

## User Interface - Bug #6016

### Selection list setScreenValue improvements

01/28/2022 03:28 AM - Marian Edu

<b>Status:</b>	Closed	<b>Start date:</b>	
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	Marian Edu	<b>% Done:</b>	100%
<b>Category:</b>		<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>		<b>case_num:</b>	
<b>billable:</b>	No	<b>version:</b>	
<b>vendor_id:</b>	GCD		
<b>Description</b>			
Related to User Interface - Bug #6038: ComboBox setScreenValue improvements <a href="#">Feedback</a>			

### History

#### #1 - 01/28/2022 05:48 AM - Greg Shah

- Description updated

From Marian:

Setting SCREEN-VALUE for a selection list was improved by Adrian as part of #5992, however validation of input value as well as the actual update of the screen-value is not exactly as in 4GL:

- screen-value can be a list (using the list's delimiter), both for single/multi
- validation should validate all entries in the screen-value and fail if any entry not valid, it doesn't matter if single/multi
- when setting a 'single value' only the first item in the list that match must be selected (aka: if list had 's' and 'S' as items only first must be selected - case insensitive)
- when setting a 'multi value' for single selection list the last entry 'win', for multiple selection one item is selected for each entry - previous selection is not clear (append).

Because of the last item in the previous list the value set on SelectionListWidget - although valid - could be different from the actual screen-value of the selection-list. For single select list it will actually be just the last entry in the list, this can be handled in SelectionListWidget. For multi selection list is not that easy since the resulting screen-value could have previously selected items and even the ordering could be different - setting '3,2' as screen-value could result in '2,3' depending on the items order in the list.

Since everything seems to go through the ScreenBuffer synchronisation I would need a way to make the 'server' side (where SelectionListWidget lives) aware of the fact that the screen-value is actually different of what is was sent. This is on the 'client' side in SelectionList implementation and I don't seem to find the way to send it back :(

LogicalTerminal.refreshFrameWidgets doesn't return anything so GenericFrame.setScreenValue have no way of knowing if the value sent is different from what is actually set so it puts the initial value in the frame's screen buffer :(

I can tweak the ThinClient.refreshFrameWidgets to also return the new widget value if there is only one widget in the list (asFrameField is false) but that will also require returning that value from ThinClient.refreshFrameWidgets and then use it in GenericFrame.

If there is another, easier, way please let me know.

## #2 - 01/28/2022 07:37 AM - Hynek Cihlar

Greg Shah wrote:

From Marian:

Setting SCREEN-VALUE for a selection list was improved by Adrian as part of #5992, however validation of input value as well as the actual update of the screen-value is not exactly as in 4GL:

- screen-value can be a list (using the list's delimiter), both for single/multi
- validation should validate all entries in the screen-value and fail if any entry not valid, it doesn't matter if single/multi
- when setting a 'single value' only the first item in the list that match must be selected (aka: if list had 's' and 'S' as items only first must be selected - case insensitive)
- when setting a 'multi value' for single selection list the last entry 'win', for multiple selection one item is selected for each entry - previous selection is not clear (append).

Because of the last item in the previous list the value set on SelectionListWidget - although valid - could be different from the actual screen-value of the selection-list. For single select list it will actually be just the last entry in the list, this can be handled in SelectionListWidget. For multi selection list is not that easy since the resulting screen-value could have previously selected items and even the ordering could be different - setting '3,2' as screen-value could result in '2,3' depending on the items order in the list.

Can't you just read the widget's screen value on the server before new one is set and merge it with the new value?

## #3 - 01/28/2022 08:05 AM - Marian Edu

Hynek Cihlar wrote:

Can't you just read the widget's screen value on the server before new one is set and merge it with the new value?

