR_INPUT));
    [javac]                ^
    [javac]     method CommonWidget.setDColor(Color) is not applicable
    [javac]       (argument mismatch; ColorSpec cannot be converted to Color)
    [javac]     method CommonWidget.setDColor(NumberType) is not applicable
    [javac]       (argument mismatch; ColorSpec cannot be converted to NumberType)
    [javac]     method GenericWidget.setDColor(NumberType) is not applicable
    [javac]       (argument mismatch; ColorSpec cannot be converted to NumberType)
    [javac]     method GenericWidget.setDColor(Color) is not applicable
    [javac]       (argument mismatch; ColorSpec cannot be converted to Color)
    [javac]     method BaseEntity.setDColor(NumberType) is not applicable
    [javac]       (argument mismatch; ColorSpec cannot be converted to NumberType)
    [javac]     method BaseEntity.setDColor(Color) is not applicable
    [javac]       (argument mismatch; ColorSpec cannot be converted to Color)
    [javac]     method BaseEntity.setDColor(int) is not applicable
    [javac]       (argument mismatch; ColorSpec cannot be converted to int)
    [javac]     method BrowseWidget.setDColor(int64) is not applicable
```

```
    [javac]        (argument mismatch; ColorSpec cannot be converted to int64)
    [javac] 2 errors
```

#109 - 03/31/2020 12:22 PM - Greg Shah

So setTitleColor(ColorSpec) was just missing. My changes fixed the conversion, but you will need to add the runtime changes as you documented.

#110 - 03/31/2020 12:42 PM - Roger Borrello

Done, but in my testing, I see the testcase in FWD displays 4052 in a popup for every browse widget. Either I need to set the error without a display, or perhaps the set is allowed before the browse is realized.

#111 - 03/31/2020 01:11 PM - Greg Shah

> I see the testcase in FWD displays 4052 in a popup for every browse widget.

Be more specific. In which testscase and for which lines of code is this happening? GUI, ChUI? What is the corresponding behavior in the 4GL?

#112 - 03/31/2020 01:45 PM - Roger Borrello

Greg Shah wrote:

> Be more specific. In which testscase and for which lines of code is this happening? GUI, ChUI? What is the corresponding behavior in the
> 4GL?

In GUI, before anything is realized, a series of popup errors display. Not so in 4GL.

```
TITLE-FONT is not a setable attribute for BROWSE b18. (4052)
TITLE-DCOLOR is not a setable attribute for BROWSE b17. (4052)
TITLE-FONT is not a setable attribute for BROWSE b17. (4052)
TITLE-DCOLOR is not a setable attribute for BROWSE b16. (4052)
TITLE-FONT is not a setable attribute for BROWSE b16. (4052)
TITLE-FGCOLOR is not a setable attribute for BROWSE b15. (4052)
TITLE-FONT is not a setable attribute for BROWSE b15. (4052)
.
.
.
```

#113 - 03/31/2020 02:16 PM - Roger Borrello

- File GUI_browse_title_color_options1.png added

Another issue I am looking at. The browse title is missing, if DCOLOR is part of the options:
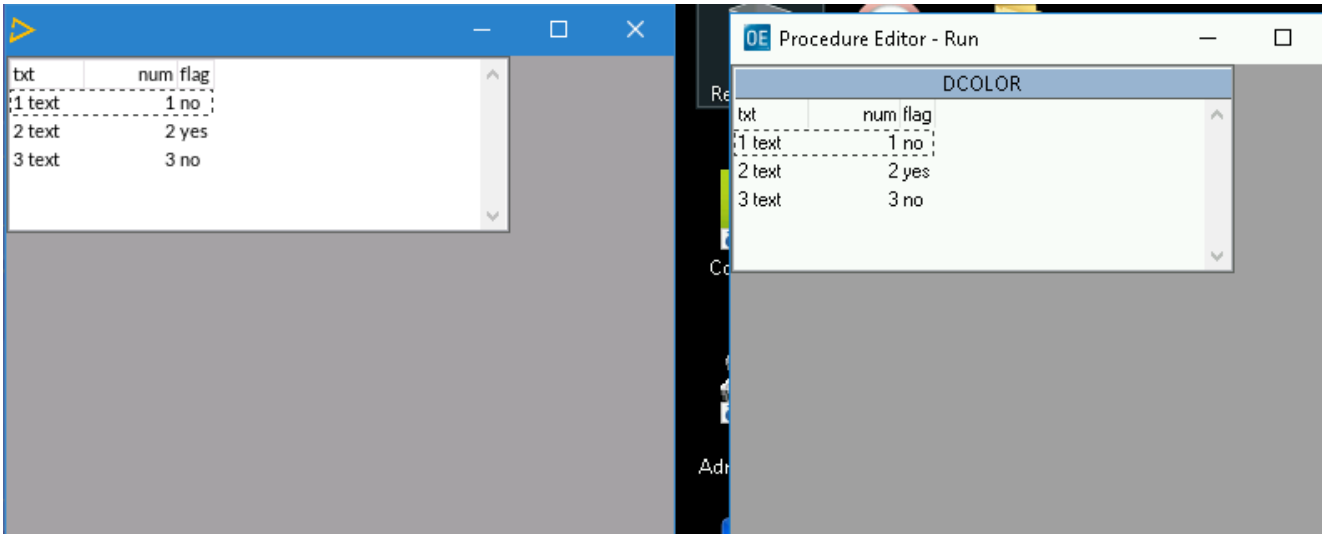
Going through all the testcases, only the ones including DCOLOR show that symptom.

#114 - 03/31/2020 02:46 PM - Greg Shah

What testcase are you using? The code in testcases/uast/browse_title_color_attributes.p doesn't look anything like your example from #4393-112.

#115 - 03/31/2020 03:11 PM - Greg Shah

I guess you are referencing testcases/uast/browse_title_color_options.p. Options are not attributes. The text above in #4393-3 should be clear: they should not generate an error.

#116 - 03/31/2020 03:12 PM - Roger Borrello

Greg Shah wrote:

> What testcase are you using? The code in testcases/uast/browse_title_color_attributes.p doesn't look anything like your example from #4393-112.

#4393-112 was a collection of error message before I added suppression of the notSettable() message via a check for config.realized. I was trying to

move past that issue, to verify the other functionality.

At the start, I thought the Title phrase items had been closed out, but it seems like the attribute methods I added are being called from Title phrases.

#117 - 03/31/2020 03:13 PM - Roger Borrello

Greg Shah wrote:

> I guess you are referencing testcases/uast/browse_title_color_options.p. Options are not attributes. The text above in [#4393-3](#) should be clear: they should not generate an error.

Yes, it the "options" test.

#118 - 03/31/2020 03:18 PM - Greg Shah

Not sure the realized flag is correct.

See the gen_color in frame_generator.xml. It should only be generating code for TITLE DCOLOR <num> "title", not other forms. You can modify the code there to use a different method in BrowseWidget if needed.

#119 - 03/31/2020 03:45 PM - Roger Borrello

Greg Shah wrote:

> See the gen_color in frame_generator.xml. It should only be generating code for TITLE DCOLOR <num> "title", not other forms. You can modify the code there to use a different method in BrowseWidget if needed.

So I should never see in the frame definition, the use of .setTitleColor, .setTitleFGColor, .setTitleBGColor on the browse widget (what about .setTitleFont)?

Where should the implementations reside? It seems like attribute methods are being called in some cases. Test 15:

```
def browse b15 query q15 no-lock no-wait
    display test
    with 5 down width 60 title fgcolor 2 font 2 "FGCOLOR, FONT".
```

generates

```
    public void setup(CommonFrame frame)
    {
        frame.setDown(1);
        b15.setDown(5);
        b15.setWidthChars(60);
        b15.setTitleFgColor(2);
        b15.setTitleFont(2);
        b15.setTitle("FGCOLOR, FONT");
```

#120 - 03/31/2020 04:02 PM - Greg Shah

No, looking back on things they are all possible. I didn't update the comment in frame_generator.xml but the notes in #4393-3 are clear.

There is a flag (GenericFrame.isInsideSetup()) which is only true when processing the frame definition. We use that in GenericFrame to protect some processing during that phase. If the browse was already connected to its frame, you could read that flag and just return immediately.

#121 - 03/31/2020 05:05 PM - Roger Borrello

- File GUI_browse_title_color_default_values1.png added

Greg Shah wrote:

> No, looking back on things they are all possible. I didn't update the comment in frame_generator.xml but the notes in #4393-3 are clear.

Should I remove the comment: "For the browse options phrase, only the TITLE DCOLOR <num> "title" form can be used..." ?

> There is a flag (GenericFrame.isInsideSetup()) which is only true when processing the frame definition. We use that in GenericFrame to protect some processing during that phase. If the browse was already connected to its frame, you could read that flag and just return immediately.

I did not have good results with that flag, or I was using it incorrectly. However, the config.realized flag is used all over BrowseWidget, so unless you think that's blatantly wrong, I'll leave it.

My change was to push the pre-realized value for DCOLOR to the client. However, the behavior of not displaying the title at all did not change. I can move on from this, however, the browse_title_color_default_values.p displays something that contradicts the initial values specified in #4393-4, namely that TITLE-FONT is always ? (unknown value) before and after realization

I'm not sure if that's correct, given what I see in 4GL. The last value is the b0:title-font value. It looks like it should be 0, unless I am misreading
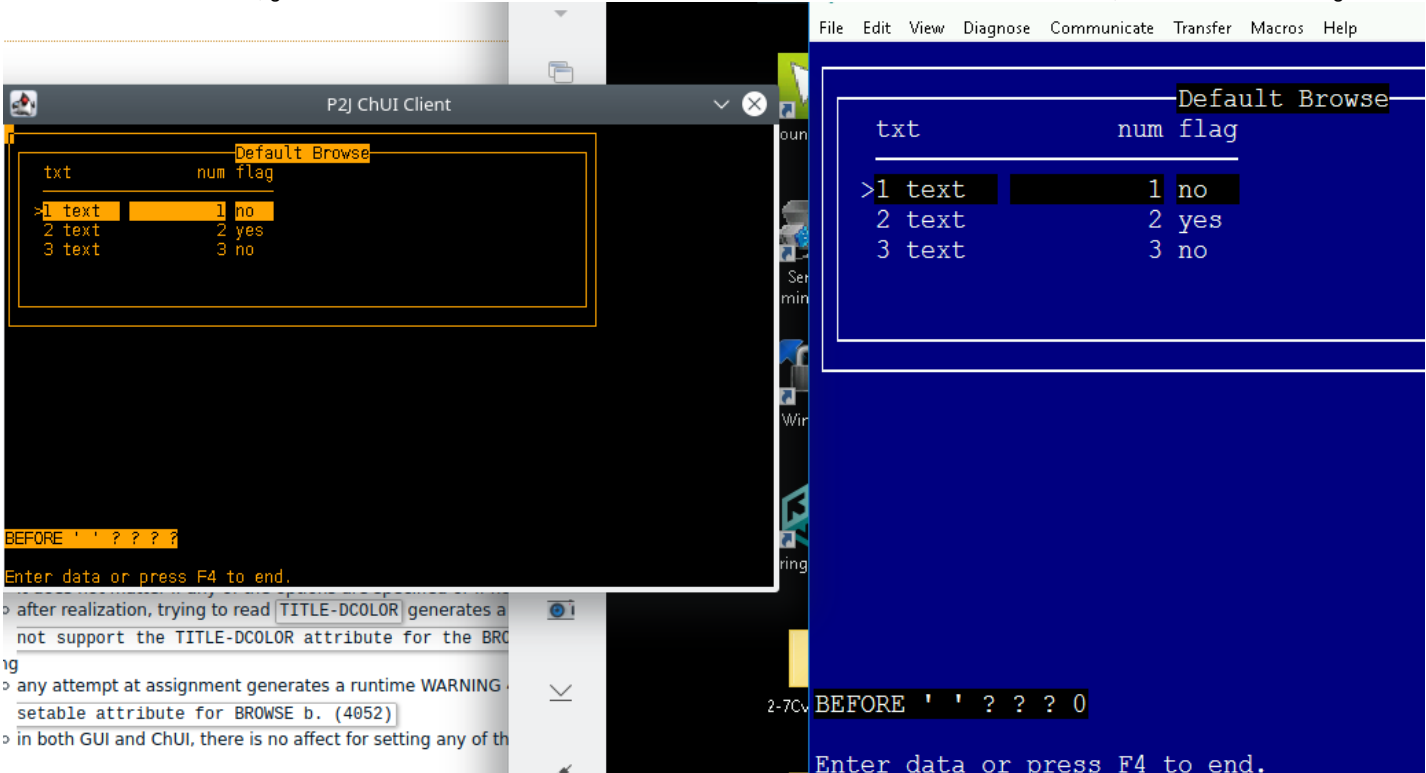
#122 - 03/31/2020 07:00 PM - Roger Borrello

Made updates in Rev11413. I was attempting to get the title to appear when DCOLOR was part of the title phrase. This is still not displaying.

The testcases, with the exception of browse_title_color_attributes.p are passing.

#123 - 03/31/2020 08:09 PM - Greg Shah

Should I remove the comment: "For the browse options phrase, only the TITLE DCOLOR <num> "title" form can be used..." ?

Well, at least edit it to be accurate.

I did not have good results with that flag, or I was using it incorrectly. However, the config.realized flag is used all over BrowseWidget, so unless you think that's blatantly wrong, I'll leave it.

OK. Put in a comment like this:

```
// when this is called for use as a browse option (instead of a widget attribute), the realized
// flag is used to differentiate the two cases
```

My change was to push the pre-realized value for DCOLOR to the client. However, the behavior of not displaying the title at all did not change. I can move on from this, however, the browse_title_color_default_values.p displays something that contradicts the initial values specified in #4393-4, namely that TITLE-FONT is always ? (unknown value) before and after realization

If you are finding any differences from my notes, please make a specific list here.

#124 - 04/01/2020 10:53 AM - Roger Borrello

Browse TITLE Attribute Defaults (before/after realization)

	ChUI (bef)	ChUI (aft)	GUI (bef)	GUI (aft)
BGCOLOR	?	?	?	?
DCOLOR	?	*	?	?
FGCOLOR	?	?	?	?
FONT	0	?	0	0

* - The attempt to read this attribute results in 4088

#125 - 04/01/2020 12:22 PM - Roger Borrello

I was re-running frame_title_color_attributes.p to try to close this task down, and I was running on OE release 10. I received a popup error I wasn't expecting with 4088. I went back to OE release 11, and no error popped. It just ran fine. Is this a difference between releases, or perhaps a configuration customization like terminal type factoring in? I don't want to second guess all the results, so I need to know how to determine the differences.

#126 - 04/01/2020 01:58 PM - Greg Shah

Yes, it could be a version difference. Run the testcases on both 10.x and 11.x and record the differences only. Then we can make a plan.

#127 - 04/01/2020 06:54 PM - Roger Borrello

- Related to Bug #4604: DCOLOR on FRAME or BROWSE TITLE phrase results in title not being displayed added

#128 - 04/16/2020 10:27 AM - Roger Borrello

Greg Shah wrote:

> I did not have good results with that flag, or I was using it incorrectly. However, the config.realized flag is used all over BrowseWidget, so unless you think that's blatantly wrong, I'll leave it.

OK. Put in a comment like this:

[...]

I will have to revisit the use of config.realized, since it creates mutual destruction between setting the values for attributes versus options. In both cases, they should result in 4052 read-only errors, but now I've prevented the error prior to realization for the attribute case.

As you said, GenericFrame.isInsideSetup() is true when processing the frame definition. Can you confirm this is the case when we are dealing with the options for a define browse ... title <options> ?
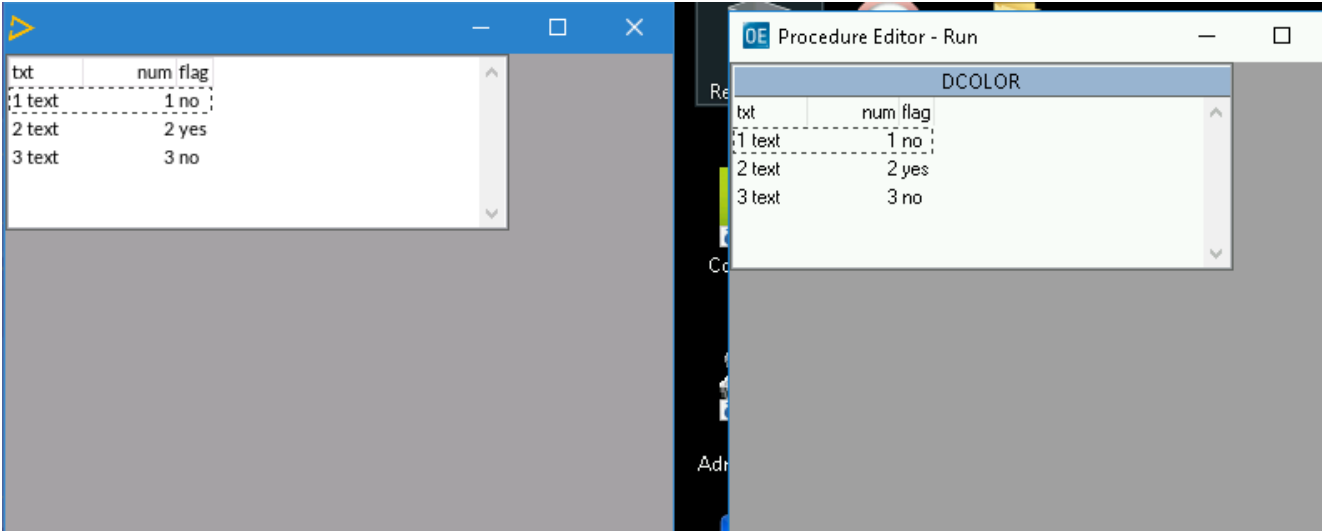
#129 - 04/16/2020 10:59 AM - Greg Shah

Yes, I think so.

#130 - 04/16/2020 11:37 AM - Roger Borrello

Roger Borrello wrote:

> Another issue I am looking at. The browse title is missing, if DCOLOR is part of the options:

> Going through all the testcases, only the ones including DCOLOR show that symptom.

I am trying to fix this issue, but I'm not sure whether it is server or client. Any hints?

#131 - 04/16/2020 01:49 PM - Greg Shah

It is not obviously on the client or server, we can't really know. But since it is drawing something and just missing part of the drawing, the best place to start is to set a breakpoint in the com.goldencode.p2j.ui.client.gui.FrameGuiImpl.draw() and step through the title processing code.

#132 - 04/16/2020 02:34 PM - Roger Borrello

Issue of the title bar missing was related to annotations/sorting.rules where DCOLOR was not included as criteria for moving the color spec to the end of the line.

#133 - 04/16/2020 04:40 PM - Roger Borrello

These changes should complete the browse attributes and options.


```
Committing to: /home/rfb/secure/code/p2j_repo/p2j/active/4231b/

modified rules/annotations/sorting.rules
modified rules/convert/frame_generator.xml
modified src/com/goldencode/p2j/ui/BrowseWidget.java
Committed revision 1146
```


#134 - 04/16/2020 06:03 PM - Roger Borrello

Regarding the menu tests, both GUI and ChUI result in the same attribute output.

However, there are different behaviors between FWD and 4GL with related to interaction with the user. For example, in menu_title_color_attributes.p, a mouse-click on the button does not do anything in FWD, while in 4GL, it activates the button and an NPE exception is thrown:


```
DRAW: Ignoring event com.goldencode.p2j.ui.client.event.PaintEvent 9 raised while drawing widget com.goldencod
e.p2j.ui.client.widget.Viewport with id [-1].
DRAW: Ignoring event com.goldencode.p2j.ui.client.event.PaintEvent 9 raised while drawing widget com.goldencod
e.p2j.ui.client.widget.Viewport with id [-1].
DRAW: Ignoring event com.goldencode.p2j.ui.client.event.PaintEvent 9 raised while drawing widget com.goldencod
e.p2j.ui.client.widget.Viewport with id [-1].
DRAW: Ignoring event com.goldencode.p2j.ui.client.event.PaintEvent 9 raised while drawing widget com.goldencod
e.p2j.ui.client.widget.Viewport with id [-1].
Exception in thread "AWT-EventQueue-1" java.lang.NullPointerException
    at java.util.concurrent.LinkedBlockingQueue.offer(LinkedBlockingQueue.java:411)
    at java.util.AbstractQueue.add(AbstractQueue.java:95)
    at com.goldencode.p2j.ui.client.driver.swing.LinuxKeyboardReader.keyPressed(LinuxKeyboardReader.java:209)
    at java.awt.Component.processKeyEvent(Component.java:6497)
    at java.awt.Component.processEvent(Component.java:6316)
    at java.awt.Container.processEvent(Container.java:2239)
    at java.awt.Window.processEvent(Window.java:2025)
    at java.awt.Component.dispatchEventImpl(Component.java:4889)
    at java.awt.Container.dispatchEventImpl(Container.java:2297)
    at java.awt.Window.dispatchEventImpl(Window.java:2746)
    at java.awt.Component.dispatchEvent(Component.java:4711)
    at java.awt.KeyboardFocusManager.redispatchEvent(KeyboardFocusManager.java:1954)
    at java.awt.DefaultKeyboardFocusManager.dispatchKeyEvent(DefaultKeyboardFocusManager.java:834)
    at java.awt.DefaultKeyboardFocusManager.preDispatchKeyEvent(DefaultKeyboardFocusManager.java:1102)
    at java.awt.DefaultKeyboardFocusManager.typeAheadAssertions(DefaultKeyboardFocusManager.java:973)
    at java.awt.DefaultKeyboardFocusManager.dispatchEvent(DefaultKeyboardFocusManager.java:799)
    at java.awt.Component.dispatchEventImpl(Component.java:4760)
    at java.awt.Container.dispatchEventImpl(Container.java:2297)
    at java.awt.Window.dispatchEventImpl(Window.java:2746)
    at java.awt.Component.dispatchEvent(Component.java:4711)
    at java.awt.EventQueue.dispatchEventImpl(EventQueue.java:760)
    at java.awt.EventQueue.access$500(EventQueue.java:97)
    at java.awt.EventQueue$3.run(EventQueue.java:709)
    at java.awt.EventQueue$3.run(EventQueue.java:703)
    at java.security.AccessController.doPrivileged(Native Method)
    at java.security.ProtectionDomain$JavaSecurityAccessImpl.doIntersectionPrivilege(ProtectionDomain.java:74)
    at java.security.ProtectionDomain$JavaSecurityAccessImpl.doIntersectionPrivilege(ProtectionDomain.java:84)
    at java.awt.EventQueue$4.run(EventQueue.java:733)
    at java.awt.EventQueue$4.run(EventQueue.java:731)
```

```
        at java.security.AccessController.doPrivileged(Native Method)
        at java.security.ProtectionDomain$JavaSecurityAccessImpl.doIntersectionPrivilege(ProtectionDomain.java:74)
        at java.awt.EventQueue.dispatchEvent(EventQueue.java:730)
        at org.GNOME.Accessibility.AtkWrapper$6.dispatchEvent(AtkWrapper.java:715)
        at java.awt.EventDispatchThread.pumpOneEventForFilters(EventDispatchThread.java:205)
        at java.awt.EventDispatchThread.pumpEventsForFilter(EventDispatchThread.java:116)
        at java.awt.EventDispatchThread.pumpEventsForHierarchy(EventDispatchThread.java:105)
        at java.awt.EventDispatchThread.pumpEvents(EventDispatchThread.java:101)
        at java.awt.EventDispatchThread.pumpEvents(EventDispatchThread.java:93)
        at java.awt.EventDispatchThread.run(EventDispatchThread.java:82)
DRAW: Ignoring event com.goldencode.p2j.ui.client.event.PaintEvent 9 raised while drawing widget com.goldencod
e.p2j.ui.client.widget.Viewport with id [-1].
DRAW: Ignoring event com.goldencode.p2j.ui.client.event.PaintEvent 9 raised while drawing widget com.goldencod
e.p2j.ui.client.widget.Viewport with id [-1].
Apr 16, 2020 6:01:27 PM com.goldencode.p2j.ui.client.event.EventManager postEvent
WARNING: Wasted event: com.goldencode.p2j.ui.client.event.PaintEvent@e1396a0
```

Any clues where to start?

#135 - 04/17/2020 07:07 AM - Greg Shah

> while in 4GL, it activates the button and an NPE exception is thrown:

This implies there is an NPE in the 4GL which I know is not possible. So I think you've just worded this in a confusing way.

We can't have to spend much time (if any) on this. However, there is only 1 method involved in the stack trace which is in FWD. Setting a breakpoint there seems like the first step.

#136 - 04/17/2020 08:48 AM - Greg Shah

Code Review Task Branch 4231b Revision 11463

The changes are good.

A minor code standards issue: split the following into 2 lines. This:

```
if (!frame.isInsideSetup()) notSettable("TITLE-FONT"); // Read-only
```

should be this:

```
if (!frame.isInsideSetup())
{
    notSettable("TITLE-FONT"); // read-only
}
```

This appears in multiple places.

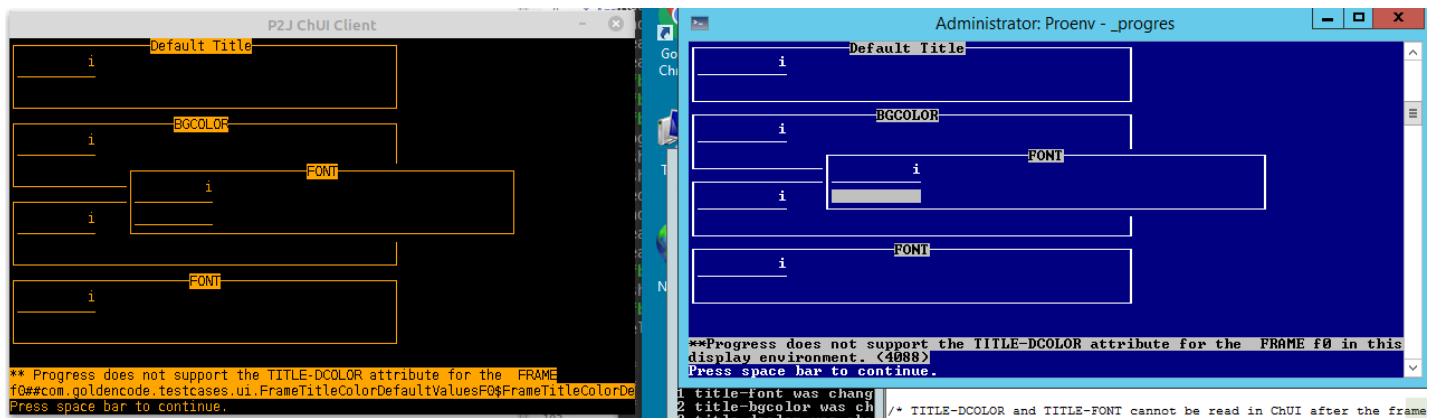**#137 - 04/17/2020 09:13 AM - Roger Borrello**

- File GUI_frame_title_color_default_values3.png added

The frame_title_color_default_values.p tests pass, however, there are a couple of differences.

1. There is space in FWD for a 5th frame in the default window. In 4GL, there is only 4.
2. The title on the dialog is left justified in FWD, but centered in 4GL.
   Should these be pursued?
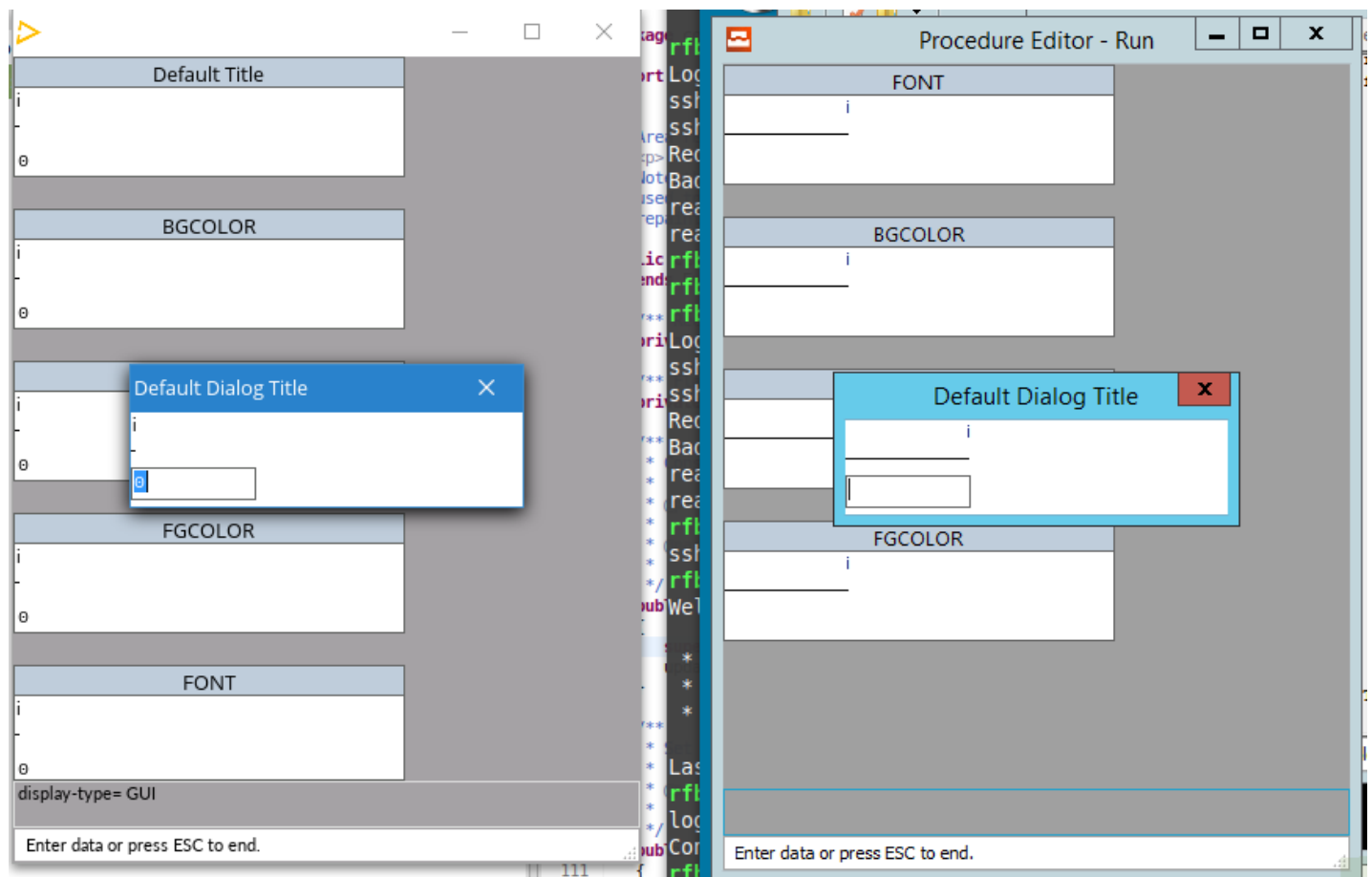
In ChUI, all things are equal... except error test is mangled. I will look into this, since it's probably very simple.

#138 - 04/17/2020 09:14 AM - Roger Borrello

- File ChUI_frame_title_color_default_values1.png added

#139 - 04/17/2020 09:33 AM - Greg Shah

Should these be pursued?

No. Create new tasks.

#140 - 04/17/2020 10:29 AM - Roger Borrello

- Related to Bug #4617: Number of GUI frames in default window differs from 4GL added

#141 - 04/17/2020 11:09 AM - Roger Borrello

- Related to Bug #4618: Default location of dialog box title is left justified instead of centered added

#142 - 04/17/2020 11:18 AM - Roger Borrello

- Related to Bug #4619: GUI fill-in widgets show in FWD, when 4GL shows nothing added

#143 - 04/17/2020 11:55 AM - Roger Borrello

Greg Shah wrote:

Yes, it could be a version difference. Run the testcases on both 10.x and 11.x and record the differences only. Then we can make a plan.

The ChUI frame default values pre-realization in 10.2B are as set forth in the testcase. In 11.6.3 they are different.

	10.2B (pre)	10.2B (post)	11.6.3 (pre)	11.6.3 (post)
BGCOLOR	?	?	?	?
FGCOLOR	?	?	?	?
DCOLOR	0	*	0	*
COLOR	0	*	0	*
FONT	0	*	? (expected 0)	*

* - The attempt to read this attribute results in 4088

What made this particularly challenging is that the 4GL code would not proceed once a testcase returned false. In order to get through all of them, the function that validated the values had to return true for the next test:

```
rc = rc and check-attributes(frame f0:handle,  1,  ?, ?, 0, 0, 0, false).
rc = rc and check-attributes(frame f1:handle,  2,  ?, ?, 0, 0, 0, false).
...
```

So after getting through all the tests, it looks like TITLE-FONT needs to be ? when dealing with the newer version of ChUI. Perhaps we should default it to that, and change to 0 on an older version?

#144 - 04/17/2020 11:56 AM - Roger Borrello

- Status changed from New to WIP

- % Done changed from 0 to 90

#145 - 04/17/2020 12:43 PM - Greg Shah

We have EnvironmentOps.versionWorker(long lvl) which can determine a version string for the 4GL version to which we are compatible. Primarily it is being used to back the PROVERSION 4GL built-in function.

One drawback of using this versionWorker() is that it reads the value from the directory every time. Instead, for performance reasons, we should cache this value the first time it is read. The EnvironmentOps.WorkArea is the place to cache it. If it is never read, then it is never cached. But as soon as it is read once, it should be cached since it will not change for the lifetime of the session. It should be stored as a string so that we can obtain

the exact answer multiple times. We would simply wrap it in a character before returning it.

I think we also should provide a way to return a simpler answer for some common cases. Instead of requiring the caller to parse/process a string, we should offer some simple helper methods public boolean isV9(), public boolean isV10(), public boolean isV11() and public int versionNumber(). We can additionally cache the version as a simple int (e.g. 11.3) and then implement these helpers with very simple math ops. We can use Java types here because it will never be called directly from converted code.

With this, you can detect the version and implement the differential behavior. I agree that you can default it to the v11 version and then detect if it is different.

Constantin: What was the code that you added recently which had to check the version numbers at runtime? We should ensure that it is reworked to use this same facility (adding any feature needed for that to work).

#146 - 04/17/2020 01:38 PM - Constantin Asofiei

Greg Shah wrote:

> Constantin: What was the code that you added recently which had to check the version numbers at runtime? We should ensure that it is reworked to use this same facility (adding any feature needed for that to work).

See LogicalTerminal.toMilliseconds

#147 - 04/17/2020 02:03 PM - Greg Shah

With my proposed change, the code could be implemented with a simple check EnvironmentOps.versionNumber() >= 11.3. The lt.useMillis would not be needed because the EnvironmentOps would already cache the version number.

Of course the "simple" version number would need to be in the form major.minor and the other parts would be dropped since they cannot be represented in an int.

#148 - 04/17/2020 06:36 PM - Roger Borrello

Just *temporarily* I updated to utilize the same method as toMillis.

By updating directory.xml:

```
    <node class="container" name="server">
      <node class="container" name="default">
        <node class="container" name="runtime">
          <node class="container" name="default">
            <node class="string" name="version">
              <node-attribute name="value" value="10.2"/>
```

This is more like a mini-documentation of how to perform the validation test. Of course the testcase would need to be adjusted so that it could handle

multiple versions of emulation.

I have updated the testcase to determine the correct value for the font to use for checking.

#149 - 04/27/2020 01:04 PM - Roger Borrello

Greg Shah wrote:

> We have EnvironmentOps.versionWorker(long lvl) which can determine a version string for the 4GL version to which we are compatible. Primarily it is being used to back the PROVERSION 4GL built-in function.
>
> One drawback of using this versionWorker() is that it reads the value from the directory every time. Instead, for performance reasons, we should cache this value the first time it is read. The EnvironmentOps.WorkArea is the place to cache it. If it is never read, then it is never cached. But as soon as it is read once, it should be cached since it will not change for the lifetime of the session. It should be stored as a string so that we can obtain the exact answer multiple times. We would simply wrap it in a character before returning it.
>
> I think we also should provide a way to return a simpler answer for some common cases. Instead of requiring the caller to parse/process a string, we should offer some simple helper methods public boolean isV9(), public boolean isV10(), public boolean isV11() and public int versionNumber(). We can additionally cache the version as a simple int (e.g. 11.3) and then implement these helpers with very simple math ops. We can use Java types here because it will never be called directly from converted code.

The versionWorker can be called with short (lvl=0) or verbose (lvl=1) to retrieve a shorter or longer version, such as 10.2B instead of 11.6.3.0.1407. In looking at caching the version, I will need to cached both types in the WorkArea.

#150 - 04/27/2020 01:20 PM - Roger Borrello

Greg Shah wrote:

> Of course the "simple" version number would need to be in the form major.minor and the other parts would be dropped since they cannot be represented in an int.

You mean float don't you?

#151 - 04/27/2020 02:19 PM - Greg Shah

Yes, float.

#152 - 04/27/2020 03:43 PM - Roger Borrello

Constantin Asofiei wrote:

> Greg Shah wrote:
>
>> Constantin: What was the code that you added recently which had to check the version numbers at runtime? We should ensure that it is reworked to use this same facility (adding any feature needed for that to work).
>
> See LogicalTerminal.toMilliseconds

Constantin, do you already have a testcase that exercises toMilliseconds so I can verify my version updates are good?

#153 - 04/27/2020 04:01 PM - Roger Borrello

Testcase ./uast/proversion_builtin.p was used to verify the caching in WorkArea. Now verifying utilization in ./uast/frame_title_color_default_values.p is valid, since that is where I had been using the same methods as LogicalTerminal.toMilliseconds. Now the FrameWidget.getTitleFont implementation uses EnvironmentOps.versionNumber() >= 11.0f;

For toMilliseconds, this code:

```
character legacyVersion = EnvironmentOps.getVersion();
if (legacyVersion.isUnknown())
{
   lt.useMillis = false;
}
else
{
   String ver = legacyVersion.toStringMessage();
   int dotIdx = ver.indexOf('.');
   String major = dotIdx < 0 ? ver : ver.substring(0, dotIdx);
   String minor = dotIdx < 0 ? "" : ver.substring(dotIdx + 1);

   lt.useMillis = major.length() >= 2        &&
                  major.compareTo("11") >= 0 &&
                  minor.compareTo("3") >= 0;
}
```

Will become:

```
lt.useMillis = EnvironmentOps.versionNumber() >= 11.3f;
```

#154 - 04/27/2020 05:13 PM - Constantin Asofiei

Roger Borrello wrote:

> Constantin, do you already have a testcase that exercises toMilliseconds so I can verify my version updates are good?

No, just use a pause 0.2 for a 200 millisecond pause (or a smaller value) and see what happens.

#155 - 04/27/2020 06:29 PM - Roger Borrello

- % Done changed from 90 to 100

Thanks. Enhancements have been added to 4231b-11492 and tested. I added ./uast/pause_millis.p for helping to that end.

This task is ready for test.

#156 - 04/28/2020 09:58 AM - Greg Shah

Code Review Task Branch 4231b Revision 11492

In Java, instances of String are immutable. Please never use new String(some_String_reference). This is unnecessary and is bad form.

```
    wa.versionShort  = (idx != -1) ? new String(version.substring(0, idx)) : new String(version);
    wa.versionVerbose = new String(version);
... and more below ...
```

#157 - 04/28/2020 10:52 AM - Roger Borrello

Code revised:

```
Committing to: /home/rfb/secure/code/p2j_repo/p2j/active/4231b/

modified src/com/goldencode/p2j/util/EnvironmentOps.java
Committed revision 11494.
```

#158 - 04/28/2020 11:24 AM - Greg Shah

Code Review Task Branch 4231b Revision 11494

The changes are good.

#159 - 04/28/2020 11:25 AM - Greg Shah

- Status changed from WIP to Test

#160 - 06/20/2020 12:00 PM - Greg Shah

- Status changed from Test to Closed

Task branch 4231b has been merged to trunk as revision 11347.

Files

ges_4393_title_color_and_font_attributes_and_options_support_built_4231b_rev_10324_20200324.zip	62.4 KB	03/24/2020	Greg Shah
title-bgcolor_not_setable.png	12.8 KB	03/26/2020	Roger Borrello
ChUI_frame_title_color_attributes1.png	47.9 KB	03/27/2020	Roger Borrello
ChUI_frame_title_color_attributes2.png	48 KB	03/27/2020	Roger Borrello
GUI_frame_title_color_default_values2.png	21 KB	03/27/2020	Roger Borrello
GUI_browse_title_color_options1.png	10.1 KB	03/31/2020	Roger Borrello
GUI_browse_title_color_default_values1.png	54.2 KB	03/31/2020	Roger Borrello
GUI_frame_title_color_default_values3.png	45.3 KB	04/17/2020	Roger Borrello
ChUI_frame_title_color_default_values1.png	24.4 KB	04/17/2020	Roger Borrello