User Interface - Bug #8260

treelist sorting column with null and empty string values does not produce correct result

02/08/2024 09:14 AM - Șerban Bursuc

Status:	Feedback		**Start date:**	
Priority:	Normal		**Due date:**	
Assignee:	Eric Faulhaber		**% Done:**	0%
Category:			**Estimated time:**	0.00 hour
Target version:				
billable:	No		**case_num:**	
vendor_id:				
Description				

History

#2 - 02/08/2024 09:30 AM - Șerban Bursuc

- File naam_desc.png added

- File naam_asc.png added

- File ext_asc.png added

Using the TreeList testcase:

We add nodes with the following cell values

```
on choose of btnCreateNode do:
do i = 1 to 5:
...
        if i MODULO 3 = 0 then do:
         {&tree}:SetCellStringValue(inewNode, 0, "null vals").
        end.
        if i modulo 3 = 1 then do:
          {&tree}:SetCellStringValue(inewNode, 0, "empty strings").
          {&tree}:SetCellStringValue(inewNode, 1, "").
          {&tree}:SetCellStringValue(inewNode, 2, "").
          {&tree}:SetCellStringValue(inewNode, 3, "").
          {&tree}:SetCellStringValue(inewNode, 4, "").
          {&tree}:SetCellStringValue(inewNode, 5, "").
          {&tree}:SetCellStringValue(inewNode, 6, "").
          {&tree}:SetCellStringValue(inewNode, 7, "").
        end.
        if i modulo 3 = 2 then do:
          {&tree}:SetCellStringValue(inewNode, 0, "ext not null").
          {&tree}:SetCellStringValue(inewNode, 3, random(100,200)).
        end.
```

nodes fieldlist, index: key1 and index: keyid are not spawned by this trigger.

We add these nodes to the root node.

From #8093 sorting with nulls has the rule that in natural order nulls are put first and when in reverse put last. This matches 4GL behavior when all column values are not null or empty string. When this isn't the case, there is a mismatch.

Please see the pictures in #8093-39.

#3 - 02/08/2024 09:32 AM - Greg Shah

- Subject changed from fdyntree: sorting column with null and empty string values does not produce correct result to treelist sorting column with null and empty string values does not produce correct result

#4 - 02/08/2024 09:56 AM - Șerban Bursuc

- File deleted (naam_desc.png)

#5 - 02/08/2024 09:56 AM - Șerban Bursuc

- File deleted (naam_asc.png)

#6 - 02/08/2024 09:57 AM - Șerban Bursuc

- File deleted (ext_asc.png)

#7 - 04/01/2024 06:41 PM - Vladimir Tsichevski

Șerban Bursuc wrote:

> Using the TreeList testcase:
>
> We add nodes with the following cell values
>
> [...]
>
> nodes fieldlist, index: key1 and index: keyid are not spawned by this trigger.
>
> We add these nodes to the root node.
>
> From #8093 sorting with nulls has the rule that in natural order nulls are put first and when in reverse put last. This matches 4GL behavior when all column values are not null or empty string. When this isn't the case, there is a mismatch.
>
> Please see the pictures in #8093-39.

Where can I see the complete code example?

#8 - 04/02/2024 08:23 AM - Vladimir Tsichevski

- Status changed from New to Feedback

- File 5084.p added

I reproduced the situation using 5084.p, and I found no difference between OE and FWD. What is wrong with FWD?

#9 - 04/02/2024 09:28 AM - Șerban Bursuc

Vladimir Tsichevski wrote:

> I reproduced the situation using 5084.p, and I found no difference between OE and FWD. What is wrong with FWD?

I downloaded the file. Just a note, it contains customer data so you should probably remove the file.

I also can't run the test. The problem is here:

```
OCXFile = SEARCH( "C:\Users\fwd\Documents\tree.wrx":U ).
chCtrlFrame = CtrlFrame:COMPONENT-HANDLE.
chCtrlFrame:LoadControls(OCXFile, "CtrlFrame":U).
```

I get the error that I couldn't load the control frame.

I tried using a wrx from another similar testcase but it still gives me an error when loading the controls. I changed the path accordingly but it still doesn't work for some reason. Maybe I'm missing something else.

As for what's different, it's a bit more apparent from the screenshots in #8039-39, but I can see that you also have a similar test scenario as there and didn't get any issue. I think we need some common ground here - either you run the test from #8039-39 or I make 5084.p work to properly test.

#10 - 04/02/2024 09:35 AM - Vladimir Tsichevski

- File deleted (5084.p)

#11 - 04/02/2024 09:56 AM - Vladimir Tsichevski

Șerban Bursuc wrote:

> As for what's different, it's a bit more apparent from the screenshots in #8039-39

There is no such note at #8039 :-(

#12 - 04/02/2024 09:57 AM - Șerban Bursuc

Vladimir Tsichevski wrote:

> Șerban Bursuc wrote:
>
> > As for what's different, it's a bit more apparent from the screenshots in #8039-39
>
> There is no such note at #8039 :-(

Sorry, I meant #8093-39.

#13 - 04/02/2024 10:11 AM - Vladimir Tsichevski

As I see this:

1. If a string tree column has non-empty values, empty string values and unknown values, when expected ascending order is:
   1. unknowns
   2. empties
   3. non-empties

   So all nulls are less than empties.
   Issue 1: in FWD empties are **more** than unknowns.

1. If a string tree column consist of empty strings and unknowns only, when reordering does not happen (effectively unknowns and empties are all equals each other.
   Issue 2: in FWD reordering **happens** in this situation.

Correct?

#14 - 04/02/2024 10:27 AM - Șerban Bursuc

- Assignee set to Eric Faulhaber

- vendor_id deleted (GCD)

Vladimir Tsichevski wrote:

> As I see this:

1. If a string tree column has non-empty values, empty string values and unknown values, when expected ascending order is:

   1. unknowns
   2. empties
   3. non-empties

So all nulls are less than empties.
Issue 1: in FWD empties are **more** than unknowns.

1. If a string tree column consist of empty strings and unknowns only, when reordering does not happen (effectively unknowns and empties are all equals each other.
   Issue 2: in FWD reordering **happens** in this situation.

Correct?

All correct.

For issue 1, I was responsible for adding this behavior. The problem was we weren't accounting for null column values, so we simply needed a null comparator, but apparently Comparator.nullsFirst and Comparator.nullsLast doesn't have the exact same precedence as OE, so this problem occured.

As for issue 2, I don't understand whether OE just refuses to sort the column, or it doesn't do anything because empty string and null are the same and it always sorts the same.

#15 - 04/02/2024 10:30 AM - Șerban Bursuc

- Assignee changed from Eric Faulhaber to Șerban Bursuc

- vendor_id set to GCD

Sorry for the random assignment, it was **not** intentional.

#16 - 04/02/2024 05:15 PM - Vladimir Tsichevski

- vendor_id deleted (GCD)

- Assignee changed from Șerban Bursuc to Eric Faulhaber

Șerban Bursuc wrote:

> As for issue 2, I don't understand whether OE just refuses to sort the column, or it doesn't do anything because empty string and null are the same and it always sorts the same.

I was not able to find the answer quickly either. I think, we may ask customers if they find this issue significant.