The workflow as I see it is like this:

1. GenericFrame.setScreenValue gets called, validate value against data type and pass it to the widget
2. SelectionListWidget.setScreenValue - value validation, put the value in frame's screen-buffer
3. GenericFrame.setScreenValue sends over the new value through LogicalTerminal->ThinClient refreshFrameWidgets methods
4. GenericFrame.setScreenValue set the value as-is in frame's screen-buffer again and reset the status as unchanged

Now, I don't know how to get the new screen-value other than by using the FrameData and adding a return value in refreshFrameWidget. Calling widget's getScreenValue there will simply return the value already set in the frame's screen buffer and even if it would be possible to make a round-trip again to get the updated screen-value why the extra effort when we can send the updated data back directly?

The problem with this approach is that the FrameData seems to be populated with values by calling getText method of widgets which for a SelectionList is only good if not using 'item-pairs'.

Is there a reason for which getText of SelectionList returns the 'display' value instead of the actual item value like screen-value does? I'm not aware of any meaning in 4GL to get the list 'display value', only than by parsing the list-item-pairs but why anyone would do that anyway? :)

#### #4 - 01/28/2022 11:45 AM - Hynek Cihlar

Marian Edu wrote:

Hynek Cihlar wrote:

Can't you just read the widget's screen value on the server before new one is set and merge it with the new value?

The workflow as I see it is like this:

1. GenericFrame.setScreenValue gets called, validate value against data type and pass it to the widget
2. SelectListWidget.setScreenValue - value validation, put the value in frame's screen-buffer
3. GenericFrame.setScreenValue sends over the new value through LogicalTerminal->ThinClient refreshFrameWidgets methods
4. GenericFrame.setScreenValue set the value as-is in frame's screen-buffer again and reset the status as unchanged

Now, I don't know how to get the new screen-value other than by using the FrameData and adding a return value in refreshFrameWidget. Calling widget's getScreenValue there will simply return the value already set in the frame's screen buffer and even if it would be possible to make a round-trip again to get the updated screen-value why the extra effort when we can send the updated data back directly?

The idea is to keep the client a generic and thin presentation logic. All the application logic should happen on the server in the context of the current session. Considering this it seems natural to me that the screen value "merging" for multi-value selection widgets should happen on the server. The screen value is stored on the server anyway so no round trip is necessary when fetching the current widget's value.

#### #5 - 01/31/2022 03:21 AM - Marian Edu

Hynek Cihlar wrote:

The idea is to keep the client a generic and thin presentation logic. All the application logic should happen on the server in the context of the current session. Considering this it seems natural to me that the screen value "merging" for multi-value selection widgets should happen on the server. The screen value is stored on the server anyway so no round trip is necessary when fetching the current widget's value.

I see, that sounds like a great idea so will go ahead and implement it there.. it's just that the 'server' side SelectListWidget it's kinda slim (1k loc) compared to the implementation on the other end so I've got distracted but you're right.

**#6 - 02/01/2022 09:38 AM - Marian Edu**

- File *patch.6016.patch* added

Attached the patch that fixes some issues we've found when setting the screen-value for selection-list. There are some issues with combo box that are left, one interesting quirk in 4GL being that when the screen-value is set to an invalid value that is shown in the combo-box as-is (for simple/drop\_down) but when asking for it the 4GL returns the last valid value that was set :(

**#7 - 02/01/2022 11:29 AM - Greg Shah**

Hynek: Please review the patch and merge it into 3821c if you have no objections.

**#8 - 02/01/2022 04:27 PM - Hynek Cihlar**

Marian Edu wrote:

Attached the patch that fixes some issues we've found when setting the screen-value for selection-list. There are some issues with combo box that are left, one interesting quirk in 4GL being that when the screen-value is set to an invalid value that is shown in the combo-box as-is (for simple/drop\_down) but when asking for it the 4GL returns the last valid value that was set :(

The patch is good. Just one minor thing. When `SLW.setScreenValue` assembles the new value it calls the super, which again performs validity of the just-assembled screen value. Please fix this.

**#9 - 02/02/2022 02:06 AM - Marian Edu**

- File *patch.6016.patch* added

Hynek Cihlar wrote:

The patch is good. Just one minor thing. When `SLW.setScreenValue` assembles the new value it calls the super, which again performs validity of the just-assembled screen value. Please fix this.

That did also bug me, can't go around super and call super's super but this will work around the duplicated validation - overriding validation method and if valid update the value right there.

Attached the modified patch, thanks.

**#10 - 02/02/2022 04:39 AM - Hynek Cihlar**

Marian Edu wrote:

Hynek Cihlar wrote:

The patch is good. Just one minor thing. When SLW.setScreenValue assembles the new value it calls the super, which again performs validity of the just-assembled screen value. Please fix this.

That did also bug me, can't go around super and call super's super but this will work around the duplicated validation - overriding validation method and if valid update the value right there.

Attached the modified patch, thanks.

The patch is fine. I did a few cosmetic changes to it and checked it in 3821c revision 13469.

**#11 - 02/02/2022 06:50 AM - Greg Shah**

I've pushed 3821c rev 13469 to xfer.

What is left open in this task? Please set the % Done.

**#12 - 02/02/2022 06:57 AM - Marian Edu**

- % Done changed from 0 to 100

**#13 - 02/02/2022 07:01 AM - Marian Edu**

Greg Shah wrote:

I've pushed 3821c rev 13469 to xfer.

What is left open in this task? Please set the % Done.

For selection list all my tests are passing now, I saw some issues with combo/radio widgets related to set screen-value but it's probably better to make a separate task for that.. unless you think this one should be used instead.

**#14 - 02/02/2022 07:17 AM - Greg Shah**

- Assignee set to Marian Edu

- Status changed from New to Closed

A separate task makes sense.

**#15 - 02/02/2022 08:12 AM - Marian Edu**

Greg Shah wrote:

A separate task makes sense.

Will do that, however the changes did introduce some regression for combo box widgets that I need to fix.

Hynek, I trust the patch was applied on 'internal' repo, can you please push that into xref so I can make a new incremental patch and also incorporate your 'cosmetic' changes?

Thanks

**#16 - 02/02/2022 08:19 AM - Greg Shah**

I've pushed 3821c rev 13469 to xfer.

**#17 - 02/02/2022 09:47 AM - Marian Edu**

- *File 6016-20220202.patch added*

Greg Shah wrote:

I've pushed 3821c rev 13469 to xfer.

Thanks, attached the patch to fix the regression on combo-box - value to be set on screen-value needs not to be split using the delimiter and then validate each entry, this is only true for selection-list.

**#18 - 02/02/2022 10:10 AM - Hynek Cihlar**

Marian Edu wrote:

Greg Shah wrote:

I've pushed 3821c rev 13469 to xfer.

Thanks, attached the patch to fix the regression on combo-box - value to be set on screen-value needs not to be split using the delimiter and

then validate each entry, this is only true for selection-list.

Marian, does it mean the other affected widgets, BROWSE-COLUMNS, COMBO-BOX and RADIO-SET, can't take multiple values in the SCREEN-VALUE attribute?

Also how does OE/4GL handle escaping of the delimiter character? How does it behave when an item value contains a delimiter character?

#### #19 - 02/03/2022 02:02 AM - Marian Edu

Hynek Cihlar wrote:

Marian, does it mean the other affected widgets, BROWSE-COLUMNS, COMBO-BOX and RADIO-SET, can't take multiple values in the SCREEN-VALUE attribute?

Yes, the only widget that accepts multiple values when setting the screen-value is the selection list, all others reports it as invalid.

Also how does OE/4GL handle escaping of the delimiter character? How does it behave when an item value contains a delimiter character?

No escaping done, normally if one uses the AppBuilder (and no-one writes UI in vim) using the delimiter is not possible inside a value at design time. At run-time there is no possibility (at least that I know of) to escape the delimiter so for list-item there will be two entries added instead on one and for list-item-pair an error will be issued because the result will be impair :)

Beside this is the whole idea of being able to change the delimiter, to use one that can't be part of item values (labels).

On the other hand, it is possible to edit the generated code and get around it like this:

```
DEFINE VARIABLE SELECT-Default AS CHARACTER  
  VIEW-AS SELECTION-LIST MULTIPLE  
  LIST-ITEM-PAIRS "default", "D",  
                 "empty", " ",  
                 "second", "s",  
                 "SECOND", "S,D"  
  SIZE 26 BY 2.14 NO-UNDO.
```

This works, no complain at run-time and get screen-value for SECOND returns S,D, however when you try to set the same value the list will select two items if multi: default and second, or last one if single - so the delimiter is used upfront event if there might be a value with exact match.

The converted code in FWD have the same behaviour.

**#20 - 02/03/2022 04:17 AM - Hynek Cihlar**

Marian Edu wrote:

This works, no complain at run-time and get screen-value for SECOND returns S,D, however when you try to set the same value the list will select two items if multi: default and second, or last one if single - so the delimiter is used upfront event if there might be a value with exact match.

The converted code in FWD have the same behaviour.

Sounds good. I checked 6016-20220202.patch in 3821c revision 13476.

**#21 - 02/03/2022 05:14 AM - Igor Skornyakov**

Hynek Cihlar wrote:

Sounds good. I checked 6016-20220202.patch in 3821c revision 13476.

Please note that the problem described in #5983-15 still exists with this revision.

**#22 - 02/03/2022 06:19 AM - Marian Edu**

Igor Skornyakov wrote:

Please note that the problem described in #5983-15 still exists with this revision.

Yes, this is combo-box widget and there are a few remaining issues with that as well... working on it while I'm here and already have a test made in testcases.

**#23 - 02/03/2022 08:40 AM - Greg Shah**

- Related to Bug #6038: ComboBox setScreenValue improvements added

**#24 - 02/04/2022 01:03 PM - Roger Borrello**

In ControlSetEntity.setScreenValue, we've lost the ability to prevent the error message from being displayed in the change that added ControlSetEntity.validateScreenValue due to this condition being removed for protecting the ErrorManager.recordOrShowWarning:



```

&& frame != null &&
    getAttr("wasRealized", () -> frame.getFrameWidget().config().wasRealized, true))

```

### My proposed changes:

```

--- /tmp/brz-diff-g17v03gw/old/src/com/goldencode/p2j/ui/ControlSetEntity.java
+++ /home/rfb/projects/fwd/3821c/src/com/goldencode/p2j/ui/ControlSetEntity.java
@@ -2,7 +2,7 @@
** Module      : ControlSetEntity.java
** Abstract    : server side grouped control widget parent class
**
- ** Copyright (c) 2005-2021, Golden Code Development Corporation.
+ ** Copyright (c) 2005-2022, Golden Code Development Corporation.
**
** #- -I- --Date-- --JPRM-- -----Description-----
** 001 NVS 20051026   @23487 Created initial version.
@@ -97,6 +97,8 @@
**      AL2 20220124      Make all control set entities to check for screen-value validity.
**      ME  20220201      Allow for multiple values list when setting screen-value (delimiter is used).
**      ME  20220202      Do not split value using delimiter on validateScreenValue - only selection-list
+ **      RFB 20220204      setScreenValue needs to be able to prevent the error message from being displaye
d,
+ **                          thus allowing a parameter to be passed to validateScreenValue.
*/
/*
** This program is free software: you can redistribute it and/or modify
@@ -2857,7 +2859,8 @@
    {
        if (inUIStmt || !internalScreenValueUsage)
        {
-         if (value instanceof character && !validateScreenValue((character) value))
+         boolean silent = !(frame != null && getAttr("wasRealized", () -> frame.getFrameWidget().config().was
Realized, true));
+         if (value instanceof character && !validateScreenValue((character) value, silent))
            {
                return false;
            }
@@ -2876,9 +2879,25 @@
        */
        boolean validateScreenValue(character value)
        {
+         return validateScreenValue(value, false);
        }
+
+ /**
+  * Validate the value(s) for SCREEN-VALUE attribute.
+  *
+  * @param value
+  * The to be validated as SCREEN-VALUE attribute.
+  *
+  * @param silent
+  * Suppress error message when true
+  *
+  * @return true if the value is a valid screen-value.
+  */
+ boolean validateScreenValue(character value, boolean silent)
+ {
+     if (!isValidScreenValue(value))
+     {
-         if (!_isRealized())
+         if (!_isRealized() && !silent)
            {
                ErrorManager.recordOrShowWarning(4058, String.format(
                    "Attribute SCREEN-VALUE for the %s %s has an invalid value of %s", type(),

```

**#25 - 02/04/2022 01:06 PM - Roger Borrello**

Or would it be better just to place that condition within validateScreenValue, as it was before?

**#27 - 02/07/2022 09:03 AM - Roger Borrello**

Roger Borrello wrote:

In ControlSetEntity.setScreenValue, we've lost the ability to prevent the error message from being displayed in the change that added ControlSetEntity.validateScreenValue due to this condition being removed for protecting the ErrorManager.recordOrShowWarning:

My proposed changes are not necessary, due to the updates in [#6038](#) fixing the base issue.

**#28 - 02/07/2022 09:34 AM - Roger Borrello**

Roger Borrello wrote:

Roger Borrello wrote:

In ControlSetEntity.setScreenValue, we've lost the ability to prevent the error message from being displayed in the change that added ControlSetEntity.validateScreenValue due to this condition being removed for protecting the ErrorManager.recordOrShowWarning:

My proposed changes are not necessary, due to the updates in [#6038](#) fixing the base issue.

Actually the customer indicates there are other combo-boxes in which the error appears (see #6044). Perhaps we should consider the code that was removed?

```
=== modified file 'src/com/goldencode/p2j/ui/ControlSetEntity.java'
--- old/src/com/goldencode/p2j/ui/ControlSetEntity.java      2022-02-05 16:24:08 +0000
+++ new/src/com/goldencode/p2j/ui/ControlSetEntity.java      2022-02-06 14:01:29 +0000
@@ -2,7 +2,7 @@
 ** Module      : ControlSetEntity.java
 ** Abstract    : server side grouped control widget parent class
 **
- ** Copyright (c) 2005-2021, Golden Code Development Corporation.
+ ** Copyright (c) 2005-2022, Golden Code Development Corporation.
 **
 **  -#-  -I-  --Date--  --JPRM--  -----Description-----
 ** 001 NVS 20051026   @23487 Created initial version.
@@ -97,6 +97,8 @@
 **      AL2 20220124           Make all control set entities to check for screen-value validity.
 **      ME  20220201           Allow for multiple values list when setting screen-value (delimiter is used).
 **      ME  20220202           Do not split value using delimiter on validateScreenValue - only selection-list
+ **      RFB 20220204           setScreenValue needs to be able to prevent the error message from being displaye
d,
+ **                               thus allowing a parameter to be passed to validateScreenValue.
 */
 /*
 ** This program is free software: you can redistribute it and/or modify
@@ -2886,7 +2888,8 @@
 {
     if (!isValidScreenValue(value))
     {
-        if (_isRealized())
+        if (_isRealized() && (frame != null) &&
+            getAttr("wasRealized", () -> frame.getFrameWidget().config().wasRealized, true))
```

```
{
  EntityManager.recordOrShowWarning(4058, String.format(
    "Attribute SCREEN-VALUE for the %s %s has an invalid value of %s", type(),
```

## Files

---

patch.6016.patch	12.9 KB	02/01/2022	Marian Edu
patch.6016.patch	14 KB	02/02/2022	Marian Edu
6016-20220202.patch	3.67 KB	02/02/2022	Marian Edu